IE3D™

IE3D User's Manual

Release 15.0

Contractor/manufacturer is:
Mentor Graphics Corporation
8005 S.W. Boeckman Road, Wilsonville, Oregon 97070-7777.
Telephone: 503.685.7000
Toll-Free Telephone: 800.592.2210
Website: www.mentor.com
SupportNet: supportnet.mentor.com/
Contact Your Technical Writer: supportnet.mentor.com /doc_feedback_form

Table of Contents

Chapter 1 Introduction

Electromagnetic simulation is an advanced technology to yield high accuracy analysis and design of complicated microwave and RF printed circuit, antennas, high-speed digital circuits and other electronic components. IE3D is an integrated full-wave electromagnetic simulation and optimization package for the analysis and design of 3D and planar microwave circuits, MMIC, RFIC, RFID, antennas, digital circuits and high-speed printed circuit boards (PCB). Since its formal introduction in 1993 IEEE International Microwave Symposium (IEEE IMS 1993), the IE3D has been adopted as an industrial standard in planar and 3D electromagnetic simulation. Much improvement has been achieved in the IE3D since then. The IE3D has become the most versatile, easy to use, efficient and accurate electromagnetic simulation tool.

In the recent years, we have improved IE3D significantly. From IE3D 11 to IE3D 12, we have seen much improvement on IE3D. One of the most important improvements is the introduction of Automatic Geometry to IE3D Flow (AGIF). AGIF contains the Automatic GDSII to IE3D Flow allowing users to create complicated planar and 3D structures from layouts in GDSII format. It also contains the Cadence Virtuoso to IE3D Flow for RFIC designs, and Cadence Allegro to IE3D Flow for signal integrity and power integrity applications. The AGIF flows simplify the model creation process significantly. By just one click, a user can build a complicated layout consisting of wire bonds, vias, ground and plane planes with thickness and solder balls, pin vias into some 3D models suitable for IE3D full-wave EM simulations. In IE3D 11.5, we have introduced the IE3D Full-wave Accelerated Simulation Technology Algorithm (FASTA). IE3D FASTA allows IE3D to simulate generally large structures using significantly reduced RAM and in a fraction of the time for the original IE3D simulations. IE3D FASTA preserves the full-wave feature of IE3D while it reduces the computational efforts significantly with slightly reduced accuracy.

Recent advancement in microwave, wireless, RF and semiconductor technologies require EDA imposes new challenging to circuit designers. New development requires higher accuracy and reduced design cycles. To meet this challenge, we have implemented the IE3D Full-Wave FastEM Design Kit in the IE3D V12. The FastEM Design Kit allows users to get high accuracy full-wave designs done in real-time at design time. It significantly reduces the design efforts and improves the quality of high frequency designs.

RFIC design has become a very important aspect of full-wave EM designs. On IE3D, we have implemented automatic extraction and optimization of lumped circuit equivalent circuit from IE3D simulations. The feature makes IE3D very handy for RFIC designers.

Full-wave EM tuning, optimization and synthesis require users to parameterize structures. To simplify the parameterization process and enhance the capability of parameterization, we have further improved our schematic-layout editor, or the IE3Dlibrary. We have introduced equation-based geometry modeling in IE3Dlibrary 11.x. On IE3DLibrary V12, we introduced Boolean objects and void objects to allow parameterization, full-wave EM tuning, optimization and synthesis of sophisticated planar and 3D structures on IE3Dlibrary. On IE3Dlibrary V14, we have implemented guides for easy locating objects. We have also implemented different ways of high-lighting and accessing the objects to make IE3Dlibrary easy to use.

What are the major features of IE3D V14: (1) IE3D integrated design environment: We have integrated polygon layout editor, s-parameters visualization and post-processing, current distribution, near-field and far visualization into one single package, FastEM real-time EM tuning and optimization in one package. Users can do most of the design works in one single piece of the software. (2) Implementation of advanced extraction feature allowing IE3D to cover wider frequency range from 1 Hz to 10 THz: Full-wave algorithms have numerical difficulty in handling low frequency simulations. For the older IE3D versions, we normally can go down to about 20 to 100 KHz. The simulation results will be become unstable below some low frequency limit. The limit is structure dependent. On IE3D V14, we have implemented an advanced feature allowing accurate and robust simulation of structures down to 1 Hz. (3) Shifting of reference planes for coupled ports: It has been a long time limitation on IE3D. IE3D allows accurate

simulation of high frequency structures. It also allows users to shift the reference plane of an isolated port accurately and efficiently. However, the older versions do not allow shift of reference planes for coupled and/or differential ports. This limitation makes design procedure less convenient for de-embedding of structure discontinuities. IE3D V14 has implemented accurate and automatic extraction of s-parameters with shift of reference planes for coupled and differential ports.

In IE3D V15 is the first Mentorized release. It has implemented Mentor Graphics License Scheme which is built on top of industrial standard FlexLM technology. Zeland Program Manager is renamed as IE3D Program Manager (IE3DPM). IE3D V15 also features significant speed improvement on OpenMP multi-CPU support for major processes in IE3D. In IE3D V15, we have implemented an automatic geometry connection for crossing 3D polygons. Signal integrity applications require high performance EM modeling. IE3D V15 has implemented Physical Component Compiler Library (PCCL) for automatic geometry generation and simulations of parameterized vias, solder balls and wire bonds etc. for both single-ended and differential structures. The implementation makes it easy for signal integrity designers to perform high quality EM simulations. IE3D V15 has also improved the User Defined Object in IE3DLibrary. The feature allows users use the capabilities to use any tools to parameterize structures. The parameterized structures can be FastEM prepared and optimized using IE3DLibrary. IE3D V15 also implemented wire bond profiles for creation and modeling of wire bonds with different user definable profiles.

This manual mainly serves as a tutorial manual. It demonstrates how the users can achieve the design goals thru many examples.

Before we start the actual examples, we will provide a brief introduction in the theory. For those users we do not want to know the theoretical part of the IE3D, they can skip Section 1 to Section 3 of this chapter. In fact, we also suggest those users who do not have much numerical simulation experience to delay reading the following sections until they get more knowledge from the next a few chapters.

Section 1.1 Basic Theory and Implementation.

IE3D is a full-wave EM solver. It solves the Maxwell Equations, which govern the macro electromagnetic phenomenon. There is no much assumption involved except the numerical nature of the method. Therefore, the solution is extremely accurate.

The original Maxwell's Equations are in differential form and the solutions of the equations are the electric (E) field and magnetic (H) field in the whole space. To solve an EM problem, we need to solve the E and H-fields numerically. Numerical solution of the original Maxwell Equations of E-field and H-field involves many unknowns. Instead, the IE3D solves the Maxwell's Equations in an integral form through the use of Green's functions. We try to represent the E-field and H-field as some weighted integrals of electric current on metallic structures and magnetic current derived from the electric field distribution on a metallic aperture. For most practical circuit and antenna structures, the metallic domain is limited and the solution domain of the IE3D is very limited. A typical example is a microstrip circuit. The solution domain is just the surface of the printed strip only. Its solution domain is significantly smaller than that of the original Maxwell's Equations.

Starting from the IE3D 10.1, we are able to solve the 3D finite dielectric problems, as well as electric current and magnetic current problems. For 3D dielectric problems, we are unable to obtain the Green's functions meeting the boundary conditions on the 3D finite dielectrics. We will need to mesh the 3D dielectrics and solve the equivalent current distribution inside the 3D finite dielectrics. For simplicity reason, our following discussion will focus on the formulation of electric current on metallic structures only. The magnetic current formulation and the 3D dielectric formulation are similarly obtained and we will not provide detail here.

For a general EM scattering problem, we assume a conducting structure in a stratified dielectric environment, as shown in Figure 1.1. An incident field is imposed to the structure to induce current

distribution on it. The induced current will create the secondary field to satisfy the boundary condition on the metallic structure. For a typical highly conductive structure, the induced current is flowing on the conducting surface and the boundary condition is,

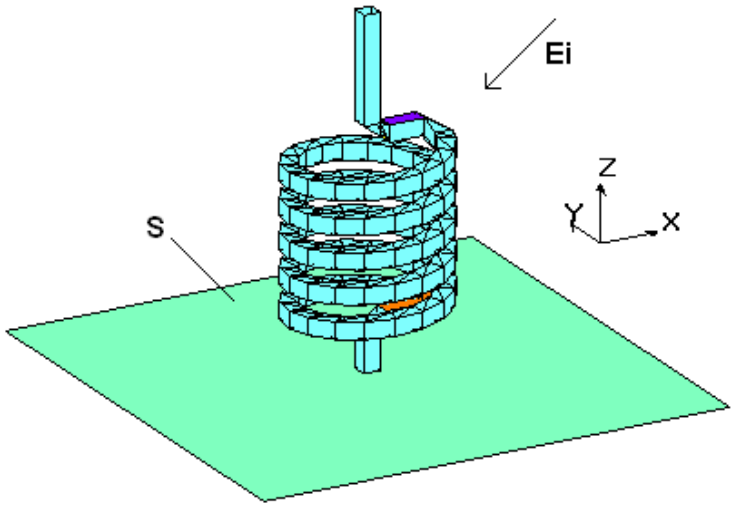


Figure 1.1 An incident field Ei is applied to a metallic structure S.

$$\boldsymbol{E}(\boldsymbol{r}) = Zs(\boldsymbol{r})\,\boldsymbol{J}(\boldsymbol{r}),\ \boldsymbol{r} \in S \tag{1}$$

Where S is the conducting surface; $\boldsymbol{E}(\boldsymbol{r})$ is the total tangential field on the surface; $\boldsymbol{J}(\boldsymbol{r})$ is the current distribution on the surface; $Zs(\boldsymbol{r})$ is the surface impedance of the conductor.

When the structure is in a layered dielectric environment, we can write down the total field as,

$$\boldsymbol{E}(\boldsymbol{r}) = \boldsymbol{E}_i(\boldsymbol{r}) + \int_S \boldsymbol{G}(\boldsymbol{r} \mid \boldsymbol{r}') \cdot \boldsymbol{J}(\boldsymbol{r}')\, ds' \tag{2}$$

Where $\boldsymbol{G}(\boldsymbol{r} \mid \boldsymbol{r}')$ is the dyadic Green's function for the dielectric environment; $\boldsymbol{E}_i(\boldsymbol{r})$ is the incident field on the conducting surface. $\boldsymbol{G}(\boldsymbol{r} \mid \boldsymbol{r}')$ satisfies the boundary conditions on the stratified dielectrics except the boundary condition on the conductor S.

Substituting (2) into (1) yields the integral equation,

$$Zs(\boldsymbol{r})\,\boldsymbol{J}(\boldsymbol{r}) = \boldsymbol{E}_i(\boldsymbol{r}) + \int_S \boldsymbol{G}(\boldsymbol{r} \mid \boldsymbol{r}') \cdot \boldsymbol{J}(\boldsymbol{r}')\, ds' \tag{3}$$

The incident field and the surface impedance are provided. The Green's function can be derived. The only unknown is the current distribution $\boldsymbol{J}(\boldsymbol{r})$.

By assuming that the current distribution is represented by a set of complete basis functions,

$$\boldsymbol{J}(\boldsymbol{r}) = \Sigma_n I_n\ \boldsymbol{B}_n(\boldsymbol{r}),\ n = 1, 2, \ldots \tag{4}$$

We obtain

$$Zs(\boldsymbol{r})\,\Sigma_n I_n\ \boldsymbol{B}_n(\boldsymbol{r}) = \boldsymbol{E}_i(\boldsymbol{r}) + \Sigma_n I_n \int_S \boldsymbol{G}(\boldsymbol{r} \mid \boldsymbol{r}') \cdot \boldsymbol{B}_n(\boldsymbol{r}')\, ds' \tag{5}$$

By taking the Galerkin's procedure, we convert (5) into a matrix equation,

$$\int_S ds\, \boldsymbol{E}_i(\boldsymbol{r}) \cdot \boldsymbol{B}_n(\boldsymbol{r}) = S_n I_n \{ \int_S ds\, Zs(\boldsymbol{r})\, \boldsymbol{B}_m(\boldsymbol{r}) \cdot \boldsymbol{B}_n(\boldsymbol{r}) - \int_S ds \int_S ds'\, \boldsymbol{B}_m(\boldsymbol{r}) \cdot \boldsymbol{G}(\boldsymbol{r} \mid \boldsymbol{r}') \cdot \boldsymbol{B}_n(\boldsymbol{r}') \},\ n = 1, 2, \ldots \quad (6)$$

The Galerkin's procedure is to enforce (5) with a complete set of test functions and the test functions are the same as the basis function. A complete set of basis functions consists of infinite number of terms. Therefore, equation (6) is an infinite dimensional problem. Equation (6) is exact when the basis functions in equation (4) are a complete set. Unfortunately, we are unable to solve equation (6) analytically except some very special structure, due to the fact it is an infinite dimensional problem. We can only get an approximated solution numerically by truncating the infinite series with finite number of terms. Mathematically, the truncation is a projection process. We project the actual solution in infinite dimensions to that of finite dimensions. If we choose the finite dimensions such that the major components of the actual solution are all in the finite dimensions, we should be able to obtain a very good approximation. After the projection, equation (6) becomes an N by N matrix equation,

$$[Z_{mn}]\,[I_m] = [V_m] \quad (7)$$

Where N is the number of the finite terms,

$$Z_{mn} = \int_S ds\, Zs(\boldsymbol{r})\, \boldsymbol{B}_m(\boldsymbol{r}) \cdot \boldsymbol{B}_n(\boldsymbol{r}) - \int_S ds \int_S ds'\, \boldsymbol{B}_m(\boldsymbol{r}) \cdot \boldsymbol{G}(\boldsymbol{r} \mid \boldsymbol{r}') \cdot \boldsymbol{B}_n(\boldsymbol{r}') \quad (8)$$

$$V_m = \int_S ds\, \boldsymbol{E}_i(\boldsymbol{r}) \cdot \boldsymbol{B}_n(\boldsymbol{r}) \quad (9)$$

The solutions of equation (7) are the coefficients of the expanded current distribution in equation (4). After the current distribution is solved, we can calculate the s-parameters, radiation patterns, RLC equivalent circuit of the structure, near field distribution and whatever other parameters of interest.

The above method is also called moment method. All moment method (or method-of-moment, MOM) formulations, no matter simple or complex, take the form of equations (7) to (9). The differences are on the choice of basis functions and the Green's functions.

There are many choices for the basis functions and the dyadic Green's function. Consideration on the basis functions and the dyadic Green's function is mainly on accurate and efficient evaluation of the double surface integrals in (8).

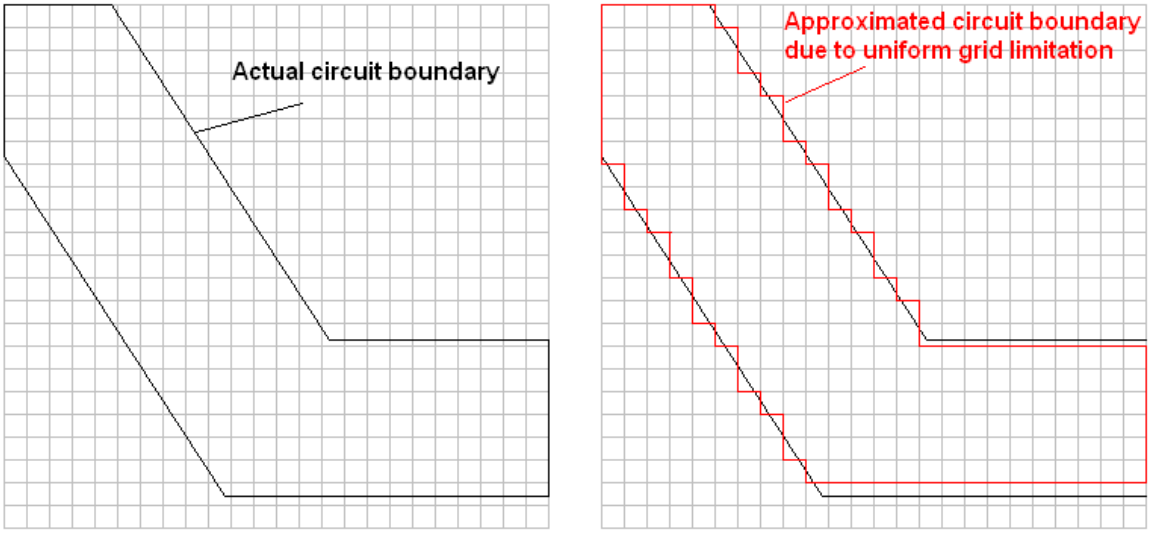
Section 1.2 Uniform Grid and Non-Uniform Meshing Basis Functions.

For general-purpose EM simulators, basis functions on a meshed structure are used. The matter is how we mesh the structure. There are two types of meshing schemes in practical applications: (1) Uniform meshing; (2) Non-uniform meshing.

Uniform meshing is simple and straightforward. It is required for those simulators using the FFT to calculate the double surface integrals in (8). For uniform grid based simulators, the layout is divided into a uniform grid. Then, a user draws a circuit as some polygons. Then, the uniform meshing scheme tries to fit the shape of a structure into the uniform grids (shown in Figure 1.2). If your structure can't be fitted completely into a uniform grid, you have two choices. One choice is to remove the portions can't be fitted and ignore them. The other choice is to make the uniform grid finer in order to get a better fit or approximation. For a planar EM simulator, making the grid twice as fine means that you are quadrupling the number of unknowns. Quadrupling the number of unknowns means that the simulation time is about 16-32 times slower and the required memory will be about 8-16 times as much.

Certainly, uniform meshing imposes the biggest accuracy and efficiency limitations on EM simulators based upon uniform grids. Uniform grid basis functions are still adopted by some other EM

simulators because of the requirement of the FFT algorithm used in calculating the double surface integrals in (8). There might be improvements to reduce the number of unknowns due to making the uniform grid finer to fit a circuit better into the uniform grid. However, the limitations imposed by uniform grids can never be removed completely. There are ways to use very fine uniform grids to approximate arbitrary shaped structures while the number of unknowns is forced not to increase so fast as regular uniform meshing. However, such enforcement will limit the flexibility of the simulators significantly. Some simulators may hide the uniformly meshed structures from users. However, you need to keep in mind that there is always approximation involved in such uniform grid based EM simulators.



(a) (b)

Figure 1.2 Uniform meshing creates large number of cells. (a) A microstrip bend structure in a uniform grid layout. (b) The bend is fitted into a uniform grid. The cells inside the fitted structure are the cells used in the actual calculation. There are about 200 cells created.

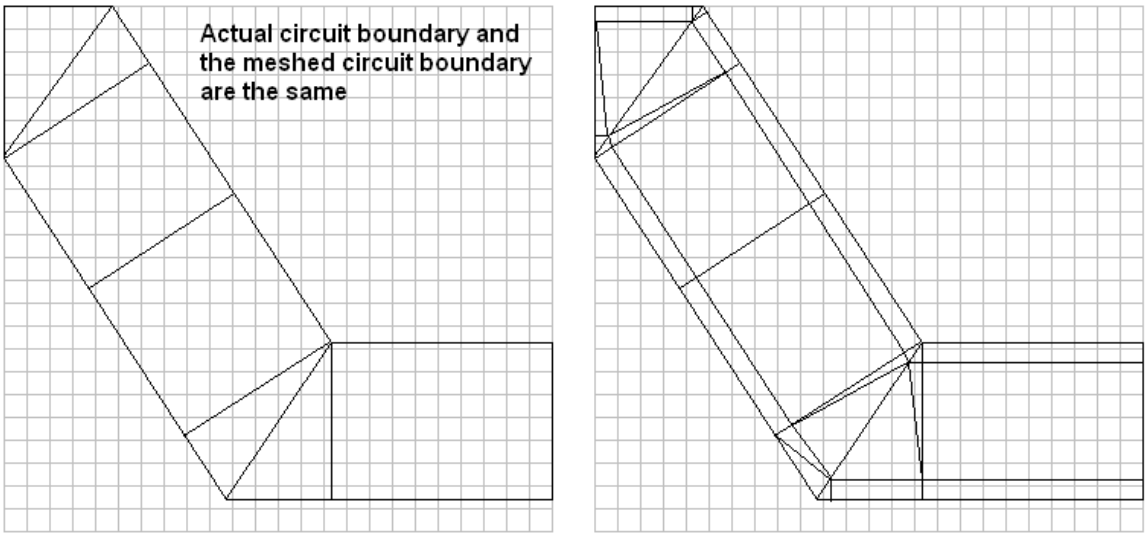


Figure 1.3 Non-uniform meshing is flexible, efficient and also accurate. (a) A coarse non-uniform meshing of the structure in Figure 1.2 (a). No approximation is made in the shape of the structure. Total 7 cells are created. (b) Small edge cells are created along the edge of the structure in order to enhance

accuracy. Total 29 cells are created. The small cells along edges of the structure are used to model the edge singularity of the current distribution.

In IE3D, we adopt a triangular and rectangular mixed meshing scheme and apply the non-uniform basis functions. As a non-uniform meshing based EM simulator, IE3D approaches a problem in a better way. A user draws a circuit as a group of arbitrarily shaped polygons on a layout editor first. Then, the simulator tries to fit a non-uniform triangular and rectangular mesh into the circuit. Shown in Figure 1.3 is the result of typical non-uniform meshing. As you can see from the comparison between Figure 1.2 and Figure 1.3, the non-uniform meshing scheme is more flexible, efficient and accurate than the uniform meshing scheme. It creates significantly fewer cells and unknowns than uniform grid based meshing. For the particular example, the non-uniform meshing in IE3D creates about 1/8 of the cells of uniform grid based meshing scheme while it matches the boundary perfectly.

Some people claimed that uniform grid based simulators are more accurate than non-uniform grid based simulators. This is actually a mis-leading concept. First of all, non-uniform meshing based simulators can approximate your actual structure much more accurate than uniform grid based simulators, since we do not need to ignore the irregular portion of the circuit. Even when a circuit can be fitted into a uniform grid, a uniform grid based simulator usually still creates more cells and therefore more unknowns in the calculation. Mathematically, more unknowns normally may mean higher accuracy. However, such a rule may not apply in the case of uniform meshing vs. non-uniform meshing. Uniform meshing yields more unknowns. However, it does not use all those unknowns in the critical region. It uses more cells everywhere. It does not concentrate on the region requiring more unknowns. On the IE3D, we can create small cells in the region requiring more unknowns and use large cells in the region requiring fewer unknowns.

Many people may have known that there is some edge singularity in the current density flowing on a microstrip line as shown in Figure 1.4. We try to use the roof-top functions to approximate the current distribution on the microstrip. A roof-top function is a ramp in the longitudinal direction and constant on the transverse direction. In a typical IE3D simulation, we use 1 to 5 cells in the transverse direction. People may wonder how we can get accurate results using just 1 to 5 cells since Figure 1.4 shows that we can not get high accuracy current density on the transverse direction even using many cells in the transverse direction. That is true for current density. In fact, no matter how many cells we use in the transverse direction, we just cannot get high accuracy for current density at the edges of a microstrip line. Fortunately, we are able to extract the circuit and antenna parameters accurately even using few cells in the transverse direction. Why is it like that?

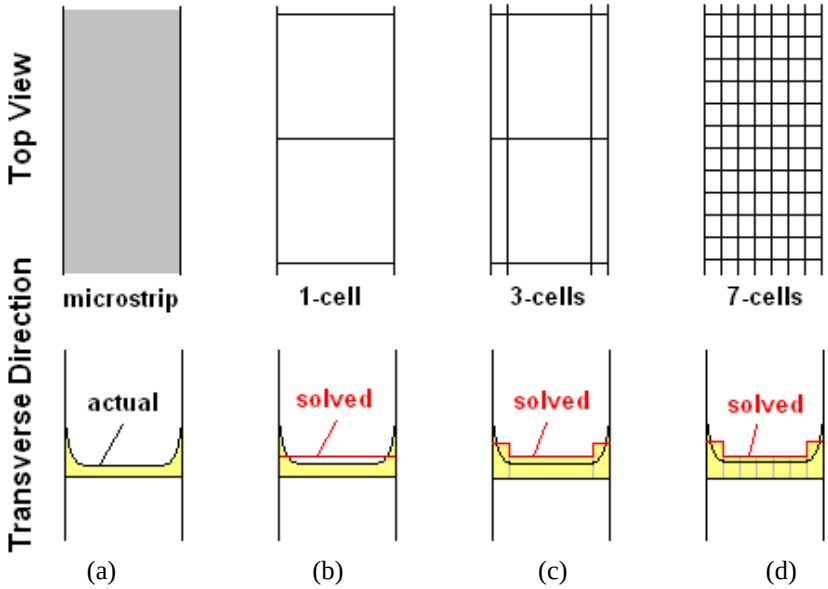


Figure 1.4 Different meshings on the cross section: edge singularity is on the actual

circuit. The 1-cell approximation takes the average of the distribution. The 3-cells approximation models the edge effect and yield high accuracy results. The 7-cells approximation does not yield much improvement over the 3-cells approximation.

For a circuit or antenna designer, the most interesting parameters are the s-parameters (or y-, z-parameters). These parameters are directly related to the total current across the transverse direction at a port. The parameter we really care about is the total current, which is the integral of current density across the transverse direction. For an antenna designer, another parameter of interest is the radiation pattern, which is also a weighted integral of current density. As you can see, the parameters of interest to both circuit and antenna designers are the integral of current density instead of current density itself.

Surprisingly, MOM codes using roof-top functions are very accurate in predicting the total current in the longitudinal direction even using 1 to 5 cells in the transverse direction. For fast simulation, IE3D uses 1 cell in the transverse direction (see Figure 1.4b) and still gets reasonable accuracy. To get higher accuracy, we use two smaller cells along the edge of the strip (see Figure 1.4c). For a uniform mesh in Figure 1.4d, the cells in the center area really do not play much role in improving the accuracy. More comparison between uniform meshing and non-uniform meshing will be available in Appendix M.

Section 1.3 Open Boundary, Close Boundary and Periodical Boundary

Traditionally, according to their solution domains, MOM EM simulators are divided into two groups: (1) Open boundary Green's function formulations; (2) Close boundary Green's function formulations.

Open boundary Green's function formulations are for the modeling of antenna structures and large layout structures without metallic enclosures. It is the exact boundary conditions for most antennas and many different RF and microwave circuits. Close boundary Green's function formulations are for microwave circuits inside enclosures. Typical examples are some microwave filters with cavities. For most of enclosed microwave circuits, the enclosures do not change the circuit properties much. Open boundary Green's function formulations normally can still be applied to yield high accuracy results. However, there are cases the metallic enclosures are critical to the performance of the microwave circuits. Typical examples are microwave filters with thick substrates. Normally, the thick substrate makes the power less concentrated on the traces. It is easier to radiate out. For a circuit not at resonance, the radiation is normally small. However, when a circuit is at resonance, much of the power can be radiated. In fact, antenna is an example of power radiation at resonance. Antenna designers make use of resonance to get the power radiated out. Filters are making use of multiple coupled resonances in order to get good performance for the pass-band and stop-band. Circuit designers make use of resonance to let power pass through a circuit with little loss at specific frequency range. They do not want the power to radiate out at the resonances. For such a case, a metallic enclosure can keep the power from radiating out. For a typical microstrip filter, the radiation is low even at resonance when the substrate is thin. However, when the substrate is too thick, the radiated power can be much higher than the transmitted power in the filter. Designers normally put a metallic enclosure to prevent the radiation. If we apply an open boundary Green's function based simulator to such a case, we may not be able to predict the performance of the thick-substrate filter very well because the open boundary Green's function formulation will predict strong radiation.

Traditionally, open boundary Green's function formulation is implemented with non-uniform meshing, and close boundary Green's formulation is implemented with uniform meshing. For this reason, microwave designers did not have good tools for flexible geometry modeling for structures inside an enclosure, even though flexible geometry modeling for open structures is available on the IE3D 7.0 and earlier versions.

The exciting thing is that close boundary Green's function formulation and non-uniform meshing are implemented into the IE3D. IE3D users are offered with the maximum flexibility and capability with the non-uniform meshing for both open boundary and close boundary and periodic boundary conditions.

Periodic boundary condition is used to model large phase arrays. A unit-cell of periodic boundary condition allows users to investigate the effects of coupling in a large phase array.

Section 1.4 IE3D Application Programs and Capability

The IE3D package consists of the seven major application programs:

MGRID -------- It is the major layout editor for construction of a structure. It allows a user to create and edit a structure as polygons and vertices. It has full control over the detail shapes and locations of geometry. Starting from V14, MGRID is renamed as IE3D EM Design System. It has integrated layout editing, s-parameters visualization and post processing, current distribution visualization, near-field and far-field post processing and visualization. It also has FastEM Design Kit for real-time full-wave EM tuning and optimization.

IE3DLIBRARY
---------- The object-oriented schematic-layout editor for parameterized geometry modeling and editing. With the introduction of FastEM Design Kit for real-time EM tuning, optimization and synthesis, parameterization becomes necessarily needed and extremely important for IE3D full-wave design. Parameterization is available on the major IE3D layout editor MGRID. However, it is limited to vertices and polygons levels. High-level parameterization can be done on IE3DLIBRARY. To make IE3DLIBRARY more flexible, we have introduced Boolean objects and void objects. The new introduction makes IE3DLIBRARY much more capable in generating sophisticated parameterized models. IE3DLIBRARY is relatively easy to use because no many commands are involved. Detailed discussion on using IE3DLIBRARY can be found from other electronic documentations.

AGIF ---------- The IE3D-SI advanced geometry modeling tool to create full-3D IE3D models directly from GDSII files, Cadence Virtuoso and Cadence Allegro.

IE3DOS --------- It is the EM simulator or simulation engine for numerical analysis. It is a DOS-style command line application. It is called in the background by the IE3D dialog to perform an EM simulation. It is normally hidden from the customers. IE3DOS supports Win32, Win64, Linux32 and Linux64. The 64-bit editions allow users to solve large structures.

MODUA ------- MODUA is the schematic editor for parameter display and nodal circuit simulation. Most of its capabilities are integrated into MGRID in V14. Mixed EM and circuit co-simulation is still the unique feature on MODUA while other s-parameter display and post processing features are integrated into MGRID.

PATTERNVIEW
---------- Post processor for radiation pattern visualization and post processing. All functionalities of PATTERNVIEW are integrated into MGRID in V14.

ADIX ---------- It is the optional ACIS/DXF/GDSII/GERBER format converter. All functionalities of ADIX are integrated into MGRID for those users choose the ADIX option.

To perform an EM simulation, a user can start from layout editor MGRID, IE3DLIBRARY or AGIF. The most fundamental one is the MGRID layout editor. On MGRID, you draw a structure as a group of polygons. After you finish constructing the structure as polygons and defining ports on it, you can invoke the simulation engine IE3D to perform an EM simulation. The simulation result is saved into a file compatible with the Agilent/EEsof Touchstone format. The saved file can be imported into other popular commercial nodal network or circuit simulators such as the ADS from Agilent/EEsof or Microwave Office from Applied Wave Research. The simulation result is also saved into the IE3D geometry file (.geo or .ie3). They can be visualized and post-processed on MGRID, MODUA, IE3DLIBRARY and AGIF of the IE3D package. MODUA is a program similar to the Agilent/EEsof Touchstone except it does not have a library with large number of elements. MODUA actually does not need such a library because any simulation result files and pre-simulated geometry files from MGRID can be used as modules in MODUA. A user can also

define lumped elements, such as R, L, C, M (mutual inductor), open circuit, short circuit and ideal connection, on MODUA to do an EM and circuit co-simulation. Before V14, MODUA is automatically invoked by IE3D to display the solved s-parameters after a simulation. Starting from V14, users can use MGRID to do visualization and post processing. The only functionality MODUA has while MGRID does not have is circuit simulation. If no EM and circuit co-simulation and optimization are involved, users even don't need MODUA on IE3D V14.

One of the major advantages of EM simulation is that the field and current distributions from a simulated structure are accessible to the users. Information on the current and field distribution in a structure can be valuable to circuit and antenna designers. On the IE3D, you can optionally save the current distribution file in a simulation. The current distribution file can be accessed on MGRID V14 while opening the geometry file. You can visualize the vector and scalar current distribution. You can also do an animation on the current distribution. You can find the radiation patterns and other parameters from the current distribution on MGRID. Finally, the radiation patterns can be visualized and post-processed on either MGRID or PATTERNVIEW. You can display the 3D patterns, 2D patterns, merge different patterns, find array radiation patterns, and find the transfer functions between the transmitting (Tx) antenna and the receiving (Rx) antenna. You can display and process the parameters of linearly polarized and circularly polarized antennas. On MGRID, you can also calculate and visualize near field distribution on the structure.

Some users may have a geometry constructed using other tools. The MGRID can import and export in GDSII and CIF formats in the standard version. The optional ADIX converter allows a user to import and export geometry in AutoCAD DXF format (for 2D or 3D), ACIS format (for 3D) and GERBER format. ADIX is fully integrated into MGRID. When the ADIX optional is enabled, MGRID is able to import and export in GDSII, CIF, DXF, ACIS and GERBER formats.

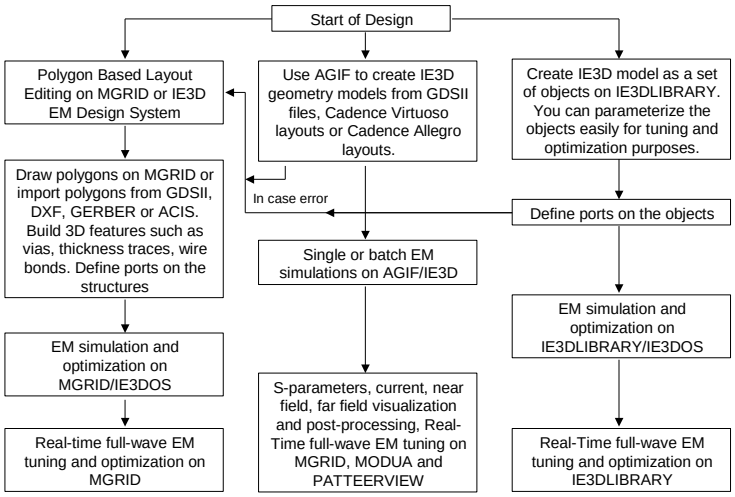


Figure 1.5 The flow chart of a basic IE3D EM simulation.

The IE3D has a great advantage over other commercial EM simulation tools no matter whether accuracy, efficiency, capability or capability is concerned. Traditionally, planar EM simulators are good for printed structures with some quasi-3D structures such as z-directed vias. Normally, vias are limited to span a small electrical distance in z-direction. Full-3D structures can only be simulated using full-3D EM simulators. IE3D can cover both planar and full-3D structures in a consistent way. For planar structures, IE3D is much faster, more accurate and more flexible than other planar EM simulators. IE3D can model full-3D metallic structures such as conical vias in PCB and helix and other general 3D structures in antennas. IE3D can also model full-3D dielectric structures such as finite substrate patch antennas and wire bonds in inhomogeneous dielectrics.

Table 1.1 shows the major capabilities of the IE3D. If you have access or knowledge on other EM simulators, you can do a comparison based upon Table 1.1. You will find that IE3D provides far more features and capabilities than other simulators. Table 1.1 also serves as a quick reference to you. If you are not sure whether a feature is available on the IE3D, at any time, you can check Table 1.1 to see whether the feature is in the list. If it is in the list, you can use the key words to find out the feature from the index or from the IE3D Help documentation.

The flow charts of a typical IE3D simulation procedure are shown in Figure 1.5. As you can see, there are multiple ways of performing an IE3D simulation on IE3D 12. Different ways provide much convenience to users with different kinds of tool sets available.

Section 1.5 Configuration of the User's Manual

We have briefly introduced the IE3D in the Chapter 1 of this manual. We will discuss the window configuration for MGRID, IE3DLIBRARY, MODUA and PATTERNVIEW in Chapter 2. The functionality of each item is briefly explained. The Chapter 3 explains the basic input techniques by illustrating the construction of a simple microstrip bend on MGRID. Chapter 4 discusses how we can ensure electrical connection between polygons in the layout editor MGRID. On IE3D, we use double precision floating point numbers to describe the vertices of polygons for best accuracy. We use many polygons to describe a circuit. Some of the polygons are connected and some are disconnected. We have easy ways to ensure polygon connections. In Chapter 4, we also discuss how we can construct 3D structures easily on the MGRID. In Chapter 5, we will discuss current distribution, pattern calculation, and full-wave electromagnetic optimization. In Chapter 6, we will discuss the procedures and the key points in importing geometry. We will explain how to accurately model closely coupled structures such as MIM capacitors. We also demonstrate how to extract lumped element equivalent circuit and enhance the accuracy of the equivalent circuit. Chapter 7 discusses mixed EM and circuit simulation and optimization including distributed and lumped elements. Chapter 8 focuses on the modeling and simulation of aperture structures using magnetic current modeling. Chapter 9 discusses the applications of the IE3D in signal integrity applications. We will discuss how to extract lumped element RLC models in SPICE format, and how to find the time transient responses of the circuits. Chapter 10 discusses the modeling of filter structures. Chapter 11 discusses modeling of wire antennas and other RF antennas. Chapter 12 focuses on accuracy and efficiency enhancement. Chapter 13 will discuss modeling of antenna arrays. The important feature "Simulate and Find Excitation" will be discussed there. Chapter 14 will discuss the modeling of open and boxed structures. Chapter 15 will focus on differential structures. Chapter 16 will focus on 3D finite dielectrics modeling. Appendices discuss some important issues that are not discussed in the formal context of the manual.

Table 1.1 Major capabilities of the IE3D Electromagnetic Simulator

FUNCTIONALITY AND CAPABILITY	GENERAL	SPECIAL CAPABILITY OR EXPLANATION
Microstrip Circuits	Yes	Multiple dielectrics, lossy and finite ground plane
Stripline Circuits	Yes	Accurate modeling of finite strip thickness
Co-planar Waveguide (CPW)	Yes	Finite thickness, lossy ground, finite or infinite ground plane
Slot-line Structures	Yes	Magnetic current modeling for infinite ground plane and electric current modeling for finite ground plane.
Suspended Stripline and Other Multi-layer Planar Circuits	Yes	No limit on the number of dielectric and metallic layers

High Speed Digital Packaging and Signal Integrity	Yes	RLCG equivalent circuit extraction in SPICE format, simulation of SPICE file in frequency domain for verification and confirmation
Printed Circuits on Lossy Silicon Substrate	Yes	IE3D's Green's functions include all the losses in the dielectrics and metals.
HTS Superconductor Circuits	Yes	Modeling of skin effect and high dielectric permittivity
Coaxial Circuits and Shielded Strip-line Circuits	Yes	modeling of any multiple conductor transmission line systems of arbitrary cross-section shape
Microstrip Antennas	Yes	Edge fed, probe-fed, proximity coupled fed and aperture coupled fed, no limitation on number of feeds and vertical pins.
Wire Antennas	Yes	Dipoles, loop antennas, cylindrical helix and conical helix antennas, quadrifilar antennas. It provides more accurate modeling than the typical wire antenna algorithms.
RF Antennas	Yes	Inverted antennas, spiral antennas and any other antennas with planar and 3D metallic structures
Plane-wave Incident and RCS Problems	Yes	Calculate monostatic and bistatic radar cross-section (RCS)
3D Capability Metallic Structures	Yes	Vertical and conical via holes, airbridges, 3D interconnect, no limitation on the shape and configuration of a 3D structure
3D Dielectric Structures	Yes	Patch antennas with finite substrate, wire bonds in inhomogeneous dielectric environment.
Arbitrarily Shaped Structures	Yes	No limitation on the shape and orientation of planar and 3D structures, meshing structure efficiently without limited by uniform grids
Open Structures	Yes	Capture all the radiation and coupling effects
Closed Structures	Yes	Electric and magnetic walls emulating enclosures. Exact boxed Green's functions are implemented in the IE3D 8.0 for precise modeling of enclosed structures.
Periodic Structures	Yes	Periodical walls to emulating periodical structures such as infinite array. Exact periodic Green's functions are implemented into the IE3D 8.0 for precise modeling of periodic structures.
Number of Ports and Port Location	No limit	Offer different de-embedding schemes for different situation: extension schemes for high accuracy, localized for highly packed circuits, differential feed for structure without an infinite ground plane. No limitation on port location and orientation.
Lumped Elements and Layout Level Simulation	Yes	Lumped elements defined in both the layout and schematic editors, s-parameter files incorporated for mixed EM and nodal simulation
Electromagnetic Optimization	Yes	Automatic adjusting the location of polygon vertices to fine tune structures
Mixed Electromagnetic and Network	Yes	The MGRID+MODUA+IE3D allow mixed

Optimization		electromagnetic and network simulation and optimization.
Back Simulation	Yes	Users are allowed to extract the effect of a geometry portion out of a simulation of a larger geometry. The extra portion is excluded from the final results.
Number of Conductor Layers	No limit	A user can define as many conductor layers as the user likes
Different Conductor Property in a Circuit	Yes	A user can define the conductor as normal conductor, HTS superconductor, or thin film resistor
Metallic Thickness Modeling	Yes	The actual geometry of a thick metallic structure can be modeled, taking into consideration of the skin effect
Number of Dielectric Layers	No limit	General formulation and implementation of Green's functions for unlimited number of dielectric layers
Complex Dielectric Constant (ε_r), Permeability (μ_r) and Conductivity (σ)	Yes	Complex ε_r, μ_r and σ available for both the dielectric layers and the metallic strips.
Frequency dependent metallic and substrate parameters	Yes	A user is allowed to define the complex ε_r, μ_r and σ as
Thin Dielectric Layers	Yes	Tested for thin dielectric layers down to 0.1 microns in MMICs.
High Dielectric Constant Material	Yes	Tested for dielectric constant up to 1000 in HTS circuits
MIM Capacitors	Yes	Optionally meshing the coupling plates into small cells for accurate modeling; aligning the meshing on both plates; automatic creation of meshed MIM capacitor with or without vias.
Spiral Inductors	Yes	Easy one-step construction of rectangular and circular spiral inductor, modeling of finite thick metal traces, modeling of air-bridges, modeling of lossy ground plane
Interactive Graphic Input of Geometry	Yes	Flexible mouse input and keyboard input of polygon vertices, strong 2D and 3D geometry checking
Convenient Geometry Editing	Yes	Copy, move, polygon and vertex elevation, automatic cutting of overlapped polygons, digging holes in geometry, polygon connectivity checking, etc.
3D Structure Display in Geometry Editing	Yes	3D display is a great help to 3D geometry editing
Automatic Generation of Geometry	Yes	One step parameterized constructions of vias, wire-bonds, circles, rings, curve-structures, spheres, fans, conical and cylindrical helix antennas, cylindrical tubes, probe-feed proximity, slots, etc.
Parameter Display	Yes	Data list, linear graph and Smith Chart display of S, Y, Z-parameters, VSWR, lumped element equivalent circuits.
Comparison of Results	Yes	Display multiple simulation and measurement

		results simultaneously
Curve-fitting and Interpolation	Yes	Curve-fitting simulation data to yield smooth result
Nodal Circuit Simulation	Yes	Connect two or more s-parameter modules or lumped elements together using idealized connection
Calculation of Port Information with Loading	Yes	The MODUA allows a user to calculate the voltage, current and waves at all the ports under different excitation and load conditions.
Equivalent Circuit Extraction	Yes	Calculate RLCG equivalent circuit for transmission line model, find the parameter values for equivalent circuit created by users
Frequency Dependent Equivalent Circuit Extraction	Yes	The MODUA allows extraction of frequency dependent equivalent circuit extraction.
3D Current Distribution Display	Yes	Display 2D vector current, 3D current density animation, 3D average current density display
3D radiation pattern display	yes	3D pattern, 3D mapped pattern, 2D pattern and 2D polar pattern for both linear and circular polarized antennas, axial ratio display, display of radiation parameters such as directivity, radiated power
Radiation Parameter Frequency Response Display	Yes	The PATTERNVIEW allows display of frequency response display of radiation parameters
Radiation Pattern Phase Display	Yes	The PATTERNVIEW allows displaying the phase of a pattern.
Radiation Pattern Comparison	Yes	The PATTERNVIEW allows comparison of radiation patterns at different frequencies and from different structures.
Radiation Patterns of Loaded Antennas	Yes	The IE3D allows users to calculate the radiation patterns of antennas with lumped elements.
General Radiation Patterns	Yes	The IE3D 10.0 allows users to calculate an N-port structure's patterns without the excitations defined. The pattern for the specified excitation can be readily obtained in the display time. It allows tuning of antenna patterns by changing the excitations only.
Pattern Optimization	Yes	The IE3D 7.0 allows optimization of radiation patterns and parameters.
Pattern Rotation	Yes	You can rotate the patterns from CURVIEW or PATTERNVIEW. This feature is very important for wireless applications because rotation of antennas is frequently encountered.
Real Ground Effect on Pattern	Yes	The CURVIEW and PATTERNVIEW allow the users to add the effects of the real ground to the pattern.
Pattern Merging	Yes	The PATTERNVIEW allows merging of radiation patterns from individual radiators. This feature allows calculation of radiation pattern from a very large structure divided into smaller sub-structures for field simulation.
3D Near Field Display	Yes	Display scalar potentials, vector potentials, E-

		fields, H-fields and Poynting vectors as curves and 3D graphs.
Save High Quality Bitmap File	Yes	Save colorful current distribution, radiation pattern or near field pictures in bitmap files.
Display Current and Field with Different Excitation and Load Conditions	Yes	Easy investigation of circular polarization, antenna with integrated source.
S-parameter files compatible with Agilent/EEsof ®	Yes	
RLCG Equivalent Circuit SPICE Compatible	Yes	RLCG segments extracted at single low frequency for low frequency applications
Frequency Independent Wide Band Equivalent Circuit Extraction in SPICE format.	Yes*	This feature is in the optional MDSPICE. The MDSPICE converts an N-port wide-band s-parameters file into a RLC network with perfectly matching s-parameters.
Frequency Dependent Equivalent Circuit Extraction and Visualization	Yes	Available on MGRID, IE3DLIBRARY and MODUA on IE3D V12.
Time-Domain Simulations on S-Parameters	Yes*	The MDSPICE simulator allows high accuracy SPICE simulation based upon the s-parameters from IE3D
GDSII, DXF, ACIS, GERBER and CIF Bi-direction Conversion	Yes*	CIF and GDSII formats are built-in. DXF, ACIS and GERBER formats are optional.
EM Tuning, Optimization and Synthesis	Yes	Users can do full-wave EM tuning, optimization and synthesis real-time at design time on MGRID and IE3DLIBRARY.

Section 1.6 Suggested Documents

This is the main IE3D manual. However, it is hard to cover every thing such a powerful EM simulator has to offer. Many different documents have been created to document and media the use of IE3D for different applications. Table 1.2 is a list of such documents and media.

Table 1.2 The different documents for IE3D. You can access the documents from the help menu or from the Help menu of any GUI programs.

Document	Description
IE3D User's Manual	IE3D User Manual
IE3DLibrary User's Manual	It is a detailed documentation for IE3DLibrary.
AGIF User's Manual	It is a detailed documentation for AGIF or IE3D-SI.
ADIX User's Manual	It is a detailed documentation for ADIX especially for GERBER import and export.
SpiralSyn User's Manual	It is for users of SpiralSyn, a module of IE3D.
MDSPICE User's Manual	It is for users of MDSPICE for time-transient analysis.
ZDS , ZDM & JobsManager Documentation	It contains the discussion on using ZDS, ZDM and JobsManager for network-based, distributed EM simulation and optimization. It also contains the discussion on using JobsManager for scheduled IE3D simulations.

Chapter 2 Basic Concepts and Window Configuration

Before going through any example, we will first explain some important concepts, and explain the basic configuration of the MGRID, IE3DLIBRARY, AGIF, MODUA and PATTERNVIEW windows.

Section 2.1 Basic Concepts

In this section, we will explain some basic concepts and terminologies frequently used in the manual. Table 2.1 documents those commonly used terminologies.

Table 2.1 The frequently used concepts and terminologies in this manual.

Terminology	Explanation
2D Layer	It is a layer with either 2D polygons or some substrate interface on it.
2D Polygon	It is a polygon with all its vertices on one single constant z-plane.
3D Layer	It is a layer with only some vertices of one or more 3D polygons on it.
3D Polygon	It is a polygon with vertices on more than 1 layer.
AEC	It is the abbreviation of Automatic Edge Cells: a feature to add small cells along the edges for high accuracy simulation accuracy.
AEC Level and Multi-Level AEC	Starting from IE3D 11, we are able to create multiple small cells along the edges for even higher accuracy simulation results. The AEC Level means the number of layers of edge cells along the edge. For normally applications such as design of a 20 dB coupler, AEC Level = 1 (one layer of edge cells) is good enough. For very high accuracy requirement such as design of a 40 dB coupler, AEC Level = 2 (two layers of edge cells) will be required.
AIF	It is the abbreviation of Adaptive Intelli-Fit: A scheme allowing users to get the frequency response of many frequency points with guaranteed accuracy by simulating just a few frequency points. It will reduce the simulation time many times. It certainly is nice to obtain fast results using data from few frequency points. However, if the scheme cannot always guarantee high accuracy, it is useless. The most important thing about the AIF in IE3D is that it always guarantees high accuracy in any case, as long as you define fine enough frequency points.
Cell	A small rectangle or triangle in a meshed structure.
Current Distribution File	The file created by IE3D for use in MGRID for the current distribution visualization, radiation pattern calculation and near field calculation and visualization. It takes .CUR as extension.
Cursor	It is the pointer of the mouse when you are using the Windows system.
De-embedding Arm	The extension on each extension port used to emulate a semi-infinite long transmission line for accurate extraction of s-parameters. It is created automatically by the simulator to stabilize the standing waves on the structure. Normally, it should not be coupled with other irrelevant structure or de-embedding arms unless it is in differential ports.
Design File	The file created and saved on the bundled circuit simulator MODUA. It takes .DSG extension.
Edge	An edge is a straight line formed by connecting two consecutive vertices of a polygon.
Electrical Connection	Two polygons are considered electrically connected if they have a common edge with 2 matched vertices. This is a very important concept and it will be discussed extensive in Chapter 4.
FastEM Design Kit	It is part of MGRID (IE3D EM Design System) and IE3Dlibrary. FastEM Design Kit allows users to parameterize a structure. Perform some pre-simulations. Then, the users can obtain full-wave EM data of the prepared

	structure in real-time at the design time. Users can perform real-time EM tuning and optimization on FastEM Design Kit.
Fmax	It is the Meshing Frequency used to mesh the structure. It is also called the Highest Application Frequency. The higher Fmax is, the finer the structure is meshed, the higher accuracy the results normally are, and the longer it takes to simulate the structure. However, the results may not be always more accurate with higher Fmax. We need to adjust the lengths of the de-embedding arms correspondingly in order to get consistent convergence. Starting from IE3D 11, we have the option to automatically adjust the lengths of the de-embedding arms with changing Fmax.
Geometry File	It is the main file created and saved on MGRID. It describes the geometry for the IE3D simulation. It takes .GEO extension.
Gridding	See Meshing
Meshing	The process to divide polygons describing a structure into rectangular and triangular cells for EM simulation.
Minimum Length	The largest length value of a grid with all the dimensions of a circuit fitting into this grid. Example: The minimum length of a rectangular strip of 2 mm by 5 mm is 1 mm.
Mouse	An input device. Two mouse buttons, left button and right button, are utilized in MGRID for selection and data input. The left button is used to select objects and menus. When it is in polygon input mode, the left button is generally used to locate a vertex. Double clicking the left button causes a set of vertices to form a polygon in the polygon input mode. The right button is used to bring up the frequently used menu items in the mode.
Ncell	The full name of Ncell is Cells per Waveguide Wavelength. It is the meshing parameter. On the IE3D, the product of Fmax and Ncell determines the meshing density. For example, Fmax = 20 and Ncell = 20 produce the same meshing as Fmax = 10 and Ncell = 40.
Node	A vertex of a polygon or a cell.
Optimization	The process of tuning the shape of a circuit automatically by the simulator to achieve an optimized circuit of required performance.
Pattern File	The file storing pattern information created by either IE3D or MGRID. It should is used by PATTERNVIEW for pattern visualization, comparison and post-processing.
Polygon	A series of vertices connected one by one to form an enclosed region. The interior of a polygon is colored on the MGRID window. A group of 2D and 3D polygons are used to describe an arbitrarily shaped structure on IE3D.
Port	A port is one or 2 terminals of a circuit. On the IE3D, a port is one terminal of the structure when there is an infinite ground plane. When there is no infinite ground plane, a user needs to define a port as a pair of positive and negative terminals. Numerical error may be introduced if we don't define a port in pair (+, -) or define a differential port (vertical localized or horizontal localized port with self-contained + and – terminals) on a structure without an infinite ground plane.
Select an Edge	There are two cases we need to select an edge on MGRID. One is for defining a port on an edge in the Port->Define Port mode of MGRID. You can click the left mouse button on the edge and the edge will be selected as a port. Another case is in Edit->Select Vertices mode when you are editing vertices and/or polygons. You can window 2 consecutive vertices of a polygon. The edge between the 2 vertices is considered to be selected after you select the 2 consecutive vertices.
Select a Polygon	In Edit->Select Polygon mode, point the cursor at a polygon of a specific layer and click the left mouse button. The polygon will turn to black color indicating it

	is being selected. Selecting a selected polygon will de-select it.
Select a Polygon by Windowing	In Edit->Select Polygon Group mode, window all the vertices of a polygon to select it. In case you window all the vertices of multiple polygons, all the polygons with all vertices windowed will be selected. The selection will focus on those checked layers in the Layer Window at the lower right corner of the MGRID window.
Select a Vertex	In Edit->Select Vertices mode, window one vertex on a specific layer to select it.
Select Edit->Copy command.	This is a frequently used phrase in the IE3D manual. It means "Select the menu item Copy in the Edit menu".
Side	It is an edge of a rectangular cell or triangular cell.
Simulate	It is the procedure to select Process->Simulate command. Enter appropriate parameters and select OK to invoke IE3D engine to perform an EM simulation on the current geometry on MGRID.
Simulation	It is the main process of IE3D to calculate the current distribution and s-parameters of a circuit or antenna.
Simulation Input File	Simulation Input File is also called .sim file on IE3D because it takes the extension of .SIM. It is a file created by the command Process->Simulate (or Process->Optimize) on MGRID or MODUA. On IE3D, all geometry information is saved in the .GEO file. However, the information about a simulation (file names, frequency points, simulation settings) is saved in the .SIM file. The simulation input file will be used as the command line variable in invoking the IE3D simulation engine.
TLN	TLN is the abbreviation for Transmission line.
Vertex	A point on a polygon boundary used in describing the polygon.
Window	Normally, "window" means a rectangle on the screen in the Microsoft Windows system. In this manual, we will also use "window" as a verb to refer to the action of windowing (see Windowing)
Windowing	It means the following action series: (1) move the mouse cursor to the upper left, press down the left mouse button, drag the mouse, while the left mouse button is still being pressed down, to lower right and release the left mouse button. The action series defines the diagonal of a window. It is used extensive in IE3D in selecting objects and defining ports.
Vertical Plane	On IE3D, the layout editing is on the top view or a constant z-plane. A vertical plane means a plane perpendicular to any constant z-plane. The normal direction of a vertical plane should have not z-component.

Section 2.2 The Configuration of IE3D's Main Layout Editor

Starting from IE3D 12, there are multiple geometry editors for the IE3D, IE3DLIBRARY and AGIF. Each has its own advantage in different applications. MGRID is the most fundamental one dealing with low-level objects such as vertices and polygons. IE3DLIBRARY is an object oriented schematic-like layout editor dealing with high level objects such as parameterized T-junction, coupled TLN, spiral inductors and filters. AGIF is also dealing with polygons as well as nets. However, it is in a script-like way for layouts from GDSII and Cadence layout editors. The commands of IE3DLIBRARY and AGIF are internally based upon the low level operations on MGRID. We will mainly focus our discussion on MGRID while we use additional documents for IE3DLIBRARY and AGIF.

Starting from V14, MGRID is renamed as IE3D EM Design System. We can do layout editing, s-parameter visualization, current distribution visualization, pattern calculation, near field calculation and visualization, lumped element extraction and real-time EM tuning and optimization all in one integrated interface. The window configuration is not completely the same for different modes. In this section, we will

mainly focus on main window configuration of MGRID in layout editing.

The basic configuration of the MGRID window is shown in Figure 2.1. Starting from V14, MGRID is an MDI interface allowing you to open multiple windows. It has one title bar, some menus, one 2D Polygon Editor with the Status window and Layer window. Multiple 3D View windows, S-parameters and Pattern Data Plot windows, Current Distribution Display windows can be opened optionally. The S-parameters plot windows and Current Distribution Display windows are available only after simulations.

Only one 2D-Polygon Editor window can be opened for an instance of MGRID. At the upper right corner of the Polygon Editor is the Status Window. In Drawing mode (or Polygon entry mode), the Status Window displays the coordinates of the mouse location (x, y, z) and the offset values to the reference point (dx, dy, dz, drho, dr, phi, theta). The reference point is the last entered vertex by default. However, the user can change the reference point by selecting Input->Set Reference Point or Input->Set Vertex as Reference Point. Also, a user can select Input->Fix Reference Point to fix it at a specific location without updating it after a new vertex is entered. The definitions of the coordinate parameters are illustrated in Figure 2.2. The child window at the lower right corner displays the polygon layers and their corresponding z-coordinates. We will refer it as Layer Window.

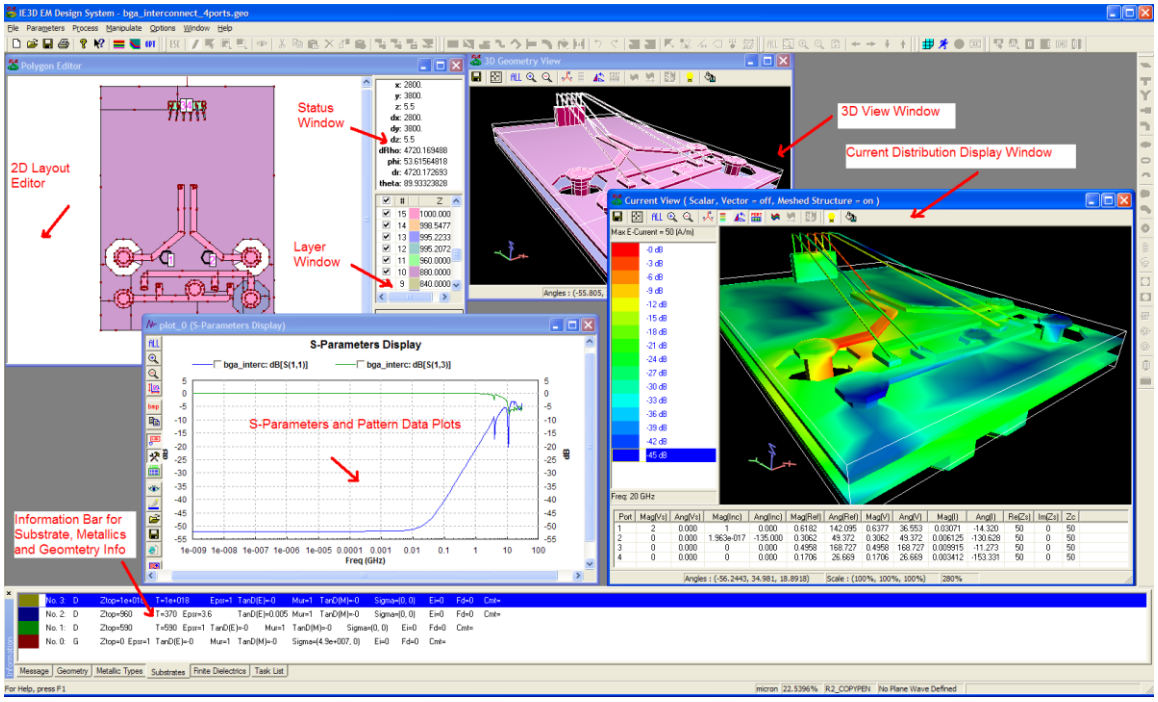


Figure 2.1 The MGRID window.

In the Layer Window, different colors are used to identify polygons on different layers. Clicking on the corresponding item in the layer window usually will enable the shift of input focus to the clicked layer. The color for each layer can be changed by selecting menu item Parameter->Layer Display Parameters. Each metallic layer has a check box. The check box for each is used for selection modes and defining port modes. There is also a check box on the title bar of the Layer Window, it is used to check or uncheck all the layers in one step. There is a combo box below the List All Layers button and above the Insert Layer button. The combo box allows the user to configure how MGRID displays polygons on different layers. The different settings are documented in Table 2.2. The Information Bar allows users to check geometry, substrates and metallic information from the tabs even though limited editing is not allowed there.

There are 10 menus for the main window: **File**, **Edit**, **Parameters**, **Input**, **Adv Edit**, **Entity**, **Port**,

Optim, **View**, **Process, Window** and **Help**. They are explained in Table 2.3 to Table 2.14. Before IE3D 9.0, there was the **Element** menu for defining lumped elements inside a structure. Since EM and circuit co-simulation can also be handled more elegantly on MODUA with lumped elements and geometry files as black box objects, we have abandoned the **Element** menu on MGRID. Users should always use MGRID to create geometry modules and use MODUA to connect s-parameters files, geometry files, and other lumped elements (R, L, C and K for mutual inductance, Short Circuit, Open Circuit etc.) together for an EM and circuit co-simulation. The MODUA way is a more consistent way because we can solve not only the s-parameters but also the current distribution and the radiation pattern of a complete circuit with structures and lumped elements connected together.

Table 2.2 The different states of the combo box for displaying layers.

State	Explanation
Elevated Layer	The current layer or high-lighted layer (with a blackened strip) on the *Layer Window* is elevated in the top view. Other layers will be displayed behind it even though they may actually above the current layer. Polygons on all layers are displayed with the proper colors to identify which layers they belong. All 3D polygons are displayed in the same color. The color of each layer is shown in the *Layer Window*.
No Other Layers	Only the polygons on the current layer are displayed with color. Polygons on other layers are not displayed. For those 3D polygons with some vertices on the current layer, they will be displayed in the 3D polygon color.
Other as Contour	The polygons on the current layer are displayed with color. Polygons on other layers are displayed as contours only.

Table 2.3 The functionality of different menus on MGRID.

Menu	Basic Function	Explanation
File	For file related operations such as Save, Open, Import and Export	This is standard for all Windows applications and we do not need detail explanation on it.
Edit	Control the modes for editing polygons and vertices which are the basic objects of a structure. The Edit menu and the Adv Edit menu are normally related to building and changing polygons describing a structure. Many of them are extremely useful.	A user can set MGRID to different modes (Draw, Select Polygon, Select Polygon Group, Select Vertices and No Mouse Entry). A user can also select some frequently used edit commands such as Copy, Paste and Move. MGRID has many advanced commands for geometry handling. The Edit menu cannot accommodate all of the commands. For this reason, we have introduced the Adv Edit (Advanced Edit) menu in the menu system. Both Edit menu and Adv Edit menu provide unparalleled geometry editing capability to users.
Parameters	This menu allows a user to change the basic parameters including the substrate layers and properties, the metallic types and the setting of the display.	Many basic parameters such as substrate parameters, metallic types and properties, finite dielectric types and properties are pre-defined and they do not need to be changed frequently. The menu items in this menu allow a user to edit these basic parameters.
Input	This mainly takes care of the drawing modes. It is mainly related to vertex entry.	This group of commands is very useful in the drawing mode. They allow a user to do very complicated geometry entry. The commands in this menu are frequently used.
Adv Edit	This menu contains the extra commands for the Edit mode. Many of them are extremely	This is the advanced editing capability for the IE3D. We have to separate many of these advanced editing commands into two menus because there are too

	useful.	many menu items for editing.
Entity	This menu contains commands for building well-defined and complicated structures in one shot. Most commands in this menu just create new polygons. They do not change existing polygons. However, some command such as Probe-Feed to Patch and Conical Vias will create polygons with guaranteed electrical connections. In order to achieve it, we may need to change some existing polygons.	On MGRID, the basic objects are vertices and polygons. In fact, complicated object containing multiple polygons is not considered as one entity on MGRID. After a user creates a structure using a command in the Entity menu, the structure is no longer an entity. They are just a group of polygons. A user cannot change the parameters of the entity because it is no longer an individual object. In case, you want to have the capability to edit an object consisting of multiple polygons, you should use the 2nd layout editor of the IE3D: the IE3Dlibrary. IE3Dlibrary is a parameterized interface. It has a list of objects. Each object may consist of a number of polygons. However, internally, we still treat an object as an entity. You can edit and change the parameters of an object any time. Both MGRID and IE3Dlibrary have their own advantage. MGRID is very flexible in handling geometry down to vertices and polygon levels, while IE3Dlibrary is very flexible in handling geometry as individual objects consisting of multiple polygons. MGRID is the default layout editor and it is very mature. IE3Dlibrary is relatively new and much work is being done to make it more powerful. Both MGRID and IE3Dlibrary should be used together to achieve the most flexibility in your design.
Port	This menu allows you to define, edit and remove ports, or plane-wave excitation. It is related to the excitation of the geometry.	There are 2 most important things in an EM simulation: (1) Create a geometry model as a group of polygons. (2) Define appropriate excitations (ports or plane-wave) to emulate the actual situation. Both are very important. In fact, defining an appropriate port is one of the most difficult tasks in a successful EM simulation. More explanation on ports will be found in this manual.
Optim	This menu allows you to define and edit optimization (or tuning) variables for full-wave EM optimizations.	IE3D is not just an EM simulator. It has extremely strong EM optimization capability. Optimization is one of the most important features of the IE3D. In the IE3D, we have implemented the Adaptive EM Optimizer which combines the advantages of local optimizer and global optimizer and other advanced technology to achieve optimized performance with the fewest EM simulations.
View	This menu is for you to adjust the views of the polygon editor window.	Starting from IE3D 14, IE3D has many different windows. The polygon editor window is the main editing window.
Process	This menu allows you to start EM simulations, optimizations. It also allows you to display the meshing. You can also invoke a MODUA for s-parameter display, another instance of MGRID for current distribution visualization, pattern	The meshing scheme on IE3D is fully automated. It is not necessary to perform a meshing on MGRID because its result will not be used in the IE3D engine. The IE3D engine accepts the geometry file containing polygons and the .sim file containing simulation parameters (such as frequency points). The display meshing function in this menu is just for

	calculation, near field calculation and visualization, a PATTERNVIEW for pattern visualization and processing.	the user to visualize the meshing. From the displayed meshing, the user may be able to control the polygons for better meshing with fewer cells.
Window	This menu is for you to open the different windows for editing and post-processing.	You can have multiple 3D views, meshing view. You can also open the current distribution, near field and pattern display windows here.
Help	This menu allows you to run access the on-line help and invoke the IE3D HelpFinder.	The IE3D HelpFinder is basically a database allowing a user to index some words or phrases and find where they are referenced in the IE3D manual. Both the on-line help and the HelpFinder are useful for you to find the appropriate topics of interests to you.

Each menu has a few menu items. A menu can be accessed by clicking the mouse on it. An alternate way is to type the underlined character of the menu while holding down the **Alt** key. For example, you can access the **File** menu by typing **F** while holding down **Alt**. This action will be referenced as **Alt+F** in the future. After a menu is opened, a menu item can be accessed by clicking the mouse on it or by typing the underlined character of the menu item without holding the **Alt** key. If a menu item has an accelerator, you can access it directly using the accelerator without opening the menu. Menu items for different menus are discussed one by one in Table 2.4 to Table 2.13 It is suggested that a user run MGRID and try to see the location of the menus and menu items.

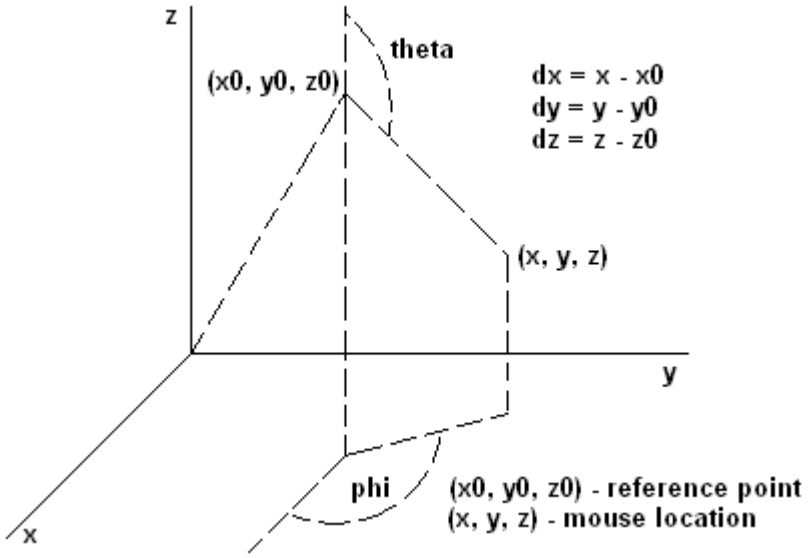


Figure 2.2 The physical meaning of the coordinates.

Table 2.4 The menu items in the File menu.

Menu Item	Explanation
New	Clear the memory and the windows for setting up a new geometry.
Open	Open a geometry file (.geo) for geometry editing by default or select the File of Type as Curview (*.cur) file for the Current Distribution File for current distribution visualization, radiation pattern calculation, near field calculation and visualization.
Save	Save the change made into file.
Save As	Save the changed geometry to a new file.
Open Current	Open the current distribution file from a simulated geometry for current

Distribution File	distribution visualization, radiation pattern calculation, near field calculation and visualization. This command is the same as Open command except the default File of Type is swapped.
Merge	Merge another geometry file into the file being edited. This feature becomes less useful after we completely implement the copy and paste command on the MGRID 9.0, since we can always open the file to be merged and copy the polygons from it and paste them into the current geometry.
Save Selected Polygons	This command allows a user to save selected polygons into a new file.
Import	Import geometry in CIF format (CalTech Intermediate Form), GDSII format (Cadence Stream Format), 3D Text format (IE3D internal script format). For the 3D Text Format, the importing will be added into the current geometry. Other format importing will clear the memory. Starting from IE3D 10.1, DXF (AutoCAD), ACIS (.SAT) and GERBER format are also available with the optional license of ADIX Format Converter (Part Number: UT080).
Export	Export geometry in CIF, GDSII and 3D Text formats, and optionally in DXF, ACIS and GERBER formats with the ADIX UT080 license enabled.
Automatic Geometry to IE3D Flow	It allows you to start AGIF from MGRID for the Automatic GDSII to IE3D Flow, IE3D Link to Cadence Virtuoso, Cadence Allegro and APD.
Save Bitmap Picture	Save the displayed top view of the geometry into a bitmap file. In fact, we can always get any displayed picture by typing Alt+PrnScn buttons to capture any window display. Alt+PrnScn is a function of MS-Window.
Print	Print the top view of the geometry in the main window.
Print Setup	Setup a printer and page for printing.
Exit	Close the MGRID application.

Table 2.5 The menu items in the Edit menu.

Menu Item	Explanation
Undo	Undo a modification. There are about 20 undo queues on the MGRID.
Redo	Redo an undo. There are about 20 undo queues on the MGRID.
Layers / Insert a Layer	This command allows a user to define a new layer or focus the input on a specific layer by specifying the z-coordinate. It corresponds to Edit->2D Input in the MGRID 9.X and earlier versions. This command is the same as the *Insert a Layer* on the *layer window*.
Layers / Change Layer Z-Coordinate	This command allows a user to change the z-coordinate of all objects (polygons and vertices) on the layer.
Layers / Grow Metal Thickness on Layer	This command allows a user to build the polygons with volumes for high accuracy modeling of strip thickness.
Layers / Select All Objects on Layers	This command allows a user to all or part of the polygons. There are options in the dialog of the command.
Layers / Change Layer Colors	This command allows a user to change the display colors of the polygon layers.
Layers / 3D Input	Set the input mode as 3-dimensional input. Mouse input is no longer available for this mode. Vertex coordinates have to be typed in using the **Input->Key In Absolute Location** and **Input->Key In Relative Location**. This feature is rarely used now, because the 2D Input can do a better job than 3D Input even for 3D construction. This mode will not be discussed in the user's manual.
Draw	Set to drawing mode. Starting from IE3D 9.3, MGRID is changed to 5 major modes: (1) Draw; (2) Select Polygon; (3) Select Polygon Group; (4) Select Vertices; (5) No Mouse Entry. If none of the selection modes and the draw mode is selected, it is No Mouse Entry mode. In any of (1) to (4) modes, if the user selects another button, MGRID will automatically shift to the selected

	mode. If the user selects the mode again, it will become No Mouse Entry mode.
Select Polygon	Set the edit mode to selecting polygons mode. The user can use the mouse to select a polygon by clicking the left mouse button on the polygon. Selecting a polygon twice will de-select it. The focus of selection is determined by the check mark of each layer on the *layer window*. On the MGRID 9.X or earlier versions, the default selection focus is always on all the layers. Starting from the MGRID 10.0, a user can configure the default selection focus, in the **Parameters->Optional Parameters**, as 1 of the 3 ways: (1) All layers; (2) The current editing layer; (3) Those checked layers before the selection of the command. To exit this mode, select **Edit->Exit Edit** or select the **Select Polygon** toolbar again. If a user selects the **Select Polygon** again, MGRID 10.0 will not resume to the default drawing mode. It will go to **No Mouse Entry** mode. If a user wants to exit **Select Polygon** mode and go into **Draw** mode, he should select **Parameters->Draw** command instead of **Parameters->Select Polygon** command. The features discussed in this paragraph also apply to the **Edit->Select Polygon Group** and **Edit->Select Vertices** commands. Starting from MGRID 9.0, a user can get into Select Polygon mode and select a polygon directly by pressing down "Shift" and click at a polygon. You can mix **Select Polygon**, **Select Polygon Group** and **Select Vertices** modes. For example, while in **Select Polygon** mode, you can directly go into **Select Vertices** mode without selecting **Exit Edit** first. All the vertices of the selected polygons become the selected vertices in **Select Vertices** mode. When you change from **Select Vertices** mode to **Select Polygon** mode, only those polygons with all vertices selected become selected in **Select Polygon** mode.
Select Polygon Group	Set to the mode for polygon group selection. To select a group of polygons, please window the polygons of interest. Windowing a polygon twice will de-select it. Starting from MGRID 9.0, a user can get into the Select Polygon Group mode and select a group of polygons directly by pressing down "Shift" and window the polygons simultaneously.
Select Vertices	Set to the mode for vertex selection. To select a group of vertices, please window the vertices of interest. Starting from MGRID 9.0, a user can get into the Select Polygon Group mode and select a group of vertices directly by pressing down "Shift" and window the polygons simultaneously. A user should understand that, when using the direct way of pressing down "shift" button and windowing a group of vertices, if the windowed vertices include all vertices of one or more polygons, MGRID will enter the Select Polygon Group mode instead of Select Vertices mode.
Select All	The menu group allows you to select all polygons, texts and/or ports of the structure.
Object Properties	Display the properties of the selected polygons or vertices. The user can edit the selected objects in this dialog. The metallic strip type can also be changed on this menu item if more than one metallic strip types are defined.
Highlighted Objects	MGRID V14 implements the highlighting of object at where the mouse is located. Users can access the highlighted object easily using the menu items in this sub-menu.
Copy	Copy the selected polygons into the clipboard.
Paste	Paste the copied polygons from the clipboard into the current geometry.
Copy and Paste	Perform the Copy and Paste in one step.

Cut	Cut the selected polygons into the clipboard.
Delete	Delete the selected polygons or vertices.
Paste from LineGauge	Allow the user to copy the polygons representing a section of transmission line (TLN) optimized using analytical formula on the LineGauge into MGRID. The substrate and metallic properties of the TLN on LineGauge will not be copied. It is the user's responsibility to make sure the properties match between MGRID and LineGauge. Otherwise, the TLN section will not behave the same as it is supposed to be. This feature is very useful when you use the IE3D as design tool. With the help of LineGauge, you can get the initial design done and optimize the structure easily on IE3D.
Change Z-coordinate	Change the z-coordinate of a group of selected vertices or polygons. If you have 2 connected polygons and you change the z-coordinate of one, you have the option to whether you want to keep their electrical connection. If you select yes, additional vertical rectangle will be created to keep the 2 polygons connected after the command.
Move Objects	Move the selected polygons or vertices. You can move the selected objects without snapping. You can move the selected objects with snapping to a vertex. You can also move the selected objects with snapping to an edge. To switch among the different snapping modes, a user can select the Change Snapping in Edit menu (or accelerator F4).
Move at an Angle	Move the selected vertex at an angle with specified radius.
Rotate	Rotate the selected polygons around the reference point.
Copy and Reflect	Copy the selected polygons by reflection. MGRID will ask you the closest distance between the copied objects to the original objects.
Copy and Rotate	Copy the selected polygons by rotation around the reference point.
Copy at an Angle	Copy the selected polygons at an angle with specified radius.
Change Snapping	Change the snapping mode in the Move Objects and Copy Objects commands.
Add Edge Vertex	Automatically insert vertices onto the edges of selected polygons to control meshing of the polygons.
Grow Metal Thickness on Polygons	Add the other three surfaces of a metallic strip on the selected polygons for accurate modeling of strip thickness.
Add Via on Edges	Build vertical polygons on selected 2D edges. Starting from MGRID 11, you have the option to make sure the end of the via will be connected to the polygons on that level if they meet. Older versions will not do that unless the edges of the polygons on the level are matching the edges of the via end.
Wire Bonding	There are two easy ways to build a wire bond: (1) You select 2 2D edges (each edge consists of 2 vertices). Then, you select Wire Bonding to build a ribbon wire strip between the two selected 2D edges; (2) You enter two points at the two locations you want to build a wire bond. Then, you select Wire Bonding. You will have the option to build a wire bond as a ribbon or as a tube. You can build multiple wire bonds based upon multiple pairs of entered vertices simultaneously.
Divide Polygon	Divide one or more polygons into more when any entered edges are across the polygons.
Exit Edit	Exit from the current editing mode and resume to the default one.
Escape	Escape from an editing mode or other modes (defining ports and optimization variables) and resume to the default edit mode.

Table 2.6 The menu items in the Parameters menu.

Menu Item	Explanation
Basic Parameters	Change the major parameters: Comment, Length Unit, Layouts and Grids, Meshing Parameters, Substrate Parameters, Metallic Strip Type Parameters,

	Enclosure Parameters and Finite Dielectric Type Parameters.
Optional Parameters	Define the optional parameters in geometry editing and simulation.
Default Editing Mode	You can setup the default editing mode. The default of default is drawing mode. However, a user can change it.
Default Selection Focus	You can setup the default selection focus mode. The default of default is that the selection focus will be at the current layer when entering the selection mode. You can configure to be all layers of interests.
Drawing and Editing Parameters	On MGRID, users can draw straight lines. You can also draw curved lines and circles. The curved lines and circles are eventually converted into line segments for the database. The drawing parameters control the precision of this conversion. You can also define the snapping distance for Copy and Move commands here.
Layer Display Parameters	You can configure the colors and transparency of each layer on the 2D and 3D views. The transparency only applies for 3D view at this time.
Substrate Display Parameters	You can changes the colors, transparency and Display Margin for substrates. The changes in this dialog will not change the physical and electrical properties of the substrates. They only affect the display. Substrates on IE3D are assumed to be infinitely extended by default. The Display Margin is to control how big the substrates are displayed on the 3D view beyond the boundary of the polygons representing the metallic strips and slots. The Display Margin applies to all substrates instead of individual one.
Metallic Type Display Parameters	You can change the pattern of the metallic strips. There is no associated color for a metallic strip. We only have associated color for a layer. The changes in this menu will not affect the physical and electric properties.
Finite Dielectric Display Parameters	You can change the color and transparency of a finite dielectric type. The changes in this menu will not affect the physical and electric properties.
Extend -X Layout	Extend the layout in -X direction.
Extend +X Layout	Extend the layout in +X direction.
Extend -Y Layout	Extend the layout in -Y direction.
Extend +Y Layout	Extend the layout in +Y direction.

Table 2.7 The menu items in the Input menu.

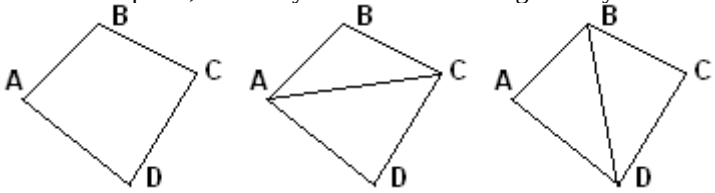
Menu Item	Explanation
Key In Absolute Location	Create a vertex by entering the absolute location.
Key In Relative Location	Enter a vertex with an offset to the reference point. The default reference point is the last vertex. You can change it using the **Set Reference Point** in **Input** menu, or fix it using the **Fix Reference Point** in **Input** menu.
Drawing/Draw Line	Set to draw a line segment in input mode. If you want to draw an orthogonal line, you should press down SHIFT in drawing.
Drawing/Draw Arc	Set to draw an arc in input mode. The arc will eventually be converted to shorter line segments. The conversion is controlled by the parameters in Parameters->Drawing and Editing Parameters.
Drawing/Draw Rectangle	Set to draw a rectangle in input mode. If you want to draw a square, you should press down SHIFT and draw the mouse.
Drawing/Draw Circle	Set to draw a circle in input mode. The 1st vertex will be the center.
Slanted Edge	Enter a vertex to form a slanted edge with specified length and angle. The slanted edge is from the reference point to the entered vertex.
Slanted Edge with Connection	There are two cases: (1) You want to build an edge at a fixed angle and you want it to be connected to an edge of an existing polygon at that angle; (2) You want to build an edge with a fixed length (and approximate angle). You want the entered edge to be connected to an edge of an existing polygon.

Rotate around Reference	Enter a vertex by a rotation around a reference.
Orthogonal Extension	Enter a vertex to form an edge orthogonal to the last edge.
Connect to Edge Perpendicularly	Enable MGRID to connect the next vertex to the closest edge and make sure the entered edge is perpendicular to the closest edge. This feature is extremely important to make sure the polygons exactly rectangular shape. Rectangular shaped structures will have advantage in meshing and simulation when efficiency and accuracy are concerned. A user can use this command together with the **Edit->Divide Polygon** to make arbitrary shaped polygons more rectangular shape.
Form Rectangle	Add the fourth vertex to form a rectangle. The condition is that there are three vertices at a right angle. In case the 3 existing vertices do not form a right angle, MGRID will prompt you to form a parallelogram.
Form Polygon	Connect the beginning vertex and the last vertex to form a polygon.
Define Text Object	Starting from IE3D V14, you are allowed to define text objects. Text objects are for display and documentation purpose only. They will not affect the results.
Grid Off	Toggle the grid between on and off.
Set to Closest Vertex	Set the next vertex to the location of the closest vertex. The default is the closest vertex on the same layer. You can type **F4** to shift to the closest 3D vertex or the (x, y)-coordinates of the closest 3D vertex. This state will be released only after the Set to Closest Grid is selected.
Set Parallel to Edge	Set the entered edge parallel to an existing edge.
Set Vertical to Edge	Set the entered edge vertical to an existing edge.
Insert Mid Point	Insert a vertex at the middle of an entered edge. It is not limited to the center of the edge. It can be anywhere except the two ends.
Create and Edit Vertices	Allow a user to edit the entered vertices and import vertices from or export vertices into a file. Together with the **Build Wire Path** in **Adv Edit**, it is very useful for building wire antennas and complicated geometry. It is a very powerful editing feature.
Drop Last Vertex	Drop the last vertex of an unfinished polygon. It is equivalent to clicking the right mouse button.
Drop All Vertices	Drop all the vertices on an unfinished polygon.
Drop Other Than Last Vertex	Drop some intermediate vertex.
Set Reference Point	Set the reference point to other location than the last vertex. In the default mode, the last entered vertex is always the reference point.
Fix Reference Point	Fix the reference point at a specified location. This item is useful when used together with Rotate Around Reference to draw curved structures.
Set Moving Reference Vertex	When we select some objects and copy (or move) those selected objects, we may need to know the locations of the selected objects and the location of the copied (or moved) objects. For this purpose, we need to have a vertex in the selected objects as the reference vertex. We use the reference vertex to locate the copied (or moved) objects. When we first enter the selection mode, MGRID will automatically find one vertex as reference vertex. We can use this command to change the reference vertex. This command is in the pop-up menu when we are in selection modes and click the right mouse button. It is better to use it on the pop-up menu. What we need to do is to point the mouse at the vertex we want and click the right mouse button to bring up the pop-up menu. Then, select the Set Moving Reference Vertex command.
Shift Moving Reference by Vertex	This command allows the user to shift the reference vertex to the next one in the order of the database. It is a fast way to shift reference vertex when it is on

	the same polygon of the current reference vertex.
Shift Moving Reference by Polygon	This command allows the user to shift the reference vertex to the 1st vertex of the next polygon in the database. It is a fast way to shift reference vertex from one polygon to anther one.
Edit Saved Point	Allow users to save some dimension information into the memory for easy use in the editing. For example, you can save some offset values into the Saved Point. Then, you can get it by a mouse click when you enter a relative location using the Key In Relative Location in Input menu.
Info on Last Entry	Display the coordinate of the two vertices and the total length of the entered edges. This feature is useful for measuring the dimensions of a structure.

Table 2.8 The menu items in the Adv Edit menu.

Menu Item	Explanation
Scale and Transformation/Change Dimension Scale	Change the scale of the geometry automatically. This feature can be useful in building and sizing structures. For example, we only need to build a sphere once. Spheres of other dimensions can be created through re-sizing the original one. Ellipsoids can also be built by sizing the original sphere.
Scale and Transformation/Flip Coordinates	Flip the coordinate systems. It may make geometry editing for 3D structures simpler.
Scale and Transformation/Shift Structure	You are allowed to shift the entire structure by providing dx, dy and dz.
Connection / Adjust Geometry for Connection	Adjust close vertices to merge them together. If a vertex is close to an edge, MGRID will move it onto the edge and insert a matching vertex on the edge to make sure the electrical connection. Starting from IE3D 14, this command will perform geometry cleaning for both 2D and 3D polygons. If you want to make sure 3D overlapped polygons are connected, you should call this command. By default, IE3D will not perform cleaning of 3D overlapped polygons. This is the only command to do 3D overlapped polygons cleaning.
Connection / Check Connection	Check and display all the polygons electrically connected to the selected polygons.
Connection / Fit Selected Objects into Grid	Allow a user to fit the selected objects' vertices into a sub-grid of a fraction of the size of the current layout grids. This feature is very useful in clean those polygons with vertices slightly off regular shape. IE3D uses floating point to describe polygons' vertices. It will be better that the polygons are off rectangular shapes, and they will be meshed into most rectangles. For those irregular polygons, MGRID will mesh them into triangles. However, it is possible that some polygons are slightly off rectangular shapes. Such a situation may cause the meshing program to create many more triangular cells than necessary. We can use this command to clean the polygons so that those slightly off rectangular shaped polygons will become rectangular.
Select Connected / Select Connected Polygons	This command allows the user to select those polygons connected to the selected polygons (or vertices).
Select Connected / Select All Vertices of Polygons with Some Vertices Selected	This command allows the user to select all the vertices of the polygons with some vertices selected.
Select Connected / Select Polygons with Some Vertices Selected	This command allows the user to select those polygons with some of their vertices selected.

Select Connected / Select Polygons with All Vertices Selected	This command allows the user to select those polygons with all their vertices selected. Those polygons with some but not all vertices selected will not be selected.
Alignment / Align Last Vertex	Align the last vertex of the entered edge to be in 0 degree or 90 degree direction. For example, if you want to make an edge at 0 or 90 degrees, you can use Set To Closest Vertex in Input menu and enter two matching vertices to the edge. Then, you select Align Last Vertex. The location of the last vertex will be adjusted to make it 0 or 90 degrees.
Alignment / Align Vertices	Align a series of vertices on the existing polygons. The edges will be aligned in 0, 45 or 90 degrees.
Alignment / Align Selected Polygon Vertices	Align the vertices of a selected polygon so that all the edges are in 0, 45 or 90 degrees.
Overlapped, Multi-Looped and Twisted Polygons/Check Polygon Overlapping	Check and display overlapped polygons. It also allows the user to select those overlapped polygons.
Overlapped, Multi-Looped and Twisted Polygons/Cut Overlapped Polygons	Cut the overlapped portions of the polygons so that they have common edges for electrical connections.
Overlapped, Multi-Looped and Twisted Polygons/Check Twisted / Multi-Looped Polygons Overlapped, Multi-Looped and Twisted Polygons/Divide Multi-Looped Polygons	A twisted polygon is a polygon with all its vertices not on a flat plane. It does not define a unique shape. An example is a polygon with 4 vertices A, B, C and D. We can divide the 4 vertices polygon into 2 triangles in 2 different ways: (1) triangle ABC and triangle CDA; (2) triangle ABD and triangle BCD. If the vertices A, B, C and D are on a flat plane, both ways are describing the same geometry except the meshing may be different. If the vertices A, B, C and D are not on a flat plane, the 2 ways describe different geometry.  This command tries to break any twisted polygon into flat polygons in one of the multiple ways. It also tries to clean multi-looped polygons so that its vertices are in the right order.
Mesh and Merge/Mesh Selected Polygons	Sub-divide selected polygons into small rectangles and triangles. This feature is used to control the meshing of some polygons. For example, we want to mesh the coupled plates in an MIM capacitor in a fine scale and the other non-coupled polygons in a normal scale. We can select the coupled plates and mesh them finer first.
Mesh and Merge/Mesh Selected Polygons On Distance	You can merge a group of polygons and try to merge them into one or multiple bigger polygons with the criteria of distance provided.
Mesh and Merge/Merge Selected Vertices	You can merge selected vertices into one by relocating them.
Remove Inserted Vertices	Remove the unnecessary vertices. Unnecessary vertices are those vertices on a straight edge and they are not required for connections.
Separate Polygons	This is a more powerful version of the Edit->Divide Polygon command.
Create Edge Cells	Allow a user to break the polygons into polygons for edge cells and interior

	polygons. This is a manual operation to control the meshing for high accuracy results. The process is fully automated in the simulator. Normally, a user does not need to do it manually unless for some special study.
Align Polygon Divisions and Meshing	Closely coupled polygons require aligned meshing. Finite dielectric blocks and the printed strips on them also need aligned meshing between them. Otherwise, you may not be able to get very accurate results. This command allows a user to align those polygons and finite dielectrics. The process is fully automated in the engine. However, we provide it here so that you can do special study on it.
Offset Polygon	You can reduce the size of some polygons by a margin.
Rectanglization	Try to sub-divide polygons of irregular shape into more rectangular shape. The feature will help editing and meshing of a geometry.
Dig Rectangular Hole	Dig a rectangular hole in a 2D polygon by specifying the center location and size.
Build Hole from Selected Polygons	This is a very useful and powerful command. It allows a user to build multiple holes of the shape of selected polygons into some other 2D polygons on layers. It also allows a user to build multiple vias connecting multiple layers. There are different options for this command. This is one of the most frequently used commands in geometry editing on IE3D. There will be much discussion on it in the context.
Build Multilayer Vias	This is a powerful command implemented into the MGRID 9.0 to ease the construction of multiple layer vias on PCBs. It allows a user to specify the locations and build the vias with different kinds of connections (pure connections, connections with via pads and connections with antipads).
Cut into Polygon on Edge	Create a cut on a polygon along a selected edge. The cut will create a notch on the edge of the polygon.
Build Via Connection on Edges	This is also a very useful command. It is very similar to the Edit->Add Via on Edges command. Both of them allow a user to create some vertical polygons between the layer of the selected edges and the specified end z-coordinate. The command also allows a user to create a vertical localized port on it. This command is very useful in defining ports on structures with finite ground planes.
Advanced Boolean Operations	This command allows you to do some advanced Boolean operations. For the Difference command, you need to select the 1st group polygons. Then, you select Adv Edit->Advanced Boolean Operations->Difference command. Immediately, you select the 2nd group of polygons and hit enter to perform the Difference command.
Find Shapes from Selected Polygons	Find the contours of the selected polygons and build them as polygons.
Build Path	Build a 2D horizontal path along the entered vertices of the same z-coordinate with specified width. The lines connecting the entered vertices will be the centerline of the path.
Build Wire Path	Build a 3D wire path along the entered vertices in the 3D space with specified radius. This menu item is a more general case of the Adv Edit->Build Path command. The command also allows users to optionally define ports at the junctions. This command is very useful in modeling wire antennas with ports or lumped elements between segments.
Continue Straight Path	This feature is a very powerful command. It can handle multiple cases. If one edge is selected, a user is allowed to build a continuing path with different widths at the start and end. The user is also allowed to enter a few vertices for a continuing path with multiple bends. If multiple edges on a vertical plane, such as the end of a horizontal tube, are selected, the user is allowed to continue the path with a specified length.

Continue Path Bend	This feature is also a very powerful command. It can also handle multiple cases. If one edge is selected, a user is allowed to build an abrupt bend junction with 0 radius provided. If a radius is provided, the bend will be a smooth one. The same rule applies for the cases with multiple edges on a vertical plane selected.
Build Connecting Path	Build a polygon to connect between two selected edges.
Build Connecting Paths Between 2 Layers	Build polygons to connect multiple matching edge pairs between 2 layers. All the edges must be on one of the two layers. Each pair of edges should have one on edge on one layer and the other edge on the other layer. The edge pair should have about the same x and y-coordinates.
Glue Polygons on Selected Edge Groups	You can select 2 edge groups with each edge group on a group of connected polygons. Then, you can glue the 2 groups of polygons together. The requirement is that each of the selected edge group should be on a vertical plane.
Remove Chamfered Corners	Replace small chamfered corners with right angle corners. Structures imported from some other tools may contain small chamfered corners. These small chamfered corners will have little effect on a circuit's performance. However, they may cause IE3D to put fine meshing at the corners for much increased simulation time without improving the accuracy. We would like to replace them with right angle corners so that the simulation will be more efficient.
Remove Redundant Vertices on Curvature	Some imported geometry has many vertices on curvatures. The vertices will force the meshing program to create many redundant cells which slow down the simulation significantly without improving the accuracy. In fact, the long and tiny redundant cells may create numerical singularities in simulation. This command helps us to reduce the number of vertices on curvatures so that we can reduce the simulation time significantly.
Convert Polygons in Shape	This command allows a user to select a group of polygons and convert the polygons into rectangles or circles with default size for each polygon or some specified size.
Convert Flat Polygons to Cylindrical Shape	This command allows a user to convert some flat polygons into cylindrical shapes. It will allow us to edit cylindrical polygons on a horizontal flat plane.
Define Dielectrics Calls	This command allows a user to define the selected polygons and extrude them for finite dielectric blocks.
Edit Dielectrics Calls	This command allows a user to edit the defined dielectric blocks.

Table 2.9 The menu items in the Entity menu (only parts of the menu items are listed. More discussion can be found in Appendix I).

Menu Item	Explanation
Rectangular Spiral	This command allows a user to create a rectangular spiral of specified dimensions.
Octagonal Spiral	This command allows a user to create an octagonal spiral of specified dimensions.
Circular Spiral	This command allows a user to create a circular spiral of specified dimensions.
Interdigital Capacitor	This command allows a user to create an interdigital capacitor with specified parameters.
MIM Capacitor	This command allows a user to create an MIM capacitor of specified parameters.
Probe-Feed to Patch	This command allows a user to build a probe with a vertical localized port defined on it. It is extremely useful in modeling probe-fed patch antennas and other small antennas with probe-feed scheme.
Conical vias	This command allows a use to build some conical via(s) between 2 specified layers.

Solder Balls	This command allows a user to build some solder ball(s) of specified size automatically.

Table 2.10 The menu items in the Port menu.

Menu Item	Explanation
Define Port	Set to define port mode. The user can use the mouse to select one or several edges of a polygon as a port. MGRID will not resume to previous edit mode until the user select **Port->Exit Port** command.
Port for Edge Group	Set to define a group of edges for a port. The user can use the mouse to window a group of open edges on a horizontal plane for a port for most of the port schemes. Starting from IE3D 11, a user can define a port with horizontal localized scheme using this command. For such a case, a user should window a group of horizontal rectangles for the port.
Selected Rectangles for Horizontal Localized Port	This command allows a user to define a group of selected rectangles for a horizontal localized port. Starting from IE3D 11, a user can define a horizontal localized port using the Port->Port for Edge Group command.
Selected Edge for Extension Ports	The alternative way to define an extension port. For the Port->Define Port and Port->Port For Edge Group commands, you select the edges after selecting the command. For this command, you select the edges first. Then, you select the command to define an extension port.
Selected Rectangles for Vertical Localized Port	The alternative way to define a vertical localized port. You select the rectangles first. Then, you select this command to define the selected rectangles for a Vertical Localized port.
Selected Rectangles for Horizontal Localized Port	The alternative way to define a horizontal localized port. You select the rectangles first. Then, you select this command to define the selected rectangles for a Horizontal Localized port.
Define Horizontal Localized Port Between Points	You can enter 2 points to across a gap between 2 polygons. Then, you build a horizontal localized port across the gap. It is a fast way to build a Horizontal Localized port.
Build Vertical Localized Port by Polygon	You can select a polygon on a layer and use its shape to build a probe of the polygon's cross-section with a V-localized port on the probe.
Ground Port Conversion	Convert all the individual negative ports into one common ground port serving as the reference to all the positive ports. This command is not used frequently. From our experience, using negative port(s) as the reference terminals for the corresponding positive port(s) is a better scheme.
Define Negative Port	This command sets MGRID to a mode so that the next defined port as a negative port for the last defined positive port. This feature allows a user to define differential feed structures. After defining a negative port, MGRID resumes to the regular port mode for defining positive ports.
Multiple Positive Port	This command allows a user to define some ports with the same positive index. When multiple ports are defined with the same positive index, identical potential will be enforced for these ports with the same positive index.
Define Reference Plane	Immediately after defining an extension port (or a Localized for MMIC port), you can define a reference plane shift to the port. The default reference plane will be where you define the port. You can use this feature to define the reference plane to some other location than the default reference plane. Reference plane shift for coupled ports may introduce error. Please use the Back Simulation concept in MODUA to perform reference plane shift for coupled port structures.
Drop Last Port	Drop the last defined port in the Port->Define Port or Port->Port for Edge Group modes. This item is useful for port definition correction without exiting

	the modes.
Ports Properties	This used to be the Port->Change Scheme command. It is renamed to Port->Ports Properties in the IE3D 11. This command allows a user to change the current port scheme in the Port->Define Port and Port->Port for Edge Group modes. This command also allows a user to change the port properties of a port.
Change Port Order	This command allows a user to change a port's index etc. It is a very useful command for differential ports. For example, you can define a series ports without worrying about the negative ports in the beginning. After you finish defining the ports, you can use this command to change some ports to the negative ports of some other indices.
Delete Port	Set the editor to delete port mode. A user can use the mouse to select a port to be deleted. MGRID automatically resumes to the default mode after a port is deleted.
Delete All Ports	Delete all the ports in one shot.
Delete Port Range	Delete a number of ports in a range.
Plane Wave Excitation	Define a plane wave excitation from some specified incident angles. Starting from IE3D 11, this command becomes a toggle between defining and un-defining plane wave excitation. The Port->Remove Plane Wave command is replaced and removed.
Exit Port	Exit the Port->Define Port or Port->Port for Edge Group mode.

Table 2.11 The menu items in the Optim menu.

Menu Item	Explanation
Variable for Selected Objects	Define the selected vertices or the vertices of the selected polygons as an optimization variable. The user will be prompted for the direction and bounds for the variable. Starting from IE3D 9, this command and the next command are interchangeable.
Add Selected Objects to Variable	Add the selected vertices or the vertices of selected polygon as a call to an existing optimization variable. The user will be prompted for the change direction and tuning rate. This command is useful to define a variable to control the locations of different vertices in different directions and/or rates.
Change Variables and Calls	Allow a user to access and edit all defined variables and calls.
Check Variables Bounds	This command allows a user to check whether the bounds of the variables are correctly defined. If the bounds of some variables are not defined correctly, the structure may become overlapped at some state with specified values for the optimization variables. For earlier versions of IE3D, the optimization will stop whenever overlapped polygons are detected. On the IE3D 11, a user has an option not to stop the optimization even overlapped polygons are detected. Instead, IE3D will try to merge the overlapped polygons. Of course, the best way is to avoid overlapping because the accidental overlapping of polygons may not be what the user wants. For this reason, we implemented this command to allow a user to run some random tests to make sure the bounds are correctly defined without overlapping. After the tests, IE3D will find the maximum ranges of the bounds without overlapping. A user has the option to use the found bounds. Due to the random feature of the command, the found bounds may not guarantee the geometry will be free of overlapping. However, the chance will be very low if you test the bounds with a large number of test points.
Display Trends	Display the low bound and high bound of optimization variables and how the geometry is changing with the variables.

Geometry Tuning	You can tune the geometry by changing the variable values and see how the variables are controlling the shape of the structure.
Exit Optim	Exit from the mode of defining optimization variables.

Table 2.12 The menu items in the View menu.

Menu Item	Explanation
Zoom 25% to Zoom 400%	Zoom in/out the structure to 25% to 400%.
Whole Structure	Display the whole structure on the window.
Zoom	Allow the user to Zoom portion of a circuit by windowing it.
Zoom In	Zoom in the structure. The user can use "+" to access this menu item.
Zoom Out	Zoom out the structure. The user can use "-" to access this menu item.
Refresh	Refresh the display and it will clean the display created by the meshing process.
Device Context Draw Mode	Change the display mode. The user can configure the window to display the layers differently.
Show Metal Type Index	Toggle between showing and not showing metal type of each polygon.
Show Metal Type Pattern	Toggle between showing and not showing the patterns of the metal type of each polygon.
Show Disconnected Vertices	Toggle between showing and not showing disconnected vertices. Any vertex of a polygon not common to another polygon is considered as a disconnected vertex.
Show Ground Connection	Toggle between showing and not showing polygon connection to ground. By default, any 3D polygon with an edge connected to the ground interface is considered grounded. However, it is difficult to tell from the top view. When this menu item is checked, the ground connection will be shown with a thick red line.
Show Port Extension	Toggle between showing and not showing the port extensions. By default, the port extensions are displayed with gray lines. Users should try avoiding the overlapping of the port extensions with the structure for accurate modeling.
Keep Meshed View	Toggle between keeping and not keeping the display of the meshed geometry. By default, the meshed structure is displayed once every time you mesh it. You can check this item to keep it until your structure is modified.

Table 2.13 The menu items in the Process menu.

Menu Item	Explanation
Display Meshing	Perform the meshing of the structure into small rectangular and triangular cells and displays the result on the window. This result is only for checking only. It is not necessary for a simulation. The circuit will be meshed again in the simulation engine when it is being simulated. The final mesh may be different due to different options.
Simulate	Setup an electromagnetic simulation of the current geometry. It will perform a checking on the geometry before and after you enter all he necessary parameters such as simulation frequency points.
Optimize	Setup an electromagnetic optimization on the structure. Different optimization schemes are available. (1) Powell Optimizer; (2) Genetic Optimizer; (3) Random Optimizer; (4) Adaptive EM Optimizer. The Powell Optimizer is a local optimizer. It is good for simple goals with monotonic error function and the starting point. The Genetic Optimizer and Random Optimizer are global optimizers. They are more robust for error functions of unknown properties. However, they are normally slower. Genetic Optimizer is normally better than

	Random Optimizer. The Adaptive EM Optimizer is a global optimizer because it is the mixture of the above 3 optimizers. It adaptively applies different optimizers and other advanced technology to achieve goals much faster and robustly with less iteration. Please understand that optimization is a mathematically unsolved problem. For a general optimization problem, you even do not know whether you will have an answer or how many possible answers you will have. Before you start an optimization, you need to make sure you are close enough and it is likely you will get an answer. Please also make sure the parameters you are optimizing are sensitive to the variables you defined. You also need to define the right bounds. Wider bounds certainly can more likely avoid going out of the bounds. However, wider bounds may give the optimizer too much freedom to go to other locations and it may take a longer time to converge.
S-Parameters and Lumped Equivalent Circuit	This is a new menu item on MGRID 12. IE3D 12 saves the simulation results into the geometry file. You can visualize the s-parameters and extract the lumped element equivalent circuit directly on MGRID using this menu.
Full-Wave EM Design Using FastEM Design Kit	It allows you to perform real-time EM tuning, optimization and synthesis on pre-simulated structure. This feature makes full-wave EM design simple and easy.
Display S-Parameters	Invoke MODUA to display the s-parameter of the current simulated geometry or other simulated structures.
Display Current Distribution	Invoke another instance of MGRID to display the current distribution of the current simulated geometry or other simulated structures.
Display Radiation Pattern	Invoke PATTERNVIEW to display the radiation patterns of the current simulated geometry or other simulated structures.
Display Log File	Invoke Notepad to display the .log file for the current simulated geometry.
Merge S-Parameter Files	Merge multiple s-parameter files of different frequency points into one file.
Merge Current Distribution Files	Merge multiple current distribution files of different frequency points from the same structure into one file.
Merge Radiation Pattern Files	Merge multiple radiation pattern files of different frequency points from the same structure into one file.

Table 2.14 The menu items in the Window and Help menu.

Menu Item	Explanation
Polygon Editor	Open the polygon editor or the main window of MGRID.
3D Geometry Display	Open a 3D geometry display window.
3D Meshing Display	Open a 3D geometry display window for the pre-simulation meshing result.
3D Current Distribution Display	Open a 3D window for current distribution visualization.
3D Near Field Distribution Display	Open a 3D window for near field visualization.
2D Near Field Distribution Display	Allow you to define some graphs to visualize the near field quantities.
3D Radiation Pattern Display	Open a 3D window for 3D pattern visualization.
2D Radiation Pattern	Allow you to define some graphs to visualize the far field quantities.
Radiation Pattern Properties	Allow you to manipulate pattern data in a list. You can add more pattern files into the list or remove them from the list. You can access the properties of pattern data from the list.
Directivity Vs.	Allow you to display other pattern or radiation properties in plots.

Frequency Display and other similar items	
Customize Toolbar	Allow you to customize the tool bars.
Status Bar	Open or close the status bar.
Information Bar	Open or close the information bar showing the geometry information.
Cascade	You can cascade the windows of the MDI system.
Tile	You can tile the windows of the MDI system.
Arrange Icons	You can arrange the icons of the minimized windows.
About MGRID	Display information about copyright and the version number of the MGRID.

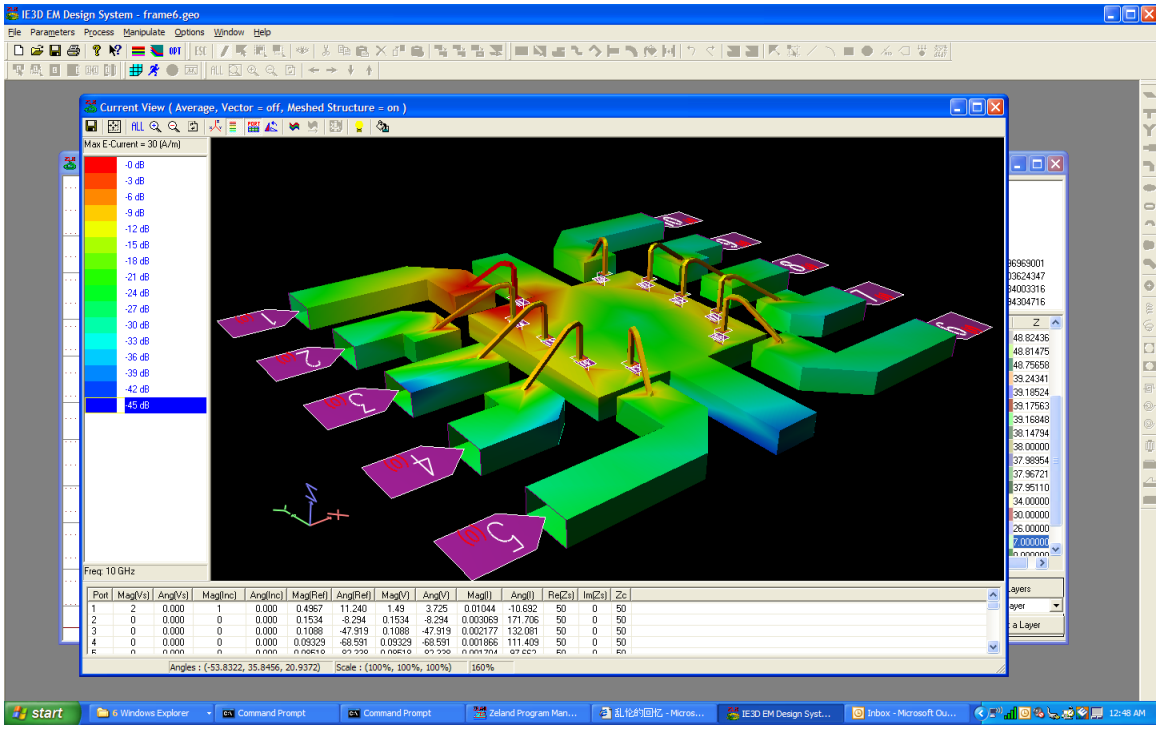


Figure 2.3 The window configuration of MGRID as the post processor for current and field.

Besides the menu items, there are some bitmap pictures in the toolbar. Each bitmap corresponds to a frequently used command, which may correspond to a menu item. The toolbar is located between the menus and the main window. Also, for the 3D view, there is a View menu. It is discussed in the next section.

Section 2.3 The Menu System of MGRID in Current, Near Field and Pattern Visualization

Current distribution display and radiation pattern calculation used to be on the CURVIEW application program before IE3D 9.0. On the IE3D 9.0, the current distribution display and pattern calculation features are improved significantly. They are also integrated into the MGRID for better integrity. The CURVIEW was phased out after version 9.0. Starting from IE3D 14, pattern visualization is also transferred from PATTERNVIEW to MGRID. There is no difference in editing mode and post-processing mode for MGRID. You are able to access all data for editing, s-parameters, current, near field and pattern visualization in one unified interface. When you open a geometry (.geo) file, you will be able to access the current distribution (.cur) data, near field (.fld) data and pattern (.mpa and .pat) data. In case, you have removed the geometry file, you can still open the current distribution file by selecting File->Open Current Distribution File command. You will be able to open the .cur file and rebuild the .geo file from the

.cur file. When a current distribution display window or near field display window is high-lighted. MGRID uses a different menu system. Most menus are the same as the main one while there are two menu groups documented in Table 2.15 to Table 2.17.

Table 2.15 The new items in the File menu of MGRID in post-processing.

Menu Item	Explanation
Save Current Density Data	Save the current density on the metallic structure with the specified excitations. The data is in an ASCII format and easy to understand. The original .cur file data is not supposed to be processed by the users. The .cur file data also does not have specified excitations.
	Other menu items are the same as the main MGRID menu.

Table 2.16 The menu items in the Manipulate menu of MGRID in post-processing.

Menu Item	Explanation
Set Background Color	Set the background color of the display window.
Transformation	Allow you to change the view angles and the display aspect ratios for the X, Y and Z-direction. For some structures with many layers close to each other, we can increase the Z-Scale Factor to see the details beside the layers.
	Other menu items are the self-explanatory.

Table 2.17 The menu items in the Options menu of MGRID in post-processing.

Menu Item	Explanation
Set Graph Parameters	Change the settings of the display.
Show Color Palette	Toggle between showing and not showing the color palette on the left of the 3D view window.
Perspective View	Choose to view the display in the perspective view.
Orthographic View	Choose to view the display in the orthographic view.
Set Light Source	Toggle between setting and not setting the spot light sources.
Show XYZ Axes	Toggle between showing and not showing the XYZ axes of the structure.
Structure Display Options	The sub-menu allows you to change how you display the structure, and which components of the structures you want to display.
Show Meshed Structure	Toggle between whether you want to show the meshed structure in pattern and near field displays.
Show 3D Average Current	Choose whether you want to show the 3D average current in the display. It is exclusive with the Show 3D Scalar Current option.
Show 3D Scalar Current	Choose whether you want to show the 3D scalar current at a specific phase of a cycle in the display. It is exclusive with the Show 3D Average Current.
Show 3D Vector Current	Choose whether you want to show the 3D vector current in the display.
Animation	Allow you to start and suspend an animation of the current flowing on the structure by showing a series of 3D scalar current pictures at continuously changing phase.
Next Frame	Go to the next frame for a paused animation.
3D Field Display Options	Setting the options for 3D near field display.
Show 3D Average Field	Choose whether you want to show the 3D average field in the display. It is exclusive with the Show 3D Scalar Field option.
Show 3D Scalar Field	Choose whether you want to show the 3D scalar field at a specific phase of a cycle in the display. It is exclusive with the Show 3D Average Field.
Show 3D Vector Field	Choose whether you want to show the 3D vector field in the display.
3D Pattern Display Options	Setting the options for 3D pattern display.

Show Pattern	
Show Vector Current	Choose to show the vector current on the structure in pattern display.
Previous Frequency	Go to the previous frequency for current, near field or pattern display.
Next Frequency	Go to the next frequency for current, near field or pattern display.

Section 2.4 The Configuration of MODUA's Window.

As we have mentioned, MODUA is basically a nodal circuit simulator. It also serves as the graphic display tool for the IE3D simulation result. It accepts circuit elements as black boxes, which are called modules.

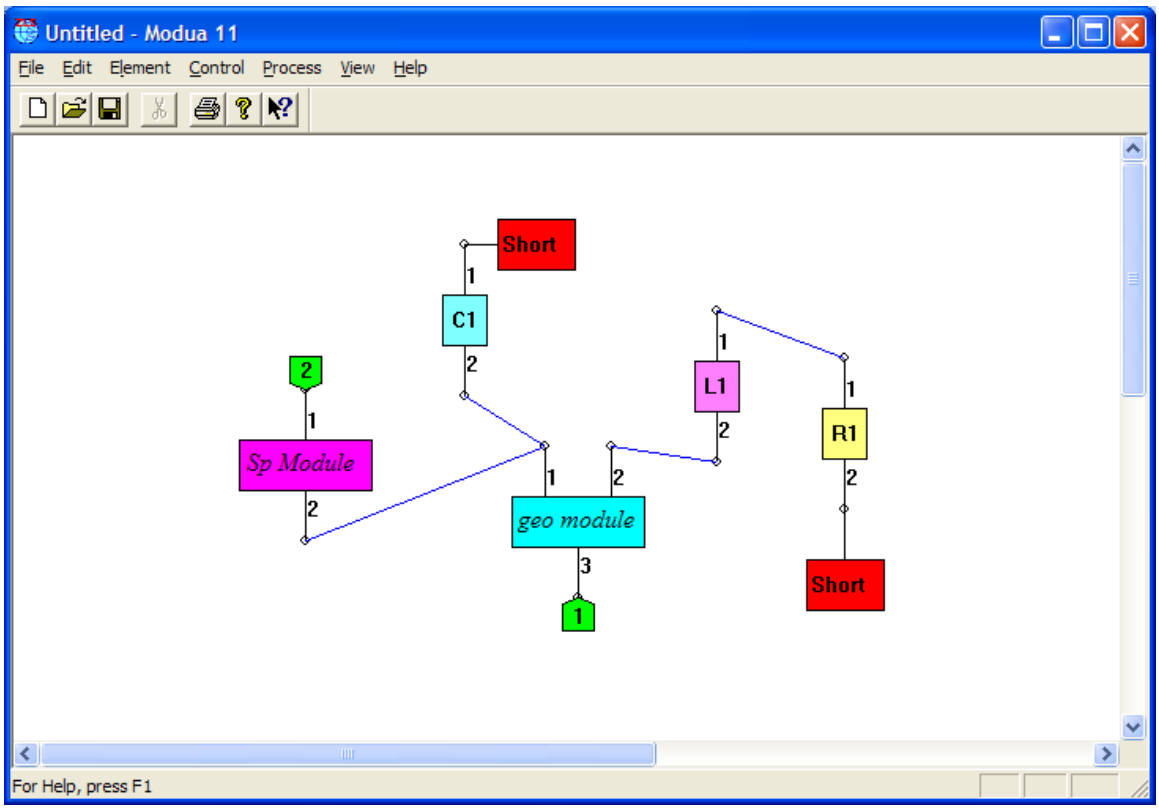


Figure 2.4 The MODUA window.

There are 10 module types in MODUA: geometry file, parameter file, resistor, capacitor, inductor, mutual inductor, port, connection, short circuit and open circuit. The 10 module types are sub-divided into two groups: independent modules and dependent modules. Geometry files, parameter files, resistor, capacitor and inductor are independent modules. They can be defined anywhere at any time. Port, mutual inductor, connection, short circuit and open circuit are dependent modules. At the time the dependent modules are defined, they must be connected to the terminals of the independent modules. When an independent module is deleted, the dependent modules connected to this independent module will also be deleted.

In some sense, MODUA is a circuit simulator similar to the old Touchstone simulator from Agilent/EEsof. However, MODUA does not have a big library. Since it is implemented for the post processing for IE3D, it really does not need a big library. It accepts geometry modules from MGRID. Therefore, you can build any structure elements on MGRID and import them into MODUA.

The basic configuration of an MODUA window is shown in Figure 2.4. It consists of seven menu

items: **File**, **Edit**, **Element**, **Control**, **Process, View** and **Help**. They are documented in Table 2.18 to Table 2.23.

Table 2.18 The important menu items in the File menu of MODUA.

Menu Item	Explanation
Add Geometry Module	Add an IE3D geometry file (.geo) as a module on MODUA.
Add Parameter Module	Add an IE3D s-parameter file (in Touchstone format) as a module on MODUA. Original Touchstone file put port information in the file extension. It is not consistent because different port number creates a file with different file extension. For this reason, we use extension .sp for IE3D's s-parameters. Also, we put the number of port into the .sp file as a comment. An example is a line "! Nport = 2" in a 2-port s-parameters. MODUA will detect the number of ports = 2 when this file is imported into MODUA.
Display Parameter Module	MODUA saves and reads files with .dsg extension. It carries the information how the elements are connected. The .dsg file does not carry the information of the s-parameters. However, we implement the feature "Display Parameter Module" here to allow a user to setup the .dsg file automatically from the .sp file. When a user selects this command, MODUA automatically create the .dsg project for the only .sp file module and get the data displayed assuming the .sp file is the only module for the .dsg file. This command is the simple way to display an existing s-parameter file.
Parameter File Queue	This feature allows a user to add parameter files into the queue for simultaneous display and comparison of different s-parameters.
Save S-Parameters into Queue	Save the current simulated s-parameters into a file and put the file into the display queue automatically
Save S-Parameters	Save the current simulated s-parameters into a file.
Open Matrix File	Open a file containing the RLC matrix elements (an IE3D specific format). For transmission line structure, RLC topology is in matrix forms. It is easier to manipulate the matrix than the SPICE netlist. A user can use the IE3D to model a small portion of a package structure and extract the RLC matrices. They can use the local RLC matrices to expand it into an overall RLC matrix using a spreadsheet application. Then, he can import the RLC matrices back to the IE3D and convert it into SPICE file.
Save Matrix File	Save the transmission line RLC equivalent circuit into an IE3D specific matrix file.
Save SPICE File	Save the transmission line RLC equivalent circuit into a SPICE net list file.
Import SPICE File	Import a SPICE netlist file (containing R, L, C and K only) into the current design. This feature is very important on checking whether an equivalent circuit is working fine. For example, you use the RLC equivalent circuit feature in IE3D to extract the SPICE circuit. You do not know how good the equivalent circuit is. You can import it back to MODUA and do a simulation. Then, you can compare the simulation result of the equivalent circuit to the original s-parameters to find out the valid frequency range of the equivalent circuit. You can even fine-tune the SPICE model using the Process->Match Queue command.
Export SPICE File	Export the current design (R, L, C, mutual inductor and short circuit only) into a SPICE net list file.
Merge Matrix Files	This is a useful command for modeling large signal integrity (SI) structure. For example, we may have a lead frame with tens or hundreds of leads. If we break it into smaller size to simulate, the result may not be accurate for some leads on the edge of the size. The Merge Matrix File allows a user to merge the

	extracted RLC circuits of the smaller sections in matrix form to get the complex RLC circuit in matrix form for the whole structure.
Save Excitation	Save the result of the **Process->Simulate and Find Excitation** command. The saved file can be imported into MGRID in post processing mode to find the current and radiation patterns of a structure with lumped elements connected.
Save TLN Information	Save the transmission line parameters into a text file.

Table 2.19 The menu items in the Edit menu of MODUA.

Menu Item	Explanation
Select Module	Set the input mode to select module mode. In select module mode, you can move the module by dragging it. You can use other menu items to control the modules. It is the default mode on MODUA.
Module Properties	Display the module properties of the selected module. You can replace the selected R, L or C element with a different R, L or C element.
Rotate Module	Rotate the direction of the module or the ports of the selected module.
Fix Module	Fix the dropped module at the current location.
Delete Module	Delete the selected modules.
Set Selected RLCM for Optim	Set the values of the selected R, L, C and mutual inductor elements as optimization variables.
Remove Optim from Selected RLCM	Remove optimization variables from the selected R, L, C and mutual inductor elements.
Delete Dependent Modules	Delete all the dependent modules (connection, short circuit, open circuit, mutual inductor) of the design.
Set all RLCM as Optimization Variables	Set the values of all the R, L, C and mutual inductor as optimization variables.
Remove RLCM Optimization Variables	Set the values of all the R, L, C and mutual inductor as optimization variables.
Exit Status	Exit the edit mode.

Table 2.20 The menu items in the Element menu of MODUA.

Menu Item	Explanation
Port	Create a port module. You need to connect the Port module to a terminal of an independent module. After you connect a port to a terminal of an independent module, you are still in the Element->Port mode for another port. In case you want to exit, you can select Element->Exit Element or click an empty spot and confirm the exit.
Define All Ports	Create one or more port modules on the spare terminals of all independent modules. A spare terminal is a terminal without any connection.
Connection	Create a Connection module. You need to connect the two terminals of the Connection module to two different terminals of independent modules. When you use a Connection module to connect two different terminals of independent modules, the two connected terminals will be of the same potential. After you finish connecting the terminals, MODUA will still be in the Element->Connection mode for a new Connection element. You can select Element->Exit Element or click at an empty spot and confirm the exit.
Short Circuit	Create an idealized short circuit. The Short module needs to connect to a terminal of an independent module. After connection, MODUA will still be in Element->Short Circuit mode. Please understand that Short Circuit does not necessarily mean 0 potential on MODUA. If it is a single ended system, a Short circuit element does mean 0 potential. However, for differential system, a Short

	circuit alone only means the return path of the element. When it is used in a differential system, please be careful not to share a Short circuit element between multiple independent terminals. If you use Connection elements to connect multiple independent terminals of differential elements together, it may result in nonphysical results. In fact, you need to use all dependent elements (Short, Open, Connection and Mutual Inductor) very carefully in a differential system.
Open Circuit	Create an idealized open circuit. The Open module needs to connect to a terminal of an independent module. MODUA will remain in Element->Open Circuit mode for a new Open circuit element after you define one Open circuit element.
Termination	Allow a user to build a resistor with one terminal short-circuited in one step. In fact, it is not an element of MODUA but a single step to create two elements (a Resistor and a Short circuit).
Resistor	Create an idealized resistor. You can define the value of the resistor as an optimization variable.
Capacitor	Create an idealized capacitor. You can define the value of the capacitor as an optimization variable.
Inductor	Create an idealized inductor. You can define the value of the capacitor as an optimization variable.
Mutual Inductor	Generate a mutual inductor. You can define the value of the mutual inductor as an optimization variable. A Mutual Inductor module is a dependent module. It must be defined between two inductors. Each pair of inductor can have only one single Mutual Inductor module. The Mutual Inductor takes value range in (-1, 1). You can control the sign of the mutual inductor by connecting to different terminals. Connecting the mutual inductor at the port 1 of each inductor is the same as connecting the mutual inductor at the port 2 of each inductor. When you connect the mutual inductor at the port 1 of one inductor and port 2 of the other inductor, it means that the value of the mutual inductor will have a sign change in the calculation.
Exit Element	Exit from the defining element mode.

Table 2.21 The menu items in the Control menu of MODUA.

Menu Item	Explanation
Frequencies	Allow a user to define frequency points for a simulation. This command is redundant because this dialog is called for almost every process in the current MODUA.
Define Display Data	Define the parameter items for listing.
Define Display Graph	Define the parameter items for graph display.
Define Display Smith Chart	Define the parameter items for Smith Chart display.
Terminating Impedance	Allow the users to change the termination or normalization impedance of each port. By default, the normalization impedance is 50-ohms.
Change Excitation	In displaying the result of the **Process->Simulate and Find Excitation** command, you can change the excitation and terminations of the ports. This feature is good for finding out the excitation and loading of antenna arrays or antennas with lumped elements, for current visualization and pattern calculation.
Display Toggle	Toggle between displaying the design topology and displaying the s-parameters.

Table 2.22 The menu items in the Process menu of MODUA.

Menu Item	Explanation
Simulate	Perform a nodal simulation on the circuit topology of the design.
Update Results	Check the .sp files and update its data to the MODUA simulation. This feature is very useful in monitoring an on-going simulation on the IE3D. You can select Display on the IE3D simulator dialog to display the results of an unfinished simulation. Then, you can use the Process->Update Results command on MODUA to update the more finished data points from time to time while IE3D is still running the simulation.
Simulate and Find Excitation	Perform a simulation on the design. After the simulation, it will display the power, voltage, current and load at each port. You can then select **Control->Change Excitation** command to adjust the excitation and terminations. The result can be saved into a file by selecting **File->Save Excitation** command. The saved excitation file (.ect) can then be imported into MGRID for displaying current distribution and calculating and radiation pattern of a structure with lumped elements.
Batch Simulate and Find Excitation	Allow a set of simulations on a design with an s-parameter file replaced for each simulation to yield a set of .ect files.
Find S-Parameters from Pi-Network	Allow a user to find the s-parameters of a frequency dependent Pi-network created using the Process->Pi-Network Equivalent command on MODUA.
Find S-Parameters from One-Port Network	Allow a user to find the s-parameters of a frequency dependent 1-port circuit created using the Process->1-Port Equiv. Ckt command on MODUA.
Optimize	Perform an optimization on the design. Please refer to the same menu item on MGRID for more information.
L-Equivalent	Perform a simulation on the current design and solve for the L-matrix. L-equivalent is good for structure with dominant inductive effect. It is not as accurate as the LC-equivalent command discussed next. However, it reduces the number of elements.
C-Equivalent	Perform a simulation on the current design and solve for the C-matrix. C-equivalent is good for structure with dominant capacitive effect. It is not as accurate as the LC-equivalent command discussed next. However, it reduces the number of elements.
LC - Equivalent	Perform a simulation on the current design and solve for the LC-matrix of a design. It is more accurate then the L-Equivalent or the C-Equivalent commands. The results are not exact while it is frequency independent and compatible with SPICE if it is extracted at one single frequency.
General Lumped Equivalent Circuit	This is a new feature implemented in IE3D 12. After the implementation, the Pi-Network Equivalent and 1-Port Equiv Ckt commands become obsolete.
Pi-Network Equivalent	Find the Pi-network equivalent circuit from a 2-port s-parameter file. The results are frequency dependent and exact. However, the Pi-network is frequency dependent and it is not compatible with SPICE.
1-Port Equiv. Ckt	Find the 1-port equivalent circuit from a 1-port s-parameter file. The results are frequency dependent and exact. However, they are not compatible with SPICE.
Match Queue File	This feature is useful in solving the frequency independent RLC equivalent circuit of specified topology of a design. The LC-Equivalent command is good only for coupled transmission line model. The model always assumes shunt C and R to ground and series L and R between input and output. They cannot be applied to any other structure such as series L and C. It is not realistic to build in different models for different structures. The Process->Match Queue File command allows a user to construct a topology based upon the elements on MODUA. MODUA will try to change the values of the RLCM elements to match the resulting s-parameters to the one saved in the Parameter File Queue.

	For example, you have a complicated structure that cannot be fitted into the transmission line model. You simulate the structure to get the s-parameter file. You put the file into then Parameter File Queue. Then, you build the RLCM model on MODUA. You set the appropriate RLCM modules as optimization variables. You define the appropriate frequency points and select the Process->Match Queue File command. MODUA will try to fine tune the optimization variables and match the performance of the design to the parameter file in the queue. If it is successful, the RLCM topology will be a good SPICE representation to the parameter file in the queue. Sometimes, it may take a few runs before you can get a good result. The Match Queue File command is also good for fine-tuning the result from LC-equivalent command. A user should understand that the LC-equivalent circuit is extracted at one single frequency point only. It works in a frequency range around where the extraction is performed. If you want to make it working in a wide frequency range, you can fine-tune the values of the LC-equivalent circuit based upon the s-parameter results. You may even need to change the LC-equivalent circuit and try to match the s-parameters in the queue for better result.
Back Simulation	This feature makes the IE3D de-embedding even more flexible. One of its most important applications is in shifting reference plane for an IE3D simulation. Shift reference plane is available on MGRID/IE3D for isolated ports. Whenever you encounter a shift of reference plane, the IE3D will try to find the transmission line (TLN) parameters by simulating a short uniform line. Then, it will use the TLN parameters to do the shifting of reference plane. There are cases the shifting reference plane feature on MGRID/IE3D may not work well. One case is that the ports are closely coupled together. When a few ports are coupled together, the TLN parameters are dependent on the excitation of the port parameters. Another case causing problem is a long distance reference plane shift. For example when we shift the reference plane by one wavelength, we need to change the phase of S21 by 720 degrees. If we introduce 0.1% error in calculating the wavelength, we will introduce 0.72-degree error in the phase of S21. If we shift the reference plane 10 wavelengths, the error in the phase of S21 may be 7.2 degrees. The problem in the magnitude of S21 is even more serious. This kind of error is actually understandable. However, you can get higher accuracy result by doing the following. You simulate your structure and simulate a 10-wavelength long TLN. Then, you use **Process->Back Simulation** command to remove the 10 wavelength long transmission line on the port. It is equivalent to the shifting reference plane on MGRID/IE3D. However, it may be more accurate. In fact, the benefit of the Back Simulation goes beyond the shift of reference plane of uniform transmission lines. You can even use it to shift the reference plane of non-uniform transmission lines.
Separate S-Parameters	This is a very powerful feature. It allows a user to separate an s-parameter file into two connected s-parameter files. It is very similar to the Back Simulation feature. However, this is a more general feature. It will be discussed in the manual.
Remove S-Parameters	This is also a very powerful feature. It allows a user to remove the effect of coupled transmission lines on the ports. It is a command can replacing the Back Simulation. The Back Simulation can only remove the effect of isolated transmission lines on the ports.
Find Transmission Line Parameters	Solve for the characteristic impedance and other TLN parameters of the s-parameter file of a structure. The structure must be a uniform 2-port structure shorter than quarter wavelength. When you use this command to find the TLN

	parameters from the 2-port s-parameters of a TLN, MODUA will prompt you for the physical length of the TLN. The characteristic impedance will be correctly calculated no matter what length value you enter. However, entering a value other than the actual one will certainly affect the propagation constant and effective dielectric constant.
Create TLN S-Parameters	Allow a user to create an s-parameter file for the frequency response of an ideal TLN based upon the electrical length, characteristic impedance and TLN parameters.
Curve Fitting and Interpolation	Curve-fitting the s-parameters with fine frequency detail based upon the existing frequency points.
Create S-Parameter File	Allow the user to define an N-port s-parameters with constant S(i,j) value over a frequency range.
Re-Normalize and Save	Allow the user to re-normalize the s-parameters and save it.
Find Perfect Load	Allow a user to find the 1-port s-parameter terminated at the output port (port 2) of a given 2-port s-parameter file yielding a perfect match looking at the input port (port 1). This is the opposite of the "Find Matching Circuit" in the next. It can be a useful tool for wide band antenna design.
Find Matching Circuit	Allow a user to find the 2-port matching circuit's s-parameters yielding perfect match at the input port (port 1) when the output port (port 2) of the matching circuit is connected with a known 1-port s-parameters.
Differential Conversion	Allow a user to convert an existing s-parameter file into a new s-parameter file with some of the ports differentially excited.
Odd and Even Mode Conversion	Allow a user to convert the s-parameters of uniform coupled TLN into the s-parameters based upon odd and even modes.

Table 2.23 The important menu items in the View menu of MODUA.

Menu Item	Explanation
Design Identification	Allow a user to enter a notation for the s-parameters of the current design in a display.
Design View Toggle	Toggle between displaying the names of the modules and displaying the values of the RLCM modules.
Graph Parameters	Allow a user to change the parameters for a graph displayed.
Display Grid Toggle	Toggle between displaying grids and displaying no grids in a graph.
Optional View Settings	Allow a user to define the settings. For example, you can setup the default display mode for MODUA.
Extend Horizontal Scroll Bar	Extend the horizontal scrollable size.
Extend Vertical Scroll Bar	Extend the vertical scrollable size.
Display Queue Items	Allow a user to enable and disable displaying the items in the display queue.

Section 2.5 The Configuration of IE3DLIBRARY's Window.

The IE3DLIBRARY is introduced in the IE3D 9.0 as the 2nd interface of the IE3D. Before the IE3D 9.0, MGRID was the only layout editor for geometry construction. MGRID has been a very powerful graphic interface. MGRID is based upon low level objects: polygons and vertices. We can create and modify polygons and vertices, which are the basic elements of a structure, to achieve circuits and antennas of general shape. Users can also create high-level objects such as circular spiral inductors, MIM capacitors with grounding via on MGRID. However, after the high-level objects are created, they will be broken down into polygons, the basic objects. The high-level objects cannot be edited as an entity on the MGRID. It would be nice if we can create and edit a high-level object as an entity. Implementing such a scheme on the

MGRID is quite tough because of the way of MGRID handling the polygons and vertices. For this reason, we have implemented the IE3DLIBRARY, the parameterized IE3D interface. The IE3DLIBRARY allows a user to create a circuit or an antenna based upon high-level objects. The high-level objects are described as a set of polygons. However, the user does not have direct access to the polygons. Any further editing has to be done on the high level objects. Therefore, the integrity of each object is maintained. Compared to MGRID, the IE3DLIBRARY is much easier to create and modify some commonly used structures. However, the IE3DLIBRARY cannot access the small details of a structure. It would not be as flexible as MGRID for general structures. Structure created using IE3DLIBRARY or MGRID can be interchangeable. In IE3D 12, we have implemented Boolean objects and void objects. It improves the usability of IE3DLIBRARY significantly. It allows us to create and parameterize complicated structures can't be imagine in IE3D 11. Combining with the equation-based geometry modeling and FastEM Design Kit, IE3DLIBRARY makes full-wave EM design simple and easy.

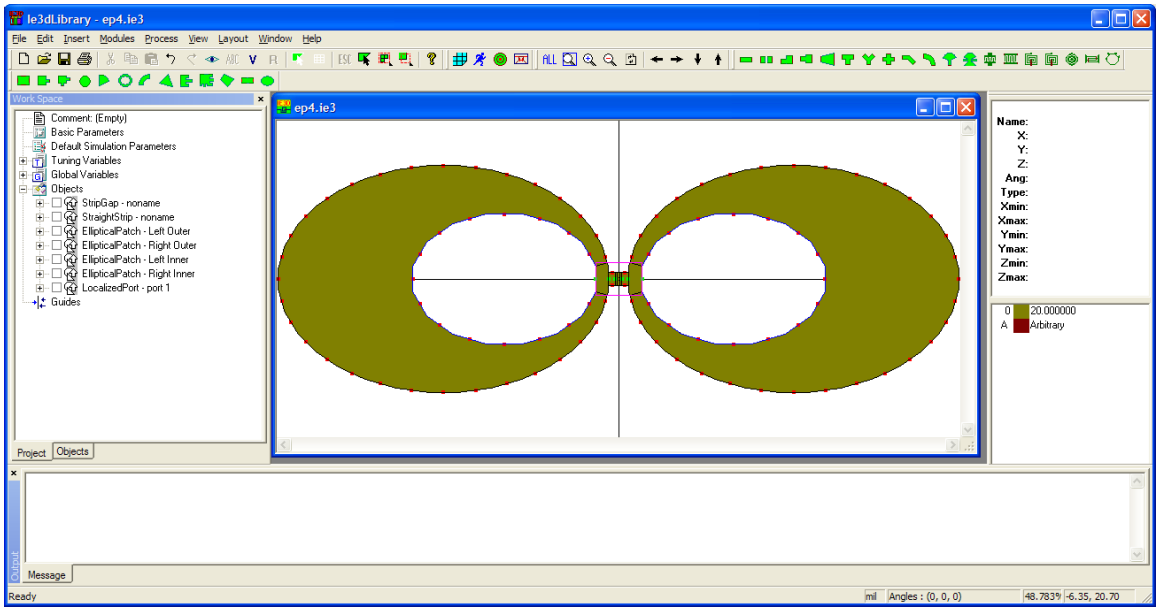


Figure 2.5 The IE3DLIBRARY window with complicated structure from Boolean operations.

The typical IE3DLIBRARY window is shown in Figure 2.5. There are 5-menus items in it. On the File menu, you can start a new file, open or save a file. The file extension is .ie3. Also, you can save a structure as .geo file and invokes MGRID for further geometry modeling on polygon and vertices level. The Edit menu has the basic select, copy, cut and paste commands. There are extra commands we will discuss later with detail. All the objects are in the Insert and Module menus. You can select any menu item in the two menus to define an object, and then place and connect it to the objects in the layout. On the Edit menu of the IE3DLIBRARY, there is a menu called "Tuning Variables…" You can define a set of variables.

The most significant new feature on IE3DLIBRARY are: (1) Equation based geometry modeling; (2) Boolean objects for building holes and cutting portions of other objects for complicated structures; (3) Void objects for parameterized geometry location; (4) Geometry tuning and FastEM Tuning for easy full-wave EM designs. More discussions and explanations on using IE3DLibray can be found from the context, the appendices and other related documents.

Section 2.6 Other IE3D Executables, IE3D Simulation and Optimization Procedures

In the IE3D versions earlier than 9.0, CURVIEW was used to display current distribution and radiation patterns. Starting from IE3D 9.0, its major functionality has been improved and integrated into the

MGRID. CURVIEW has been provided in IE3D 9.X and IE3D 10.X for old customers. CURVIEW is still provided for those customers need it. However, it is completely phased out in IE3D 11 and no support will be available on it.

On MGRID, bi-direction conversion between IE3D and GDSII stream format is included in the standard editions. However, conversion between IE3D, DXF (AutoCAD), ACIS (SAT) and GERBER is optional through the ADIX conversion program. Starting from IE3D 10.1, ADIX is fully integrated into MGRID. The conversion feature of ADIX will be fully functional on MGRID when a user purchases the optional ADIX license, even though the stand-alone ADIX is still provided.

For the IE3D 10.X, an IE3D simulation goes through the following procedures: (1) You create a geometry model and define the appropriate ports on MGRID.EXE. (2) You run the simulation setup procedure to define the frequency ranges on MGRID.EXE. (3) MGRID.EXE invokes the simulation engine IE3D.EXE to perform the EM simulation. (4) After simulation, IE3D.EXE will invoke MODUA to display the resulting s-parameters, another MGRID.EXE as post processor to display the current distribution on the structure, and PATTERNVIEW.EXE to display the radiation patterns. IE3D.EXE 10.x is a dialog-based application. It will perform an EM simulation and display the simulation progress on the dialog. Such an arrangement caused inconvenience.

On the IE3D 11, IE3D.EXE no longer contains the engine inside. It is just a dialog showing the progress of a simulation (or optimization). Underneath it is the true EM simulation engine, the IE3DOS.EXE, a command prompt DOS-style application. During an IE3D simulation, the IE3DOS.EXE will be running in the background while the IE3D.EXE will be displaying the simulation progress. In fact, on the MGRID 11, you have the option to perform either an in-process simulation (Invoke In-Process IE3D) or an external simulation (Invoke External IE3D). For the external simulation procedure, MGRID.EXE will invoke IE3D.EXE and IE3D.EXE will invoke IE3DOS.EXE. After IE3D.EXE is invoked, you can do other things using MGRID or close it. For the in-process simulation, the IE3D dialog is part of MGRID. You cannot use the in-process MGRID to do other things. You have to wait until IE3DOS.EXE finishes the simulation or you cancel the process.

Another useful application of the IE3D package is the ZDibAnimator. MGRID in post-processing mode allows you to perform animation of current distribution on a structure. However, you can save the animation pictures into a series of bitmap files. The saved bitmap files can be played on ZDibAnimator later for a movie of the flowing current distribution.

Most of the simulations documented in this manual are performed on a Pentium 4, 2.8 GHz notebook computers with 512 M RAM. Because the IE3D is a computational intensive program, the simulation speed is almost independent of the RAM amount when it is enough and no swapping is required. Its speed is almost proportional to the clock-rate of the CPUs for the same class of CPUs. Table 2.24 lists the speed comparison. The IE3D 6.x and later has an implementation of the SMSi matrix solver, which is the default matrix solver. The SMSi is about 4-10 times of the speed of the SMS matrix solver. The SMSi matrix solver is available on any Win32 machines with Intel Pentium Pro, Pentium II, Pentium III, Pentium 4 and AMD Athlon CPUs. Couple years ago, Intel released the Centrino CPU especially for Notebook computers. Similar to Athlon CPUs from AMD, Centrino CPU at lower clock-rate may be able to outperform Pentium 4 CPU with higher clock-rate for some business applications due to the differences in the architectures. However, we found that the Centrino CPU and Athlon CPUs are about 2/3 of the speed of a Pentium 4 CPU of same clock-rate for a medium size problem on the IE3D. IE3D runs the fastest on Pentium 4 and Core 2 Duo computers. We also found that IE3D does not benefit from the extra cache on Xeon CPUs. Xeon CPUs are about the same speed as the Pentium 4 CPUs running the IE3D.

IE3D 12 and 14 partially supports multiple CPU or multi-core systems. Significant improvement can be found on multi-CPU or multi-core computers.

Table 2.24 The speed comparison for the IE3D 9.0 on different platforms

CPU	486DX2-66	P-90	P-200	Pentium-Pro 200	P-II 400
Clock-Rate	66 MHz	90 MHz	200 MHz	200 MHz	400 MHz
IE3D Version	IE3D 1.0	IE3D 3.0	IE3D 4.0	IE3D 6.0	IE3D 7.0
Speed	1	2	4.5	6-9	12-18
CPU	P-III 800	Athlon	P-4 2.8 G	P-4 2.8 G	Core 2 Quad
Clock Rate	800 MHz	800 MHz	2.8 GHz	2.8 GHz	2.6 GHz
IE3D Version	IE3D 8.0-9.0	IE3D 8.0-9.0	IE3D 11	IE3D 12	IE3D 14
Speed	24-36	36-54	80-300	300-1500	600-6000

Section 2.7 Distributed EM Simulation and Optimization on ZDS and ZDM for Windows and Linux

Starting from the IE3D 8.2, the IE3D Distributed Service (ZDS) and IE3D Distributed Agent (ZDM) are available for network based distributed computing for the IE3D. It allows a single simulation of multiple frequency points to be distributed onto multiple computers and/or multiple CPUs on some computers in the whole network. The Adaptive Intelli-Fit (AIF) scheme is fully implemented into the ZDS/ZDM environment enabling extremely fast EM simulation and optimization over a network. All the procedures are fully adaptive.

The requirement for the ZDS/ZDM 10.x is that you need to configure the networked computers to access the IE3D files in common network drives. We need to allow all the IE3D engines in the network to access the necessary files in the same directory in a simulation. On the IE3D/ZDS/ZDM 11, we are able to remove the above requirement. It is not necessary to configure the directory. A user can save a geometry file in a local directory and submit the simulation into ZDS directly. The ZDS will automatically grab the necessary files from the local machine and distribute the files and jobs into the ZDM machines in the whole network. On IE3D/ZDS/ZDM 12.1, we implemented the JobsManager for managing local and distributed computing jobs.

Typical IE3D/ZDS/ZDM distributed EM simulation and optimization flow charts are shown in Figure 2.6 and Figure 2.7, respectively. Starting from IE3D 12.1, we can also submit a job to let one single ZDM agent to take care all the frequency points of the job instead of breaking it into many single frequency jobs.

Linux OS has been becoming more and more popular. For this reason, we have ported the IE3DOS engine and the ZDM into the Red Hat Linux OS, while ZDS and the IE3D GUI are still running in the Windows environment. You can do layout editing and start an EM simulation on a Windows machine. The simulation will be submitted to the ZDS server and the ZDS server can distribute the job into multiple ZDM/IE3DOS agents on Linux machines to speed up an EM simulation. After the simulation is done, all the results are sent back to your local machine. As a user, you will feel that the simulation is performed on your local machine while it is multiple times faster. The IE3D/ZDS/ZDM handles all the file transfer and simulations transparently to you.

We will start the IE3D tutorial in the next chapter. All the files described in this manual will be saved in the **C:\MentorGraphics\<latest_release>IE3D\SDD_HOME\IE3D\ie3d\samples** directory. The benchmark example files are saved in **C:\MentorGraphics\<latest_release>IE3D\SDD_HOME\IE3D \ie3d \examples** directory. The IE3D executables are located in **C:\MentorGraphics \<latest_release>IE3D\SDD_HOME\IE3D\exe** directory. To simplify the file name, we will assume the directory is **.\ie3d\samples**. It is suggested that you do not change them. You should save your practice files into **.\ie3d\practice** directory or your own directories.

Old IE3D users may have known that the files for simulation results (.sp, .cur and .pat) used to be

stored in the same directory as the geometry files (.geo) and the simulation input files (.sim). Starting from IE3D 11, the result files are stored in the OUTPUT directory below the directory where the .geo and .sim files are. However, users can re-configure IE3D with output files in the same directory as the .geo files.

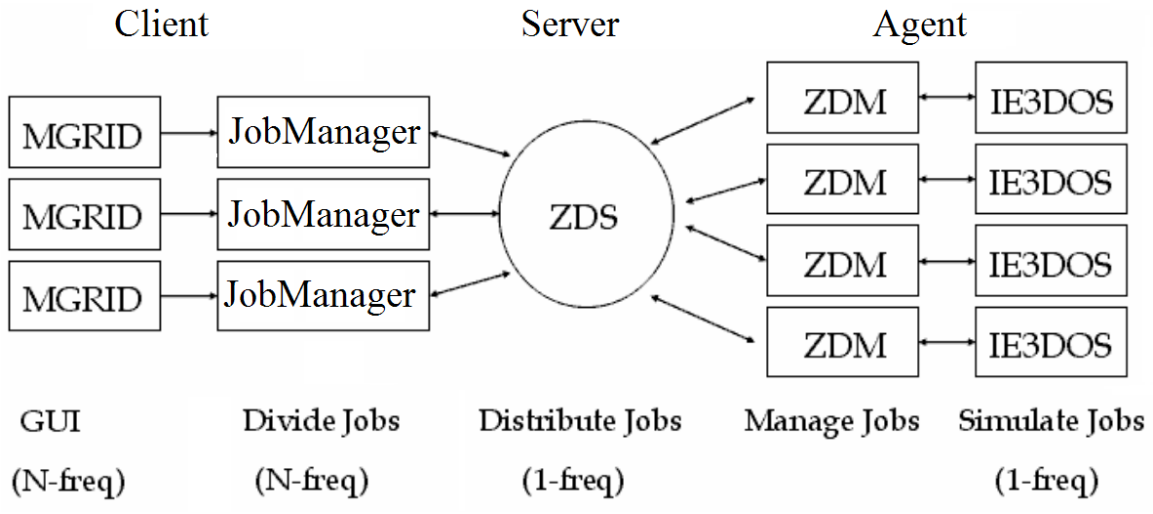


Figure 2.6 The ZDS/ZDM distributed EM simulation flow chart.

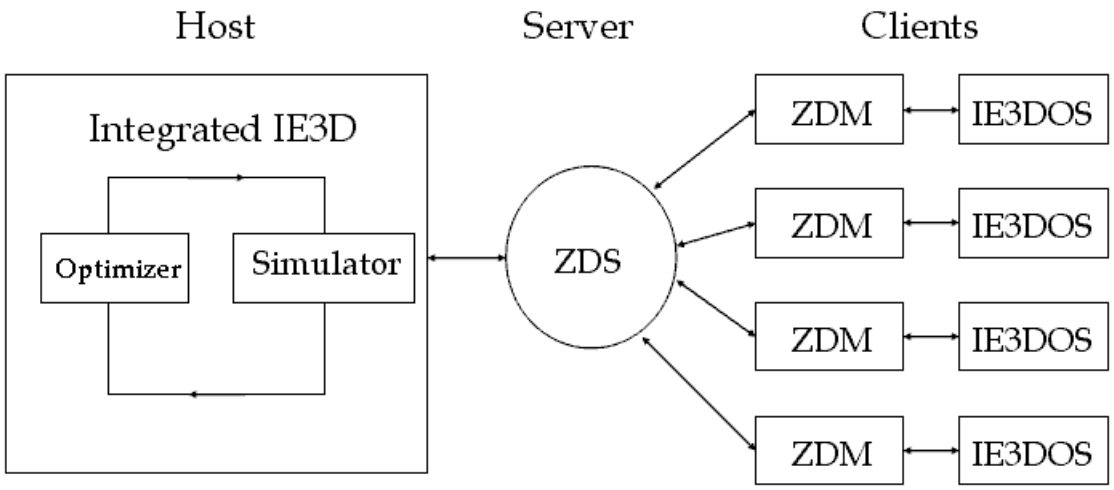


Figure 2.7 The ZDS/ZDM distributed EM optimization flow chart.

Chapter 3 Basic Techniques

IE3D is a general purpose EM simulator. It is used to design high frequency circuits and antennas. Without loss of generality, we will call a circuit or an antenna as a structure. On IE3D, a structure is described as a set of polygons, and a polygon is described as a set of vertices. This chapter will explain the basic editing skill by going through, step by step, the simulation on a microstrip bend structure.

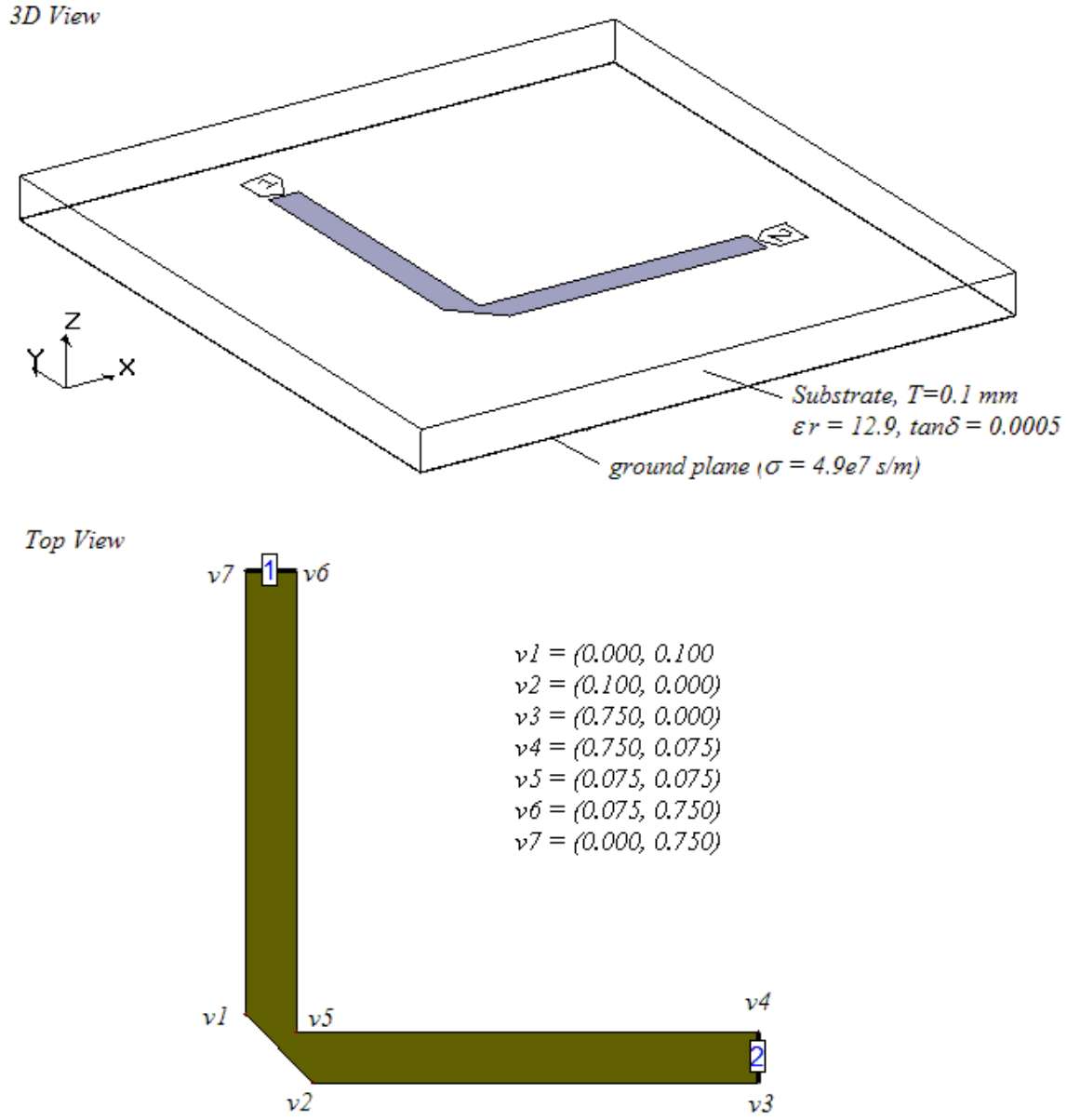


Figure 3.1 A simple chamfered microstrip bend structure.

There are many ways to create a structure on IE3D (see flow chart in Figure 1.5). You can import it from other standard formats such as GDSII, DXF, ACIS and GERBER using MGRID and ADIX. You can also convert it directly from GDSII, Cadence Virtuoso and Cadence Allegro using AGIF. You can convert it directly from Microwave Office using EMSocket. You can create them from a set of library elements on MGRID or IE3DLIBRARY. You can even draw it. Drawing mode is the default mode on MGRID. When

you manage to draw a structure with the many advanced geometry modeling commands on MGRID, you will be able to model and design very complicated circuits. All other ways on MGRID, IE3DLIBRARY and AGIF are based upon the geometry modeling capability underneath MGRID. If you can manage MGRID in doing geometry modeling, you will be able to manage the different ways in other applications of IE3D. They are built on top the same geometry modeling engine.

Section 3.1 Defining Basic Parameters

We start from MGRID. The parameters of the microstrip bend are shown in Figure 3.1. Before you draw the structure, please make notes to your structure. First you should introduce the x and y-coordinate system. The x and y-coordinates of each vertex should be marked. Distances between vertices should be calculated if it is necessary. Try to break a structure into several pieces when a complicated structure is modeled. Try to find a grid size such that your structure can be best fitted into the uniform grid. The uniform grid is used for snapping in mouse input only; it will not be used for meshing. The above steps are beneficial to construction of a structure. We list the (x, y) coordinates of each vertex in Figure 3.1.

We are going to construct the bend structure with the top view shown in Figure 3.1. By default, there will be an infinite ground plane underneath the substrate. The small rectangles with numbers on them are the ports of the circuit. We can build structure as 3 connected polygons as illustrated in Figure 3.2. As connection of polygons will be discussed in the next chapter, we will build the microstrip bend as a single polygon in the following.

Step 1 Run MGRID by double-clicking the MGRID icon in the IE3D executables location C:\MentorGraphics\<latest_release>IE3D\SDD_HOME\IE3D\exe. (It will ask the user to start IE3D Program Manager to open MGRID. Click 'Yes' to start IE3D Program Manager). You can also run it from the corresponding icon on the IE3D Program Manager (IE3DPM or ie3dpm.exe) assuming that MGRID.EXE is in the same directory of IE3DPM. If you put them into different directory, you will not be able to run MGRID from IE3DPM.

Step 2 Select **File->New** command to start a new project. MGRID will prompt you for the *Basic Parameters* dialog (see Figure 3.3). You need to enter the basic parameters first.

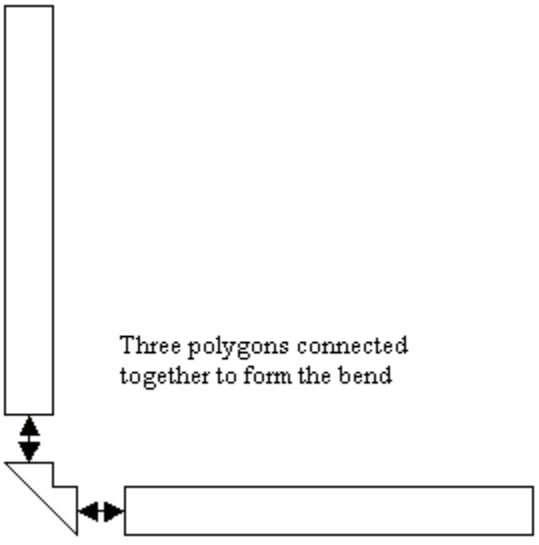


Figure 3.2 The chamfered bend broken down into three polygons.

The Basic Parameters include 6 groups of parameters: (1) Comment: The comment for the structure; (2) Length: the Length Unit and Minimum Length for the structure; (3) Layouts and Grids: the parameters for the uniform grid system for the layout editing; (4) Enclosure: The walls for the enclosures used in the structure; (5) Meshing Parameters: the parameters control the meshing of the geometry; (6). Substrate Layers: The parameters of substrates and infinite ground planes; (7) Metallic Strip Types: The parameters for different types of metallic strips used in the structure; (8) Dielectric Types: The dielectric types for 3D dielectrics and finite size substrates.

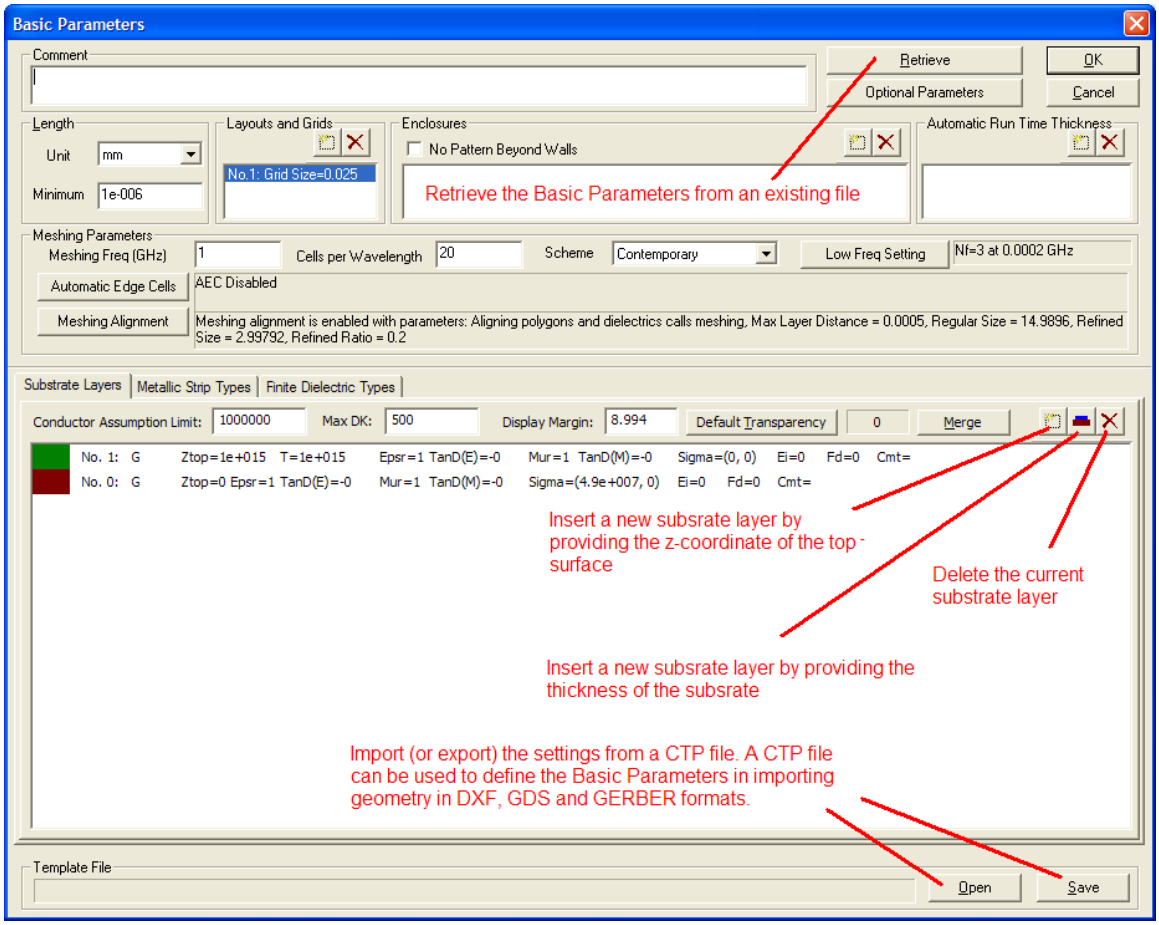


Figure 3.3 The Basic Parameters dialog.

For this structure, we will use "mm" for the length unit. We can accept the default Minimum Length which controls the accuracy of vertex location in mouse input. Please note the buttons above some list boxes in Figure 3.3. The meanings of the buttons are shown in Figure 3.4.

Figure 3.4 The meaning of the buttons for the list boxes in Figure 3.3.

Step 3 You can define multiple "Layout and Grid" items. There is a default one in the list box. Please double click at the only item in the Layouts and Grids listbox (see Figure 3.3). MGRID prompts you for the Edit Grid Size. The default Grid Size = 0.025 mm. It is what we want.

Select OK to accept the default value for the default grid size. You are back to the Basic Parameters dialog.

Step 4 Change the Meshing Frequency (Fmax) from 1 to 40 GHz because we want to simulate the structure up to 40 GHz. Remain the Cells per Wavelength (Ncell) as 20.

The higher the Meshing Frequency is, the finer mesh a structure will be discretized to. Finer meshing is preferred for higher accuracy, but you pay for the substantial increase in the simulation time. Many users are concerned about the meshing of a structure, because they understand the simulation results of some EM simulators are very sensitive to the meshing. IE3D is a method of moment (MOM) simulator. Simulation results usually are very stable as long as we use about 15-20 cells per wavelength and edge cells along open edges, unless the structure is electrically small where finer meshing with respect to wavelength is required. We will discuss accuracy improvement and related issues in Chapter 12.

After we change the Fmax from 1 to 40 GHz, the Low Freq Setting is changed from "Nf=3 at 0.002 GHz" to "Nf=3 at 0.008 GHz". The Low Freq Setting is new in IE3D 14 for obtaining high accuracy results at low frequency using extrapolation. "Nf=3 at 0.008 GHz" means that we will use 3 frequency points starting at 0.008 GHz for extrapolation. You can double click at Low Freq Setting to change the Nf and the starting frequency. Normally, we can accept the default setting.

Step 5 Double click at the Automatic Edge Cells button (see Figure 3.3). The Automatic Meshing Parameters comes up (see Figure AL.1). There are many parameters in the dialog: Fmax and Ncell are in this dialog too. The Automatic Meshing Parameters have changed much in the IE3D 11. For more discussion on Automatic Meshing Parameters, please read Appendix AL.

As it is pointed out in Appendix AL, Automatic Edge Cells (AEC) improves simulation accuracy significantly. It can yield expert results for novice users. However, IE3D may still be able to yield accurate results for many simple structures without strong coupling. For the 1st example, let's try to simulate the structure without AEC. Please select AEC Layer: No" (see Figure AL.1).

You can select Contemporary meshing scheme. Contemporary scheme is a newer scheme which normally may yield similar accuracy with fewer unknowns. Classical scheme is an old scheme and it is normally more reliable.

You can select the parameters so that the dialog looks like what is shown in Figure AL.1. Please select OK to finish defining the Automatic Meshing Parameters.

Section 3.2 Defining Substrate Parameters

Substrate parameters are also part of Basic Parameters. They include the number of dielectric substrate layers, the z-coordinate of the top surface of each substrate layer, the substrate permittivity, permeability and conductivity. Complex permittivity, permeability and conductivity are accepted in IE3D. The No.0 substrate layer is assumed to be the ground plane by default. The infinite ground plane is defined as a substrate with a very high conductivity. The z-coordinate of the top surface of No.0 substrate layer is always equal to 0 and cannot be modified. Other parameters can be modified according to the actual cases. For example, you can define the conductivity of the No.0 layer as 0 in order to replace the default ground plane with air.

For multiple layer substrates, you just need to define the top surface z-coordinate of each layer and define the corresponding permittivity, permeability and conductivity. You can define as many substrate layers as you like. You have to define at least one substrate layer in the upper half space. It means there should be at least 2 substrate layers (including the No.0 layer) in a structure. When the upper half space is filled with a single type of dielectrics, you can define a substrate layer with the top surface z-coordinate of top surface as a very large number (such as 1.0e+15 mm). Examples of defining different dielectric substrates are documented in Appendix T.

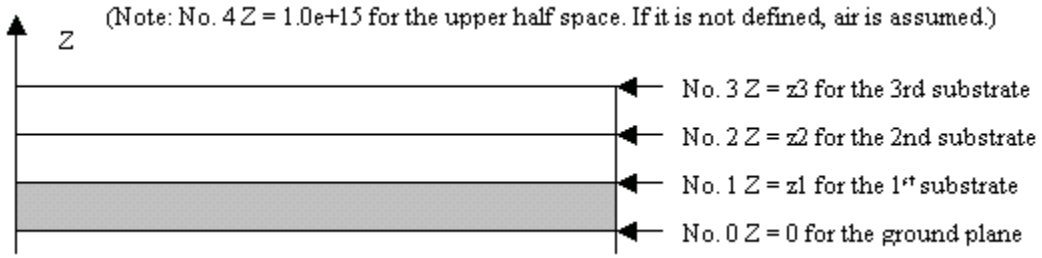


Figure 3.5 The substrate stack up and definition on MGRID.

By default, MGRID creates 2 substrate layers for you. The No.0 layer is a good conductor with conductivity = 4.9e+7 s/m, which is gold. The No.1 layer is air and its top surface Z-coordinate, Ztop = 1.0e+15 mm, meaning that the whole upper space is filled with air.

For the structure we are going to simulate, we have three dielectric layers (including the ground plane). The No.0 layer is the ground plane. The No.2 layer is the upper half space air, which can be omitted. The No.1 layer is the dielectric slab with the following parameters:

Top Surface Z-Coordinate, Ztop = 0.1 mm
Real Part of Permittivity, Re(EPSr) = 12.9
Loss Tangent of EPSr = 0.0005
Real Part of Permeability, Re(MUr) = 1.0
Loss Tangent of MUr = 0.0
Real Part of Conductivity = 0.0 s/m
Imaginary Part of Conductivity = 0.0 s/m

Dielectric permittivity is also called dielectric constant. There are 2 ways to define substrate loss in IE3D: (1) defining the imaginary part (or loss tangent) of the permittivity and (2) defining the conductivity. Theoretically, they are interchangeable, and they are defined as:

$$\varepsilon_{rc} = \varepsilon_r - j\,\sigma / (\omega\varepsilon_0) = \varepsilon_r (1 - j \tan\delta) \qquad (3\text{-}1)$$

where ε_r is the real dielectric constant; σ is the conductivity; ω is the angular frequency; ε_0 is the complex dielectric constant in free space (8.86×10^{-12} F/m); $\tan\delta$ is the loss tangent. In the other word, we have,

$$\tan\delta = -\mathrm{Im}(\varepsilon_{rc}) / \mathrm{Re}(\varepsilon_{rc}) = \sigma / (\omega\varepsilon_0 \varepsilon_r) \qquad (3\text{-}2)$$

The Loss Tangent is always a non-negative number and the imaginary part of permittivity is always a non-positive number. In practical applications, users normally use either $\tan\delta$ or σ as frequency independent values. However, we can see from (3-2) that, when $\tan\delta$ is frequency independent, σ will be frequency dependent, or vice-versa. In order to resolve this inconsistency, we define the $\tan\delta$ and σ in the IE3D as in (3-3):

$$\varepsilon_{rc} = \varepsilon_r (1 - j \tan\delta) - j \sigma / (\omega \varepsilon_0) \qquad (3\text{-}3)$$

You should understand that this is not the true definition of the tanδ and σ. It is just for the convenience of users. If you want to define the material with frequency independent tanδ, you should define σ = 0 for the dielectrics. If you want to define the material with frequency independent σ, should define tanδ = 0 for the substrate. However, if you define non-zero values for both tanδ and σ, both will be used to calculate the ε_{rc} value in (3-3).

Starting from IE3D 9.2, we have implemented frequency dependent tanδ and σ. Frequency dependent tanδ and σ are discussed in the Appendix AD.

You may notice that a parameter called Conductor Assumption Limit (CAL) is displayed in the dialog. This parameter is defined in the Optional Parameter in Parameters menu (Appendix A). In IE3D, highly conductive substrates are treated differently from normal substrates while they are defined in the same section. The CAL is used to identify which substrate will be treated as high conductivity. Each substrate will be displayed with a CFactor. If the CFactor of a substrate exceeds the CAL, the substrate will be considered as high conductivity substrate or ground plane. Another parameter is called Max DK. The Max DK is used to identify which substrates will be used to estimate the waveguide wavelength for the meshing. If the CFactor of a substrate exceeds Max DK, the substrate will not be used for estimation for the meshing density. Other Parameters are Substrate Display Margin and Default Transparency. They are used to control how we display the substrates in the 3D view window. They will not affect the meshing and the simulation. Starting from IE3D 14, Substrate Display Margin is in the true length dimension, while it is the ratio with respect to the wavelength in the older versions.

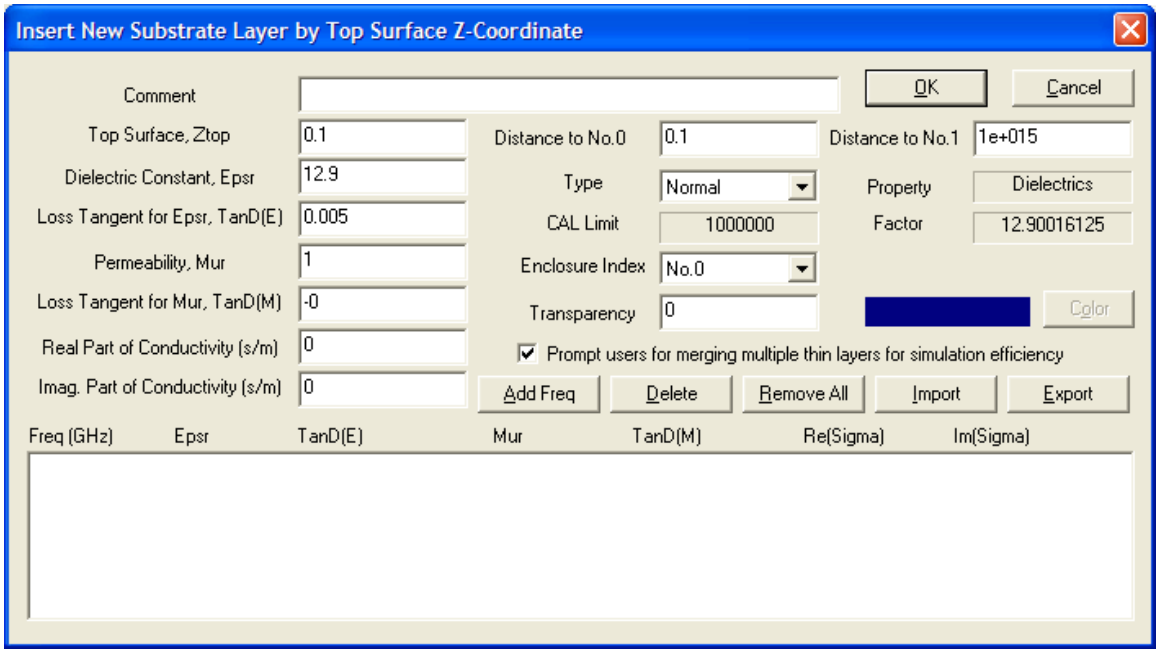


Figure 3.6 The Edit Substrate dialog with the No.1 substrate parameters entered.

Step 6 Select the "Insert" button (see Figure 3.3 and Figure 3.4) above the Substrate Layers listbox. MGRID will prompt you for a new substrate layer (see Figure 3.6).

Step 7 Make sure it is Normal type. Enter the parameter of the No.1 substrate into the dialog (see Figure 3.6). We can specify an **Enclosure Index** to each substrate. This feature allows us to

define multiple enclosures separated by ground planes. We can define an enclosure between each pair of ground plane. Typical applications are cavity-backed antennas. Multiple enclosures are discussed in Appendix H. Also, you can select HTS II type for the substrate. We will explain the HTS II type when we discuss the metallic strip parameters in the next.

Please select OK to enter the new substrate into the dialog. We will have the Substrate Layers listed in Figure 3.7.

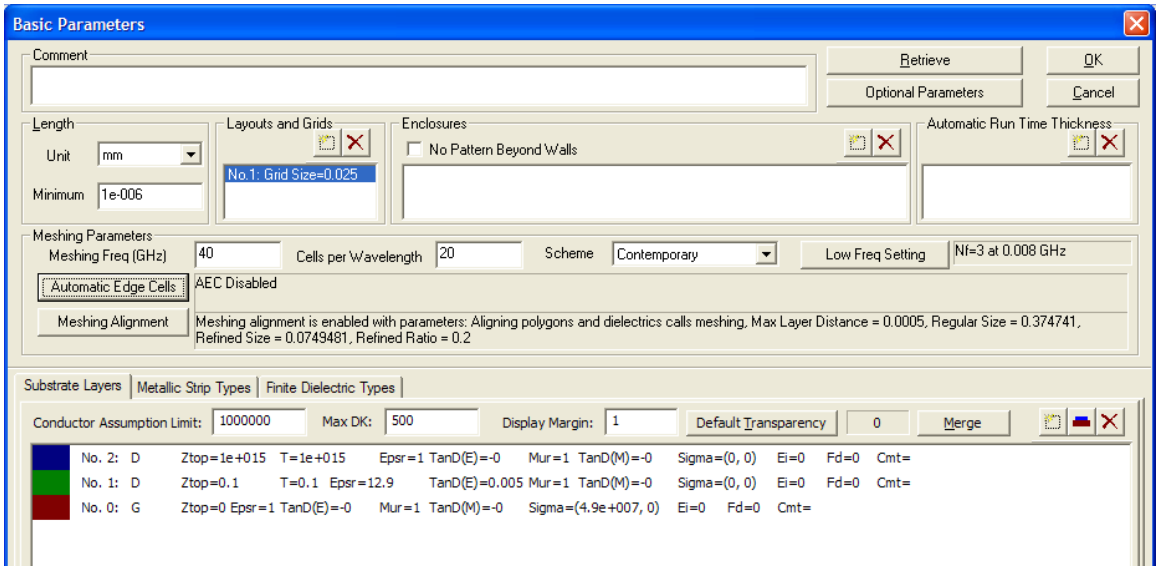


Figure 3.7 The substrates layers after we inserted the layer at 0.1 mm.

Section 3.3 Defining Metallic Strip Parameters

The next parameters we should define are metallic strip types. For a normal conductor, metallic strip parameters include the strip thickness, surface roughness, permittivity, permeability and conductivity. You need to define at least one metallic strip. We assign a metallic strip type to each polygon created. When you enter a polygon during geometry editing, the polygon is normally assumed to be of the 1st printed strip type. To modify the metallic strip type of some polygons, you select the polygons you want to change the metallic strip type. Then, you select **Edit->Object Properties** command. Then, select G**roup Change Metal Type** button in the dialog to change it.

You can also define HTS I, HTS II and thin film resistor for the metallic strip types. The HTS I and HTS II do not mean they are different types of HTS material. They just mean we use two different formulas for the HTS modeling.

To define a metallic strip type, please make sure you select the Metallic Strip Types tab and select "Insert" button above the listbox, the layout editor will prompt you to define the parameters.

For this circuit, we have one metallic strip type. The parameters of the No.1 metallic strip type are listed below. The parameters are the same as the default No.1 metallic strip type in the listbox.

Strip thickness = 0.002 mm
Surface Roughness = 0.0
Real part of permittivity = 1.0
Imaginary part of permittivity = 0.0

Real part of permeability = 1.0
Imaginary part of permeability = 0.0
Real part of conductivity = 4.9e+7 s/m
Imaginary part of conductivity = 0.0 s/m

Step 8 Double click at the No.1 type in the Metallic Strip Types listbox of the Basic Parameters dialog. The Edit Metallic Type dialog comes up and let you edit the parameters (see Figure 3.8).

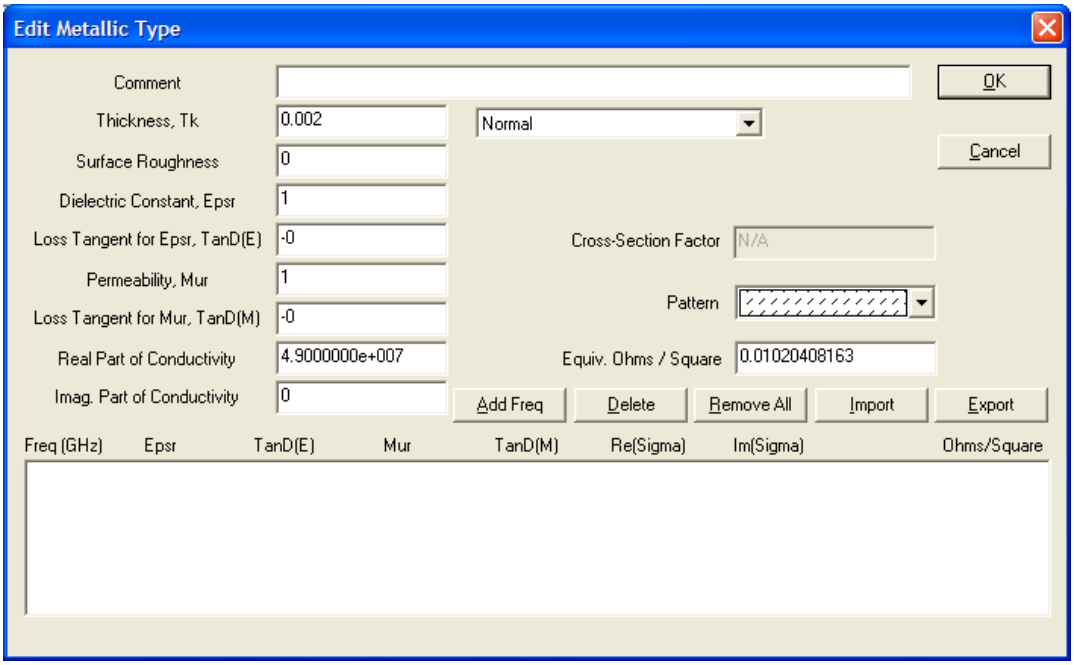


Figure 3.8 The Edit Metallic Type dialog with the No.1 metallic strip type parameters.

You are allowed to define frequency dependent metallic strip types. Frequency dependent metallic strip types and substrates are discussed in Appendix AD.

Also, MGRID lists the Equiv. Ohms/Square value for the provided conductivity value. You should understand that the Equiv. Ohms/Square value is the value at DC. It will change at high frequency. In fact, it should also have imaginary part at high frequency.

We do not need to change the parameters because they are what we want. Select OK and MGRID will show all the Basic Parameters on the dialog.

Step 9 Select OK to close the Basic Parameters dialog. MGRID will be ready for us to create the polygons for the structure.

Another group of parameters in Basic Parameters are the Dielectric Types. They are for the modeling of finite dielectrics. For this example, it does not involve finite dielectrics and we do not need to worry about it.

The layout and grid of the MGRID is completely shown now. The small red dots are the grids. The red lines represent the x and y-coordinate system. For the open boundary formulation, the IE3D assumes the boundary of the substrate and ground plane (if defined) extend to infinity. For closed boundary formation, the IE3D allows a user to define the boundary of the enclosure. There are still some other

parameters in the Parameters menu. We do not need to change the Optional Parameters every time we construct a structure. We will explain the Optional Parameters later when you get familiar with the IE3D.

Section 3.4 Editing Polygons on Layers

What we need to do is to build a structure as a set of polygons. The polygons can be on some layers. We call the layers as *metallic layers*. When all vertices of a polygon are on the same layer, we call it a 2D polygon. Otherwise, we call it a 3D polygon.

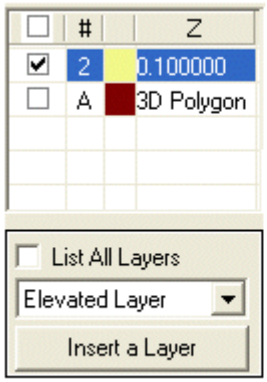
There are 2 layer systems on the IE3D: (1) The *substrate layers*; (2) The *metallic layers*. The *substrate layers* are those we defined in the Basic Parameters. They specify the infinite dielectric configuration of the structure.

The *metallic layers* are those displayed on the layer window at the lower right location of MGRID. Normally, metallic layers are automatically sorted out by MGRID based upon the z-coordinates of the vertices of the polygons and the top surface z-coordinates of the substrates. It is possible that you may see some metallic layers you do not expect to have 2D polygons on them. This may mean that some vertices of 3D polygons are on the layer or some substrate's top surface z-coordinate is on the layer. When you are building strong 3D structures, you may have many different layers on the layer windows even there are no 2D polygons on them. If you want MGRID not to display those 3D layers, you can un-check the *List All Layers* (see Figure 3.9) on the *layer window* to hide them.

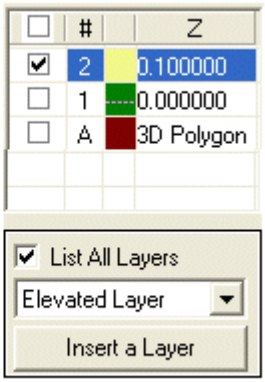
Also, it is possible that a metallic layer does not define some metallic polygons on it. When the layer is on the interface of high conductivity (or metallic) substrate, the substrate is basically a ground plane and the metallic layer defines a layer of slots on the ground plane. When such case happens, you will see a horizontal line across the color strip representing the metallic layer. The Figure 3.9b is an example. Z = 0 is the ground plane and the metallic layer on z = 0 has a horizontal line across it, meaning that any 2D polygon on this layer defines a slot on the ground plane.

There are many ways to enter vertices in MGRID. The simplest way is to use the mouse to draw them. Mouse input can be easy and accurate when there exists a good minimum length in the circuit. Good minimum length means that you can fit the circuit into a uniform grid with the minimum length as the grid size. Unlike some other simulators, the layout grid on the IE3D has nothing to do with the meshing in a simulation. Smaller layout grid does not slow down an IE3D simulation because we do not need to fit a structure into the uniform grid. We will use mouse to construct the chamfered bend in the following.

Please pay attention to the layer window on the lower right corner. Depending upon whether the List All Layers checkbox is checked or unchecked, we will get either Figure 3.9a or Figure 3.9b.



(a) With *List All Layers* unchecked



(b) With *List All Layers* checked

Figure 3.9 The layer window (a) with *List All Layers* unchecked; (b) with *List All Layers* checked.

Step 10 Please click at the color strip representing No.2: Z = 0.1 layer in the Layer Window to shift the 2D input focus on the No.2 layer at z = 0.1 mm.

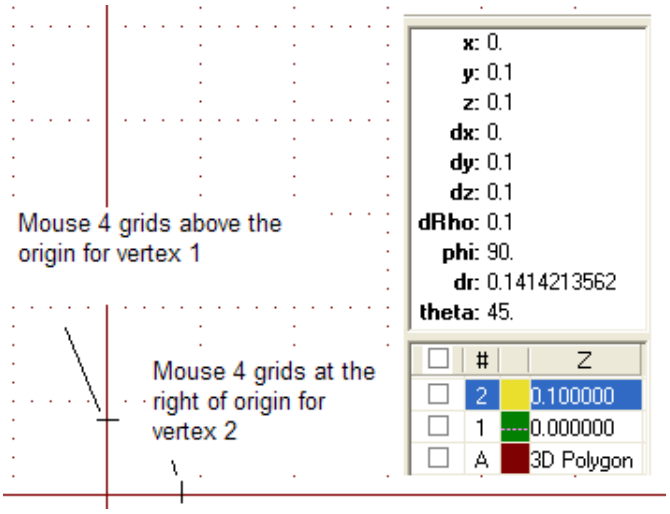


Figure 3.10 The mouse location and the status window.

Step 11 Move the mouse 4 grids over the origin. You will see x=0 and y=0.1 on the status window at the upper right corner (see Figure 3.10). Click the left mouse button. The vertex 1 of Figure 3.1 is entered at (x, y) = (0.0, 0.1) mm. It is displayed as a small dot on the window. A line linking the vertex to the mouse cursor is also displayed on the window.

We need not to locate the mouse cursor exactly on the grid point at (x, y) = (0.0, 0.1) mm as long as the cursor is closest to the grid point. MGRID will automatically snap the vertex to the grid. When you move the mouse, you can count how many grids the mouse cursor is from a reference point (such as the origin), and keep checking the coordinates displayed in the status window. If a vertex is entered at an unexpected grid point, you can drop the vertex by selecting **Input->Drop Last Vertex** To drop all entered vertices, please select **Input->Drop All Vertices**. The commands will appear on the pop up menu if you click right mouse button.

Step 12 Move the mouse 4 grids right to the origin (see Figure 3.10). The status window shows (x, y) = (0.1, 0.0) mm. Click the left mouse button to enter the 2nd vertex of Figure 3.1 at (x, y) = (0.1, 0.0) mm. An edge between vertex 1 and vertex 2 is created.

Step 13 Move the mouse 26 grids right to the last vertex. You do not need to count how many grids you have moved. You just check the status window to make sure it shows (x, y) = (0.75, 0.0), and click the left mouse button. The vertex 3 of Figure 3.1 is entered and an edge between vertex 2 and vertex 3 is formed.

Step 14 Move the mouse 3 grids up from the last vertex. The status window shows (x, y)=(0.75, 0.075). Click the left mouse button for the vertex 4 of Figure 3.1 to form an edge between vertex 3 and vertex 4.

Step 15 Move the mouse to the left of the last vertex. Please monitor the status window to make sure it shows (x, y) = (0.075, 0.075). Click the left mouse button to enter the vertex 5 of Figure 3.1.

Step 16 Move the mouse up from the last vertex, and monitor the status window to make sure (x, y)=(0.075, 0.75). Click the left mouse button to enter the vertex 6 of Figure 3.1.

Step 17 Move the mouse 3 grids left to the last vertex, and monitor the status window for (x, y) = (0.0, 0.75). Click the left mouse button to enter the vertex 7 of Figure 3.1.

Vertexes 1 thru 7 are sequentially connected, but they still have not formed the polygon. To form a polygon, we need to connect vertices 1 and 7. We can enter the last vertex at the same location as vertex 1 to close it. However, there are other ways.

Step 18 Select **Input->Form Polygon** command. The polygon is formed and colored. The color of the polygon is the same as the color strip labeled with z=0.1 in the layer window. It means that the polygon is on the layer with a vertical coordinate at z = 0.1mm. The final result is shown in Figure 3.11.

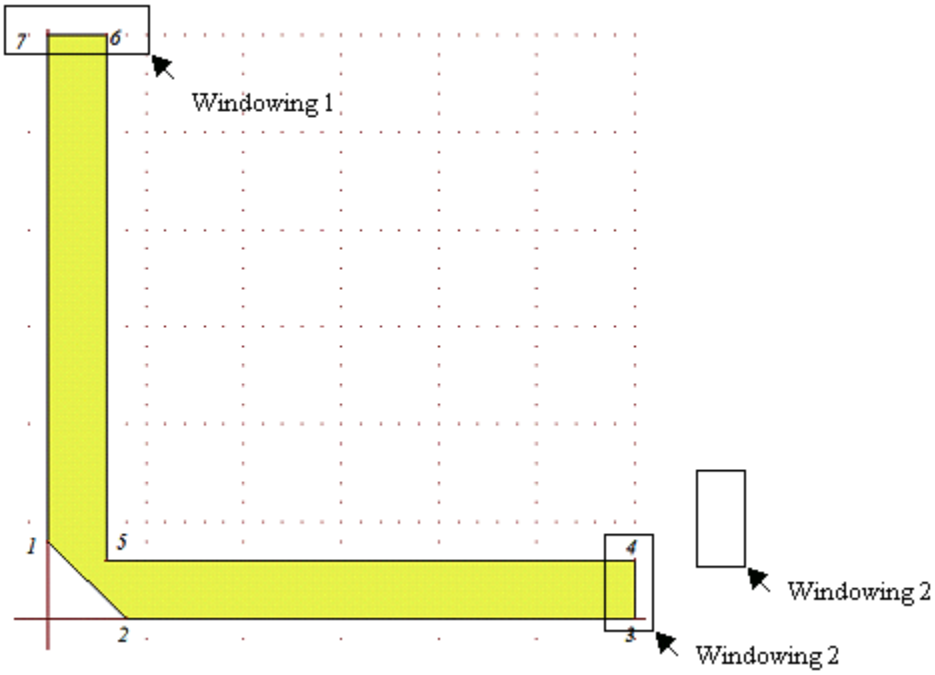


Figure 3.11 The entered polygon of the bend (vertex labels are no on actual MGRID window).

Section 3.5 Defining Ports and the Advanced Extension Scheme

Construction of the bend is finished at Step 18. Now we need to define the ports. The simulation engine will not run without any port (or planewave excitation) on a structure. A port is defined on polygon edges. There are multiple ways to define a port. The Port->Define Port command allows you to select an edge by clicking on it. The Port->Port for Edge Group command allows you to select a group of edges by windowing them. The Port for Edge Group command is more convenient even though there are cases the Define Port is a better choice. No matter which command you use, the end result should be the same if you select the same edges. There are other ways to define ports, which we will discuss later in the manual.

Step 19 Select **Port->Port for Edge Group** command. MGRID will prompt for the de-embedding scheme to be used (see Figure 3.12).

Every port is tied with a de-embedding scheme on IE3D. Different port may need different de-embedding scheme. One de-embedding scheme can only solve some kind of problems. To make the IE3D more flexible, we implemented total 7 de-embedding schemes: (a) **Advanced Extension**; (b) **Extension for MMIC**; (c) **Localized for MMIC**; (d) **Extension for Waves**; (e) **Vertical Localized**; (f) **50-Ohms for Waves**; (g) **Horizontal Localized**.

The extension de-embedding schemes, including (a), (b), (d) and (f), normally are the most accurate ones when we are simulating a sub-circuit divided from a larger circuit. However, there are cases we can only use the localized schemes, including (c), (e) and (g), to yield reasonable results. Among the extension schemes, the Advanced Extension scheme is the most accurate one.

For more information on which de-embedding scheme you should use, please refer the Appendix AM. For this example, we will choose the Advanced Extension scheme for it.

For extension port schemes, you need to decide the Cells on De-embedding Arm value. The default value is 5 and it is normally a good value. However, you should adjust it depending upon your real situation. Starting from IE3D 11, we have implemented the Auto Adjustment which makes it even easier for you to decide. The parameters for the General Extension Port Parameters section in Figure 3.12 are good values. In IE3D 14, we have implemented accurate extraction of s-parameters with shift of reference planes for coupled ports and differential ports. For more information about them, please read the Appendix AM.

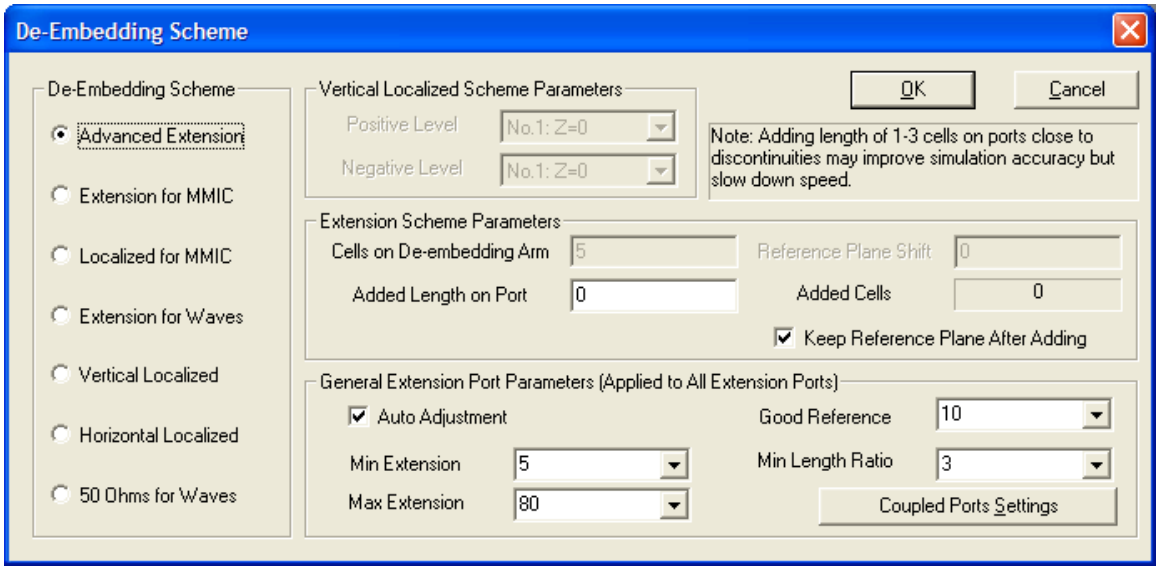


Figure 3.12 The De-Embedding Scheme dialog.

Step 20 Select **Advanced Extension** scheme. Check Auto Adjustment. Select OK to continue.

MGRID will be waiting for you to select the edges for defining a port. Please pay attention to the Status Window and the Layer Window. The Status Window is no longer display the mouse coordinates. It is displaying a message “Port for Edge Group: Window edge group on a vertical plane…” The Status Window basically tells what mode you are in and give you a hint what you should do.

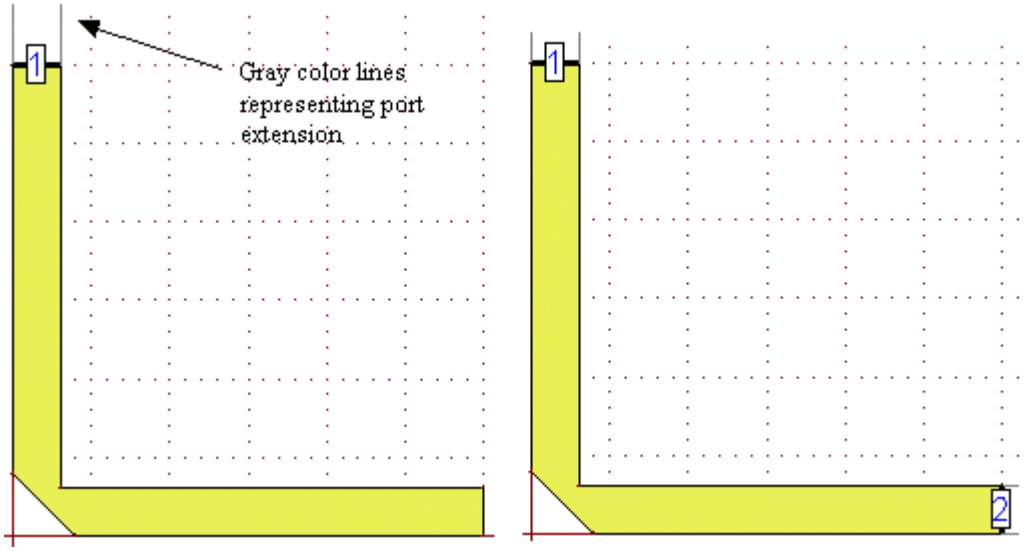
On the Layer Window, you may notice that there is a check box for each layer. Depending upon the Default Selection Focus setting in Parameters->Optional Parameters dialog, you may have different default states for the check boxes on each layer.

Please make sure the check box for No.2: Z = 0.1 is checked (see Figure 3.9). The reason we want the Z = 0.1 layer to be checked is the following: When we window some edges to define the edges of the polygons as a port, only the edges on the checked layer will be selected. If some vertex of an edge is on a layer not checked, the edge will not be selected even it is being windowed.

If you are an old IE3D user, you may notice that the polygon where mouse is pointing it while you are moving the mouse is highlighted. This is a new feature on IE3D 14. You are able to access the highlighted object (polygon, port or text) easily on IE3D 14 or later.

Step 21 Move the mouse to somewhere on the upper left corner of vertex 7 of Figure 3.12. Press down the left button. Drag the mouse to the lower right corner of vertex 6 while keeping the left button pressed. Release the left button at the lower right corner of vertex 6.

The above action is called windowing. You just defined the windowing 1 in Figure 3.12. The port 1 is defined on the edge formed by vertex 6 and 7 (see Figure 3.13a). A small rectangle with "1" is displayed on the edge by vertices 6 and 7. You will see 2 gray color lines extending from the strip. The gray color lines represent the port extension. The length of the gray color lines defines how long a fictitious section of transmission line will be attached on the port location in an actual simulation. You should be careful not to let other polygons to come across this port extension. Otherwise, you may get unreasonable results. More discussion can be found in Appendix AM.



(a) The structure with port 1 defined (b) The structure with both ports defined.

Figure 3.13a The structure with ports defined.

Step 22 MGRID is still in **Port->Port for Edge Group** mode. Please move the mouse to upper left corner of vertex 4 and window the vertex 4 and vertex 3 (see Figure 3.12). After you release the left mouse button, the port 2 is defined. The structure will look like what is shown in Figure 3.13b.

Step 23 MGRID is still in the **Port->Port for Edge Group** mode. Select **Port->Exit Port** command. MGRID will resume to the drawing mode. You can also select the button on the toolbar for Port for Edge Group mode to release it from the mode.

Step 24 Select **File->Save** command, and enter "**.\ie3d\practice\c_bend.geo**" for the geometry file name.

You can always intermediate result anytime. It is always recommended to save changes frequently when constructing large structure in order to avoid accidental loss of data.

Section 3.6 Meshing Structure on MGRID for Visualization

We have defined the polygons representing the strip and we have defined the ports of the structure. It is ready for simulation. We can select **Process->Simulate** command to simulate the structure now. However, you may be curious about how it looks like after a structure is meshed. We will demonstrate how to mesh the structure. The meshing on MGRID is just for display only, and its result is not used in the IE3D engine. The meshing is always performed on the IE3D simulation engine when you start a simulation.

Step 25 Select **Process->Display Meshing**. MGRID will prompt you for the Automatic Meshing Parameters (see Figure AL.1). Meshing Parameters are defined in the Basic Parameters dialog. However, you can change them when you perform a meshing or a simulation setup.

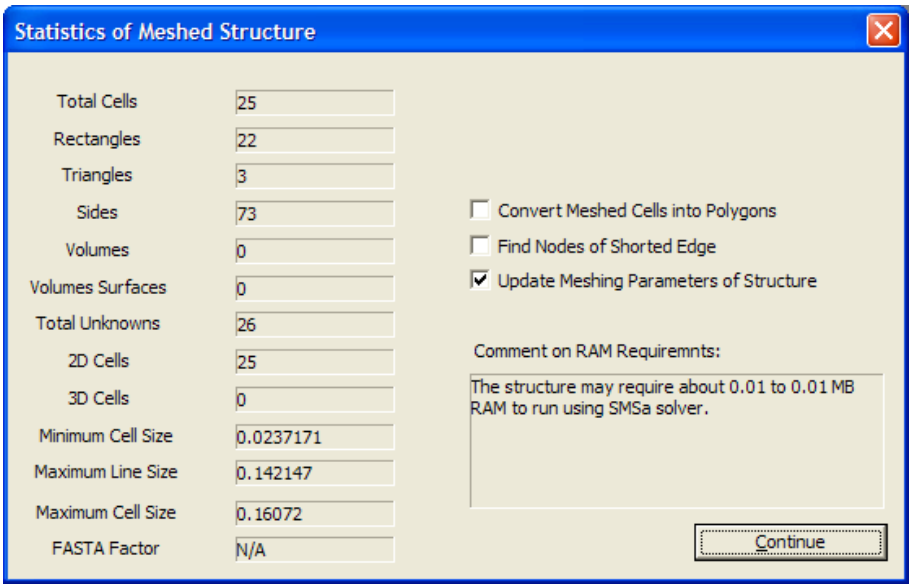


Figure 3.14 The meshing results in tabular format.

Step 26 Please choose Contemporary meshing scheme. It is a new scheme implemented in IE3D 11. It is supposed to create less cells and more regular edge cells for higher efficiency and higher accuracy.

Remember to select AEC Layer = No to disable the Automatic Edge Cells, which we will discuss later. Select **OK** to accept the default values. A pop-up window comes up to indicate the meshing process. After the process, MGRID will prompt you with the dialog "Statistics on Meshed Structure (see Figure 3.14). Only 25 cells are created in the meshing process with Maximum Line Size = 0.142 mm and Maximum Cell Size = 0.1607 mm. Maximum Line Size is the maximum size of any adjacent vertices of all the cells. Maximum Cell Size is the maximum distance between any two vertices of a cell for all the cells. You may see parameters such as

"Volumes" and "Volume Surfaces". Those are for the finite dielectrics. There are no finite dielectrics defined in this structure.

The meshed cell density is determined by the product of the Fmax and Ncell, as well as the AEC. If you need to change the cell density, you can change either Fmax or Ncell for the same result as long as the product of Fmax and Ncell is the same. If you choose a too big Fmax or Ncell such that the structure becomes too big and the estimated minimum cell number exceeds the Warning Limit, MGRID will issue a warning on it to prevent the case of endless meshing due to large number of cells created. The Warning Limit can be changed in **Parameters->Optional Parameters** dialog.

There are additional information and options available on the IE3D 14. It indicates "The structure may require about 0.01 to 0.01 MB RAM to run using SMSa solver. This is a tiny structure for IE3D because the major data size is the matrix and it is about 0.01 MB. The SMSa solver the default matrix solver for IE3D. It is very efficient and accurate and it makes use of multi-CPUs for multi-core systems.

On the dialog, you also see an option "Update Meshing Parameters of the Structure". For the current structure or any un-simulated structure, the default for this option is checked. If you select CONTINUE, MGRID will update the meshing parameters with the ones defined in the Meshing command. If you uncheck it and select OK, MGRID will display you the meshing results only. It will not change the meshing parameters of the structure. Starting from IE3D 12, the simulation results or the s-parameters are saved into the current geometry file. Changing of the geometry including the meshing parameters will cause removal of the simulation results. On IE3D 12, users can't perform meshing again after a structure is saved. Otherwise, the results will be lost in the geometry. If you save the geometry file after meshing, the simulation results will be lost forever. If you change the geometry and you don't want to lose the results, you should not save the changed geometry. It is inconvenient, even it is not impossible, to check the meshing of a geometry after a simulation. On IE3D 14, we make it easier. You can still perform a meshing with different meshing parameters. If there are results in the current geometry, the default for the option "Update Meshing Parameters of the Structure" will be un-checked. After you select CONTINUE, the meshing parameters of the structure will not be updated if you leave the option unchecked.

Step 27 Select **Continue** button in the **Statistics of Meshed Structure** dialog. MGRID will show you the meshed structure in a separate 3D Meshing View window. IE3D 14 no longer displays the meshing on the Polygon Editor window. The menu system of MGRID will be changed if you select the 3D Meshing View window. IE3D 14 adopts a new 3D view with many powerful commands and features (in Manipulate, Options and Windows menu). For example, you can switch between Perspective View and Orthographic View easily by selecting the corresponding menu items in the Options dialog. You see the infinite substrates are displayed as some frames on the 3D view window. The size of the frames is defined as the Display Margin in the Substrates section of Basic Parameters or the menu item Parameters->Substrate Display Parameters or Options->Structure Display Parameters->Substrate Display Parameters. On IE3D 14, the ports are also displayed on the 3D view window. The handling of 3D view may be different too. For example, to zoom on the 3D view, you can just slide the mouse wheel. To pan the 3D view, you press down SHIFT while you move the mouse. In the earlier versions, you hold down the right mouse button while you move the mouse. Clicking right mouse button is reserved for bringing up a popup menu for other commands on IE3D 14. To zoom a portion of the structure, you press down CTRL while you window the portion.

As you can see, rectangular cells are used in the regular region while triangular cells are used in the irregular region. The efficiency of rectangular cells and the flexibility of triangular cells are smartly combined in IE3D.

To get a hard copy of the meshed geometry, you can select **File->Print** command while the 3D Meshing View is the active window. To get a snap shot of the top view, please select File->Save To Bitmap Files command. MGRID will allow you to save the picture into a bitmap file. You can also select ALT+PrtScrn buttons to capture the MGRID window or the 3D Meshing View window into the clipboard. Then, you can paste it onto the MS-Paint program in the Accessory group or other graphic application programs.

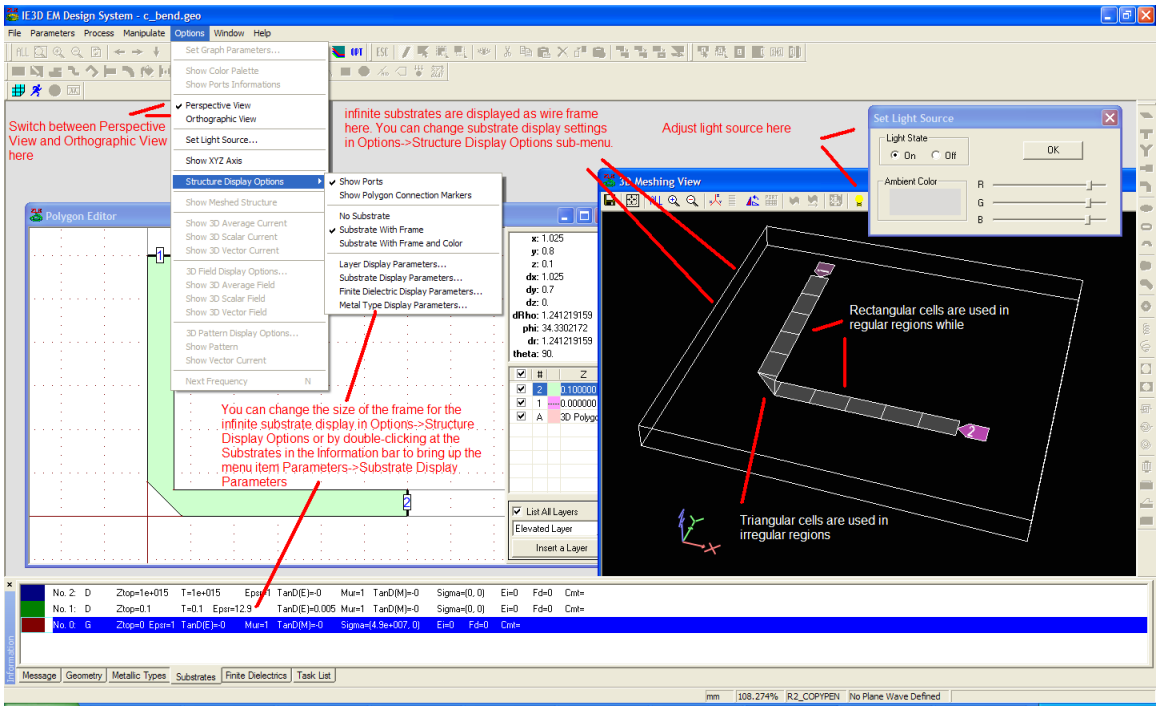


Figure 3.15 The meshed geometry displayed on 3D Meshing View and the different options.

Section 3.7 Electromagnetic Simulation and Adaptive Intelli-Fit (AIF)

We have just demonstrated how to view the meshed structure on MGRID. As it is mentioned earlier, meshing on MGRID is not necessary for a simulation. It is just for display only. We can perform an EM simulation without the meshing process on MGRID. We will show how to setup a simulation in this section.

Step 28 Select the Polygon Editor window and you will see the menu system switch back to the one for editing. Select **Process->Simulate** command. MGRID prompts you for the Simulation Setup dialog (see Figure 3.16). This is a shared dialog for Simulation Setup and Optimization Setup. Some controls for EM tuning and EM optimization may be disabled because MGRID detects no optimization variables are defined. The Simulation Setup dialog allows you to define and change Meshing Parameters, Accuracy Control Parameters and the simulation frequency points.

There is a Capture button for you to capture the frequency points from a simulated result saved in the .sp (.spt or .s2p) files. There is another button called “Retrieve”. It allows you to retrieve most of the data for the Simulation Setup dialog from some previous created simulation input file with extension .SIM. The data retrieved including the frequency points and optimization

objectives (for optimization), etc. The .SIM file is created after a simulation setup process. There are two files involved in an IE3D simulation: (1) .GEO file storing the geometry description; (2) .SIM file storing the simulation control parameters such as frequency points, matrix solver, and most of the parameters defined in the Simulation Setup dialog. Starting from IE3D 14, the data in the .SIM file has a copy inside the .GEO file. It is possible future versions of IE3D may abandon the .SIM file because all its data has a copy in the .GEO file.

The Meshing Parameters are important ones and they are also accessible from this dialog by the **Automatic Edge Cells** button and the **Meshing Alignment** button. The most important parameters you need to enter are the frequency points for the simulation.

IE3D needs to solve matrices. The time for a solving matrix is one of the two most important parts in an IE3D simulation (The other part is the time for filling the matrix). We have implemented multiple matrix solvers for different purposes. In IE3D 11, the default matrix solver is the Adv. Symmetric Matrix Solver (SMSi). It does not support multi-thread and multi-CPUs. We have implemented the Adaptive Symmetric Matrix Solver (SMSa). SMSa can utilize multiple threads and multiple-CPUs. It can be more than 4 times faster than SMSi for large structures. However, SMSa may take about twice as much memory as SMSi. An adaptive scheme is implemented into SMSa. In case no enough RAM is available, SMSa automatically switches to the SMSi scheme. The default matrix solver is defined in the **Parameters->Optional Parameters** command. More discussion on matrix solvers is available in Chapter 12. The default of default is SMSa matrix solver.

In most cases, you are interested in the frequency responses of the s- (y-, z-) parameters. If you are simulating the structure with many frequency points, you should always enable the Adaptive Intelli-Fit (AIF). The AIF is an extremely efficient and robust way to get fast and accurate frequency response. AIF can adaptively select some of the defined simulation frequency points, simulate them and extrapolate at other frequency points from the solved frequency points. It will automatically detect convergence. Normally, it can save 90% of the simulation time with guaranteed accuracy at the extrapolated frequency points. You can feel safe to use it without worrying about whether the extrapolated data are accurate.

In case you need to save the current distribution for visualization and/or the radiation pattern, you can check the **Current Distribution File (.Cur)** and **Radiation Pattern File (.Pat)**. It will save the current distribution and/or radiation pattern at the simulated frequency points. When AIF is enabled, you will not know which frequency points will be simulated (except the Start, Mid and End points). In case you want to make sure some specific frequency points will be simulated with AIF enabled, you can check the check boxes before the specific frequency points in the list box of Frequency Parameters.

In IE3D 12, you have the option "Save S-Parameters into FastEM Data". This option allows IE3D to save the s-parameters into the geometry file (.geo) in addition to the s-parameter file (.sp). You can do integrated s-parameter visualization and processing on MGRID 14. Automatic lumped element extraction is also implemented into IE3D 12. By default, "No Lumped Model Output" is selected. If you select it, IE3D will output the frequency dependent lumped element model for you automatically.

Traditionally, IE3D s-parameter results are saved into a Touchstone compatible file (.sp), and they are displayed and processed on MODUA, a bundled circuit simulator. On IE3D 14, you are able to visualize and process s-parameters on MGRID. Most of the s-parameter processing functionalities of MODUA are integrated into MGRID, while the circuit simulation part is not.

Circuit simulation is better implemented on MDSPICE. MDSPICE will be eventually integrated with IE3D to cover the full functionalities.

If you want to display the s- (y-, z-) parameters immediately after simulation without selection, you can select the Define Graphs button on Simulation Setup dialog (see Figure 3.16) to define some graphs. Then, after simulation, the results will be automatically display. If you select Invoke MODUA in Post Processing, MODUA will also be invoked to display the s-parameters saved into the .sp file.

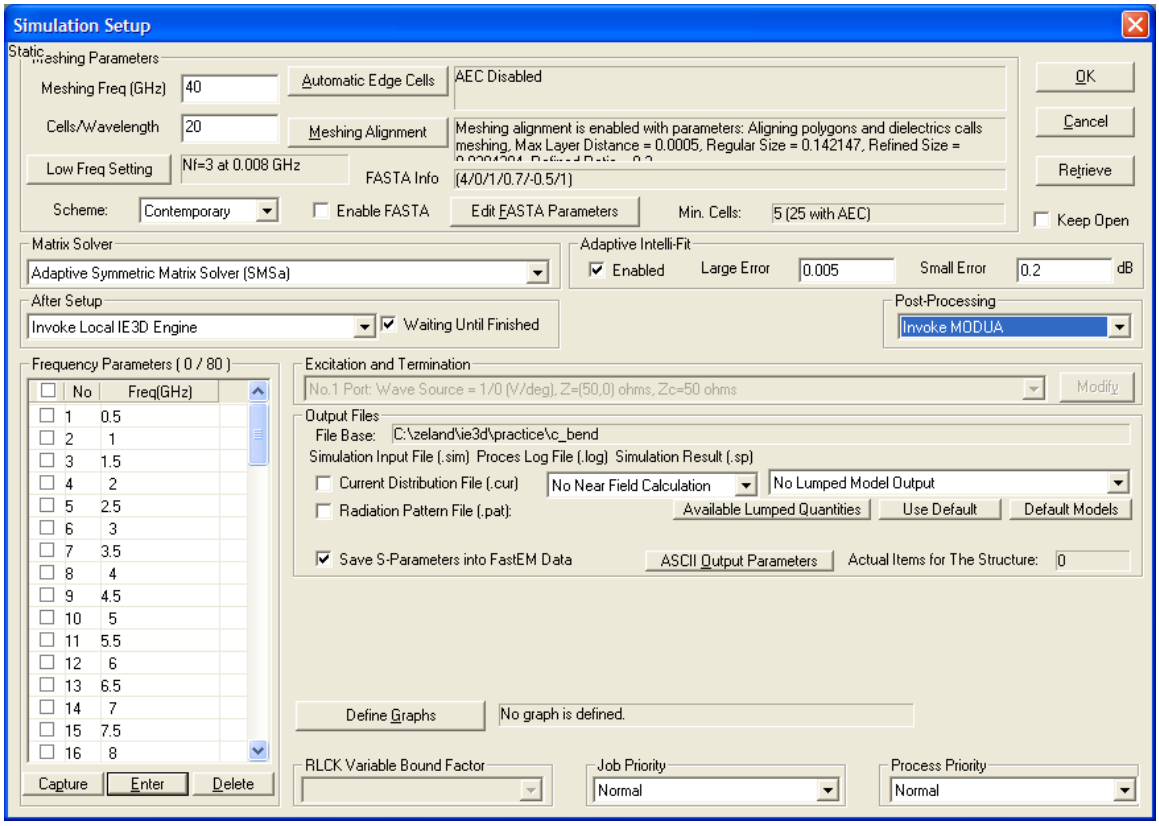


Figure 3.16 The Simulation Setup dialog with frequency points defined.

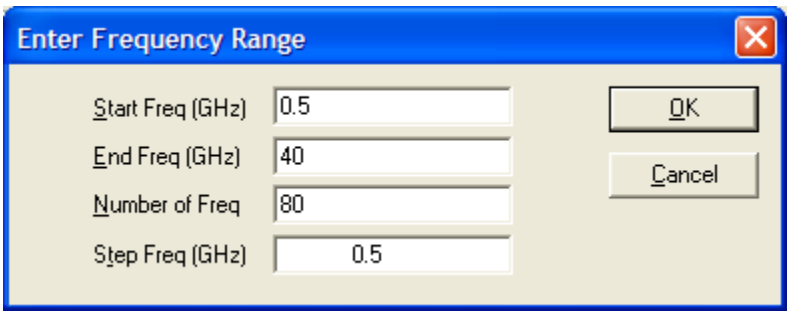


Figure 3.17 The Enter Frequency Range dialog.

Step 29 Please select the Enter button in the **Frequency Parameters** section. MGRID will prompt you for the **Enter Frequency Range** dialog (see Figure 3.17). Please enter Start Freq = 0.5 GHz,

End Freq = 40 GHz and Number of Freq = 80. MGRID will automatically calculate the Step Freq = 0.5 GHz. Select OK to continue. You will get the exact frequency points in the **Frequency Parameters** list box. You are allowed to enter multiple frequency ranges. MGRID will automatically sort the frequency points for you.

Table 3.1 The different choices for the **After Setup** and **Wait Until Finished** option.

After Setup	Wait Until Finished	Description
Invoke Local IE3D Engine	Checked	It is the default mode. MGRID will invoke the IE3D engine to perform simulation. It will use the local CPU power for the simulation. It will wait until IE3D engine to finish and then MGRID will display the results. No extra license is required.
Invoke Local IE3D Engine	Un-checked	MGRID will submit the simulation job to the JobsManager. It may invoke JobsManager in case JobsManager is not started. JobsManager is implemented for: (1) Managing network distributed simulation jobs; (2) Replacing IE3D Simulation Queue in the older versions for scheduled simulations and optimization on local computers. When you choose this option, the simulation job is sent to JobsManager. You can do many simulation setups and send the jobs to JobsManager for scheduled IE3D simulations. At a specific time, there might be multiple JobsManager running on your account. It is ok because they are synchronized automatically. No extra license is required.
Create .SIM File Only	N/A	It will create the .sim file for you. You are able to use the .sim file and the .geo file to perform IE3D simulation later. MGRID will not do it automatically. You can use this way to call the IE3D engine externally for some purposes. For example, you can use it for EM optimization using external optimizer. Please read the Appendix for more information on using external optimizer. No extra license is required.
Invoke IE3D Network Distributed Simulation	Checked	MGRID will send the job to JobsManager. JobsManager will break the job into multiple single frequency small jobs and submit it to ZDS (IE3D Distributed Service). ZDS will distribute the small jobs to the network for distributed IE3D simulation (or optimization). MGRID will be waiting for JobsManager to finish the simulation. The IE3D simulation is actually not done on the local computer. However, it is as if you are running the job locally and MGRID will display you the results after finish. It requires ZDM licenses for distributed EM simulation and optimization. If you have multiple ZDM licenses on the network, you might be able to speed up the simulation by a big factor.
Invoke IE3D Network Distributed Simulation	Un-checked	It is similar to the case with "Wait Until Finished" checked except MGRID will not wait for the simulation results. It requires ZDM licenses for distributed EM simulation and optimization.
Invoke IE3D Network Simulation	Checked	It is similar to the case "Invoke IE3D Network Distributed Simulation". However, the simulation job will not be broken down into single frequency jobs. Instead, the job will be distributed into one ZDM license on the network to run the simulation by ZDS.
Invoke IE3D Network Simulation	Un-checked	It is similar to the case with "Wait Until Finished" checked except MGRID will not wait for the simulation results.

Step 30 Please select Invoke Local IE3D Engine in the After Setup combo box and check **Wait Until Finished**. Starting from IE3D 12.1, you have more choices on how you want MGRID to perform an IE3D simulation. The different choices are documented in Table 3.1

Step 31 Select OK to continue. MGRID will invoke the IE3D engine in the background to perform the simulation while the progress is monitored on the IE3D dialog.

IE3D dialog will display the intermediate simulation information (see Figure 3.18). If you want to see whether the IE3DOS.EXE is running, you can activate Task Manager of Windows and find the IE3DOS.EXE from the Processes tab. You can see how much memory and how much CPU resource it is using.

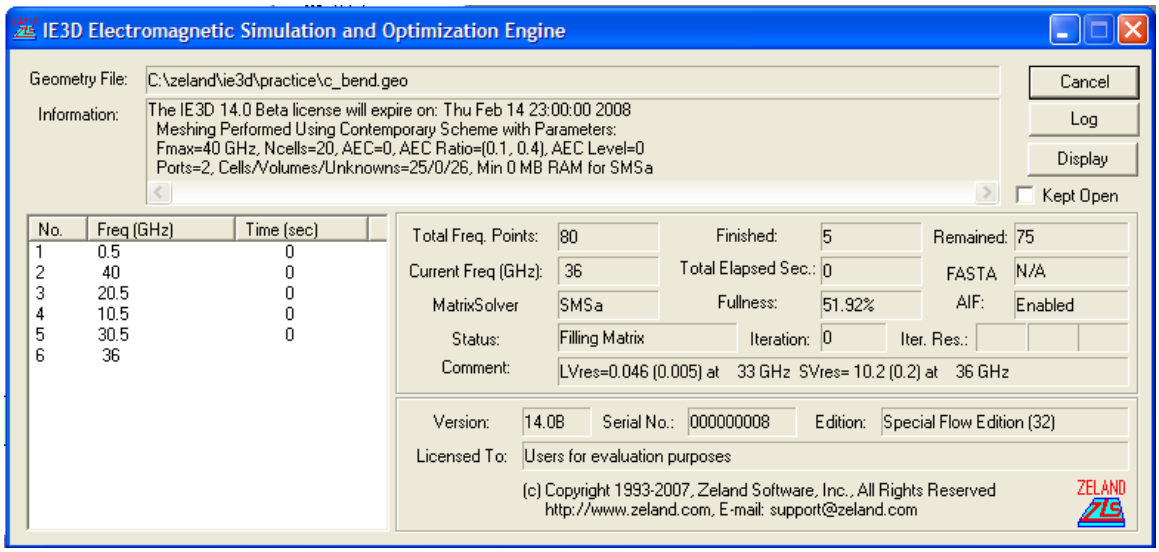


Figure 3.18 The IE3D dialog displaying the intermediate simulation data.

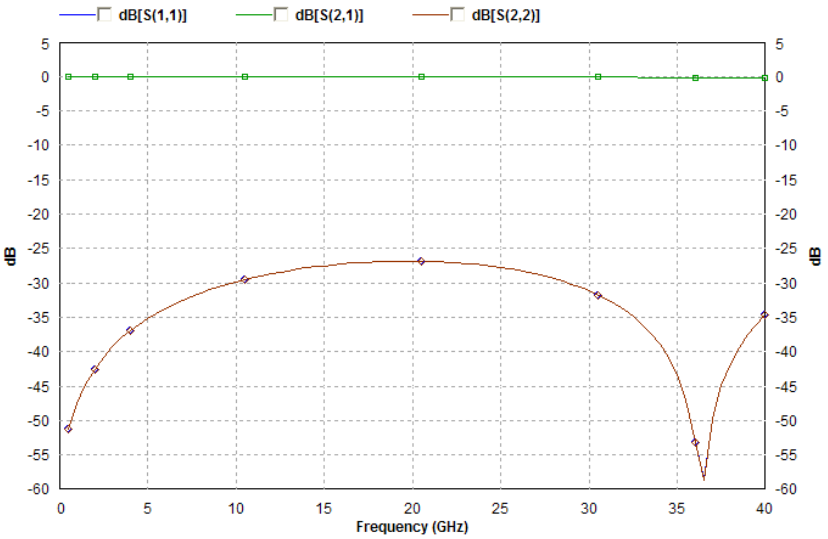
Table 3.2 The explanation on some of the intermediate parameters in the IE3D dialog in Figure 3.18.

Item	Description
Information	This item describes that the structure has 2-ports and it is meshed into 25 cells and 26 unknowns. Minimum 0 MB RAM is required for the SMSa matrix solver.
Total Freq Points	It indicates total 80 frequency points are required (Not all of them will be simulated if AIF is enabled).
Finished	It indicates 5 frequency points are finished.
Remained	It indicates total 75 frequency points are not simulated.
Matrix Solver	It indicates that the SMSa matrix solver is used.
Fullness	It indicates the matrix fullness is 51.92%.
Status	It indicates the status is “Filling Matrix”. For every frequency point, it will go through “Filling Matrix” and “Solving Matrix” processes. IE3D dialog will display the progress of “Filling Matrix”. It cannot display the progress of “Solving Matrix”.
Iteration	This is not useful for the SMSa matrix solver. It is only meaningful for iterative matrix solvers.
Comment	It displays “LVres=0.046 (0.005) at 33 GHz and SVres = 10.2 (0.2) at 36 GHz”. It is the AIF convergence information. It means that the current maximum large value error for AIF is 0.046 at 33 GHz and the convergence criteria are 0.005. The current maximum small value error is 10.2 at 36 GHz and the convergence criteria are 0.2. The criteria “0.005” and “0.2” are the “Large Error” and “Small Error” defined in the AIF section of the

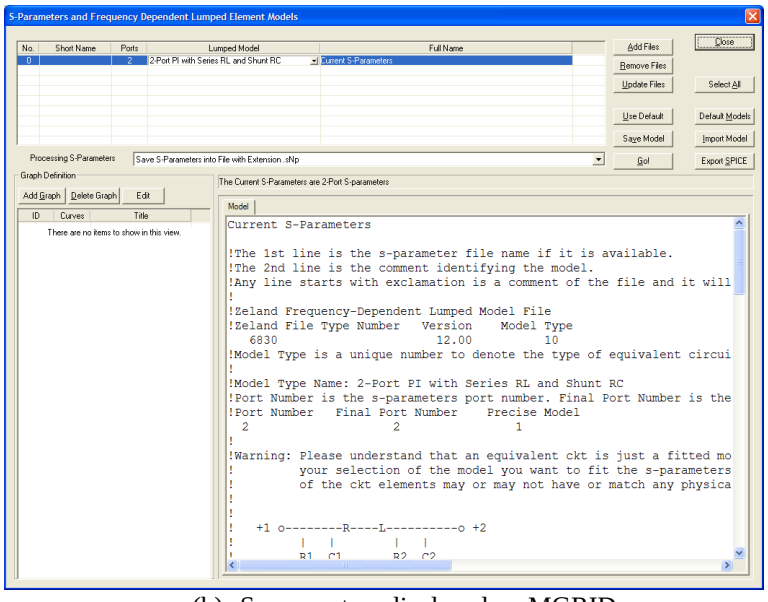
	Simulation Setup dialog (see Figure 3.16). The default values are fully tested and you do not need to change them. The LVres and SVres values are for AIF convergence only. They are not related to the accuracy of the simulation.

For this particular example, the whole IE3D engine may take less than 1 second. You may not be able to see how the IE3D dialog looks like and see the IE3DOS.EXE information from the Task Manager of Windows before it finishes.

A snapshot of the IE3D dialog in Figure 3.18 is captured in a very short time. As you can see, 5 frequency points are done while it takes less than 1 second. There are many parameters displayed in the dialog. Their meanings are explained in Table 3.2.



(a) S-parameters displayed on MODUA.



(b) S-parameters displayed on MGRID.

Figure 3.19 The simulated s-parameters of the microstrip bend structure.

After the simulation, IE3D will save the s-parameters into .\ie3d\practice\output\c_bend.sp and .\ie3d\practice\output\c_bend.spt files. Because you selected "Invoke MODUA" in Post-Processing in the Simulation Setup dialog, it will invoke the MODUA to display the s-parameters (see Figure 3.19a). Starting from IE3D 14, it is not necessary to invoke MODUA for processing s-parameters. You can do it on MGRID or IE3D EM Design System internally. You have checked Save S-Parameters into FastEM Data in the Simlation Setup dialog, the s-parameters is also saved into the .geo file. It automatically invoked the Process->S-Parameters and Lumped Equivalent Circuit command for post-processing s-parameter as shown in Figure 3.19b. We will call this dialog as S-Parameters Processing dialog. As you can see, the default 2-Port Series RL and Shunt RC equivalent circuit is automatically found. You can select Save Model to save the data into an ASCII file. Please understand that the lumped element model is just a fitted model. It may not have any physical meaning. It is your responsibility to pick the right physical model and IE3D will try to find the values of the lumped elements for your model.

Most of the s-parameter processing capabilities on MODUA and MDSPICE are implemented into the Processing S-Parameter section in the dialog. For example, you can save the s-parameters into .sNp format with the default file extension for Touchstone format. You can perform conversion from single-ended s-parameters to differential ones, and from differential ones to single-ended ones. You can also check passivity of the s-parameters, etc. More information on s-parameter processing can be found from Appendix BG.

You can select Add button in the Graph Definition on the left of the dialog to define multiple graphs to display the s-, y-, z-parameters and the lumped equivalent circuit data. With this new feature on IE3D, displaying the s-parameters on MODUA is no longer necessary.

Multiple files are involved in an IE3D simulation. The involved files are documented in the Appendix AN. As it is documented in Appendix AN, when AIF is enabled, the .spt file saves the s-parameters at the frequency points truly simulated by the IE3D engine while the .sp file saves the s-parameters at all the frequency points. The .sp file contains extrapolated data and its accuracy is guaranteed. If AIF is disabled, the .spt file will not be created and the truly simulated frequency points are saved into the .sp file.

You may see some markers on the curves on the MODUA display. They are meaningful because they indicate which frequency point has true EM simulation from IE3D. Other frequency points are accurately obtained from the simulated frequency points using some extremely robust interpolation scheme with AIF. For this particular example, the EM simulation is actually performed at 7 frequency points (0.5, 2, 4, 10.5, 20.5, 30.5, 36 and 40 GHz) and the results are in the .spt file. All the other frequency points are interpolated data with guaranteed accuracy saved in the .sp file and the .geo file.

You can use MODUA to display S, Y and Z-parameters in different forms (Cartesian graphs, data lists, and Smith Chart). Please read Appendix AO on using the MODUA for different display. Basically, you can select Control->Define Display Data, Control->Define Display Graph and Control->Define Display Smith Chart to display S, Y and Z-parameters in different form after you have simulated a circuit or displayed an s-parameter on MODUA. For example, while we have MODUA displaying the dB[S(1,1)] , dB[S(2,1) and dB[S(2,2)], please select Control->Define Display Graph command on MODUA. Please select "dB and Phase of S-Parameters" in "Display Parameters" and select OK. Uncheck "dB[S(2,2)]". Check "Ang[S(1,1)]" and "Ang[S(2,1)]". Make sure only 4 items ("dB[S(1,1)]", "dB[S(2,1)]", "Ang[S(1,1)]" and "Ang[S(2,1)]" are selected and elect OK. MODUA will display the frequency responses of the 4 items as shown in Figure 3.20.

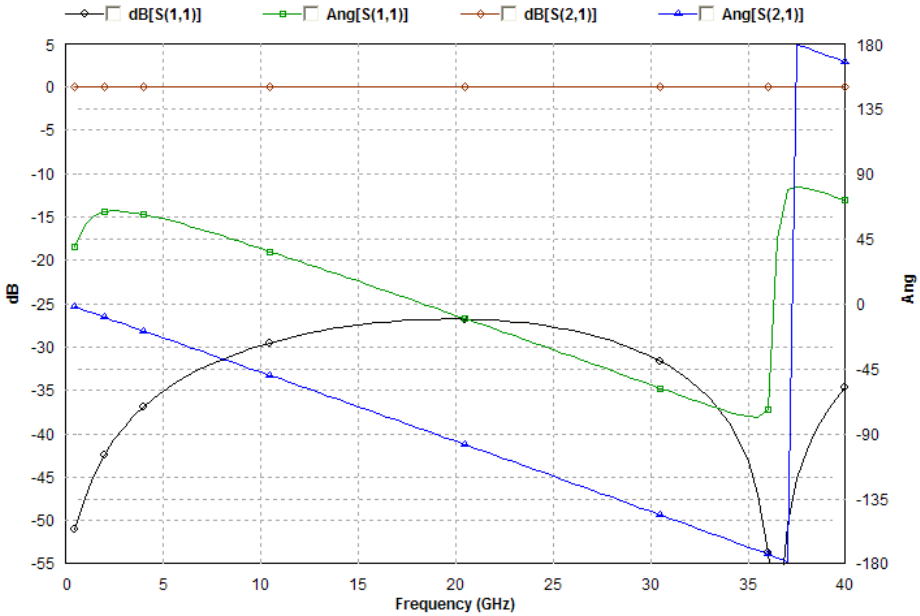


Figure 3.20 The dB and Ang of different s-parameter items displayed on MODUA.

You can do much display and analysis on the s-parameters on MODUA. For example, you can define markers and find the exact value of an item at a specified frequency. For detail information, please read Appendix AO.

Starting from IE3D 14, it may not be necessary to display and processing the s-parameters on MODUA. You may be able to do a better job on MGRID. We will show you the feature in the next sections.

Section 3.8 High Accuracy Simulation Using Automatic Edge Cells (AEC)

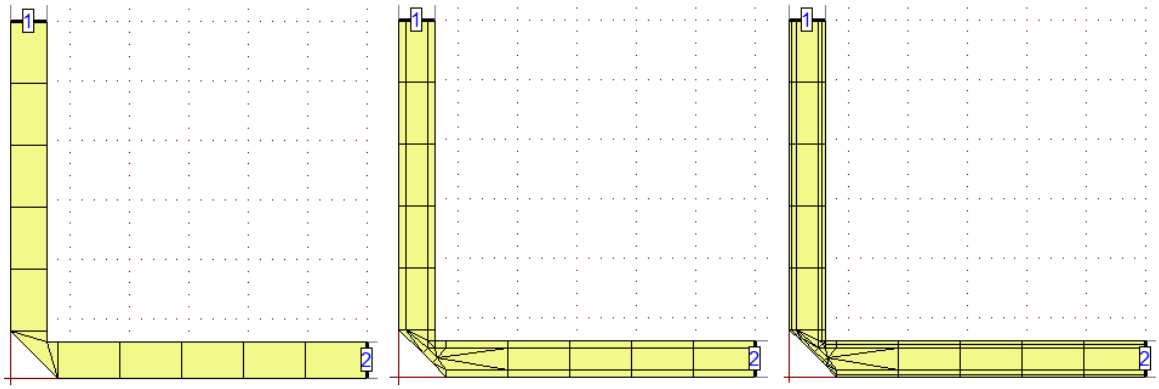
Are the results displayed in Figure 3.20 accurate? The answer can be yes or no depending upon your requirements for accuracy.

We have claimed that the AIF scheme yields "accurate" result over a wide bandwidth by just simulating a few frequency points. However, when we say "accurate", we do not mean that we have proved the result is accurate compared to the measured data or the converged data. When we say the AIF provides "accurate results", we just mean that the result of AIF will perfectly match the true simulation results under the same condition (same meshing and same accuracy control parameters). Whether a simulation is accurate or not, we need to do many validations by comparison to measured data or converged data. In the past 15 years, we have been validating the IE3D by comparing its results to measured data or by performing convergence studies. We have come up with different rules for accurate electromagnetic modeling using the IE3D. We will discuss different accuracy enhancement methods in Chapter 12 since it may need a user to read more chapters before the user can understand the schemes. In this section, we would like to touch base with the Automatic Edge Cells (AEC), a simple and efficient way to improve simulation accuracy. Generally, AEC allows a novice user to obtain expert results.

Step 1 Select the Close button in the "S-Parameters and Frequency Dependent Lumped Models" or the S-Parameters Processing dialog on MGRID. Select **File->Save As** to save the geometry as "**.\ie3d\practice\c_bend3.geo**".

Step 2 Select **Process->Display Meshing**. We are going to show how the Automatic Edge Cells (AEC) affects meshing. The Display Meshing process is not necessary for a simulation. We just want to show you the meshing. Otherwise, we can go directly to Step 4.

MGRID will prompt you for the Automatic Meshing Parameters (see Figure AL.1).



(a) AEC Layer = No (b) AEC Layer=1, AEC Ratio=0.1 (c) AEC Layer = 2, AEC Ratio=0.1

Figure 3.21 The meshed structure with different AEC Layer values.

Step 3 Select **AEC Layer** = 1. Make sure the **AEC Ratio = 0.1**. MGRID will automatically calculate the AEC Width = 0.0142147 mm (see Figure 3.22). Again, starting from IE3D 11, we allow you to enter the AEC Ratio instead of the AEC Width. In this way, the AEC Width will be in normally range and it will avoid accidentally problem caused by a user forgetting to define the correct AEC Width value after the Fmax and Ncell are changed.

Select OK to continue. MGRID will prompt you the Statistics on Meshed Structure. It indicates that total 81 cells and 134 unknowns are created. Select CONTINUE button. MGRID will display the meshed structure (see Figure 3.21b). You can see some narrow cells are created along the edges. You can repeat the Process->Display Meshing multiple times with different AEC Layer selections; you will get different meshing results. When you choose different AEC Layer value, please choose the Multi-Layer Ratio = 0.4. Normally, you should not change this value.

When you get the Statistics on Meshed Structure, please pay attention to the option "Update Meshing Parameters of Structure". The default for this time is un-checked. Please remember that the default was checked when we do it the 1st time. Why is it un-checked this time? The difference is that there was no s-parameters in the .geo the 1st time we do the meshing after we created the geometry. This time, we have simulated the structure and the s-parameters are saved in the geometry. If we change the parameters of the structure (including the shapes and the meshing parameters), the results will be lost. However, we can still do meshing to check how the meshed structure looks like with other meshing parameters. By default, we will not update the meshing parameters. Please leave it un-checked.

Step 4 Select **Process->Simulate**. If you check the Automatic Edge Cells section, you will see AEC is not enabled because we did not update the meshing parameters in the last step by checking the "Update Meshing Parameters of Structure". Please select Automatic Edge Cells button to bring out the Meshing Parameters dialog. Please select AEC Layer = 1. Select OK to go back to Simulation Setup dialog. Select OK again. MGRID will warn you that s-parameters results are available and re-simulation to cause removal of the existing s-parameters results. Please select

YES to continue because we don't need the old results. It will take 1-2 seconds to finish the simulation. MODUA is invoked to display the results and the "S-Parameters and Frequency…" dialog on MGRID is also invoked. Comparing the new results with the old one and you will see the dB[S(1,1)] and Ang[S(1,1)] are different while the S(2,1) items are almost the same. |S(1,1)| is a small quantity. Its values are more affected by simulation accuracy.

Step 5 Select the Close button in the S-Parameters Processing dialog on MGRID. Select **File->Save As** to save the geometry as "**.\ie3d\practice\c_bend5.geo**".

Select Process->Display Meshing. Select **AEC Layer** = 2. Make sure the **AEC Ratio = 0.1**. Make sure it is "Contemporary" meshing scheme. Select OK. MGRID will show you the Statistics on Meshed Structure dialog. Please check "Update Meshing Parameters of Structure". Select CONTINUE button. This time, MGRID warns you that updating the meshing parameters will cause removal of the s-parameters. Select YES to continue. We will get what is shown in Figure 3.21c. As you can see, the strip is divided into 5 cells with 2 narrow edge cells on each side. Total 139 cells and 247 unknowns are created.

Step 6 Select **Process->Simulate**. The Simulation Setup dialog comes up and you will see the AEC is enabled with **AEC Layer** = 2 because we have updated the meshing parameters in the last step. Please select OK to continue. MGRID will invoke the IE3D engine internally to do the simulation. It takes 2-3 seconds to finish it. The results will be saved in: and .\ie3d\practice\output\c_bend5.sp.

Section 3.9 S-Parameter Visualization and Processing

Traditionally, s-parameters visualization and process are on the MODUA. You also perform circuit simulation on a circuit consisting of s-parameter black boxes and lumped elements such R, L, C and mutual L on MODUA. On IE3D 14, we have implemented more s-parameter visualization and processing capabilities into MGRID. MODUA is no longer the default visualization tool even though we still need to perform circuit simulations and mixed EM and circuit simulations on MODUA.

We will discuss how we can use MGRID for s-parameter visualization and processing. With s-parameters processing integrated into MGRID, we normally don't need MODUA for s-parameter processing. By default, we suggest users to select "No Display" for Post-Processing. In case, you want to display the s-parameters immediately after simulation, you should select Add Graphs in Simulation Setup to pre-define some graphs before simulation starts. Certainly, we can always define graphs after simulations.

Step 1 Run MGRID and open file: ".\ie3d\practice\c_bend.geo".

It is a simulated structure while the results are automatically saved into the .geo file. Please note that any change to the structure and meshing parameters will cause removal of the s-parameters in the data. When you do some potential change, MGRID will warn you anyway. In case you change it and the s-parameter data is lost, you will not be able to access the data from MGRID. However, it is still available in the file if you do not save the change into the file.

Step 2 Select Process->S-Parameters and Lumped Equivalent Circuit command or Window->S-Parameters Display->Define Plots (Note: We provide the commands with the same functionalities for your convenience). The "S-parameters and Frequency Dependent Lumped Element Models" or S-Parameters Processing dialog comes out (see Figures 3.22 or 3.19b).

Please pay attention to the message at the middle right of the dialog "The Current S-Parameters are 2-port s-parameters". It is for the selected item or the only No.0 item in the list box. You are

allowed to define the graphs in Simulation Setup dialog or in Window->S-Parameters Display->Define Plots even though the s-parameters may not be available. In such a case, MGRID will indicate to you that the s-parameters are fake ones for defining the graphs only in this message.

Step 3 Please double click at the Short Name section of the No.0 item in the list box to change its Short Name from "" to "AEC Layers = 0" (see Figure 3.22).

Step 4 Select "Add Files" button on the right. Please select both c_bend3.sp and c_bend5.sp in the directory: .\ie3d\practice\output by holding down the Ctrl key. The two files will be added into the list box in the dialog. MGRID will automatically detect the strings "c_bend3" and "c_bend5" for their short names, respectively. Please double click at the Short Name section of the two items to change them to "AEC Layers = 1" and "AEC Layers = 2".

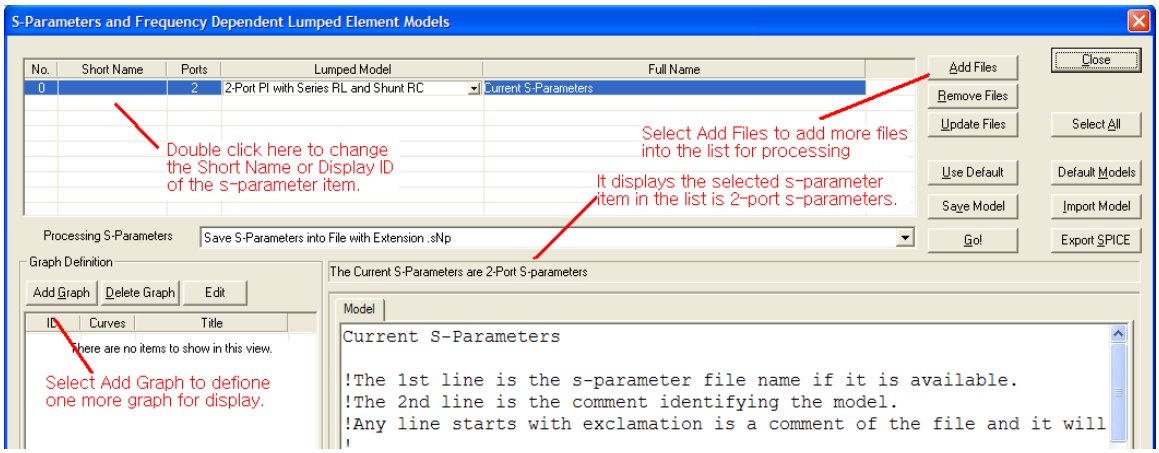


Figure 3.22 The S-Parameters Processing dialog.

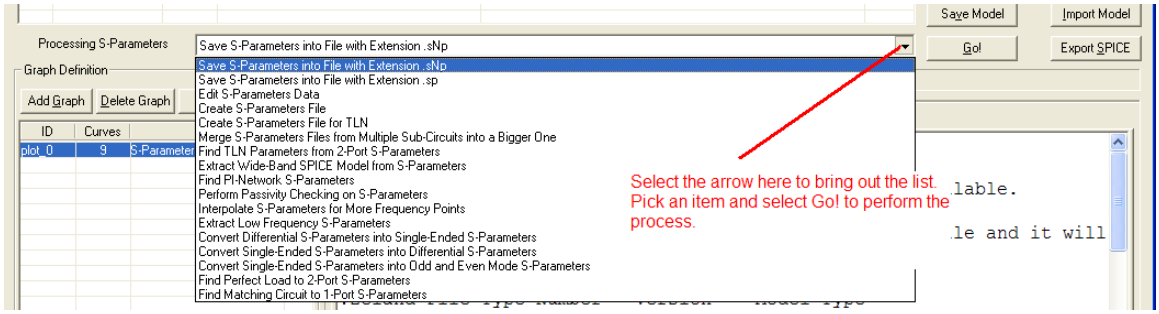


Figure 3.23 The S-Parameters Processing dialog for s-parameter processing.

Step 5 Select the No.0 S-parameters item in the top list. Select the downward button on the right for the combo box in Processing S-Parameters (see Figures 3.22 and 3.23). It will bring out a list of possible processing features for the selected s-parameters or others.

Step 6 Select "Perform Passivity Checking on S-Parameters" from the list. Select "Go!" button. MGRID will perform the passivity checking for you and it will create a dialog for the report (see Figure 3.24). It tells you whether the s-parameters data is passive at a specific frequency in the list. When the "Minimum Value" is non-negative, the s-parameters are passive at the frequency. For this particular example, MGRID detect the circuit passive for all the 80 frequency points. You can select Save Report to save the data into an ASCII file. In case, there are some non-

passive points, you have the option to enforce passivity at the points. MGRID will try to adjust the s-parameters to enforce passivity. Certainly, it will change the s-parameters and the accuracy of the final results can't be guaranteed.

For a structure without active devices, it certainly should be passive. However, it is impossible to always guarantee an EM simulation result will always be passive even though non-passive results may happen occasionally. There are multiple reasons: (1) There are numerical errors involved. (2) There might be positive feed back loop between the source and your open structure and the positive feed back loop may cause non-passive s-parameters. There are more chances for such a case to happen especially for extension ports if the ports are not defined properly so that the port extensions, which are supposed to be not part of the circuit, are coupled with the structure. In fact, it is impossible to de-couple the port extensions and the structure. We just need to avoid the coupling is to be significant so that it will affect our results.

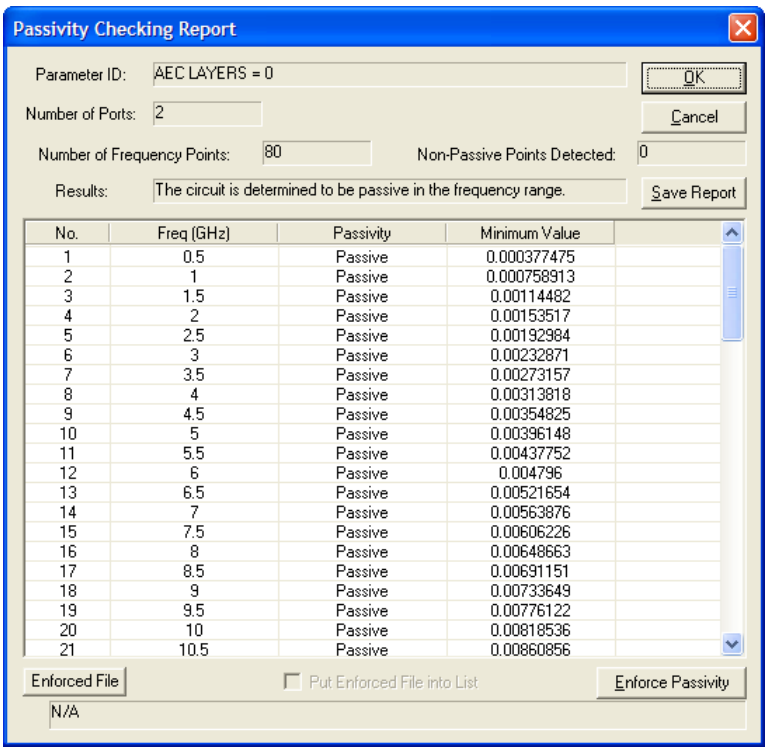


Figure 3.24 The passivity checking results.

Step 5 Please select OK to close the passivity checking result dialog.

There are many s-parameters processing features in the list and you can explore them. There will be more implemented in the future.

Step 6 Select Add Graph on the left. MGRID will prompt you for the Graph Type. Please select S-Parameters from the list. MGRID will set the default title of the graph as "S-Parameters Display" and you can change it to whatever you like. Select OK and MGRID will prompt you for the display selection (see Figure 3.25). Please click at the No.0 item in the list box "Root=Current S-Parameters". Check the items "dB[S(1,1)]", "dB[S(2,1)]" and "Ang[S(2,1)]". The repeat the selections for the "c_bend3.sp" and "c_bend4.sp". Select OK to go back to the S-Parameters Processing dialog. The defined graph "plot_0" is listed in the list box. The preview window for

"plot_0" is created as a tab page on the right. Click at the tab will show you the results. The s-parameters from the three files are compared (see Figure 3.26).

You can define more graphs if you like. Each additional graph will correspond to a tab page on the right. You certainly can check the data in the tab pages even though they are small. They are actually for your preview only and they don't contain the tables for the user defined markers for measurement.

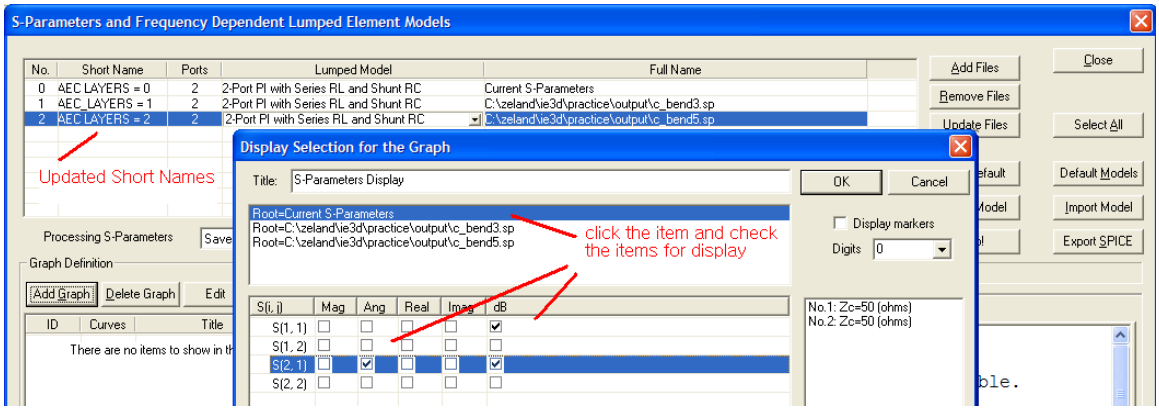


Figure 3.25 The Display Selection for the Graph dialog.

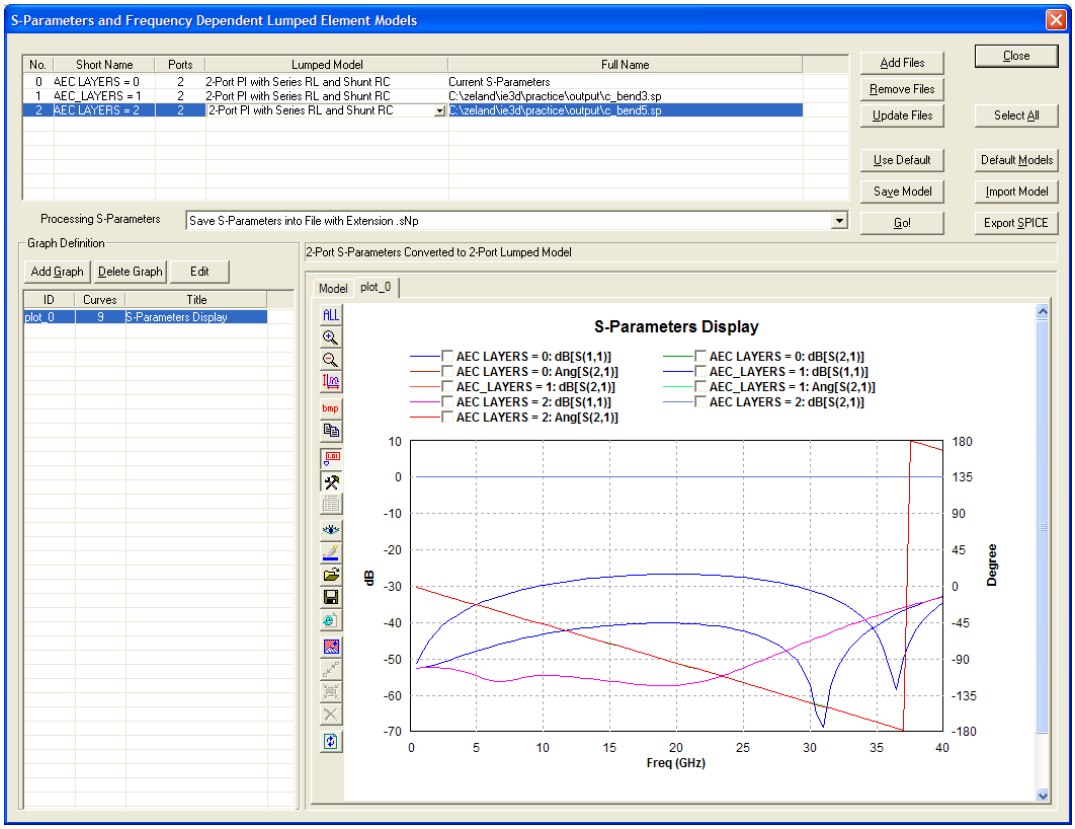


Figure 3.26 The defined s-parameters plot is previewed in the dialog.

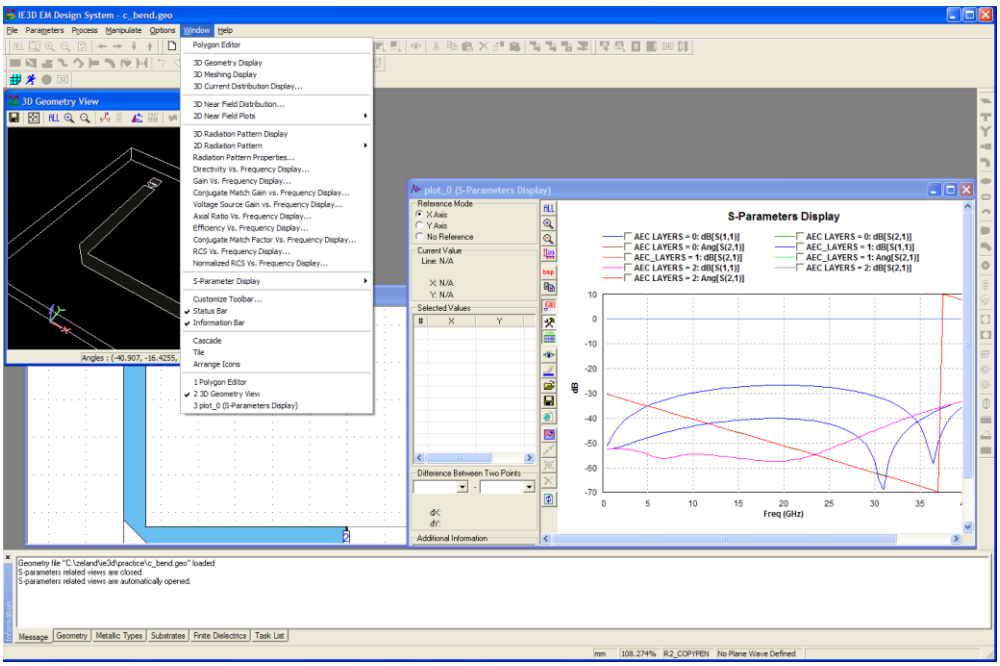


Figure 3.27 The defined plot is displayed on the MGRID main workspace.

Step 7 Select Continue button in the S-Parameters Processing dialog. The s-parameter plot is displayed in the MGRID workspace.

Old IE3D users may realize the big difference between IE3D 14 and older versions. The older MGRID is a SDI (Single-Document Interface). It can only show one window, and the user can open the 3D view. However, the 3D view is always on top of the drawing window and you will feel the 3D view window is in the way. We invoke MODUA to process the s-parameters, PATTERNVIEW to process radiation patterns, and another instance of MGRID to display the current and near field distributions. All the different windows from different application programs (MGRID, MODUA and PATTERNVIW) are overlapping each other and it causes confusion to users.

Starting from IE3D 14, we have integrated all the functionalities into MGRID. You can do all the different post-processing in the single interface MGRID or IE3D EM Design System. You have a big list of functionality in the Window menu of MGRID (see Figure 3.27). For this particular case, we have not saved the current distribution data (.cur file), the radiation pattern (.mpa and .pat files) and the near field (.nff). We can't process them right now. If you have enabled them, you will be able to access the data from the menu.

Again, the simulation results are saved either in the .geo or other related files. You can't modify the parameters of current structure that may affect the simulation results. Otherwise, the results will be removed even though you may still be able to access the results saved into the files. Certainly, you can change those parameters related to the display only. For example, you can change the colors of the polygons layers, the colors and display margin of the substrate layers, etc. For changing the substrate display parameters, you should not go to Parameters->Basic Parameters to change them because there are many parameters in the dialog will affect the results. You can select Parameters->Substrate Display Parameters to access those display

parameters not affecting the simulation results. You can also double click at the Substrates tab in the Information Bar to bring up the dialog.

For comparison, the major parameters for the 3-cases are listed in Table 3.3 and a summary is listed in Table 3.4. The simple example here demonstrates how easy you can obtain reliable accurate and reliable results from IE3D efficiently.

Table 3.3 The various simulation cases with different AEC Layer settings.

File Name	C_BEND.GEO	C_BEND3.GEO	C_BEND5.GEO
AEC Layer	0 (or No)	1	2
AEC Ratio	N/A	0.1	0.1
Multi-Layer Ratio	N/A	N/A	0.4
Cells	25	83	142
Unknowns	26	137	251
Zc at 10 GHz	52.429-j0.163	50.831-j0.193	50.320-j0.206
Total Time	1	2	4

Table 3.4 Summary on Simulation of the Simple Microstrip Bend.

Summary	Explanation
Single cell in the cross-sectional direction may still yield good results for simple and isolated structures	When the structure is not strongly coupled, edge condition may not affect the s-parameters very significantly. As you can see from Figure 3.27, the dB[S(2,1)] and Ang[S(2,1)] are still quite reasonable using just single cell in the cross-section. The most difference is in the dB[S(1,1)] while the dB[S(1,1)] is below -27 dB. It mainly comes from the difference in predicting the Zc value (about 4% as shown in Table 3.3). How to find the Zc of a TLN is discussed in Appendix AQ. For such a simple structure, there is no doubt we should use AEC. However, for some large structures, the number of unknown will become too big with AEC. We may have to disable AEC or use edge cells wisely.
AEC Layer = 1 normally can yield very accurate results. For most applications, we can use AEC Layer = 1 to obtain precise results. For normal applications, there is no need to use AEC Layer larger than 1.	From Figure 3.27, you may still see 10 dB difference between AEC Layer = 1 and AEC Layer = 2. However, the dB[S(1,1)] is below -40 dB and it is very small. For transmission line structure close to matching, S(1,1) is the most sensitive parameter. The sensitivity also comes from the accuracy in modeling the Zc of the TLN. As you can see from Table 3.3, the difference between the Zc values predicted by AEC Layer = 1 and AEC Layer = 2 is less than 1%. It is very small. However, the difference it brings to dB[S(1,1)] is 10 dB with the base at -40 dB.
Fine-tuning AEC Ratio value may improve the accuracy.	Using AEC will certainly improve the accuracy significantly. The AEC Width is also very critical. AEC Width should be about 5-20% of the cell width. The default value of AEC Ratio = 0.1, or the AEC Width is about 10% of the standard cell size. IE3D is using non-uniform meshing. It is impossible to control AEC Width to be exact 10% or it may need to adjust for some structures. For example, if there is a narrow gap between two traces, we may need to adjust AEC Ratio to make the edge cell width not significantly larger than the gap width in order to get high accuracy results.
Normally, more and smaller edge cells will make the simulator predict more accurate loss effect.	More edge cells will capture the edge effects better. One of the major edge effects is the non-uniform current distribution in the cross-section. More and smaller edge cells can capture such a change better.

Section 3.10 Simple Ways to Build a Chamfered Bend and Other Path Structures

We discussed how we build the bend in Sections 4 and 5. MGRID allows you to enter the bend as a 7-vertex polygon. We do not mind entering the vertices one by one for this 7-vertex polygon. However, if a structure contains many bends and other geometry, it will be tedious to construct it as one ore more polygons with many vertices. You can import complicated structures from other CAD tools using GDSII, DXF, GERBER and ACIS. Numerous powerful commands are implemented into MGRID to help you build and clean structures. We will show you some simple ways to build the bend. Let's analyze it first.

This is a bend with the centerline defined by three vertices: (0.0375, 0.75), (0.0375, 0.0375) and (0.75, 0.0375). The three vertices and the width determine the basic shape of the bend except the corner. We can use the Adv Edit->Build Path command to build the bend easily. Please note that the three vertices are not on the grid points. We cannot use mouse input to enter the 3 points. We can use keyboard input for them.

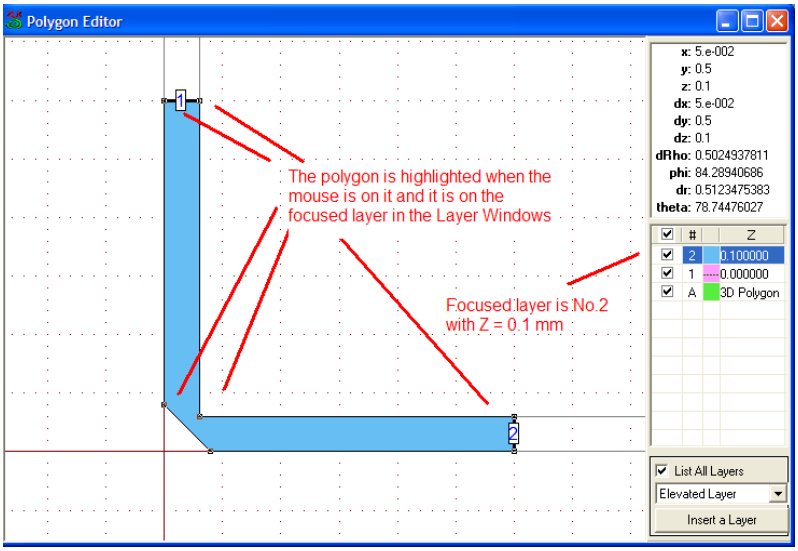


Figure 3.28 The object or polygon is highlighted when mouse is pointing at it.

Step 1 Run MGRID. Open the file "**.\ie3d\practice\c_bend.geo**". Click at the No.2 layer with Z = 0.1 mm in the Layer Window to focus the input on the layer. Move the mouse to point at the bend. The vertices of the bend are displayed with some markers indicating the bend polygon is highlighted.

We are going to delete the bend and rebuild it using another way.

Step 2 While the bend polygon is highlighted, click the right mouse button and you will see three useful commands in the pop-up menu:

Highlighted Object Properties
Delete Highlighted Object
Select Highlighted Object

Allowing highlighted objects is a new feature on MGRID. The object can be a polygon, a text object or a port. For our case, we just want to delete the polygon.

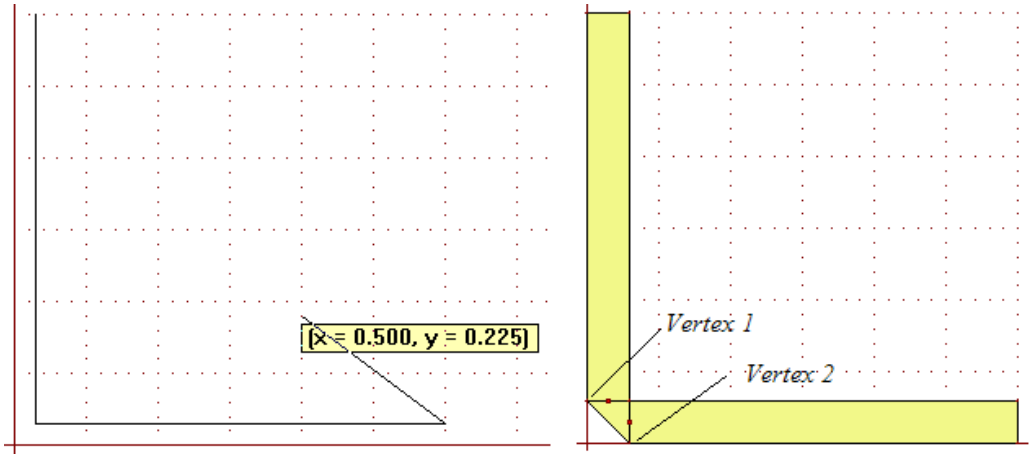
Step 3 Please select Delete Highlighted Object from the pop-up menu. MGRID will warn you that the s-parameters results will be lost. Select Yes to continue. The bend polygon will be removed. In case the s-parameter display graph was still opened, the graph will be automatically closed.

There are many ways to delete the polygon in IE3D 14 while the above way is the simplest. You can select Edit->Select All->Polygons to select the polygon. The selected polygon will turn black color when it is selected. The status window will display "Select Polygon Mode…". Then, you can select Edit->Delete command to delete the selected polygon.

Step 4 Select **Input->Key in Absolute Location** command. MGRID will prompt you to enter the X and Y-coordinates of the vertex. Enter X-Coordinate = 0.0375. Type the **Tab** key to shift the focus to the Y-Coordinate edit control. Enter the Y-Coordinate = 0.75. Then, hit **Enter** key (or select OK) to create the 1st vertex at (x, y) = (0.0375, 0.75).

Step 5 Type **Shift+A**. The **Shift+A** means that you press down the **Shift k**ey and hold it. Then, you type the **A** key. Finally, you release the hold **Shift** key. It is equivalent to selecting **Input->Key In Absolute Location** command. You will be prompted for the location of the next vertex. Enter (X, Y) =(0.0375, 0.0375) at the two fields and hit **Enter** key. The 2nd vertex is entered at (x, y) = (0.0375, 0.0375).

The **Shift+A** is the accelerator for the menu item **Input->Key In Absolute Location** command. It is listed in the right hand side of the menu item. You can access the menu item using the accelerator.



(a) The entered 3 vertices (b) The bend built using Build Path command

Figure 3.29 The chamfered bend created using the Build Path in Adv Edit menu.

Step 6 Type **Shift+R**. You will be prompted for the offset of the next vertex to the last vertex. The next vertex should be located at (x, y) = (0.75, 0.0375). The last vertex is at (x, y) = (0.0375, 0.0375). The offset should be (dx, dy) = (0.7125, 0).

The relative location between the 2nd and 3rd vertices is not straightforward. We use this command just because we want to demonstrate how we can use the Input->Key In Relative Location command. The **Shift+R** is the accelerator for it.

Step 7 Enter X-offset = 0.7125 and the Y-offset = 0.0. Hit **Enter** key. The 3rd vertex is entered at (x, y) = (0.75, 0.0375). We will see the 3 entered vertices forming a right angle in Figure 3.24a.

The above steps demonstrate how we can enter vertices using keyboard input (**Input->Key In Relative Location** and **Key In Absolute Location** commands).

Step 8 Select **Adv. Edit->Build Path** command. Enter the Path Width = 0.075 and select the Path Style as Smooth Corner (the default). Hit **Enter**. A chamfered bend is automatically created on MGRID as three polygons (see Figure 3.29b).

The bend is at the same location as the one created in Sections 4 and 5. However, the cut at the corner is smaller. This bend is automatically created. You may not have the control over some of the small detail. The small difference in the corner may not have significant effect to the performance of the structure. You may leave it as it is. However, if you want to make it exactly like the bend we constructed in Section 4, you need to move the vertex 1 in Figure 3.29a in the y-direction in by 0.025 mm, and the vertex 2 in Figure 3.29b in x-direction by 0.025 mm.

Step 9 While you are still in the default drawing mode, press down "Shift" and window the vertex 1 only in Figure 3.29b. What "window" means here is defined in Chapter 2. You move the mouse to the upper left corner of vertex 1. Press down the left mouse button. Drag it to the lower right corner of vertex 1. Then, you release the left mouse button.

A small rectangular marker is displayed at the vertex 1 location, indicating the vertex 1 is selected. Certainly, you can follow the standard way to select the vertex 1: Select **Edit->Select Vertices** command. Then, window the vertex 1 only in Figure 3.29b.

Step 10 Press down "Shift". Press down the left mouse button and move. MGRID automatically gets into the **Edit->Move Objects** mode. This is a short cut to select **Edit->Move Objects** command. It only works when MGRID is in selection mode.

Move the mouse upward somewhere and click the left button. The "Move Object Offset to Original" dialog comes up and display the offset values (Figure 3.30).

MGRID is a very precise layout editor. The coordinates of each vertex of the polygons are described precisely using double precision floating point values. When we use the move and copy commands, we need to preserve the precision. Therefore, we always prompt you to confirm the coordinates on the move and copy commands. You can always enter precise values which mouse entry cannot capture.

There are in fact 3 snapping modes for the move and copy commands: (1) Snap to Vertex, (2) Snap to Edge, (3) Snap to Vertex. The default snapping mode is "Snap to Vertex". To change the snapping mode, you can type F4 while you are moving the vertex. If you use the **Edit->Move Objects** command, you don't need to keep the left mouse button pressed. You can select **Edit->Change Snapping** command (corresponds to F4) to change the snapping mode. In the **Edit->Move Objects** mode, you can also change the **Moving Reference Vertex** by clicking the right mouse button or use the **Input->Set Moving Reference Vertex**, or **Shift Moving Reference by Vertex** or Shift **Moving Reference by Polygon** command. The Moving Reference Vertex is the vertex following the cursor in the move and copy command.

For our current moving, there isn't any other vertex near the vertex 1. It will not snap to any vertex. We do not need to worry about it because we know the offset values as: dx = 0 and dy = 0.025 mm.

Step 11 Enter the **X-offset** = 0 and the **Y-offset** = 0.025. Hit **Enter** to continue. The vertex 1 in Figure 3.29b is moved in the y-direction by 0.025 mm. The vertex 1 is de-selected after the moving and the result is shown in Figure 3.31a. Vertex 1 is moved to the right location right the vertex 2 still needs to be moved.

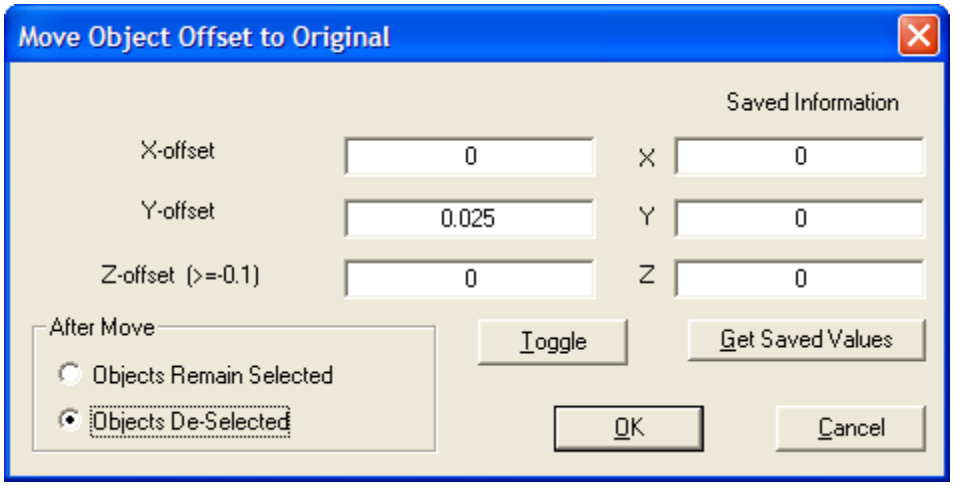
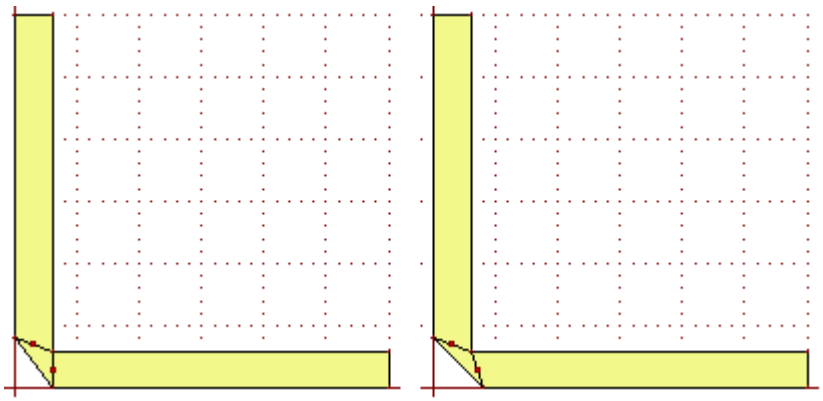


Figure 3.30 The Move Object Offset to Original dialog.



(a) The shape after moving vertex 1 (b) The shape after moving vertex 2.

Figure 3.31 The shape of the bend after the vertices are moved.

Step 12 Press down "Shift" and window the vertex 2 only in Figure 3.29b to select it. Press down "Shift" and press down the left mouse button and move the mouse to move the vertex 2. Click somewhere at the right. The "Move Object Offset to Original" dialog will come up. Change the **X-offset** = 0.025 and the **Y-offset** = 0. Hit Enter.

The vertex 2 is moved with the offset as (dx, dy) = (0.025, 0). MGRID resumes to the geometry Drawing mode automatically because the changed default for **After Move** is **Objects De-Selected**.

The bend after Step 12 will be exactly as the bend we want (see Figure 3.31b). We built the bend as a single polygon in Section 4 while we built it as 3 connected polygons. The results of both

structures will be identical if the meshing is the same. In case different meshing is used, the difference between the results should be very small.

We just demonstrate a new way to create the bend. We would like to define some comment in the file to denote it.

Step 13 Click some point away from the polygon to enter a vertex. Type "t". It is equivalent to selecting Input->Define Text Object command. It will bring out the "Edit No.0 Text Object" dialog shown in Figure 3.32. Enter the text as "Bend built using alternative commands". Select OK. MGRID will create the text in the layout window (see Figure 3.33). As you can see, there is underline for the string, and the color of the underline is the same as the polygon, indicating both the polygon and the text are on the same layer. A text string also has layer and location.

Allowing text objects is the new feature of IE3D 14. Text objects are comments only. They will not affect the simulation results.

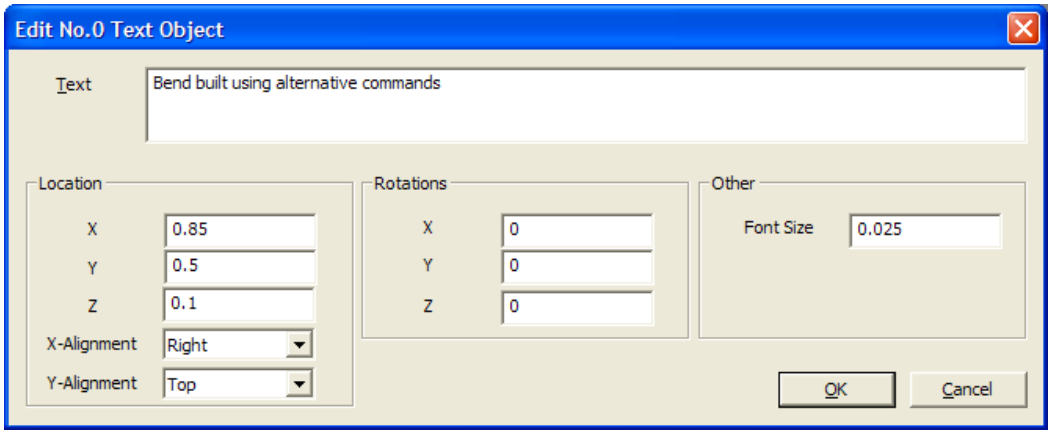


Figure 3.32 The Edit Text Object dialog.

Step 14 Define the port 1 and port 2 on the structure. We will not repeat the steps because they are discussed earlier. Save the geometry file as: .\ie3d\practice\c_bend2.geo. If you simulate it, you should get the same or close results as the one built in c_bend.geo.

There is still another simpler way to build the chamfered bend. It is using the **Entity->Chamfered Bend** command documented in the Appendix I. You can try it out if you are interested in it.

When we build the structure as 3 polygons. We need the 3 polygons to be connected "electrically". We have built the 3 polygons using automatic way. The "electrical connections" among the 3 polygons are guaranteed. The guaranteed electrical connections are indicated by the red dots at the centers of the common edges (see Figures 3.26). In case we need to build a structure with many polygons with some of them connected and some of them not connected, how could we guarantee the "electrical connection" between them? Polygon connection or "electrical connection" is a very important concept in the IE3D. We will discuss about it in the next chapter.

Section 3.11 Parameterization of Structures

Before we discuss polygon connections in the next chapter, we would like to touch base of parameterization of structures.

IE3D is not only good for analysis. It is also good for tuning and optimization. To tune or optimize a structure, we will need to parameterize it first. For this simple bend structure, we know we can optimize the cut of the corner for best performance. The question is how we can optimize it.

Tuning and optimization start from parameterization of the structure. The 1st step is to parameterize the corner cut of the bend. We are able to parameterize the cut on MGRID by controlling the location of the vertices in the same way as we discussed to change the shape of the cut by moving the vertices. We select the vertex 1 in Figure 3.29b. Then, we select Optim->Variable for Selected Objects. We define its Tuning Angle = 90 degrees. Then, use the mouse to move around to define the Low Bound and the High Bound. We are able to define the location of vertex 1 as an optimization variable. Then, we select vertex 2 in Figure 3.29b. We select Optim->Add Selected Objects to Variable. Select "Vertices Mapped to No.1 Variable", define Tuning Angle = 0, and Tuning Rate = 1. We are able to associate the change of vertex 2 to the change of vertex 1 so that the corner cut will be kept symmetrical while the cut size is changing.

It is possible that you may not be able to follow what the discussion. It is ok because we will have special chapters devoted to tuning and optimization. The parameterized chamfered bend is created in: .\ie3d\samples\c_bend2_tuning.geo. You can open it. Then, please select Optim->Geometry Tuning. Select the Variables Slides tab on the right. Drag the slider representing the only variable in the geometry, you will be able to see how the variable is controlling the cut size of the corner and the structure is kept symmetry while it is changing.

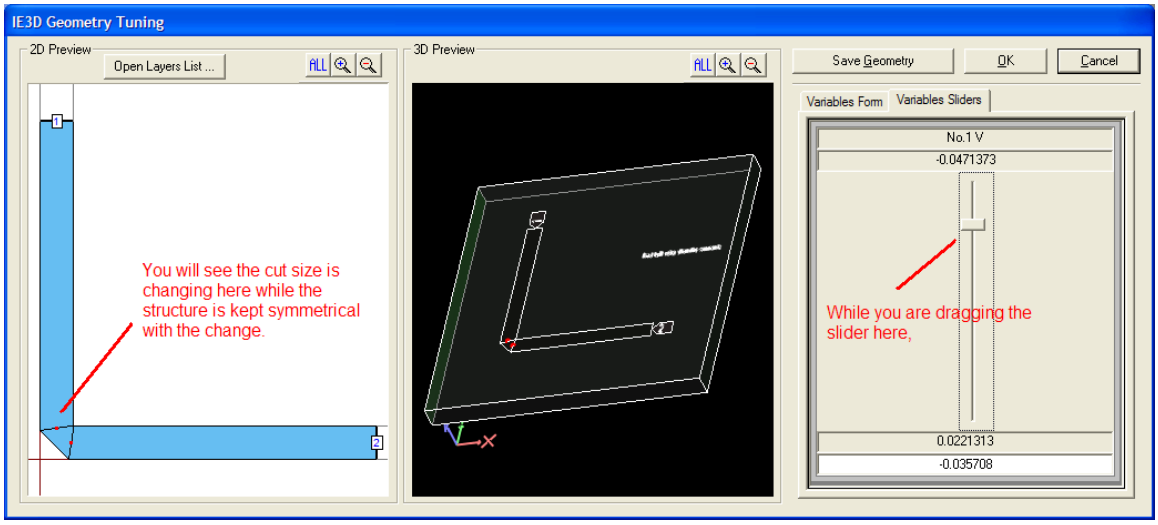


Figure 3.33 Geometry tuning for parameterized structure.

As MGRID is able to parameterize structures by controlling the vertices of the polygons, IE3DLibrary is much easer and more powerful in parameterize structures. On IE3DLibrary, a structure is created as a group of parameterized objects. An object is not a polygon, but a higher level abstraction such as a T-junction, a straight-strip or a circular patch. An object is described as a set of dimensional parameters such as length and width etc. Each dimensional parameter can be an equation and the equation can be a function of tuning variables. We can easily parameterize the chamfered bend on IE3DLibrary. Saved in .\ie3d\practice\c_bend2_tuning2.ie3 file is an example of the IE3DLibrary. For this structure, the cut size is parameterized as a tuning variable while the line length and the line width are parameterized as global variables. You can easily tune the cut size and even the line length and the line width easily if you hook up the global variables with some more tuning variables. If you open the file from IE3DLibrary. Select Edit->Geometry Tuning. You will see something similar to what is shown in Figure 3.33. Detail on using IE3DLibrary will be discussed in later chapters.

After a structure is parameterized, we can do a preparation for real-time full-wave EM tuning and optimization on the FastEM Design Kit, which is an integrated part of IE3D. We can also perform IE3D full-wave EM optimization on a parameterized structure. We will discuss this advanced topic in later chapters.

Chapter 4 Geometry Modeling of Planar and 3D Structures

You have learned how to build and simulate a microstrip bend using MGRID/IE3D in Chapter 3. Most structures cannot be described as one single polygon. A typical structure is described by multiple or many polygons. Some polygons might be connected together. It is very critical to determine whether two polygons are connected together as we want in building a complicated structure with many polygons. In this chapter, we will concentrate on practicing more on using many of the most frequently used IE3D commands in geometry modeling.

Section 4.1 Polygon Connection

In the IE3D, two polygons are considered to be connected only when they have a common edge with two matching vertices. Otherwise, they will be considered as disconnected even if they look as if they are connected. The difference between connected and disconnected polygons is shown in Figure 4.1. For the case (a), two polygons are next to each other and no visible gap between them. There might be two possibilities: (b) There is a matching vertex on the left polygon and both polygons have a matching edge. In this case, the two polygons are connected; (c) There is no matching vertex on the left polygon and there is no matching edge between the two polygons. They are not electrically connected.

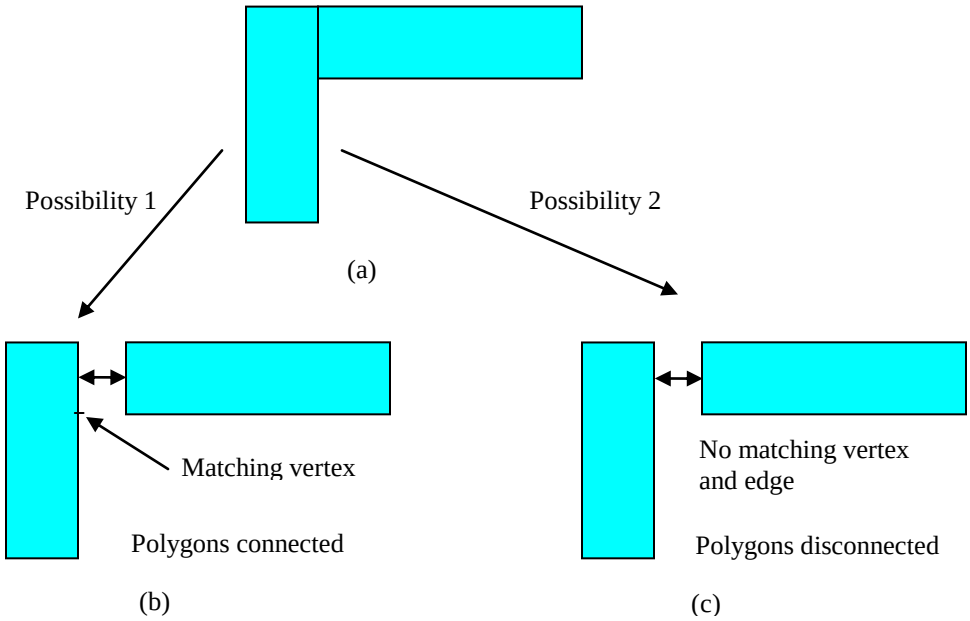


Figure 4.1 Polygon connection.

On MGRID, a common edge is denoted by a red spot at the center of the edge. In order to let users better understand polygon connection, we will show the difference between connected polygons and disconnected polygons in the following:

Step 1 Run MGRID. Select **File->Open** command. Select file **.\ie3d\samples\connect.geo** and select **OK** to continue. The geometry file **.\ie3d\samples\connect.geo** is opened (see Figure 4.3). As you can see, there are some red dots on the two common edges of the three polygons. These red dots are used to denote the polygons are connected on the common edges. Some vertices in polygon 2 are off-grid. As it is mentioned before, the grids on MGRID are used for geometry entry purpose only. An off-grid structure will still be simulated exactly as its shape on the IE3D. No any approximation is taken in fit the geometry into the grids.

The structure in connect.geo is a suspended stripline structure. You can select **Parameters->Basic Parameters** and check the **Substrate** sections to see how to define substrate for a suspended stripline structure.

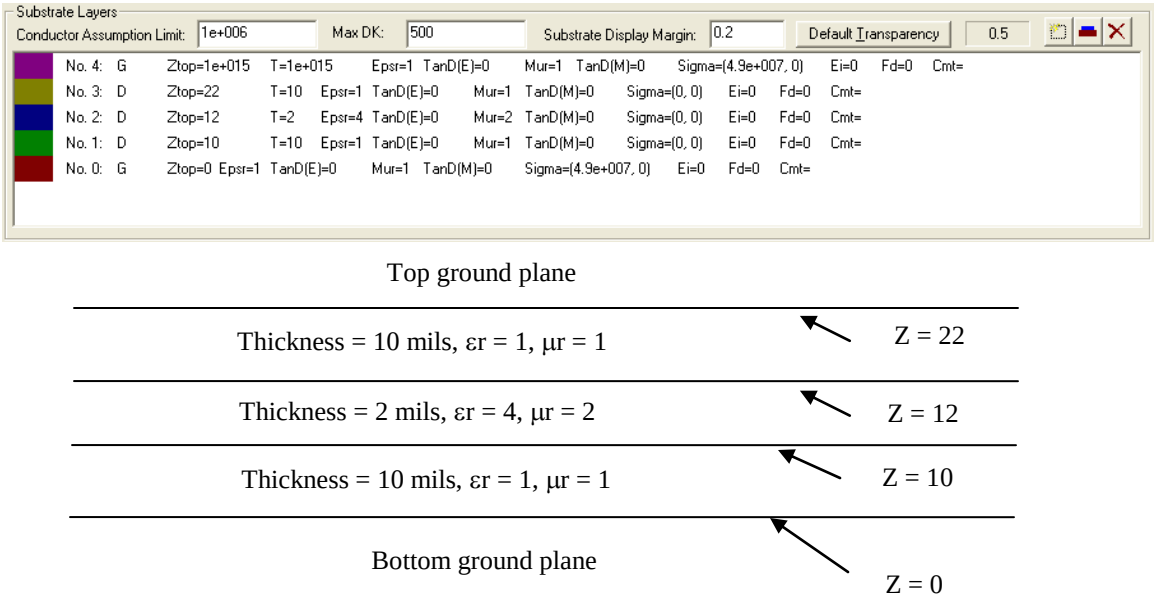


Figure 4.2 The substrate configuration on Basic Parameters dialog and the illustration.

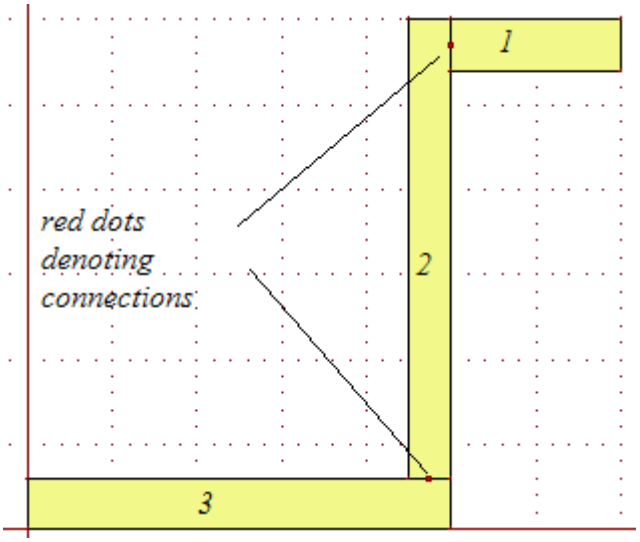


Figure 4.3 Connected polygons in **.\ie3d\samples\connect.geo**

Step 2 Select **Edit->Select Polygon** command. The Status Window shows "Select Polygon Mode…". Click at the No.2 Z = 10 layer on the Layers Window. Click at the polygon 1 to select it. The polygon 1 in Figure 4.3 is changed to black color indicating it is being selected. This is a standard way to select one polygon. Certainly, we can use the Select Highlighted Polygon command and it is the fastest way.

Step 3 Select **Adv Edit->Connection->Check Connection** command. The other two polygons (polygons 2 and 3 in Figure 4.3) become cyan color as shown in Figure 4.4. It means that the

two other polygons are electrically connected to polygon 1. Therefore, we confirmed there are perfect electrical connections between the 3 polygons.

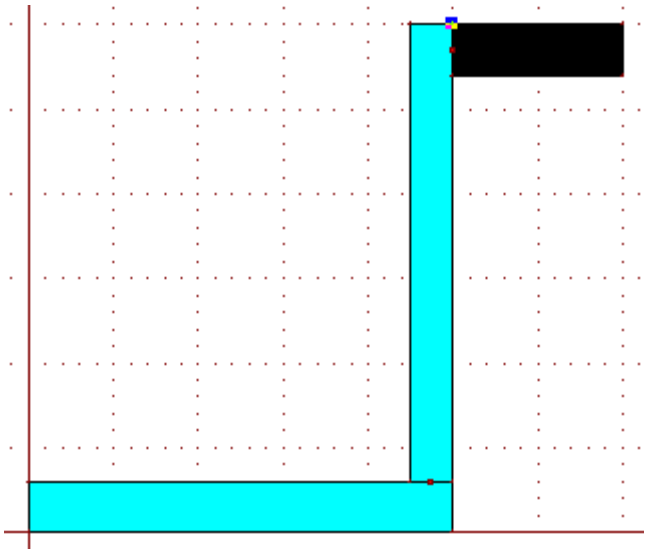


Figure 4.4 The black polygon is the selected one and the cyan polygons are electrically connected to the selected polygon.

Section 4.2 Polygon Overlapping Detection and Handling

Overlapped polygons may introduce ambiguity in electrical simulation. They will not be considered as electrically connected if they are not specially treated. Please keep in mind that polygons are considered electrically connected on IE3D only when they have common edges. Many mechanical oriented layout editors do not have the same electrical connection concept as IE3D. Many technicians draw a geometry as it looks like what they want. They may not care whether it is meaningful and whether it contains enough geometry information. For example, a technician may draw a polygon as multiple isolated line segments. The isolated line segments may look like they form a polygon. However, they are still considered as some line segments on IE3D and most layout editors.

We have implemented strong capability on the IE3D to handle overlapped polygons. The IE3D engine can automatically clean overlapped polygons in case they are any. However, such cleaning is automatically done and it may not be what you want. It would be nice you can clean them before you start a simulation. In this section, we will demonstrate how we handle overlapped polygons on MGRID.

Step 1 Select **File->Open** command and select file **.\ie3d\samples\overlap.geo**. The structure is opened on MGRID with a warning message "**Overlapped Polygons Detected**" (see Figure 4.5). Polygon 2 and polygon 3 shown in Figure 4.6 are overlapped.

The dialog "Overlapped Polygons Detected" may appear quite frequently when you are doing editing. You have 5 options in such a case. The different options are discussed in Table 4.1.

Step 2 Select Highlight Them button in Figure 4.5. MGRID will show the opened structure. However, MGRID will show the polygons 2 and 3 in Figure 4.6 with pink color instead of the color for polygon 1 even the three polygons are on the same layer. The pink color indicates that the polygons 2 and 3 are overlapping with each other.

Figure 4.5 The Overlapped Polygons Detected dialog.

Table 4.1 The options for the Overlapped Polygons Detected dialog.

Option	Description
Clean Them	MGRID will try to cut the overlapped polygons to make sure they will not be overlapping each other.
Select Them	MGRID will get into Edit->Select Polygon mode and select the overlapped polygons.
Highlight Them	MGRID will not change anything. However, it will highlight the overlapped polygons.
No Action	MGRID will do nothing and it will not highlight the overlapped polygons.
Cancel	If MGRID detects the overlapped polygons after an editing command and you choose Cancel, MGRID will undo the change of the editing command. If it is opening a file, it will be like No Action.

Figure 4.5 The "Overlapped Polygons Detected" dialog.

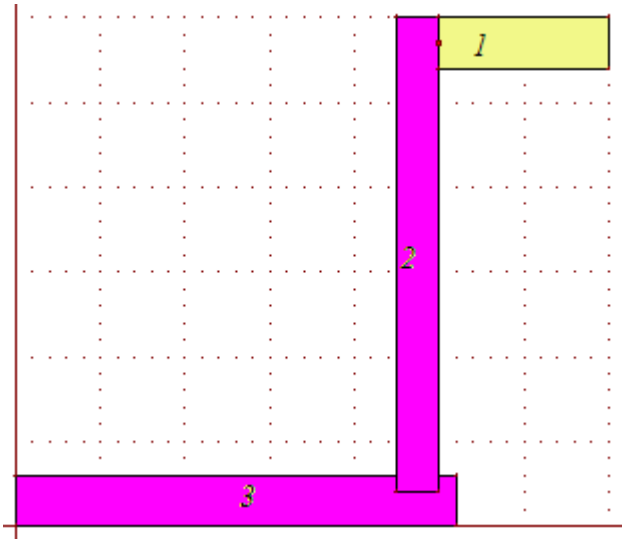


Figure 4.6 overlapped polygons in **.\ie3d\samples\overlap.geo**.

Step 3 Select **Adv Edit->Overlapped, Multi-Looped and Twisted Polygons->Cut Overlapped Polygons** command. MGRID will prompt you the Cut Overlapped Polygons dialog (see Figure 4.7). There are different options there. The list box for the Cutting Option is disabled. It is only highlighted if you select the command after you select some polygons. There are options for "After Cutting Process" for you. You can choose to merge the polygons after the cutting process. You can also choose to remove redundant vertices on curvature. For this option, MGRID may change the shape of curved structure slightly with some vertices removed. The "Remove

Redundant Vertices on Curvatures" is a command in the Adv. Edit menu. We will discuss about it later in this manual.

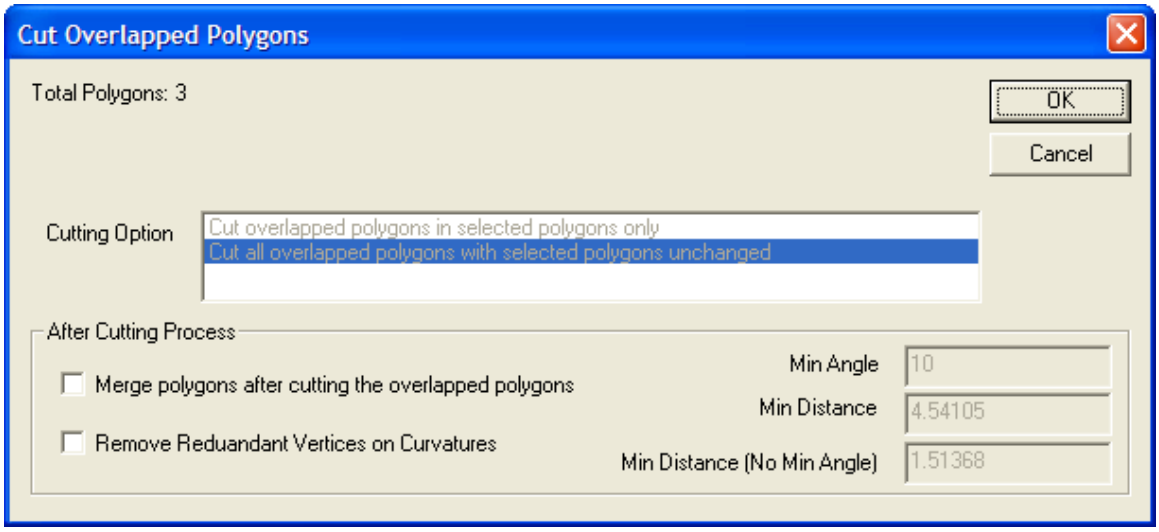
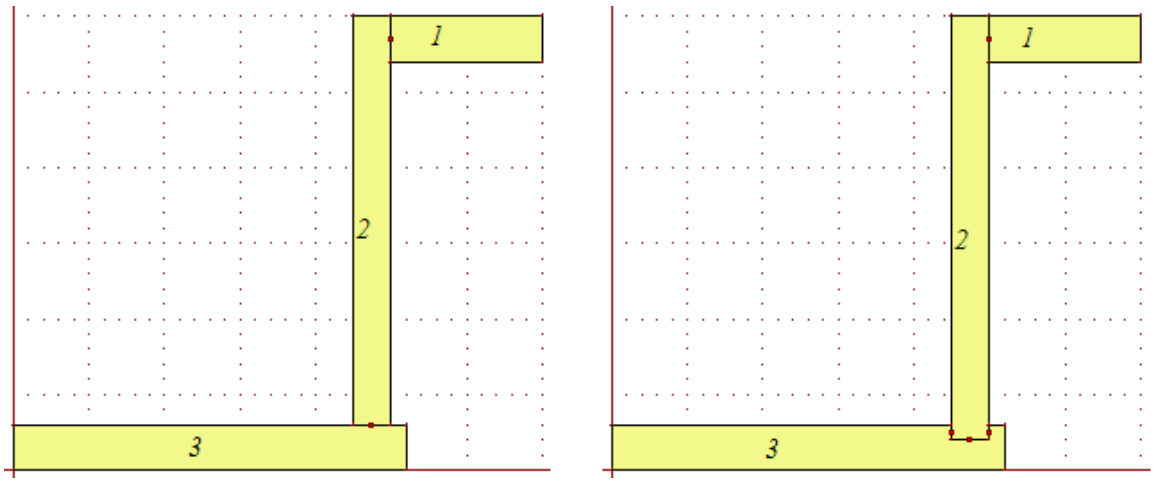


Figure 4.7 The Cut Overlapped Polygons dialog.



(a) The connected polygons after Step 4

(b) The connected polygons after cutting if you Select the polygon 2 before the command.

Figure 4.8 The results of cutting overlapped polygons with different options.

Step 4 Select OK and MGRID will cut the overlapped polygons. You will see a red dot between the polygons 2 and 3 after the cutting, meaning the two polygons are connected after the connecting (see Figure 4.8a). If you select **Adv Edit-> Overlapped, Multi-Looped and Twisted Polygons Check Polygon Overlapping** command, MGRID will inform you that no overlapped polygons are detected.

In case you select the polygon 2 before you select **Cut Overlapped Polygons** command. The Cutting Option list box will be highlighted. If you select "Cut all overlapped polygons with selected polygons unchanged" in the Cutting Option, you will get the result in Figure 4.8b. General speaking, the connection in Figure 4.8b is not as good as the one in Figure 4.8a. We just want to show you some feature you may possibly need in building geometry on the MGRID. We may need this kind of flexibility occasionally.

We have demonstrated how we can detect and clean overlapped polygons in the above examples. However, please keep in mind the commands we used and the automatic detection are only good for 2D polygons only. They do not apply to 3D overlapped polygons. The reason for it is that 3D overlapped polygons are much more difficult to detect and cleaned. The process may take a long time when the number of polygons is big. In IE3D 14, we have implemented a command to clean 3D overlapped polygons.

Step 5 Open file .\ie3d\samples\overlap_3d.geo. You can see there are three 2D (or horizontal) polygons. There is one 3D polygon spanning between z = 0 and z = 22 connecting between the bottom and top grounds. There are some red dots on two edges lying on the bottom and top ground planes. The red dots denote the polygons are connecting to both the bottom and top ground planes. On IE3D, when an edge of a 3D polygon is lying on an infinite ground plane, it means that the polygon is connected to the infinite ground.

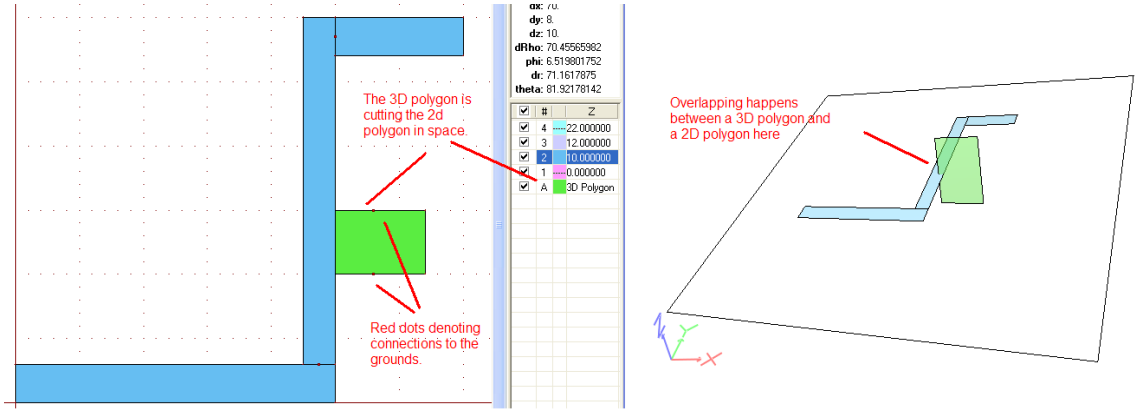


Figure 4.9 The top view and the 3D view of the overlap_3d.geo file.

When you open the file, MGRID does not complain about the overlapped polygons. This is normally because the Check Overlapped Polygons command only considers the case with 2D polygons overlapping with 2D polygons. For this case, the 2D polygon and the 3D polygon are overlapping and this overlapping is transparent to the command.

If you have 2 metallic strips crossing each other in your design, they are electrically connected. For the above IE3D model, it is not a correct model because it may not consider the polygons are connected. If you simulate it as it is, it may not be able to yield accurate results. How can we fix the structure?

Step 6 Select Adv Edit->Connection->Adjust Geometry for Connection. MGRID will prompt you a dialog. Please select OK and MGRID will cut the 2 polygons and connect them for you. The results are saved in: .\ie3d\samples\connect_3d.geo. For this structure, the 3D polygon is electrically connected to the 2D polygon and it is also connected to the bottom and top ground planes.

The Adjust Geometry for Connection is an old command on MGRID. However, the feature to divide overlapped 3D polygons is a new feature implemented in IE3D 14. It is not automatically called in the geometry modeling at this time considering the fact this is a very sophisticated command. It may take a long time and it may need more time to make sure it is robust enough so that we can integrate it into the geometry modeling routine. To make sure 3D overlapped

polygons are electrically connected, you do need to select this command from time to time. Also, please check the results. Again, it is a new command. We can't be sure the results are 100% reliable. Table 4.2 summarizes the different situations in overlapping detection and cleaning.

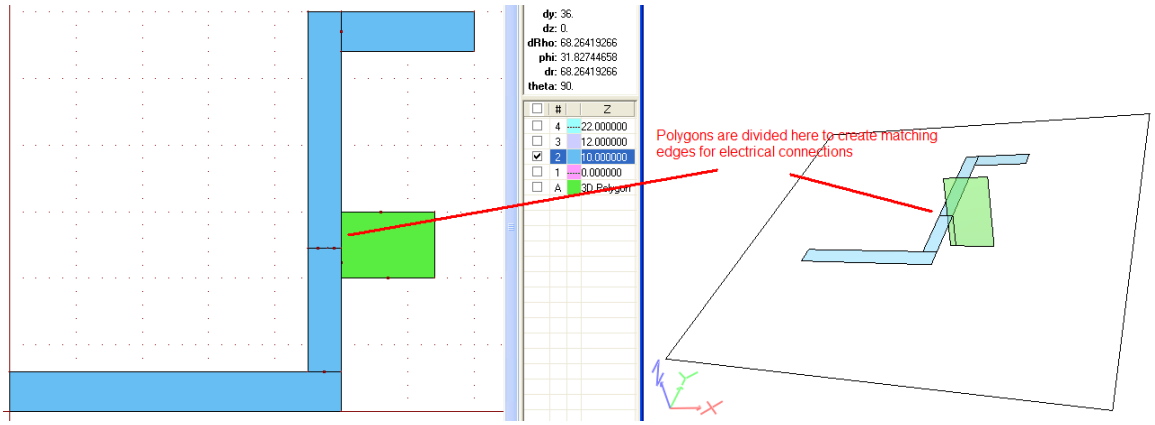


Figure 4.10 Polygons are divided to create matching edges for connections.

Table 4.2 The different situations of polygons overlapping

Case	Detection	Cleaning
2D polygons overlapping with 2D polygons	It will be automatically detected by MGRID.	You can select Clean button to let MGRID divide the polygons for connection for you automatically. You can also select Cut Overlapped Polygons command to clean them.
3D Polygons overlapping with 3D or 2D Polygons	It will not be detected by MGRID. You can only visually check them.	The Cut Overlapped Polygons command will not be able to clean them. You need to select Adjust Geometry for Connections command on MGRID 14 or later versions to clean them.

Section 4.3 Snap to Vertex and Merge Polygons

Step 1 We are going to show some other case with disconnected polygons and show you how to handle them. Select **File->Open** and select file **.\ie3d\samples\gap.geo**. The geometry is shown in Figure 4.11a. Polygon 1 is formed by vertices 10, 11, 12 and 13. Polygon 2 is formed by vertices 5, 6, 7, 8 and 9. Polygon 3 is formed by vertices 1, 2, 3 and 4. There are two pairs of matching vertices: vertices 8 and 10, and vertices 9 and 11 in the geometry.

Vertices 1 and 5 are supposed to be matching vertices, but now there is a gap of 1 mil between them. There are multiple ways to connect the polygons. One way is to build another polygon is to fill the gap. Another way is to move the vertices 5 and 6 to close the gap. We will show the first way first. The possible difficulty is that vertex 1 is off-grid. The question is how we can grab it to make sure the electrical connection when we build the additional polygon.

Step 2 Click at the No.2 layer Z = 10 to focus the input at the layer where the polygons are on. **Select Input->Set to Closest Vertex**. In this mode, MGRID will snap any vertex entered by mouse to the closest vertex on the layer.

In fact, there are 3 modes for the Set to Closest Vertex. They are documented in Table 4.3.

Table 4.3 The different modes for Input->Set to Closest Vertex interchangeable by typing F5 key.

mode	Description	Status Window
Set Connecting to 2D Closest Vertex	It is the default mode in Input->Set to Closest Vertex. MGRID will find the closest vertex at the layer and snap to it.	Set Connecting to 2D Closest Vertex F5 to rotate
Set Connecting to 3D Closest Vertex	It is the 2nd mode. MGRID will snap to any vertex (not necessary on the current layer) closest to the cursor. MGRID shifts to this mode if you hit F5 when you are in the "Set Connecting to 2D Closest Vertex" mode.	Set Connecting to 3D Closest Vertex F5 to rotate
Set to the (X, Y) of the Closest Vertex	It is the 3rd mode. MGRID will find the closest vertex (not necessary on the current layer) and assign the (X, Y) of the vertex to the entered vertex while the Z-coordinate of the entered vertex is of the Z-coordinate of the focused layer.	Set to the (X,Y) of the Closest Vertex F5 to rotate

Step 3 Click at the vertex 6 in Figure 4.11a to snap a vertex at it. Click at vertex 5 in Figure 4.11a to snap the 2nd vertex at it. Click at vertex 1 in Figure 4.11a to snap the 3rd vertex at it.

Step 4 Type "Shift+F". It is equivalent to selecting Input->Form Rectangle command. MGRID will fill the gap with a rectangle (see Figure 4.11b). You will see red dots on the both edges next to other polygons. It means that the entered polygon is connected to the other polygons.

Step 5 Press down "Shift" button and window the polygon 2 and polygon 4 in Figure 4.11b to select them. Select **Adv Edit->Merge Selected Polygons** command. The Merge Polygons command comes up with the option "Remove Inserted Vertices After Merging" checked. The "Remove Inserted Vertices After Merging" allows MGRID to remove some unnecessary vertices after the merging process. What is an unnecessary vertex? For example, if an edge of a polygon has 3 vertices a, b and c. The vertices a, b and c are on a straight line while edge a and b, and edge b and c are not common edges with other polygons. Then, the vertex b is an unnecessary vertex. An unnecessary vertex may cause IE3D to mesh the structure into more polygons. We normally should remove them. A command for such a purpose is the Adv Edit->Remove Inserted Vertices.

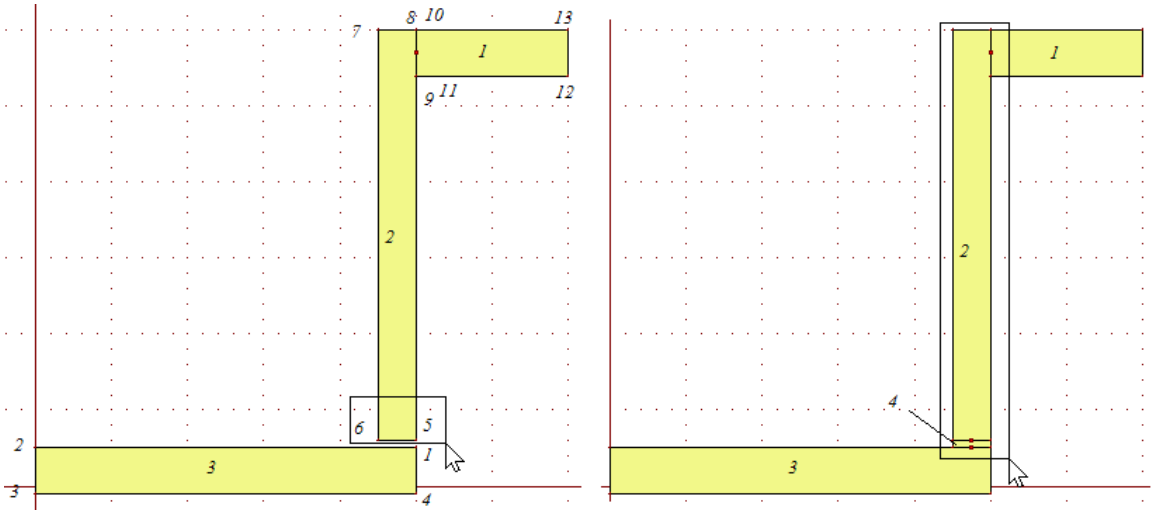


Figure 4.11 The pictures before and after the gap is filled.

Step 6 Select OK to continue. MGRID will merge the polygons 2 and 4 in Figure 4.11b. The result will be like the Figure 4.8a. You can save the revised file in another name. You should not save the change in gap.geo because we are going to demonstrate you an alternative way to fill the gap.

Section 4.4 Select and Move Vertices, and Change Moving Reference Vertex.

Step 1 Open **.\ie3d\samples\gap.geo** again.

Step 2 Press down "Shift" button and window the vertices 5 and 6 in Figure 4.9a to select them. We should avoid selecting the vertex 1 in Figure 4.11a, which is close to vertex 5. The suggested windowing size is shown in Figure 4.11a.

If the display size is too small, you can zoom it using the View->Zoom or Zoom In commands. If you accidentally window the vertex 1 as well as the vertices 5 and 6, you can window the vertex 1 again to de-select it so that only vertices 5 and 6 are selected.

We are going to connect the polygon 2 to polygon 3 by moving the vertices 5 and 6 with appropriate offset. The question is how much we should move. Certainly, it should be the distance between vertex 1 and vertex 5. However, it will take 3 steps in order to get the distance between vertices 1 and 5 (see Appendix AR on measuring distances). The question is whether we can have a fast way to make sure the connection without calculating the distance? The answer is positive.

Step 3 Please pay attention to the 2 selected vertices 5 and 6. They are displayed with markers. However, the marker on vertex 5 is kind of a solid one while the marker on vertex 6 is not a solid one. Basically, vertex 5 is the so-called Moving Reference. It will be used as the reference when we move or copy selected objects. We can shift the moving reference vertex to another one by selecting the **Input->Shift Moving Reference by Vertex** command and the **Input->Shift Moving Reference by Polygon** command any time after we select polygons or vertices and any time after we select **Edit->Move Objects** or **Edit->Copy** command..

There is another faster way to change the moving reference vertex. You point the mouse at one of the selected vertex and click the right mouse button to bring up a pop-up menu. You select the Set Moving Reference Vertex to set the vertex you pointed at before clicking the right mouse button.

Please type Shift+Tab (for **Shift Moving Reference by Vertex**) and Tab multiple times. You will see the Moving Reference Vertex shift between the two selected vertices 5 and 6. For our case with 2 vertices selected, Shift+Tab and Tab commands are the same. Please make sure the Moving Reference is at vertex 5 in Figure 4.12.

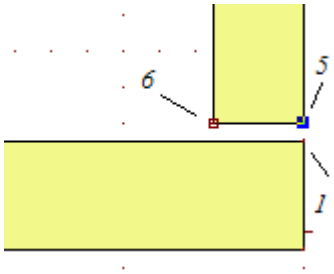


Figure 4.12 The selected vertices 5 and 6 with moving reference as vertex 5.

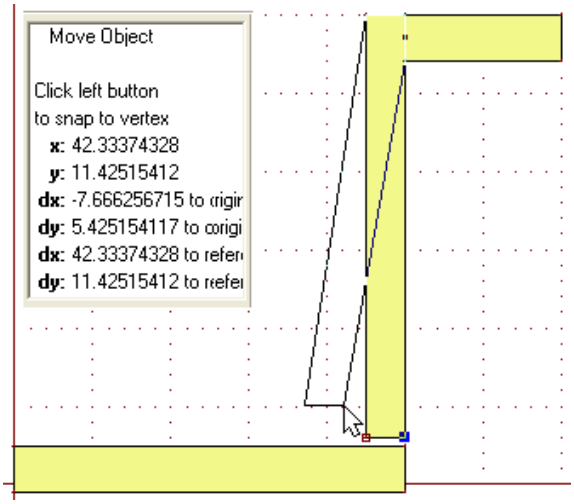


Figure 4.13 The picture when you drag the selected vertices.

Step 4 Press down the "Shift" button, press down the left mouse button and drag. MGRID is automatically set to **Edit->Move Objects** mode. It is equivalent to selecting the **Edit->Move Objects** command.

The shape of polygon 2 is changing with the moving mouse cursor (see Figure 4.13). After you see the change, you can release the left mouse button because it is in the Move Objects mode already. You can still move the mouse and you will see how the polygons are changed with your moving mouse. When you move the objects, you should pay attention to two things: (1) The Moving Reference Vertex; (2) The status window.

For our example, it is shown on the status window "Move Object, Click left button to snap to vertex, …" (see Figure 4.13). In fact, there are 3 snapping modes: (1) Snap to Vertex: The Moving Reference Vertex will snap to the closest vertex; (2) Snap to Edge: The Moving Reference Vertex will snap to the closest edge; (3) Snap to Grid: The Moving Reference Vertex will snap to the closest grid.

If it is not in the **Snap to Vertex** mode, while you keep dragging on the mouse, please type F4 to shift the snapping mode. It is equivalent to select **Edit->Change Snapping** command.

Step 5 Move the mouse cursor (or vertex 5) to about the location of vertex 1. Click the left mouse button. MGRID will snap the moving vertex 5 to vertex 1. It will calculate the offset values between the original vertex 5 and the moved vertex 5 (or the vertex 1) in Figure 4.11. It will prompt you to change the default values.

The offset values should be dx = 0 mil and dy = -1 mil. You should not change them normally because they are automatically calculated by MGRID for guaranteed connection.

Step 6 Check the **Objects De-Selected** in the **After Move** options. Select **OK** to continue.

The vertices 5 and 6 are re-located and vertex 5 is moved to the same location as vertex 1. It also automatically creates a matching vertex to vertex 6 on the edge between vertices 1 and 2 (see Figure 4.11). The matching vertex guarantees the connection between polygons 2 and 3.

We have shown two ways of connecting the disconnected polygons. In fact, there are many editing capabilities in the MGRID to achieve it. We will not show all of them here. When you learn more, you will understand the power of the MGRID layout editor. You can discard the change made here and we are going to learn other things.

Section 4.5 Editing of Multi-Layer and 3D Structures

We have discussed how to create and handle polygons on one single layer. Creating polygons on different layers is similar. We just need to select the "Insert a Layer" button on the Layer Window or on **Edit->Layers->Insert a Layer** command to define the z-coordinate of the layer and draw the polygons on the layer. The challenge is to build the connecting polygons between different layers. We will demonstrate how we can achieve such a goal easily on the MGRID.

Step 1 Select **File->Open** command, and select the file: ".\ie3d\samples\bridge.geo" to open it. The shape of the structure is exactly the same as the one saved in: ".\ie3d\samples\connect.geo". The only difference is that there is a section of strip in different color (see Figure 4.14). You can see from the Layer Window that the color indicates that the polygon is on the Z = 12 mil level. It is a bridge structure with some vertical polygons connecting a polygon on Z = 12 mils to those on the Z = 10 mils level. However, we cannot see the vertical polygons on the top view window.

Step 2 Select **Windows->3D Geometry Display** command. The 3D view of the structure is shown (see Figure 4.14). You can do much handling on the 3D view or use the mouse button. Please read the Appendix AS.

Please hit the arrow keys: ←, ↑, → , ↓ and the HOME and END keys to rotate the 3D view in 3-different directions. Please hit "+" and "-" to zoom in and zoom out the view. You can also use the mouse wheel to zoom in and zoom out. Hold the SHIFT key and dragging the left mouse button can pan the 3D view.

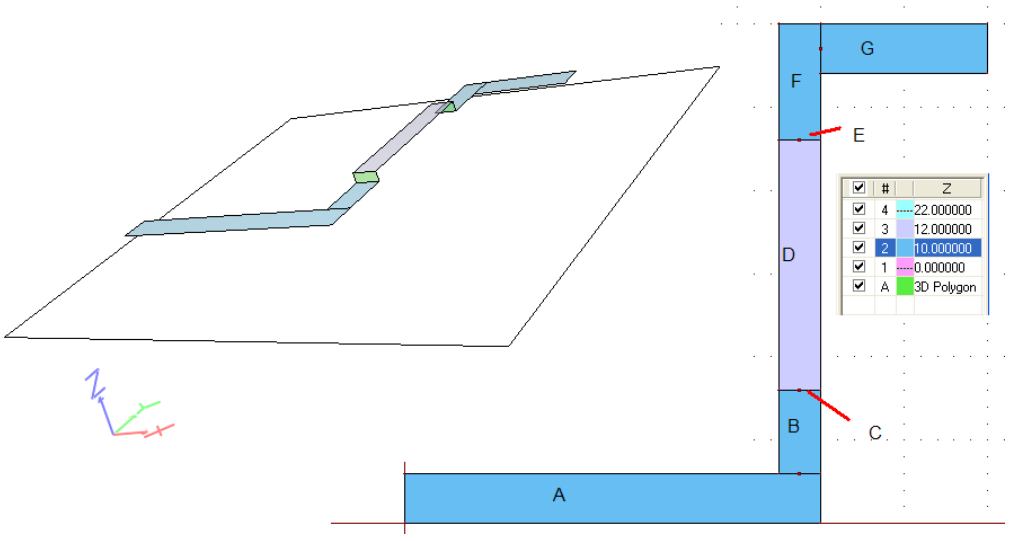


Figure 4.14 The 3D View, Top View and Layer Window.

Section 4.6 Delete 3D Polygons.

We are going to re-construct the bridge.geo. We will demonstrate how to delete polygons first. The bridge.geo consists of 3 portions: (1) The polygons on z = 10 layer; (2) The polygon on Z = 12 layer; (3)

The vertical polygons connecting the z = 10 and z = 12 layers. We can follow the ways we discussed in Chapter 3 to build the polygons on z = 10 and z = 12. The question is how to build the 3D polygons connecting the layers. We will select and delete the 3D polygons first and re-construct them.

Step 1 While MGRID is still showing the bridge.geo structure, select Edit->Select Polygon Group command.

Step 2 Check the Layer Check Boxes to the states shown in Figure 4.15.

Please read Appendix AT for the definition of Layer Check Boxes and what the Layer Window states for. Basically, the 3D polygons C and E in Figure 4.13 have vertices on Z = 10 and Z = 12 layers. We would like to select the 3D polygons C and E while we do not want to select the 2D polygons D on Z = 12 and the B and F on Z = 10. We checked the Z = 10 and Z = 12 layers and select the gray state of 3D Polygon layer because we want to select the 3D polygons with vertices on Z = 10 and Z = 12 only while we do not want to select the 2D polygons no both Z = 10 and Z = 12.

Step 3 Window the polygons C, D and E in Figure 4.15. You will see little change in the display of the top view main window. If you open the 3D view, you will see the two 3D polygons change color to black color meaning that the two polygons are selected.

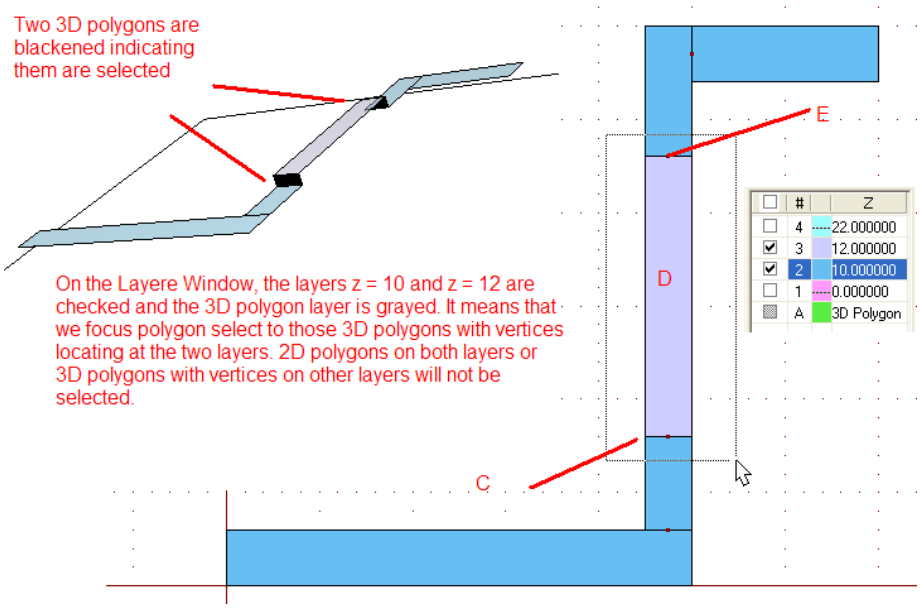


Figure 4.15 The Layer Window status and the windowing to select the 3D polygons.

Step 4 If you are still not sure whether the two 3D polygons are selected, you can select Edit->Object Properties command (or the bitmap with an eye on it on the tool bar). MGRID will show you the Object Properties dialog (see Figure 4.16. It is indicated there are total seven polygons and the only two 3D polygons are selected while the other 2D polygons on Z = 10 and Z = 12 are not selected.

This step is not necessary. It is just to demonstrate to you how you can get more information on MGRID. The Object Properties allows you to see the polygons and vertices if you are in drawing mode. If it is in select polygon modes, you will see the polygons. If you are in Select

Vertices mode, you will see the vertices information. In fact, you can do the editing such as selection inside the Object Properties dialog.

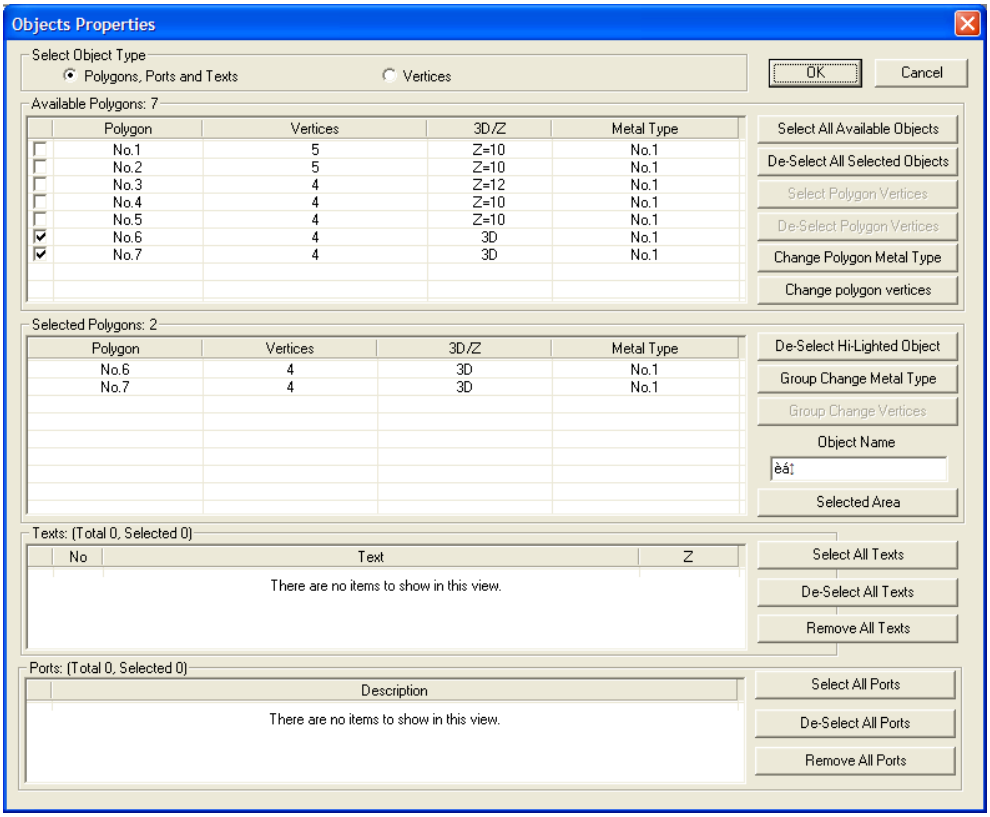


Figure 4.16 The Edit->Object Properties dialog.

Please select Cancel to resume to the previous state before this step.

Step 5 Select Edit->Delete command to delete the two selected 3D polygons. You may not see much change in the main top view. However, you will see the two 3D polygons are gone in the 3D View window (see Figure 4.17).

Section 4.7 Building 3D Polygons

We have demonstrated how to delete the 3D polygons. Now, we are going to show you how we can re-build the 3D polygons C and E in Figure 4.17. We will demonstrate two simple ways.

Step 1 Select Edit->Select Vertices. Check the Layer Check Boxes so that only the Z = 10 and Z = 12 layers are checked. The state of the 3D Polygon layer is not important in the Select Vertices mode. We can leave it as whatever state it is in.

Step 2 Window the edges for the 3D polygon C in Figure 4.17 to select four vertices. They belong to one edge (or two consecutive vertices) on Z = 10 and one edge (or two consecutive vertices) on Z = 12 are selected.

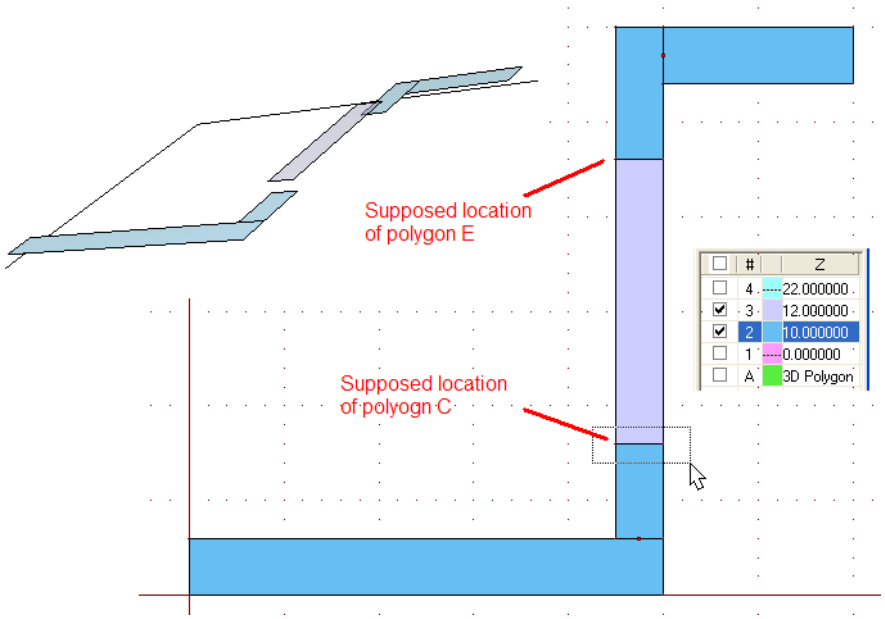


Figure 4.17 The 3D view window after deleting 3D polygons and the windowing for rebuilding polygon C.

Step 3 If you want to make sure the 4 vertices are selected, you can select Edit->Object Properties command. You will see the Object Properties dialog for Select Vertices mode (see Figure 4.18). It is indicated that the vertices 0 and 3 of polygon 3 (on Z = 12) and the vertices 0 and 3 of polygon 4 (on Z = 10) are selected

The Step 3 is not necessary and we just want to demonstrate to you how you can get more information from MGRID. Please select Cancel in the dialog to resume the state before Step 3.

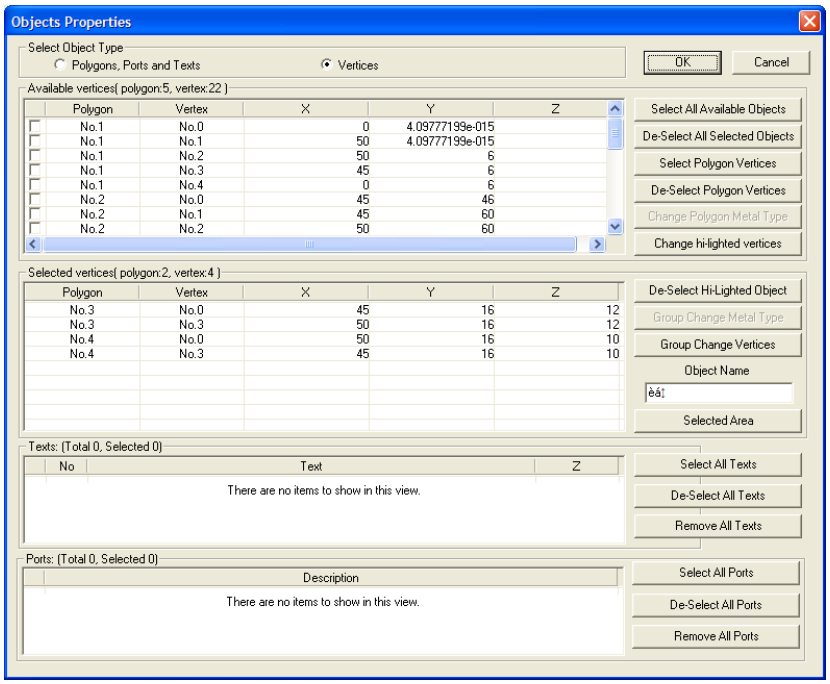


Figure 4.18 The Objects Properties dialog in Select Vertices mode.

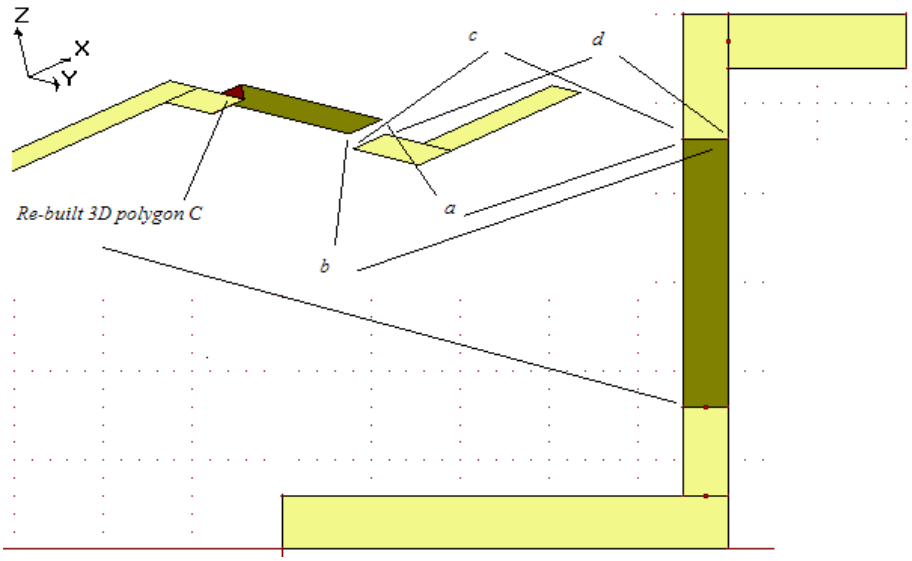


Figure 4.19 The re-built 3D polygon C and the vertices for 3D polygon E.

Step 4 Select Adv Edit->Build Connecting Path command. MGRID will re-build the 3D polygon C shown in Figure 4.19. The Adv Edit->Build Connecting Path command allows you to build a connecting polygon between 2 selected edges. Each of the edge can be 2D (two vertices on the same layer) or 3D (two vertices on different layers) and both edges are not required to be on the same layer.

Our next step is to re-build 3D polygon E. We certainly can use the same command to re-build it. However, we are going to demonstrate another simple way to re-build it.

Step 5 Make sure MGRID is in Drawing mode (Make sure Edit->Draw is checked).

Step 6 Click at No.3 Layer Z = 12 to focus input on Z = 12 layer. Select Input->Set to Closest Vertex for the snapping mode.

Step 7 Click at the vertex a in Figure 4.19 to snap a vertex at vertex a. Click at the vertex b in Figure 4.19 to snap another vertex at vertex b. The entered two vertices are for the polygon E and they are on Z = 12 layer. We are going to enter the vertices d and c on Z = 10.

Step 8 Click at the No. 2 Layer Z = 10 to focus input on Z = 10 layer. Click at the vertex d in Figure 4.18 to snap the 3rd vertex at vertex d. Click at the vertex c in Figure 4.19 to snap the 4th vertex at vertex c. By this step, we have entered four vertices (two on Z = 12 and two on Z = 10).

Step 9 Select Input->Form Polygon command to form the 3D polygon E. We get back to the same structure as the original bridge.geo.

Section 4.8 Re-Build the Bridge Structure by Change of Polygon Z-Coordinate

We have demonstrated to you how we can delete and build 3D polygons in Sections 6 and 7. There are many powerful commands in MGRID to help you build complicated structures. We are going to demonstrate to you an even simpler way to re-building the bridge.geo structure. The purpose of this demonstration is to let you practice on using the different commands wisely. We are going to delete the polygons C, D and E in Figure 4.14 and try to use a simpler way to re-build them.

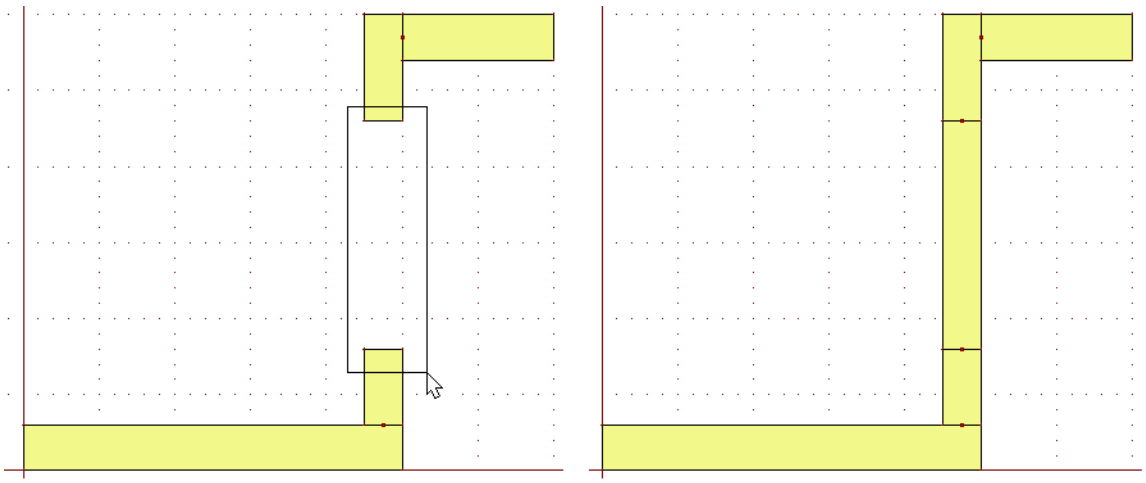
Step 1 Select Edit->Select Polygon Group.

Step 2 Check the No.2 Layer at Z = 10, No.3 Layer at Z = 12 and the 3D Polygon layer in the Layer Window. Whether other layers are checked is not critical. This time, we want to focus our selection on the Z = 10, Z = 12 and the 3D layers because we want to select the polygons C, D and E in Figure 4.14.

Step 3 Window the polygons C, D and E in Figure 4.13 to select them. Select Edit->Delete to delete the three polygons. Figure 4.20a shows the result and the windowing for the next step.

Step 4 Select Edit->Select Vertices command. Make sure Z = 10 layer is checked in the Layer Window. Window the 2 edges of the polygons B and F (see Figure 4.20a) to select the 2 edges. Select Adv Edit->Build Connecting Path and MGRID will fill the gap with a rectangle (see Figure 4.20b). By this step, the structure is of the exact shape as the structure in connect.geo we discussed in Section 1 except this structure consists of more polygons. How can we build the bridge from it?

Step 5 Select Edit->Select Polygon command. Check the Z = 10 layer in Layer Window. Click at the polygon for the gap in Figure 4.20b (or the polygon D in Figure 4.14). It is on Z = 10 right now.



(a) After polygons C, D and E deleted (b) After the gap is filled.

Figure 4.20 The structure after polygons C, D and E deleted and after the gap is filled.

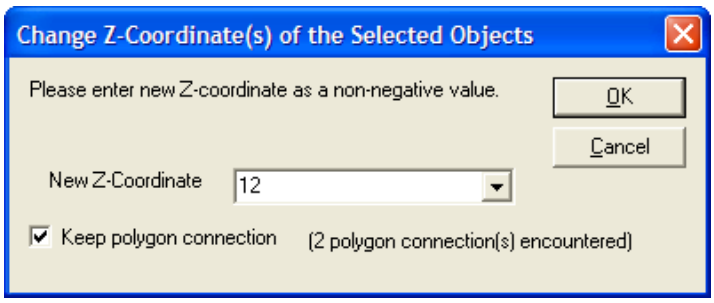


Figure 4.21 The Change Z-Coordinate dialog.

Step 6 Select Edit->Change Z-Coordinate. The Change Z-Coordinate dialog comes up (see Figure 4.21). Change the New Z-Coordinate from 10 to 12. Remember the check "Keep polygon

connection" (default setting). Select OK. MGRID will elevate the selected polygon on Z = 10 to Z = 12. It also keeps the connections between the elevated polygon with the other polygons on Z = 10 or it builds two 3D polygons to connect them automatically. By this step, we re-built the bridge.geo structure and we should see the picture in Figure 4.14.

Section 4.9 Re-Build the Bridge Structure with Rectangular Solid Vias

The original bridge.geo has two ribbon vias. The re-built structure is exactly the same as the original ones. In case we want to build some square solid vias instead of ribbon vias (see Figure 4.26), how can we do it?

We start from the structure in Figure 4.20b with the gap filled. If you have changed the structure discussed in last section, you can try to rebuild it following the instructions in Section 8.

Step 1 Select Edit->Undo to recover the structure in Figure 4.20b from the bridge structure created by elevating the polygon in the gap for the bridge. You should get exactly what is shown in Figure 4.20b.

Step 2 Press down Shift key and click at the polygon at the filled gap to select it. Select Edit->Change Z-Coordinate command. The Change Z-Coordinate dialog shown in Figure 4.21 comes up again.

Step 3 Please change the New Z-Coordinate from 10 to 12. Remember un-check the Keep Polygon Connection check box before you select OK button.

The polygon filling the gap is elevated. However, the connections to other polygons are not kept or there are no 3D polygons between this polygon at Z = 12 to other polygons at Z = 10 (see Figure 4.22). We are going to build two via pads on the polygon at Z = 12 from the edge (1, 2) and edge (3, 4). Edge (1, 2) means the edge formed by vertices 1 and 2.

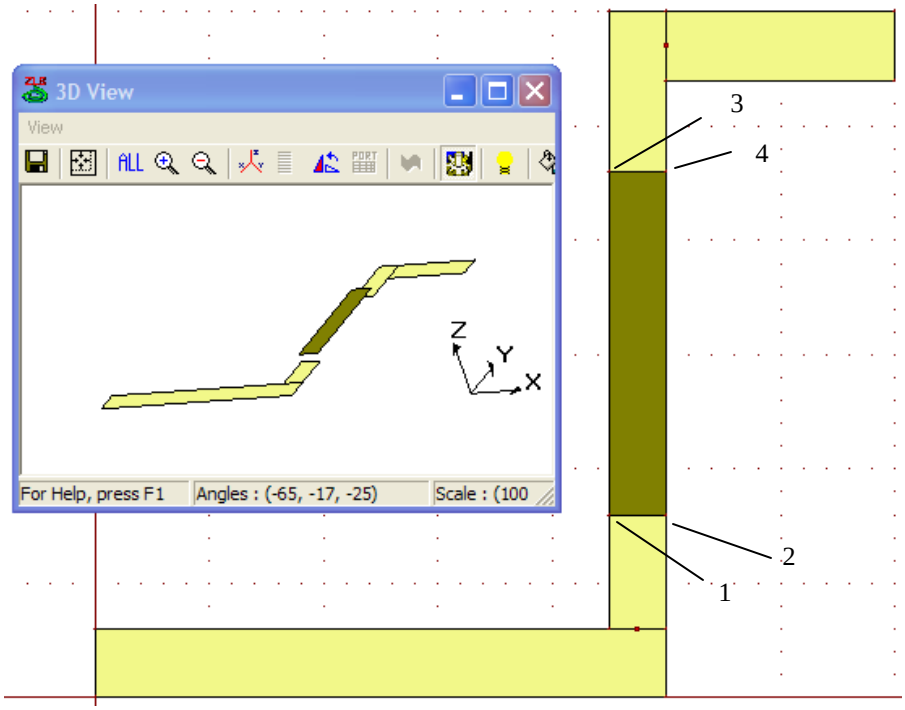


Figure 4.22 The elevated polygon without connections to Z = 10.

Step 4 Select Edit->Select Vertices command. Check the No.3 Layer at Z = 12 and uncheck the No.2 Layer at Z = 10. We are going to select the vertices 1 and 2 at Z = 12 while we do not want to select the corresponding vertices at Z = 10.

Step 5 Window the vertices 1 and 2 to select the edge (1, 2) at Z = 12. Select Adv Edit->Continue Straight Path command. MGRID will detect the width of the path = 5 and set some default values for the Path Start Width, Path End Width and the Path Length. In the Continue Straight Path dialog (see Figure 4.23).

We are allowed to create a path with tapered width of any length you want. Since we are building a square via pad, we just select OK to continue. A square pad is continued from the edge (1, 2) as shown in Figure 4.24. The connection between the rectangle and the square pad is guaranteed. The square pad should be exactly over portion of the polygon at Z = 10. This is important for us to use the following command to create the via. We are going to select all the vertices (or all the edges of the square pad) to build the via.

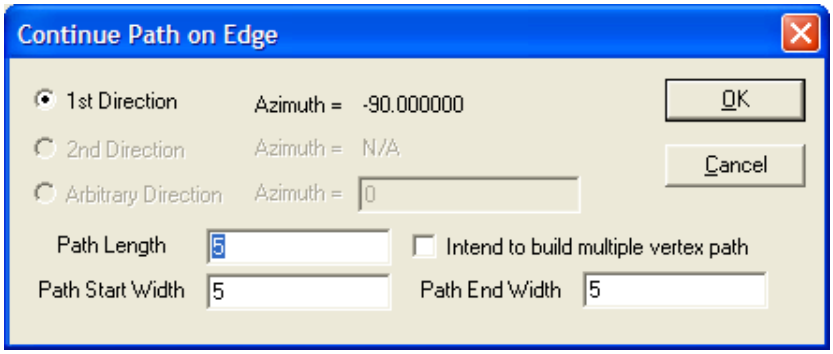


Figure 4.23 The Continue Straight Path dialog.

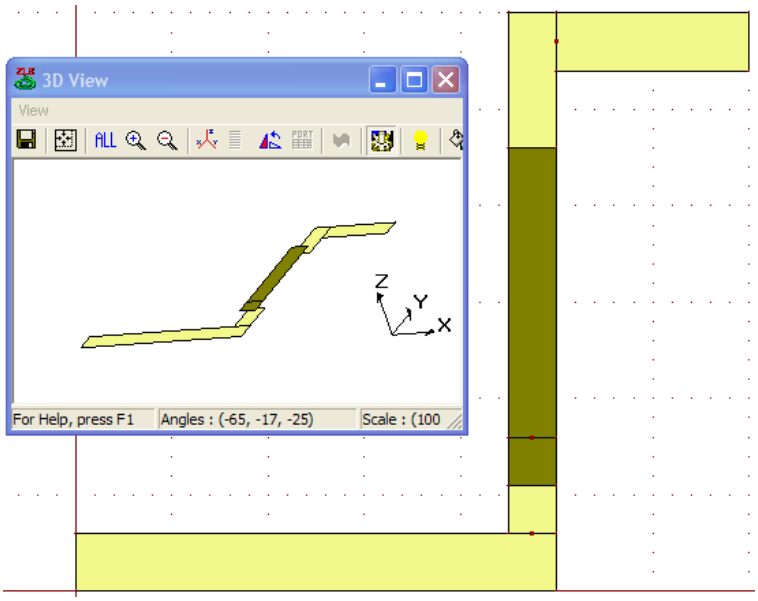


Figure 4.24 The structure with the 1st square pad created.

Step 6 Select Edit->Select Vertices again. Check the No.3 Layer at Z = 12 with other layers unchecked in the Layer Window again. Window all the vertices of the square pad to select them. In fact, the vertices 1 and 2 of the rectangle in Figure 4.21 are also selected. However, the edge (1, 2) is coincident with the corresponding edge of the square pad. We will consider only 4 edges of the square pad are selected.

Step 7 Select Edit->Add Via on Edges command. The Add Via on Edges dialog comes up (see Figure 4.25). It lists all the selected edges. Again, it indicates 4 edges are selected even though an additional edge (the edge (1, 2) from the rectangle in Figure 4.22) is selected. MGRID automatically detects the edge is coincident with another selected edge and it will not count it.

In IE3D 10.x and earlier version, MGRID will not do any connection at the end for the added via if no matching vertices or matching edges are available. On MGRID 11, you can see there is an option "Build Electrical Connection at End Z".

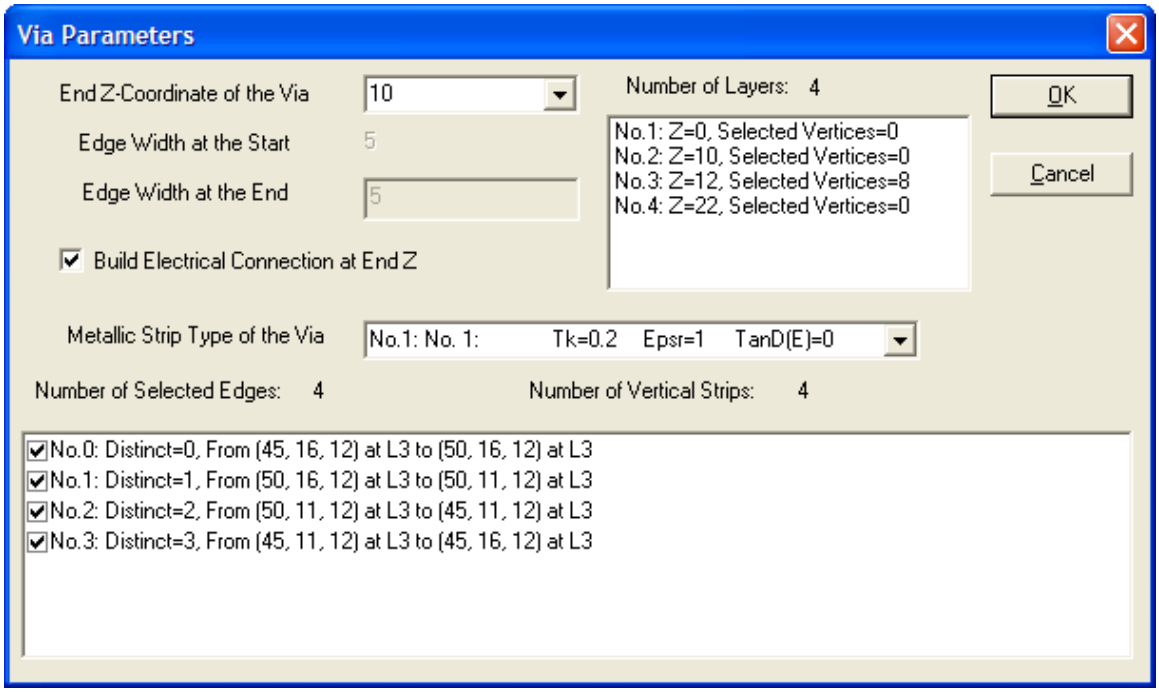


Figure 4.25 The Add Via on Edges dialog.

Step 8 Enter the "End Z-Coordinate of the Via" as 10. We want to build the via from Z = 12 to Z = 10. Please make sure the "Build Electrical Connection at End Z" is checked. Select OK to continue.

MGRID built the via. The 3D view looking from the bottom is shown in Figure 4.25. You can see the rectangle at Z = 10 underneath the via is broken down into 2 polygons. One of the polygon is a square matching the shape of the via pad. Basically, MGRID automatically build the connection for you starting from version 11. If you want to make sure the electrical connection is there, you can select the rectangle at Z = 12 and use the Adv Edit->Connection->Check Connection command to check it. MGRID will indicate to you the connection from the rectangle at Z = 12 to the via and to the polygons at Z = 10.

Step 9 Please repeat the Step 4 to Step 8 for the other via on the edge (3, 4) in Figure 4.21. You should get the structure saved in .\ie3d\samples\bridge1.geo with rectangular solid vias. The final structure is shown in Figure 4.26.

The solid vias are in fact hollow vias with metal on the sides. This is not an approximation. Mathematically, they are the same as solid vias and they are very accurate in the modeling.

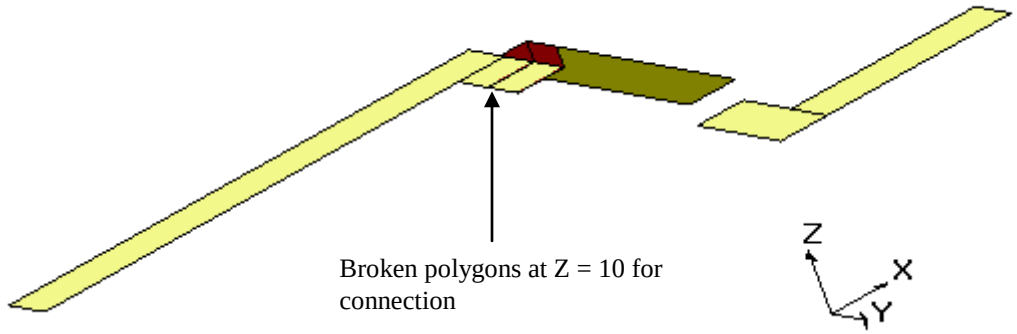


Figure 4.26 The via built with the Edit->Add Via on Edges command with guaranteed connection.

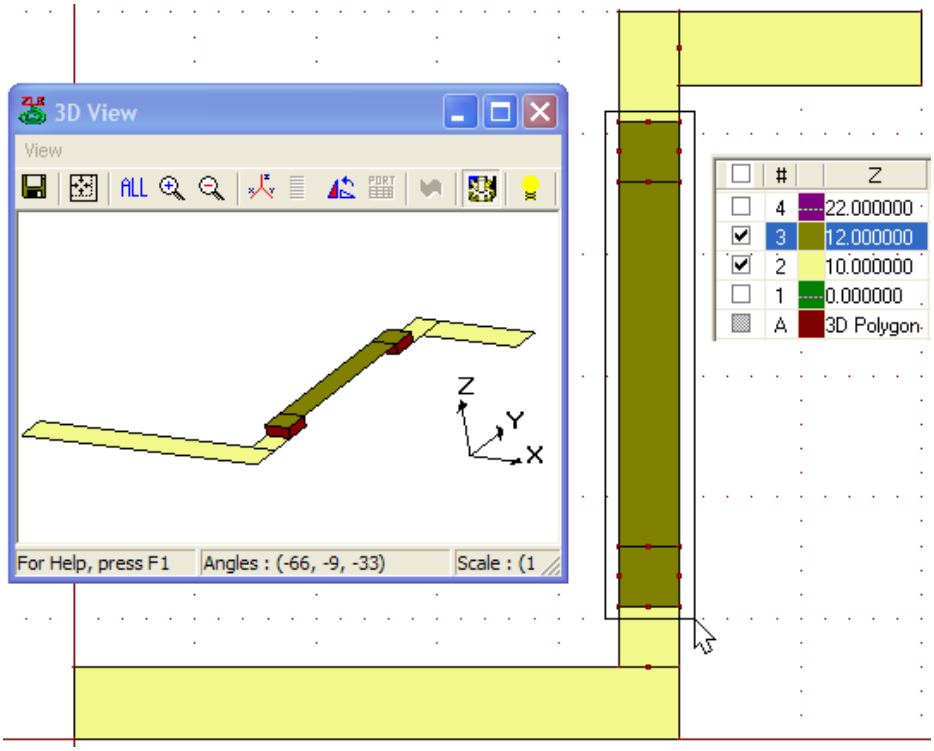


Figure 4.27 The bridge structure with rectangular solid via.

Section 4.10 Re-Build the Bridge Structure with Cylindrical Solid Vias

In the last section, we have demonstrated to how we can build a square via of the same shape as the via pad. You may wonder how we can build a cylindrical via. This section will demonstrate to you how we can build cylindrical vias. The square is of size 5 by 5 mils. We would like to build a cylindrical via of radius 1.5 mils at the center of each square via pad. We will take the square via structure bridge1.geo as our starting point.

Step 1 Open .\ie3d\samples\bridge1.geo. Select Edit->Select Polygon Group command. Check the 3D Polygon layer on the Layer Window to gray state. Check the No.2 Layer Z = 10 and the No.3 Layer Z = 12. You should get the Layer Window to what is shown in Figure 4.27.

Step 2 Window the bridge as shown in Figure 4.27. Only the 3D polygons are selected while the 2D polygons on Z = 10 and Z = 12 are not selected. This is because of the gray state of the 3D Polygon layer in the Layer Window (Please read the Appendix AT for detail).

Step 3 Select Edit->Delete command to delete the selected 3D polygons. The via pads and other 2D polygons are still in the MGRID window (see Figure 4.28).

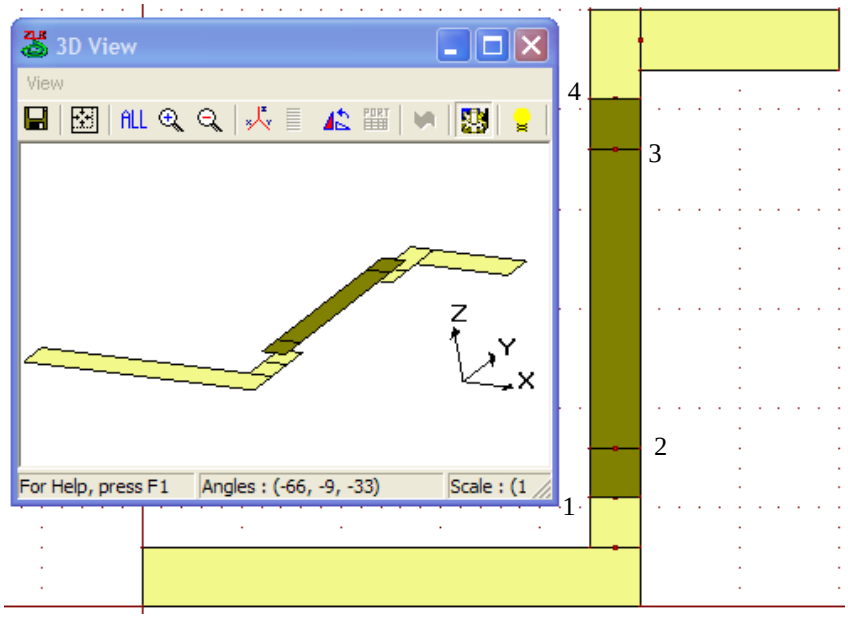


Figure 4.28 The bridge1.geo structure after the 3D polygons are deleted.

Step 4 Save the file as: .\ie3d\samples\bridge1a.geo. We will use it later when we demonstrate how we can build conical vias.

Step 5 We are going to create some polygons on some fictitious layer to representing the cylindrical vias first. We will use the via pads on Z = 10 to create the polygons.

Select Edit->Select Polygon Group command. Check No.2 Layer at Z = 10 and un-check all other layers. Window the bridge as shown in Figure 4.27. You will see only the two via pads at Z = 10 are selected. Select Edit->Copy command to copy the selected squares at Z = 10 into the clipboard. Select Edit->Paste to paste the two squares from the clipboard. You will see the shapes of the two squares are following the mouse when you move it.

Step 6 Click some where at the window. MGRID will prompt you the Copy Object Offset to Original dialog. Please enter the X-offset = 0, Y-offset = 0 and Z-offset = 50. It means that we want to copy the two squares to some where at the same (X, Y)-location. However, we want its Z-coordinate to be 50-mils larger or at Z = 60 mils. Change the After Paste option from "All Objects De-Selected" to "Only Pasted Objects Selected" (see Figure 4.29). Select OK to continue. MGRID will copy the two squares at the same (X, Y)-location and Z = 60 mils (see Figure 4.30).

You will notice the two copied squares are being selected. In case you forgot to check "Only Pasted Objects Selected", the two copied squares will not be selected. You can use the Edit->Select Polygon Group command to select them for the next steps.

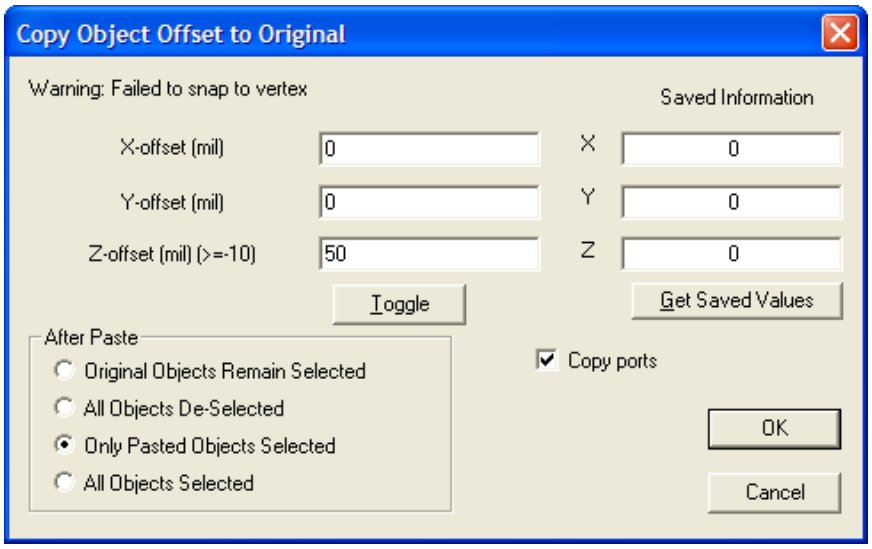


Figure 4.29 The Copy Object Offset to Original dialog.

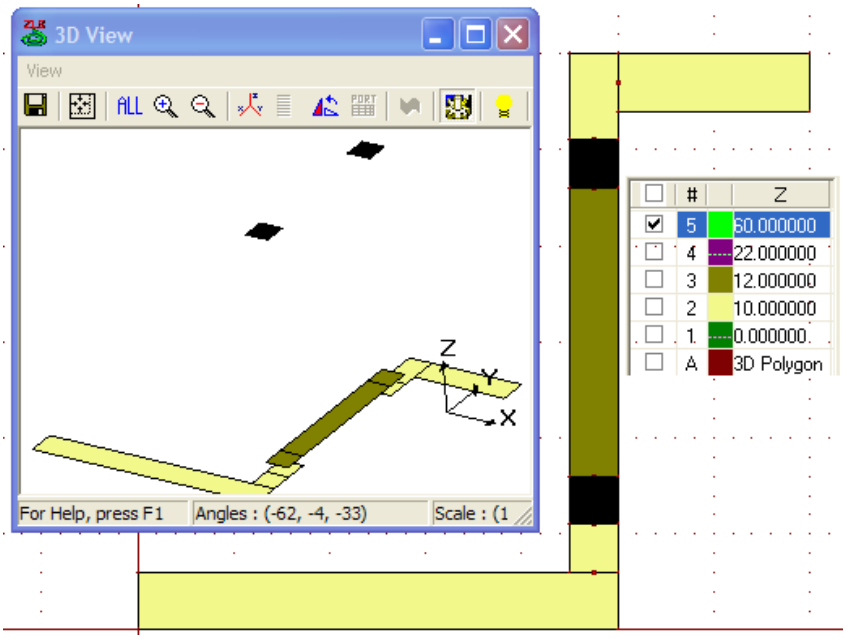


Figure 4.30 The two squares at copied at Z = 60 and the same (X, Y) location.

Step 7 We are going to select the shapes of the square at Z = 60 to create the shapes of the cylindrical vias. The shape of each cylindrical via is of circular shape. On IE3D, we will model a cylindrical via as a polygonal via. When the via's size is electrically small, we can use ribbon via or square via for it. Normally, it is very accurate to use hexagonal or octagonal via to model a circular via. In this example, we will try to use hexagonal via for it. We want to create the hexagons first.

Select Adv Edit->Convert Polygons in Shape command. The Convert Polygons dialog comes up. Select the "No.1 …" in the list box. Make sure the Converted Shape is "Circle". Change the Radius to 1.5. Enter Segments/Circle = 6. Enter the Circle Start Point = 0 (see Figure 4.31). You will see MGRID is updating the information in the "No.1 …" item in the list box while you are entering the parameters.

You may enter "Start Point" much when you use many of the commands related to circular shape. Basically, we are using a polygonal shape to model a circular shape. There is a question where the angle of the start vertex is at. We define the "Start Point" as the fraction of one division angle for the polygonal shape in approximating the circular shape. The Start Point = 0 corresponds to that the 1st vertex is at an angle of 0 degree. The Start Point = 0.5 corresponds to that the 1st vertex is at an angle of 50% of one division (see Figure 4.32).

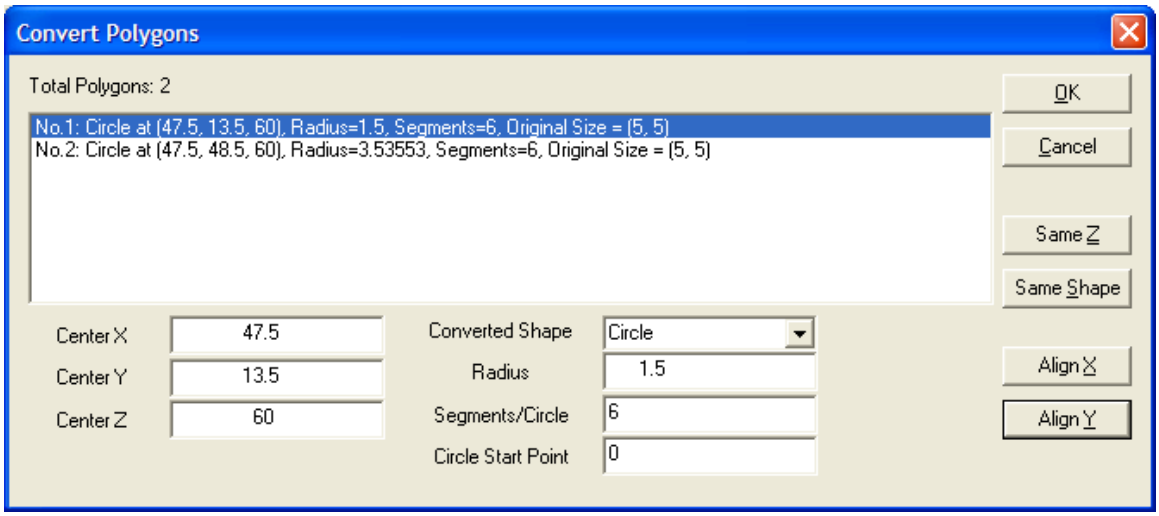
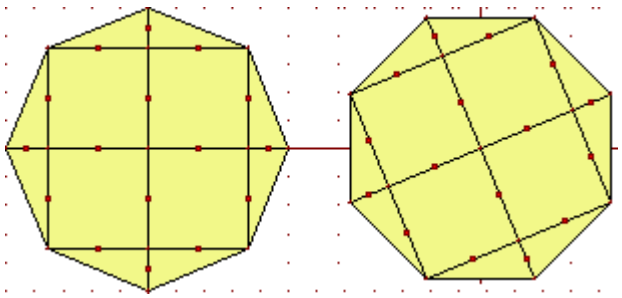


Figure 4.31 The Convert Polygons dialog after you define the parameters for the 1st shape.



(a) Start Point = 0 (b) Start Point = 0.5

Figure 4.32 The difference between different Start Points for circular shape.

Step 7 Click at the Same Shape button in the Convert Polygons command. MGRID will update the shape information about the No.2 polygon with the information defined for the No.1 polygon while it will not change the center location of the No.2 polygon. Basically, we want to have the two rectangles converted to hexagons of the same shape at their individual center locations.

Step 8 Select OK. MGRID will convert the two rectangles into hexagons as shown in Figure 4.33. We are going to use the hexagons to build the vias between Z = 10 and Z = 12.

Step 9 Select Edit->Select Polygon Group command. Check the Z = 60 layer in the Layer Window only. Window the two hexagons to select them. Select Adv Edit->Build Holes and Vias from Selected Polygons. The dialog comes up.

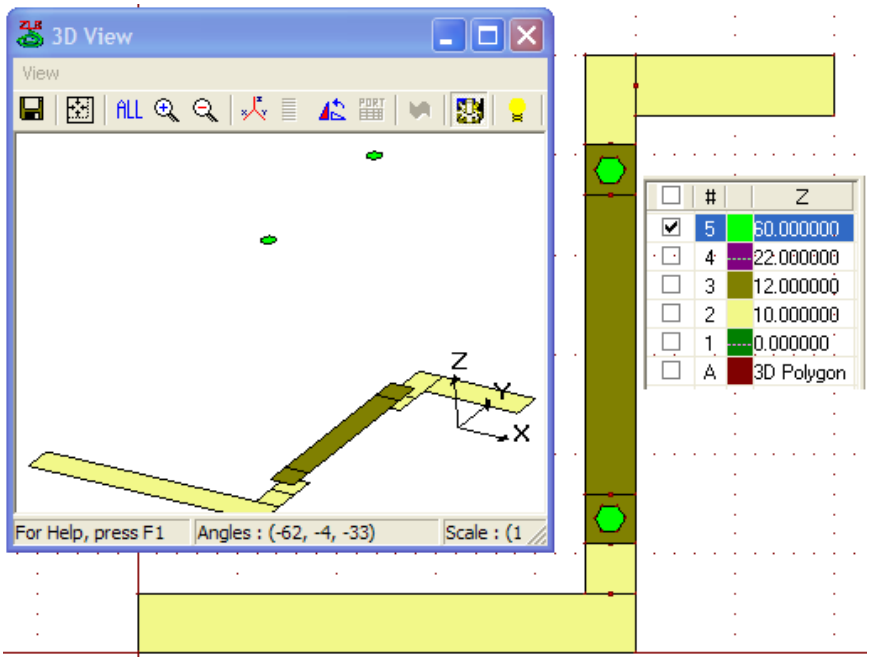


Figure 4.33 The two rectangles at Z = 60 are converted into hexagons.

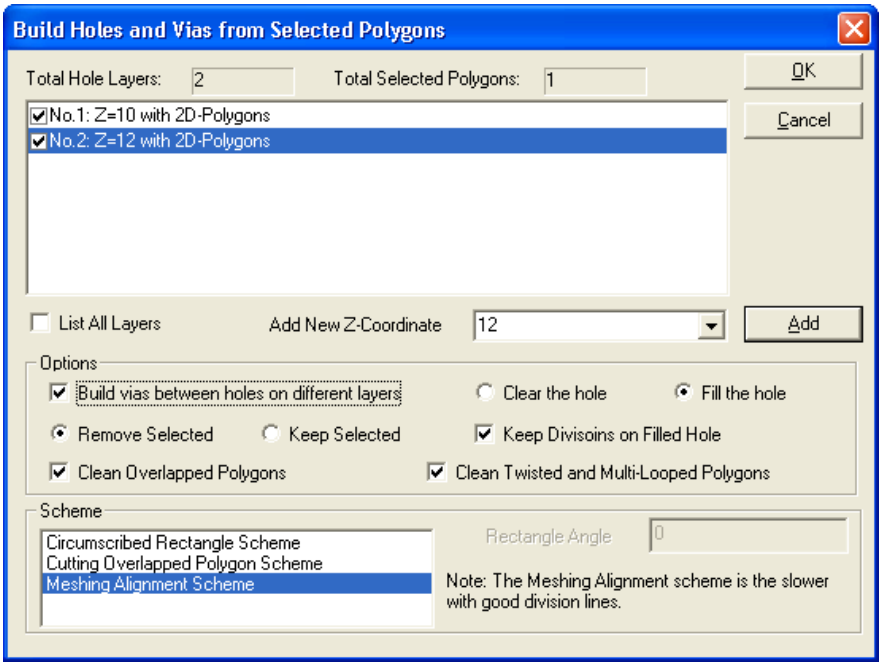


Figure 4.34 The Build Holes and Vias from Selected Polygons dialog after you enter all parameters.

There are many different options for the Build Holes and Vias from Selected Polygons dialog. There are three schemes to build the holes: (1) Circumscribed Rectangle Scheme"; (2) Cutting Overlapped Polygons Scheme; (3) Meshing Alignment Scheme. The Circumscribed Rectangle Scheme has been the scheme used before IE3D 11. It is very robust. One shortcoming is that it relies upon a circumscribed rectangle to build the hole. Then, we try to duplicate the shape of the polygon inside the circumscribed rectangle. If the polygons where we are building the whole shape into cannot cover the circumscribed rectangle of the selected polygon, it may create extra polygons we do not want. Another shortcoming is that it cannot preserve the division lines of the polygons we want to build the hole.

To overcome the above limitations of the Circumscribed Rectangle Scheme, we have introduced the Meshing Alignment Scheme. It is a very powerful scheme. Normally, it can build the holes cleaner than the Circumscribed Rectangle Scheme while you can have the option to Keep Divisions on Filled Hole. It was implemented in IE3D 11 and it is now very reliable and robust. It is the default scheme to be used.

You have the option to "Clear the hole" or "Fill the hole". Basically, MGRID builds a hole of the same of a selected polygon on a specified layer added into the list. If you choose "Clear the hole", MGRID will the hole only and it will not fill the hole after it creates it. If you choose "Fill the hole", MGRID will fill the hole with the polygon shape. You have the choice to Keep Division Lines on Filled Holes. This option is useful when you are building vias on both sides of a layer. You may apply the Build Holes and Vias from Selected Polygons command twice to the layer: Once for the via connection from above to the layer and the 2nd time for the via from below to the layer. The 2nd time may meshed up the polygon division and create disconnected polygons for the 1st via from above to the layer. If you choose the "Keep Division Lines on Filled Holes for the 2nd time, you will be able to guarantee the two vias are connected to the layer.

Another option is to "Remove Selected" or "Keep Selected". It is to inform MGRID whether you still want to keep the selected polygons after you use their shapes to build the holes.

When you checked or added more than one Z-coordinate to the list for holes, MGRID will activate the option for you to "Build vias between holes on different layers". Since we are building vias using this command, we should check it.

Step 10 By default, the layers z = 10 and 12 are listed but not checked. Please check the Z = 10 and Z = 12 layers. If necessary, you can select Add button to add new values the list. Now, the "Build vias between holes on different layers" option is activated. Check it. We will get the dialog exactly as shown in Figure 4.34.

Step 11 Select OK to continue. MGRID will build the cylindrical vias as shown in Figure 4.35. It is what we want.

The Build Holes and Vias from Selected Polygons command is very powerful and it is frequently used in the IE3D for Boolean operations. You should practice it more and learn its power. It will help you much in building vias and interconnects in complicated structures.

Section 4.11 Re-Build the Bridge Structure with Conical Solid Vias

We have demonstrated to you how we can build a cylindrical via in the last section. You may ask how we can build a conical via because conical vias are also encountered quite frequently in MMIC circuits. This section will demonstrate to you how we can build conical vias. We assume the conical via has a radius of 2 mils at Z = 10 and 1 mil at Z = 12. We will use the bridge1a.geo saved earlier to do it.

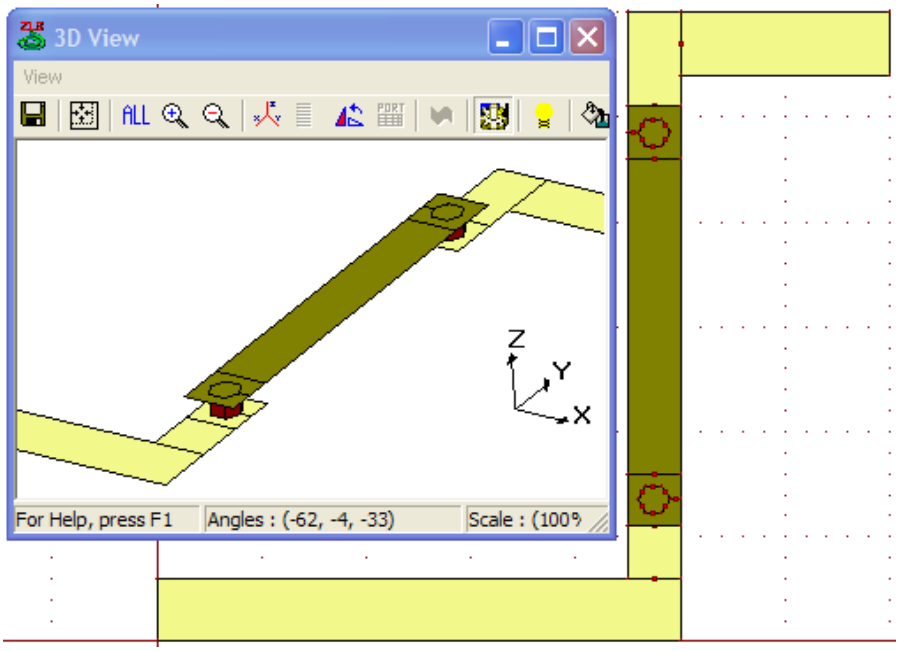


Figure 4.35 The structure with cylindrical (hexagonal) vias built.

Step 1 Open .\ie3d\samples\bridge1a.geo. It is the structure with all the four via pads on the two layers for the two vias.

Step 2 Click at the No.2 Layer at Z = 10 to focus the input at Z = 10. Select Input->Set to Closest Vertex for the snapping mode. Click at the vertex 1 in Figure 4.22 to snap a vertex at it. Click at the vertex 2 in Figure 4.22 to snap the 2nd vertex at it. We basically entered two vertices along the diagonal of the 1st via pad.

Step 3 Select Input->Insert Mid Point command. MGRID will prompt you the Insert Mid Vertex dialog. Select OK to accept the default with the "Drop Last 2 Vertices". MGRID will drop the two entered vertices. However, at the mid-point of the two vertices, it created one vertex.

Step 4 Click at the vertex 3 in Figure 4.22 to snap another vertex at it. Click at the vertex 4 in Figure 4.22 to snap another vertex at it. By this step, you will have three vertices entered: The 1st one at the center of the 1st via pad, and the 2nd and 3rd ones along the diagonal of the 2nd via pad.

Step 5 Select Input->Insert Mid Point command again. Select OK to accept the default settings. MGRID will drop the 2nd and 3rd vertices along the diagonal of the 2nd via pad. Instead, it creates a vertex at its center. By this step, we have two vertices left: the 1st one at the center of the 1st via pad and the 2nd one at the center of the 2nd via pad (see Figure 4.36). There is a line connecting the 2 vertices and another line connecting the 2nd vertex to the mouse location. Basically, we can use the two entered vertices to locate where we want to build the conical vias.

Step 6 Select Entity->Conical Via to bring up the Conical Via Parameters dialog (see Figure 4.37). Change the Number of Segments for Circle to 6 (original 0 for square via). Enter the Start Z-Coordinate = 10, the Start Radius = 2, the End Z-Coordinate = 12 and the End Radius = 1. Select the Division Option = Aligned Meshing. Select Close Cap option. The final state of the dialog is shown in Figure 4.37. Select OK to continue. The conical vias are built as shown in Figure 4.38.

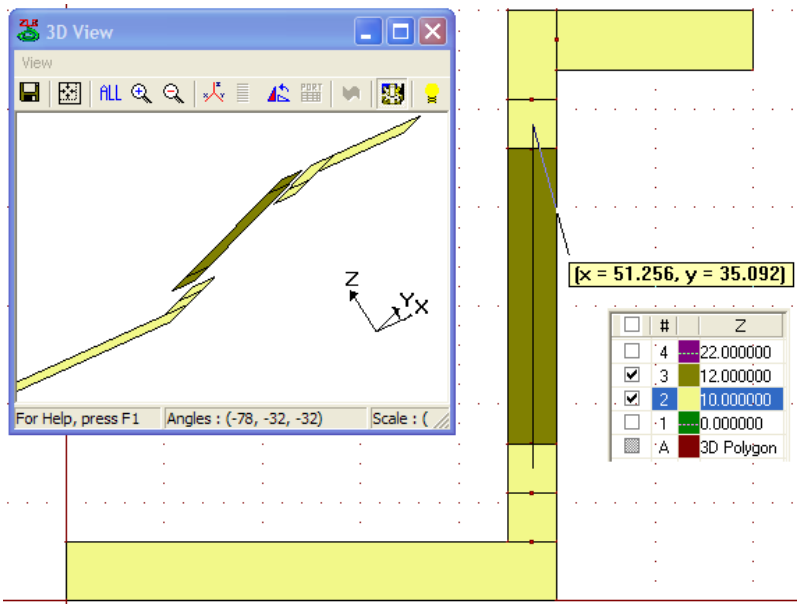


Figure 4.36 The structure with the two center locations of vias entered.

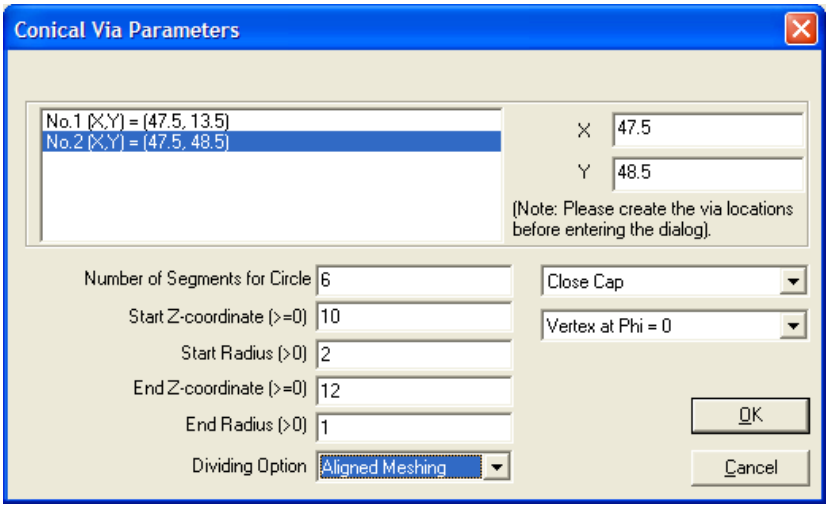


Figure 4.37 The Conical Via Parameters dialog.

Section 4.12 Re-Build the Bridge Structure with Ramp Ribbon Vias.

We have demonstrated quite some commands in constructing different vias for the bridge structures. Many of those commands are frequently used in the geometry modeling. If you know how to use the commands wisely, you should be able to create and modify most of the structures you encounter. When you practice it enough, you will be able to find a command immediately when you encounter a specific problem. For example, we would like to ask you how you could modify the bridge.geo file to create ramp ribbon vias instead of vertical ribbon vias (see Figure 4.39). Can you find out what commands you should use to modify the structure for it?

In fact, the procedure is quite simple. Assume each ramp spans 2 mils. What we can do is to select the vertices 1 and 2 at Z = 12 in Figure 4.38a. Then, we select Edit->Move Objects to move the two vertices

by X-offset = 0, Y-offset = 2 and Z-offset = 0. Then, we can select the vertices 3 and 4 at Z = 12 in Figure 4.39a. We move the vertices by X-offset = 0, Y-offset = -2 and Z-offset = 0. We will get the ramp ribbon via from the vertical ribbon via.

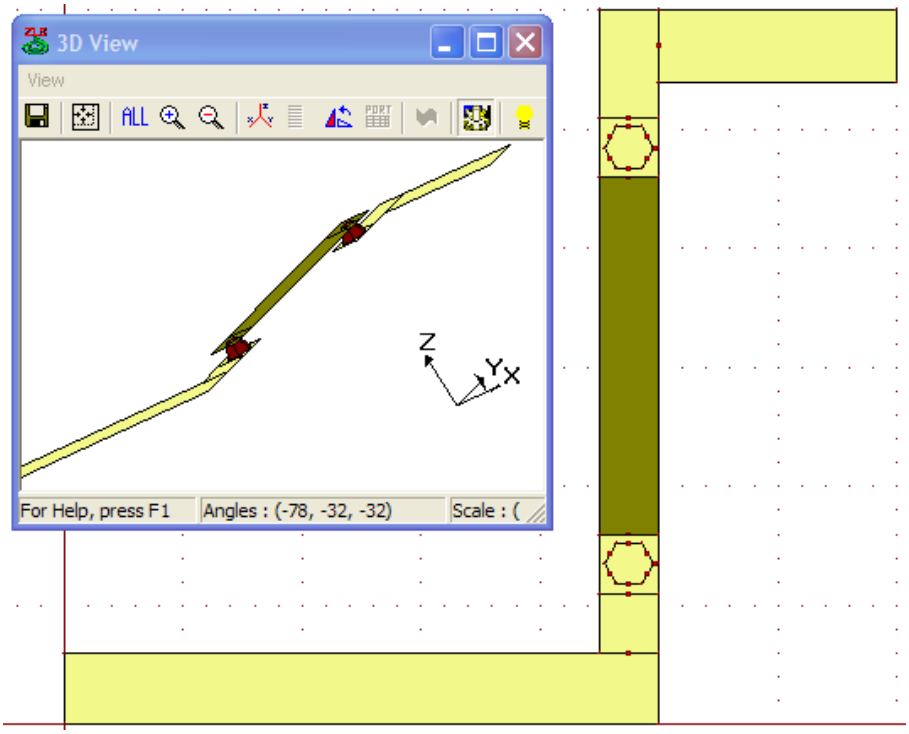
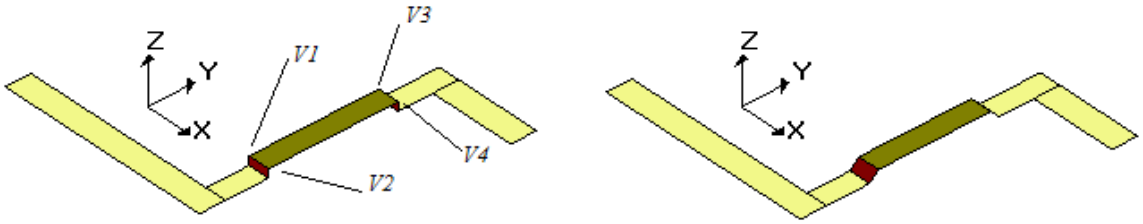


Figure 4.38 The structure with conical vias built.



(a) Vertical ribbon vias (b) Ramp ribbon vias

Figure 4.39 Comparison of vertical ribbon vias and ramp ribbon vias.

Section 4.13 Handling of Pin Via Structures

For microwave, antennas and PCB applications, we will use big solid vias and ribbon vias. For RFIC applications, pin array vias are normally used. A big group of tiny pin vias are created in a small region to yield good electrical connection. An example is in .\ie3d\samples\bridge_pins_vias.geo shown in Figure 4.40. For this structure, there are 2 groups of pin vias and each group consists of 100 tiny vias. The detail of a group is shown in Figure 4.40.

An RFIC layout is normally in GDSII format. The traces and the pin via shapes are represented as layered polygons. IE3D has powerful GDSII handling capability. Simple GDSII importing capabilities are integrated into MGRID while the powerful GDSII handling is performed on AGIF (Automatic Geometry to IE3D Flow). AGIF is part of the IE3D MM08X or Special Flow Edition. On AGIF, you are able to build an

IE3D model directly from a GDSII file. You are even able to streamlined IE3D batch simulations directly from a batch of GDSII files. For more detail, please read the AGIF User's Manual.

Saved in .\ie3d\samples\bridge_pins.geo is an intermediate structure with pin vias represented as some flat polygons on Z = 42. We will demonstrate how you can build the bridge structure with vias from it.

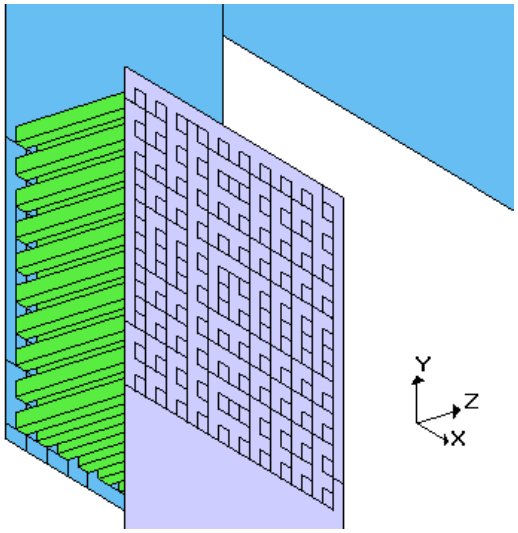


Figure 4.40 Pin array vias in RFIC designs.

Step 1 Open .\ie3d\samples\bridge_pins.geo.

There are 2 layers of polygons on Z = 10 and 12 and they are the traces. There are two groups of small squares on Z = 42. They are the shapes of the vias. If we select the polygons on Z = 42 and select the Adv Edit->Build Holes and Vias from Selected Polygons. Pick the layers on Z = 10 and Z = 12. Check Build Vias between Holes on Different Layers. We will be able to build the pin array via structure saved in .\ie3d\samples\bridge_pins_vias.geo. The building via process will take some minutes while the simulation will take even more due to the huge amount of detail.

In practical design, we should not model the pin vias as they are. We should merge a group of small polygons into a big polygon to build a big via on it.

Step 2 Select Edit->Select Polygon Group. Check the Z = 42 layer on the Layer Window. Window the two groups of small squares to select them (see Figure 4.41). The small squares will turn to black color after they are selected.

Step 3 Move the mouse to an edge of a small square and pay attention to the Geometry tab of the Information bar. In case the Information bar is not available (on the bottom of the window), please select View->Information Bar to activate it. When the mouse is on the right layer (Z = 42) and it is close to an edge of a polygon, the Geometry tab of the Information bar is showing the information on the edge. It shows the (x, y, z) of each vertex of the edge and the relative location of the two vertices. It shows Rho = 0.2380952381 and it means the horizontal distance of the edge is about 0.2380952381 mils. The size is much bigger than a regular pin polygon size in RFIC. It is ok because we just use the example to demonstrate what we should do when we encounter pin vias.

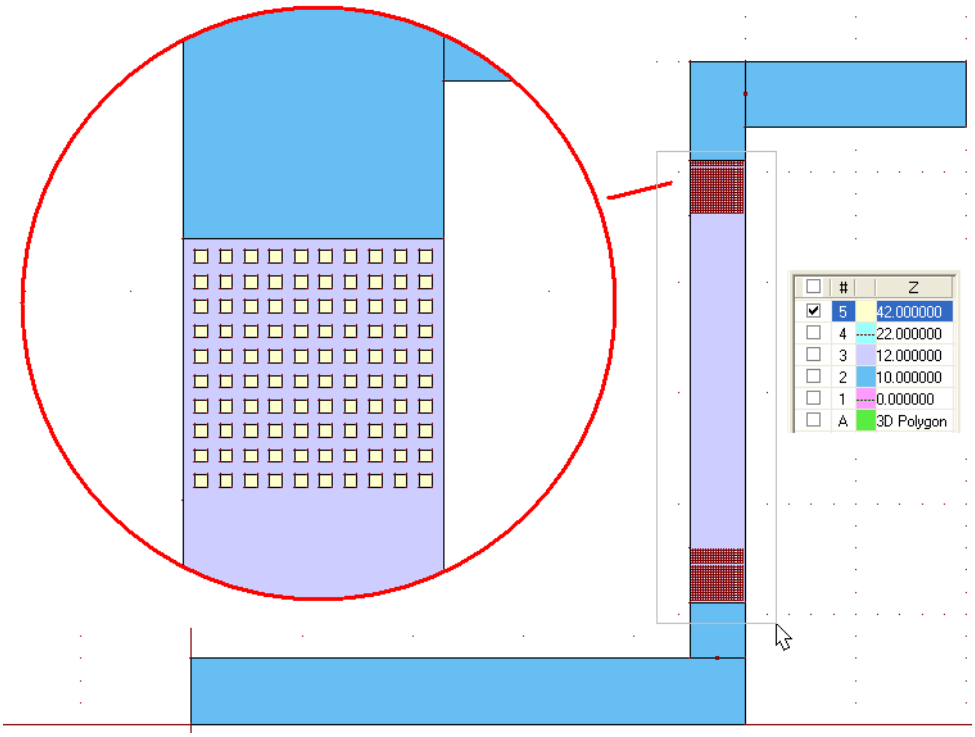


Figure 4.41 The detail of a group of small squares and the setup to select the two groups of them.

Step 4 Select Adv Edit->Mesh and Merge->Merge Selected Polyogns on Distance to bring up the dialog (see Figure 4.43). Enter the value for Minimum Distance Between Groups as "0.8". This value should be larger than the gap between two adjacent squares while it should be smaller than the distance between groups. We have two groups of squares and the distance is much bigger than this value. The gap between two squares is about 0.238 mils. A value of 0.8 is good.

Step 5 Select OK and MGRID will merge the small squares into two big squares (see the top view in Figure 4.44).

Step 6 Select Edit->Select Polygon Group. Focus the selection on Z = 42. Window the two big squares merged from the small squares to select them. Select Adv Edit->Build Holes and Vias from Selected Polygons. Check the Z = 10 and Z = 12 layers. Check Build Via Between Holes on Different Layers. Check Fill the Holes option. Select OK. MGRID will build the bigger vias on the bridge (see the 3D view in Figure 4.44.

Section 4.14 Summary

IE3D is more than a regular planar or 2.5D EM simulator. It can handle arbitrary 3D metallic structures. It can even handle structures with finite dielectrics. It is able to solve many full 3D structures while the applications of IE3D are focused on structures in layered dielectric environment.

A layered structure mainly consists of layered traces and vias between them. This chapter has discussed the most important geometry modeling commands used in building models on trace and via structures. From this chapter, you should have learned some of the most frequently used commands and techniques in build an IE3D model. We will give more examples in the chapters as follows.

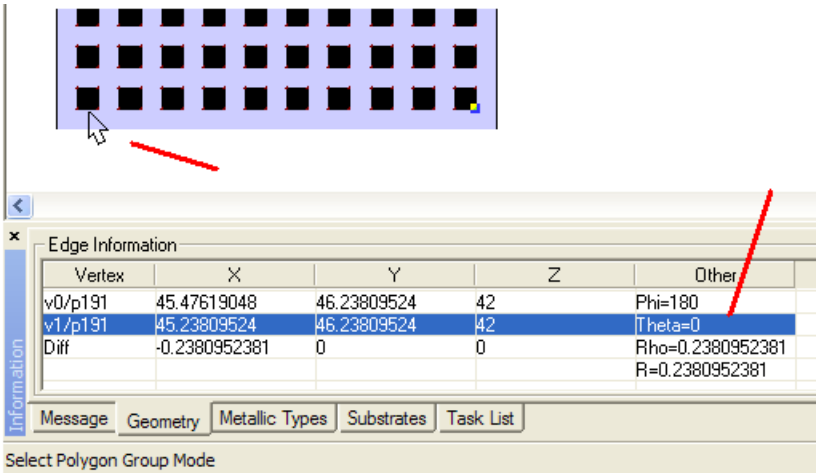


Figure 4.42 The Geometry tab in the Information bar are showing the detail information on the edge.

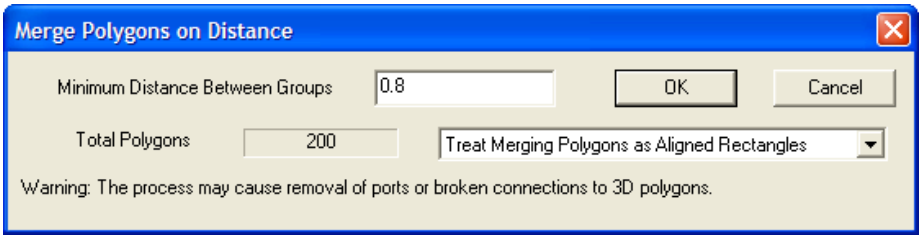


Figure 4.43 The Merge Selected Polygons on Distance dialog.

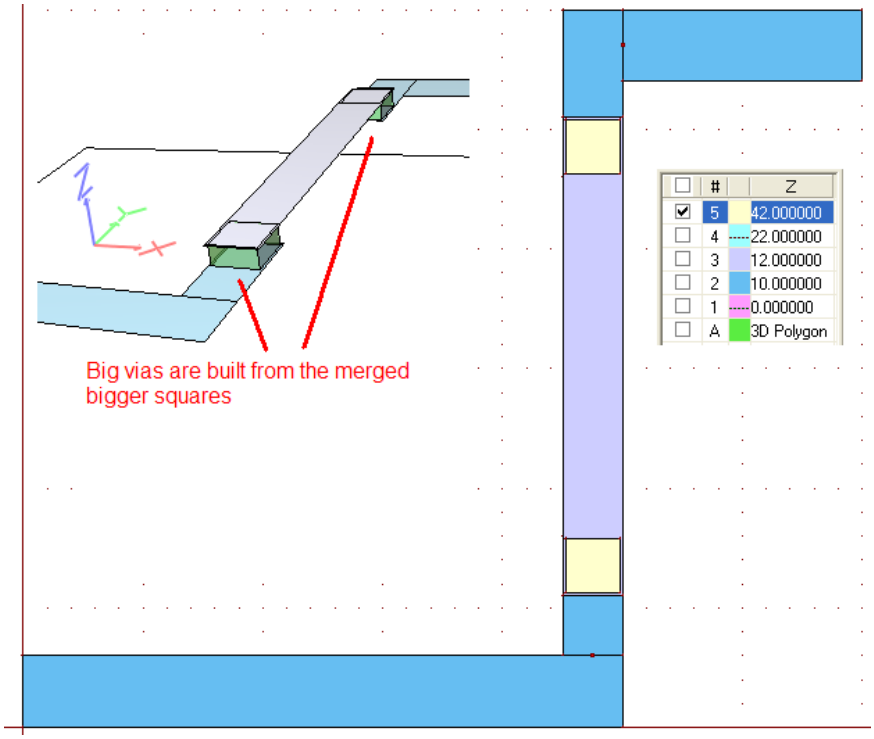


Figure 4.44 The merged bigger squares and the big vias built from them.

Chapter 5 Antennas, Radiation Patterns and EM Optimization

We have discussed the basic technique and geometry modeling in the last two chapters. We are going to demonstrate the applications of IE3D in practical designs in the chapters starting from Chapter 5.

One of the most important applications of the IE3D is for antenna design. We will demonstrate how we can use IE3D to perform simulation and optimization, current visualization and radiation pattern calculation.

Section 5.1 Modeling of an Edge Fed Antenna with an Inset.

The first example is an edge fed rectangular patch antenna with inset. The top view is shown in Figure 5.1. The parameters of the antenna are shown in Table 5.1. The question is how we can build the antenna. We will show you how you can use the efficient commands on MGRID to build the structure in just a few steps.

Table 5.1 The parameters of the patch antenna.

Substrate Thickness	31 mils	Dielectric Constant	4.4
Patch Length, L	1512 mils	Patch Width, W	1500 mils
Inset Width, S	115 mils	Inset Depth, D	452 mils
Strip Width, T	60 mils	Feed Line Length, F	750 mils

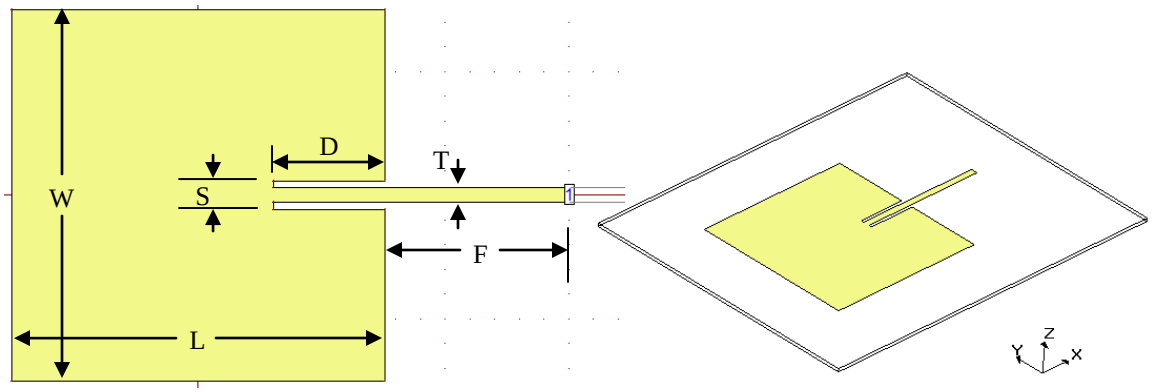


Figure 5.1 The illustration of a simple edge-fed patch antenna.

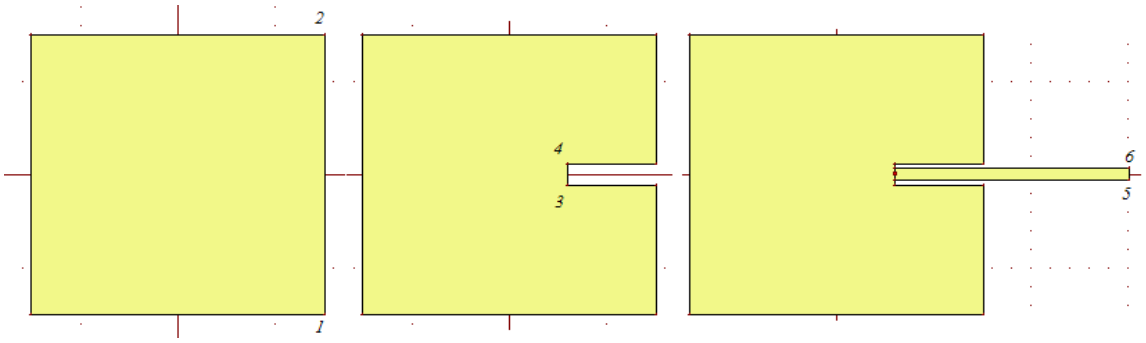
Step 1 Run MGRID. Select File->New command. MGRID shows you the Basic Parameters dialog. In the Length group, select the Unit as "mil". In the Meshing Parameters group, change the Highest Frequency (Fmax) to 2 GHz and the Cells per Wavelength (Ncell) to 15. Double click the "No.1 Grid Size…" in Layout and Grids group. Change it to 100 mils and select OK when you are prompted to edit the Grid Size. Select the Insert (or New) button in the Substrate Layers group. MGRID will prompt you for "Insert New Substrate". Enter "Top Surface, Ztop" as 31 (mils). Enter the Dielectric Constant, Epsr = 4.4. Select OK and the entered substrate is listed in the Substrate Layers list box. We do not need to change the Metallic Strip Types because metallic loss is normally not critical to antennas when the frequency is not very high. Select OK to continue. MGRID is ready for geometry input.

If you do not know how to define the Basic Parameters, please open .\ie3d\samples\rpatch1.geo file and select Parameters->Basic Parameters to see how we define them.

Step 2 We are going to build a rectangle without the inset first. Select Entity->Rectangle. MGRID will prompt you for the parameters of the rectangle you want to build. Enter (X, Y, Z) = (0, 0, 31).

Enter Length = 1512 and Width = 1500 in the Rectangle Properties group. Select OK to continue. A rectangle is created with center at (X, Y, Z) = (0, 0, 31) as shown in Figure 5.2a.

Step 3 Press down "Shift" and window the vertices 1 and 2 in Figure 5.2a to select them. Select Adv Edit->Cut into Polygon on Edge command. MGRID prompts you for the parameters. Please enter Cut Width = 115 and the Cut Depth = 452. Select OK. MGRID will build the inset as shown in Figure 5.2b.



(a) Square patch (b) With inset (c) With inset and feed line

Figure 5.2 The shape of the structure at different stages.

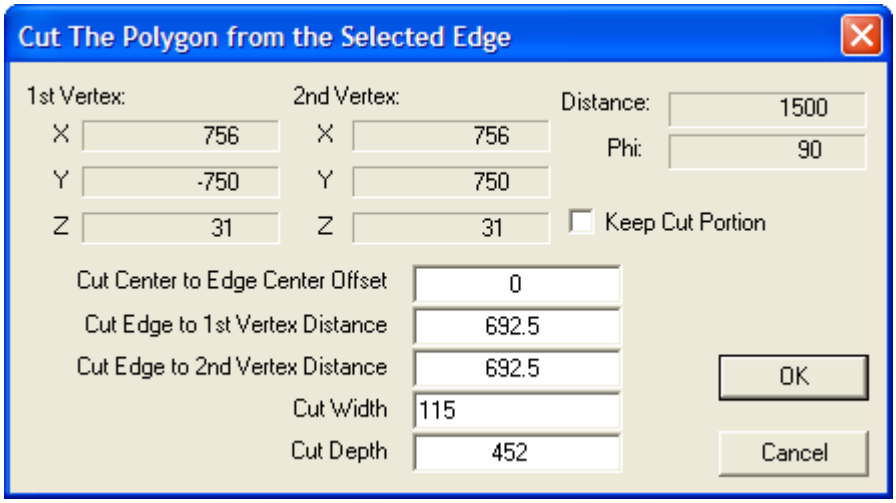


Figure 5.3 The Cut into Polygon on Edge dialog.

Step 4 Press down "Shift" and window the vertices 3 and 4 in Figure 5.2b to select them. Select Adv Edit->Continue Straight Path. Enter the Path Length = 1202 (=452+750), Path Start Width = Path End Width = 60. Select OK. MGRID will build the feed line as shown in figure 5.2c.

Step 5 Select Port->Port for Edge Group. Choose Advanced Extension for the De-Embedding Scheme. Check the Auto Adjustment. Select OK to accept other default settings. Window the edge formed by vertices 5 and 6 in Figure 5.2c to define the port. Select Port->Exit Port to exit the state. Save the structure as: .\ie3d\practice\rpatch1.geo. The structure is ready to be simulated.

Step 6 Select Process->Display Meshing. Select AEC Layers = 1. The default AEC Ratio = 0.1 and the corresponding Width = 23.9433 mils. We know the feed line is 60 mils and the gap between the feed line and the edge of the patch is about 27.5 mils because it is 0.5*(115-60) mils. We may want to reduce the AEC Ratio to 0.05 so that the AEC Width becomes smaller compared to the gap width.

Step 7 Change the AEC Ratio from 0.1 to 0.05. The AEC Width becomes 11.9716 mils.

Please select the Meshing scheme as: Classical. Select OK to continue. MGRID will mesh the structure and create 148 cells and 256 unknowns. Select OK and MGRID displays the meshing result in Figure 5.4. The meshing result is very good. For this particular structure, the Contemporary meshing result is not as good.

Again, Process->Display Meshing is not a necessary step for a simulation. However, it is a good habit to check the meshing before some simulations in order to make sure there is no potential accuracy problem, especially when you get more experience and understand how meshing may affect simulation accuracy and efficiency.

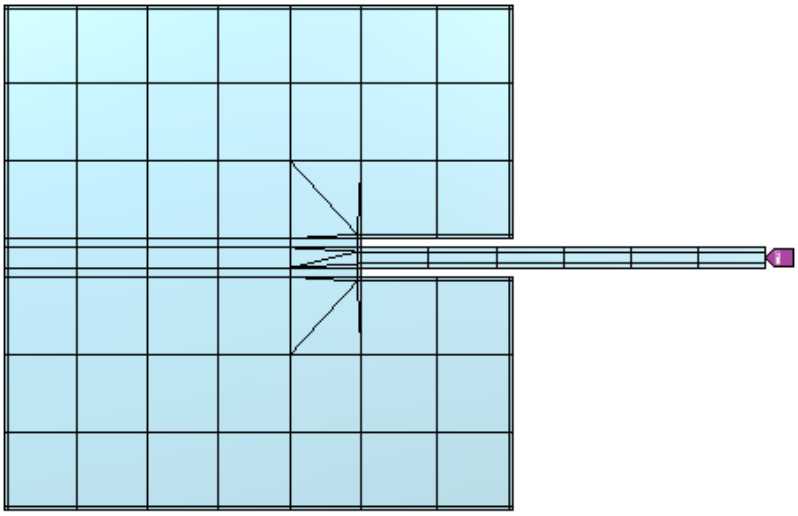


Figure 5.4 Classical meshing result on the patch antenna.

Step 8 Select Process->Simulate command. The Simulation Setup dialog comes up (see Figure 5.5). We expect the resonant frequency to be about 1.88 GHz. We would like to simulate the structure from 1.7 go 2 GHz with 151 frequency points. Select the Enter button in Frequency Parameters dialog. MGRID prompts you for the frequency range. Enter Start Freq = 1.7, End Freq = 2, and Number of Freq = 151. Select OK to add the frequency parameters into the list box. The AEC setting should be the same as before. Make sure AIF is checked.

Please select No Display for Post Processing. We don't need to use MODUA to display the s-parameters normally. We are going to use the integrated visualization on MGRID. Select the Define Graphs button (Figure 5.5). It brings up the "S-Parameters and Frequency…" dialog. Select Add Graph button. You are prompted for the Graph Type. Select S-Parameters. You will be prompted for Display Selection for the Graph. Select dB[S(1,1)]. Select OK. The defined graph is added into the list. Select Add Graph button again. Select Z-Parameters and OK. Select Re[Z(1,1)] and Im[Z(1,1)]. Select OK to define the second graph. Select Add Graph button again. Select Smith Chart and OK. Select S(1,1) and OK to define the third graph. Select Continue to go back to the Simulation Setup dialog. You will see "3 graphs are defined" next to the "Define Graphs" button. What we want to do is to jump into s-parameter visualization directly by pre-defining the graphs.

Select OK to continue. MGRID will invoke IE3D to perform the simulation in the background. It takes seconds to finish. After simulation, IE3D will create 3 graphs for the s-parameters dB

plot, z-parameters and the Smith Chart displays (see Figure 5.6). On MODUA, you can only display one graph at a time. In the integrated IE3D EM Design System, you can visualize different parameters simultaneously.

As you can see, the resonance is not exactly at 1.88 GHz. It is resonating at about 1.87 GHz with best return loss as -5.8 dB.

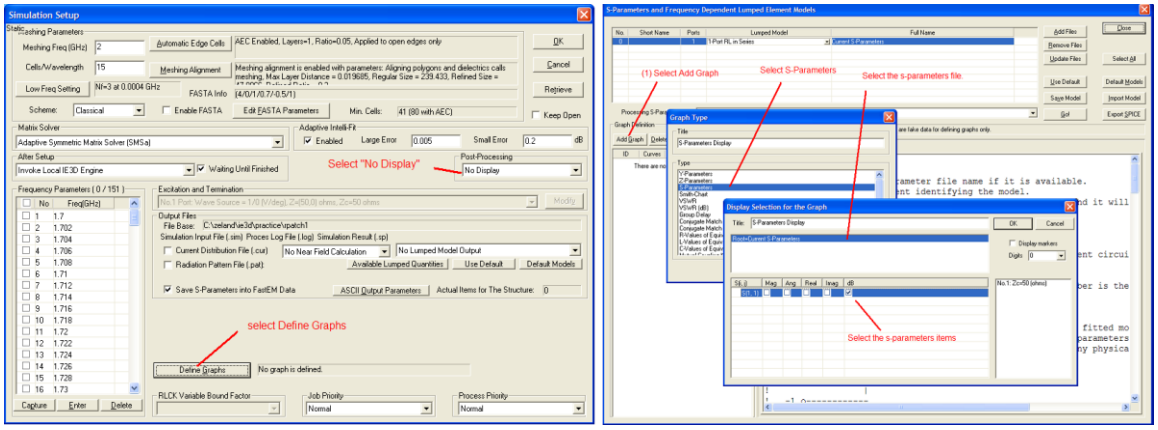


Figure 5.5 The Simulation Setup dialog and the dialogs for defining a graph.

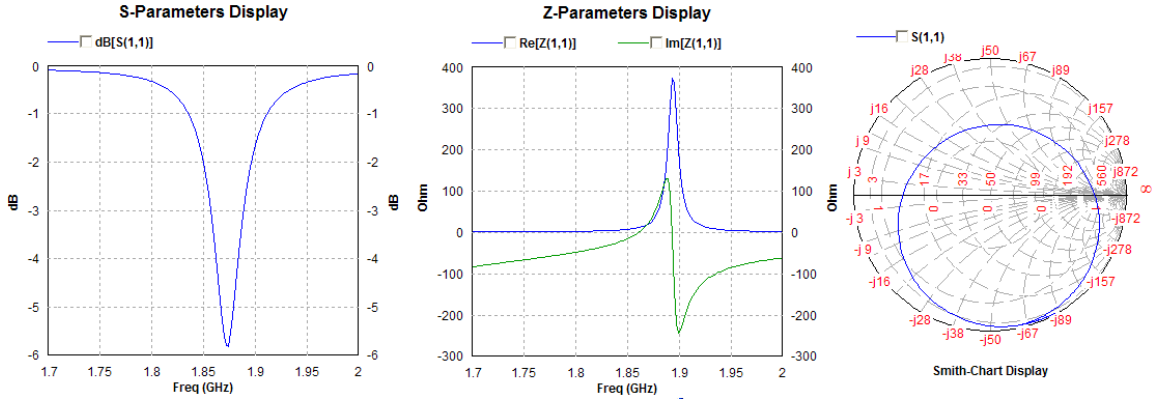


Figure 5.6 The simulation results from different meshing schemes.

Section 5.2 EM Optimization.

Assume we want to optimize this antenna to a perfect match at 1.88 GHz. We would like to study what we can do for it. EM optimization is one of the most important features of IE3D. It can help designers to achieve their goal with much less effort. Some goal may even be impossible by manual tuning. EM optimization may yield excellent results for you. However, please understand the following fact: EM optimization is not supposed to replace designers. Users should know what they want to do and what they can do before they jump into optimization. If you throw a garbage design into an EM optimizer, it is quite likely it will throw a garbage design back to you.

Before you start an optimization, you should first consider the followings: (1) What are your goals? (2) What geometry dimensions we can adjust to achieve the goals?

For the antenna we are discussing, we can adjust the length L to change the resonant frequency. We can adjust the inset depth D to tune the matching. Also, both variables are not independent or changing L

may change the matching and changing D may change the resonance. If they are independent, we can optimize one dimension at a time and it may make an optimization much easier. Our goals are: Re[S(1,1)] = 0 and Im[S(1,1)] = 0 at 1.88 GHz.

On MGRID, structures are not described as parameterized objects. All structures are described by polygons and polygons are described by vertices. To change the shape of a structure, we need to change the locations of vertices. We need to identify which vertices we should adjust to change the L and D values. IE3DLIBRARY has complete parameterized structure objects with equation-based dimensions. It is extremely flexible in optimizing structures. In this chapter, we will concentrate on MGRID.

We show two groups of vertices in Figure 5.7: Vertex Group 1 includes vertices 7, 8, 9 and 10. Vertex Group 2 includes vertices 3, 4, 11 and 12.

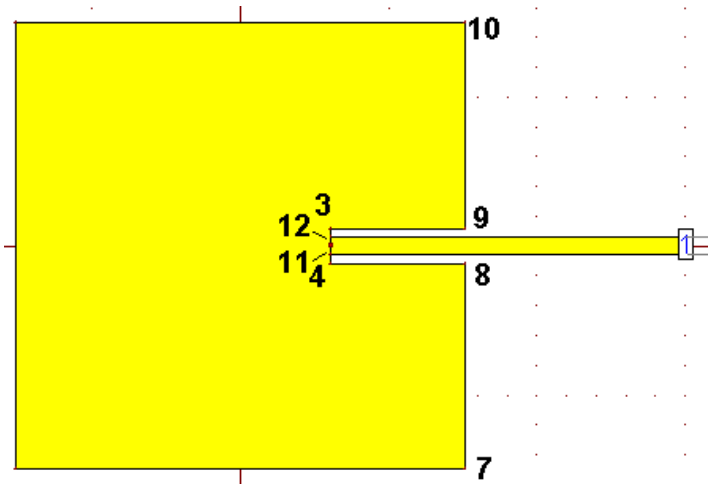


Figure 5.7 The patch and its vertices to be defined as optimization variables.

To change the patch length L, we can change the X-coordinate of Vertex Group 1. To change the inset depth D, we can change the X-coordinate of Vertex Group 2. Can we define the X-offset values of the two groups as optimization variables? Certainly, we can do it. However, it is not the best.

The reason is related to the bounds of the optimization variables. We know the difference in the X-coordinate between Vertex Group 1 and Vertex Group 2 is D = 452 mils. If we define the X-offset of Group 1 as the first variable V1, and the X-offset of Group 2 as the second variable V2, we need to make sure the high bound of V2 can not be too big and the low bound of V1 can not be too small. Otherwise, we may cause the vertex 3 to go to the right hand side of vertex 8. For this particular structure, such a situation may not create an invalid geometry. However, there is quite a chance we may create invalid geometry when the bounds of optimization variables are not defined properly. As you will see later, we are defining the variables and their bounds one by one. We may not know the bounds or ranges of previous variables when we are defining a specific variable.

For this particular structure, we would like to define the X-offset of the two groups as V1. Then, we define the extra X-offset of Group 2 as the V2. In this way, when we define the high bound of V2, we just need to make sure V3 will not go to the right hand side of V8 visually.

Step 1 Select Edit->Select Vertices. Window the Vertex Group 1 (or vertices 7, 8, 9 and 10) only. We can select the Vertex Group 1 and Vertex Group 2 simultaneously to define them as the No.1 variable. It is simpler. However, we would like to demonstrate how we could associate different vertex groups to one single variable by taking two steps with one step for one group.

Step 2 Select Optim->Variable for Selected Objects. MGRID will prompt you for the Optimization Variable Definition dialog. The default is "Vertices Mapped to New Variable" (see Figure 5.8). Please make sure the Tuning Angle = 0 because we want to change the x-coordinate of the vertices. Select OK. MGRID is in the Low Bound definition mode. Move the mouse to the left and you can see the frame of the structure is changing. Click the mouse at appropriate location. MGRID will prompt you to Set Low Bound. Please enter the Low Bound = -100. We know the L should be close and we do not need to define very large bounds. MGRID may warn you the change of geometry will lose the results. Select YES to continue.

After you select OK, MGRID is in the mode for defining the high bound. Move the mouse to the right and you can see the frame of the structure changing. Click the mouse at appropriate location. MGRID will prompt you to Set High Bound. Please enter the High Bound = 100 and select OK. MGRID will prompt you the Defining No.1 Variable Finished dialog. It allows you to check bounds. It also allows you to make some random trials to see whether the bounds are properly defined. The random check may be able to capture possible improper settings of the bounds causing invalid structure. In such a case, MGRID allows you to find the bounds which may avoid invalid geometry problem.

Step 3 Select Continue Without Action button. MGRID will finish the definition of the first variable.

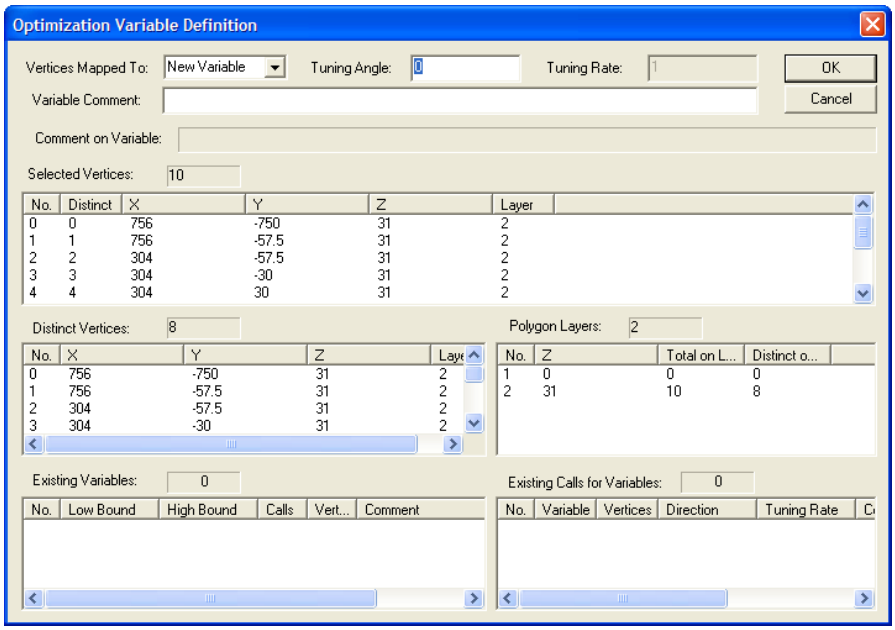


Figure 5.8 The Optimization Variable Definition dialog for the Optim->Variable for Selected Objects

Step 4 Select Edit->Select Vertices command. Window the Vertex Group 2 (or vertices 3, 4, 11 and 12) to select them. We are going to associate the vertices with the first variable.

Step 5 Select Optim->Add Selected Objects to Variable command. The Optimization Variable Definition dialog comes up again. It is very similar to Figure 5.8 except it indicates "Vertices Mapped to No.1 Variable" and the Tuning Rate is activated with a default value of 1. The default is that you want to add the selected vertices to the existing No.1 variable. If you like, you can change it to "Vertices Mapped to New Variable" to define the second variable. For this case,

we will demonstrate how you define both Vertex Group 1 and Vertex Group 2 as the No.1 variable. We should accept the default.

We also want the Vertex Group 2 to change in Tuning Angle = 0. We want the Tuning Rate = 1 so that the Vertex Group 2 will change at the same speed as the Vertex Group 1. Please note that the Tuning Angle and Tuning Rate can be set different for every Optim->Add Selected Objects to Variable command. You have the flexibility for tuning very complicated structures.

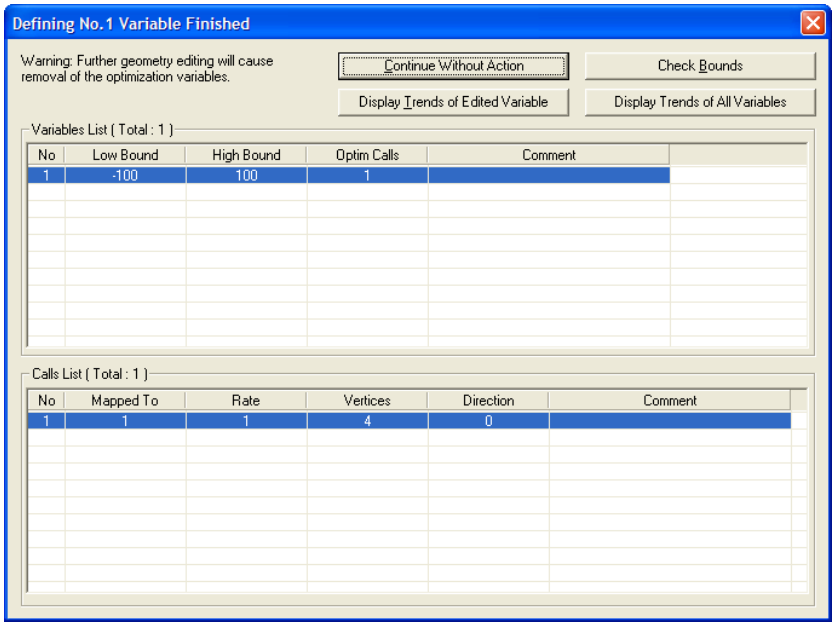


Figure 5.9 The Defining No.1 Variable Finished dialog.

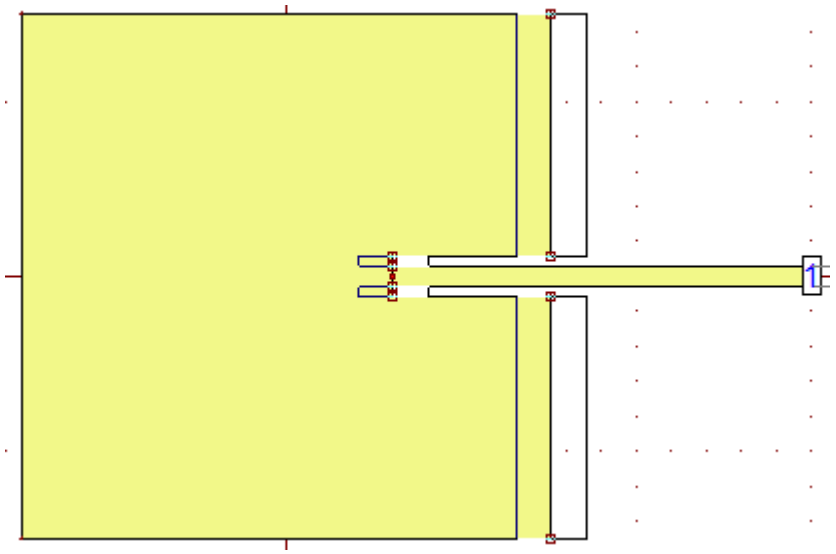


Figure 5.10 The trends and bounds of the No.1 variable.

Step 6 Select OK. MGRID will show the Defining No.1 Variable Finished dialog. Select Display Trends of Edited Variable. MGRID will close the dialog and show you the trends of the bounds and you can see the range of the vertices when it is tuned (see Figure 5.10).

Step 7 Select View->Refresh to clear the display of the bounds. Select Edit->Select Vertices. Window the Vertex Group 2 again. We will define the Vertex Group 2 alone as the second variable. Select Optim->Variable for Selected Objects. The Definition of Optimization Variable dialog comes up. By default, it is "Vertices Mapped to New Variable". The Tuning Angle = 0 and it is what we want. Select OK to continue. MGRID is in defining Low Bound mode. Move the mouse to the left and click. Enter the Low Bound as -200. Select OK. MGRID is in defining High Bound mode. Move the mouse to the right and click. Enter the High Bound as 200. Select OK. MGRID prompt you "Definition of No.2 Variable Finished" dialog. You can choose Continue Without Action or Display Trends of Edited Variable or Display Trends of All Variables. If you are not sure whether the bounds of the variables are defined correctly, you can also select Check Bounds to let MGRID to check whether the bounds are properly defined.

Step 9 Select Display Trends of All Variables. MGRID finished the definition and display the trends of the two variables for you. You can also select Optim->Geometry Tuning to see how the variables are controlling the shape of the antenna.

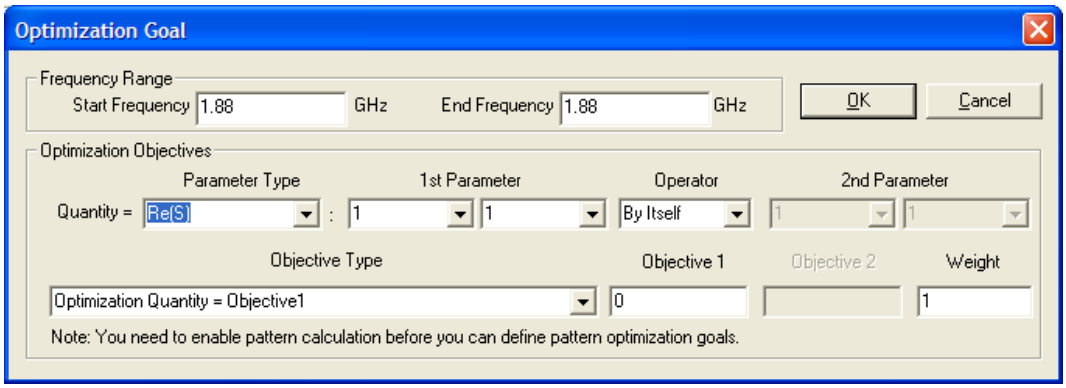


Figure 5.11 the Optimization Goal dialog.

Step 10 We are ready to optimize the structure. Please save it as: .\ie3d\practice\rpatch2.geo. Select Process->Optimize command. The Optimization Setup dialog comes up. It is the same as the Simulation Setup dialog except there is some minor change. Select Delete button in Frequency Parameters group. Choose Delete All and select OK to define the 151 frequency points in the list. Select Enter in the Frequency Parameters group. Enter the Start Freq = 1.88, End Freq = 1.88 and Number of Freq = 1. Select OK to add the frequency parameters into the list.

Please select the Optimization Scheme as the Powell. It is good local optimizer even though the Adaptive EM Optimizer is the best for general optimization. Please read the Appendix AU for more information on different schemes.

Select the Insert button in the Optimization Definition group. MGRID will prompt you to define an Optimization Goal. Please define Start Frequency = 1.88, End Frequency = 1.88, Quantity = Re(S), 1st Parameter = (1, 1), Operator = By Itself. Objective Type: Optimization Quantity = Objective1, Objective1 = 0, and Weight = 1. Basically, it defines Re[S(1,1)] = 0 at 1.88 GHz (see Figure 5.11). Select OK to ad the defined goal into the list.

Step 11 Select Add button again in the Optimization Definition group again. Change the Quantity = Im(S) and select OK to accept other settings. Basically, we define the goal as: Im[S(1,1)] = 0 at 1.88 GHz. Select OK and MGRID puts the 2nd goal into the list. We will get the Optimization Definition dialog as shown in Figure 5.12.

Step 12 Select OK to continue. MGRID will invoke IE3D to perform an optimization for the goals. The optimization dialog is shown in Figure 5.12. It is trying to perform a smart iteration until the goals are achieved.

It takes 100+ simulations before it achieves the convergence. The optimization information can be found from .\ie3d\practice\output\rpatch2.log file. The optimized geometry is saved in .\ie3d\practice\rpatch2m.geo.

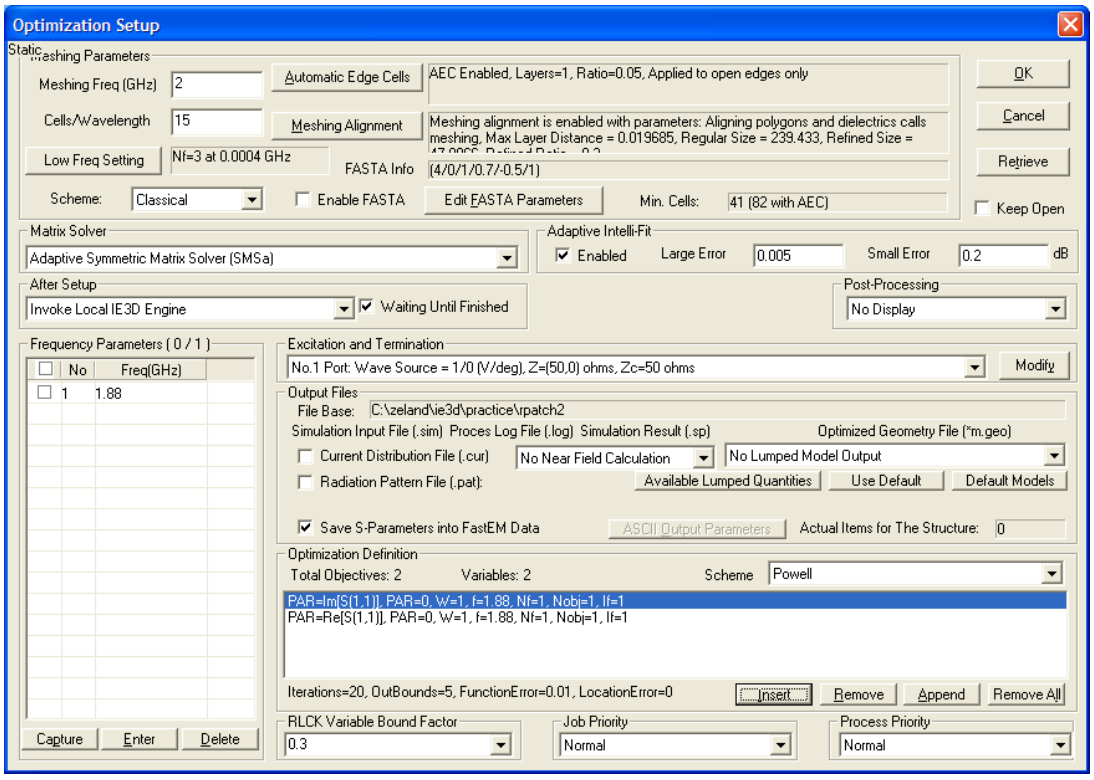


Figure 5.12 The Optimization Definition dialog after the goals are defined.

Step 13 Open .\ie3d\practice\rpatch2m.geo. Select Process->Simulate. Select Enter in Frequency Parameters dialog. Enter Start Freq = 1.7, End Freq = 2.0, Number of Freq = 151. Select Ok to add the frequency points into the list.

Step 14 Scroll the frequency list box until you see the No.91 f = 1.88 GHz. Check the check box for it. Starting from the IE3D 11, we can specify which frequency points we want the IE3D simulator to simulate it even with the AIF enabled. We want to use AIF for the 151 frequency points while we also want to make sure the IE3D engine actually simulates the 1.88 GHz. Please note that the No.1 frequency point in the list is always simulated.

Also, please check the Radiation Pattern File check box. The Current Distribution File check box is automatically checked (see Figure 5.12). Starting from IE3D 11, we can save the .cur file and .pat file even with AIF enabled. IE3D will save the .cur file and .pat file at the frequency points actually simulated. In case MGRID complains about no enough angles for the pattern, please select Modify button for the Excitation and Termination and Pattern Angles dialog. Please select Add Theta button to add 37 angles from 0 to 180 degrees and Add Phi button to add 37 angles from 0 to 360 degrees. We need to have enough angles for accurate pattern data.

Starting from IE3D 11, we will save two radiation pattern related files: Radiation Pattern File (.pat) and General Pattern File (.mpa). The .pat file contains the pattern with the excitation specified and the excitation can't be changed. The .mpa file contains all the pattern data with the specified excitation while the excitation can be redefined with any possible excitation. You can always find the .pat file for a specified excitation combination from the .mpa file on MGRID or PATTERNVIEW in no time whenever necessary. More information about .pat file and .mpa file can be found in Appendix AV. Starting from IE3D 14, we can add an .mpa file into the pattern list for visualization and comparison on MGRID/PATTERNVIEW. They will treat an .mpa file as if it is a .pat file while you can change the excitation of an .mpa file in the list.

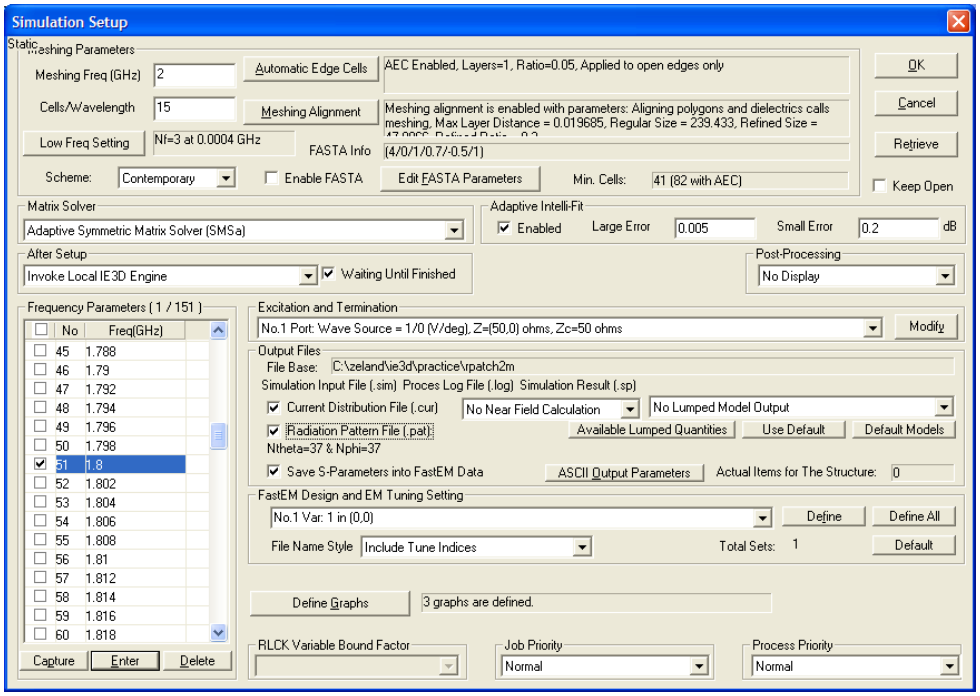


Figure 5.13 The Simulation Setup dialog after Step 14.

Step 15 Select OK to start the simulation for the optimized geometry. It takes seconds to finish it. The three pre-defined graphs are created automatically to display the simulation results similar to what are shown in Figure 5.14. There should not be the legend "Optimized' and the curve for the "Before" is not there. As you can see, the dB[S(1,1)] is below -35 dB and it is very close to perfect match and our goals are achieved.

Step 16 Select Window->Display S-Parameter Graphs->S-Parameters and Lumped Equivalent Circuit to bring out the dialog for s-parameter processing (see Figure 5.15). Change the Short Name to "Optimized". Select Add Files button. Select the file: .\ie3d\practice\output\rpatch1.sp". Change the Short Name to "Before". Double-click at the S-Parameter Display graph to edit it. Select the second s-parameter file in the list in the Display Selection for the Graph dialog. Check dB[S(1,1)]. Select OK to get back the s-parameter processing dialog. Select Close. MGRID will refresh the displays with the s-parameters compared in the S-Parameter Display window with the correct legends (see Figure 5.14).

As it is shown in the comparison, the optimized antenna is perfectly matched at 1.88 GHz and its performance is significantly improved from the original one.

You may wonder what the L and D in the final optimized geometry are. There are multiple ways to measure them. The simplest way is to select Geometry tab in the Information Bar. Click on Z = 31 on the Layer Window to focus the mouse input on the layer. Move the mouse to an edge for the L. MGRID will show the edge's information to you. The finial L should be about 1500.72 mils and the final D should be about 615.65 mils.

We demonstrated how we could optimize the s-parameters of a structure. In fact, you can optimize many parameters such as the Y and Z-parameters, antenna gain, directivity, axial ratio, and efficiency etc. You can even optimize the shape of a pattern. When you try to optimize pattern parameters, you should first enable "Radiation Pattern File" before you define the optimization goals.

For the example demonstrated here, we defined the length and the depth of the notch as optimization variables. We can also vary other dimensions. Parameterizing a structure is much easier on IE3DLibrary. The corresponding IE3DLibrary file with all dimensions parameterized is saved in: rpatch2_tuning2.ie3. Interested users can explore it.

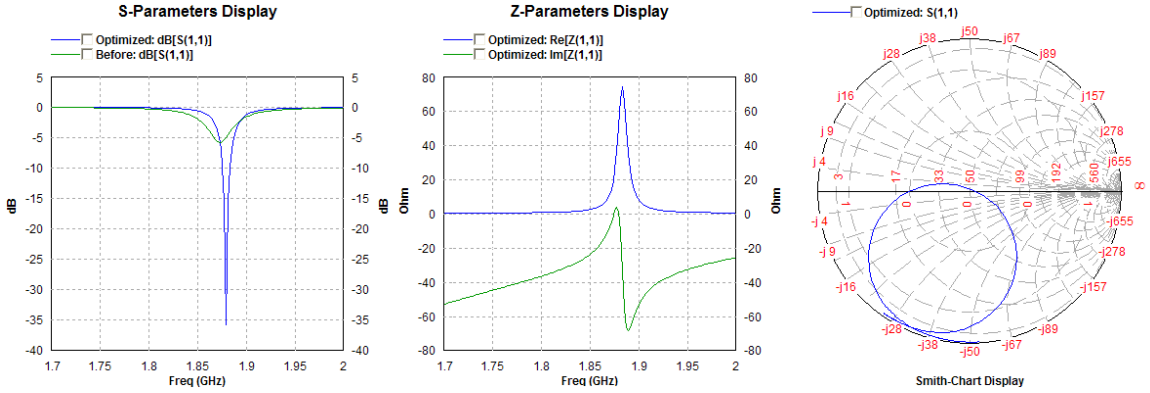


Figure 5.14 Comparison in S-parameters between original and optimized structures.

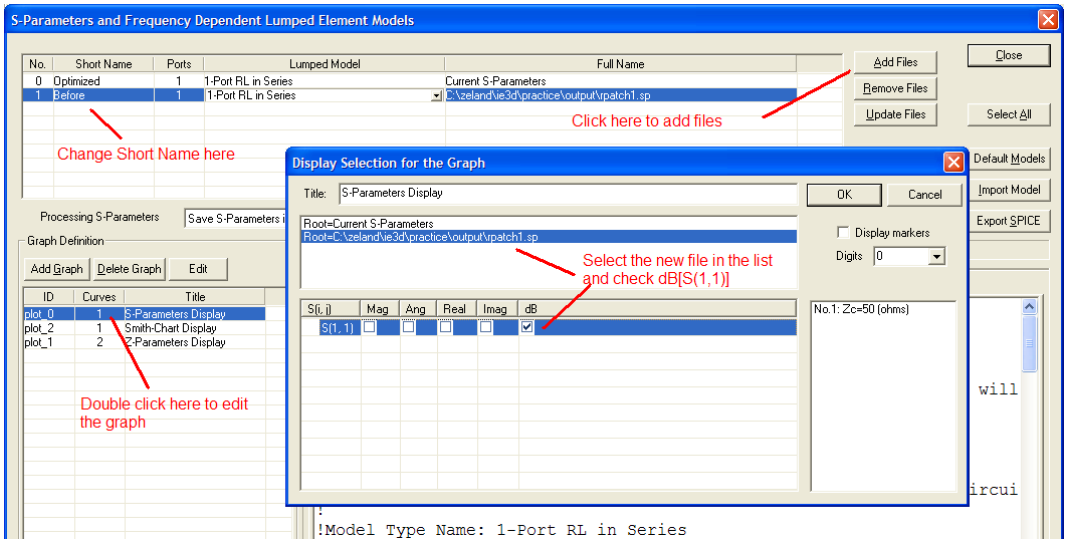


Figure 5.15 Adding more items into the graph for comparison.

Section 5.3 Current Density Distribution Visualization

After an IE3D simulation is finished, the current distribution data is saved into the file: .\ie3d\practice\output\rpatch2m.cur. Current density visualization was used to be performed on MGRID in the post-processing mode. We can still use MGRID to open the .cur file to visualize and post-processing the current distribution. However, it is not necessary for IE3D 14. After an IE3D simulation is done, the s-parameters are saved backed to the .geo file. The saved current distribution, radiation pattern data and near field data are saved into the results files (.cur, .mpa, .pat, .fld etc). However, we will automatically build a link between the .geo and the result files. When we open the .geo file again, we are able to access the current density data in the .cur file and the .mpa files through an internal link..

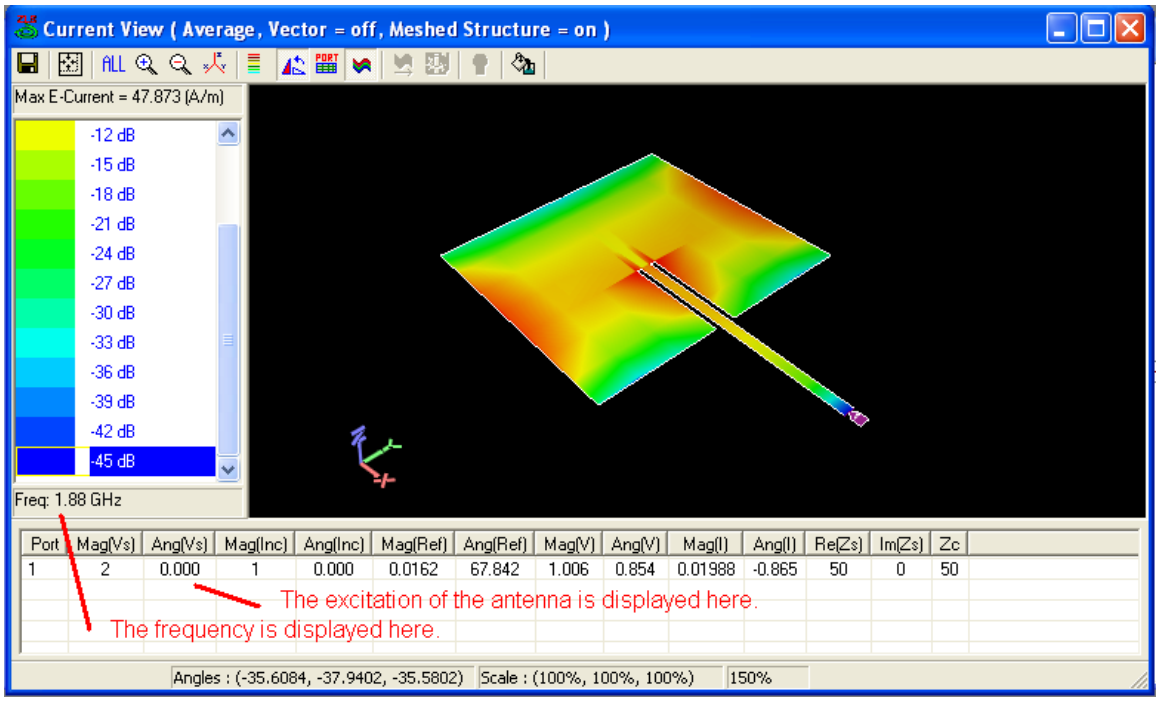
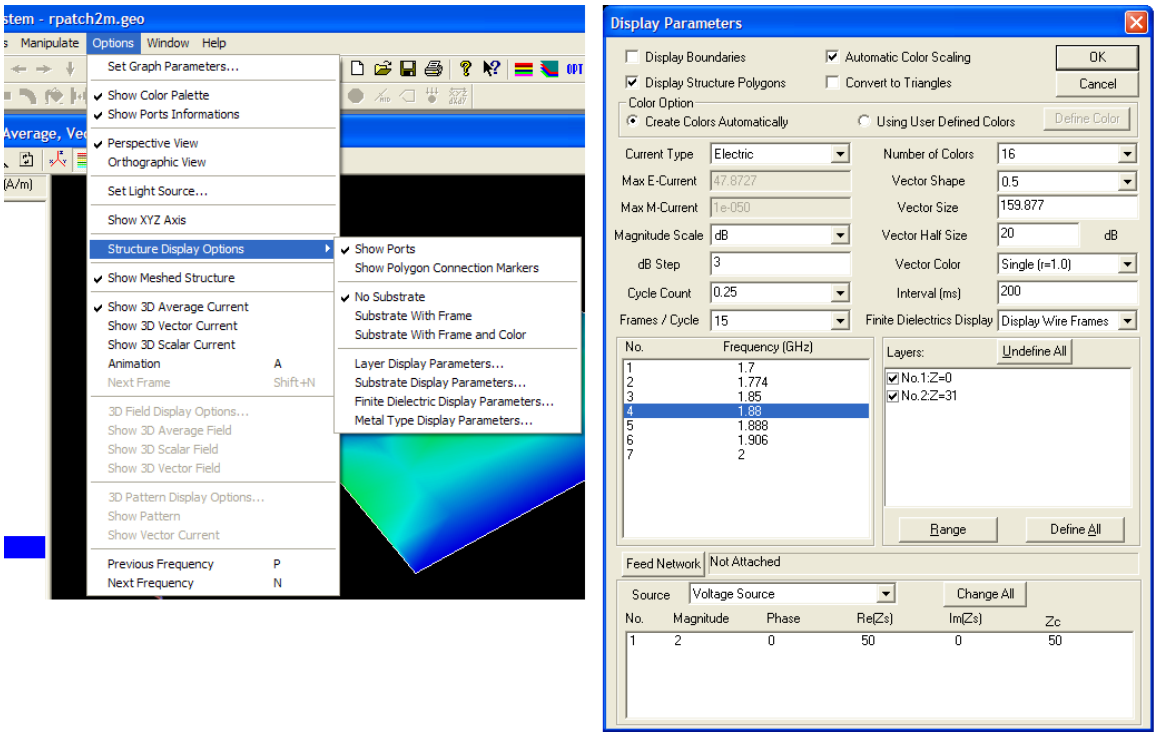


Figure 5.16 The average current distribution display at 1.88 GHz.

Step 1 While MGRID is displaying the geometry and the s-parameter results, please select Windows->3D Current Distribution Display. MGRID will show you the average current distribution similar to what is shown in Figure 5.16 except the frequency is 1.7 GHz which is the 1st one in the list. We have simulated total 151 frequency points from 1.7 to 2.0 GHz and we got accurate s-parameters over the whole band. However, we didn't actually simulate every frequency point but only selected ones. What is shown by default is the average current distribution at the 1st simulated frequency point. We can change the frequency by typing "N" for next frequency point or "P" for previous frequency point.

Please type "P" or "N" a few times to get the frequency point at 1.88 GHz or the resonant frequency. You will see the current is the densest at this point because of resonance.

There are many settings you can configure the current density display in the Options dialog (see Figure 5.17a. For example, you can choose Prospective View or Orthographic View. You can choose whether you want to show the Color Palette on the left or the Port Information at the bottom. You can adjust the Light Source. In the Options->Structure Display Options, you can configure how you want to display the structure. You can show or disable the display of substrates.



(a) The many items in the Options menu (b) The Set Graph Parameters dialog.

Figure 5.17 The Options menu and the Set Graph Parameters dialog.

$$\boldsymbol{J}(x,y,z,t) = [\, Jx(x,y,z)\, \boldsymbol{x} + Jy(x,y,z)\, \boldsymbol{y} + Jz(x,y,z)\, \boldsymbol{z} \,] \exp(\, j\omega t) \qquad (5\text{-}1)$$

Table 5.2 The different current visualization schemes

Scheme	3D Average Current	3D Scalar Current and Animation	3D Vector Current and Animation
Display Quantity	It is showing the $\lvert\boldsymbol{J}(x,y,z,t)\rvert$ or the absolute value of the complex vector current in the form of colors on the structure.	It is showing $\lvert\mathrm{Re}[\boldsymbol{J}(x,y,z,t)]\rvert$ with t defined by Cycle Count in Figure 3.17b in the form of color on the structure Cycle Count = .0.25 means $\omega t = \pi/2$.	It is showing $\mathrm{Re}[\boldsymbol{J}(x,y,z,t)]$ with t defined by Cycle Count in Figure 3.17b. Cycle in the form of vectors.

Step 2 To change the color scale and other settings, please select Options->Set Graph Parameters dialog and you have the dialog shown in Figure 5.17b. The color palette is automatically set by MGRID. If you want to change the corresponding 0 dB value from the default (automatically detected as 47.873 A/m), you can un-check Automatic Color Scaling and change the Max E-Current value, the Magnitude Scale, and the dB Step value.

By default, what is displayed is the average strength of the time-harmonic current density distribution at a specific frequency. The color represents the average strength of the current density at a specific point. For the default continuous tone display, the red color means strong current density. The blue color means weak current density. We can display the 3D Average Current, 3D Scalar Current, 3D Vector Current, and Current density. The current density is described as a complex vector as function of (x, y, z, t) shown in (5-1), where bold and italic characters mean vectors. The differences between the different displays are shown in Table 5.2.

3D Scalar Current and 3D Vector Current are the snap shots of the current at a specific time while 3D Average Current is the average current distribution in a cycle. We can display either 3D Average Current or 3D Scalar Current with the 3D Vector Current together. We can also animate the current distribution either in scalar and/or vector forms. When it is in animation, you can suspend it by selecting Options->Animation again. When it is paused, you can select Options->Next Frame to view the study the current changing frame by frame.

You can control current visualization in Options->Set Graph Parameters dialog (see Figure 5.17b). You can choose the display electric current (E-current) or magnetic current (M-current). You can pick which layers you want to see and which layers you don't want to see. You can change the color scaling on Magnitude Scale, dB Step, and the Color Option. For scalar and vector current display, you can use the Cycle Count to change the time or phase for the snap shot. For animation, you can change the Frame/Cycle and Interval (ms) to control the updating speed. For vector display, you can change the Vector Shape, Vector Size, Vector Half Size, and Vector Color. Due to the fact that current density may change wildly at different location, we have implemented the Vector Half Size to control it. Vector Half Size = 20 dB means the following: If the length of vector A is 50% as big as vector B in the display, vector A is 20 dB smaller than vector B.

Step 3 Select Options->Show 3D Vector Current to show the vector current distribution. Select Options->Set Graph Parameters and adjust the parameters. You will get a similar display as what is shown in Figure 5.18.

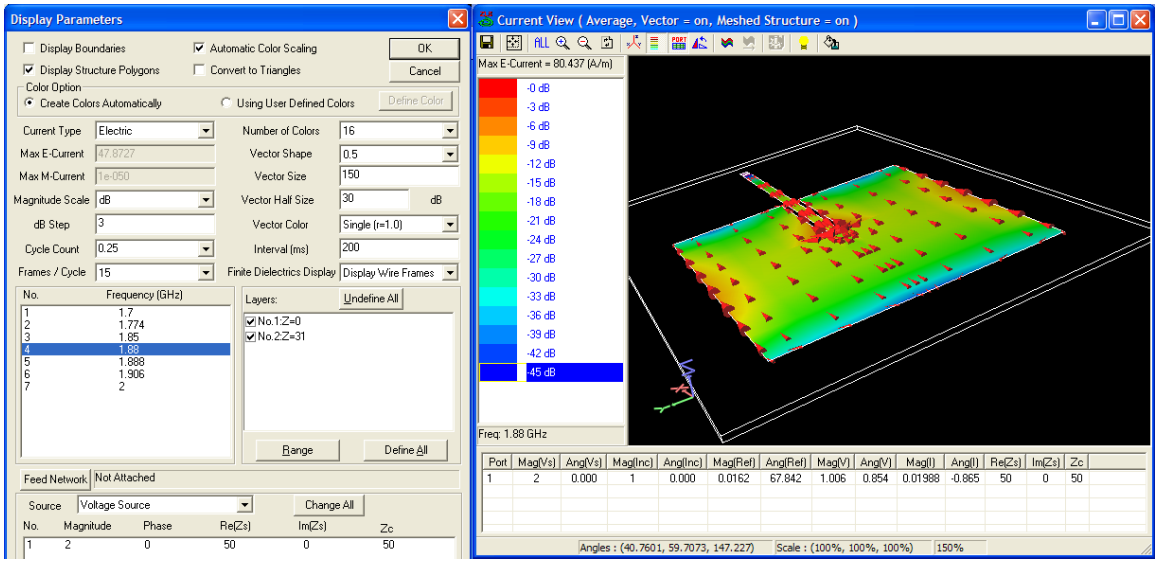


Figure 5.18 Vector current display and its setting.

Step 4 Select Options->Animation. MGRID will start animating the time-harmonic current distribution in real-time. You will see the color representing the current density distribution on the patch changes from blue to red and cycling. On the patch, it is resonating and there is little net power traveling going on. On the feed line, you will see the peak in red traveling into the patch antenna. It is indicating the antenna has reasonably good matching and power is delivered to the patch antenna at 1.88 GHz.

The animation is performed in real-time. For simple structures, you should see very smooth display. However, for complicated structures, updating each frame in animation may take some

time. You may not be able to see the animation very smooth. In such a case, you should try to perform the animation on ZDibAnimator by creating the bitmap files.

Step 5 You can select File->Save to Bitmap Files. MGRID will prompt you for the Bitmap files name prefix. If you just want to save one case, you can leave it blank. Please check Invoke ZDibAnimator for animation. Select OK. MGRID will save a series of bitmap pictures recording the progress of current distribution with time. Then, it will invoke ZDibAnimator to do the animation. You can play an animation on the saved bitmap files anytime later by running the ZDibAnimator from IE3D executables location C:\MentorGraphics\<latest_release>IE3D \SDD_HOME\IE3D\exe\ZDibAnimator.exe.

We have touched base with current visualization. There are many options in current visualization. Please try to explore them.

Section 5.4 Radiation Pattern Visualization

When we check "Radiation Pattern File" in the Simulation Setup dialog, MGRID will invoke IE3D engine to simulate the structure and perform pattern calculation automatically. The pattern data will be saved into the .mpa and .pat files. The data saved into the .pat file is the pattern with the specified excitation. The data saved into the .mpa file is the general pattern. You are able to create the pattern file (.pat) for a specific excitation from a general pattern file.

Traditionally, pattern data is processed and visualized on PATTERNVIEW before version 14. If you have chosen Invoke MODUA in Post-Processing in the Simulation Setup dialog on IE3D 14, PATTERNVIEW will still be invoked to process the radiation pattern data.

On PATTERNVIEW, you are able to add a list of pattern files into the Pattern List window. Then, you can visualize and compare the data. In PATTERNVIEW 14, you can add either a general pattern file (.mpa) or a specific pattern file (.pat) into the list. If it is a specific pattern file (.pat), you will not be able to change the excitation. If it is a general pattern file (.mpa), you are allowed to change the excitation anytime you want.

In fact, all the functionalities of PATTERNVIEW are implemented into MGRID on IE3D 14. You are able to post-process and visualize radiation patterns on MGRID 14 completely. On both MGRID 14 and PATTERNVIEW 14, the primary pattern files are the general pattern files (*.mpa). You can add a number of pattern files into the Pattern List. If a pattern file is a general pattern, you can even change its excitation anytime you want. We may define excitations on ports for a particular structure. It is also possible we define plane wave excitation on the structure. The radiation properties of a structure can be quite different depending upon whether it is excited by ports or by plane waves. When we add a number of files into the Pattern List for comparison, we will need to make sure all of the files are excited by either ports or plane wave simultaneously. In the Pattern List of MGRID 14 and PATTERNVIEW 14, we will list the excitations of the patterns on the Pattern List.

Step 1 While you are opening the geometry file on MGRID, please select Windows->3D Radiation Pattern Display. The 3D Pattern Selection dialog comes up (see Figure 5.19). The rpatch2m.mpa file is automatically added into the list. In case you want to add more pattern files into the list, you can select Pattern List button to add more pattern files into the list. You can also access the pattern's properties inside Pattern List.

Step 2 For this example, we just want to display the 3D pattern, please select the only file in the list. Then, select No.3: 1.88 GHz from the frequency list. You will see the Max(dB) is updated to 4.7903 (see Figure 5.19). This is the maximum dB value of the selected field item. By default,

the selected field item is the dBi(Gain) of the E-total field. It means the maximum E-total field for dBi(Gain) is 4.7903 dB. Select OK. The 3D radiation pattern is displayed (see Figure 5.20).

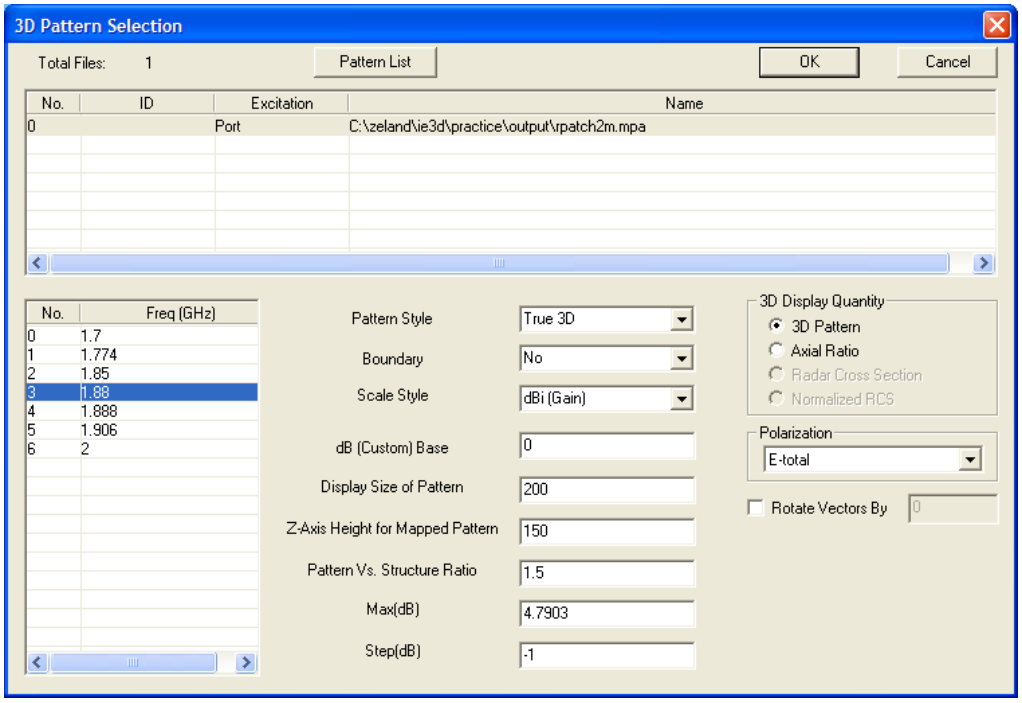


Figure 5.19 The 3D Pattern Selection dialog.

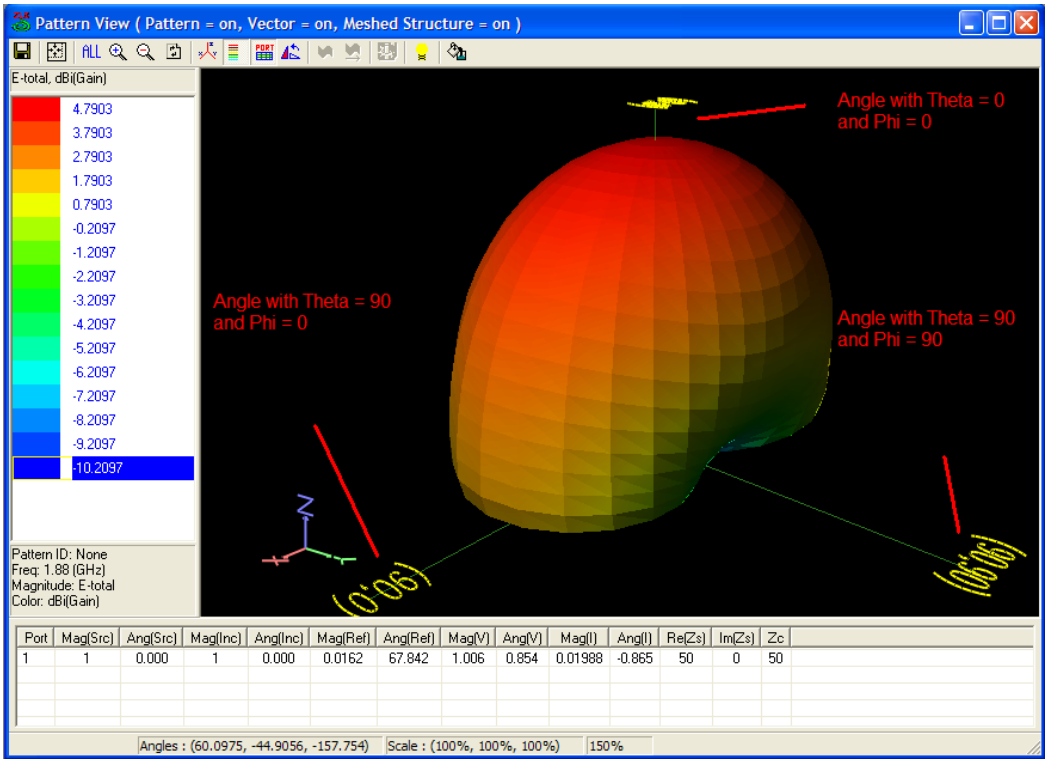


Figure 5.20 The 3D radiation pattern.

On the 3D pattern display, there are 3 axes. Those are the axis for the three main angles: (Theta, Phi) = (0, 0), (90, 0) and (90, 90).

There are many parameters you can access in the 3D pattern display (see Figure 5.19). You can choose the 3D Pattern Quantity as: 3D Pattern, Axial Ratio, Radar Cross Section (RCS) and Normalized RCS. The RCS and normalized RCS are for plane-wave excitation only. Normalized RCS is defined by the ratio of RCS and the square of wavelength. The display in Figure 5.20 is for the 3D pattern case. In case you choose Axial Ratio, the display quantity will be the axial ratio in 3D form.

For the Polarization, it is E-total. You can choose E-theta for the theta-field, E-phi for the phi-field, E-left for left-hand circularly polarized field, and E-right for right-hand circularly polarized field. The definitions are documented in Table AV.2 in Appendix AV. User can also rotate the theta and phi vectors on the (theta, phi) plane by an angle (see Figure 5.19). For such a case, the E-theta and E-phi are no longer the original theta and phi fields, but the theta and phi-fields after the rotation.

For the Scale Style, the default is dBi (Gain). It means we are displaying the gain of the antenna in dBi. dBi means 0 dB reference is the based upon an isotropic antenna pattern. You can choose dBi (Directivity) for displaying directivity in dBi, dB(Custom) for displaying the pattern with use specified dB(Custom) Base, Linear scale with the maximum at 1, Phase for the phase of the field, and dBi (Conj. Match Gain) for the gain based upon conjugate match. The standard gain definition is based upon the port impedance. For RFID applications, designers are not interested in the gain with matching to the port impedance. They are more interested in the antenna gain and efficiency based upon conjugate match. For this reason, we have defined Conjugate Match Efficiency and Conjugate Match Gain.

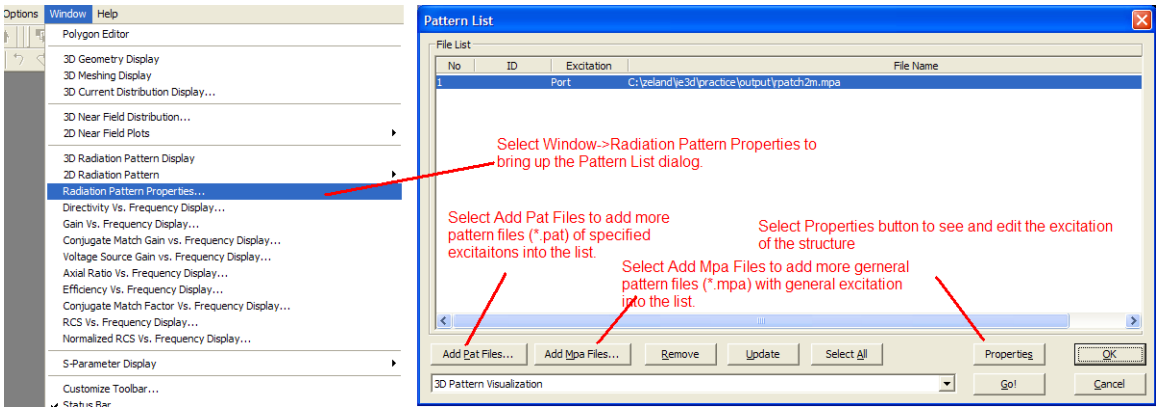
You may notice we do not use the terminologies of E-plane and H-plane patterns and Linear Gain. You may want to know how you can get those parameters. The answer is that we do not provide those parameters. The reason is due to the fact that IE3D is a general EM simulator. You have the complete freedom to create any antenna structure you like and simulate it. IE3D does not know what the E-plane and H-plane of the antenna are because the planes are defined by the designers. We cannot provide such information. Instead, we provide the information about Total Field, Theta-Field, Phi-Field, LH Circular Field and RH Circular Field. If your antenna is linear polarized antenna, you can consider the Total Field or the Theta Field pattern as your primary radiation pattern while the Phi Field pattern as the cross-polarization. When it is circular polarized antenna, you should use either LH Circular Field or RH Circular Field as your primary polarization and the other one as the cross-polarization. Please note the following fact: For a circular polarized antenna, the Total Field does not represent the linearly polarized field. The pattern data for E-total is assuming all the fields (both E-theta and E-phi) are received completely.

There are many items in the Window menu of MGRID (see Figure 5.21a), allowing you to visualize different parameters of your structure. You can display the 2D patterns of the pattern on specified phi and/or theta angles. You can display the plot with gain vs. frequency etc.

Step 2 Select Window->Radiation Pattern Properties. MGRID will show you the Pattern List (see Figure 5.21b). You can add more files into the list for display and comparison purposes.

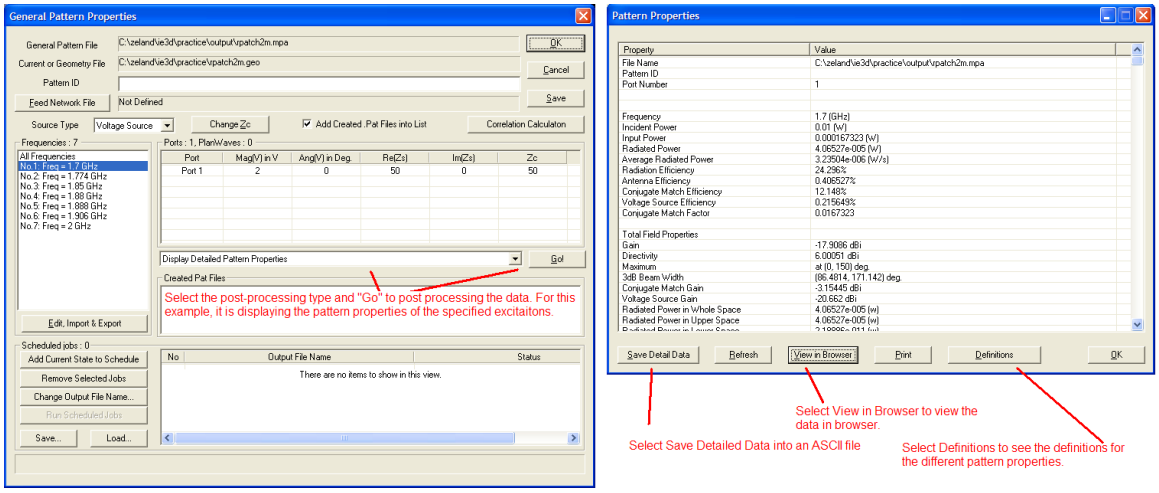
Step 3 Select the only pattern file in the list. Select the Properties dialog. The General Pattern Properties dialog will come up (see Figure 5.22a). In the dialog, you are able to change the excitations and terminations of the structure for each frequency point and post-process the

pattern. In case the selected pattern file in the Pattern List dialog is a pattern with specified excitation (*.pat), you will see the specified pattern properties shown in Figure 5.22b when you select the Properties dialog in the Pattern List.



(a) (b)

Figure 5.21 The Pattern List window.



(a) General Pattern Properties dialog. (b) Pattern Properties for specified excitations.

Figure 5.22The pattern properties dialogs for general pattern and pattern with specified excitations.

Step 4 Select "Display Detailed Pattern Properties" in the combo box (see Figure 5.22a). Select GO button. MGRID will show you the detailed pattern properties of the antenna in Pattern Properties dialog (see Figure 5.22b). Select View in Browser. MGRID will save the pattern properties data into an XML file and open a browser for it. There are many pattern parameters available in the dialog. The meanings of the parameters are documented in the Definitions dialog (see Figure 5.22b). They are also explained in the Appendix AV. We will not provide details here. The selected parameter parameters at selected frequency points are shown in Table 5.3.

The Input Power, Radiated Power, Radiation Efficiency, Directivity, etc. are for general antenna applications. The Antenna Efficiency and Gain have included the mis-match loss to a wave source. They are normally for microwave antenna applications. RFID are widely used nowadays. For an RFID application, the source impedance is normally a voltage source with complex source impedance. For such applications, we should use the Conjugate Match Factor (CMF) and

Radiation Efficiency to judge how good the RFID antenna or system is. CMF with maximum value at 1 is a criterion on how good the antenna impedance and the source impedance conjugate matching. The CMF(dB) = 20 log10(CMF) is the measure of how much loss in the delivered power to the antenna due to impedance mis-match between the source and the antenna impedances. Please read Appendix BI for more discussion on RFID designs and CMF.

Table 5.3 The pattern parameters of the optimized antenna (rpatch2m.geo) at selected frequency points.

Freq (GHz)	1.7	1.88	2.00
Incident Power (mW)	10.0	10.0	10.0
Input Power (mW)	0.167323	9.99737	0.270496
Radiated Power (mW)	0.0406527	7.19081	0.00623073
Radiation Efficiency	24.296%	71.9269%	23.0344%
Antenna Efficiency	0..406527%	71.9081%	0.623073%
Conjugate Match Efficiency	12.148%	35.9635%	11.5172%
Voltage Source Efficiency	0.215649%	36.1751%	0.197646%
Conjugate Match Factor	0.0167323	0.999737	0.0270496
Total Gain (dBi)	-17.9086	4.7903	-15.3823
Total Directivity (dBi)	6.00051	6.22253	6.67236
Conjugate Match Gain (dBi)	-3.15445	1.78114	-2.8065
Theta Gain (dBi)	-17.9086	4.7903	-15.3823
Theta Directivity (dBi)	6.00051	6.22252	6.67233
LH Circular Gain (dBi)	-20.9113	1.77843	-18.3568
LH Circular Directivity (dBi)	2.99776	3.21165	3.69786

Step 5 Select OK twice until you get back to Pattern List dialog.

Step 6 Select “3D Pattern Visualization” at the bottom of Pattern List dialog and select “Go!”. MGRID will show you the 3D Pattern Selection dialog (Figure 5.19). Select the No.3 Freq = 1.88 GHz. Select Pattern Style as “True 3D”. Select OK. MGRID will open a window and show you the “True 3D” pattern (see Figure 5.20 and Figure BH.3a).

Step 7 You can select Windows->3D Radiation Pattern Display to access the dialog. Repeat Step 6 while you select the Pattern Style as “Mapped 3D”. MGRID will show you the “Mapped 3D” pattern (see Figure BH.3b).

Step 8 You can either select Windows->2D Radiation Pattern->Define 2D Pattern Plot or select Windows->Radiation Pattern Properties->2D Pattern Visualization->Go to get into Define 2D Pattern Plots dialog (see Figure 5.23).

Step 9 Select Add Plot to bring up the 2D Pattern Display dialog (see Figure 5.24). You can scroll the list box to pick up any combination of theta and phi angles at any frequency. In case you have multiple files, you can also pick the items belonging to different pattern files. For this example, scroll to f = 1.88 GHz. Check the “E-Total at Phi = 0” and “E-Total at Phi = 90” (see Figure 5.24). The two cuts at Phi = 0 and Phi = 90 are the two major cut planes for the pattern. Select Pattern Style as “Polar Plot”. Select OK to finish defining the plot and it is added into the list of plots.

Step 10 Select Add Plot again. Select the same items “E-Total at Phi =0” and “E-Total at Phi=90” at f = 1.88 GHz. However, please choose the Pattern Style as “Cartesian Plot”. Select OK to add the plot into the list.

Step 11 Select Continue and the two defined plots will be displayed on MGRID. We will have four windows showing the pattern in the forms of "True 3D" , "Mapped 3D", "Polar Plot" and "Cartesian Plot" (see Figure BH.3). They are representing the same data but in different forms to you. For the differences between the four forms, please read Appendix BH.

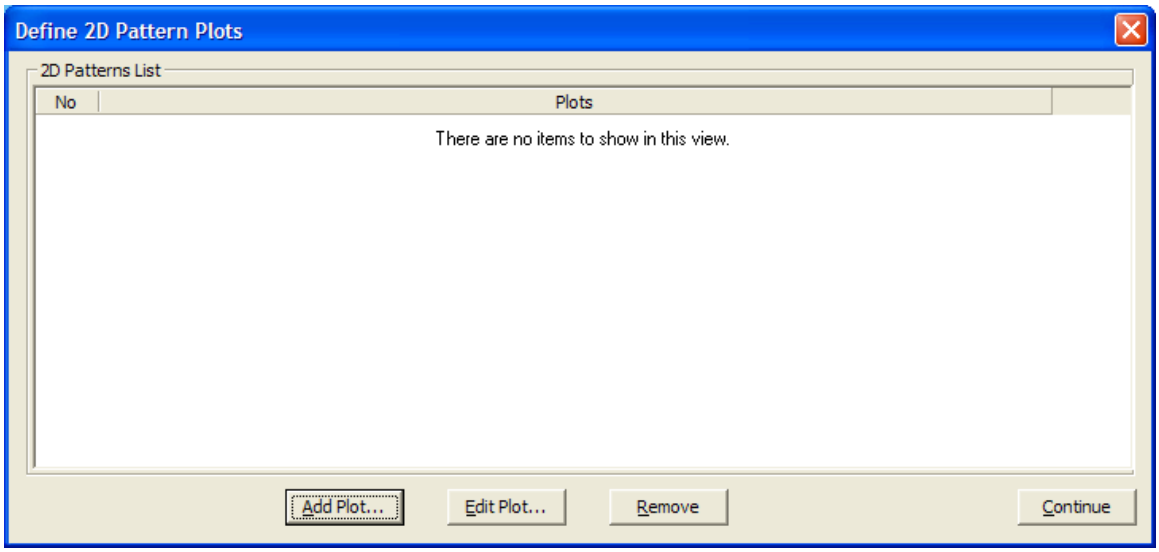


Figure 5.23 The Define 2D Pattern Plots dialog.

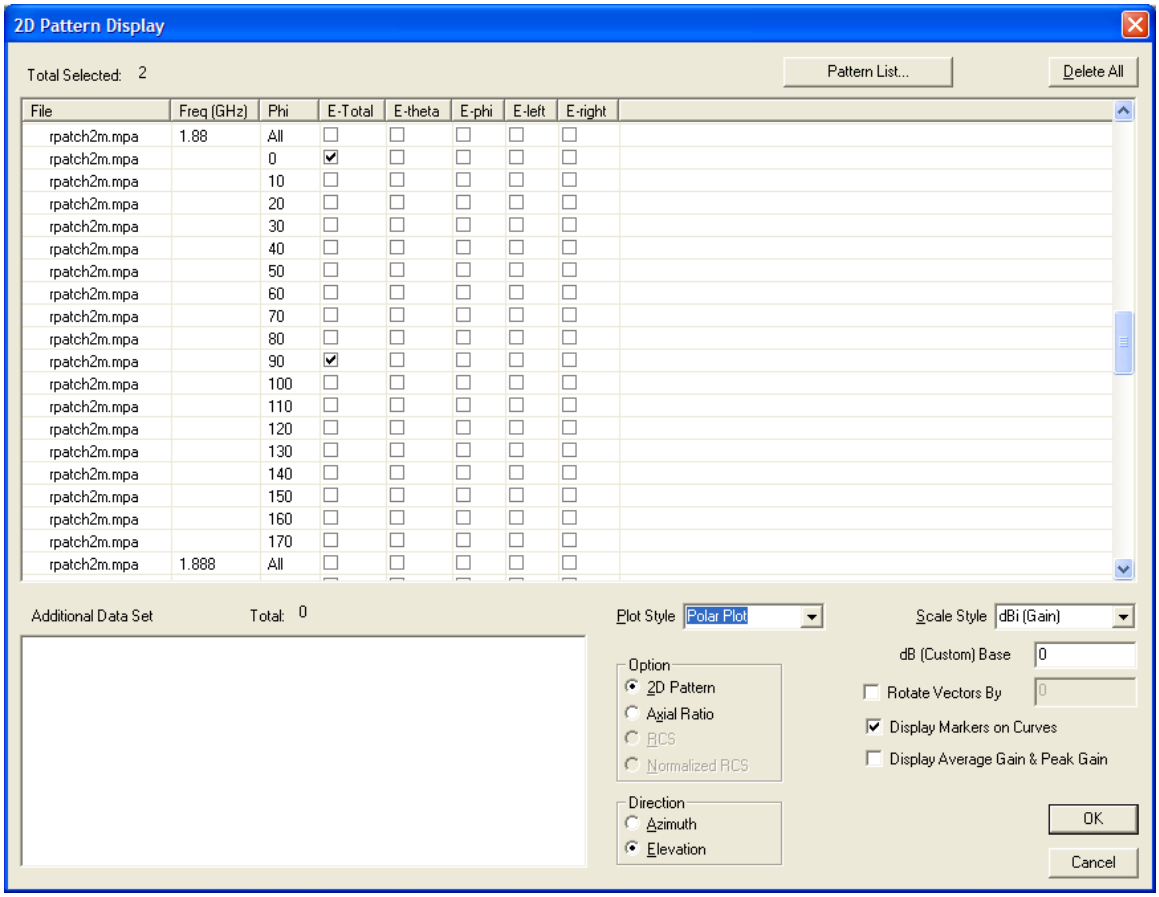


Figure 5.24 The 2D Pattern Display dialog.

Step 12 Select Windows->Gain Vs. Frequency Display. MGRID will prompt you the Frequency Respond Display dialog. Check the item: Phi = 0 and Theta(1) with Theta = 0 degree (see Figure 5.25). Select OK. MGRID will open a window to display the Gain Vs. Frequency with gain at Theta = 0 and Phi = 0 (see Figure 5.25). There are many other visualization capabilities. We will not explain all of them here. Please check the Windows menu.

Again, pattern visualization and processing used to be on PATTERNVIEW. For the convenience of users, we have transferred all the functionalities into MGRID. You can do the same procedures on PATTERNVIEW. You only need the general pattern files (.mpa) and/or the specific pattern files (.pat) on PATTERNVIEW.

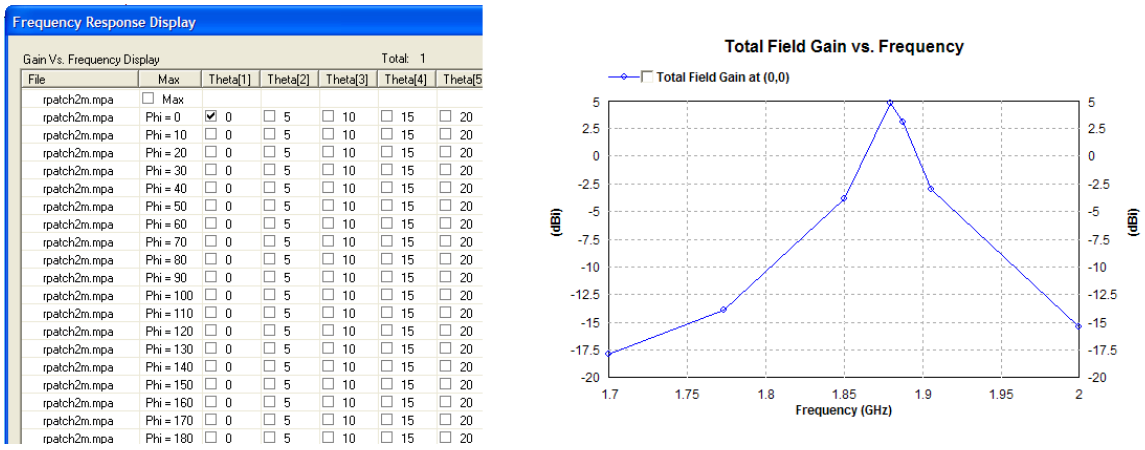


Figure 5.25 The Total Field Gain Vs. Frequency plot.

Section 5.5 Pattern Calculation in Post Processing

When you check Radiation Pattern File (.pat) button in Simulation Setup dialog, IE3D engine will save the current distribution and perform radiation pattern calculation for you automatically. In case you only save the current distribution and you did not check Radiation Pattern File in the Simulation Setup dialog, IE3D will save the current distribution data for you. It will not perform pattern calculation for you. However, you can still do pattern calculation when the current distribution data is available.

Step 1 While MGRID is still opening the file ".\ie3d\practice\rpatch2m.geo", please select Process->Pattern Calculation. MGRID will warn you that General Radiation Pattern (.mpa) is already available.

Normally, there is no need to perform the pattern calculation again because you are able to find the pattern at any excitations and terminations as you like from the .mpa file. However, there are cases you may want to do the pattern calculation again. For example, the .mpa and .pat files are lost while the .cur file is still available. We want to have finer angles for the theta and phi for better accuracy. In our example, we just want to show you that you are still able to perform pattern calculation again from the saved current distribution data.

Step 2 Select Yes to continue. The Pattern Calculation Information dialog comes up (see Figure 5.26). You can add more theta and phi angles for more accurate patterns. You can also change the default excitations and terminations for the general pattern.

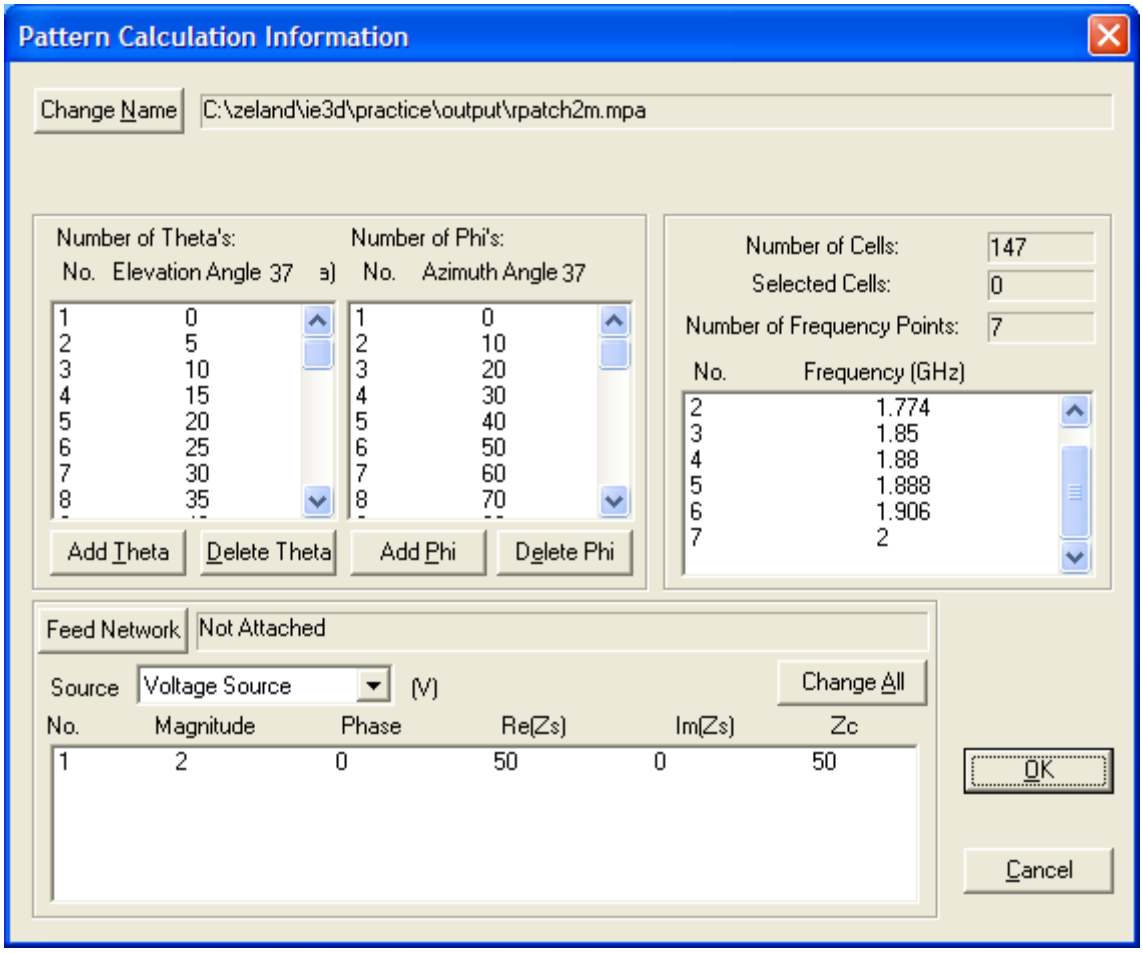


Figure 5.26 The Pattern Calculation Information dialog.

Step 3 Select OK to start the pattern calculation. For this simple structure, it takes seconds to finish the pattern calculation. After pattern calculation, Pattern List dialog comes up with the generated .mpa file updated in the list. You can continue to perform visualization and processing of the pattern data.

Section 5.6 Adding Array Factors to Radiation Patterns

To improve the directivity and gain of an antenna or change the pattern shapes, antenna designers put multiple antenna elements with appropriate spacing and excitation. Normally, there will be mutual coupling between the adjacent elements. We should simulate the coupled elements simultaneously in order to get accurate modeling of the array in case the mutual coupling between elements is strong. However, the simulation time will increase rapidly with more and more elements. We will discuss simulations of antenna arrays with mutual coupling included later. In this section, we will discuss a simple way to predict the pattern of an antenna array using the antenna array factor. When the spacing between elements is large enough, the mutual coupling between elements may not be very critical. In such a case, we can use array factor to predict the pattern of an array with reasonable accuracy.

Step 1 On MGRID, select Window->Radiation Pattern Properties to open the Pattern List dialog.

Step 2 Select Find Array Radiation Pattern in the combo box at the bottom of the dialog. Select the "Go!" button on the right of the combo box. The Antenna Array Parameters dialog comes up. Select the Add button (see Figure 5.27). The Add Elements dialog comes up. For X-Location, enter the From = 0, To = 12400 mils. For the Y-Location, enter the From = 0, To = 12400 mils. Please note that 3100 is half a wavelength in air at 1.9 GHz. We are adding a 5 by 5 antenna array with spacing of half a wavelength. We will get the dialog shown in Figure 5.26. Select OK. The Antenna Array Parameters dialog is resumed. We will see total 25 elements are added into the list.

We created a uniform 5 by 5 antenna array. In fact, we can define the Phase Step for the added elements. We can enter a set of array elements in the 3D space in each Add command. We can add more to them. For each Add command, we can designate which Element Index it is using. For our current situation, we just use one element in the list: the rpatch2m.pat. In case we have multiple elements in the pattern list window, we can choose different elements for its Element Index (starting from 0). We can mix-use different types of elements for the antenna array.

We can mix-use different types of elements for the array, for each Add command and also define the Rotation angles of all the elements to be added. For the Rotations, we need to enter the 1st Phi, 1st Theta and 2nd Phi. What do they mean? The angle or matrix transformations are like that: (1) We keep the z-axis unchanged and we rotate the φ angle for the 1st Phi; (2) Then, we keep the x-axis unchanged and we rotate the θ angle for the 1st Theta; (3) Finally, we keep the z-axis and we rotate the φ angle for the 2nd Phi.

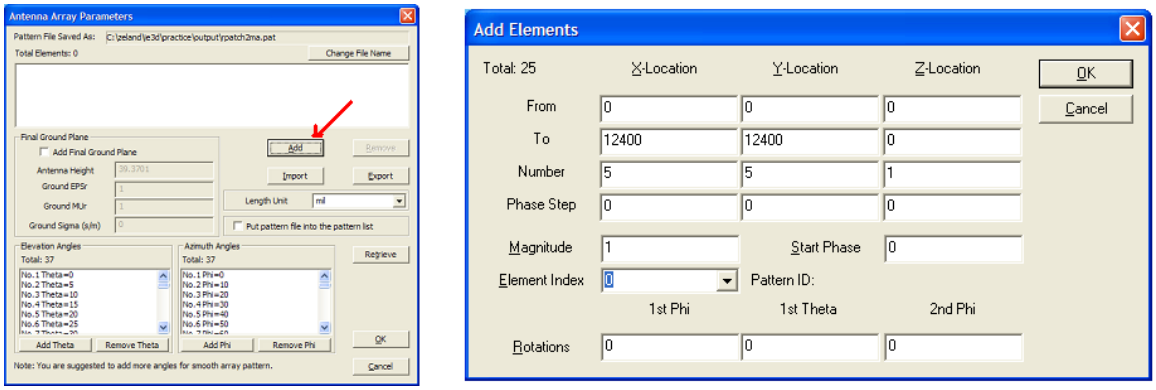


Figure 5.27 The dialogs for defining array factor.

Step 3 Select Export button and enter the file name as: rpatch2m.arr. This step is not necessary. We are saving the elements into the array file so that we can import it for other arrays of similar dimensions but different elements.

Step 4 Select Add Theta. The Add Elevation Angles dialog comes up. Enter Start = 0, End = 180, Number = 73. Select OK. The theta angles are added into the list. For more elements, we need to add more angles for smoother curves in the pattern.

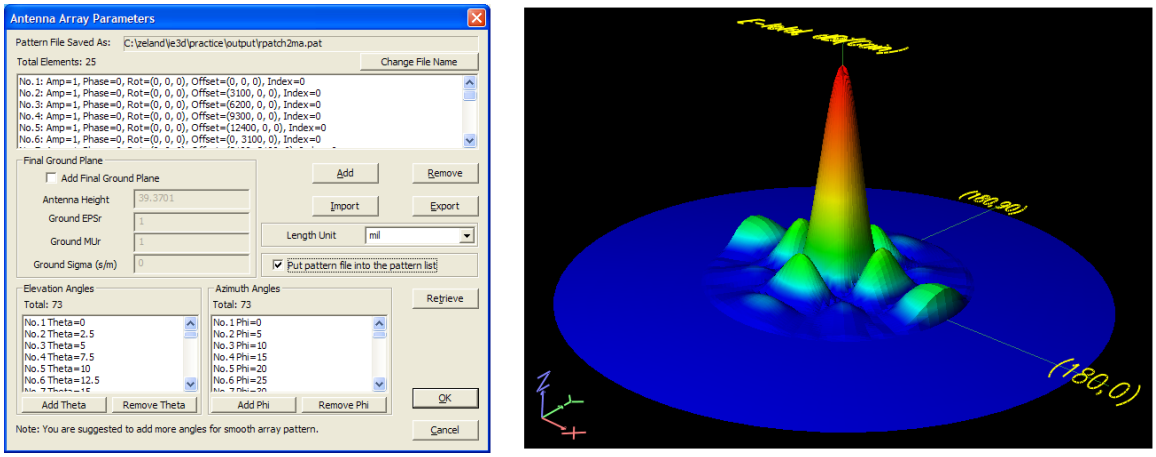
Step 5 Select Add Phi. The Add Azimuth Angles dialog comes up. Enter Star t = 0, End = 360, Number = 73. Select OK. The phi angles are added into the list.

Step 6 Check the Put Pattern into the Pattern List. We will get the picture shown in Figure 5.28a. Select OK to continue.

On the MGRID in processing the patterns, it is allowed to include the effect of the Final Ground Plane (see Figure 5.28a). This is for the case we want to include the effects of the real ground plane. When we design an antenna, we always have a ground plane that is the reference of the patch. However, this ground plane may not be the final ground. For example, if we use an antenna on a base station. The final ground plane is the earth. If we want to know the effect of the earth, we can add it as the Final Ground Plane. In this example, we do not need it.

MGRID will start the calculation of the array pattern based upon the element pattern and the array factor. After it finishes the calculation, it will save the result of the array pattern into: .\ie3d\practice\output\rpatch2ma.pat. Then, it will put it into the Pattern List window. It is the No.1 pattern file in the list.

Step 7 Select "3D Pattern Visualization" in the list box at the bottom. Select "Go!". The 3D Pattern Selection dialog comes up. Select the No.1 pattern file, or the one just created with array factor, in the list. Select No.3 Frequency = 1.88 GHz. Select Pattern Style as "Mapped 3D". Select OK. The 3D Mapped pattern is displayed in Figure 5.28. How to interpret 3D mapped pattern? The (180, 0) means the point at the edge is Theta = 180 and Phi = 0. From the mid-point of the radius to the full radius, the color is blue meaning very low field value in the region. The mid-point of the radius is where the Theta = 90 angle corresponds. It means that the field is low in theta from 90 to 180 degrees. In fact, due to the infinite ground plane, the field is 0 for the angle range. The (180, 90) means that the point at the edge is Theta = 180and Phi = 90.



(a) The settings for array (b) The 3D mapped pattern for the array.

Figure 5.28 The dialog for the array factor and the 3D mapped pattern.

Section 5.7 Pattern Merging, Rotation, Wave Propagation Prediction and Tx/Rx Transfer Functions

A few more post processing features in the Pattern List of MGRID (or the Edit menu of PATTERNVIEW) may be very useful to you. They allow you to rotation a pattern, add the final ground plane's effect to a pattern, and merge multiple patterns of individual structures together to find the pattern with all the elements in one single structure (mutual coupling neglected). It also allows you to predict the field propagation and distribution in the far field zone. The Wave Propagation prediction uses the far field pattern to predict the field density at a set of specified x, y and z-locations. Such a feature will be good for the wireless applications such as planning of base station locations. Another new feature introduced in the IE3D 10.2 is the Tx/Rx transfer function. Basically, it allows you to find the transfer function between a Tx

pattern and an Rx antenna based upon radiation patterns of both antennas. The transfer function can be imported into our s-parameter SPICE simulator MDSPICE for finding the waveform received in a reception antenna. Such a feature should be very useful for UWB applications. This feature is discussed more in the Appendix. We will not show more examples here. Interested users can try to explore the features on their own.

Section 5.8 Geometry Tuning

We have demonstrated EM optimization capability of the IE3D in antenna design. The optimized structure achieves the designed performance. It is suggested that users should try to define the variables very carefully so that the geometry will be kept in good shape when the variables change. Users should also try to define the variables in reasonable ranges so that it will not become invalid. On IE3D 11 and earlier versions, users can use the command Optim->Display Trends to display the rough bounds and trends of the variables. On IE3D 12, we have implemented the geometry-tuning feature. It allows the users to examine how the geometry is tuned by the variables before any EM tuning and optimization.

Step 1 Run MGRID and open the optimized geometry file .\ie3d\practice\rpatch2m.geo file. The original optimization variables are still defined on the structure while the default shape is the optimized shape.

Step 2 Select Optim->Geometry Tuning. The IE3D Geometry Tuning dialog comes up. The left side is the top view of the structure. The color may be different due to the difference in some default setting. The right hand side has 2 tabs. One tab allows you to change the variables values in a form format. The other tab allows you to slide the bars to change the variables continuously.

Step 3 Select the Variables Sliders tab. Slide the bars and you will see how the variables are controlling the shape of the structure (see Figure 5.29)

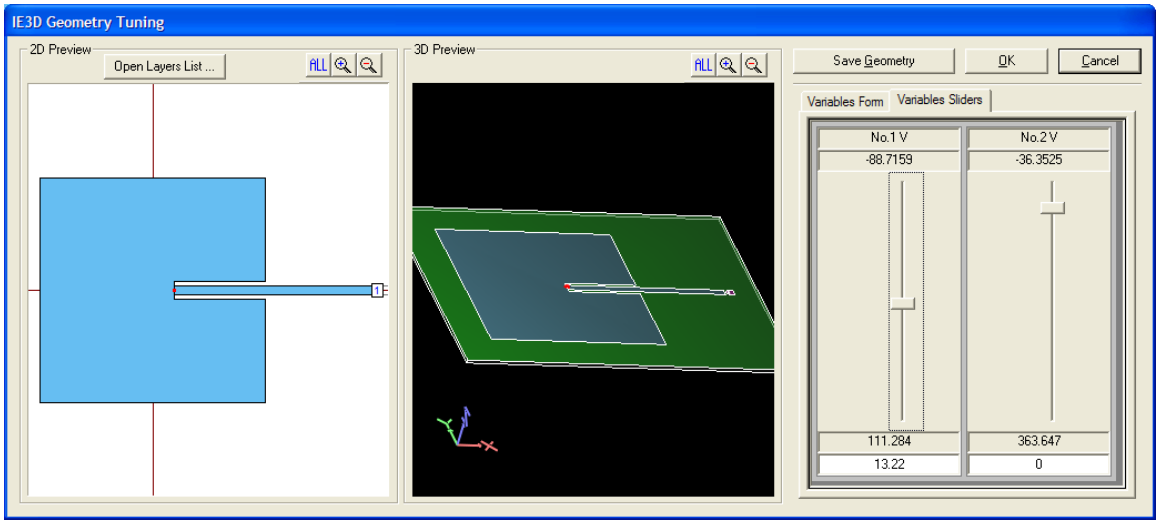


Figure 5.29 The IE3D Geometry Tuning dialog.

Step 4 After you finishing testing the sliders, select CANCEL to exit the dialog. If you select CANCEL, MGRID will recover the original offset values of the variables on exit. If you select OK, it will set the variables to the current positions of the bars. Since this structure is optimized and simulation results are available, we would like to keep the state. Certainly, as long as you don't save the revised geometry, the results are always accessible from the original file.

The geometry-tuning feature certainly provides much convenience to you. You can use it to check the ranges of variables in the structure before you do any simulation and optimization. In case there is any problem, you can fix it before you do simulation and optimization, which normally take longer time.

You may wish that you would be able to see the results in real-time while you are sliding the bars. Many people may consider real-time full-wave EM tuning may exist in dreams. Full-wave EM simulations are very accurate. However, due to the numerical nature, full-wave EM simulations are relatively slow for large structures. Normally, it can't yield real time results and designers can't do a real-time EM tuning. However, let's make one step back. Can we do a real-time full-wave EM tuning and optimization at design time? The answer is yes with the IE3D FastEM Design Kit .

Section 5.9 FastEM Design Kit and Real-Time EM Tuning, Optimization and Synthesis

FastEM Design Kit consists of three portions; (1) Parameterization; (2) FastEM data preparation phase; (3) FastEM tuning, optimization and synthesis.

We need to prepare the FastEM data. Defining the optimization variables or parameterize the structure is the 1st step. The second step is to perform EM simulations on the parameterized structure. IE3D can automatically extract the signature from the EM simulations on the parameterized structure.

Step 1 Open the geometry file .\ie3d\practice\rpatch2m.geo file. Save it into rpatch2m1.geo.

Step 2 Select Process->Simulate. Simulation Setup dialog comes up. Select Delete button in Frequency Parameters section to delete all the frequency points in the list. Select Enter button in the section to enter the frequency range from 1.83 to 1.93 for 101 frequency points. Please select Define All button (see Figure 5.30) to define multiple offset values for each optimization variable. MGRID will prompt you for the Tuning Range for each variable (see Figure 5.30).

We have two variables. For the 1st variable that controls the size of the patch, we define the range from –15 to 15 with 11 points. For the 2nd variable, we define the range from –10 to 10 with 11 points. The actual ranges of the tuning variables are much larger. However, you may consider defining appropriate ranges for them because too wide a range may require more points. We need to define appropriate number of points so that the simulations will not be too long while the extracted FastEM signature is accurate enough. For this particular example, it is an optimized structure; we just limited the ranges of the variables so that we can demonstrate this advanced feature without taking too much waiting time.

Step 3 Select OK in the Simulation Setup dialog. MGRID will prompt you for loss of existing s-parameters due to change. Select YES to continue. MGRID comes up with the dialog summarizing the FastEM Design Kit Data generation preparation (see Figure 5.31).

We have two variables. There are 11 points for each variable. Totally, we have 11 by 11 or 121 combinations. By default, IE3D is required to simulate minimum 4 combinations and maximum 100 combinations. You can change the min and max combinations you want to simulate. The residual limit is for the criteria of curve-fitting. If the residual is below the specified limit after the N-th simulation, it will automatically stop the simulation and save the signature. Please select OK to continue.

MGRID/IE3D starts the EM simulations and FastEM data preparation process. There are two windows come up (see Figure 5.32). The upper one is for all the combinations. The lower one comes up and disappears many times. It is for each simulation. For this particular structure, each

simulation takes couple seconds on a modern PC, it will not take much time and you can wait until it finishes. In real cases, this process may take a long time, it is suggested that you should understand that a FastEM data preparation may take many minutes, some hours or tens of hours. For example, you can leave it to run over night or over the weekends.

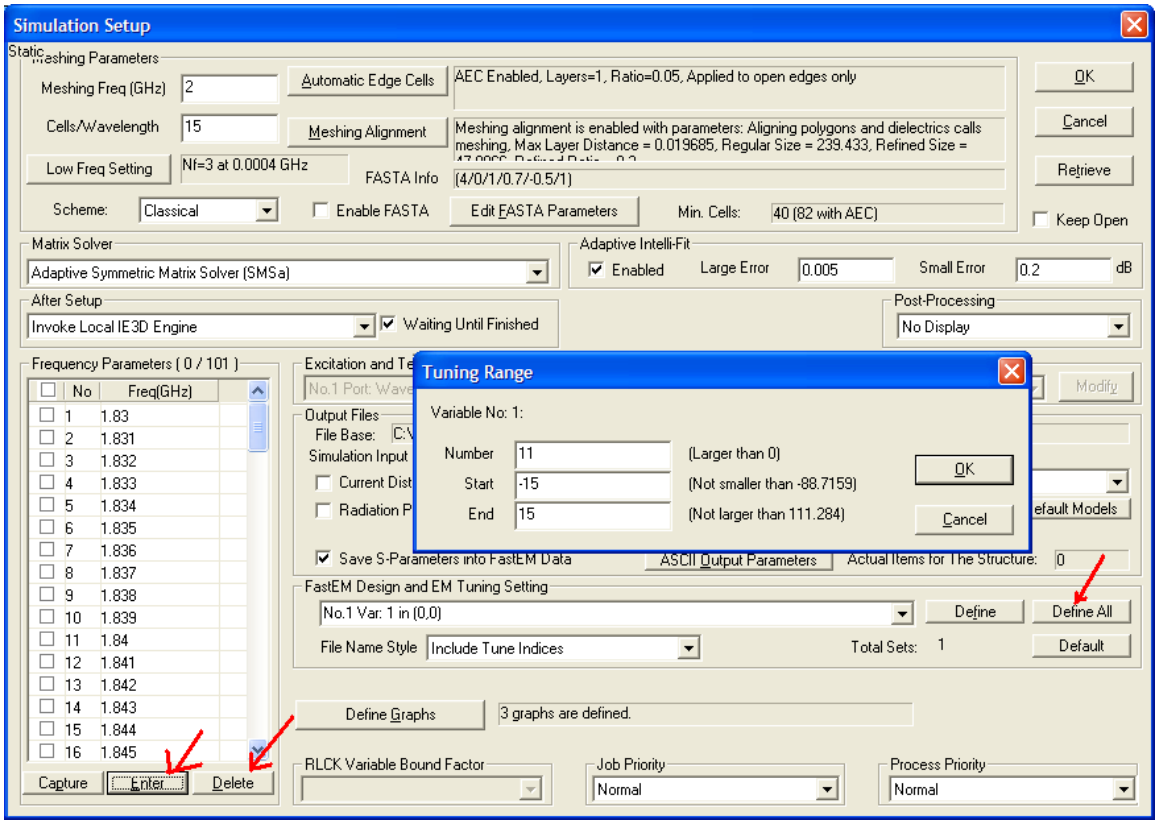


Figure 5.30 Defining the Tuning Range for each variable in Simulation Setup dialog.

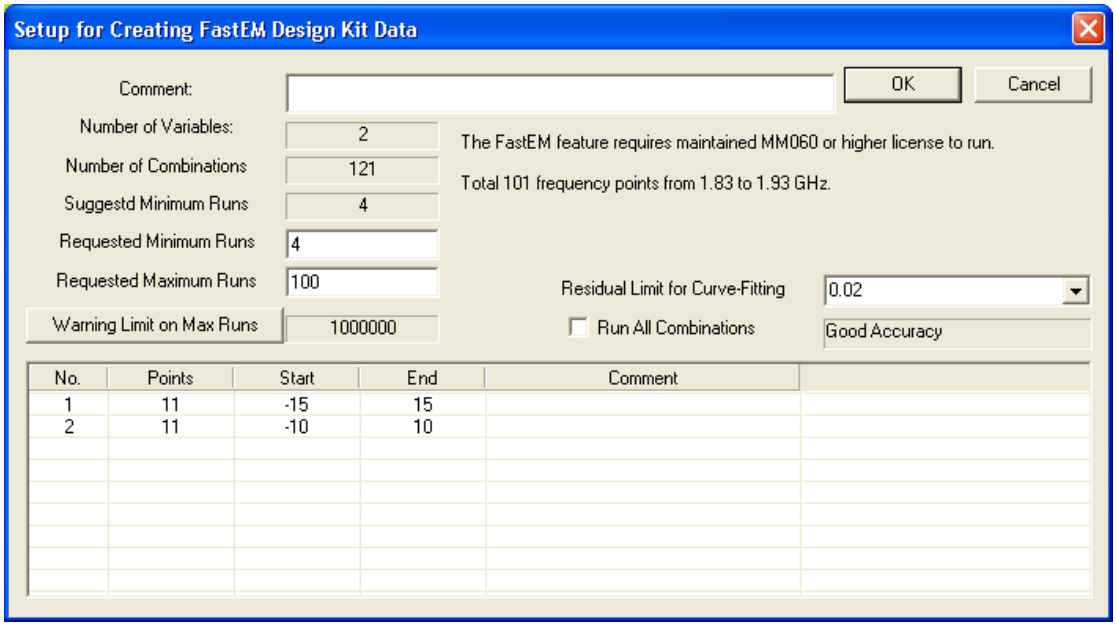


Figure 5.31 The dialog gives you a summary on the preparation of FastEM data.

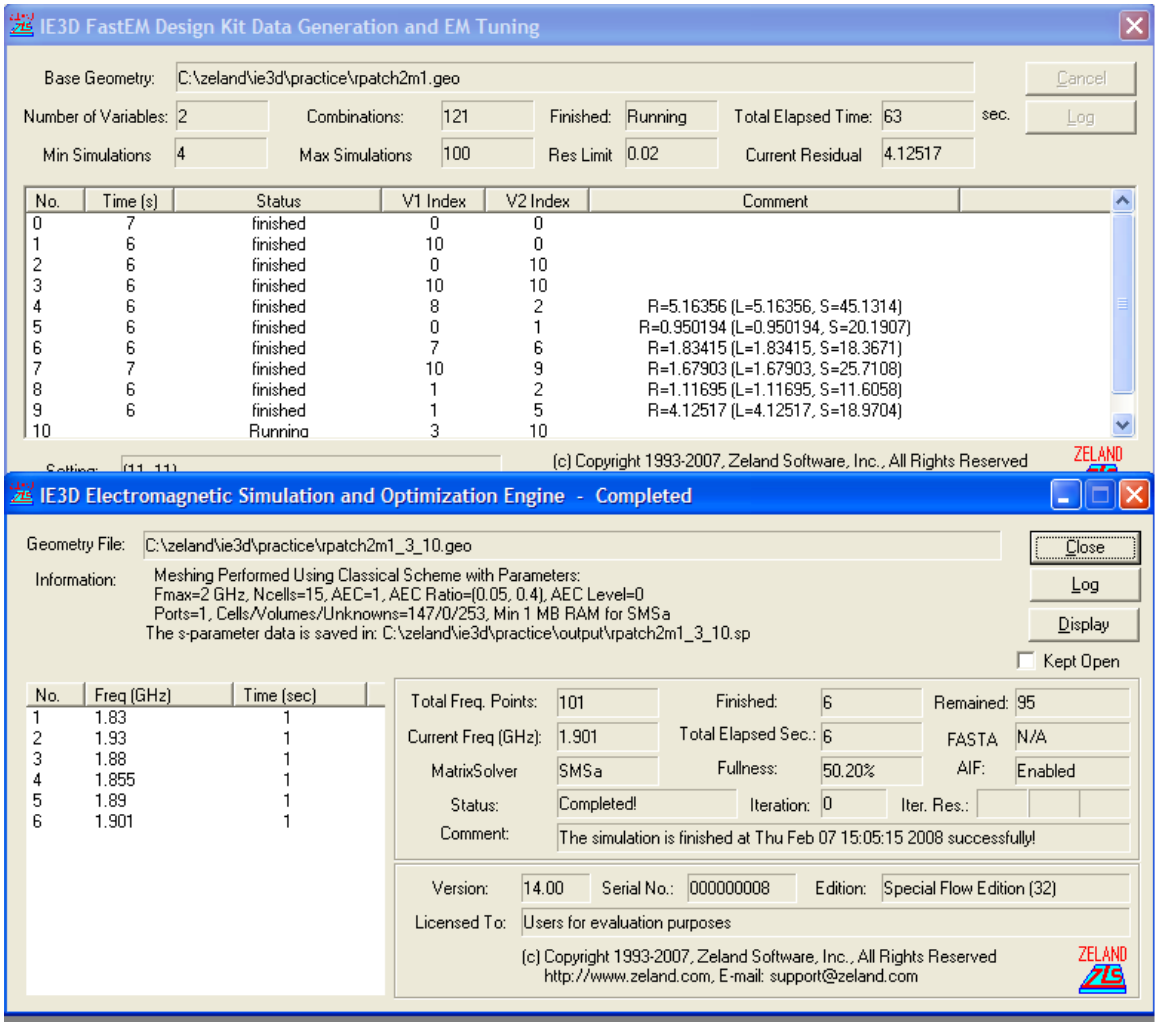


Figure 5.32 The IE3D windows during FastEM data preparation process.

Step 4 IE3D finishes the preparation in a few minutes. It shows the dialog in Figure 5.33. It ran the maximum 100 combinations. The final residual is 0.051041 and it does not meet the requirement of 0.02. It is not easy to meet the default criteria. It is OK because the extracted data will be accurate over most of the regions in the swept domain. Certainly, you can select Run All Combinations to let IE3D run all the cases (see Figure 5.31).

By this time, the extracted FastEM data is already saved into the geometry file: .\ie3d\practice\rpatch2m.geo. The IE3D Full-Wave FastEM Design Kit dialog comes up automatically (see Figure 5.33). In case you close open the file with the FastEM data saved on MGRID, you can select Process->Full-Wave EM Design Using FastEM Design Kit command to get the same dialog shown in Figure 5.33. You don't need to do the preparation again. You can go ahead to do the tuning and optimization directly on the extracted data.

The FastEM Design Kit dialog is similar to the Geometry Tuning dialog. However, there are many more things. First, you can create multiple FastEM data sets into one file. You can select which set of data

you want. For this particular example, we have only 1-set and we are selecting the No.0 data set by default. In case you want to delete one or multiple sets of FastEM data from the file, you should also go to this dialog and delete them by selecting the appropriate buttons.

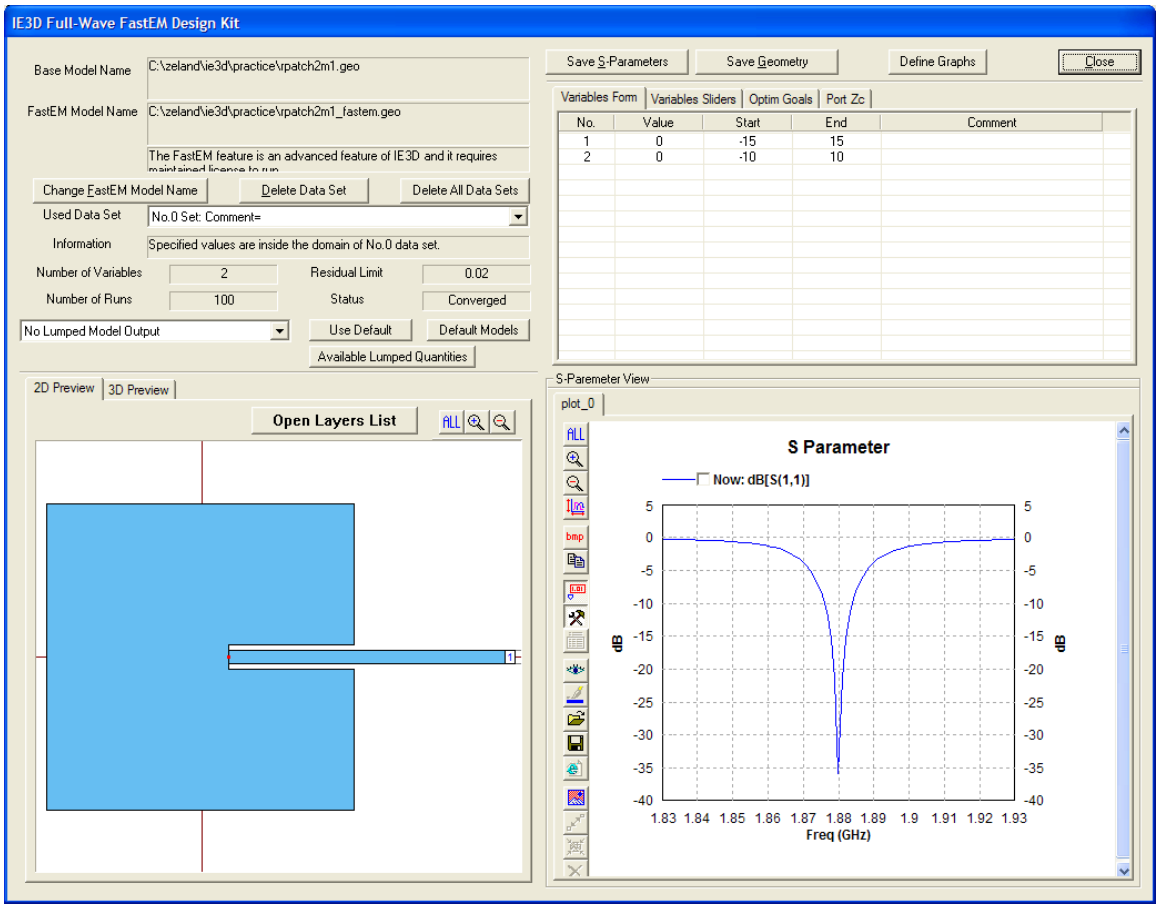


Figure 5.33 The FastEM Design Kit dialog.

You can choose to create a frequency dependent lumped element model. If you want to optimize and display the circuit parameters such as the series R, the series L or the Q-factor, you should choose a lumped model for it.

To define or change the graphs, you can select Define Graphs button.

We can select the Variables Sliders and slide the bars to tune the structure and see the geometry updated in the layout view and the s-parameters updated in the s-parameters view. You will get the results in real time. If you want to save the s-parameters into a file, you can select Save S-Parameters button. If you want to save the current geometry and s-parameter data into files, you can select Save Geometry and S-Parameters to save it. MGRID will create a new geometry file: rpatch2m1_fastem.geo file and a new s-parameters file: rpatch2m_fastem.sp for you. Please understand that the rpatch2m1_fastem.sp may not be the actually IE3D simulated data. It is based upon some extracted data from IE3D simulations. In most of the cases, it is almost identical to truly IE3D simulated results. However, it is not guaranteed. After you get an optimum design from FastEM, you are suggested to run the IE3D simulation again to make sure the results are as expected.

By default, IE3D s-parameters are always normalized to 50-ohms. However, you can visualize s-parameters normalized to other than 50-ohms. You can change the Zc in the Define Graphs button. In fact, you can display s-parameters normalized to different Zc values using different graphs.

We have optimized the structure to achieve perfect match at 1.88 GHz. We know it works perfectly at 1.88 GHz and further tuning may not help to further improve the results. Let's change our goals to demonstrate the real-time EM tuning and optimization using FastEM. We would like to achieve perfect match at 1.89 GHz by adjusting the variables. We will define our goals as:

Lumped Element R value, R(1,1) = 50 ohms at 1.89 GHz.
Lumped Element L value, L(1,1) = 0 nH at 1.89 GHz.
(L in nH and C in pF are assumed on FastEM tuning).

The goals are equivalent to the Re[S(1,1)] = Im[S(1,1)] = 0 at 1.89 GHz. We just try to demonstrate different optimization capabilities of IE3D by defining the goals alternatively.

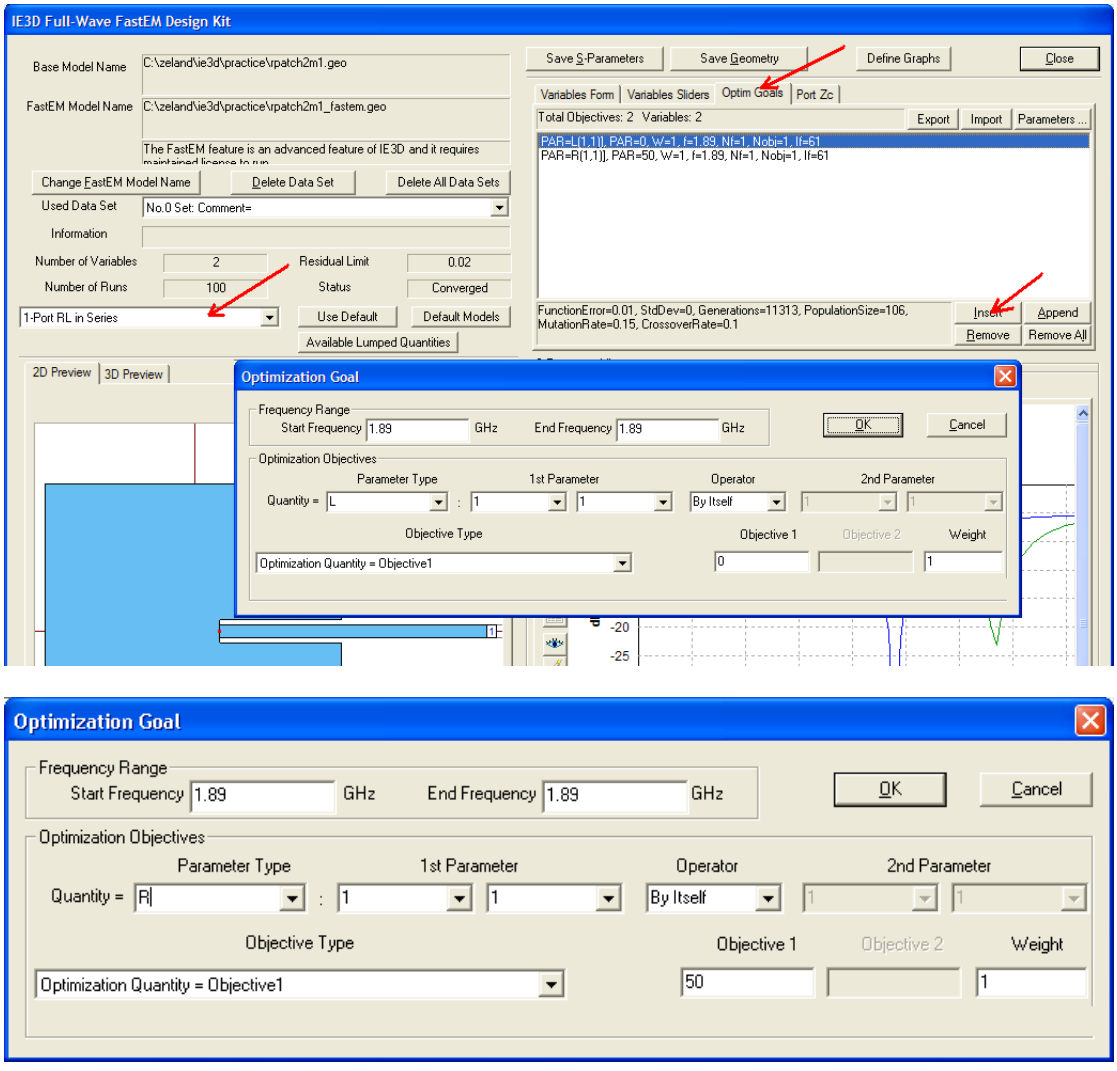


Figure 5.35 The dialog for the 1st goal of R = 50 at 1.89 GHz.

Step 5 Please change the combo box “No Lumped Model Output” to “1-Port Series R and L” (see Figure 5.35). Please select the Optim Goals tab. Select the Insert button in the tab to define the 1st goal as: R(1,1) = 50-ohms at 1.89 GHz (see Figure 5.35). Please select the Insert button again and define the 2nd goal as: L(1,1) = 0 nH at 1.89 GHz (see Figure .5.35).

After you define the goals, please select the Variables Sliders tab. Please slide the bars for the variables V1 and V2 (see Figure 5.36). You will see the structure is updated in the 2D View window and the s-parameters are updated in the s-parameters display window in real-time. This is the manual real-time EM tuning. You can tune the geometry and the corresponding s-parameters in real-time. The FastEM Design Kit also shows you the Current Error and Best Error with respect to goals you defined. As it is shown in Figure 5.36, the Best Error is 0.1364. It does not match the convergence criteria of 0.01. However, you can see the dB[S(1,1)] is quite good at 1.89 GHz. The current best result is quite good. Can we do even better? For sure we can.

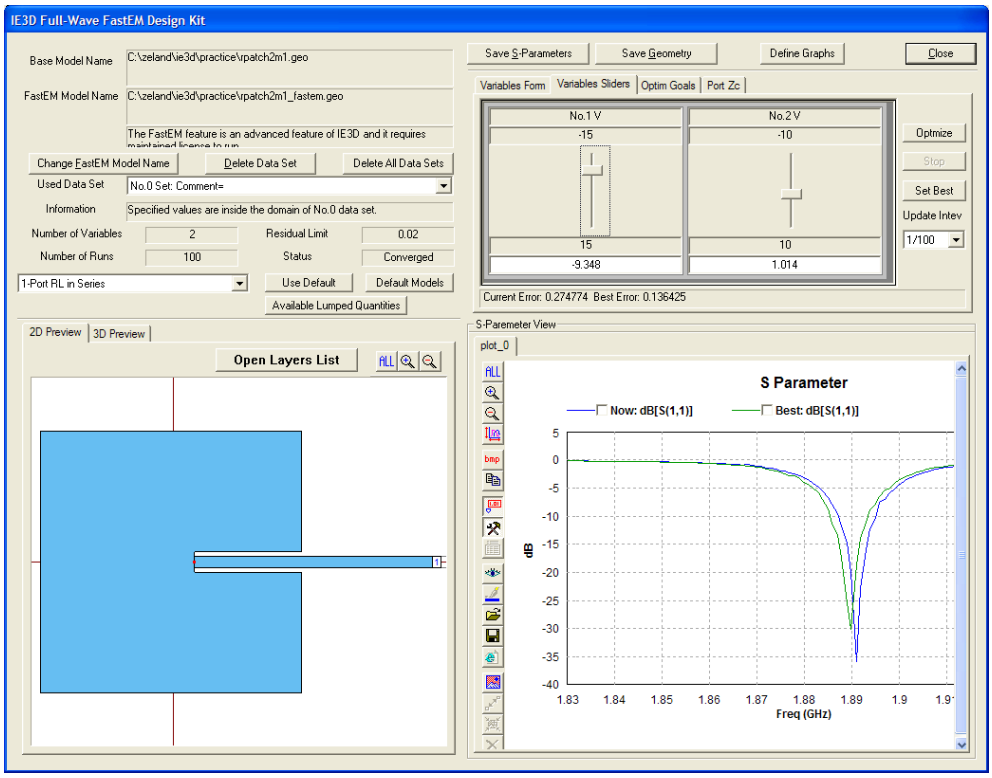


Figure 5.36 The results of real-time FastEM tuning by sliding the bars.

Step 6 Select the Optimize button in the Variables Sliders tab. FastEM Design Kit starts the automatic optimization. You can see the slider bars jumping up and down, the geometry changing and the s-parameters updating. You can see the current s-parameters and the best s-parameters displayed simultaneously. From time to time, the Best Error drops to a smaller value. After some seconds, FastEM Design Kit finishes the optimization with 11,301 iterations. The Best Error is about 0.02 and it is not as low as 0.01 or the limit. However, you can see dB[S(1,1)] = -40 at 1.89 GHz (Figure 5.37). It certainly is a very good result.

FastEM prompts you the optimization is completed (see Figure 5.37). You can select the Set to Best button in the dialog to get the best results. If you select Continue without Setting to Best, it will get back to the FastEM Design Kit dialog with the current state without updated. You can still select Set to Best in the FastEM Design Kit dialog to set to the best if you like.

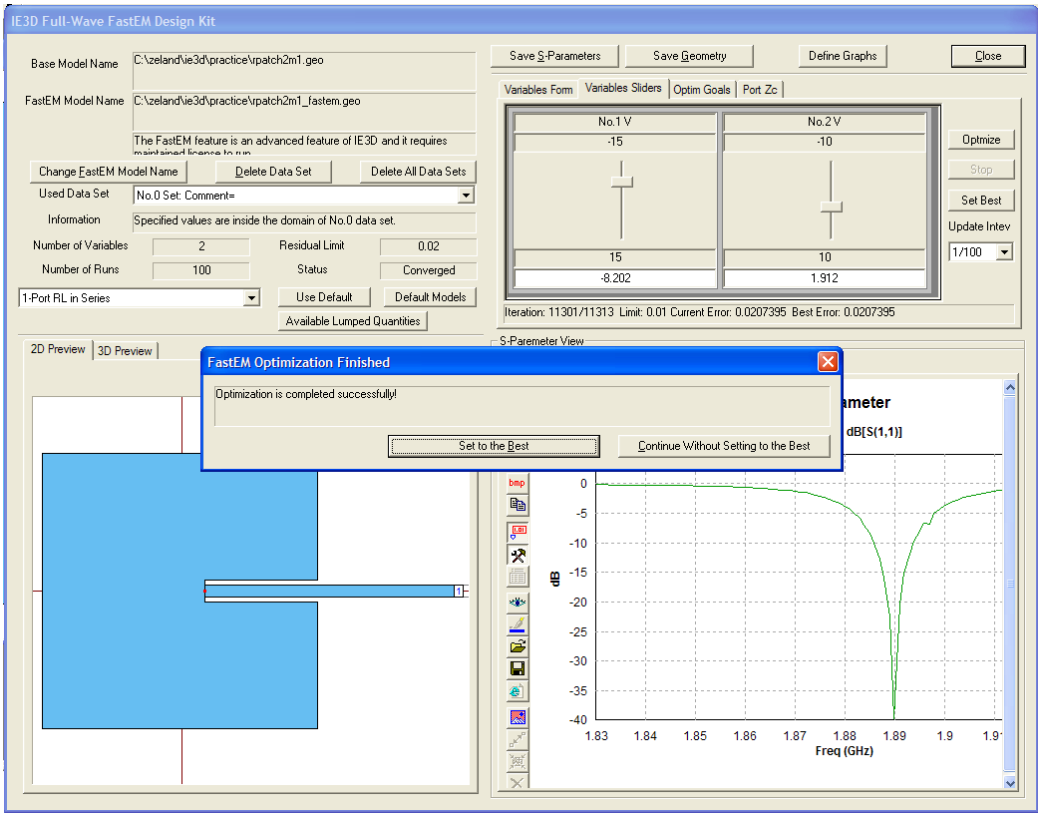


Figure 5.37 The state when FastEM finished the automatic optimization.

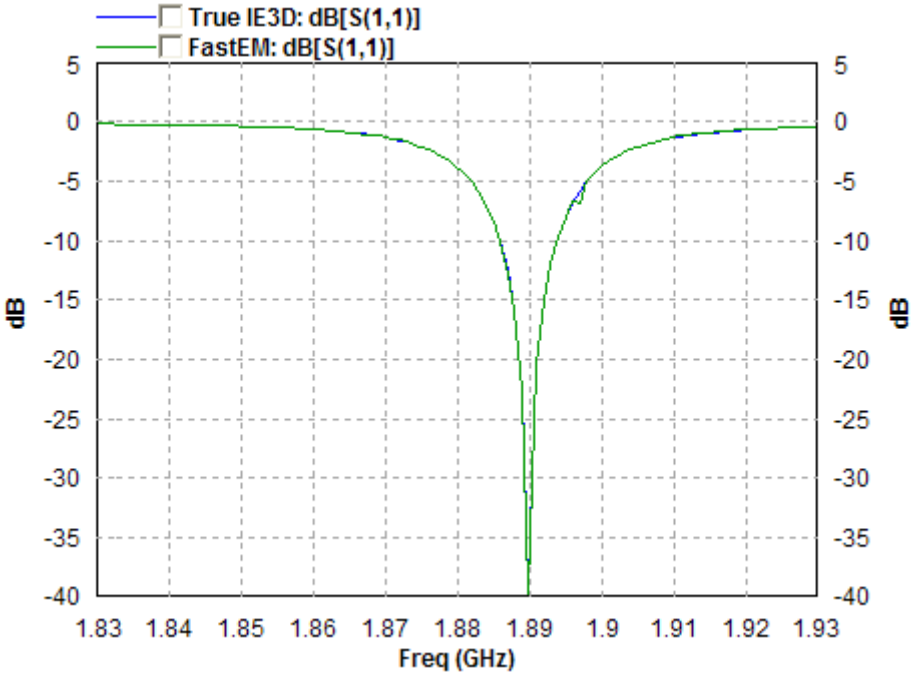


Figure 5.38 The comparison of FastEM results and true IE3D results.

Step 7 Please select Save S-Parameters button to save the FastEM results into rpatch2m1_fastm.sp. Save Geometry and S-Parameters save the FastEM optimized geometry into: rpatch2m1_fastem.geo. Another MGRID is invoked to display file: rpatch2m1_fastem.geo.

Step 8 You can save the rpatch2m1_fastem.geo on the 2nd MGRID into another name and compare its s-parameters to the FastEM result rpatch2m1_fastem.sp file created by the FastEM Design Kit. Renaming is necessary. Otherwise, the simulation results will overwrite the rpatch2m1_fastm.sp file created by FastEM. You can compare the FastEM results with a truly IE3D simulated results for this structure (see Figure 5.38). You can see that they are almost identical. This is normally the case for most of the FastEM results we tested.

You may notice some non-smoothness in the FastEM results. This can be reduced or removed by using more points for each variable. However, using more points may slow down the preparation time. You should try to balance it. You can close the MGRID window. The FastEM data is saved into the geometry file. You can do more tuning in real-time in the future without the preparation again.

Section 5.10 Summary

We have discussed wide scopes of topics in this chapter: (1) Antenna simulation; (2) EM optimization; (3) Current distribution display and animation; (4) Pattern calculation and visualization; (5) Finding array pattern from array factor and element pattern. (6) Other patterns post-processing capabilities. (7) Geometry tuning; (8) FastEM Design Kit for real-time EM tuning and optimization at design time. As you can see, IE3D is an extremely capable EM simulator for antenna and other high frequency applications.

FastEM Design Kit opens the door for general purposes full-wave EM designs including tuning, optimization and synthesis. It can reduce your design cycles significantly and achieve goals you can't imagine in the old time. There are many technologies involved to bring this wonderful feature to the users. The example in this chapter demonstrates the usefulness of this great feature. We will try to demonstrate more examples of FastEM in this manual and in a separate documentation specifically devoted to IE3DLibrary. Table 5.4 summarizes the applications of FastEM. Tips of using FastEM are provided in Table 5.5. We will end this chapter here.

Table 5.4 The major applications of FastEM Design Kit.

Application	Explanation
Full-Wave EM Tuning	You can easily tune your structure. You can parameterize your structure. Then, you can change the parameters and see how they affect the performances of your structure. From this tuning, you can understand different mechanisms of your designs, and what you can do to make it better.
Unlimited Full-Wave EM Models	You can create a library of basic elements with high accuracy full-wave models. Circuit simulators are based upon simple elements. They are fast and simple to use. However, the element libraries are limited to some regular shapes with know performance. Also, the accuracy of some elements is not good. Using FastEM Design Kit, you can create a library of elements of your own without limitation. You can find high accuracy results of your elements real-time at the design time.

Full-Wave EM Optimization	Full-wave EM optimization has been in IE3D for more than 10 years. It is extremely useful in high performance design. If we just need to create one design from a specific structure, FastEM optimization will not have much advantage over regular IE3D full-wave optimization. The reason is that the FastEM data preparation time may be even longer than a regular IE3D full-wave optimization. The advantage of FastEM optimization is in the re-usability of the designs. For example, you need to design a series of inductors with different L and Q values with the same process and same geometry configuration. If you use regular IE3D full-wave optimization, you have to perform an optimization of many full-wave simulations for each of the inductor. After an inductor is designed, the intermediate IE3D simulation results created in the optimization process is abandoned. It is a waste of resource compared to FastEM. Using FastEM, you just need to prepare the FastEM data one time, you can use the FastEM data for all your inductor designs and you can use it forever.
Full-Wave EM Synthesis	A synthesis process requires large amount of optimizations. Thanks to the re-usability of FastEM data, you are able to perform large amount of EM optimization in a reasonable time period. It makes full-wave EM synthesis possible.

Table 5.5 Tips in using FastEM Design Kit.

Tips	Explanation
Choose the right structures for FastEM process	As we always say, you will need to know what you try to do and what you may be able to achieve. You need to provide a reasonable structure to IE3D FastEM Design Kit for synthesis. In some sense, synthesis tools are not to replace engineers; they are to help the engineers. As a designer, you need to know what you are doing and what you can do. The FastEM Design Kit is just a tool for you, the master, to use to create great designs.
Parameterize structures wisely	Parameterization is the necessary part in FastEM design. You need to parameterize the structure. On IE3D/MGRID, you are able to define the locations of vertices and polygons as optimization variables. For simple structure, you can do it quite easily. However, for some sophisticated structures, you may find it tedious or even impossible to do on IE3D/MGRID, the polygon based layout editor. To make FastEM Design Kit useful, we have implemented the same scheme and many other advanced features on IE3DLIBRARY to make it simple. For example, you want to parameterize an elliptical shape patch antenna with an elliptical hole inside. It is impossible to define variables on MGRID to control the shapes and locations of the elliptical patches. However, you can do it easily on IE3DLIBRARY. It may be simple to parameterize the size of a spiral on MGRID. However, it is extremely difficult to parameterize the strip width and gap width of the spiral. However, such a parameterization is completely built-in in IE3DLIBRARY. You can't imagine how you can parameterize the radius, the height and shape of wire bond on MGRID. However, such a parameterization is just a simple task on IE3DLIBRARY. In this chapter, we touch base with FastEM Design Kit on IE3D/MGRID. In fact, the combination of FastEM and IE3DLIBRARY will make it much more powerful. Additional documentations on using FastEM on IE3DLIBRARY will be provided.

Limit the number of variables, ranges of variables, and number of frequencies	Large number of data is involved in a FastEM process. Just use the patch antenna we are discussing in this chapter as example. There are two variables, more than 100 different combinations, and 101 frequency points. We need to simulate all or at least a big portion of those combinations and extract the "signature" out from the data. The extracted "signature" is much smaller than the original data. However, it still contains large amount of data. It may need much time in the preparation phase. We should try to limit the number of variables (Nv), the number of simulation points in the range of each variable (Np), and the number of frequency points (Nf). When Nv = 1 and Np = 10, we may need to simulate about Nt = Np = 10 combinations. When Nv = 2 and Np = 10, we may need to simulate a big fraction of Nt = Np · Np = 100 combinations. When Nv = 3 and Np = 10, we may need to simulate a significant portion of Nt = Np · Np · Np = 1000 combinations. When Nv = 4 and Np = 10, we may need to simulate a fraction of Nt = Np · Np · Np · Np = 10000 combinations. As you can see, the increase is very fast with increase in Nv or Np. We should try to limit the Nv or number of variables, and the number of points for each variable at least for those variables with smaller range. You should also try to decide more or fewer points depending upon how important your model is. If you require high accuracy for your model, you will use the model very much in your design, and you have a weekend to prepare the data, you can use more points and more variables. You just let it run over the weekend. In case a preparation does not stop after a long run, you can terminate it and save the data for use even though it may not converge completely.

Chapter 6 Modeling of MMIC, RFIC and PCB Structures

We have learned the basic skills to create and simulate structures on the IE3D in the previous chapters. In this chapter, we will discuss the applications of IE3D in designing MMIC, RFIC and PCB structures.

IE3D is a general purpose EM simulator with extreme capability, flexibility, accuracy and efficiency in modeling different kinds of high frequency structures. It is widely used in MMIC, RFIC and PCB design as well as antenna design. In this chapter, we will concentrate on modeling of MMIC and RFIC structures.

Most MMIC and RFIC structures are quite complicated. The layout may be done in other layout tools and we need to import them into IE3D and perform the high accuracy EM simulations on them. We will start this chapter on importing.

Section 6.1 Different Import and Export Formats Available on IE3D

The most popular layout formats in the electronic industry are the GDSII (Calma Stream) format, DXF (AutoCad) format, GERBER format and ACIS format. IE3D is to import and export files in GDSII format and CIF (CalTech Intermediate Form) format directly. Import and export of DXF, GERBER and ACIS formats are through the optional ADIX license. When ADIX license is enabled, you will be able to perform the bi-directional conversion IE3D⇔GDSII⇔DXF⇔GERBER⇔ACIS on MGRID and ADIX.

On the File->Export command, we have the following options: (1) DXF; (2) GDS; (3) 3D Text (Mentor Graphics IE3D); (4) ACIS; (5) GERBER; (6) CIF; (7) FIDELITY. The 3D Text is a format created by Mentor Graphics IE3D. Basically, it allows you to write a structure as a script file. The 3D Text format is discussed in Appendix O. The FIDELITY format is for export only. You can export a geometry in MGRID into the FIDELITY format for our FIDELITY simulator. It allows you to define thickness to polygons for 3D objects. Only horizontal polygons (z-directed), x- and y-directed polygons can be exported. Polygons of other orientation will be omitted. Exported FIDELITY structures normally need clean up on FIDELITY before a simulation can be done. More automated exporting will be available in the future.

In recent years, we have implemented the Automatic Geometry to IE3D Flow (or AGIF). AGIF can do 3 jobs: (1) Convert GDSII layout into IE3D models automatically. After the conversion, the IE3D geometry can be a full-3D model containing vias, wire bonds, solder balls, and ports and ready for simulation. Users can create one template for one MMIC or RFIC process. The template can be used for all different GDSII files of the same process to do batch IE3D simulations directly from GDSII files. (2) Perform IE3D simulation on a cell or selected part of a cell from Cadence Virtuoso Custom IC Design Platform. Cadence Virtuoso is the most popular RFIC design layout tool. AGIF can simplify EM simulations of RFIC significantly. In the automatic conversion, AGIF can clean the structure for high efficiency IE3D EM simulations without losing accuracy. For example, pin-vias are parts of many semiconductor processes. Simulating pin-vias exactly is prohibitive because of the large number of pin vias can kill a simulation easily. AGIF allows users to merge pin vias into large vias without changing the electrical properties of the structure. The process is done automatically with specifications from users. Many advanced editing capabilities of IE3D fully automated into AGIF for streamlined IE3D simulations from GDSII and Cadence Virtuoso. (3) Create IE3D models directly from Cadence Allegro Interconnect Design Platform and Cadence APD. Cadence Allegro is a PCB tool and APD is a packaging tool. AGIF allows users to define selected nets as critical nets, reference nets and coupled nets. Then, it can automatically build the IE3D model for the critical nets including the all parts of the important parts of the reference nets and coupled nets. The created model can be ready for simulation. There is a separate documentation on AGIF. In this menu, we may give a simple example of using AGIF for automated IE3D simulation.

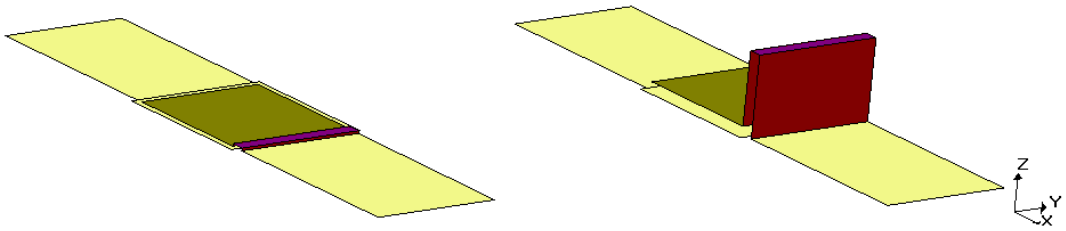
The critical part of the Automatic GDSII to IE3D Flow and the link to Cadence Virtuoso is the creation of the AGIF templates. The process is for you to define the technology file AGIF and program the automatic IE3D commands to perform the conversion on GDSII files to build IE3D model. For one class of GDSII files of the same process with the same number of ports, you may only need to define one AGIF template. You can apply the template to all the GDSII files. The AGIF template creation process may require you to be an experienced IE3M EM user because you may need to understand why we use some commands in such ways in the automatic procedure. Before you jump into AGIF, you are strong suggested to read this user's manual first and get familiar with the manual importing process. The AGIF template can also be a tedious process when there are many layers involved. To simplify the process of creating AGIF templates, we have implemented automatic conversion from TSMC Unified Technology File to AGIF template in IE3D 14. It can automatically convert a TSMC Unified Technology File into an AGIF template and you only need to do some minor manual adjustment and it will be ready for you to convert associated GDSII files into IE3D models automatically. There will be more implementations on other technology files. In case you have a technology file format and you would like it to be automated, please do contact Mentor Graphics on implementing such technology file.

In the last couple years, we have also implemented the EMSocket together with Applied Wave Research, Inc. EMSocket allows users to create layouts from Microwave Office of AWR, perform IE3D full-wave EM simulations seamless from MWO. The advantages of IE3D and MWO are combined together to make it convenient for designers. EMSocket is available automatically to IE3D and MWO users free of charge.

In the next section, we will show the importing of a GDSII file. We will use a simple MIM capacitor as an example. The procedure for importing a DXF file is similar except you may need to scale the imported structure after importing because a DXF file created for older versions of AutoCAD may not carry absolute dimensional information.

Section 6.2 Importing an MIM Capacitor in GDSII Format

The shape of the MIM capacitor is shown in Figure 6.1. The bottom plate is on a 100-micron GaAs substrate. The dielectric between the two plates is of thickness 0.2 microns and dielectric constant 6.7. The dimensions of the top plate are 100 microns by 100 microns. The air bridge is at 102-micron level. The GDSII file is in **.\ie3d\samples\mimcap.gds**.



(a) Original structure with two plates (b) Z-direction magnified by a factor of 30

Figure 6.1 The shape of the MIM capacitor to be built from importing.

Step 1 Run MGRID. Select **File->Import** command. MGRID will prompt you the Import Options dialog. Select **GDSII** for Import Format. Many of the options are grayed out because they are for the DXF conversion.

For DXF import, you may need to select Length Unit. You also need to define Circular Min Radius (Rmin) and the Segment (Smin), Circular Max Radius (Rmax) and the Segment (Smax). The parameters are for converting circular structures to polygons which are the only objects accepted by IE3D. When we encounter a circular shape with radius R, we will convert it into a

polygon with S-number of segment. The S is calculated based upon the formulas in Table 6.1. It is a linear interpolation based upon R, Rmin, Rmax, Smin and Smax. We should choose the Rmin, Rmax, Smin and Smax values wisely. For some applications where the circular shapes can be electrically large, we should choose Smin and Smax as 12, 16, 24, 36 or even 64. For some applications, the circular shapes are normally for vias and they are electrically very small. We can choose Smin and Smax as 4, 6 or 8. Normally, 6 is a good number. Choosing too big a number will cause IE3D create too many unknowns and the simulation time will increase substantially while the accuracy is not improved.

For GDSII structure, circular shapes are already converted into polygon shapes. We do not need to worry about it unless there are many segments in the circular shapes. We can use the commands Remove Chamfered Corners, Remove Redundant Vertices on Curvature and Convert Polygons in Shape in Adv Edit menu to reduce the segments for high efficiency EM simulation without reducing the accuracy. We will discuss the techniques later.

Table 6.1 The formula for Segments (S) based upon Radius (R).

R Range	S Formula
R < Rmin	S = Smin
Rmin < R < Rmax	S = Smin + (Smax – Smin) (R – Rmin) / (Rmax – Rmin)
R > Rmax	S = Smax

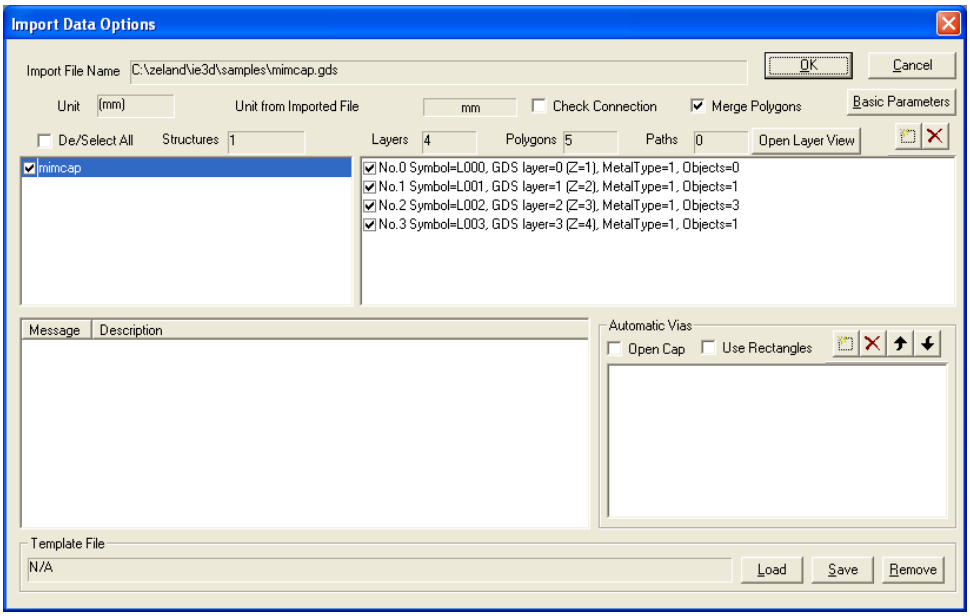


Figure 6.2 The Import Data Options dialog before the template file mimcap.ctp is loaded.

Step 2 Please select OK to continue. MGRID will prompt you for the GDSII file. Select file **.\ie3d\samples\mimcap.gds** and select OK continue. MGRID may prompt you to remove the existing structure for the imported one in case you had polygons before you select Import command. Select Yes to continue. MGRID will import the structure from the file. It comes up with the Import Data Options dialog (see Figure 6.2) with the list of all the structures detected in the file. There might be multiple structures in a GDSII file. You can select which structure you want to import. For this particular GDSII file, there is only one structure: mimcap and it is checked. All the layers detected in the structure are shown in the list box at the right. There are some parameters you need to enter.

The first group of parameters you need to enter are the Basic Parameters for the structure. They contain the substrates and the metallic strip types and the meshing parameters etc. The second group of parameters you need to enter are the layer information. A GDSII file contains only the layer information and the (X, Y) coordinates. It does not contain the Z-coordinates. We need to map the layers to some Z-coordinates. It is also possible a GDSII file use polygons to define the shapes of vias. However, they do not contain the third dimensional information about how the vias are connecting the layers.

We need to define all the above data in the Basic Parameters button, the layer information list box at the right and the Automatic Via list box at the lower right corner of Figure 6.2. Certainly, we can define some temporary data first. After we import the geometry into MGRID, we can change the layer z-coordinates, redefine the Basic Parameters, and build the vias using the polygon shapes using the Adv Edit->Build Holes and Vias from Selected Polygons command. It can be a tedious process. We can use the Import Template File (.ctp) to ease the process.

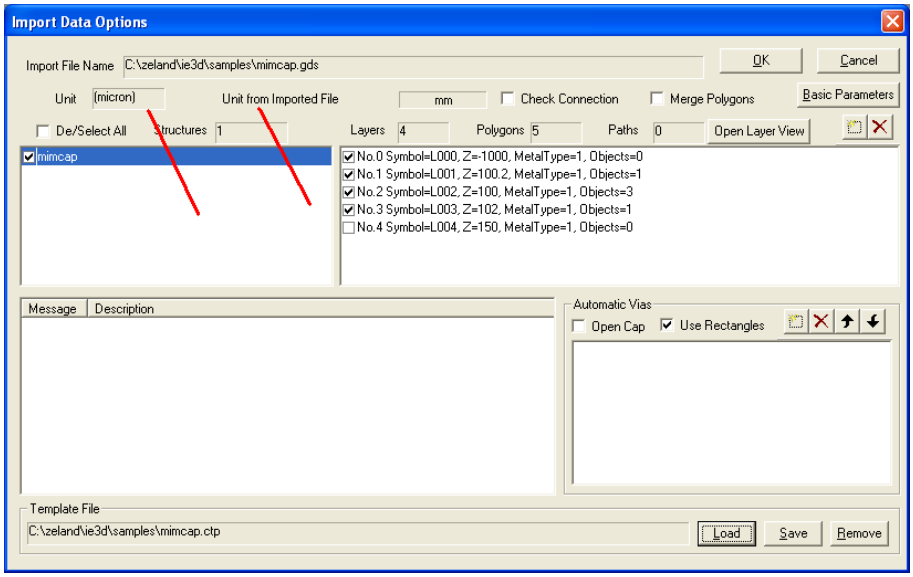


Figure 6.3 The import Data Options dialog after the template file is loaded.

Step 3 We have already created the Import Template File: .\ie3d\samples\mimcap.ctp file for you. Please select Load button. Select the .\ie3d\samples\mimcap.ctp file. MGRID will load the .ctp file and use the information in the .ctp file to define the Basic Parameters and the Layer Information dialog (see Figure 6.3). All the layers are placed at the supposed z-coordinates they should be on. The Basic Parameters are also defined. If you choose the Basic Parameters button, it will show you the GaAs substrate configuration.

In case you have to manually enter the import option parameters, after you define all the data, you can select the Save button to save the defined data into an Import Template File (.ctp). You can use the .ctp file for future importing of similar structures without going through the same procedure. In fact, we are doing so by using the mimcap.ctp file here now. Please pay attention to the Unit and the Unit from Imported File. The Unit from Imported File is the length unit from the GDSII file. The Unit is the one you want to use and it may be from Basic Parameters from

Step 4 Please un-check the Merge Polygons option. If you check it, it will try to merge the imported polygons if they are connected. For this case, we would like not to do so. Select OK to continue. The polygons are imported and placed at the correct Z-coordinates (see Figure 6.4). If you select

Parameters->Basic Parameters, you will see all the substrate layers for the structure is correctly defined. The length unit is micron.

There are five polygons imported (see Figure 6.4). Polygon 1 is on Z = 100.2 and it is the top plate of the MIM capacitor. Polygon 2 is on Z = 102 and it represents the air-bridge connecting the top plate to the polygon 3 on Z = 100. Polygons 3, 4 and 5 are on Z = 100. Polygons 4 and 5 are connected together while polygon 5 is the bottom plate of the MIM capacitor.

Step 5 Please save the file as: .\ie3d\practice\mimcap.geo first. We are going to clean it. The polygon 2 represents the air-bridge at Z = 102. We can delete it because we can re-build it later.

Step 6 Select Edit->Select Polygon. Click at No.4 Layer at Z = 102 to focus selection at the layer. Click at polygon 2 in Figure 6.4 to select it. Select Edit->Delete to delete it. We will get what is shown in Figure 6.5a.

We have labeled the vertices a and b for the right edge of polygon 1 and the vertices c and d for the left edge of polygon 3. We are going to re-build the air-bridge with vertical strips on it.

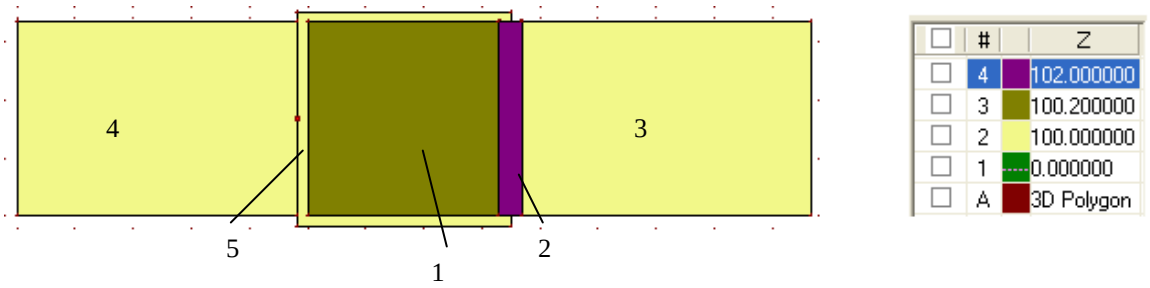
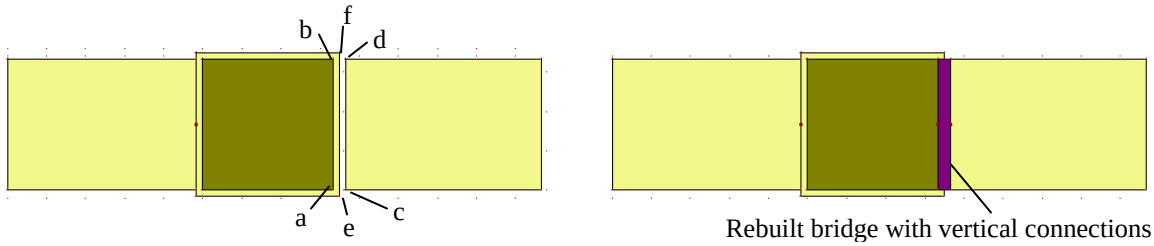


Figure 6.4 The imported polygons and the Layer Window.



(a) The structure with the polygon 2 deleted (b) The structure with the bridge rebuilt.

Figure 6.5 The intermediate structures in cleaning.

Step 7 Select Edit->Select Vertices. Check the No.2 Layer at Z = 100 and the No.3 Layer at Z = 100.2 to focus the selection on the two layers. Window the vertices a, b, c and d to select them. Please avoid selecting vertices e and f on polygon 5 at Z = 100. If you have difficulty to select vertices a, b, c and d in one selection, you can select them one by one, each with a windowing.

Step 8 Select Edit->Add Via on Edges. Change the End Z-Coordinate of the Via to 102. Select OK. MGRID will build two vertical strips: one from 100.2 to 102 at the vertices a and b location, and the other from 100 to 102 at the vertices c and d location. You cannot see them from the top view. However, you will be able to see them on the 3D view. You may want to zoom the 3D view large enough. You also change the Transformation (accessible from the tool bar of 3D view) to change the zoom ratio for the z-direction. You can make it many times bigger in displaying the z-direction.

Step 9 You should be in the drawing mode right now. Click at No.4 Layer at Z = 102 to focus input at the layer. Select Input->Set to Closet Vertex for the snapping mode. Click at vertices “a”, “b”, “d”, “c” and “a” sequentially to build the polygon on Z = 102 (see Figure 6.5b).

Step 10 Select Port->Port for Edge Group. Select Advanced Extension port scheme. Select OK to continue. Make sure the Layer at Z = 100 is checked. Window the left edge of polygon 4 to define the port 1 on it. Window the right edge of polygon 3 to define the port 2. Select Port->Exit Port to exit the mode. Save the structure as: .\ie3d\practice\mimcap1.geo. We should get the exact structure we want to model (see Figure 6.6). Select Windows->3D Geometry Display to display the 3D view. You will get the similar picture in Figure 6.1a. While the 3D view is the active window, select Manipulate->Transformation. Changing the Z-Scale Factor from 100 to 3000, we will get similar display in Figure 6.1b.

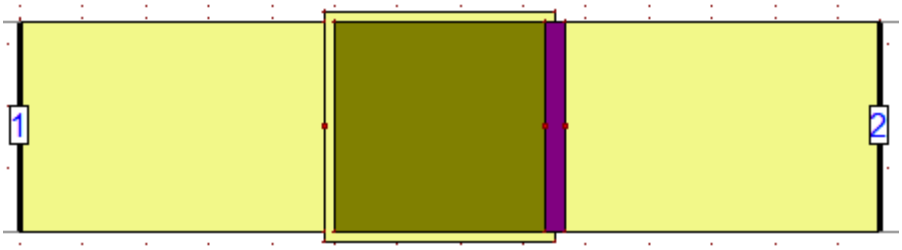


Figure 6.6 The final structure ready for simulation.

Section 6.3 Automatic Meshing Alignment

Is the structure in Figure 6.6 ready for simulation? The answer is yes. Will the model yield accurate results? The answer is Yes or No, depending upon which version of IE3D you use and how you setup the simulation.

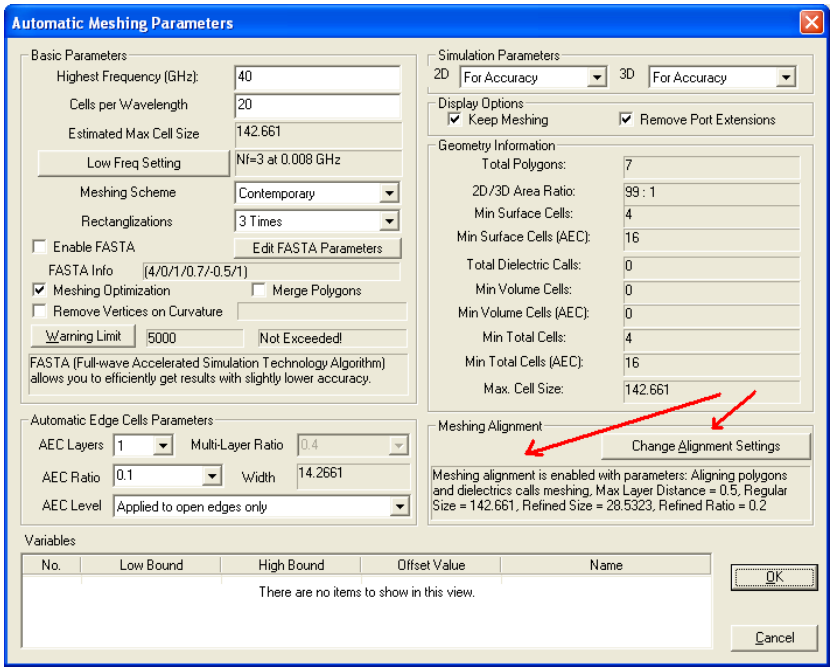


Figure 6.7 The Automatic Meshing Parameters dialog.

Old IE3D users may have known about the trick. For tightly coupled structure such as the two plates in a MIM capacitor, we need to match the meshing between the two plates. If we do not match the meshing, we will not be able to yield high accuracy results. Normally, the predicted C value for a MIM capacitor will be away from the parallel plate model which we know is quite close to the real value. Before IE3D 11, we require you to do a manual meshing alignment for it. You need to follow a fixed procedure in order to match the meshing between the two plates especially when grounding via present at the bottom of the parallel plate, which is quite common in MMIC design. The procedure is tedious even though it is quite standard and reliable. Manual meshing alignment is documented in Appendix AE.

One of the most significant implementations in IE3D 11 is the automatic meshing alignment. Let's see how it affects the meshing.

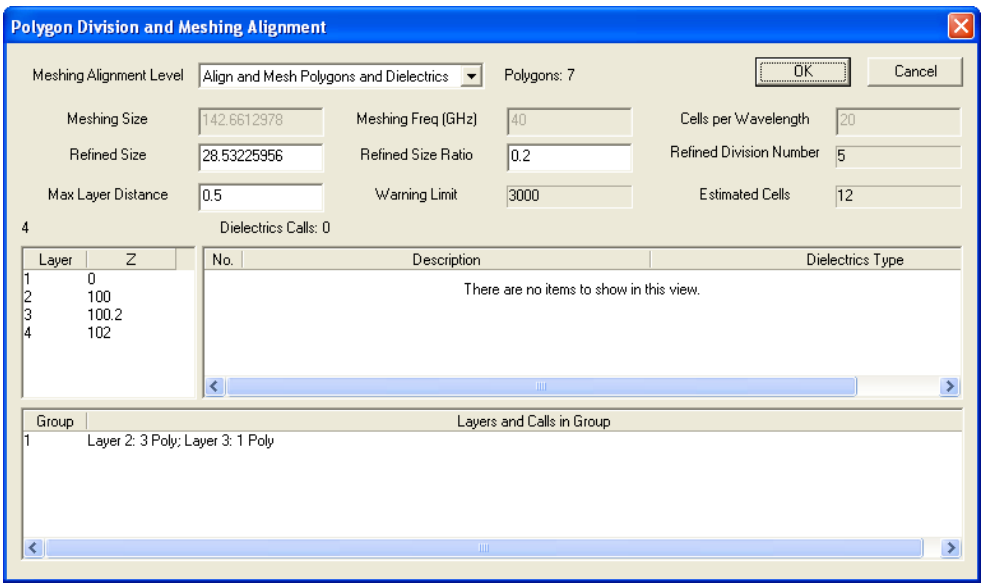


Figure 6.8 The Meshing Alignment dialog.

Step 1 Select Process->Display Meshing. The Automatic Meshing Parameters dialog comes up (see Figure 6.7). Please pay attention to the Meshing Alignment section. There is a button "Change Alignment Settings" and a description in it. Please click at the "Change Alignment Settings" button to bring up the Meshing Alignment dialog (see Figure 6.8). The Meshing Alignment Level combo controls whether and what level of alignment we want to have. Table 6.2 documents the different levels.

You may want to know what the definition of "Tightly Coupled" is. It is defined by the "Max Layer Distance" in Figure 6.8 and it is configurable by the users.

By default, the Meshing Alignment Level = Align and Mesh Polygons and Dielectrics. The Refined Size Ratio = 0.2 or the Refined Size is 20% of the Mesh Size or 28.5323 microns.

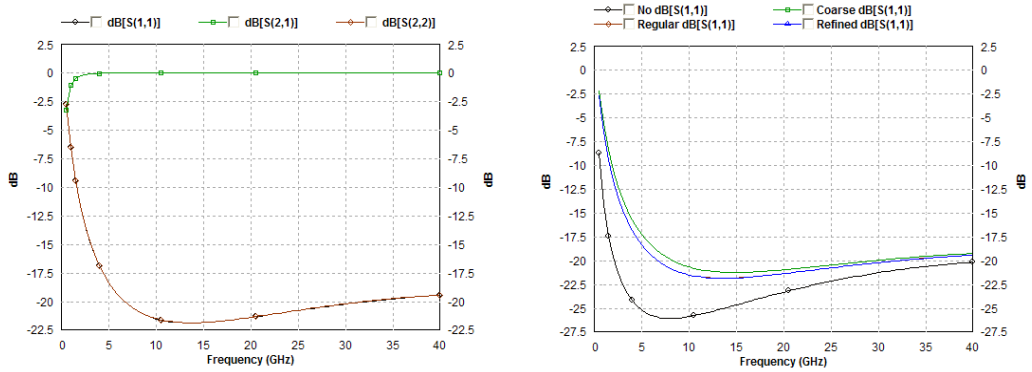
Step 2 Select OK to accept the parameters. Select OK on the Automatic Meshing Dialog to view the results. The meshed structure, using Classical Meshing scheme, is shown in Table 6.3. The results of Contemporary Meshing scheme may be different while the focused dense cells will be similar. Try different Meshing Alignment settings and see how they affect the results (see Table 6.3).

Step 3 Select Process->Simulate to simulate the mimcap1.geo structure from 0.5 to 40 GHz with 80 points. The result is shown in Figure 6.9a.

We can simulate the structure using the different meshing settings in Table 6.3. The comparison is shown in Figure 6.9b. You can see that the results Regular and Refined cases are almost identical. The Coarse result is slightly off the Regular case while there is an obvious difference between the No case and the Regular case.

Table 6.2 The Meshing Alignment Levels

Level	Description
No Alignment	It does not do any alignment and meshing on polygons and finite substrates.
Align Polygon Divisions	Align the divisions of tightly coupled polygons only. It will not mesh them into small cells.
Align Polygon and Dielectric Divisions	Align the divisions of tightly coupled polygons and finite dielectrics. It will not mesh them into small cells.
Align and Mesh Polygons	Align and mesh the polygons into small cells. The cell size is determined by the Refined Size (see Figure 6.8). For modeling MIM capacitors, we need to mesh the coupled plates into Refined Size which should be much smaller than the Mesh Size which is the regular cell size. The aligned and refined meshing will guarantee high accuracy.
Align and Mesh Polygons and Dielectrics	Align and mesh the polygons and dielectrics into regular cells and small cells. If two polygons are tightly coupled, we will align and mesh them into small cells of Refined Size. If a polygon is on a finite substrate, we will align and mesh them into cells of Mesh Size (regular cell size). This is the default option with user controllable parameter.



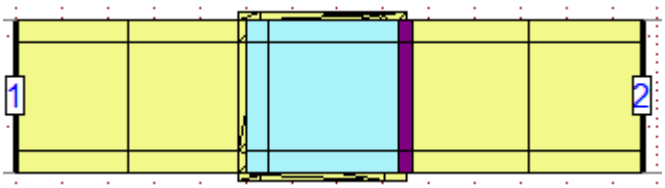
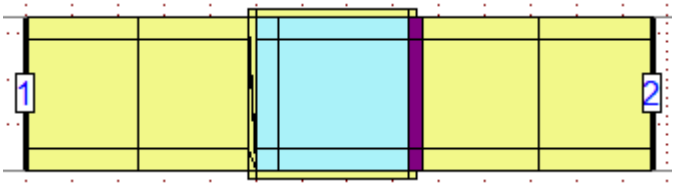
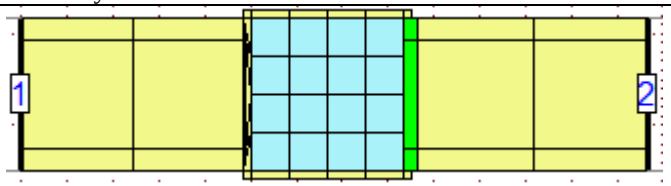
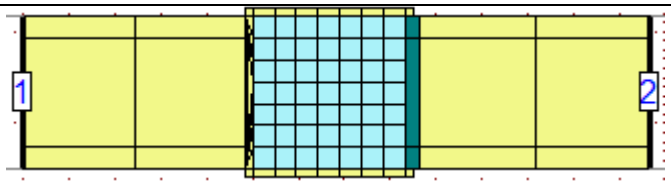
(a) S(1,1), S(2,1) and S(2,2) or regular meshing result. (b) Comparison of different meshings

Figure 6.9 The simulation results and comparison.

Section 6.4 Find the Equivalent Circuit of the MIM Capacitors

Are the results accurate? One way to see how accurate the results are is to find the mutual capacitance of the equivalent circuit between the two plates shown in Figure 6.10. It should be very close to the result of the parallel plate formula. We can use the scheme discussed in Appendix AW to extract the LC-equivalent circuits from the s-parameters on MODUA. You may notice that the scheme discussed in Appendix AW is for coupled TLN and interconnects structures. Our structure is not a coupled interconnect structure. However, we can trick the MODUA to get accurate equivalent circuit using the same scheme. In fact, we can also use MGRID to extract it. We will demonstrate the way on MODUA first.

Table 6.3 The different cases with different Meshing Alignment Settings.

Case	Settings	Meshed Structure and Description
No Alignment	No Alignment (mimcap1_no.geo)	 The cells on the tightly coupled plates are completely not aligned.
Coarse Alignment	Align and Mesh Polygons, Refined Size=142.661 microns (Refined Ratio=0.2) (mimcap1_coarse.geo)	 The cells on the tightly coupled plates are aligned while edge cells may not.
Regular Alignment	Align and Mesh Polygons and Dielectrics, Refined Size=28.5323 microns (Refined Ratio=0.2) (mimcap1.geo)	 The tightly coupled plates are aligned and meshed into smaller cells (about 20% of the regular cell size).
Refined Alignment	Align and Mesh Polygons and Dielectrics, Refined Size = 15 (Refined Ratio = 0.105144) (mimcap1_refined.geo)	 The tightly coupled plates are aligned and meshed into very smaller cells (about 10.5% of the regular cell size).

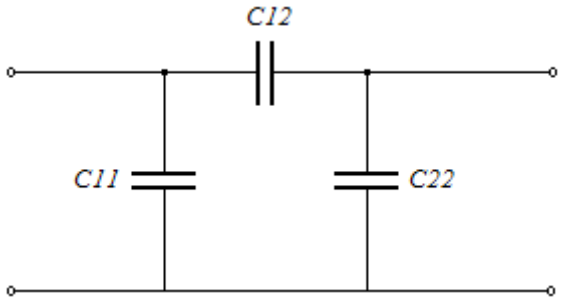


Figure 6.10 The equivalent circuit of the MIM capacitor at low frequency.

Step 1 Run MODUA. Select File->Display Parameter Module. Select .\ie3d\practice\mimcap1.sp. MODUA will display the s-parameters.

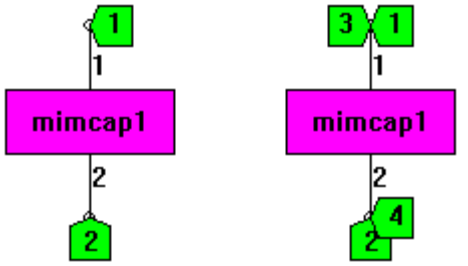
Step 2 Select Control->Display Toggle and we will get the automatic setup for the s-parameter module and the port modules from the nodal simulator (see Figure 6.11a). Select Element->Port.

MODUA will inform you that your change may cause loss of s-parameter data which is in fact the loaded s-parameters from the file. Please select YES because we do not need to worry about it because the data can be loaded again. Then, a port module is following the mouse. Click the right mouse button couple times to rotate the shape of the port module. Click at where the port 1 module is connected to or the terminal 1 of the module "mimcap1" to connect the port 3 module to it. MODUA is still in defining port module mode. Click the right mouse button once to rotate the port module. Click at the where the port 2 module is connected to or the terminal 2 of the module "mimcap1" to connect the port 4 module to it. We will get the picture shown in Figure 6.11b. MODUA is still in defining port module mode. Select Element->Exit Element to exit the mode. We have finished the setup for the extraction.

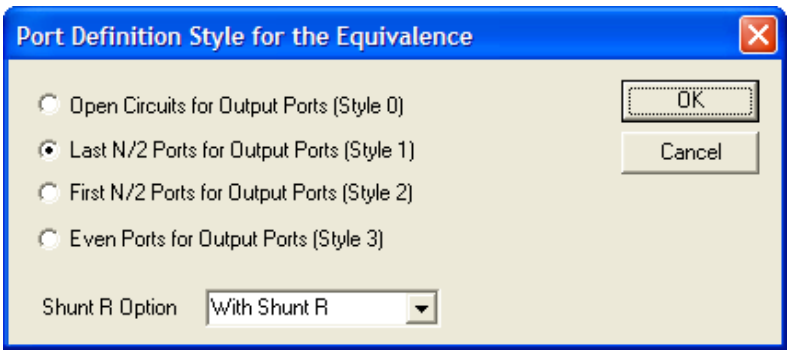
Step 3 Select Process->C Equivalent command. MODUA will prompt you for the frequency. All the frequency points are checked. If you choose more than 1 frequency points, we can find the values of the circuit components at each frequency point. However, the result will not be compatible with SPICE simulators.

Supposedly, the LC-Equivalent command is better for a general coupled interconnect structure. However, for our structure, there is no electrical length from port 1 to port 3 and from port 2 to port 4 (see Figure 6.11b). We will not have L values anyway. We should use C-Equivalent command.

Step 4 Please un-check all frequency points. Then, check the No.2 Frequency = 1 GHz. We are going to find the equivalent circuit at 1 GHz. Select OK. MODUA will prompt you for the Port Definition Style for the Equivalence (see Figure 6.12c). Our setup is to use the Last N/2 Ports for Output Ports (Style 1). We can also choose First N/2 Ports for Output Ports (Style 2). For the L-Equivalent command and C-Equivalent command, the 2 options are the same. For the LC-Equivalent command, they are slightly different in the locations of the components.



(a) default setup (b) setup for extraction



(c) The option for extraction.

Figure 6.11 MODUA setup and he option for LC-Equivalence command.

Table 6.4 The Pi-network parameters of the mimcap1.geo at 1 GHz.

C (pF)	1	2	R (ohm)	1	2
1	5.782e-2	3.069	1	3.949e+6	4.691e+4
2	3.069	3.48e-2	2	4.691e+4	2.689e+7

Table 6.5 The solved C(1,2) from different meshing alignment settings.

1 GHz	Parallel Plate	No Alignment	Coarse	Regular	Refined
C(1,2) pF	2.97	8.598	2.648	3.069	3.070

Step 5 Please select OK to accept the default setting. MODUA will perform the extraction and show the results on the window (see Table 6.4). Please select File->Save SPICE File to save the extracted result into: .\ie3d\practice\mimcap1.lib.

As it is shown, the R(i,j) values are very large and we can consider them as open circuits. We only have the C(i,j). The port 2 is connected to the top plate of the capacitor. We can see the C(2,2) is smaller than the C(1,1) and it is reasonable. The most critical parameter is the C(1,2) which is predicted as 3.069 pF.. Is it accurate? We can check it based upon the parallel plate mode. According to the parallel plate model, we have,

$$C(1,2) = \varepsilon 0\ \varepsilon r\ \ L\ W / D = 8.86\text{e-}12 \times 6.7 \times (0.1\text{e-}3)^2 / (0.2\text{e-}6) = 2.97\ \text{PF}.$$

The actual value of C(1,2) should be close to the value even it may not be exact. IE3D predicts C(1,2) = 3.069 pF and it is very close to the parallel plate predicted C(1,2). You may be interested in seeing what the other settings may yield. Table 6.5 shows the comparison on C(1,2). As you can see, the No Alignment result is far from the expected value. The Coarse Alignment case is obvious different while the Regular Alignment and Refined Alignment cases yield consistent results.

It is interesting to note that the Coarse Alignment yields quite reasonable s-parameters while the extracted C(1,2) value is obvious off. We can see the C(1,2) is more sensitive to numerical error. For the No Alignment case, we can increase the meshing density to see how it converges to the correct value. You will see the extracted C(1,2) value is converging to the expected value. However, the converging speed is slow with increased meshing density if the meshing is not aligned.

This example demonstrated the importance of meshing alignment and dense meshing to modeling tightly coupled structures such as MIM capacitors.

In practical design, the thin layers for the MIM capacitor may be just in a small area where the MIM capacitor is in. We can also use finite dielectric modeling for the insulator. We can easily model the thin layer as finite dielectrics on IE3D. We will delay the discussion to Chapter 16. The result of the finite dielectrics agrees perfectly with our aligned meshing result.

Step 6 As it is mentioned, we can also use MGRID to find the equivalent circuit. In the s-parameters processing dialog. Select “2-Port PI with Shunt RC and Shunt RC” in Lumped Model (see Figure 6.12). MGRID will list the extracted frequency dependent RC values of the model in the dialog. You can select Save Model button to save the extracted data into an ASCII file. It is a frequency dependent model and the results can’t be fitted into a SPICE format.

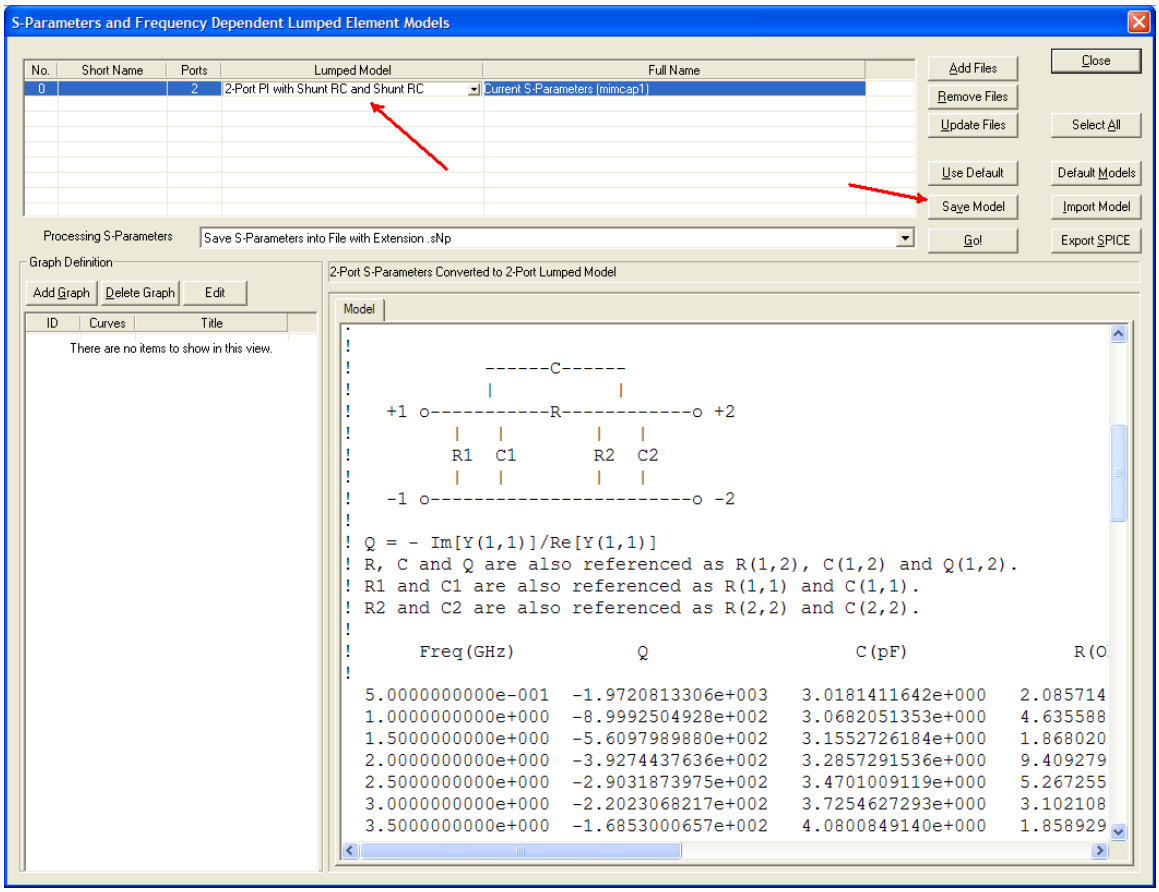


Figure 6.12 Extracting the equivalent circuit on MGRID.

Section 6.5 Comparison of Original Frequency Response and SPICE Equivalent Circuit

Another question you may have is how good the frequency independent RC model extracted from MODUA in SPICE format is in approximating the MIM capacitor. The accuracy of the IE3D simulator has been verified by numerous designers. Normally, its results are very accurate. For our MIM capacitor example, the mimcap1.sp should be very accurate. However, the equivalent circuit is extracted at 1 GHz. It should be accurate at 1 GHz and some low frequency range. However, it may not be very accurate at high frequency. From microwave theory, the circuit becomes more distributive with increasing frequency. We cannot expect the equivalent circuit will still accurate at very high frequency. Until what frequency is the equivalent circuit still good? We are going to show you how you can verify the equivalent circuit and how you can improve it.

Step 1 Run MODUA. If you are running MODUA with some elements or display on it, please select New and confirm the abandoning the results to get a clear window.

Step 2 Select **File->Import SPICE File** command. select **.\ie3d\practice\mimcap1.lib**. The SPICE file is imported and displayed on the window (see Figure 6.13).

Step 3 Select **Process->Simulate** command on MODUA. MODUA will prompt you for the simulation frequency points. Please select the Capture button. Select the mimcap1.sp file. MODUA will capture the frequency points in mimcap1.sp and list them on the list box. Select OK to continue.

MODUA will simulate the circuit in Figure 6.13 in seconds. MODUA may display the dB[S(1,1)] and dB[S(2,1)] for you automatically.

Step 4 Please select Control->Define Display Graph. Select dB and Phase of S-Parameters. Select dB[S(1,1)], dB[S(2,1)] and Ang[S(2,1)] to display them.

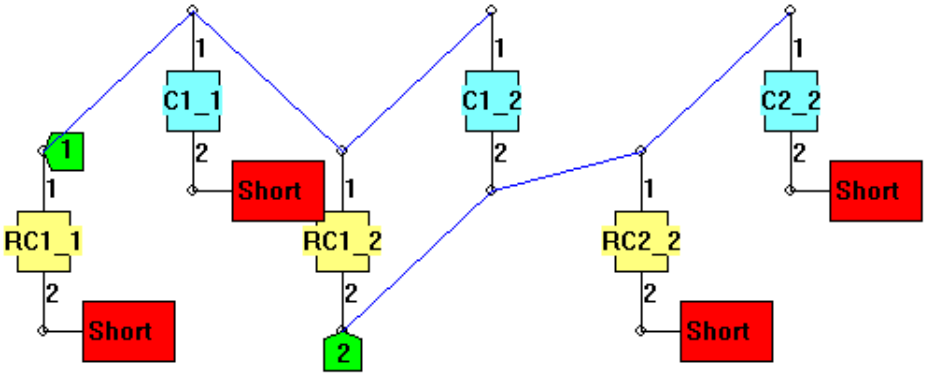


Figure 6.13 The extracted SPICE circuit imported back to MODUA.

Step 5 Please select **File->Parameter File Queue** on MODUA to add the mimcap1.sp into the queue and compare them (see Appendix AP for comparing s-parameters). Figure 6.14 shows the comparison. (original for the mimcap1.sp and Eq Ckt for the SPICE equivalent circuit). As you can see, the 2 results match quite well below 3 GHz. However, the differences become bigger and bigger with increased frequency.

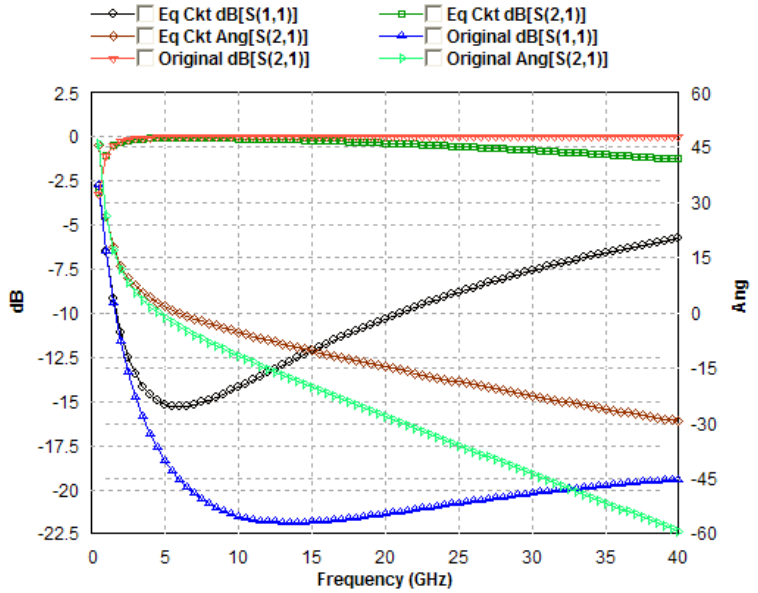


Figure 6.14 The comparison of the equivalent circuit and the original s-parameters in mimcap3.sp.

Section 6.6 Improvement on the Equivalent Circuit

We cannot expect the PI-network with capacitors and resistors can match the original very well at high frequency. There should be some inductance missing. In order to get better matching to the original s-parameters at the high frequency, we should introduce some inductance in series with the C(1,2).

Step 1 We can estimate the L-value based upon the electrical length of the structure in mimcap1.geo. Saved in .\ie3d\samples\thru_for_mimcap1.geo is a thru geometry with the same port locations as the mimcap1.geo. However, the MIM capacitor is replaced by a uniform TLN. We can simulate and find the L-value for this TLN section using the Process->LC-Equivalent command (see Appendix AW) or the Process->PI Network Equivalent command. We will find the L-value to be 0.1834 nH.

Step 2 On the MODUA with the mimcap1.lib SPICE file imported, please select **Control->Display Toggle** command to get the design view shown in Figure 6.13.

Step 3 Please click at the "connection" module connecting the terminal 1 of RC1_2 and the terminal 1 of C1_2 in Figure 6.15. The "connection" module is selected and it turns to red color.

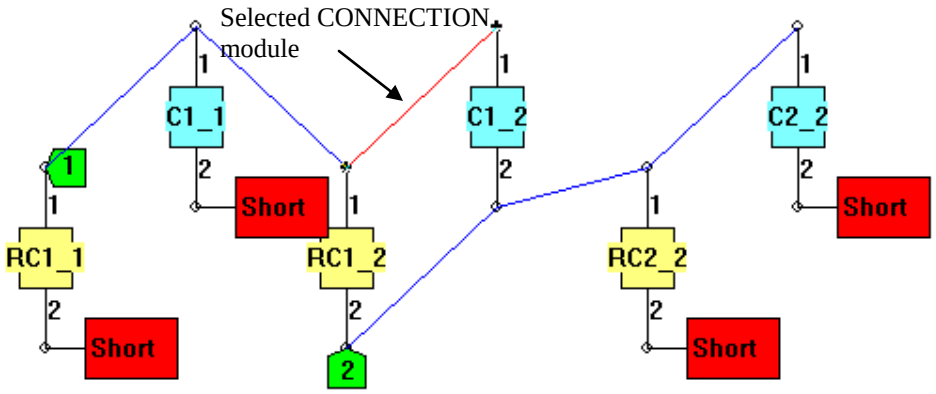
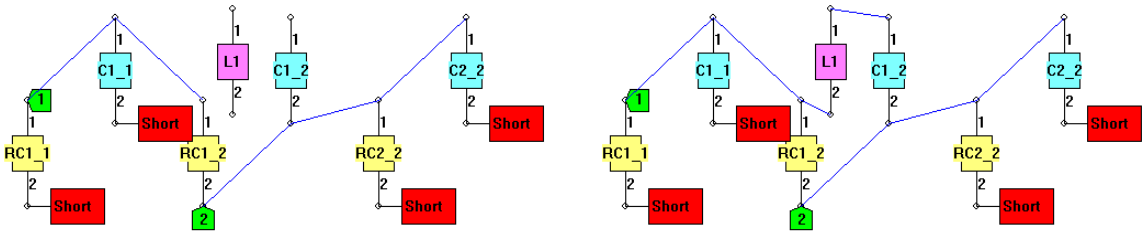


Figure 6.15 The "connection" module is selected.

Step 4 Please hit **Delete button** (or select **Edit->Delete Module** command) on MODUA. MODUA will prompt you that the simulation results will be abandoned. Select OK and MODUA will delete the "connection" module. We are going to connect an L of 0.1834 nH in series with the C1_2 there.



(a) The design with L1 added (b) The design with L1 in series with C1_2

Figure 6.16 The modified design to include the L1 in series with C1_2.

Step 5 Select **Element->Inductor** command. MODUA will prompt you for the value of the L. Enter "0.1826 " nH and select OK. The black box for the inductor is following the mouse cursor. Please move it somewhere on the left of C1_2 and click the left mouse button. The L1 is created there (see Figure 6.16a). The four corners of the L1 module are still highlighted, meaning they are still being selected. Please click at an empty spot on MODUA window to de-select it (equivalent to selecting **Edit->Fix Module** or **Edit->Exit Status** command).

Step 6 Select **Element->Connection** command. MODUA is set to creating connection module mode. Click at the terminal 1 of L1. A connection is connecting the terminal 1 of L1 to the mouse cursor. Click at the terminal 1 of C1_2. The connection is finished and it is connecting terminal 1 of L1 and the terminal 1 of C1_2. MODUA is still in the mode of defining connection module. Click at the terminal 1 of RC1_2. Then, click at terminal 2 of L1 to define another connection. Click an empty spot on MODUA window and select NO to exit the mode. It is equivalent to select **Element->Exit Element** command. We will get the design as shown in Figure 6.16b.

Step 7 Select **File->Save** command to save the design into .\ie3d\practice\mimcap1_eq.dsg file. Select **Process->Simulate** command on MODUA to simulate the modified design. After the simulation, MODUA displays the new results compared to the ones in mimcap1.sp file (see Figure 6.17). As you can see, the dB[S(2,1)] and Ang[S(2,1)] compare quite well in the whole frequency range. There is slight difference in dB[[S(1,1)]. The revised equivalent circuit yields excellent results. Can we get better match? The answer is yes. We can optimize the value of L1 (or even other components for it).

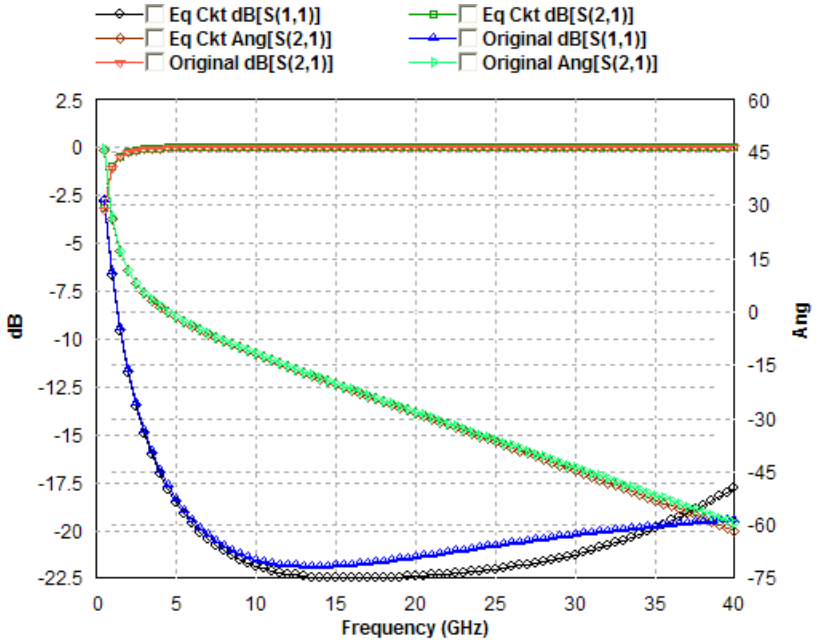


Figure 6.17 The comparison of the modified equivalent circuit and the original s-parameters.

Section 6.7 Optimization of the Equivalent Circuit

Step 1 Select Control->Display Toggle on MODUA to get the design view in Figure 6.16b. Click at L1 to select it (4-corners high-lighted). Select **Edit->Set Selected RLCM for Optim** command. The value of L1 becomes an optimization variable. Click an empty spot on MODUA to de-select the L1.

Step 2 Select **File->Save As** to save the modified design as .\ie3d\practice\mimcap1_eq_opt.dsg.

Step 3 Select **Process->Match Queue File** command on MODUA. This feature allows us to optimize the current design to match the 1st s-parameter file in the Parameter Queue Files list.

MODUA prompts you the simulation results will be abandoned. Select YES to continue. MODUA will list the defined frequency points. We do not need to change them. Select OK to

continue. MODUA will prompt you for the option of the optimization. There are different options for the weight and goals.

Step 4 Select OK to accept the default settings. The Optimization Setup dialog comes up. The goals are automatically created based upon the options. Select the Powell Optimizer for the Optimization Scheme. We know the goal is close and Powell Optimizer can do a good job. Select OK until IE3D is invoked to perform the optimization. It takes some seconds to finish it. The minimum residual is about 4.90873 and it is far from the 0.01 limit. Normally, we should not expect the minimum residual could go below the limit.

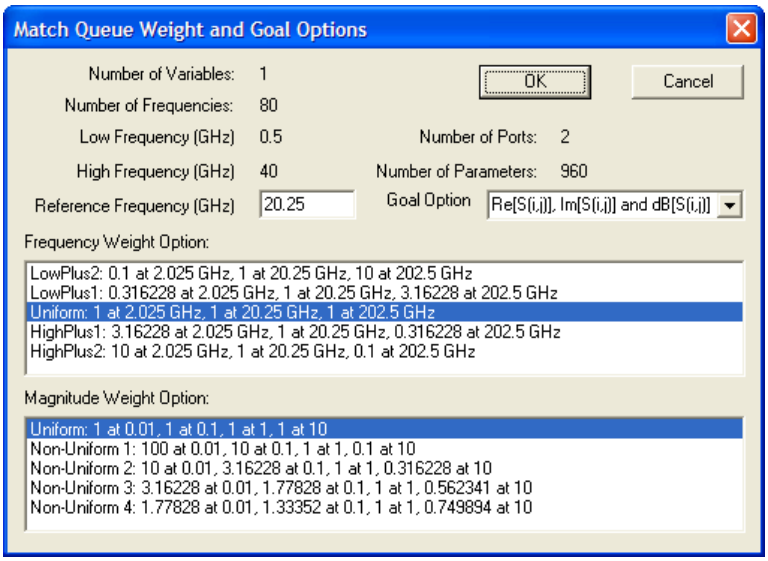


Figure 6.18 The Match Queue Weight and Goal Options dialog.

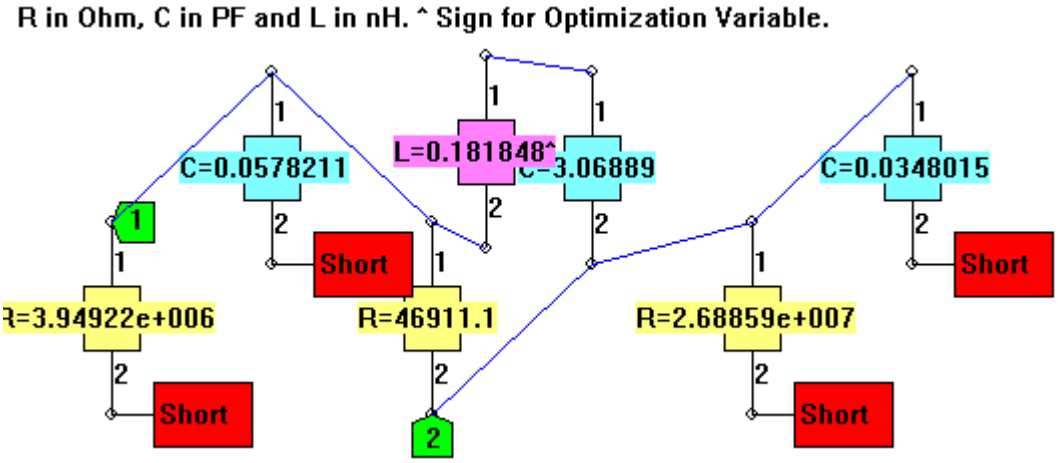


Figure 6.19 The mimcap3_eq1m.dsg with the values of the modules displayed.

Step 5 Select File->Open on MODUA to open the design: mimcap1_eq_optm.dsg file. It is the optimized design. MODUA will prompt you whether you want to save the change to the current design file. Please select NO and the mimcap1_eq_optm.dsg file is opened. It looks exactly the same as the mimcap1_eq_opt.dsg. However, the value of L1 should be changed.

Step 6 Select **View->Design View Toggle** command. MODUA will show the values of the different elements (see Figure 6.19). As you can see, the L-value on mimcap3_eq1m.dsg is changed to 0.181848 NH now.

Step 7 Please select Process->Simulate on MODUA to simulate the optimized circuit. Please compare it the s-parameters in the queue (see Figure 6.20). It is supposed that the optimized results are slightly better than the original result in mimcap1_eq_opt.dsg even you may not see it visually.

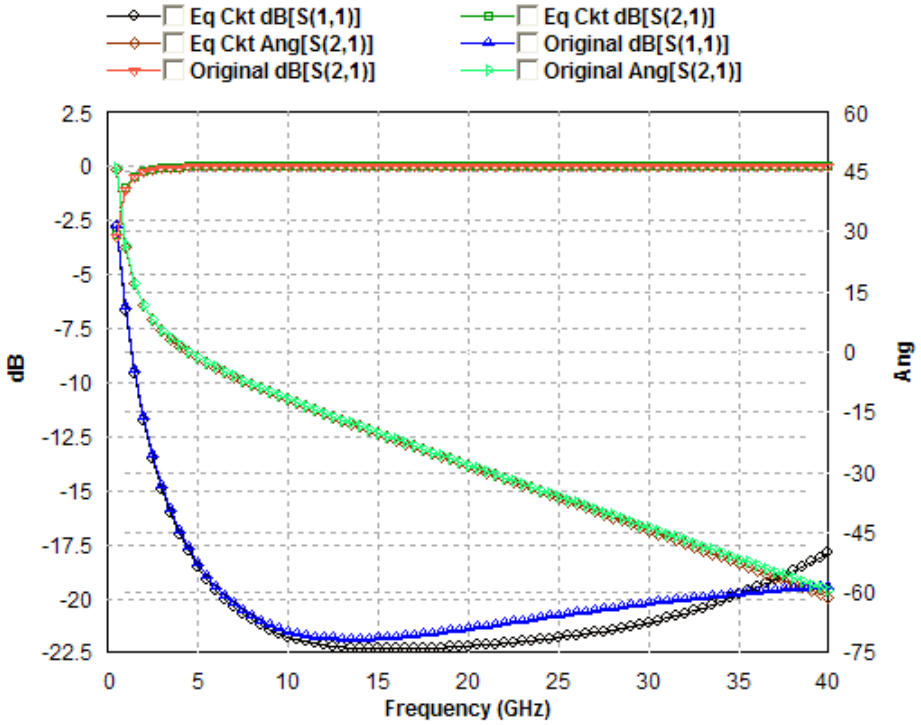


Figure 6.20 The comparison of mimcap3_eq1m.dsg and the original s-parameters.

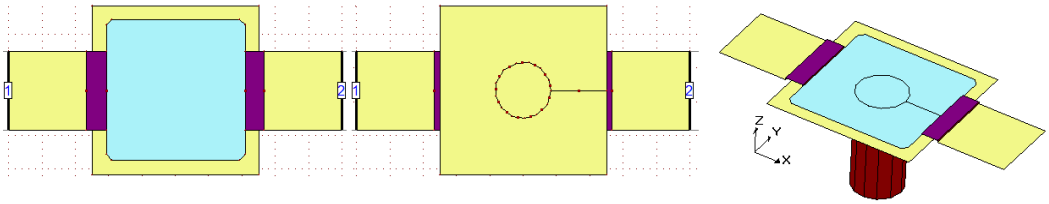
Section 6.8 Modeling Thin Dielectrics as Finite Dielectric Blocks

In practical applications, the thin layer for the MIM capacitors can be in a limited area. It might be more precise to model the thin layer as finite dielectric blocks. The thin layer has significant effect to MIM capacitors while it has no effect to the other portion of the MMIC circuit. However, modeling the thin layer as finite dielectric blocks gives us an alternative way to model such structures. It may further improve the accuracy and even the efficiency, even though IE3D has well implemented to handle one or multiple thin layers. Finite dielectric modeling is discussed in Chapter 16. We will demonstrate the modeling of the same MIM capacitor using finite dielectric blocks in Chapter 16. It shows that both the infinite thin layer model and the finite dielectric block model yield very close results.

Section 6.9 Importing Grounded MIM Capacitor

As you can see, if we can improve the simulation accuracy of MIM capacitors significantly by using aligned meshing between the top and bottom plate. The automatic aligned meshing significantly simplifies the process for accurate EM simulation of MIM capacitors. We can also get accurate equivalent circuit by optimization.

Another kind of structures requiring meshing alignment is the MIM cap with grounding vias (see Figure 6.21). It is a file saved in: .\ie3d\samples\viacap7.geo. These kinds of structures are frequently encountered in MMIC design. The feature of such a structure is that it has two coupled plates with a via connected to the bottom plate. We need to do a few processes simultaneously: (1) We need to build the via from a polygon and connect it to the bottom plate; (2) We need to align the meshing between the top and the bottom plates; (3) We need to make sure the aligned meshing match those of the via's shape connected to the bottom plate. We have a fixed manual procedure to do the above. We will show it to you how the automatic meshing alignment can help. We will also show to you how we can clean circular structures to simplify a simulation without losing accuracy.



(a) Top Plate (b) Bottom Plate (c) 3D View

Figure 6.21 An MIM capacitor with a grounding via.

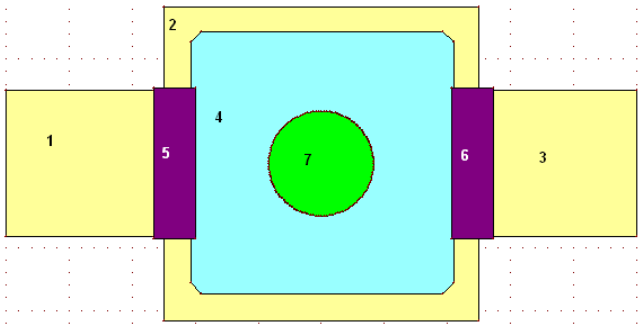


Figure 6.22 The imported 2D polygons put on the right layers specified in the .ctp file.

Step 1 Run MGRID. Select **Import** in **File** menu. Select **GDSII** for the Format. Select **OK** to continue. MGRID will prompt you for the GDSII file. Select file **.\ie3d\samples\viacap.gds**. The GDSII data is loaded and showing the layers to you. Select Load button. Select the template file: .\ie3d\samples\viacap.ctp. All the layers are put to the right locations and the Basic Parameters are set correctly. Select **OK** to continue. MGRID will display the imported polygons (see Figure 6.22).

There are 7 polygons imported. Polygons 1, 2 and 3 are on Z = 100. Polygon 2 is the bottom plate. Polygon 4 is the top plate at Z = 100.2. Polygon 5 and 6 at Z = 102 represent air bridges connecting the feed lines (polygons 1 and 3) to the top plate. Polygon 7 at Z = 150 represents a via from ground at Z = 0 to the bottom plate at Z = 100. We are going to clean the structure in some steps. Please save it as: .\ie3d\practice\viacap.geo first.

Section 6.10 Remove Redundant Vertices on Curvatures, Building Bridges and Vias.

Step 1 The polygon 7 basically represents a circular shape. Instead of true circle, it is modeled as a polygon with many vertices. The many vertices will increase the number of cells substantially with little improvement in the accuracy. We want to reduce the number of vertices on the circle. The diameter of the circle is about 50 microns. You can enter a line across diameter of the circle

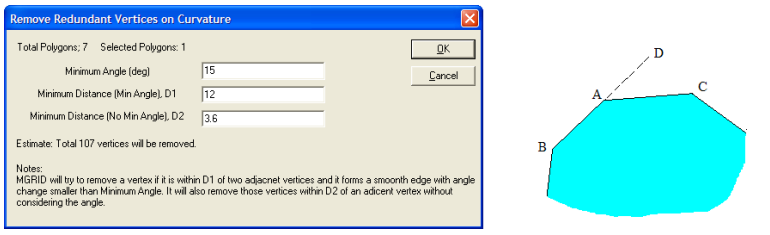
and use the **Input->Info on Last Entry** command to measure it. We need to know the approximate size of the circle before we can clean it.

There are multiple ways to clean it. One way to clean it is to use the Adv Edit->Convert Polygon in Shape command to convert it into a polygon of fewer vertexes to represent the circle. We have shown this command in Chapter 4. In fact, this is the best way. However, we are going to show you how to use the command: Adv Edit->Remove Redundant Vertices on Curvature.

Step 2 Click at the No.5 Layer at Z = 150 to focus the input to the layer. Press down the "Shift" key and window polygon 7 to select it.

Step 3 Select **Adv Edit->Remove Redundant Vertices on Curvature** command. MGRID will prompt you for the parameters (see Figure 6.23a).

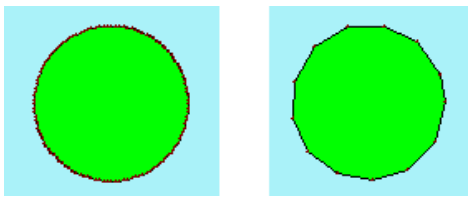
You are required to enter the Minimum Angle, Minimum Distance (Min Angle), Minimum Distance (No Min Angle). The 3 parameters control how we remove the redundant vertices on the curvatures. Their meanings are illustrated in the Figure 6.23b. For example, we are deciding whether we should remove vertex A on the curvature BAC. We denote the distance between A and B as Dab, and the distance between A and C as Dac. The angle between the line AC and AD is φ. There are a few things we will check: (1) If both Dab and Dac are smaller than the Minimum Distance (No Min Angle) specified in the dialog, the vertex A will be removed; (2) If any of Dab and Dac is smaller than the Minimum Distance (Min Angle) and the φ is smaller than the Minimum Angle specified in the dialog, vertex A will be removed. Otherwise, the vertex A will be kept.



(a) dialog (b) meaning and illustration

Figure 6.23 The Remove Redundant Vertices on Curvature dialog and illustration.

Step 4 Enter the Minimum Angle = 15, Minimum Distance (Min Angle) = 12 and Minimum Distance (No Min Angle) is automatically set to 3.6. MGRID will estimate to you that 107 vertices will be removed. Select OK to continue. MGRID will confirm to you that 107 vertices are removed. Select OK. You will see that the circle is less smooth (see Figure 6.24). However, it will lose little accuracy while it will improve the efficiency significantly. In practical applications, we can use 6-sided polygons, even square, to approximate small circular structures. The results normally are very good especially for the 6-sided polygon approximation. For our case, the circle has 120 sides before removing the vertices. After the removing vertices, there are still about 13-segments on the circle. It should be accurate enough.



(a) Before (b) After

Figure 6.24 The circle before and after the removing vertices.

Step 5 Save the file as: .\ie3d\practice\viacap1.geo.

Step 6 Select the polygons 5 and 6 and delete them. We would like to insert vertices I between the edge E and F, and we want to make sure AI is perpendicular to EF. We also want to insert vertices J, K and L with similar condition.

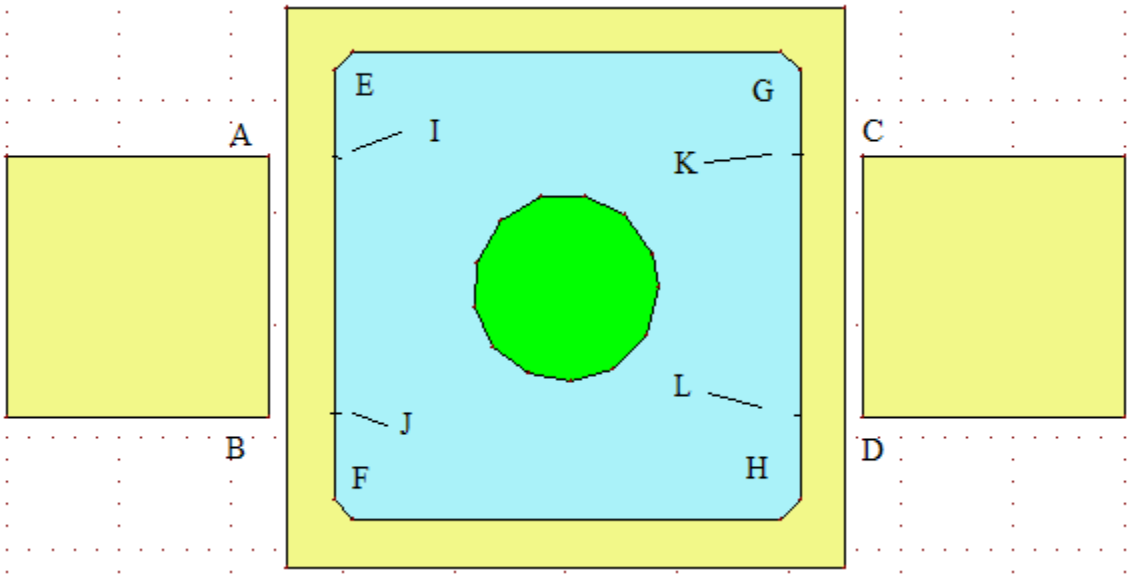


Figure 6.25 The structure with the polygons on Z = 102 deleted.

Step 7 Click at No.2 Layer at Z = 100 (where A is on). Select Input->Set to Closet Vertex for the snapping mode. Click at A to snap a vertex at it.

Step 8 Click at No.3 Layer at Z = 100.2 (where E is on). Select Alt+C. It is equivalent to selecting Input->Connect to Edge Perpendicularly. Click at somewhere between edge EF. MGRID will automatically find and insert a vertex at I to make sure AI perpendicular to EF. Hit ESC button to drop the entered vertices.

Step 9 Repeat Steps 7 and 8 to insert vertices at J, K and L.

Step 10 Select Edit->Select Vertices. Check No.2 Layer at Z = 100 and No.3 Layer at Z = 100.2. Window vertices A, B, C, D, I, J, K and L to select them only. Select Edit->Add Via on Edges. Enter End Z-Coordinate = 102. Select OK. MGRID will build 4 vertical strip from Z = 100 to 102 or Z = 100.2 to 102. Then, we can enter the two horizontal strips for the bridges at Z = 102. Similar steps are demonstrated in Chapter 4 when we were dealing with the bridge.geo file. Please save the file as: .\ie3d\practice\viacap2.geo. It is shown in Figure 6.26.Our next step is to build the via and it should be straightforward.

Step 10 Select the polygon at Z = 150 representing the circular via. Select Adv Edit->Build Holes and Vias from Selected Polygons. Select The Z of New Hole = 0 and select Add button to add it into the list. Select The Z of New Hole = 100 and select Add button to add it into the list. Check the Build Vias Between Holes on Different Layers (see Figure 6.27). Select OK to build the via. Define port 1 on the left edge and port 2 on the right edge. Save the file as:

.\ie3d\practice\viacap3.geo. It is the final structure we want (see Figure 6.21). Simulation over a wide band should take couple minutes.

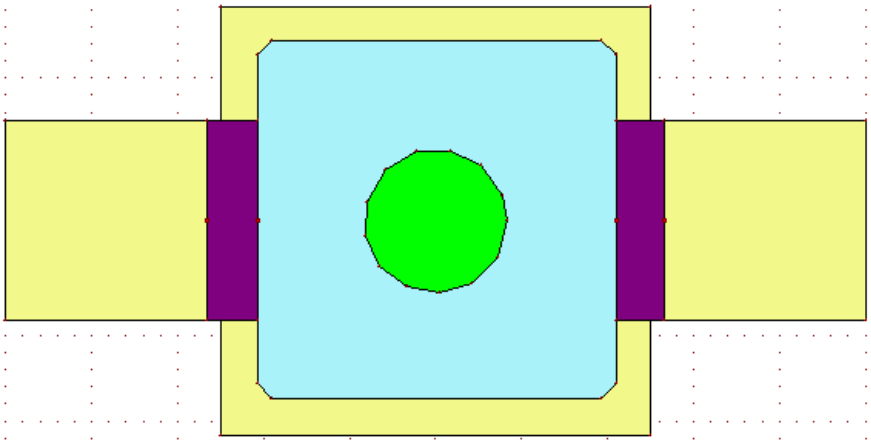


Figure 6.26 The structure with the bridges built.

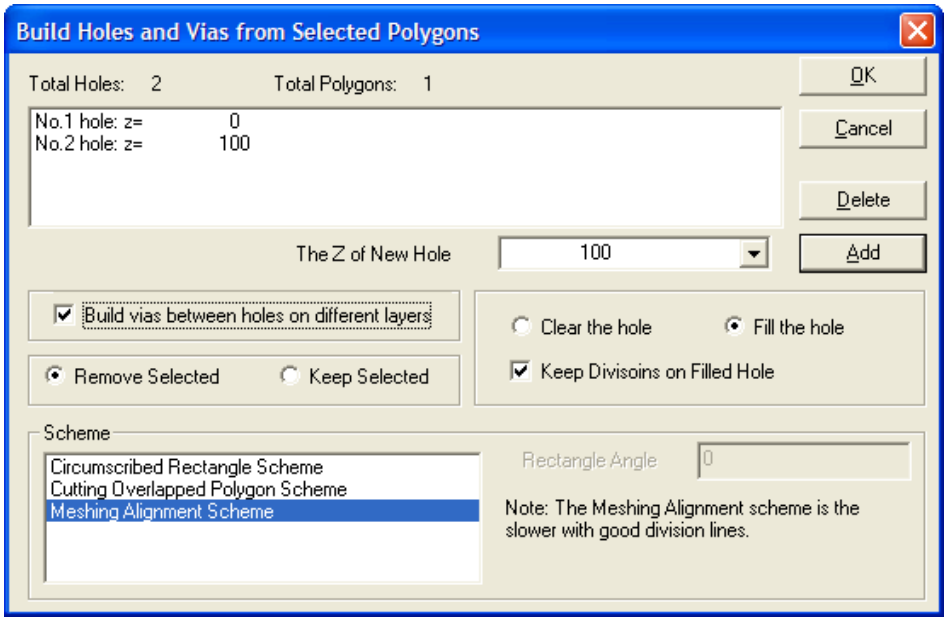


Figure 6.27 The Build Holes and Vias from Selected Polygons command.

For the IE3D 10.x and earlier version, the structure in viacap3.geo is not a structure ready for simulation because the meshing between the plates is not aligned. You would need to manually align the meshing before you can simulate it with confidence on accuracy. Fortunately, the automatic meshing alignment on IE3D eliminates such a need. The simulation result of the grounded via structure is shown in Figure 6.28, showing the typical response of S(2,1) with a notch at a specific frequency. Normally, we require the notch to be at a specific frequency. In case it is not, we can optimize it to achieve the notch at a specified frequency. The Automatic Meshing Alignment on IE3D 11 does not just eliminate the tedious work for manually meshing alignment for MIM capacitor with grounding via, it also makes it possible to optimize it. Without the automatic meshing alignment, we could not change the shape of such a structure and guarantee the aligned meshing for high accuracy. The Automatic Meshing Alignment makes such a goal possible.

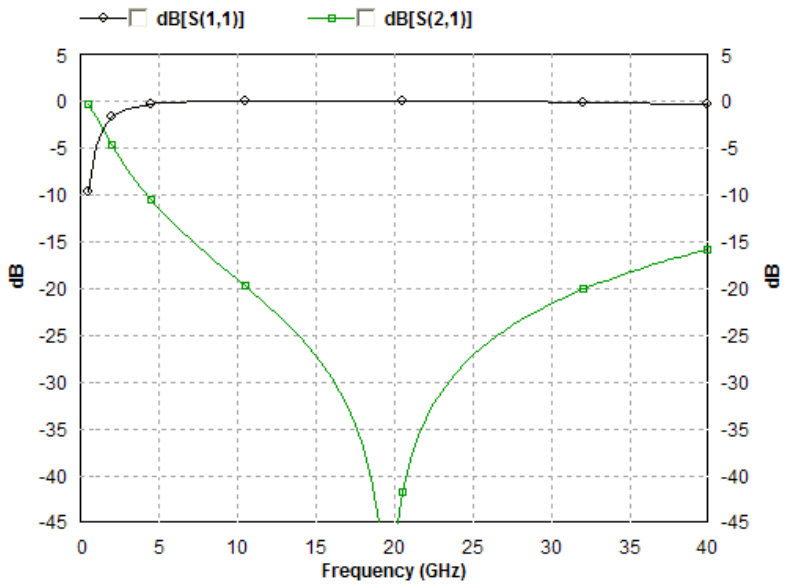


Figure 6.28 The simulation result of grounded via structure.

Section 6.11 Importing Structures with Automatic Vias Using Import Templates

As you can see from the above discussion, the Import Template makes it convenient to import the same types of structures easily. Also, the **Adv Edit->Build Holes and Vias from Selected Polygons** command is very powerful in building vias and other structures requiring Boolean operations.

Upon the request from customers, we have implemented the automatic vias in the Import Parameters dialog and even the Import Template file. You can configure the MGRID to create automated vias upon importing. We will discuss this feature in modeling multi-layer structures in this section.

Step 1 Run MGRID. Please select File->Import command. MGRID prompts you for the Import Options dialog. Select "GDSII" for the format. Select OK to continue. MGRID will prompt you for the GDSII file to be imported. Select the file: c:\ie3d\samples\vias_on_layers.gds file. MGRID will read the file and detect all the structures in the file. Then, it will prompt you the Import Data Option dialog (see Figure 6.29). As you can see, all the layers are defined with some assumed Z-coordinates by default. If you select the Basic Parameters button, you will see the substrates and metallic types are also defined with the default values.

Step 2 Select the Load button in the Template File section. Select the file: .\ie3d\samples\vias_on_layers.ctp file. The template file is read and the data inside the file is used to define the layers (see Figure 6.30). If you chose the Basic Parameters dialog, you will see that the substrate parameters are also re-defined. All the information is from the vias_on_layers.ctp template file. This file is saved by selecting the Save button on the Template File section after we defined all the layers and substrate in a previous importing process. The layers are documented in Table 6.6. The configuration is shown in Figure 6.31. Basically, the layers and the polygons on the layers define a microstrip to strip line transition with finite ground plane in a PCB. The polygon on the No.5 layer defines the shape of the hole on the No.3 layer or the top ground plane. The polygon on the No.6 layer defines the shape of the via connecting the two traces on the No.2 and No.4 layer and going through the hole on the No.3 layer. The polygons on the No.7 layer define the shapes of the vias connecting the 2 ground plane layers or the No.1 and No.3 layers.

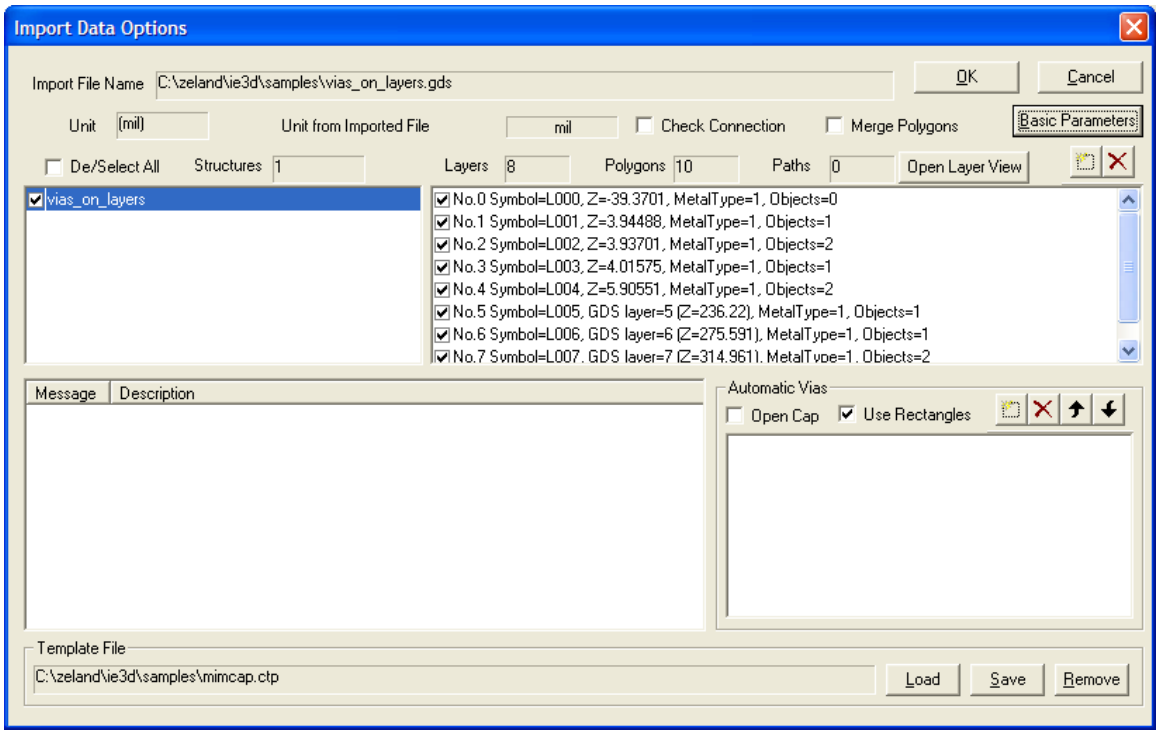


Figure 6.29 The Import Data Options dialog before opening the: vias_on_layers.ctp file.

Table 6.6 The layers and their descriptions

No.	Z (mils)	No. of Polygons	Description
0	-39.3701	0	This layer is not used.
1	0	1	This layer is the bottom finite ground of the stripline configuration.
2	20	2	This is the stripline trace layer.
3	40	1	This layer is for the top finite ground of the stripline or the finite ground of the microstrip configuration.
4	60	2	This is the microstrip trace layer.
5	80	1	This layer defines the shape of the hole on the top finite ground. The hole is for the via connection between the traces
6	100	1	This layer defines the shape of the via connecting the 2 traces.
7	120	1	This layer defines the shape of the vias between the 2 ground planes.

Step 3 Select OK to continue. The polygons are imported and the layers are put into the right Z-coordinates and the Basic Parameters are setup correctly. Please save the file as: .\ie3d\practice\vias_on_layers.geo. Based upon the description in Table 6.6, we are able to use the polygons on different layers to create the holes and vias for the final structure. However, this process can be automated using the Template File. We are going to demonstrate it.

Step 4 Repeat the Steps 1 and 2. We get all the layers defined in the: vias_on_layers.ctp file, and the Import Parameters dialog is shown in Figure 6.30.

Step 5 Click at the No.5 Symbol=L005, MetalType = 1, Objects = 1 layer in the Layer Definition list box to highlight it (no double-clicking it). Then, please hit the Insert button (see Figure 6.32).

The Edit Polygon Layers for Vias dialog comes up. Please check the Layer 3 (see Figure 6.32). Select OK to continue. The 1st Automatic Vias procedure is defined. It is list in the Automatic Vias list box on the right (see Figure 6.33). The No.5 layer on the Layer Definitions is also changed. It defines a procedure MGRID will take after you select OK button. The procedure is that MGRID uses the polygons on the No.5 Layer at Z = 80 to build holes on the No.3 Layer at Z = 40. Because No.3 Layer is the only passive layer for this procedure, MGRID will not build any via. In case, you checked more than 1 layer, there will be multiple passive layers in this procedure. In this case, MGRID will build vias between the passive layers.

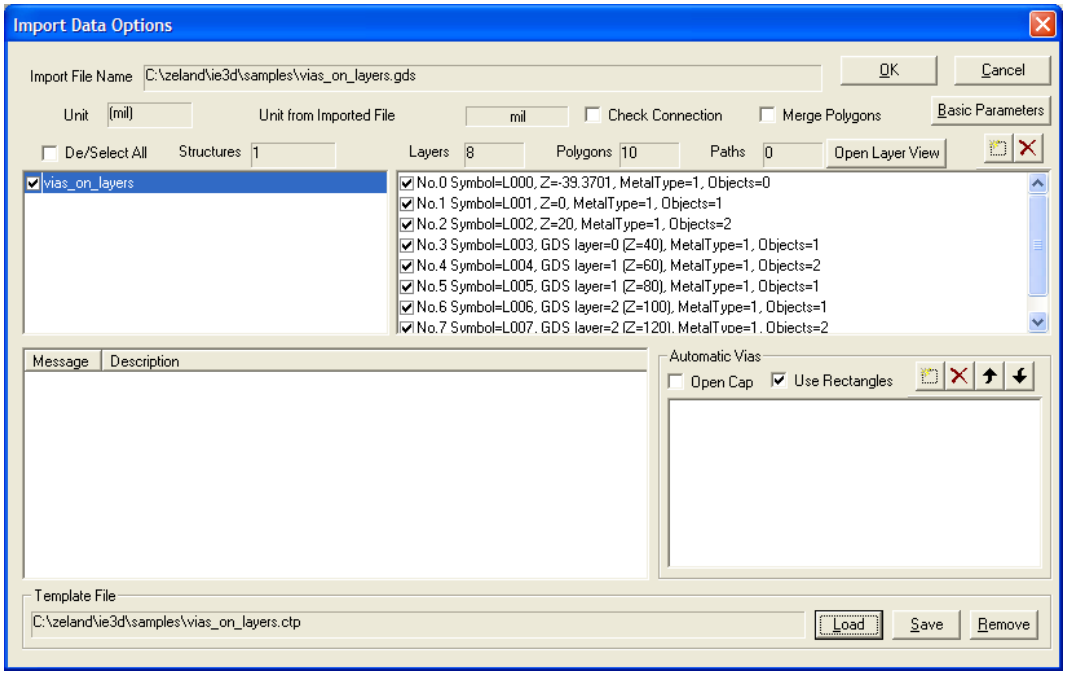


Figure 6.30 The Import Data Option dialog after opening the: vias_on_layers.ctp file.

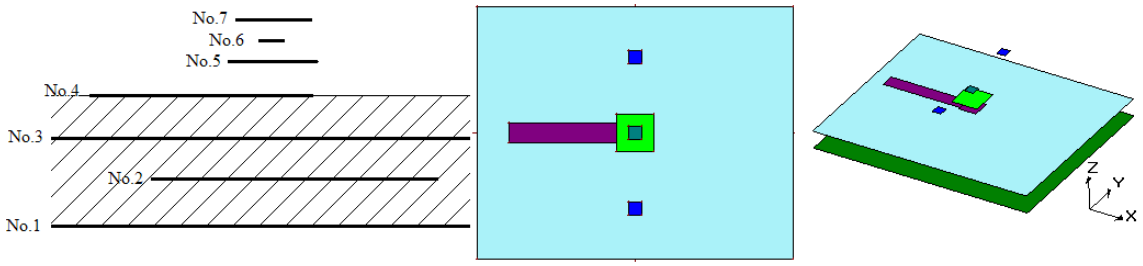


Figure 6.31 The illustration of the layer configuration, the top view and 3D view.

Step 6 Click at the: "No. 6 Symbol = L006,…" in the Layer Definitions list box to highlight it. Select the Insert button again. The Edit Polygon Layers for Vias dialog comes up. Pleas check the Layer 2 and Layer 4. Select OK to add the Automatic Vias procedure into the list. The procedure defines that MGRID will use the polygons on the No.6 layer to build vias between the two passive layers: No.2 and No.4 layers (the trace layers).

Step 7 Click at the: "No.7 Symbol = L007,…" in the Layer Definition list box. Select the Insert button again. The Edit Polygon Layers for Vias dialog comes up. Check the Layer 1 and Layer 3. Select OK to add the procedure into the list. Basically, the procedure defines that MGRID will use the

polygons on the No.7 layer to build vias between the 2 finite ground plane layers. Check the Open Cap option. We will get the Import Parameters dialog as shown in Figure 6.34.

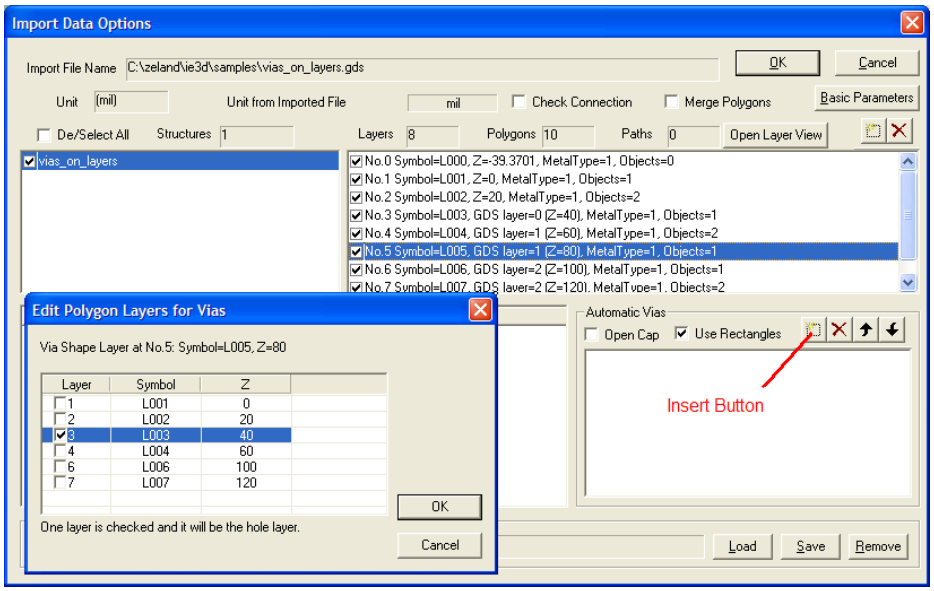


Figure 6.32 The Import Parameters dialog and the Edit Polygon Layers for Vias dialog.

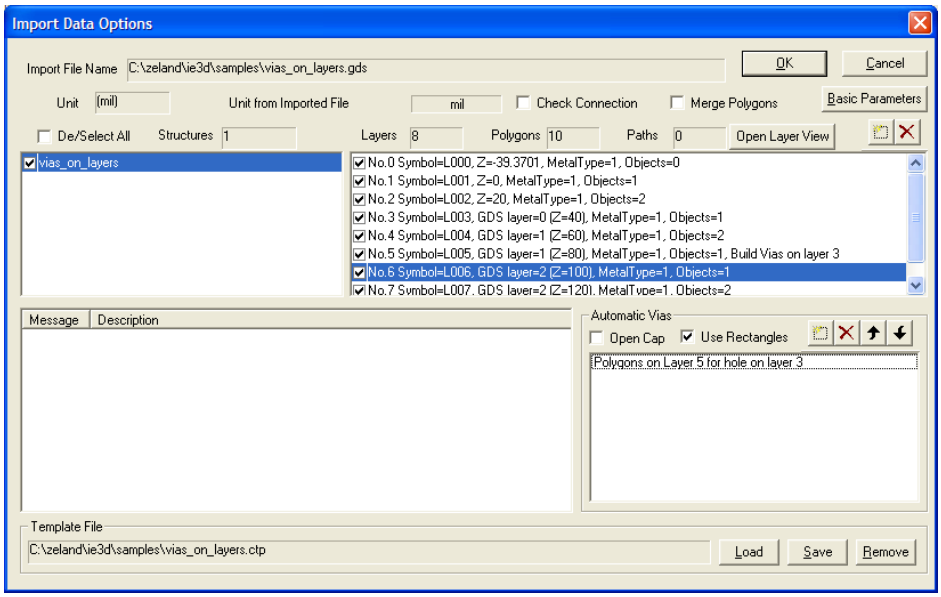


Figure 6.33 The Import Parameters dialog with the 1st Automatic Vias procedure defined.

Step 8 If we select OK on the Import Data Options dialog, MGRID will import the polygons into the right layers and build the holes and vias as instructed. However, we would like to save the data into a template file first for future use.

Select the Save button in the Template File and save it as: .\ie3d\practice\vias_for_layers2.ctp. Select Cancel button to cancel the process. We are going to demonstrate how we can use the vias_for_layers2.ctp to define the layers and the vias automatically.

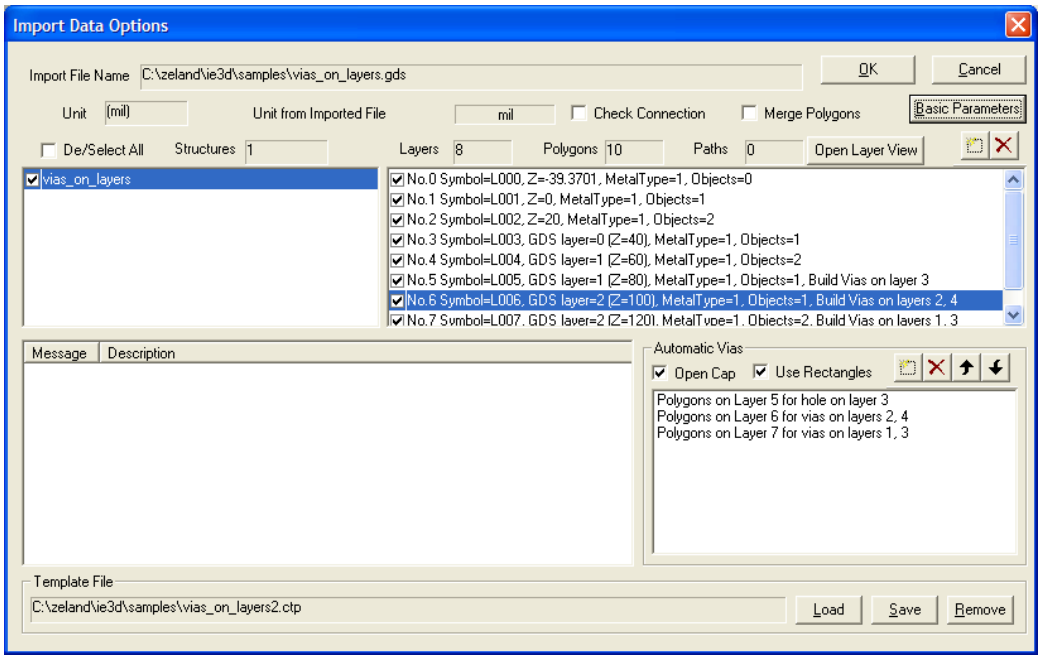


Figure 6.34 The Import Parameters dialog with all Automatic Vias procedures defined by Step 7.

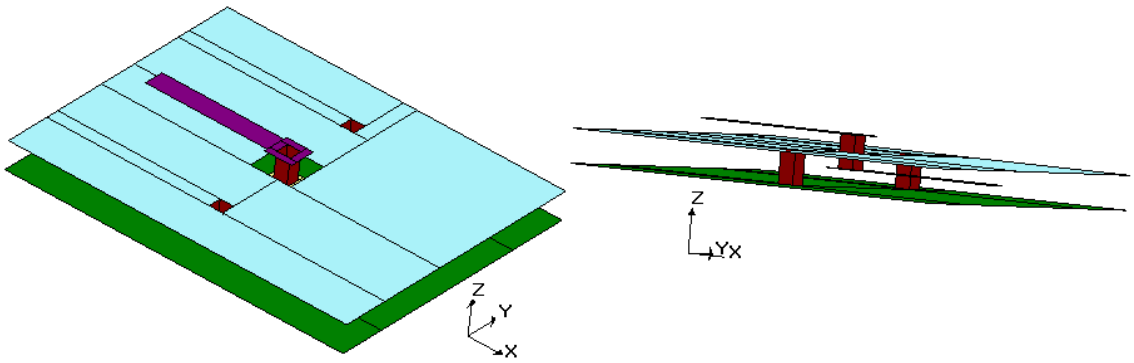


Figure 6.35 The 3D view of the imported structure imported and processed.

Step 9 Repeat Steps 1 and 2. Then, please select Load button on the Template File section. Select the file: .\ie3d\practice\vias_for_layers2.ctp file we just created. MGRID will set the layers and have the automatic vias procedures automatically defined.

Step 10 Select OK and MGRID imports all the polygons on different layers. Then, it will automatically build the holes and vias (see Figure 6.35 and Figure 6.36).

Section 6.12 Defining Localized Ports on Finite Ground Plane Structures

So far, we have discussed how to define extension ports on structures with infinite ground planes (defined as high conductivity layers on the substrates). For structures without enough rooms for the port extensions, we should use localized ports. For structures with finite ground planes or ground planes defined by polygons, we need to use differential ports. A Vertical Localized port or a Horizontal Localized port is a differential port by itself. It requires a user to define the positive terminal (level or direction) and the

negative terminal. For extension ports, we need you to define one (or more) positive port and one (or more) negative port as a differential port. The differential port concept will be discussed more. For the stripline to microstrip transition with finite ground planes in Figures 6.35 and 6.36, if we want to define some ports on them, we may want to use Vertical Localized ports.

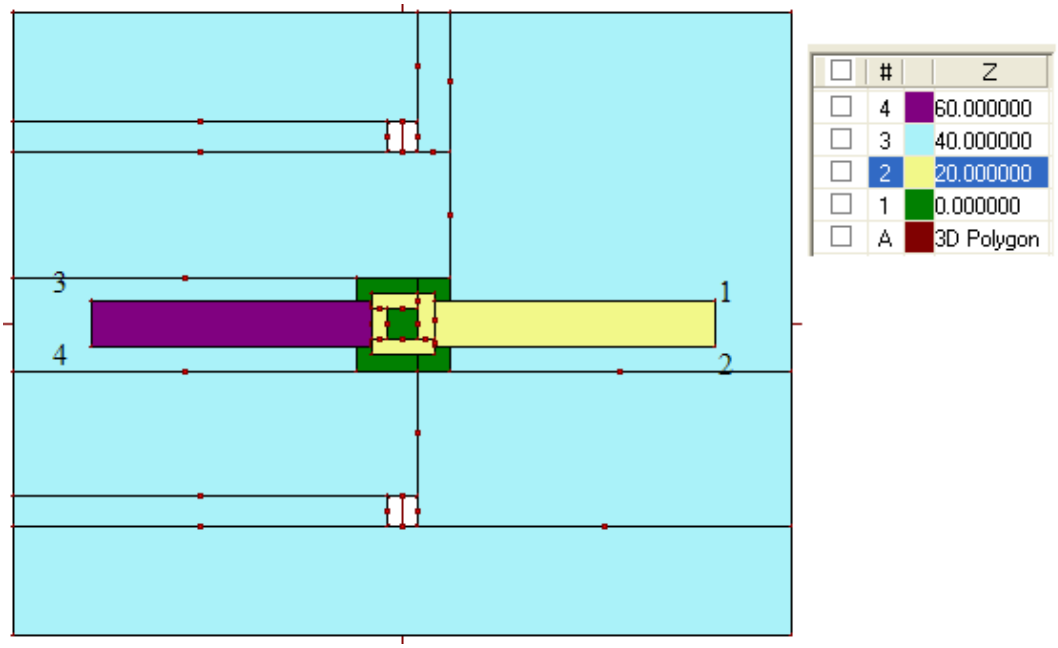


Figure 6.36 The traces on Z = 20 mils and on Z = 40 mils.

Defining a Vertical Localized port may be tedious. We will discuss the basic procedure to define a Vertical Localized port in later chapters. For this example, we will demonstrate a fast way to define Vertical Localized ports on the stripline and the microstrip traces.

Step 1 Please click at the Z = 20 mils layer to focus input on Z = 20 mils (see Figure 6.36). Please select **Edit->Select Vertices** command. Please make sure only the No. 2 Layer at Z = 20 mils is checked. Window the vertices 1 and 2 in Figure 6.36 to select the edge on Z = 20 mils.

Step 2 Select **Adv Edit->Build Via Connection on Edges** command. The Build Via and Port on Edges dialog comes up (see Figure 6.37).

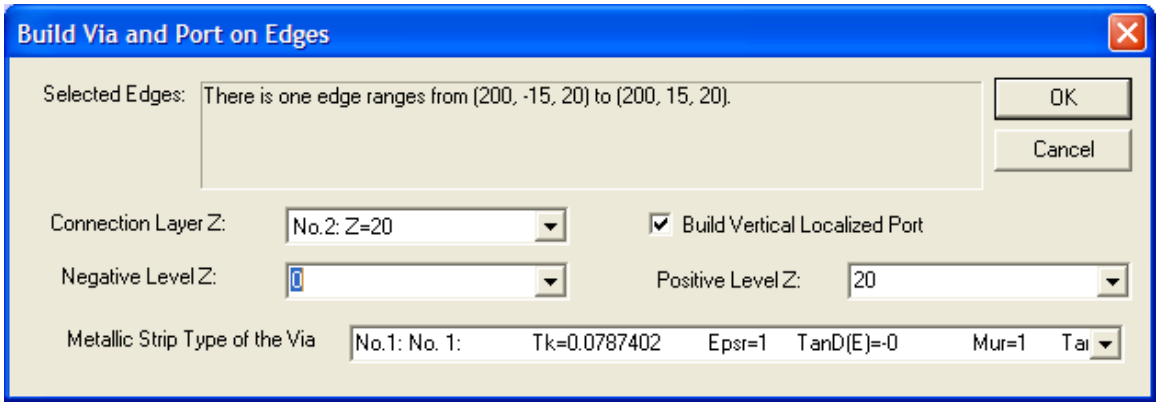


Figure 6.37 The Build Via and Port on Edges dialog.

Step 3 Please select Connection Layer Z as: No.1: Z = 0. Please select Negative Level Z as: 0. Please select Positive Level Z as: 20. Please check the Build Vertical Localized Port check box. We will get the exact picture shown in Figure 6.37.

Step 4 Please select OK. The port 1 is built at the end of the stripline trace. It is referencing the finite ground at Z = 0.

Step 5 Please click on the Z = 60 mils layer to focus input on Z = 60. Please select **Edit->Select Vertices** command. Make sure only the No. 4 Layer at Z = 60 mils is checked. Please window the vertices 3 and 4 to select the edge on Z = 60 mils in Figure 6.36.

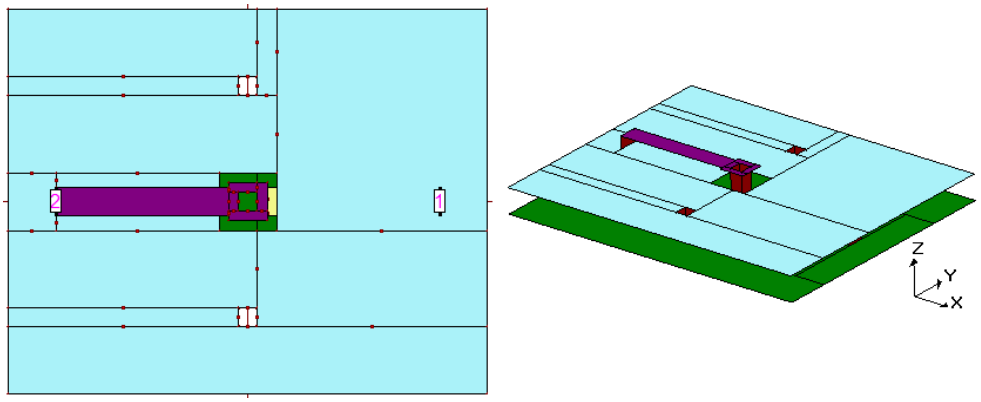


Figure 6.38 The PCB with ports built.

Step 6 Please select **Adv Edit->Build Via Connection on Edges** command to bring up the dialog. Select Connection Layer Z as: No.3 Z = 40, Negative Level Z as: 40, and Positive Level Z as: 60. Please check the Build Vertical Localized Port check box. Select OK to continue. The port 2 is defined on the edge of the mictrostirp trace. It is referencing the finite ground on Z = 40. We get the structure with ports shown in Figure 6.38. You can see some vertical rectangles at where the ports are defined in the 3D View window. They look like shorting circuit the trace. They are used to identify the references (or the return paths) of the ports. They are not short circuit in a real simulation and you do not need to worry about them. Please save the geometry as: .\ie3d\practice\vias_on_layers2.geo. It is a structure ready for simulation. The simulation over a wide band should take minutes using AIF.

The above example demonstrated how we can simplify importing procedure using an import template. The import template is just the starting point only. Based upon the idea of import template, we have implemented the Automatic Geometry to IE3D Flow (AGIF). AGIF has many advanced importing capabilities. One of them is the Automatic GDSII to IE3D Flow. It allows streamlined IE3D EM simulations directly from a batch of GDSII files. You don't need to build the vias and ports manually. They can be completely automated. We will discuss AGIF later in this chapter.

Section 6.13 Modeling Spirals and Other RFIC Circuits

IE3D is not only good for MMIC, PCB and antenna applications, it is excellent for RFIC design. It is being used by many leading RFIC companies for their daily designs. RFIC involves lossy and multiple thin substrates. Numerically, they are extremely difficult to model. Also, the trace thickness is no longer small compared to trace width. Metallic thickness may be very critical to accuracy. We will need to use the Thick Strip model instead the Thin Strip model for it. Please read the Appendix S for more detail on Thick Strip model. IE3D's Thick Strip model provides extremely accurate results on structures with thickness. We will

demonstrate how we can do it on modeling RFIC structures in this section. We will use a simple spiral inductor to demonstrate the concepts.

Step 1 Please select File->Import on MGRID. Select GDSII file format. Select OK to continue. Select .\ie3d\samples\spiral.gds file. The Import Data Options dialog comes up. Select Load button in the Import Template section. Select .\ie3d\samples\rfic.ctp file. MGRID loads the data from the file and set the Basic Substrate and the layers for the imported structure. Select OK. An octagonal spiral inductor is imported. They are two polygons on Z = 275, and one polygon on Z = 277. We need to build the bridge based upon the polygon on Z = 277.

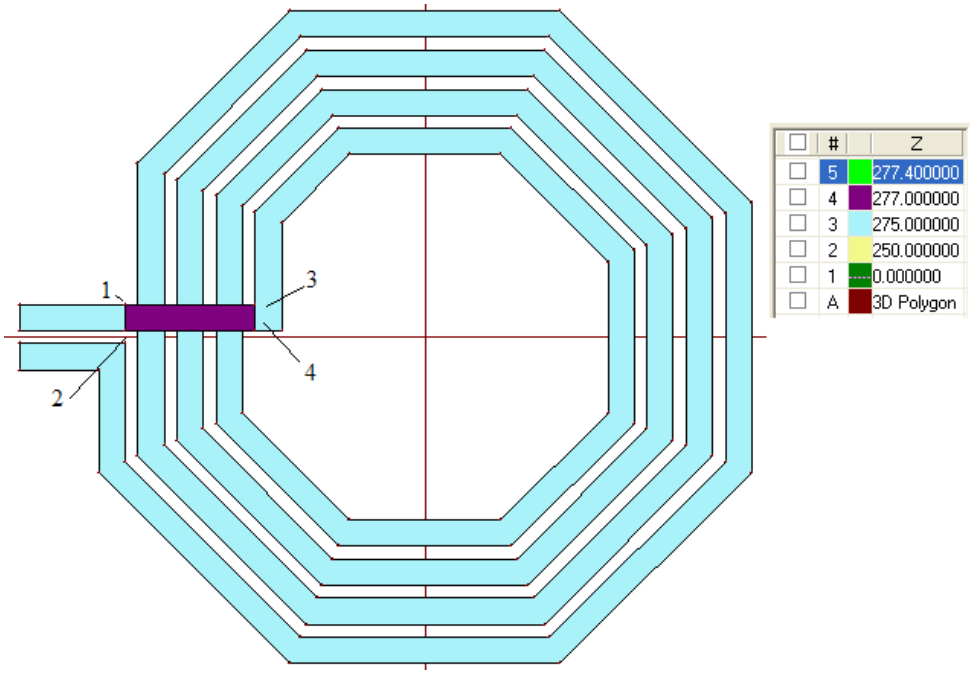


Figure 6.39 An octagonal spiral inductor imported.

Step 2 Please select Edit->Select Vertices. Check the No.4 Layer at Z = 277. Window the vertices 1 and 2 to select them. Select Edit->Add Via on Edges command. Enter the End Z-Coordinate = 275. Select OK. A vertical polygon is built for it.

Step 3 Please select Edit->Select Vertices. Check the No.4 Layer at Z = 277. Window the vertices 3 and 4 to select them. Select Edit->Add Via on Edges command. Enter the End Z-Coordinate = 275. Select OK. Another vertical polygon is built to finish the connection. The 3D View is shown in Figure 6.40.

Since the Edit->Add Via on Edges command can be used to add multiple vias together, you may wonder whether we can select the vertices 1 and 2, and vertices 3 and 4 together to build the two vertical rectangles simultaneously. In fact, you can do it and you will be able to build the two vertical rectangles, in additional to two more rectangles simultaneously. The reason is that MGRID will consider 4 edges, edges (1,2), edge (2,3), edge (3,4) and edge (4,1), are selected instead of two edges. It is not what we want.

After the connections are built, you can select one of the polygons and use the Adv Edit->Connection->Check Connection to make sure the polygons are truly connected. It should be connected.

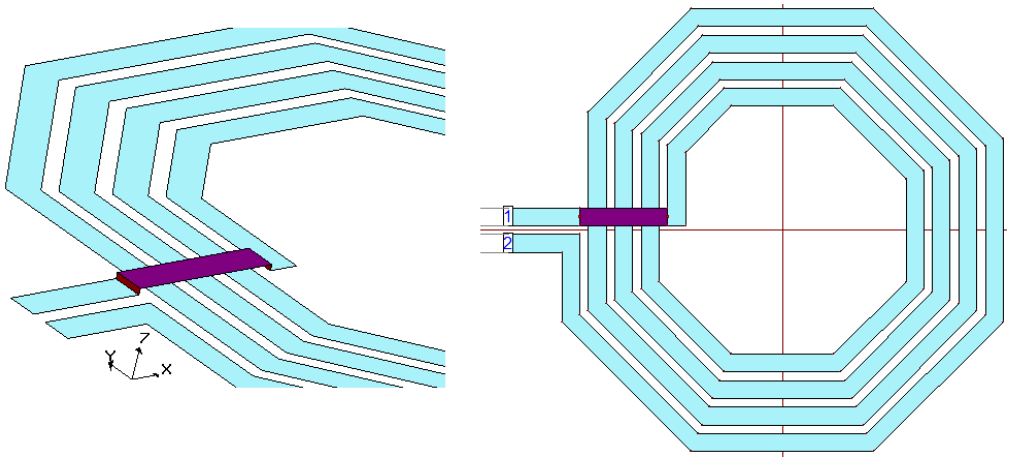
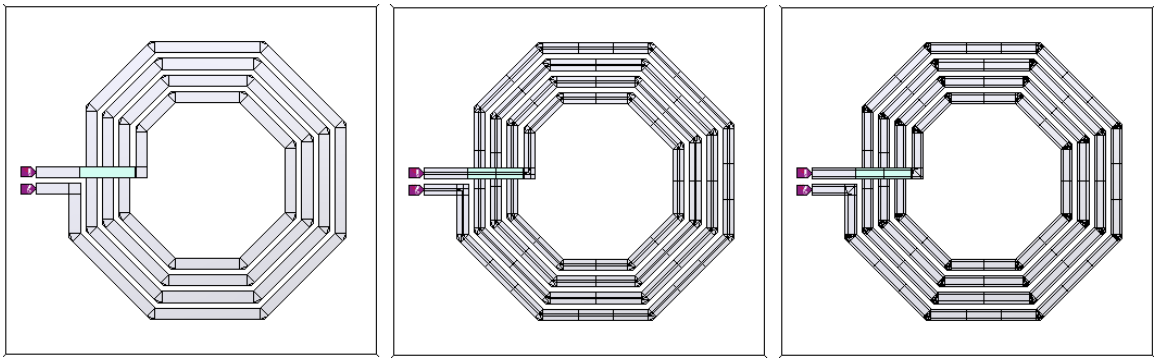


Figure 6.40 The vertical polygons for the bridge and the complete spiral with ports



(a) Fmax=15, Ncells=15, AEC Ratio=0.1 (b) Fmax=200, Ncells=15, AEC Ratio=0.1 (c) Fmax=200, Ncells=15, AEC Ratio=0.05

Figure 6.41 The different meshings for the spiral inductor.

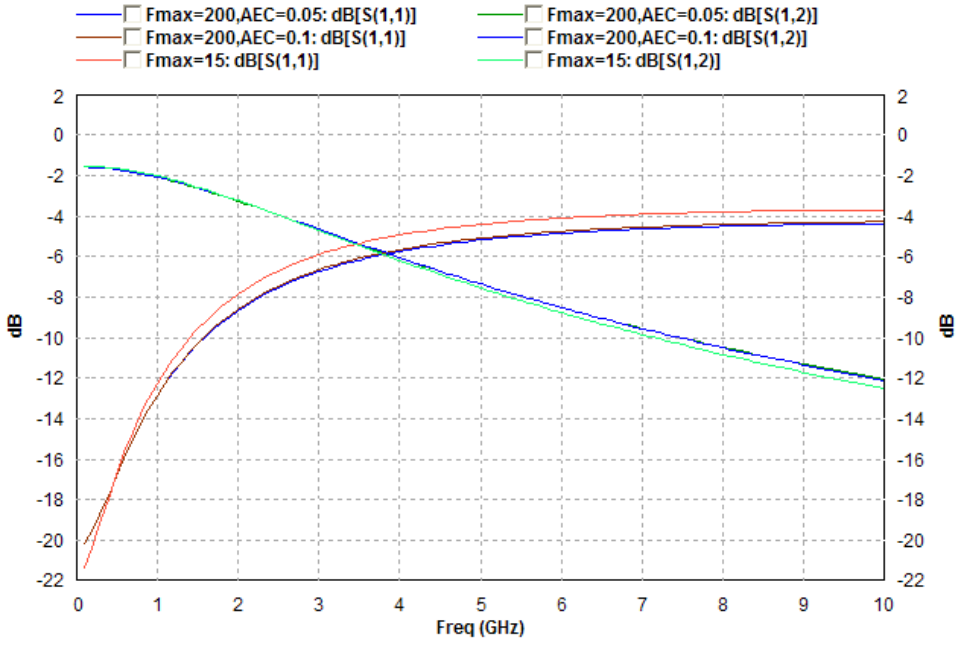


Figure 6.42 The simulation results and comparisons.

Step 4 Define two Advanced Extension ports on the two feed lines and save it as .\ie3d\practice\spiral1.geo. On IE3D 14, you may see the some symbol "(0)" at the lower right corner of the port index on both port 1 and port 2. It is an indication that port 1 and port 2 are coupled group 0. We will discuss coupled ports later.

Is it ready to simulation? Yes. However, you do need to pay attention to the meshing. Please do a meshing and see how the meshing looks like. Using Fmax = 15 GHz and Ncells = 15 cells/lambda, and AEC Ratio = 0.1, we get the meshing shown in Figure 6.41a. No edge cells are created. If we go into Process->Display Meshing, we can see the AEC Width is 12.028 microns with the AEC Ratio = 0.1. Basically, the trace is very narrow and it is small compared to regular cells size. We should reduce the AEC Width for it by reducing the AEC Ratio. However, we would like to use other way. A spiral in an RFIC is electrical small structure. However, the current changes direction very rapidly. Using the regular meshing size as shown in Figure 6.41a, the simulation results may not be accurate enough. Please pay attention to the outer segment shown in Figure 6.41a. It is only one cell and it means that the regular cell size may be much bigger than its size. Please increase the Fmax = 200 GHz while keep Ncells = 15. You will see the most inner segments start or are about to break into 2 cells in the longitudinal direction. By this time, the meshing should be good (see Figure 6.41b). You may notice the edge cells are relatively wide compared to the interior cells. You can reduce the AEC Ratio to 0.05 and you will get the meshing shown in Figure 6.41c. Simulation from 0.1 to 10 GHz for 100 frequency points should take couple minutes using AIF. The results and comparison are shown in Figure 6.42. As you can see, the difference between the two Fmax = 200 GHz cases is invisible. There is some slight difference in S(1,1) for the Fmax = 15 GHz case. S-parameters are the less insensitive parameters for RFIC modeling. We may want to see how the L, R and Q values compare.

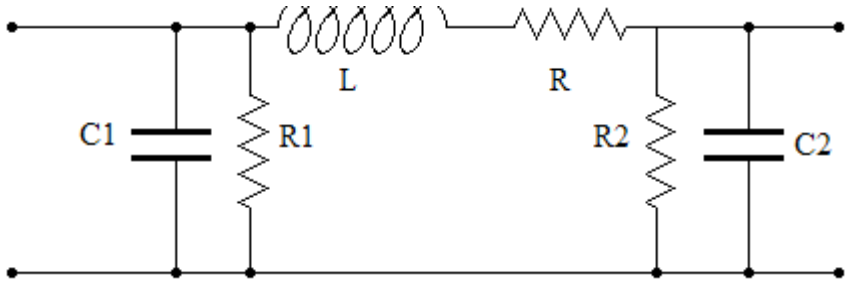


Figure 6.43 The PI-network equivalent circuit for "2-Port PI with Series RL and Shunt RC".

Section 6.14 Finding PI-Network, L and Q-Values

In older IE3D versions, we have to use the Process->PI-Network Equivalent command on MODUA to find the R, L and Q values of the spiral inductor. We did not have any utility to display them and we had to use Excel or other tools to display them on graph. On IE3D 14, the results are readily available on the Process->S-Parameters and Lumped Equivalent Circuit command. Just select the "2-Port PI with Series RL and Shunt RC" (see Figure 6.44). Then, you can select Add Graph to define some graphs displaying R, L and Q.

The L-values from different meshing schemes are compared in Figure 6.44 while the R and Q are compared in Figure 6.45. As you can see, the Fmax=200 GHz cases agree very well while the Fmax = 15 GHz case is a little bit off. The case with Fmax = 200 GHz and AEC = 0.05 have better edge cells. It is supposed to be more accurate.

Please note that there are different definitions for the equivalent circuit. For example, some people may define the R, L and Q as: $R = - 1 / Re(Y12)$, $L = - j / [\omega Im(Y12))$ and $Q = Im(Y12) / Re(Y12)$. On

IE3D, you have different options. You can define the R, L and Q based upon Figure 6.43 with $Q = \omega L / R$. You can also convert the 2-port s-parameters into 1-port and define the R, L and Q based upon the 1-port (or 2-terminal) series R and L model.

On the s-parameters processing dialog (Figure 6.44), you can select s-parameters data in the list box and select Save Model to save the R, L and Q values into an ASCII file. Table 6.7 shows a sample of it.

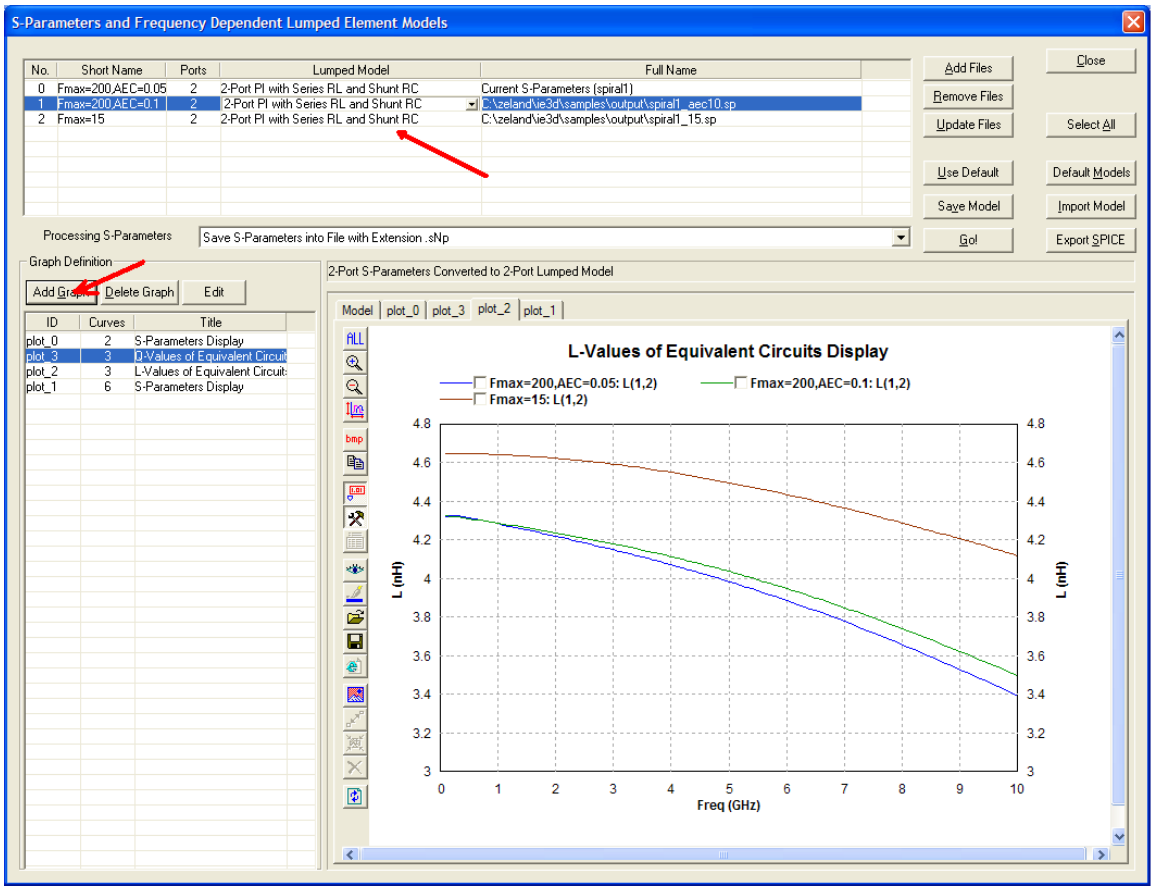
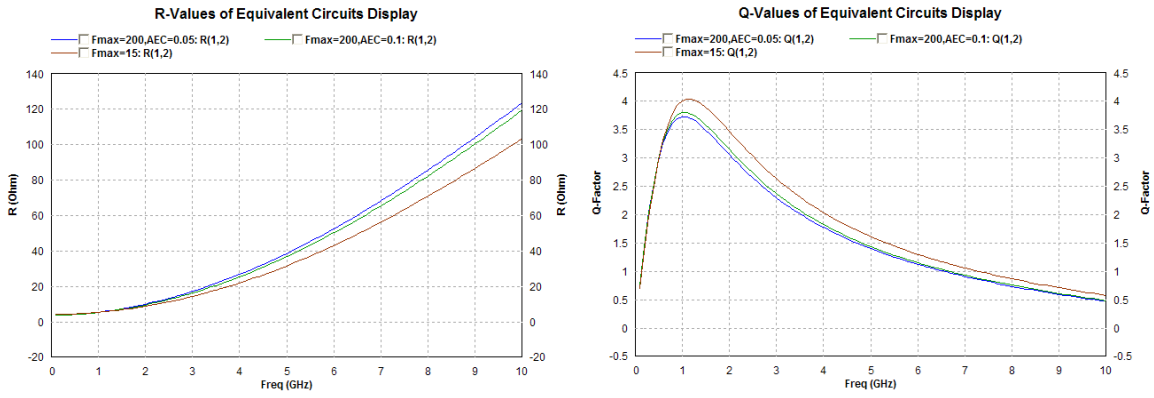


Figure 6.44 The L value of the PI-network with different meshing schemes.



(a) R (b) Q

Figure 6.45 The comparisons on R and Q between different meshing schemes

Table 6.7 The PI-equivalent circuit of the spiral simulated in .\ie3d\practice\spiral1.geo.

```
Current S-Parameters (spiral1)
Fmax=200,AEC=0.05
!The 1st line is the s-parameter file name if it is available.
!The 2nd line is the comment identifying the model.
!Any line starts with exclamation is a comment of the file and it will be discarded in parsing.
!
!IE3D Frequency-Dependent Lumped Model File
!IE3D File Type Number   Version    Model Type
  6830                  12.00        10
!Model Type is a unique number to denote the type of equivalent circuit.
!
!Model Type Name: 2-Port PI with Series RL and Shunt RC
!Port Number is the s-parameters port number. Final Port Number is the equiv ckt port number.
!Port Number   Final Port Number    Precise Model
 2              2             1
!
!Warning: Please understand that an equivalent ckt is just a fitted model. It is based upon
!       your selection of the model you want to fit the s-parameters into. The values of
!       of the ckt elements may or may not have or match any physical meaning.
!
!
!   +1 o--------R----L----------o +2
!       | |      | |
!       R1  C1      R2  C2
!       | |      | |
!   -1 o------------------------o -2
!
! Q = - Im[Y(1,1)]/Re[Y(1,1)]
! R, L and Q are also referenced as R(1,2), L(1,2) and Q(1,2).
! R1 and C1 are also referenced as R(1,1) and C(1,1).
! R2 and C2 are also referenced as R(2,2) and C(2,2).
!
!     Freq(GHz)          Q                L(nH)            R(Ohm)           C1(pF)           R1(Ohm)          C2(pF)           R2(Ohm)
!
 1.0000000000e-001  7.3780896936e-001  4.3209333742e+000  3.6285341532e+000 -5.3360178889e-002  3.9412225663e+002  9.9215002933e-002  2.8234154099e+002
 2.0000000000e-001  1.4276489037e+000  4.3265578100e+000  3.7004578143e+000 -1.3623883536e-002  3.9916701826e+002  6.1206256881e-002  2.7981183719e+002
 3.0000000000e-001  2.0349909078e+000  4.3238568884e+000  3.8032906979e+000  2.8088889461e-003  4.0131505969e+002  4.4962796786e-002  2.7873728373e+002
 4.0000000000e-001  2.5425982816e+000  4.3191327246e+000  3.9365441674e+000  1.0113578634e-002  4.0227487619e+002  3.7508618084e-002  2.7822843097e+002
 5.0000000000e-001  2.9457406370e+000  4.3136378014e+000  4.0997722449e+000  1.3803949326e-002  4.0275504876e+002  3.3572471946e-002  2.7794180219e+002
 6.0000000000e-001  3.2493656822e+000  4.3077828340e+000  4.2925681202e+000  1.5869818131e-002  4.0301861957e+002  3.1244749012e-002  2.7775559306e+002
 7.0000000000e-001  3.4644408835e+000  4.3017379479e+000  4.5145598021e+000  1.7121309514e-002  4.0317465265e+002  2.9747694709e-002  2.7762096414e+002
 8.0000000000e-001  3.6046597127e+000  4.2955830907e+000  4.7654053166e+000  1.7929237007e-002  4.0327323002e+002  2.8724055964e-002  2.7751555162e+002
 9.0000000000e-001  3.6840589578e+000  4.2893582985e+000  5.0447878658e+000  1.8479945086e-002  4.0333884663e+002  2.7992224579e-002  2.7742769909e+002
 1.0000000000e+000  3.7155917127e+000  4.2830839143e+000  5.3524113018e+000  1.8874032028e-002  4.0338369214e+002  2.7451753958e-002  2.7735051049e+002
 1.1000000000e+000  3.7104536428e+000  4.2767698582e+000  5.6879961796e+000  1.9168834996e-002  4.0341359610e+002  2.7042986298e-002  2.7727940992e+002
 1.2000000000e+000  3.6779053964e+000  4.2704203078e+000  6.0512765400e+000  1.9398397124e-002  4.0343097980e+002  2.6728271311e-002  2.7721107960e+002
 1.3000000000e+000  3.6253743664e+000  4.2640362331e+000  6.4419974737e+000  1.9583563237e-002  4.0343645119e+002  2.6482590984e-002  2.7714297840e+002
 1.4000000000e+000  3.5586854764e+000  4.2576168344e+000  6.8599134267e+000  1.9737402557e-002  4.0342971969e+002  2.6288592239e-002  2.7707311492e+002
 1.5000000000e+000  3.4823301179e+000  4.2511603777e+000  7.3047871466e+000  1.9868269321e-002  4.0341013762e+002  2.6133814485e-002  2.7699993320e+002
 ....
```

Section 6.15 Modeling True Thickness of the Spiral Inductor

Step 1 Run MGRID. Open .\ie3d\practice\spiral1.geo. Please select Parameters->Basic Parameters to see the strip thickness. It is 1.5 microns. Select Cancel. Click at the No. 3 Layer at Z = 275 to focus on the layer. Select Edit->Layer->Grow Thickness on Layer. The dialog comes up (see Figure 6.46). Make sure the Thickness Growing Direction is Positive Z-Direction and the Ports on All Edges are checked. Select OK to accept the default setting of 2nd Degree Compensation with Pre-Processing. This is the best setting for accurate modeling of loss. The Positive Z-Direction for the Thickness Growing Direction means that we want the thickness spans from Z = 275 to Z = 276.5. The Ports on All Edges will guarantee an extension port will be on all the edges of a thick trace for high accuracy results.

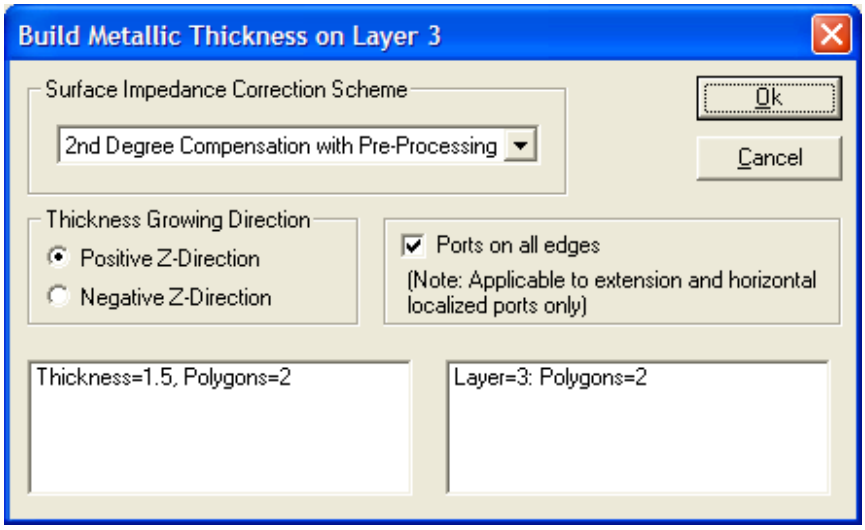


Figure 6.46 The Grow Thickness on Layer dialog.

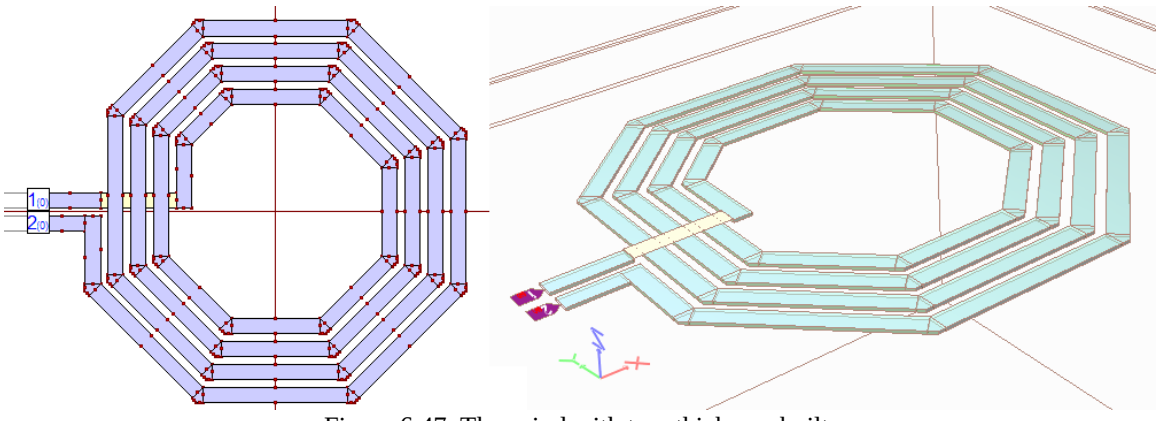


Figure 6.47 The spiral with true thickness built.

Step 2 Select OK. MGRID will prompt you for removing the existing s-parameters. Please select YES because the results are saved. The thickness is built. Please save the file as: .\ie3d\practice\spiral1_tk.geo. The top view and the 3D view are shown in Figure 6.47. You can see the connection markers on the supposed open edges in the top view and the true thickness in the 3D view. The current on the four sides of the trace will be accurately modeled. At low frequency, current is not concentrated on the surface. You may wonder whether the model is

accurate enough at low frequency. In fact, the model is good from very low frequency to very high frequency. More explanation is included in Appendix S on modeling thickness.

Simulation on spiral1_tk.geo takes couple minutes only. A comparison in the s-parameters between the Thin Model (spiral1.geo) and Thickness model (spiral1_tk.geo). is shown in Figure 6.48. There is some slight difference. Figure 6.49 shows the comparison in the Q-values. There is about 5% difference in the L value while the differences in R and Q are smaller. The difference in the calculated L is expected because the thin model is an approximation while the thick model is closer to the reality. Both models should be accurate results while the Thickness model should be more accurate.

You may notice that the meshing for the Thickness model does not have edge cells. This is due to the fact that the vertical polygons along the edges already model the edge effects very well. There is no need to use edge cells unless very high accuracy is required. More discussion can be found in Appendix S.

This spiral does not have very high Q-value and it should not be a good design. We use this example to demonstrate to you how to accurately and efficiently simulate spiral and transformer circuits in RFIC. How to improve the Q-value is beyond the topics of this manual.

All the examples we demonstrated here are simple ones. It takes minutes to simulate. More complicated structures can be simulated in the same way. IE3D is the most accurate, efficient and capable EM simulator for MMIC, RFIC and PCB applications. You can use it to solve much tougher EM problems.

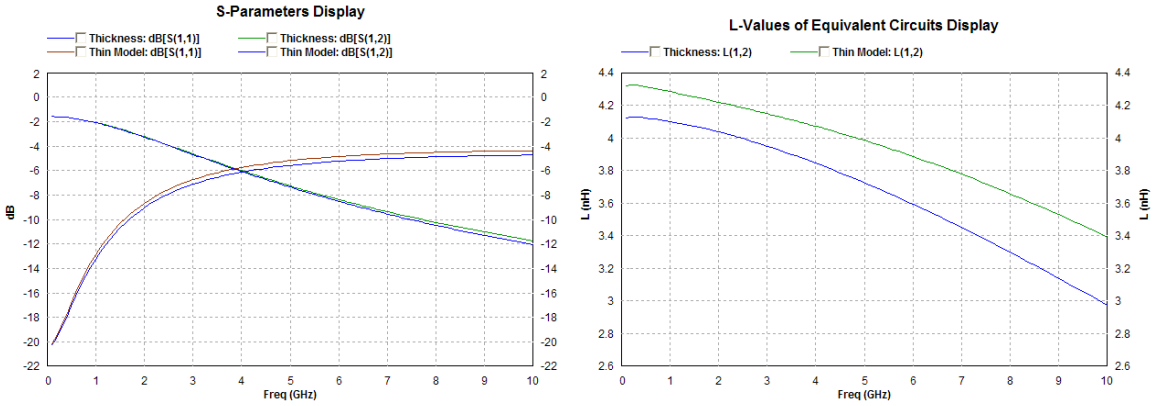


Figure 6.48 The comparison in s-parameters between Thin Strip and Thick Strip models.

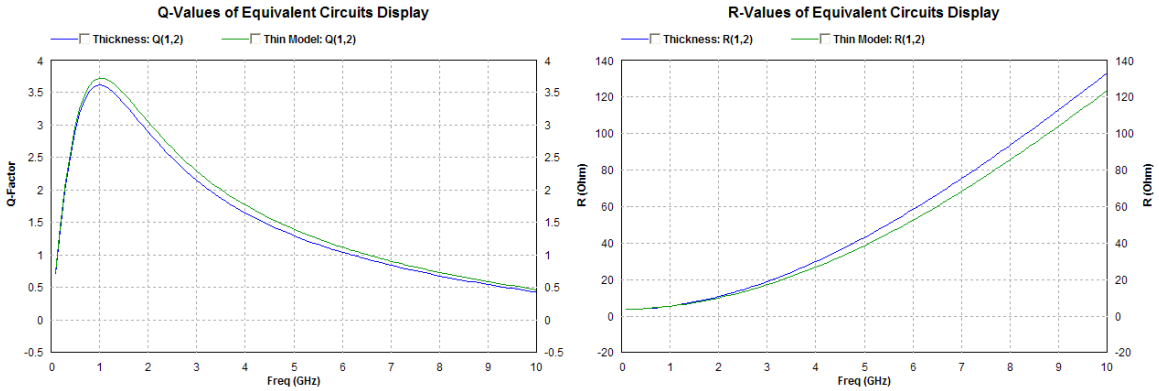


Figure 6.49 The comparison in Q-values between Thin Strip and Thick Strip models.

Section 6.16 Automatic Run-Time Thickness

In the above example, we have demonstrated how we can use thickness model for structures with thick traces. Thickness model can include all the effects of thickness precisely. It should be the model we should use when the strip thickness is no longer much smaller than strip width and high accuracy is required. However, as you can see, the thickness model creates many more polygons. Further editing of the structure will be much more difficult. We suggest users to keep the thin model before "growing" thickness so that you can do the editing based upon the thin model and re-grow the thickness after change.

Another option is to use the Automatic Run-Time Thickness (ARTT). ARTT allows you to keep the thin strip model while it will automatically create the thickness model in run-time. It allows you to do editing on the thin model while you do the simulation on the thickness model transparently. Let's use the spiral structure as our example to demonstrate the ARTT feature.

Step 1 Run MGRID. Open .\ie3d\practice\spiral1.geo. Select Parameters->Basic Parameters to check the strip thickness. It is 1.5 microns. The spiral is on Z = 275 microns. Please select the Insert button in the Automatic Run Time Thickness group (see Figure 6.50). MGRID will prompt you for the z-coordinate where you want to create thickness (see Figure 6.51). You also have the choice to choose the thickness growing in +z or –z direction. Enter the Layer Z-Coordinate as "275". Choose the option of Positive Z-Direction because we want the thickness growing in +z. Select OK to close the dialog. Select OK to close the Basic Parameters dialog. MGRID may prompt you for multiple thin layers. You should select not to merge the thin layers. Merging thin layers normally can make the simulation faster but a little accuracy decrease might occur. For this example, the simulation is quite fast and you don't need to worry about it.

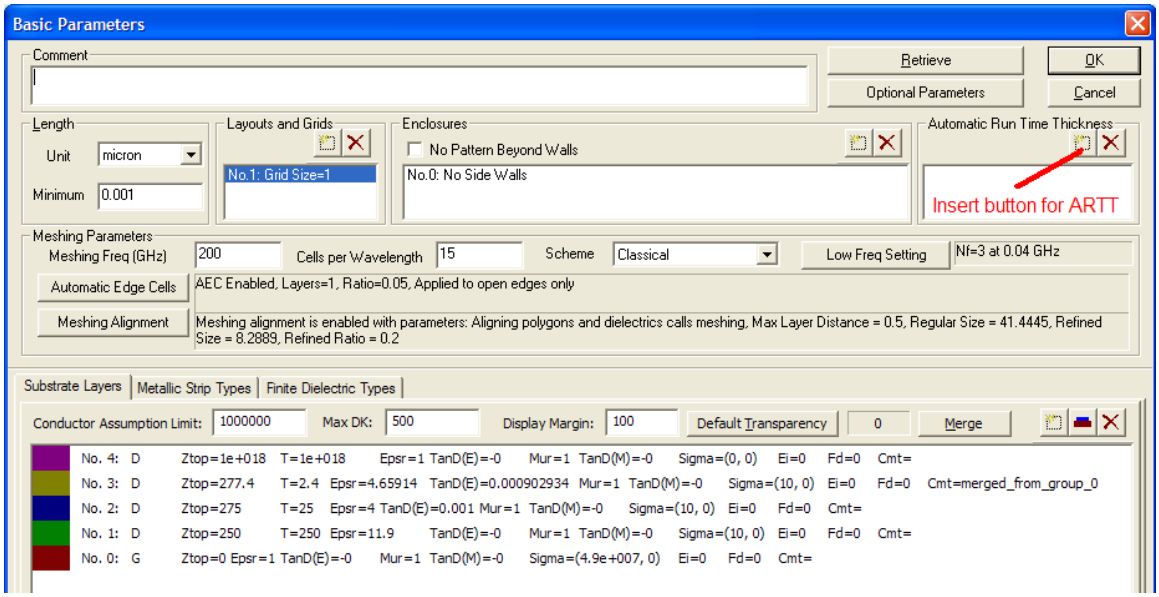


Figure 6.50 The Insert button of ARTT in Basic Parameters.

Step 2 Save the file as: spiral1_artt.geo. The structure looks exactly like the thin model. However, the structure being simulated should be like the thickness model. How can you know it is like the thickness model?

Step 3 Please select Process->Create Run-Time Thickness Model. MGRID will prompt you for the new geometry file name. The default is: derived.geo. Select Save and MGRID will create the

thickness model into derived.geo and open another instance of MGRID for it. The derived.geo will be a thickness model with polygons for the 4 sides of the traces. If you check the Basic Parameters, you will see the ARTT list box is empty in derived.geo. Certainly, ARTT is no longer required and should not be defined for the thickness model. The derived.geo is the exact structure you simulate if you select Process->Simulate on sprial1_artt.geo. If you simulate it and save the current distribution, you will also see the thickness on the spiral1_artt.cur file.

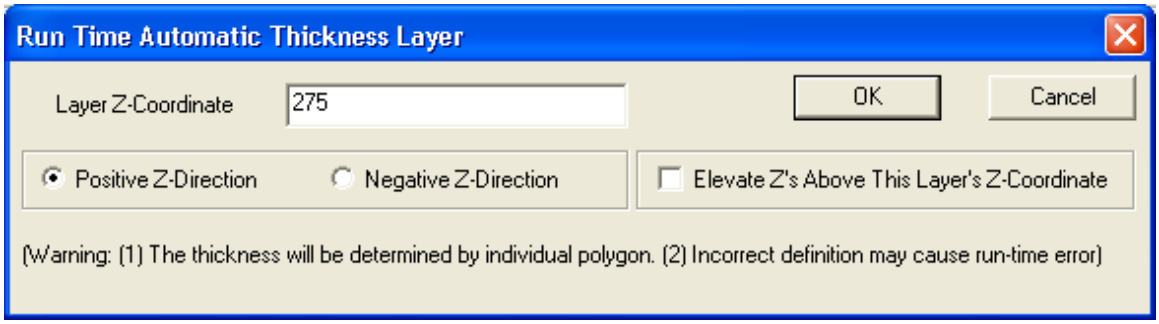


Figure 6.51 The ARTT dialog inside Basic Parameters dialog.

Section 6.17 Automatic Geometry to IE3D Flow for GDSII and Cadence Virtuoso

Automatic Geometry to IE3D Flow (AGIF) contains 3 parts. The 1st part is Automatic GDSII to IE3D Flow and the name AGIF comes from this part in early 2005 when we first introduced AGIF. AGIF is part of IE3D-SI.

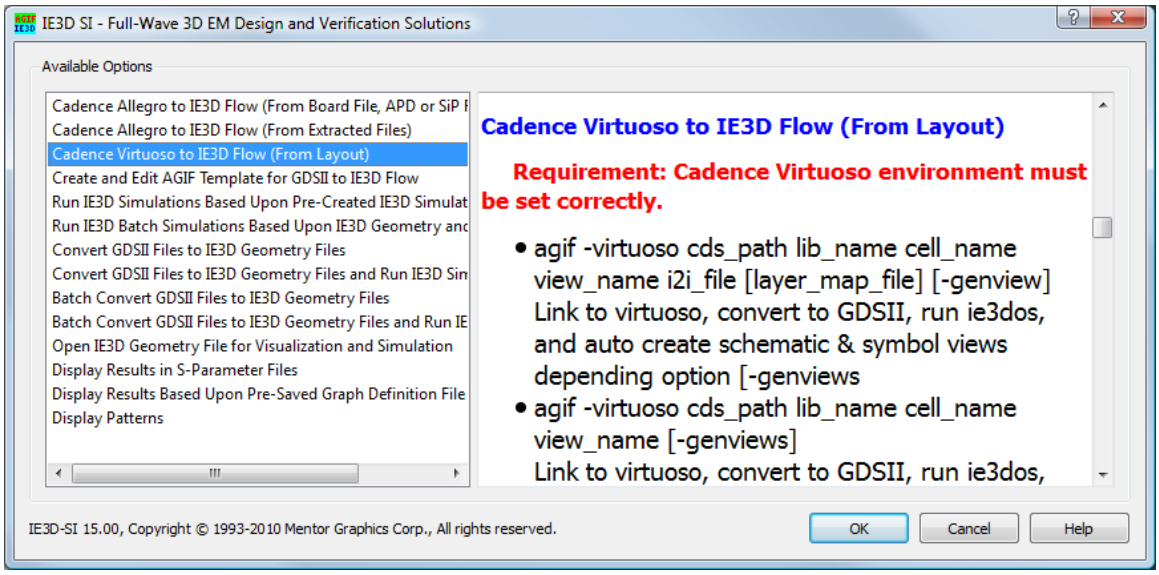


Figure 6.52 The AGIF main dialog.

AGIF is completely documented in the AGIF User's Manual in the installation. We will just try to demonstrate how powerful such a feature is when it is being used in the design of MMIC and RFIC.

You can start AGIF from IE3DPM or from FILE->Automatic Geometry to IE3D Flow on MGRID. You will get the starting dialog shown in Figure 6.52. You can do many jobs using AGIF besides of the

automatic conversion feature. You can display s-parameters, radiation patterns. You can do a batch simulation with pre-created .geo files and/or .sim files. These functionalities are available to all IE3D users, while the automatic conversion features are available to IE3D MM08X users. AGIF is available on both Windows and Linux. In some sense, AGIF allows you to perform IE3D simulations directly from GDSII or Cadence layouts on Linux.

Step 1 Please run AGIF and select Create and Edit AGIF Template for GDSII to IE3D Flow. Select OK. The Automatic GDSII to IE3D Flow dialog comes up (see Figure 6.53).

Step 2 Select Load Template File button. Select .\ie3d\agif\for_two_port_spirals.i2i file. It is a pre-defined AGIF template file. It contains a graphically defined script telling AGIF how to interpret the polygons of the different layers. We will not discuss the detail definition process here. Interested users can refer the AGIF User's Manual (AGIF.PDF) in the installation. The main idea is that the template file contains some Layer Self Operations and Layer Mutual Operations. Each layer has a Layer Self Operation on which z-coordinate we put them, whether the layer polygons are used, how you would like us to clean the polygons on the layer. Each Layer Mutual Operation contains information how we want to use the polygons on one layer to define vias and ports, etc on other layers. The AGIF template records the procedure we should take manual for importing a geometry file. The recorded procedure can be applied to many different GDSII files of the same process.

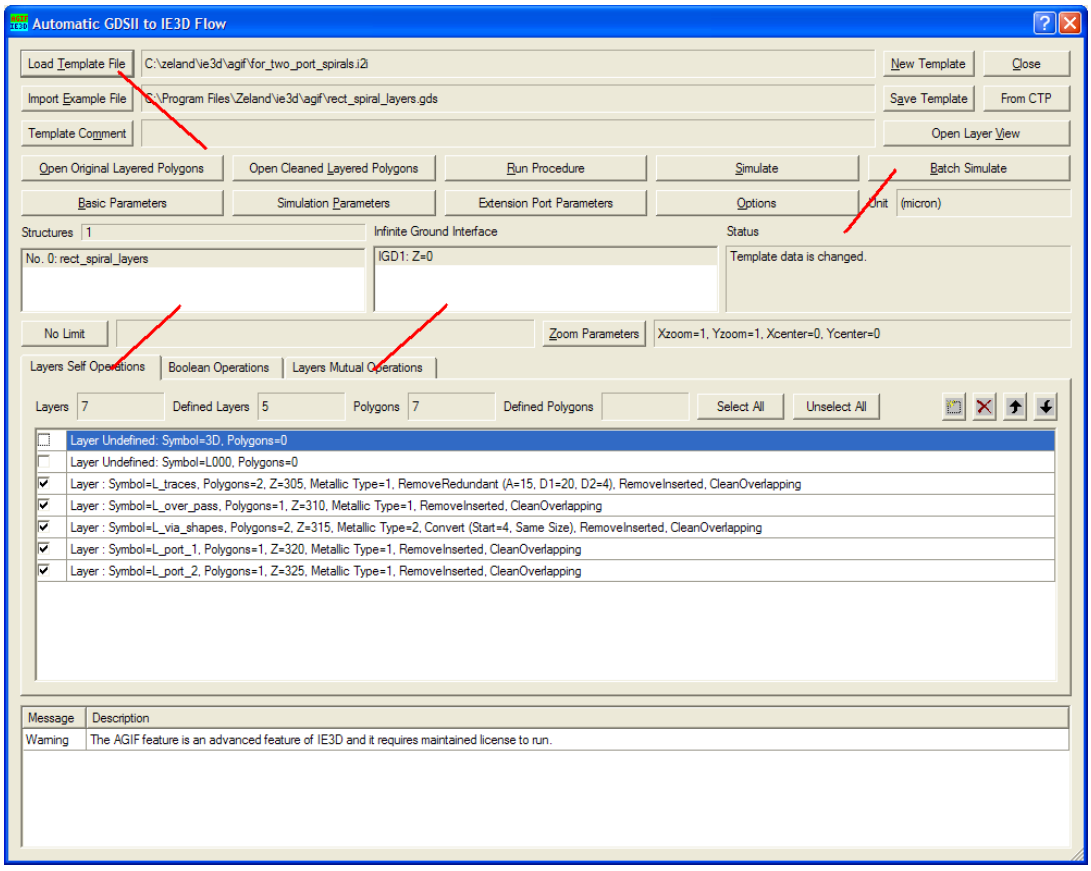


Figure 6.53 The Automatic GDSII to IE3D Flow dialog.

Step 3 Please select Batch Simulate button. The Automatic GDSII to IE3D Flow Batch Simulation dialog comes up (see Figure 6.54). You can see 4 GDSII files are in the list. Click at the four

files sequentially while you hold the Shift button in the keyboard to select them in the list. Select Open Layer View menu. Four 2D Views are opened with each showing the layered polygons in each GDSII file. As you can see, they are of different shapes but the same layer configuration.

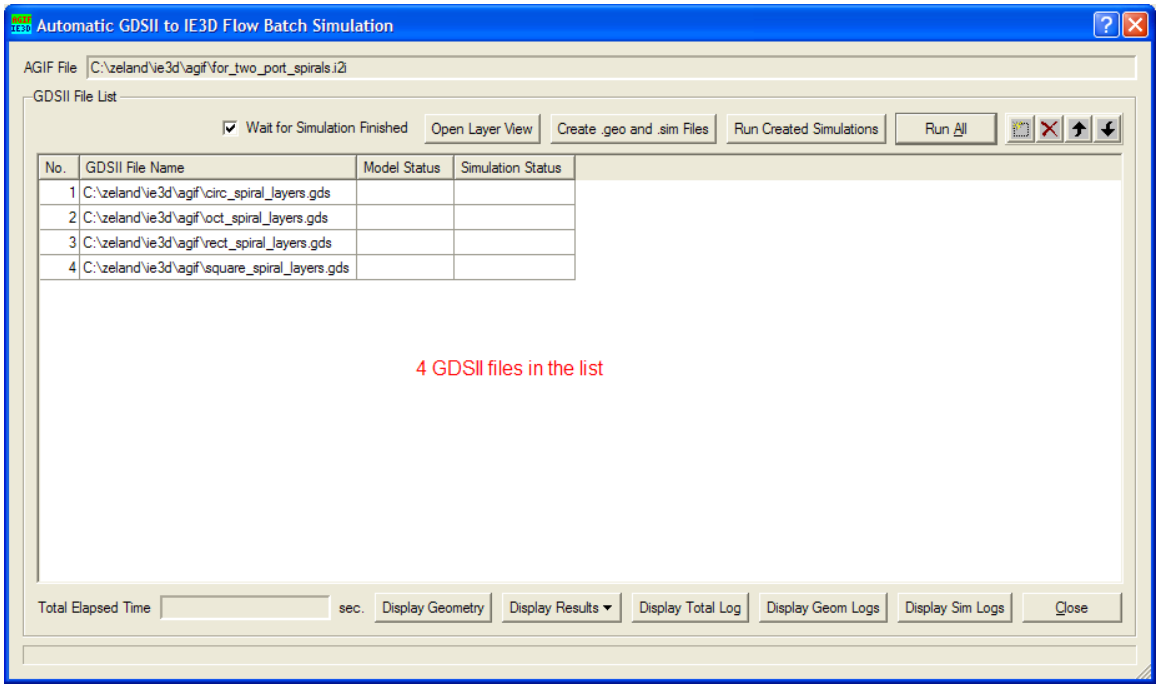


Figure 6.54 The Automatic GDSII to IE3E Flow Batch Simulation dialog.

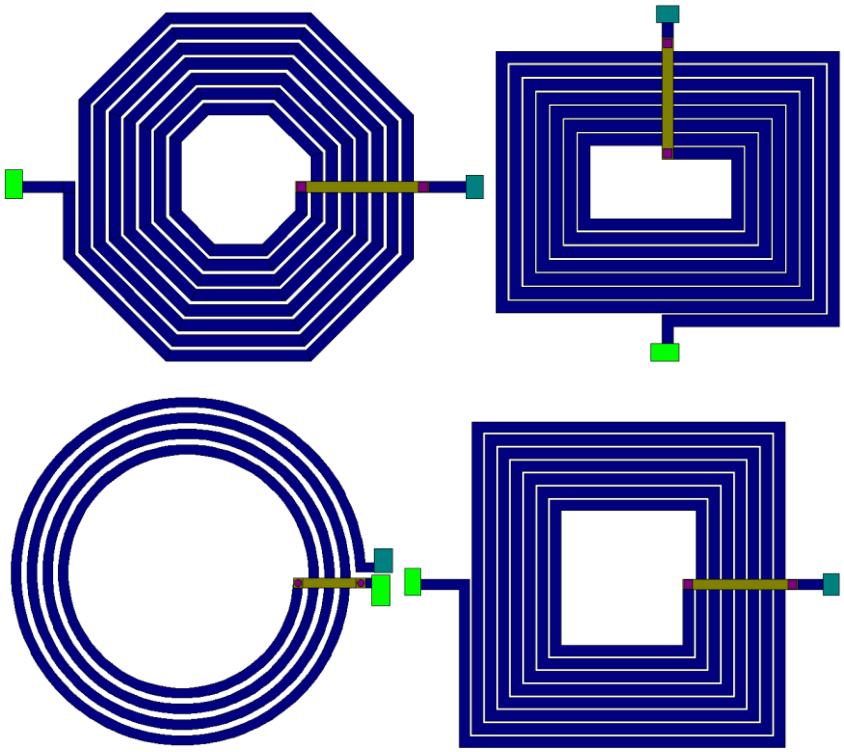


Figure 6.55 Four different structures from four different GDSII files of the same layer configuration.

Step 4 Please close the four 2D view dialogs to go back to the Automatic GDSII to IE3D Flow Batch Simulation dialog (see Figure 6.54). Please select the Run All button. AGIF will automatically convert the GDSII files into MGRID files and perform the IE3D EM simulations. After simulations, you can visualize the s-parameters dialog on AGIF or on MGRID.

AGIF is extremely powerful. It has implemented automatic merging of pin vias and other advanced features to model RFIC structures accurately and efficiently.

What does it mean to you? Basically, you can create one template to convert a class of structures defined in GDSII files into IE3D files and perform the IE3D simulations directly from GDSII files or Cadence Virtuoso. It can save you much effort. The key to the Automatic GDSII to IE3D Flow and the Cadence Virtuoso to IE3D Flow is to create the template files. AGIF works almost the same on Cadence Virtuoso. You are able to configure AGIF on the menu of Cadence Virtuoso. You can select a cell on Cadence Virtuoso and select IE3D from the menu. Then, you can pick a pre-built AGIF template or create a new AGIF template for the selected cell. Based upon the AGIF template, you can automatically invoke IE3D engine to convert the cell into IE3D geometry and simulate it. After simulation, the s-parameter results are sent back to Cadence Virtuoso as a symbol view and a schematic view. IE3D and AGIF are transparently integrated into the environment.

Section 6.18 Automatic Geometry to IE3D Flow for Cadence Allegro and APD

IE3D is not only good for MMIC and RFIC. It is also excellent on PCB and packaging analysis and designs. There are many topics in such applications: cross-talk, IR drop, signal integrity, power integrity. MCM, SoP, SiP. As a general can powerful full-wave EM tool, IE3D is capable of all the aspects of such applications. PCB and MCM structures are normally very complicated. They may contain vias, complicated traces, solder balls and wire bonds. The complexity is so big that it is almost impossible to draw the structures on any layout editor. They are normally described by nets.

AGIF allows you to import the nets and cross-section directly from a Cadence Allegro (.brd) file or APD (.mcm) file. You can just pick some nets of interests to you, and select some options. Select Build Full-Model and AGIF can build the IE3D full-wave geometry model for you automatically. The model may contain vias, solder balls, wire bonds and even ports. The automatically created model is ready for IE3D simulation. Interested users please read the AGIF User's Manual (AGIF.PDF) for more information on the Automatic GDSII to IE3D Flow, Cadence Virtuoso to IE3D Flow, and Cadence Allegro/APD to IE3D flow. We will end this chapter here.

Chapter 7 Mixed EM and Network Simulation and Optimization

We have discussed simulation and optimization of individual structure elements. In a complicated circuit, we may have many elements including distributed and lumped elements. These elements are coupled together. It would be good idea for us to simulate the whole circuit with all the coupling effect between elements taken into consideration. Certainly, we can include the coupling effects between the distributed elements, but we may not be able to include the coupling between the distributed elements and the lumped elements. Simulation of lumped elements such as diodes may involve other physics than the Maxwell's Equations. No accurate and efficient approach has been found for such modeling. A practical approach is to model lumped elements as s-parameter modules or VI curves. In this way, we can perform a mixed electromagnetic simulation on structures and network simulation on s-parameters on a complete circuit.

The key issue of successful complete circuit simulation is how we can extract the s-parameters of structures in a limited space and connect different s-parameters together. In some sense, we can always separate a circuit into the distributive part and the lumped element part. We simulate the distributive part and the lumped element part separately and connect them together on a nodal circuit simulator. The critical issue is whether an EM simulator can simulate the distributive part as one or multiple loosely coupled structures.

Some EM simulators cannot do a robust complete mixed EM and circuit co-simulation because they have various kinds of limitation, such as the limitation on the port locations (The ports are limited to the boundary of a circuit). When we have a lumped element inside a highly packed circuit, we must be able to define some ports to replace the lumped element. IE3D has localized de-embedding schemes. We are able to perform mixed EM and network co-simulation and co-optimization on highly packed structures with lumped elements. In order to better understand the issues discussed in this chapter, you are suggested to read the Section 4 of Chapter 12 for a better understanding on the de-embedding schemes used in the IE3D.

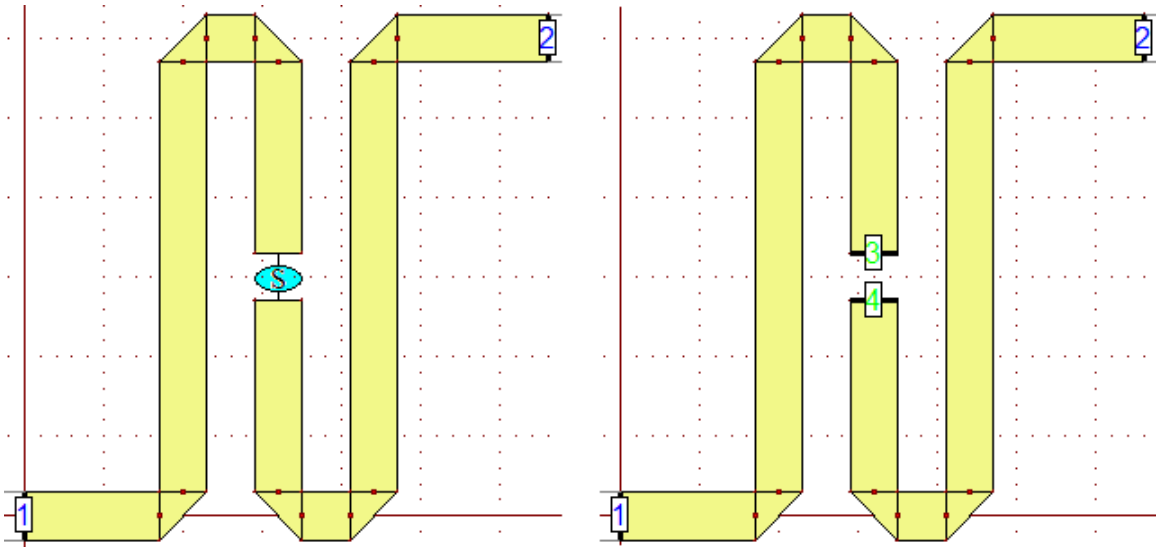
There are some limitations on the **Advanced Extension** port scheme we used in the last a few chapters, and other extension de-embedding schemes (**Extension for MMIC, Extension for Waves** and **50 Ohms For Waves**). The extension de-embedding schemes are accurate and efficient schemes. We should use them whenever possible. The problem is that an extension port needs an extension arm on where you define a port. In some cases, we may not have enough room for the extension arms. The port extensions are shown on the MGRID window as gray color. Whenever you see they are overlapping with each other or the other parts of the circuit or even coupled with other parts of the circuits, you should try to revise the circuit to avoid it. One way to avoid it is to change the direction of the port. Another way is to use the **Localized for MMIC**, **Vertical Localized** and **Horizontal Localized** schemes. The three localized schemes are specially designed for parameter extraction of highly packed structures. Usually, the three localized schemes are less accurate than the three extension schemes. However, we need to use the localized schemes when we do not have enough space for the port extensions. For some applications such as probe-fed antennas, the localized schemes are very accurate and the extensions schemes do not apply.

Section 7.1 A Lumped Element in a Highly Packed Structure.

The first structure we are going to discuss is a meander line saved in .\ie3d\samples\meander1.geo. It is shown in Figure 7.1a. It is a 75-micron meander line on a 100-micron thick GaAs substrate. It is a 2-port structure with a gap somewhere on the line. The ports 1 and 2 are defined with the Advanced Extension scheme.

The first question you may have is how we create the polygons of the structure in Figure 7.1a. Using the knowledge you learn, you should be able to build easily. We can consider the structure as a trace with a gap at the center. From the coordinates of the center line along the trace, we should be able to build the trace using the **Adv Edit->Build Path** command with specified width. Then, we can build another rectangle of the width of the gap on another layer first. Its length can be slightly larger than the width of the trace we

just built. We can select the polygon representing the gap and use the **Adv Edit->Build Holes and Vias from Selected Polygons** command to build the gap in one shot. The procedure should be simple and we will not discuss the detail here.



(a) Original structure with a lumped element. (b) The element replaced by localized ports on IE3D.

Figure 7.1 A meander line structure with a 2-port s-parameter module connected.

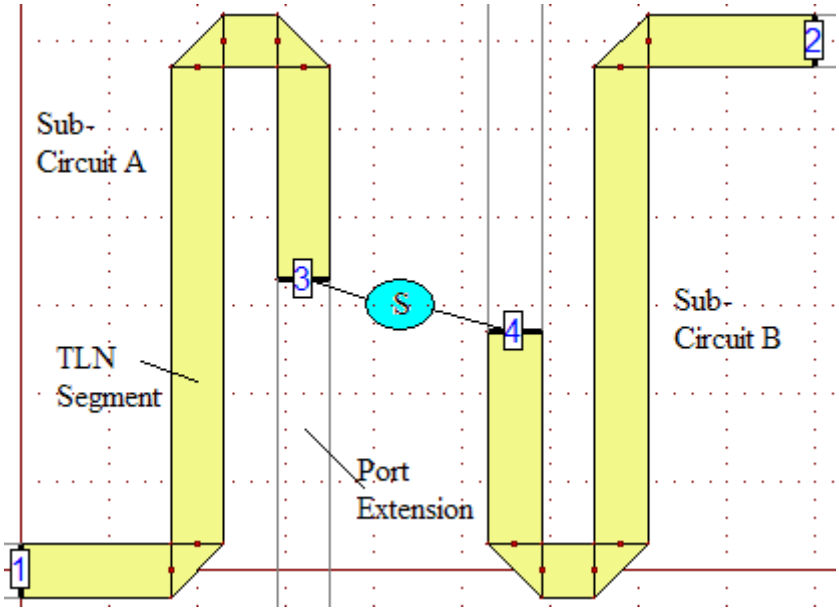


Figure 7.2 The circuit is separated into 3 parts: A, B and S.

Assume we want to insert a 2-port s-parameter file into the gap. The 2-port s-parameter file is the .\ie3d\samples\amp.sp. It is an idealized amplifier with S11 = S22 = S12 = 0 and S21 = 2 from 0.5 to 30 GHz with 0.5 GHz step. How can we solve such a problem accurately?

In order to connect the s-parameters into the structure, we need to define two ports to replace the 2-port lumped element (see Figure 7.2). You may consider it a natural way to separate the circuit into three

parts: Sub-Circuit A, Sub-Circuit B and S (see Figure 7.2). You can simulate the Sub-Circuit A, Sub-Circuit B separately. Then, you use a nodal circuit simulator to combine the s-parameters of A, B and S together to get the final 2-port s-parameters. Such an approach might be the only choice on a nodal circuit simulator. However, it may create a big problem in an EM simulator. In fact, it is a serious accuracy problem on a circuit simulator too and the users may not even be informed of it.

The A and B are strongly coupled. If we separate them into two EM simulations, the coupling between them will be neglected in the simulation. The results may be different from the reality.

Assume we need to simulate the A alone in Figure 7.2. There will be another serious problem if we use the extension port on the port 3. The problem is that the port extension of port 3 is strongly coupled with the TLN segment in A. The port extension on an extension port is used to remove the effect of the higher order modes. However, if it is coupled strongly to the structure itself, it will create higher order modes together on the structure. For our example, it is creating the odd and even modes together with the adjacent parallel line. It will introduce numerical error into the simulation.

On IE3D, we have implemented the Localized for MMIC, Vertical Localized and Horizontal Localized port schemes to address this problem. They will allow the users to extract the s-parameters in a highly packed circuit without losing accuracy.

Section 7.2 Defining Localized Ports on Structures

Step 1 Open the file **.\ie3d\samples\meander1.geo**. It will look like the picture shown in Figure 7.1a without the lumped element.

Step 2 Select **Port->Port for Edge Group** command. MGRID will prompt you for the De-Embedding Scheme. Select Localized for MMIC Circuits. Select OK to continue. We are going to use the localized ports.

Defining a Localized for MMIC Circuits port is similar to defining an extension port. We will select the edges. For the Localized for MMIC scheme, the edges must be horizontal edges (with constant Z). The second requirement is that the port must be close to the infinite ground plane or the No.0 substrate with high conductivity. Normally, the port-to-ground distance should not exceed 8% of the waveguide wavelength. When the port-to-ground distance is too large, it will introduce ambiguity. The ambiguity is similar to the one in your practical design when you are connecting a resistor across a gap. When the gap size is significant compared to wavelength, the s-parameters of the resistor is no longer determined by the value of the resistor, it will include other parasitic effects.

Step 3 Window the edge for the port 3 in Figure 7.1b. The port 3 will be defined. The port number "3" on it is in green color. Also, you will not see the gray color port extension. If you define an extension port on the same edge, you will see a gray color port extension extending beyond the gap and overlapping with the polygons on the other side. In such a case, the simulation result may not be correct. In fact, the MGRID will issue you a warning when you setup a simulation on a structure with the port extensions overlapping with other polygons.

Step 4 Window the edge for the port 4 in Figure 7.1b. The port 4 will be defined. We will get the exact picture in Figure 7.1b. Select **Port->Exit Port** command to exit the mode. Save it as: .\ie3d\practice\meander2.geo.

In the IE3D version earlier than 9.0, you are allowed to connect a lumped element to the structure on MGRID. You can start a mixed-electromagnetic and nodal simulation and optimization on the MGRID. This feature was introduced in version 2.0. In fact, similar

capability has been implemented into MODUA with better compatibility with other post processing tools. The lumped element on the MGRID is abandoned on the version 9.0 for better compatibility with the other programs of the IE3D. When lumped elements are involved, we will use both MGRID and MODUA for the mixed electromagnetic and nodal simulations and optimizations.

For the simulation of the current circuit with lumped element, it is not necessary to perform the EM simulation and the nodal simulation simultaneously, even though we can do it. You are suggested to simulate the structure first. Then, you can connect any s-parameters of the same ports to it to get the final s-parameters. By doing so, we can separate the structure from the lumped element. When we change the lumped element, we do not need to re-simulate the structure again. We can connect the changed lumped element with the pre-simulated s-parameters of the structure to get the final results easily.

Section 7.3 Simulating the Distributed Part at Captured Frequency Points from File.

Step 1 Select **Process->Simulate** command of MGRID.

We are going to simulate the 4-port structure. However, we want to simulate it at the same frequency points as the lumped element: .\ie3d\samples\amp.sp.

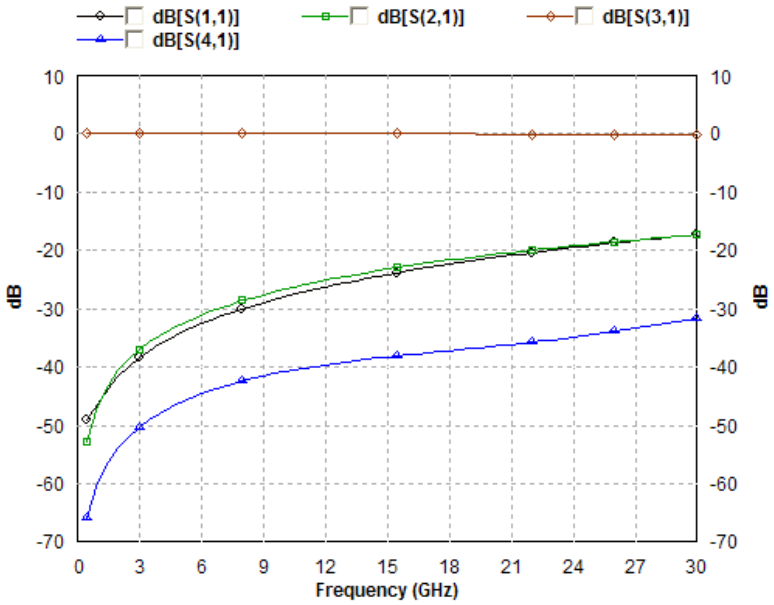


Figure 7.3 The 4-port s-parameters of the meander2.geo.

Step 2 Select the **Capture** button in the **Frequency Parameters** block of the **Simulation Setup** dialog. MGRID will prompt you for the parameter file name. Select the .\ie3d\samples\amp.sp file. The frequency points in the amp.sp file will be captured and listed in the frequency listbox. Select OK to simulate the structure in seconds. The simulation results are saved into .\ie3d\practice\output\meander2.sp file as well as the geometry file. Figure 7.3 shows the dB[S(1,1)], dB[S(2,1)], dB[S(3,1)] and dB[S(4,1)]. The results in Figure 7.3 are very meaningful: (1) The dB[S(3,1)] is close to 0 dB over the whole frequency range. This is expected; (2) The dB[S(1,1)] starts from–50 dB and goes up to about –17 dB at 30 GHz. The Zc of the isolated trace should be close to 50-ohms. The high dB[S(1,1)] at high frequency is caused by the strong coupling between the TLN segments; (3) The dB[S(2,1)] goes up to about –17 dB at 30 GHz. Again, it is due to the strong coupling between the trace; (4) It is interesting to

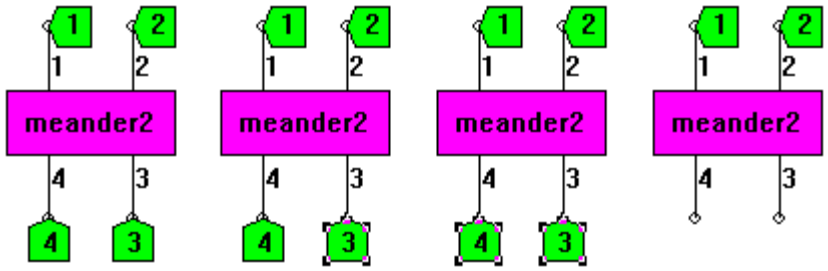
see the dB[S(4,1)] is below –30 dB, much lower than the dB[S(2,1)]. It is believed that the couplings between the different sections are canceling each other.

We have obtained the s-parameters of the distributive portion now. How can we find the s-parameters of the complete circuit? We will demonstrate it in the next section.

Section 7.4 Connecting S-Parameters of Different Sub-circuits using MODUA

Step 1 Select **File->Display S-Parameter Module**. Select .\ie3d\practice\output\meander2.sp to display it. Select **Control->Display Toggle** command. We will get the display in Figure 7.4a. We are going to connect the 2-port s-parameters of amp.sp to the terminals 3 and 4 of the 4-port s-parameters meander2.sp. We should do it on MODUA.

Step 2 Click at the port 3 on MODUA. The 4-corners of port 3 are highlighted, indicating it is being selected (see Figure 7.4b). Click at the port 4 on MODUA. The port 4 is also selected (see Figure 7.4c). Select **Edit->Delete Module** command (or hit Delete button) on MODUA. MODUA will prompt you the simulation results will be abandoned. Select OK to continue. The selected ports 3 and 4 are deleted and we will get the Figure 7.4d.



(a) (b) (c) (d)

Figure 7.4. The different states of selecting and deleting the ports.

Step 3 Select **File->Add Parameter Module** command on MODUA. MODUA will prompt you the file name of the s-parameters you want to add.

You can add s-parameter files (using **File->Add Parameter Module** command), geometry files from MGRID (using **File->Add Geometry Module** command), resistors, capacitors and inductors (Resistor, Capacitor and Inductor commands in the Element menu of MODUA), etc. into a circuit on MODUA.

Step 4 Select the .\ie3d\samples\amp.sp file to import it. MODUA will prompt you for the ID of the s-parameter file. Select OK to accept the default ID "amp". The black box for the amp.sp is following the mouse cursor.

Click somewhere close the "meander2" module. The "amp" module is dropped close to the "meander2" module. The 4-corners of "amp" are still highlighted, indicating it is still being selected (see Figure 7.5a). Click at an empty space on MODUA. The highlights on "amp" will be gone (see Figure 7.5b), meaning the "amp" module is fixed. It is equivalent to selecting the **Edit->Fix Module** command on MODUA.

Step 5 Select **Element->Connection** command on MODUA. Click at the terminal 3 of "meander2". You will see a line connecting the mouse cursor and the terminal 3 of "meander2". Click at the terminal 1 of "amp". A line will be connected between the two terminals (see Figure 7.5c). Click at the terminal 2 of "amp". You will see a line connecting the cursor and the terminal 2 of

"amp". Click terminal 4 of "meander2". A line will be connected between the two terminals (see Figure 7.5d).

Click at an empty space on MODUA. MODUA will prompt you that no terminal is selected for defining more connection and whether you want to repeat. Please select NO. MODUA will exit from the **Element->Connection** mode. It is equivalent to selecting **Exit Element** in **Element** menu.

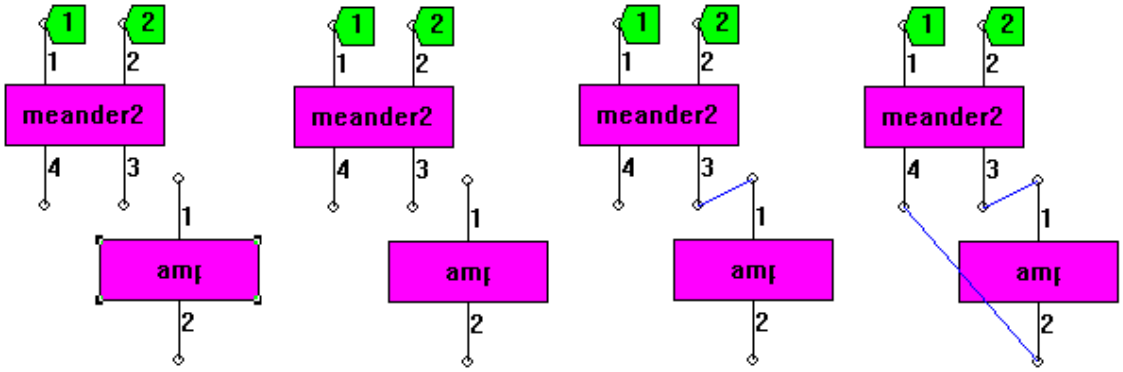


Figure 7.5 The module connections on setting up a circuit simulation on MODUA.

Step 6 Save the file on MODUA as: .\ie3d\practice\amp2.dsg. This is the finished file for a complete circuit simulation.

Step 7 Select Process->Simulate command on MODUA. MODUA will find all the frequency points from amp.sp and list them on the checklist box. Select Define All button. The total 60 frequency points will be checked. Select OK to continue. MODUA will finish the circuit simulation in no time. MODUA will display the s-parameters on the window. Select the **View->Graph Parameters** command on MODUA to adjust the range of the display. We can get a picture similar to Figure 7.6.

Please select **File->Save S-Parameters** command on MODUA to save the s-parameters as: .\ie3d\practice\amp2.sp.

The results in Figure 7.6 are very interesting. The dB[S(1,1)] is increasing with frequency and it is as expected. At low frequency, the dB[S(2,1)] is about 6 dB and it is as expected. However, the dB[S(2,1)] is about 8 dB at 30 GHz. Where does the extra 2 dB come from?

Experienced amplifier designers should realize that the extra 2 dB is from the coupling between the sections of the meander line. The coupling forms a feedback loop. If the coupling is too strong, it will make the amplifier unstable.

On the mixed EM and nodal simulation discussed above, we have included the coupling among the different sections of the distributive part of the circuit. For the lumped element, we use s-parameters to describe it. The coupling between the lumped element and the distributive part can also be included into the s-parameters if the users can measure it. The IE3D model should yield accurate results in the amplifier design. In fact, such a feature has been demonstrated in industrial and academic design of multi-stage amplifiers.

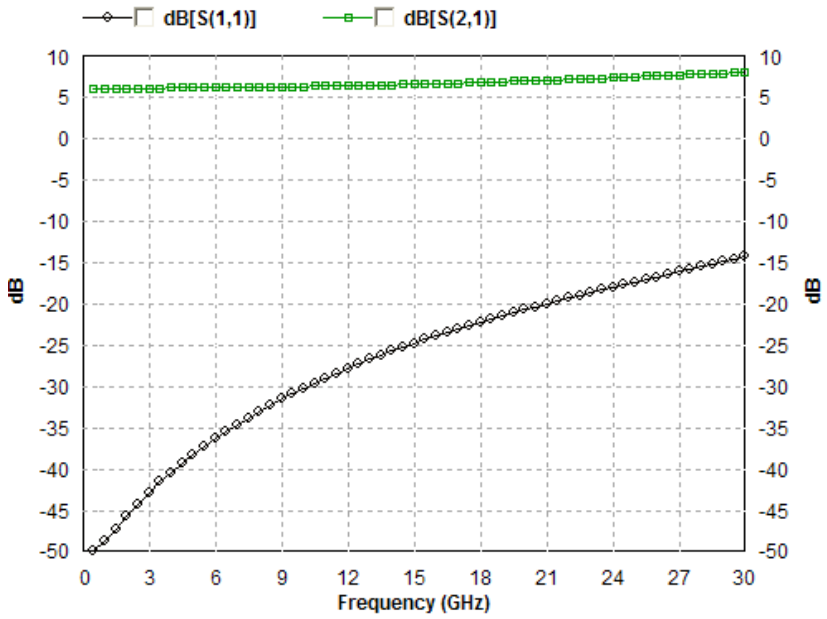


Figure 7.6 The frequency response of the complete amplifier circuit.

Section 7.5 Coupled Model and Uncoupled Model

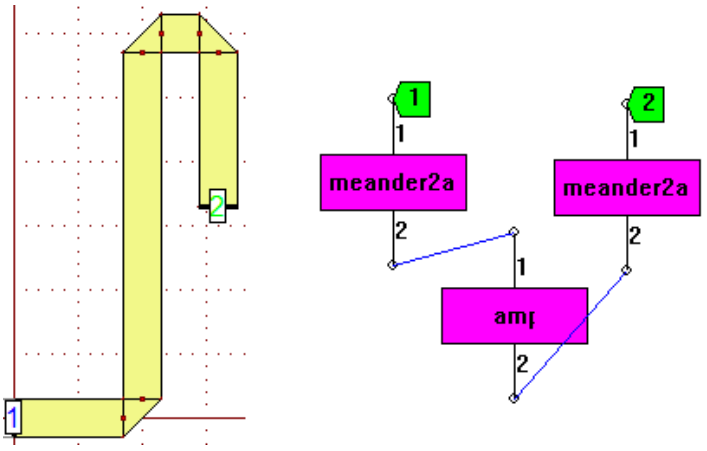
In our model connecting the s-parameters of meander2.geo and the amp.sp, we include all the couplings between the different sections of the meander line. We should be able to get very accurate result. We will call this model the "Coupled Model" in our discussion in this section in the following.

One may ask what the result may look like if we neglect the coupling between the sub-structures A and B. That is the way a regular circuit simulator or an EM simulator based upon sub-circuits does.

If we use extension port for sub-structures A and B, the accuracy will be affected by the coupling between the meander line and the port extensions. However, we can use localized ports for the ports 3 and 4 in the sub-circuits A and B. The sub-structure A is created and saved in .\ie3d\samples\meander2a.geo (see Figure 7.7a. It becomes a 2-port structure and the port 2 is a Localized for MMIC port. We can simulate this structure and connect the s-parameters on MODUA as shown in Figure 7.7b. We do not need to simulate the sub-structure B because its s-parameters are the same as the sub-structure A. We can use the s-parameters of sub-structure A in place of that for the sub-structure B (see Figure 7.7b).

Building the "Uncoupled Model" in Figure 7.7b on MODUA is straightforward. A user can use the same steps in Section 4 to add the s-parameter files, the connections. For the ports, the user can use the Port command in Element menu. "Port", "Short-Circuit", "Open-Circuit", "Connection", and "Mutual Inductor" are the so-called dependent modules on MODUA. They should be defined on the terminals of the independent modules (parameter modules, geometry modules, resistors, capacitors and inductors). We have shown the users how to define the connection elements. Defining other dependent modules is similar to defining the connection modules.

Step 1 Build the "Uncoupled Model" shown in Figure 7.7b. Save it as .\ie3d\practice\amp2a.dsg file. Example file is available in c:\ie3d\samples directory. However, the path for the parameter files may not be the same as yours. You need to check it. If they are different, you need to select each parameter module and use the Replace Module in Edit menu to replace the parameter file with the one in appropriate directory.



(a) (b)

Figure 7.7 The "Uncoupled Model" for the complete circuit.

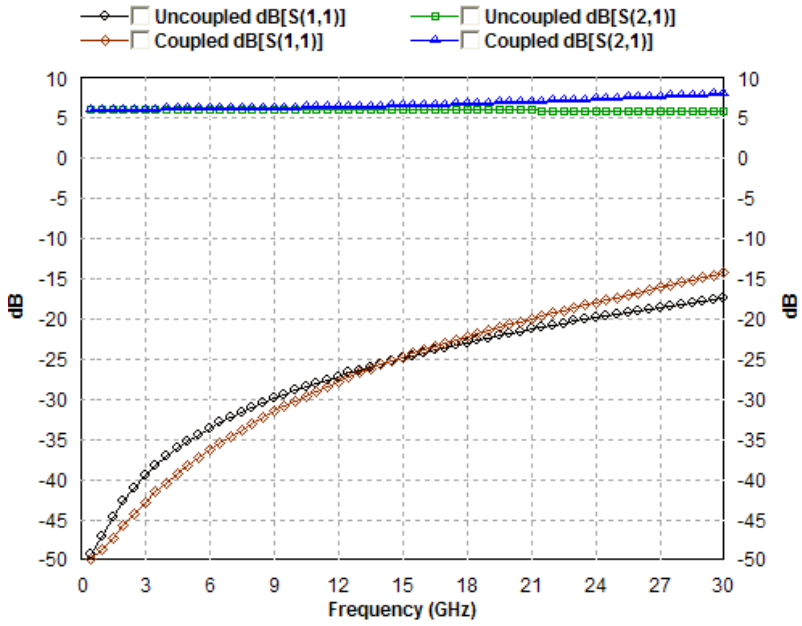


Figure 7.8 The comparison of the "Coupled" and "Uncoupled" models.

Step 2 Simulate the "Uncoupled Model" from 0.5 to 30 GHz. Its result will be displayed on MODUA. Select **View->Design Identification** command on MODUA and change it to "Uncoupled". Select **File->Save S-Parameters** command on MODUA to save simulation result as: amp2a.sp.

Step 3 Select **File->Parameter File Queue** command on MODUA. Select **Add** button to add the parameter file: amp2.sp into the queue. Define its ID as "Coupled". Select the **Display** button to go to **View->Display Queue Items** dialog. Select the queue file amp2a.sp and select the dB[S(1,1)], dB[S(2,1)] of amp2a.sp. Please check the Display Queue check box to enable it. Select OK to continue. The comparison is shown in Figure 7.8.

As you can see, the dB[S(1,1)] of the two cases is quite close. However, the dB[S(2,1)] for the "Uncoupled" case is always about 6 dB, while the dB[S(2,1)] for the "Coupled" case is from 6 dB at low frequency to about 8 dB at 30 GHz. The "Coupled" model should be closer to reality and the "Uncoupled" model failed to predict the positive feedback loop, due to the fact it neglects the coupling between the different segments of the meander line.

Section 7.6 Finite Ground Plane and Differential Modeling

Assume our meander line structure has a finite ground plane instead of an infinite ground plane. We have discussed modeling a finite ground PCB in the last chapter. We will give more discussion on modeling finite ground plane structures here.

Step 1 Run MGRID and open .\ie3d\samples\meander3.geo. It is the initial structure for the finite ground plane structure.

We will get the picture in Figure 7.9a. The ports 1 and 2 are still there. If you select **Param->Basic Parameters** command on MGRID, you will see that the No.0 substrate is no longer a high conductivity layer. It is air now. Basically, there is no infinite ground plane. Instead, there is a polygon on Z = 0. It will be serving as the finite ground plane.

As it is documented in Appendix AM, the **Localized for MMIC** port can only be used when there is an infinite ground plane on Z = 0. For this finite ground plane case, we will not be able to use the **Localized for MMIC** ports. In fact, we cannot even use the **Advanced Extension** ports as they are defined on the meander3.geo. The reason is that the reference point for an **Advanced Extension** port is not clear in a structure without an infinite ground plane. For most examples we have discussed, there is always one or more infinite ground plane. By default, the infinite ground plane becomes the reference point of an extension port. When there is no infinite ground plane on a structure, there will not be any default reference point for the extension ports. We need to define an **Advanced Extension** (or **Extension for MMIC**) port with one or multiple negative ports.

Out of the 7 existing de-embedding schemes implemented into the IE3D 10.0, the **Extension for Waves**, the **50-Ohms for Waves** and the **Localized for MMIC** ports are used exclusively for single-ended ports with the infinite ground plane as the reference point. They cannot be used for differential ports.

The **Advanced Extension** (or the **Extension for MMIC** scheme) can be used for differential ports. However, it will come with positive port(s) and negative port(s). The **Vertical Localized** and **Horizontal Localized** ports are differential ports by themselves. They require the users to define the positive and negative terminals no matter where they are used as single-ended ports or differential ports.

For our current example, we will demonstrate how we can use differential **Advanced Extension** ports and **Vertical Localized** ports to model structures without infinite ground planes.

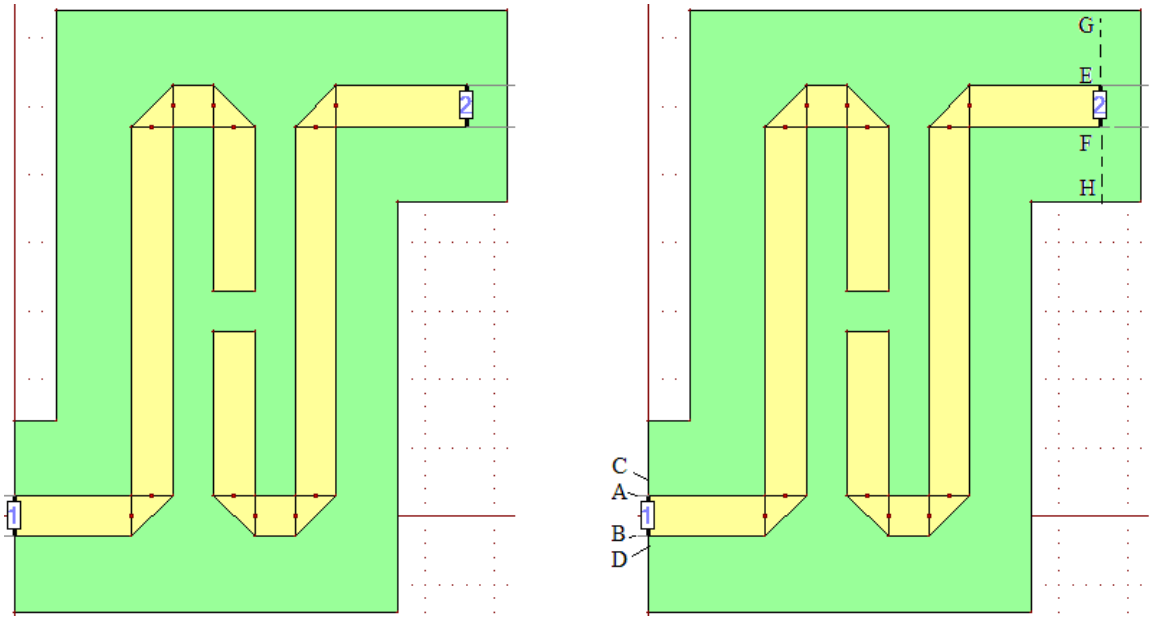
Step 2 Select **Port->Delete All Ports** command on MGRID.

We are going to define a differential **Advanced Extension** at the original port 1 location. The 1st step is to create some edge on the finite ground plane slightly wider than the edge on the trace where the port 1 is.

Step 3 Click at the No.2 Layer at Z = 0.1 mm layer on the Layer Window to focus the input on Z = 0.1. Select **Input->Set to Closest Vertex** in command to snap to the closest vertex. Click at the vertex A in Figure 7.9b. A vertex will be entered exactly at A.

Step 4 Click at the No.1 Layer at Z = 0 on the Layer Window to shift the input focus to Z = 0. Type Shift+R (equivalent to selecting **Input->Key In Relative Location** command). MGRID will prompt you for the shift offset values. Enter X-Offset = 0 and Y-Offset = 0.025 mm. Select OK.

MGRID will prompt that the vertex is close to an edge. Select YES to confirm connecting to the edge. A vertex will be inserted on the edge as vertex C, and the vertex C is on Z = 0. Its y-coordinate is 0.025 mm larger than the y-coordinate of vertex A.



(a) Original Structure (b) Structure for ports to be defined.

Figure 7.9 The meander line with finite ground plane.

Step 5 We are going to insert a symmetrical vertex at D. Please type Shift+R (equivalent to selecting **Input->Key in Relative Location** command). MGRID will prompt you for the offset of the next vertex to the last entered vertex. Enter X-offset = 0 and Y-offset = -0.125 mm. MGRID will prompt you that the vertex is close to an edge. Select YES to confirm connecting to the edge. A vertex will be inserted on the edge as vertex D.

Why should we enter the Y-offset = -0.125 mm? Basically, the edge AB is 0.075 mm long. The horizontal distance of vertices A and C is 0.025 mm. We want the horizontal distance between vertices B and D as 0.025 mm.

We would like to define a positive **Advanced Extension** port on the edge formed by vertices A and B (edge AB) on Z = 0.1, and a negative Advanced Extension port on the edge formed by vertices C and D (edge CD) on Z = 0. The edge CD is slightly wider than the edge AB. How much wider it should be? Our experience tells that the width may not be a critical parameter. Also, the edge CD does not necessarily to be symmetrical to edge AB completely. We just made the edge CD slightly wider than the edge AB.

Step 6 Type Ctrl+D. It is equivalent to selecting **Input->Drop All Vertices** command. The entered vertices A, C and D are dropped. However, the inserted vertices C and D remain there. You can also type ESC to drop all the vertices.

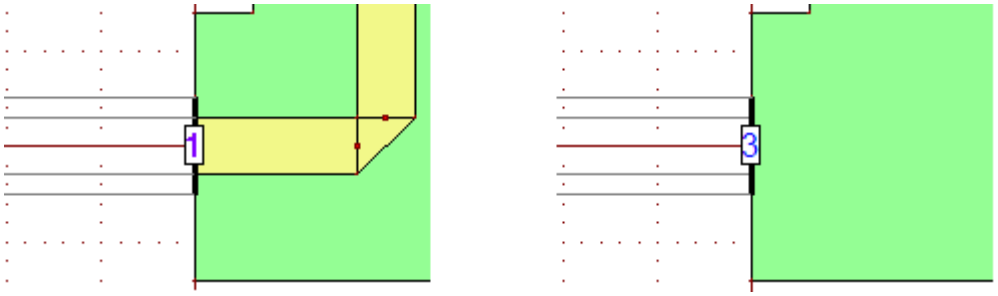
Step 7 Select **Port->Port for Edge Group** command. MGRID will prompt you for the de-embedding scheme. Select **Advanced Extension** scheme. Select OK to continue. MGRID will be ready for you to select edges for a port.

Step 8 Click at No. 2 Layer Z = 0.1 mm to display the edge AB on Z = 0.1 mm. All the check boxes on all the layers should be checked, meaning that all the layers are on focus.

Window the edge AB (or vertices 1 and 2) only. The vertices C and D should not be windowed. The port 1 is defined on the edge AB.

Step 9 Please window the edge CD. When we window the edge CD, the edge AB with port 1 is also windowed. However, MGRID will detect a port is on it and will not select the edge AB for the port. It will only select the edge CD for the port.

We will see multiple gray lines displayed on the left. You will see the symbol "1" on where we define a port (see Figure 7.10a). We do not see the new port we just defined. Please click at No.1 Layer at Z = 0 mm to focus the input at Z = 0 layer, we will see the "3" symbol (see Figure 7.10b).



(a) The "1" shown when focused on z = 0.1 (b) The "3" shown when focused on z = 0

Figure 7.10 The differential port defined and displayed.

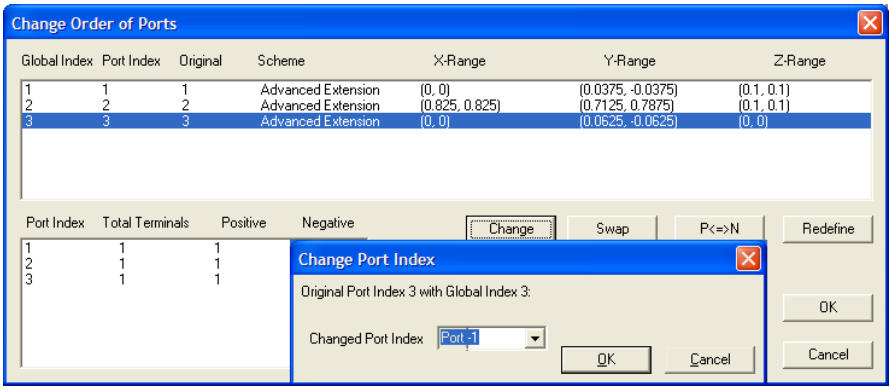


Figure 7.11 The Change Port Orders dialog and the Change Port Index dialog.

Step 10 Select **Port->Exit Port** command to exit the mode.

Step 11 We would like to make the port 3 to be the port "-1" in order to make a differential port using the **Advanced Extension** port scheme. Select Port->Change Port Orders for the dialog (see Figure 7.11). Select the No.3 item (Global Index = 3, Port Index = 3) in the upper list box. Select Change button. The Change Port Index dialog comes up. Change it to "Port -1". Select OK. MGRID will issue you a warning on the change. Select YES to get back to the Change Port Order dialog. You will see the No.3 item's Port Index is changed from 3 to -1. You will see the port 1 is changed to port "+1" and the port 3 is changed to port "-1".

Step 12 Select File->Save As and save the structure as .\ie3d\practice\meader4.geo.

Section 7.7 Aligning Edges for Differential Ports

We are going to define the original port 2 as a differential Advanced Extension port too. However, the polygon on the ground plane is slightly longer than the trace for the port 2 (see Figure 7.9b). In order to preserve accuracy, we would like to align the edges of the trace (on Z = 0.1) and the edge of the finite ground plane (on Z = 0). There are many ways to achieve it. We will use the simplest way for it.

Step 1 Click at Z = 0.1 mm. Select **Input->Set to Closet Vertex** to snap to the vertex. Click at the vertex E in Figure 7.9b to enter a matching vertex at E. Click at the vertex F in Figure 7.9b to enter another matching vertex at F. We have entered the edge EF on the Z=0.1 layer. What we want is to cut the finite ground along the line of edge EF.

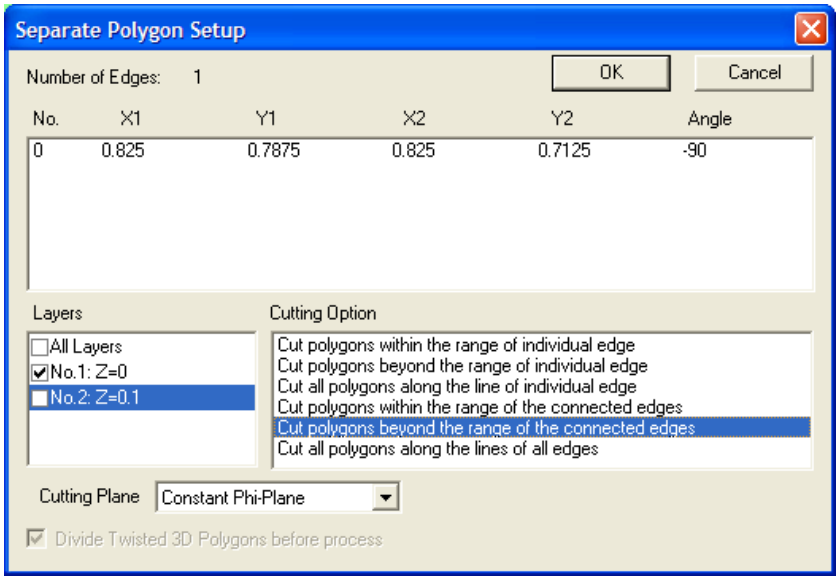


Figure 7.12 The Separate Polygon Setup dialog.

Step 2 Please select **Adv Edit->Separate Polygon** command. The Separate Polygon Setup dialog comes up. Please check No.1 Z = 0 in the Layers list box and uncheck the No.2 Z = 0.1 because we only want MGRID to cut the Z = 0 layer. Make sure the Cutting Plane is "Constant Phi Plane". Select the Cutting Option as either "Cut polygons beyond the Range of the connected edges" or "Cut all polygons along the lines of all edges" (see Figure 7.12).

Step 3 Select OK to continue. MGRID will cut the polygon on Z = 0 into two polygons along the edge EF (see Figure 7.13).

Step 4 Press down "Shift" and window the polygon 2 in Figure 7.13 to select it. Hit Delete button to delete the polygon.

Section 7.8 Using Continue Straight Path to Insert Vertices on an Edge.

Step 1 We are going to insert 2 vertices on the finite ground to make a slightly wider edge than edge EF. We will define the edge EF as the +2 port and the slightly wider edge on Z = 0 as –2 port. Certainly, we can follow the same steps to insert vertices C and D in Figure 7.9b to insert the vertices. However, we can try some other way for it.

Step 2 Please select **Edit->Select Vertices** mode. Focus selection on Z = 0. Window the vertices G and H one by one to select them only.

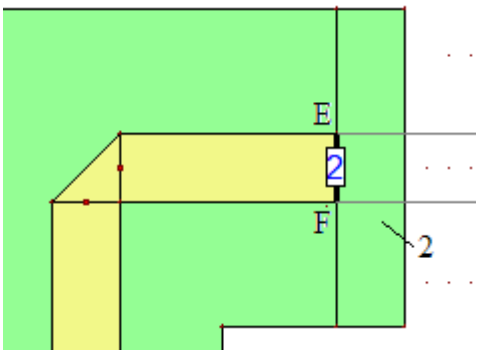


Figure 7.13 The polygon for the ground plane is cut along edge EF into two.

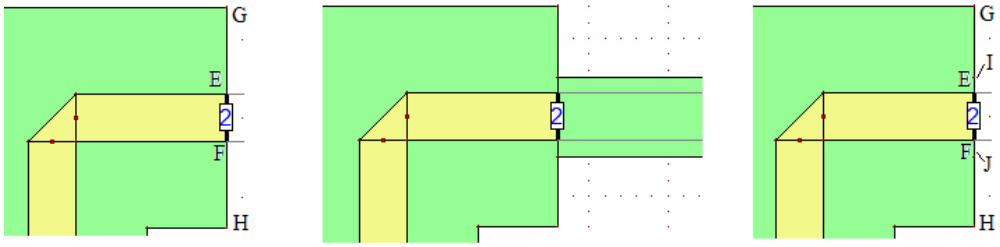
Step 3 Select **Adv Edit->Continue Straight Path** command. The **Continue Straight Path** dialog comes up.

Step 4 Please change the **Path Start Width** = 0.125 mm. The **Path End Width** is automatically updated as 0.125 mm. Select OK to accept the default **Path Length**. We do not need to worry about the Path Length because we will eventually delete the path. We just want to use its width to insert two vertices on the edge GH.

A path is created from the edge (see Figure 7.14b).

Step 5 Press down "Shift" and click at added rectangle from the path to select it. Hit Delete button to delete the added rectangle. Click at Z = 0.1 on layer window to focus on Z = 0.1. We will get the same picture in Figure 7.14c. It is similar to Figure 7.14a. However, there are two inserted vertices I and J on Z = 0.

Step 6 Select Port->Port for Edge Group command. Choose the Advanced Extension scheme. Select OK to continue. Click right mouse button to bring up the pop-up menu. Select Define Negative Port. Check the No.1 Layer at Z = 0. Window vertices I and J only (vertices E and F are enclosed. However, MGRID will detect a port is on it and it will not select the edge). The port "-2" is defined on edge IJ at Z = 0.



(a) Aligned edge (b) Adding a rectangle for the edge (c) Inserted 2 vertices.

Figure 7.14 Inserting two vertices K and L on Z = 0.

Step 7 Select **Port->Exit Port** command to exit. Save the file as: .\ie3d\practice\meander5.geo. By this step, we built the ports 1 and 2 as differential feeds. The port 1 actually contains a differential pair "+1" and port "-1", and the port 2 contains a differential pair port "+2" and port "-2". The next steps are to build the ports 3 and 4.

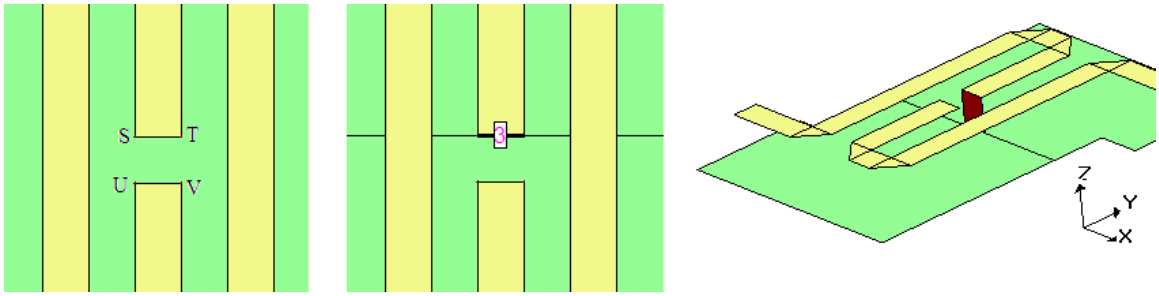
Section 7.9 Building Vertical Localized Ports on Finite Ground Plane Structures

We cannot define Localized for MMIC ports for the ports 3 and 4 for the finite ground plane structure. We have to use the Vertical Localized Port. A Vertical Localized Port is defined between some edges and its reference points. The edges and the reference points are determined by some vertical polygons.

Step 1 Press down "Shift" button and window the vertices S and T in Figure 7.15a to select the edge ST on Z = 0.1.

Step 2 Select **Adv Edit->Build Via Connection on Edges** command. The Build Via and Port on Edges dialog comes up.

Step 3 Please select the Connection Layer Z as: No.1: Z = 0. Change the Negative Level Z as 0 and the Positive Level Z as 0.1. Please check the Build Vertical Localized Port check box (see Figure 7.16). Select OK to continue. The polygon on Z = 0 is separated. The Vertical Localized port 3 is defined on a rectangle created from edge ST and connected to the edge on the Z = 0. Please click at the Z = 0.1 layer and we will see the picture in Figure 7.15b. Open the 3D View and you will see the vertical rectangle (see Figure 15c).



(a) The vertices S, T, U and V (b) The Vertical Localized port 3 (c) The vertical rectangle.

Figure 15. The Vertical Localized port 3 defined on the edge ST.

Step 4 Repeat Steps 1, 2 and 3 on the edge UV in Figure 7.15a. We will be able to define the Vertical Localized port 4 as shown in Figure 17.

Step 5 Please save the file as: c:\ie3d\practice\meander6.geo. Please simulate it and compare its result with the infinite ground plane results from meander2.geo. The comparison is shown in Figure 7.18. As you can see, the infinite and finite ground plane results are quite close. There are some dB differences in dB[S(4,1)]. However, dB[S(4,1)] is in the level of -30 dB and it is small anyway. For the s-parameters combined with the lumped element, the dB[S(2,1)] matches very well while there is some small difference in the dB[S(1,1)].

You may notice that we have to define "+" and "-" port for the Advanced Extension port for a differential port. However, for the Vertical Localized port, we just need to define one. Why is it like that? For a Vertical Localized port, we ask you for the "+" level and "-" level. It already implies that the port is a differential port by itself. We do not need to define a pair for it. The same is true for the Horizontal Localized port.

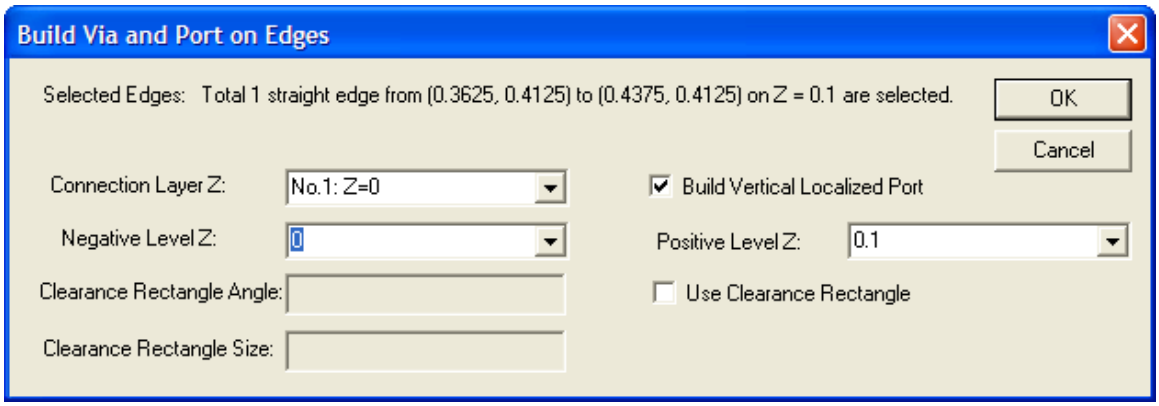


Figure 7.16 The Build Via and Port on Edges dialog.

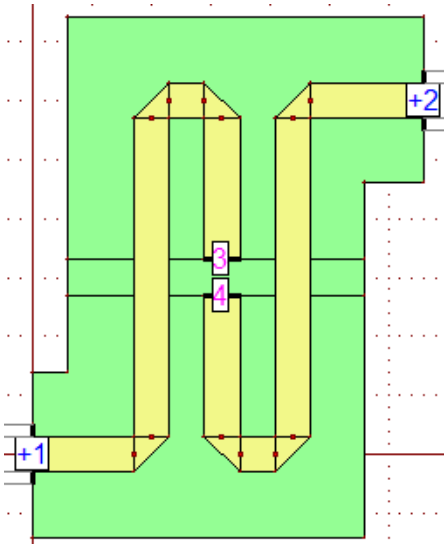
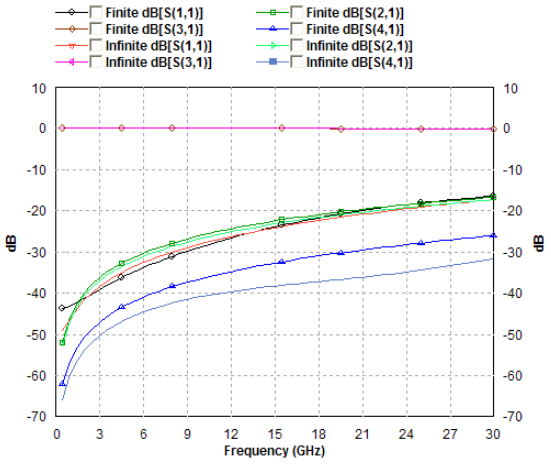
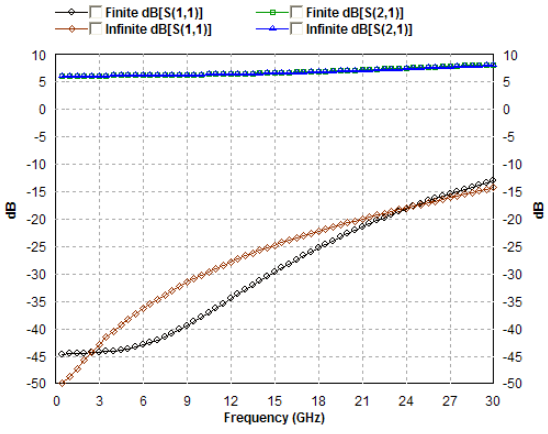


Figure 7.17 The Vertical Localized ports 3 and 4 defined.



(a) Comparison on the original 4-port s-parameters



(b) Comparison on the 2-port s-parameters with lumped element.

Figure 7.18 The comparison of the infinite ground and finite ground structures.

Section 7.10 Mixed EM and Network Optimization

Take the meander2.geo we built in Section 3 as an example. The complete circuit including the lumped element yields dB[S(2,1)] = 8 dB at 30 GHz. The gain of the lumped element (or amp.sp) is only 6 dB. The extra 2 dB comes from the positive feedback through the coupling between the meander line segments. Too much positive feedback may cause the amplifier to be unstable or resonating. It is impossible to completely eliminate the coupling for a microstrip structure. However, a well-designed circuit can reduce the unwanted coupling. We will use this example to demonstrate how we can use the mixed EM and nodal circuit optimization in IE3D for high accuracy and high performance design.

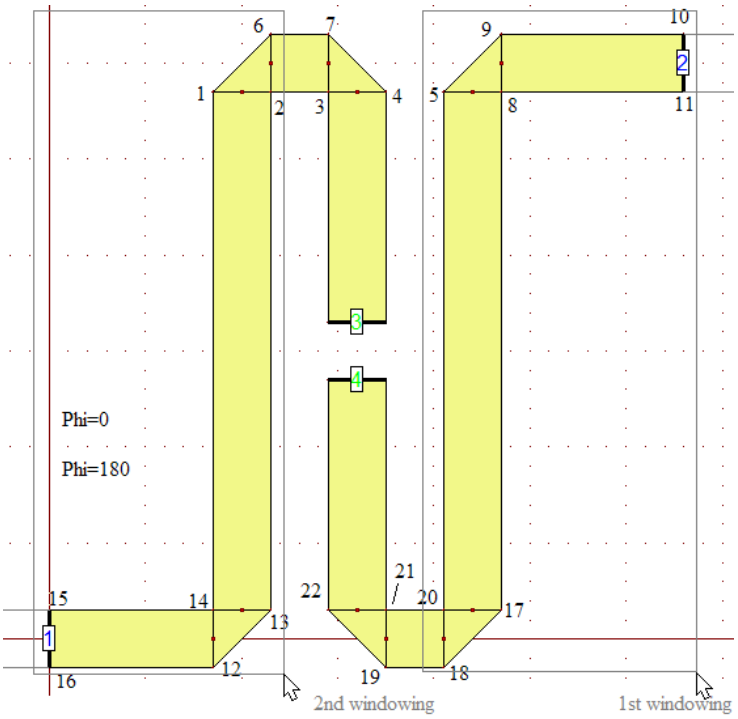


Figure 7.19 The meander2.geo with most vertices labeled.

Figure 7.19 shows the meander2.geo with its vertices labeled. We are going to define variables on it. To reduce the coupling, we should increase the width of the gaps between vertices 2 and 3, vertices 4 and 5. The goals we want to achieve are listed in Table 7.1. How can we define the gap width as an optimization variable while we can still keep the 2 conditions defined as goals 2 and 3 in Table 7.1?

Table 7.1 The goals for the optimization.

Goal	Description
1. Find the minimum gap width which can achieve dB[S(2,1)] < 6.5 dB for the frequency range from 0 to 30 GHz	The wider the gap is, the lower the dB[S(2,1)] is. However, we may not want it to be too wide. We want it to be small enough that the dB[S(2,1)] will not exceed the specified range.
2. Keep the width of the 2 gaps to be the same size	We do not want the widths of the 2 gaps to be different.
3. Keep the total length of the meander line to be unchanged	We want the phase of Ang[S(2,1)] to be about the same during the optimization.

Please understand that the goals defined in Table 7.1 are not from any practical design. We use this example just to demonstrate how we can perform accurate and efficient electromagnetic optimization on the IE3D.

Step 1 Our first task is to define the optimization variables on the geometry. Please run MGRID and open .\ie3d\practice\meander2.geo.

Step 2 Select **Edit->Select Vertices** command. Please make sure that the check box on the Z = 0.1 layer is checked. Window the vertices 5, 8-11, 17-18, and 20 in Figure 7.19 to select them (see the 1st windowing in the figure)

You may wonder why we do not use the short cut of "Shift" button to select the vertices. If we try to press down "Shift" and window the vertices 5, 8-11, 17, 18 and 20, we will not be able to select the vertices. Instead, MGRID will detect 3 polygons are enclosed and it will automatically get into **Edit->Select Polygon Group** mode. Only when MGRID does not detect any complete polygon is enclosed, it will check the vertices enclosed and set to the Select Vertices mode.

Step 3 Select **Optim->Variable for Selected Objects** command. The Optimization Variable Definition dialog comes up. Select **Vertices Mapped** to **New Variable**. Enter the **Tuning Angle** as 0. Enter the Variable Comment as "Gap Width". Select OK to accept other default values. MGRID is set to the defining Low Bound mode.

Step 4 Move the mouse somewhere on the left and click. Enter the Low Bound as "0" because we know we should increase the gap width and we can set the Low Bound to 0. Move the mouse somewhere on the right and click. Enter the High Bound as "0.1". Select OK to continue. MGRID will prompt you the process is finished. Select Continue Without Action to finish it.

By this step, we have defined the gap width on the right hand side as optimization variable. In the other word, we are able to use the variable to control the right gap width in Figure 7.19. We also want the left gap width in Figure 7.19 changing simultaneously with the right gap width.

Step 5 Select **Edit->Select Vertices** in command. Make sure the check box on the Z = 0.1 layer is checked. Window the vertices 1, 2, 6, 12-16 in Figure 7.19 to select them (see the 2nd windowing in the figure).

Step 6 Select **Optim->Add Selected Objects to Variable** command. The Optimization Variable Definition dialog comes up. Select **Vertices Mapped To**: **No.1 Variable**. Enter the **Tuning Angle** = 180 degrees. Enter the **Tuning Rate** = 1. Enter the **Call Comment** as: 2nd call.

We want to relate the newly selected vertices to the No.1 variable we defined in Step 2. The originally selected vertices for No.1 variable in Step 2 is moving toward φ = 0 degree angle, when the No.1 variable has positive offset value. The newly selected vertices in Step 5 should move toward φ = 180 degree angle at the same rate. That is the reason why we define the **Tuning Angle** = 180 degrees and the **Tuning Rate** = 1. We can define the **Tuning Angle** = 0 degree and the Tuning Rate = -1. They will have the same meaning.

Step 6 Select OK to continue. The vertices are associated with the No.1 variable as the 2nd call.

Step 7 How does the No.1 variable control the widths of the 2 gaps? Let's check it. Select **Process->Display Meshing** in command. MGRID prompts you for the Optimization Variable Offset. Click on the No.1 variable in the list box. Enter the Offset Value = 0.1. The value will be automatically updated. Select OK. The Automatic Meshing Parameters dialog comes up. Select

OK. MGRID will perform the meshing with the offset value to the optimization variable. After finishing it, MGRID will prompt the user on the statistics of the meshing. Select Continue.

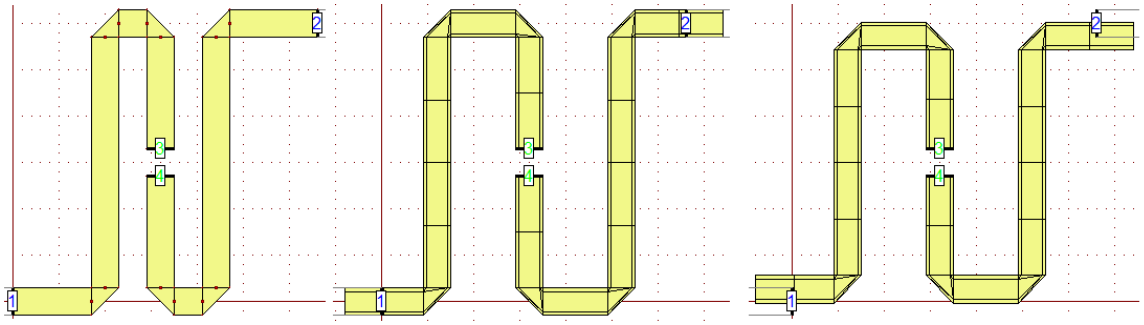
MGRID shows the meshed geometry in Figure 7.20b, compared to the original structure in Figure 7.20a. Please note that the locations of the ports 1 and 2 are not updated in the tuned geometry. They should be at the ends of the two traces.

As you can see, the meshed geometry has a wider gap between the traces. Also, the widths of both gaps are kept the same. However, the total length of the trace is increased too. We can analyze how much longer the trace is compared to the original length. The width of each gap is increased by 0.1 mm. There are two gaps. The total increase in the trace length is 0.2 mm. How can we keep the length when we increase the gap width? What we can do is to move the vertices 1-11 toward the negative y-direction (or ϕ = -90 direction) and move the vertices 12-22 toward the positive y-direction (or (or ϕ = 90 direction). Moving the vertices 1-11 toward the negative y-direction by 0.1 mm will reduce the trace length by 0.3 mm. Moving the vertices 12-22 toward the positive y-direction by 0.1 mm will reduce the trace length by 0.3 mm. Totally, it will reduce the trace length by 0.6 mm. Therefore, we should map the vertices 1-11 to the No.1 variable with Tuning Angle = -90 degrees and Tuning Rate = 0.333333. We should also map the vertices 12-22 to the No.1 variable with Tuning Angle = 90 degrees and Tuning Rate = 0.3333333333.

Step 8 Select **Edit->Select Vertices** command. Make sure the check box for Z = 0.1 layer is checked. Window the vertices 1-11 in Figure 7.21 to select them. Select **Optim->Add Selected Objects to Variable** command. The Optimization Variable Definition dialog comes up. Select **Vertices Mapped To**: **No.1 Variable**. Enter the **Tuning Angle** = -90 degrees. Enter the **Tuning Rate** = 0.3333333333. Enter the **Call Comment** as: 3rd call.

Step 9 Select **Edit->Select Vertices** command. Make sure the check box for Z = 0.1 layer is checked. Window the vertices 12-22 in Figure 7.21 to select them. Select **Optim->Add Selected Objects to Variable** command. The Optimization Variable Definition dialog comes up. Select **Vertices Mapped To**: **No.1 Variable**. Enter the **Tuning Angle** = 90 degrees. Enter the **Tuning Rate** = 0.3333333333. Enter the **Call Comment** as: 4th call. Select OK to continue.

Step 10 Select Process->Display Meshing command. Select the No.1 variable (or the only variable) in the list. Enter the Offset Value = 0.1. Select OK to continue. MGRID will prompt you for the Automatic Meshing Parameters dialog. Select OK to continue. MGRID will finish the meshing in no time and prompt you the statistics on the meshing. Select Continue button. The meshed structure is shown in Figure 7.20c. This time, when the gap width is increasing, the dimension of the meander line in the y-direction is reduced. The change rate is chosen such that the total length of the trace is unchanged when the optimization variable is changing.



(a) Original Structure (b) The meshing with gaps tuned (c) The meshing with gaps and width tuned.

Figure 7.20 The structure at different stages of the defining the tuning variable.

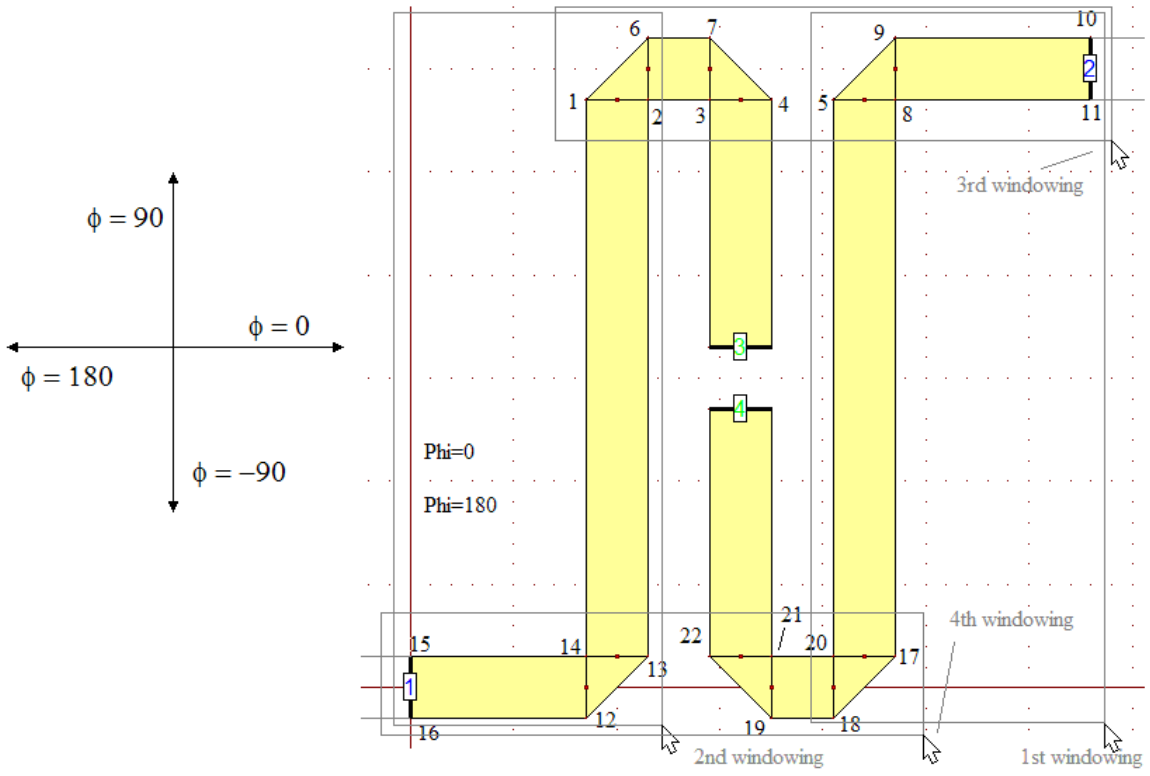


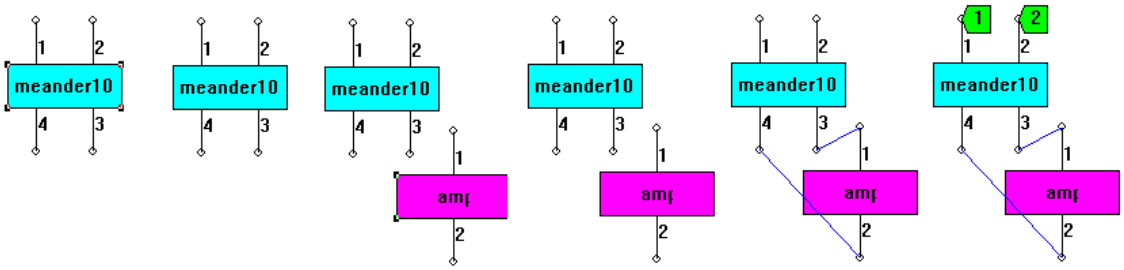
Figure 7.21 The 3rd and 4th windows on the structure.

Step 11 Save the geometry as: .\ie3d\practice\meander10.geo. The optimization variable is defined. The geometry is ready for simulation and optimization. However, we can not optimize it directly on MGRID/IE3D. The reason is that we are going to optimize the structure with the lumped element connected.

Step 12 Run MODUA and start a new project. Select **File->Add Geometry Module** command on MODUA. Select the .\ie3d\practice\meander10.geo file. MODUA prompts you for the Module ID. Select OK to accept the default ID of: meander10. The black box for the "meander10" is following the cursor. Click at an empty space on the window. The "meander10" is dropped and the 4 corners are highlighted, indicating it is still being selected (see Figure 7.22a). Click the mouse at an empty space to fix the "meander10" (the marks at the 4 corners are gone, see Figure 7.22b).

Step 13 Select **File->Add Parameter Module** command on MODUA. Select the .\ie3d\samples\amp.sp file. MODUA prompts you for the Module ID. Select OK to accept the default ID of: amp. Click at an empty space on the window to drop the "amp" module. It is still being selected with the highlight at the 4 corners (see Figure 7.22c). Click at an empty space to fix the "amp" module (see Figure 7.22d).

Step 14 Select **Element->Connection** command on MODUA. Click at the terminal 3 of "meander10" and click at the terminal 1 of "amp". A line is created to connect the 2 terminals. Click at the terminal 4 of "meander10" and click at the terminal 2 of "amp}. A line is created to connect the 2 terminals. We will get the Figure 7.22e. Click an empty space to exit the Connection mode.



(a) (b) (c) (d) (e) (f)

Figure 7.22 The different states in setting up the design on MODUA.

Step 15 Select **Element->Define All Ports** command on MODUA. MODUA reports that total 2 spare terminals are available. Select Yes to continue. The terminals 1 and 2 of "meander10" are connected with ports 1 and 2 (see Figure 7.22f).

Step 16 Save the file as .\ie3d\practice\amp2b.dsg. It is ready for optimization with the geometry module "meander10" and the s-parameter module "amp".

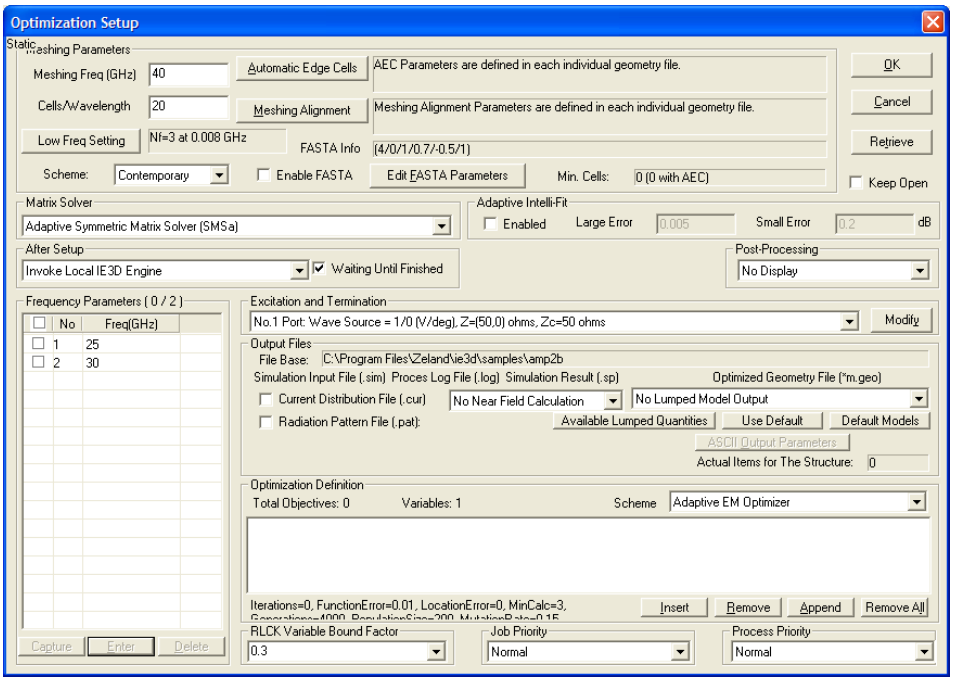


Figure 7.23 The Optimization Setup dialog on MODUA.

Step 17 Select **Process->Optimize** command on MODUA (Note: If you see this menu item is not activated, it is likely you did not save the design file). MODUA will prompt you to select a list of available frequency points. They are from the **amp.sp** file. There are 60 possible frequency points from 0.5 to 30 GHz with 0.5-GHz frequency step. Check the "25" and "30" GHz for 2 frequency points. We are going to optimize the circuit at the 2 frequency points.

We are trying to make sure the dB[S(2,1)] of the circuit below 6.5 dB. Due to stronger coupling at high frequency, we just need to make sure the dB[S(2,1)] < 6.5 dB at the high end. We do not need to worry about the low end frequency because we are sure that the dB[S(2,1)] will be below 6.5 dB if the dB[S(2,1)] < 6.5 dB at the high end.

Step 18 Select OK to continue. The **Optimization Setup** dialog comes up. It is similar to the MGRID's Optimization Setup dialog (see Figure 7.23). The 2 frequency points: 25 and 30 GHz are listed in the list box. Uncheck the Adaptive Intelli-Fit (AIF).

Step 19 Select **Add** button in the **Optimization Definition** dialog. The **Optimization Goal** dialog comes up. The Start Frequency = 25 GHz and the End Frequency = 30 GHz. The Quantity = dB(S). Select the 1st Parameter as (2, 1). Select the Operator as By Itself. Select the Objective Type as: objective1 < Optimization Quantity < objective2. Enter the objective1 = 6.3 dB and the objective 2 = 6.5 dB (see Figure 7.27). Select OK. The optimization goal is added into the Optimization Definition listbox.

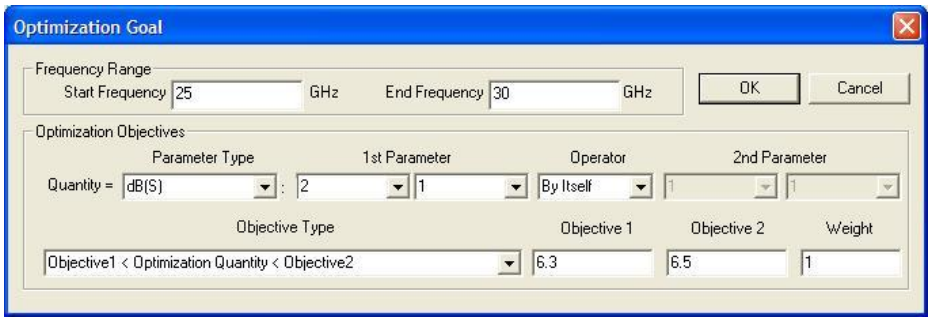


Figure 7.24 The Optimization Goal dialog when all the parameters are entered.

Step 20 By default, the Adaptive EM Optimizer is used. This is a simple case. We can choose Powell Optimizer to get fast result.

Also, MODUA uses the same Simulation Setup dialog and Optimization Setup dialog as MGRID. Even after you can change AEC parameters, they will not affect those used in a structure simulation for a geometry module. MODUA and IE3D will automatically use the information saved into the geometry module for AEC and other meshing related parameters.

Select OK and MODUA will invoke the IE3D to perform the mixed EM and nodal network optimization. It will take a fraction of a minute finish the optimization because it is a simple optimization problem. After the optimization, IE3D will save the optimized design into .\ie3d\practice\amp2bm.dsg file. It takes about 8 simulations. The simulation log is in .\ie3d\practice\amp2b.log.

Step 20 Run MODUA. Open the optimized design file: .\ie3d\practice\amp2bm.dsg. You will see the module "meander10" in amp2b.dsg is changed to "meander10m". Click at the "meander10m" module to select it. Click the right mouse button to bring up the pop-up menu. Select Module Properties command. MODUA will display its properties. The module is changed from .\ie3d\practice\meander10.geo to c:\ie3d\practice\meader10m.geo. Select OK to continue. The dialog is closed, but the module is still being selected. Click any empty spot to de-select the module.

Step 21 Select **Process->Simulate** command on MODUA. MODUA will prompt you to define the frequency points. Select Define All. Select OK. The Simulation Setup dialog comes up. Make sure AIF is enabled. Select OK to continue. The IE3D is invoked to perform the simulation on the optimized design. After the simulation, the simulation results are saved in amp2bm.sp. Please compare the results in amp2bm.sp and amp2.sp in .\ie3d\practice\output directory (see Figure 7.25).

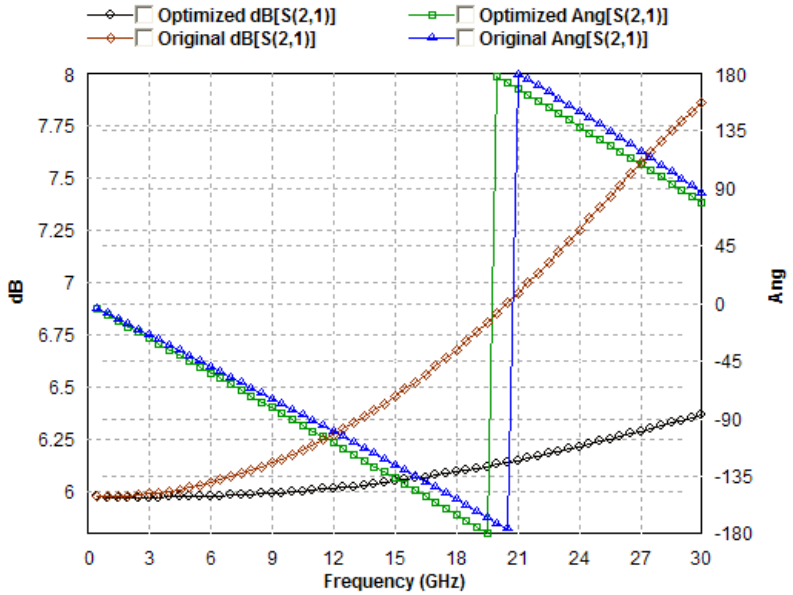
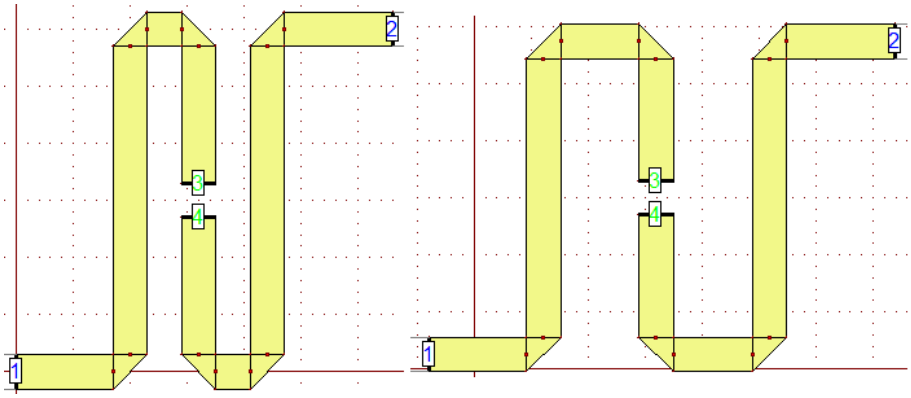


Figure 7.25 The comparison of the original and optimized s-parameters.



(a) Original Structure in meander10.geo (b) Optimized Structure in meander10m.geo

Figure 7.26 Comparison between the original and the optimized structures.

As you can see, the maximum dB[S(2,1)] is reduced from 7.85 dB to about 6.4 dB, while the Ang[S(2,1)] is almost kept unchanged. The physical length of the trace is not changed. However, the electrical length is slightly changed due to the change in the coupling caused by the change of shape. In fact, if we define the length of the segments as the 2nd variable, instead of mapping it to the 1st variable, we will even be able to optimize the length of the trace to keep the electrical length unchanged. Interested users may try it out.

You may want to know how much wider the gap width is. The optimized structure (meander10m.geo) is compared against the original geometry (meander10.geo) in Figure 7.26. You can see the gap width in the optimized geometry is obviously wider. By entering two vertices across the gap and using the Input->Info on Last Entry command on MGRID, we will be able to find the difference in the gap. The original gap width is 0.075 mm. The optimized gap width is 0.1727 mm. When you measure it, you have to remove the optimization variable. As long as you do not save the change, the optimization variable will still remain in the file.

Section 7.11 Modeling of Antennas with Lumped Element Loads

In the above examples, we have demonstrated how we can use MGRID+MODUA+IE3D to perform high accuracy and powerful mixed EM and nodal network simulations and optimizations on highly packed circuits with lumped elements. In practical design, such a feature is very important and useful for successful and high quality microwave circuit design. In fact, such a feature is also very important in the design of antennas with lumped element loads.

An antenna is a structure. Lumped element loads are circuit elements. To simulate it, we certainly need the feature of mixed EM and nodal network simulation. Furthermore, we may even want to display the current distribution and the radiation pattern in the loaded condition. The IE3D is well designed for such a purpose. We will demonstrate an example in the following.

Table 7.2 The parameters of the 2-probe fed patch antenna with finite ground plane.

Substrate Parameters	Thickness = 2.184 mm, ε_r = 2.33, tanδ = 0.0012
Patch Parameters	Circle Diameter = 40 mm, The 1st probe is 7.5-mm off the center on the x-axis. The 2nd probe is .7.5-mm off the center on the y-axis. Probe radius = 0.5 mm
Finite Ground	50-mm wide square

The example we are going to demonstrate is a 2-probe fed circular patch antenna with finite ground plane (see Figure 7.27). The parameters are listed in Table 7.2. The original structure is a 2-probe fed patch antenna published by J. T. Aberle and D. M. Pozar and C. R. Birtcher, IEEE Trans. Antennas Propagat., Vol. AP-39, Dec. 1991, pp.1691-1696. The example is documented in the IE3D Benchmark Example Brochure showing excellent agreement with the measured and simulated results from the literature. Here, we will create a finite ground plane version and use the 2-probes to demonstrate the capability and flexibility of the IE3D simulator. The infinite ground plane structure is saved in: .\ie3d\samples\pfed.geo and the results are saved in: .\ie3d\samples\output\pfed.sp.

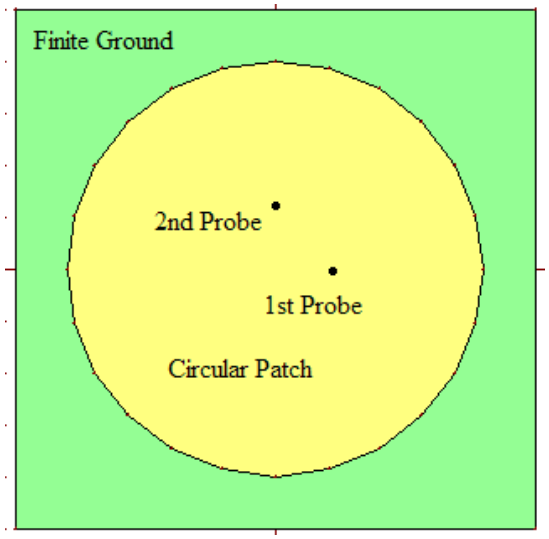


Figure 7.27 The 2-probe fed patch antenna.

Step 1 Run MGRID and open the file: .\ie3d\samples\pfed1.geo. It is the initial geometry with the basic parameters defined.

We are not going to do the Basic Parameters and the entry of the finite ground and the circle. We will use an existing file. It has a square on Z = 0 and a circle on Z = 2.1844 mm. No ground plane is defined. The substrate is from Z = 0 to Z = 2.1844. Interested users can check the Basic Parameters in Parameters menu. They will be exactly the substrate parameters documented in the Table 7.3. The square on Z = 0 will be used as the finite ground plane. It can be built using the Rectangle in Entity menu, or numerous other ways on the MGRID to build it. The circle can be created using the Circle in Entity menu. We are using 24-segments to approximate the circular patch. It should be accurate more than enough for most cases. If you try to re-build it using the Circle entity, remember to set the Style as Single Polygon. Otherwise, the circle will be created as a set of polygons instead of one single polygon. For our discussion in the next, we would like the circle to be a single polygon. In case it consists of a set of polygons, we can always use the Merge Selected Polygons command in the Adv Edit menu to merge them before we build the probe.

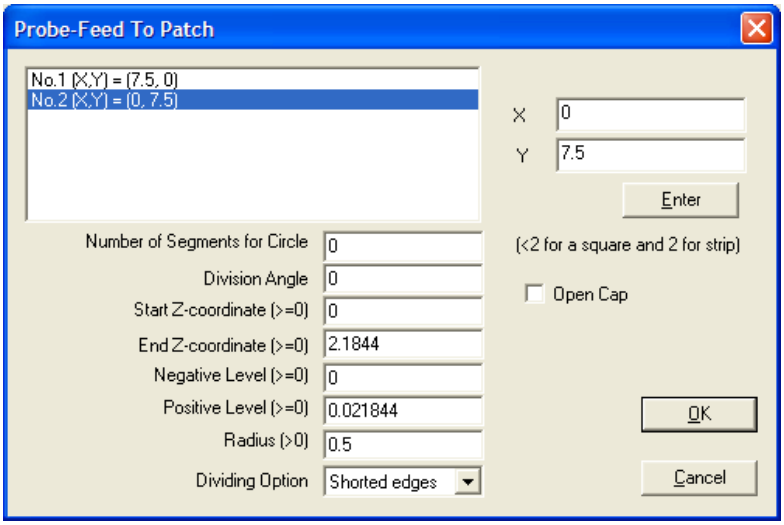


Figure 7.28 The dialog for the Probe-Feed to Patch command in Entity menu.

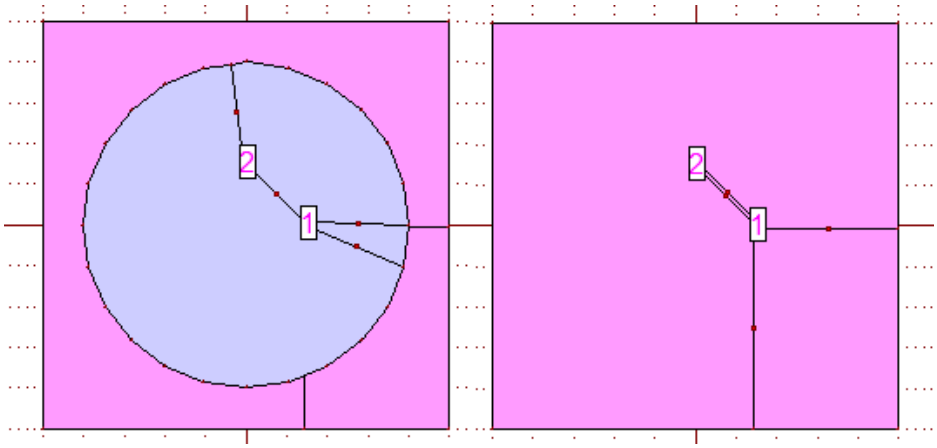
Step 2 Click at the Z = 2.1844 mm on the layer window to focus the input at Z = 2.1844. We are going to enter the locations of the 2-probes.

Step 3 Type Shift+A. It is equivalent to selecting Key In Absolute Location in Input menu. Enter the X = 7.5 mm and Y = 0. Select OK. The location of the 1st probe is entered at (X, Y) = (7.5, 0.0) as a vertex.

Step 4 Type Shift+A. Enter the X=0 mm and Y = 7.5. Select OK. The location of the 2nd probe is entered at (X, Y) = (0.0, 7.5) as the 2nd vertex.

Step 5 Select **Entity->Probe-Feed** to Patch command. The Probe-Feed to Patch dialog comes up (see Figure 7.28). Change the radius to 0.5.

We do not need to change other parameters. The default value for Number of Segments for Circle, Nseg = 0. As it is noted in the dialog, Nseg = 0 means that the probe will be of square shape. By default, the Start Z-Coordinate and Negative Level of the port is at Z = 0. The End Z-Coordinate is at 2.1844 mm and it is connecting to the patch. The default Positive Level is 0.021844 mm. It is 1% of the substrate thickness. This is a very good model for coaxial probe-fed patch antennas. Numerous examples have proved its accuracy when the probes are thin. You have the option to have an open cap at the location of the probe feed.



(a) top layer (b) bottom layer

Figure 7.29 The top and bottom layers of structure: pfed2.geo.

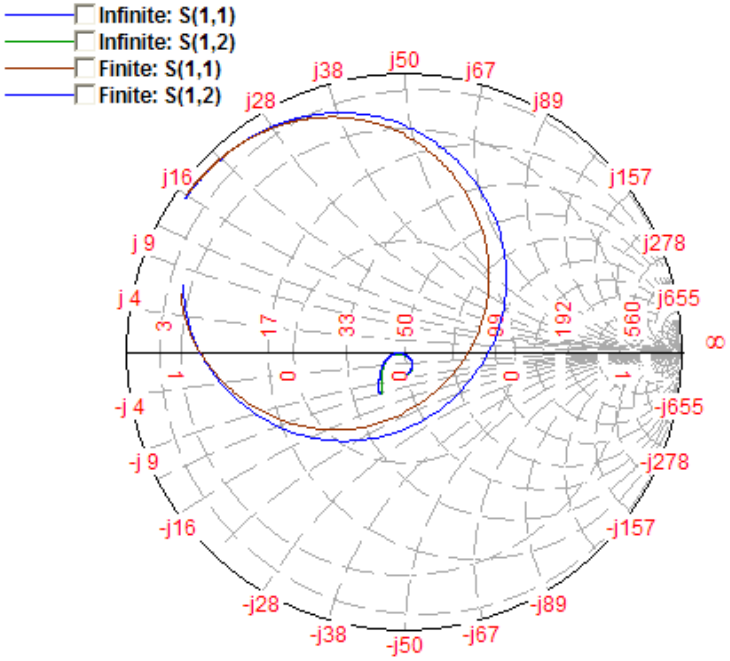


Figure 7.30 The comparison on s-parameters between the infinite and finite ground plane models.

Step 6 Select OK to continue. The 2-probe fed antenna is built and the top layer and the bottom layer of the structure are shown in Figure 7.29. Save the structure as: .\ie3d\practice\pfed2.geo.

As it is, the pfed2.geo is the structure ready to simulate. Ideally, it is a symmetric structure along the φ = 45 degrees across the origin. However, polygons are not symmetrical. We will expect the final meshing of the structure will not be completely symmetrical. The IE3D is a very stable simulator. Even the final meshing of the structure may be far from symmetrical, the 2-port s-parameters and the radiation patterns should be quite symmetrical. There are ways to make the polygons more symmetrical so that the meshing is more symmetrical. We will not discuss it here.

Step 9 Simulate the infinite ground plane model (.\ie3d\samples\pfed.geo) and the finite ground model (.\ie3d\practice\pfed2.geo). Compare the results in Smith Chart on S-parameters processing of MGRID (Process->S-Parameters and Lumped Equivalent Circuit).

Shown in Figure 7.30 is the comparison. The difference on the s-parameters between the infinite and finite ground plane models is quite small.

For the pfed2.geo, the ground plane size is only about 5 mm larger than the patch on each edge. The margin is slightly larger than twice of the substrate thickness. The s-parameters are not changed much. This example demonstrates that the s-parameters or input impedances of low profile antennas such as microstrip antennas are not very sensitive to the ground plane size. However, the shapes of the patterns will be affected much by the ground plane size. Apparently, the infinite ground plane model pfed.geo predicts no back radiation.

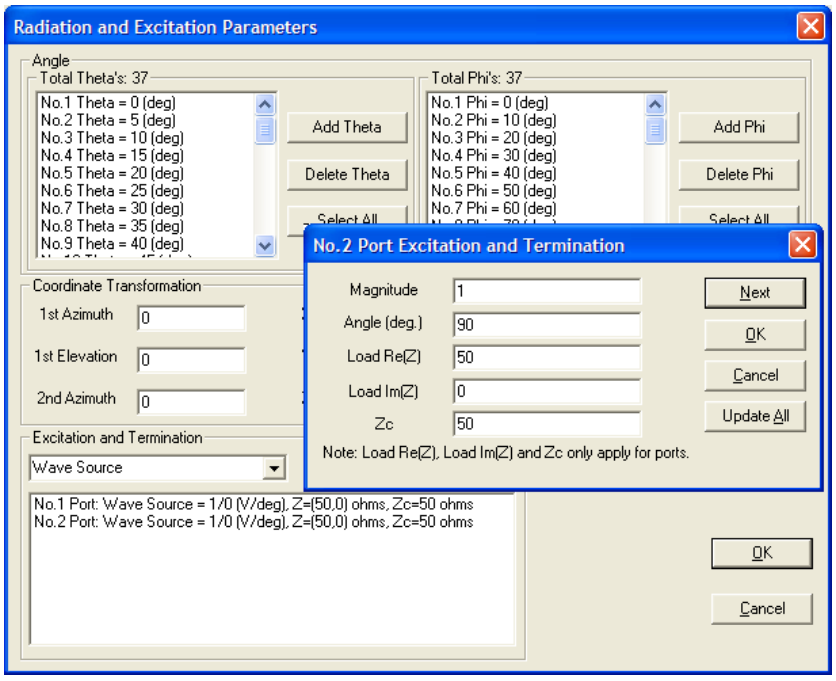


Figure 7.31 The Radiation and Excitation dialog.

Section 7.12 Quadrate Phasing Excitations on Two Probes for Circular Polarization

What is the use of the 2-probes? They can be used for many purposes. If we excite the 2-probes with same magnitude and 90-degree phase difference, we will be able to make a good circularly polarized antenna. We will demonstrate such an application first.

Step 1 Save the pfed2.geo as pfed3.geo. Select Process->Simulate command on MGRID. Please select Delete All to delete all the frequency points in the list box. Enter Start Freq = 2.66, End Freq = 2.90 and Number of Freq = 9. Select Enter to define the 9 frequency points. Disabled AIF. Select Radiation Pattern File. MGRID will prompt you for the pattern angles, Excitations and Terminations: The No.1 Port: Wave Source, Magnitude = 1 and Phase = 0. The No.2 Port is the same as No.1 Port. We need to change the No.2 Port.

Step 2 Double click at the No.2 Port in the source list to edit it. Change the Angle to 90 degrees (see Figure 7.31). Select OK to continue. We will have the phase of the No.2 port as 90 degrees in

the Radiation and Excitation dialog. Select OK to get back to Simulation Setup dialog. Select OK to finish the simulation setup.

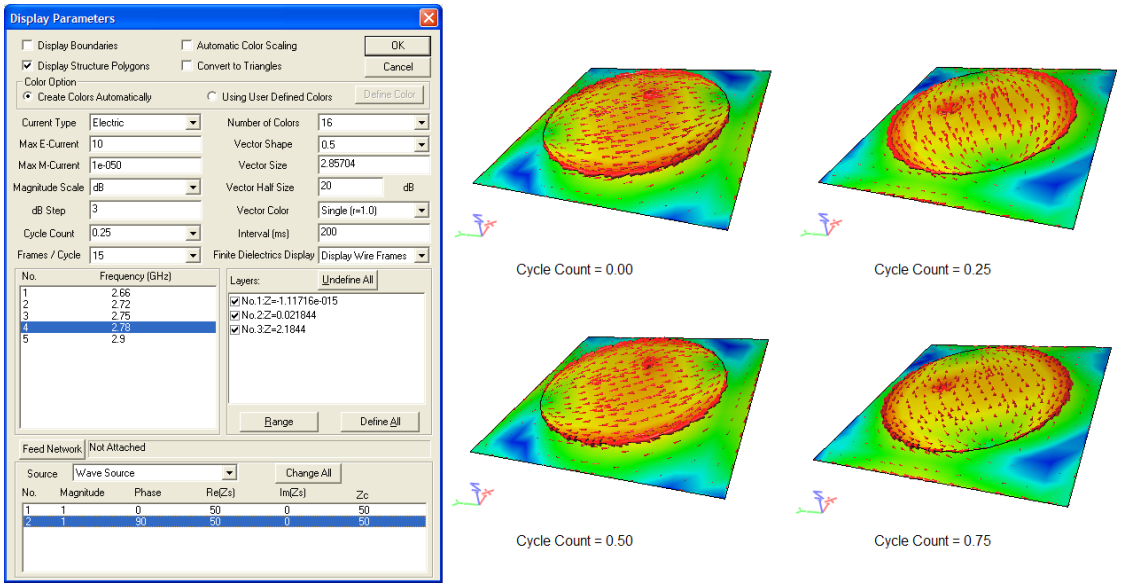
MGRID will issue you a warning, "Minimum forced meshing size happens… with the size 0.021844 mm (<0.026918)…". This warning is to let the users know that some small size polygons are encountered. They may create small cells which will affect the simulation results if the small cells are extremely small.

Normally, when we see such a message with the statement "…with the size X (<Y)…". We should check the X and Y values. Normally, when the X is not 2-3 orders smaller than Y, we do not need to worry about. If the X is 2-3 orders smaller than the Y, it is possible that the meshing on the structure may create irregular cells. The irregular cells may create singularity problem. For our case, the X = 0.021844 and Y = 0.026918. X is not much smaller than Y. We do not need to worry about.

Step 3 Select YES. MGRID will invoke the IE3D to perform the simulation. It takes seconds to finish the simulation. The s-parameters are saved into pfed3.sp as well as the geometry file. The s-parameters processing dialog should be invoked to display the s-parameters.

Please close the dialog for s-parameters because we would like to visualize the current distribution. Select Window->3D Current Distribution Display. The current distribution at the 1st frequency point or 2.66 GHz will be displayed. The default excitations may not be what we want.

Step 4 Select Options->Show 3D Vector Current. Select Options->Set Graph Parameters. Change the parameters to what are shown in Figure 7.32. We should get the vector current display for the Cycle Count = 0.25. Select Options->Set Graph Parameters. Change the Cycle Count to different value and you will see the vectors are rotating (see Figure 7.32). The rotating vectors with different Cycle Count indicate the polarization of the antenna is kind of circular polarized.



(a) Setup for Vector Current Display (b) Vector Current Display at Different Phases

Figure 7.32 The Current Distribution Display Parameters dialog and the displays.

Step 5 Select Options->Animate. MGRID will be animating the current and you will see the vector current changing. You can select Options->Set Graph Parameters to change the Frames / Cycle

to a larger value for smoother animation. Also, when the structure is larger, the update may be slow. You can select File->Save to Bitmap Files command to save the pictures into a series of bitmap files. The ZDibAnimator will be automatically invoked to do the animation. The animation on ZDibAnimator will be much smoother no matter how complicated the structure is because it does not need much time to process the frames. The bitmap files will be saved there for future animation using ZDibAnimator.

Step 6 Select Options->Animate again to stop the animation. Select Window->2D Radiation Pattern->Define 2D Pattern Plots. Select Ad Plot button in the Define 2D Pattern Plots dialog. Scroll to 2.78 GHz. Select E-left and E-right at Phi = 0 and Phi = 90 degrees (4-items). Select OK to add the plot into the list. Select Continue in the Define 2D Pattern Plot dialog to display the plot. The left hand and right hand circular polarization patterns on the 2-principal planes are shown in Figure 7.33. It is a very good left-hand circular polarization, which is guaranteed by the exact 90-degree out of phase excitations on the 2-symmetric ports.

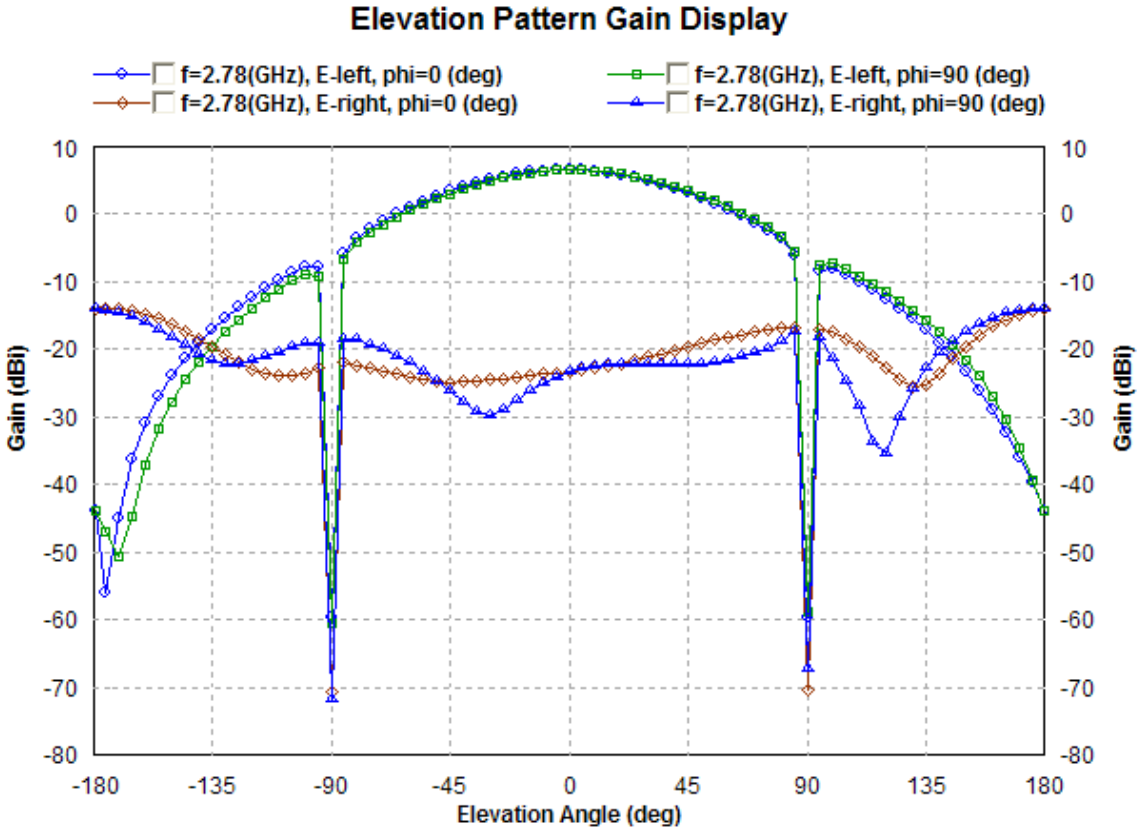


Figure 7.33 The left hand CP and right hand CP patterns.

Step 7 Select Window->Display Radiation Pattern Properties to bring up the Pattern List dialog. Select Detail Properties to show the detailed pattern properties on a dialog. Some selected data is shown in Table 7.3. Please be careful on the E-Total properties of an non-linearly polarized antenna. For a non-linearly polarized antenna, the E-Total Gain is the maximum possible gain if it is excited properly. Basically, we use $\sqrt{(|E\theta|^2 + |E\varphi|^2)}$ to calculate the E-Total field properties. The antenna is a very good LH (left-hand) circular polarized antenna with the specified excitations. If you use a LP (linear polarized) antenna to receive its signal or if you use it to receive a linearly polarized antenna, its maximum gain should be about 3dB smaller. Precisely,

E-Total pattern properties should not be considered as linear pattern properties for linear antennas. In standard antenna textbooks, linear antenna properties are based upon some specific cuts and some specific polarization (E-plane and H-plane, etc). The definitions of the cuts, E-plane and H-plane are quite arbitrary. It can even be different for the same antenna at different frequency ranges. We are unable to provide unified definitions for E-plane and H-plane for a general antenna on IE3D.

Table 7.3 The pattern properties of the circular polarized antenna at 2.78 GHz.

Linear Field Properties		Circular Field Properties	
Total Field Gain	6.44 dBi	LH Circular Field Gain	6.44 dBi
Total Field Directivity	7.30 dBi	LH Circular Field Directivity	7.30 dBi
Theta Field Gain	3.70 dBi	RH Circular Field Gain	-13.99 dBi
Theta Field Directivity	4.56 dBi	RH Circular Field Directivity	-13.13 dBi
Phi Field Gain	3.70 dBi		
Phi Field Directivity	4.56 dBi		

Section 7.13 Find the Current and Radiation Patterns with Lumped Elements and Feed Network

In practical applications, we may need to connect lumped elements into an antenna and find the current distribution and radiation pattern with the lumped element connected. Just take the 2-probe fed circular patch antenna as an example. In order to get good circular polarization, we need to feed the 2-probes with a phase difference of 90 degrees. We still need to use some transmission line to achieve the 90 degree phase difference. The implementation may not guarantee exact 90 degree phase difference, and it may not guarantee the magnitudes of the 2 feeds are exactly equal.

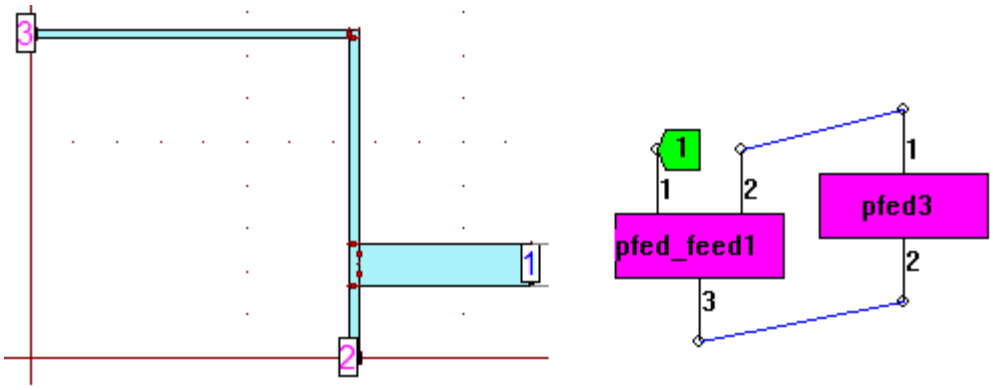
Saved in the .\ie3d\samples\pfed_feed1.geo is a power divider for such a case (see Figure 7.34a). Please understand that it is not an optimized one even though the phase difference between the 2 ports is about 90 degrees at about 2.85 GHz. We just try to use it to demonstrate the feature of the IE3D. We would like TO connect the port 2 of the divider to the port 1 of the antenna, and the port 3 of the divider to the port 2 of the antenna. We would like to know the current distribution on the antenna and the pattern of the antenna with the power divider feeding the antenna.

Step 1 Run MGRID and open the file: .\ie3d\samples\pfed_feed1.geo. Save it as: .\ie3d\practice\pfed_feed1.geo. Select **Process->Simulate** command. Select Capture button and select the .\ie3d\practice\output\pfed3.sp file to capture the frequency points from the file.

We will be finding the radiation pattern of the antenna in pfed3.geo fed by pfed_feed1.geo. We need to simulate the pfed_feed1.geo at the same frequency points.

Step 2 It is not necessary to save the current distribution and the pattern data of pfed_feed1.geo because its radiation is negligible. Please select OK to simulate the structure. It takes seconds.

Step 3 Please select File->New on MODUA. Select File->Add Parameter Module. Select .\ie3d\practice\output\pfed_feed1.sp into it. Click it somewhere to drop it. Click it anywhere to de-select it. Select **File->Add Parameter Module**. Select .\ie3d\practice\output\pfed3.sp file onto MODUA. Drop it somewhere at the right of "pfed_feed1". Click an empty spot to de-select it. Select **Element->Connection** to connect the terminal 2 of "pfed_feed1" to the terminal 1 of "pfed3". Connect the terminal 3 of "pfed_feed1" to the terminal 2 of "pfed3". Select Element->Exit Element to exit the mode. Select Element->Define All Ports to define the terminal 1 of "pfed_feed1" as the port 1. We will get the display in Figure. 7.34b. hat is shown in Figure 7.34 is the circuit connection between the antenna and the feed network.



(a) The 3-port power divider for the feed network (b) The connection to the antenna.

Figure 7.34 The feed network and the MODUA setup for connecting the antenna to the feed network.

Step 4 Please save the design as: .\ie3d\practice\pfed_ant.dsg file. Please select Process->Simulate on MODUA to simulate the s-parameters. Please select **File->Save S-Parameters** to save the 1-port s-parameters of the antenna fed by the feed network. The Smith Chart display of the 1-port s-parameters is shown in Figure 7.35 (only part of the unit circle). Accidentally, it is quite a good match even though the feed network is not optimized.

We are not only interested in the s-parameters of the antenna fed by the feed-network. We are also interested for the current distribution and the pattern of the antenna fed by the feed-network (or lumped element).

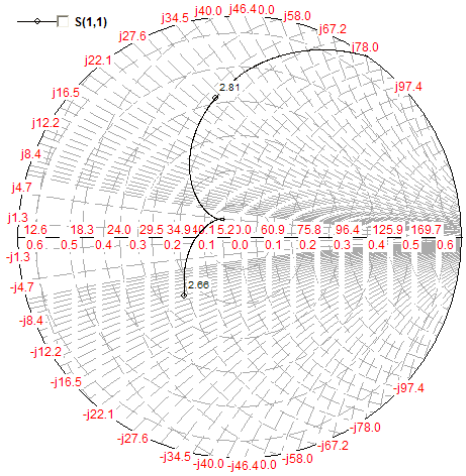


Figure 7.35 The 1-port s-parameters of the antenna fed by the feed network.

Step 5 Please select **Process->Simulate and Find Excitation** on MODUA. MODUA will simulate the design defined in Figure 7.34b. MODUA will prompt you for the excitation of the final 1-port circuit.

Step 6 Please select OK to accept the default settings. MODUA will list the excitations information on the antenna and the feed network on MODUA. Select File->Save Excitation to save the file into .\ie3d\practice\pfed_ant.ect. It will be opened in a Notepad. Listed in Table 7.4 is the excitation at one of the frequency points. Following is the explanation:

The No.1 Independent Module is the 2-port pfed3.sp. The No.2 Independent Module is the 3-port pfed_feed1.sp. The final circuit has one port. It is excited with 1-volt Incident Wave and it is terminated with Z0 = 50 ohms.

At 2.66 GHz, the Incident Wave is 1 V with Angle = 0 degree. The Reflected Wave is 0.20 V with Angle = -141.5 degrees. The Input Power at the port is 0.0096 W.

The 50-ohms Normalized IncidentWave at the terminal 1 of the No.1 Independent Module (pfed3.sp) is 0.65 V with Angle = -76.4 degrees. The Reflected Wave is 0.41 V with Angle = -1.2 degrees. The IncidentWave at the terminal 2 of pfed3.sp is 1.034 V with Angle = -121.9 degrees. The Reflected Wave is 0.61 V with Angle = -51.3 degrees.

The excitation information on each terminal of the pfed_feed1.sp is also listed.

We can take down the information at the 2 ports of pfed3.sp. If we enter the information when we display the current and calculate the radiation patterns for the pfed3.cur file, we will be able to find the current distribution and the radiation pattern of the antenna fed by the feed network. Actually, the process is fully automated with the pfed_ant.ect file we just saved.

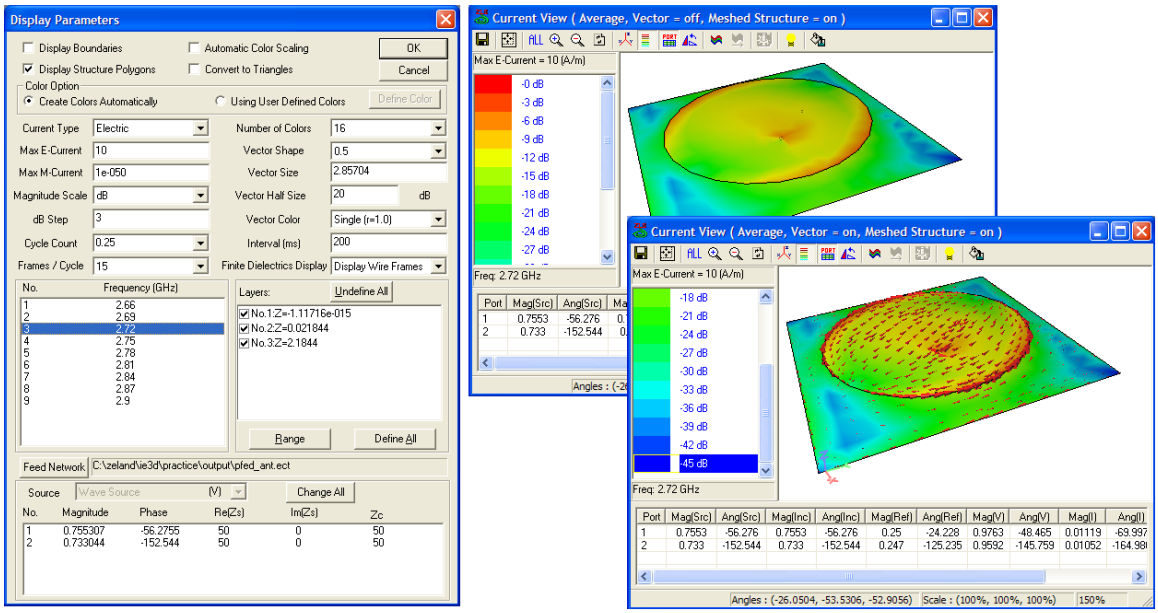
Table 7.4 The solved excitation information on the antenna and the feed network.

```
Version 10.0    Source: 1
Number of Ports = 1
Number of Independent Modules = 2
Ports for Independent Modules = 5
Number of Frequencies = 9

No. 1 Independent Module: Terminals = 3 Parameter File = C:\MentorGraphics\<latest_release>IE3D\SDD_HOME\IE3D\ie3d\practice\output\pfed_feed1.sp
No. 2 Independent Module: Terminals = 2 Parameter File = C:\MentorGraphics\<latest_release>IE3D\SDD_HOME\IE3D\ie3d\practice\output\pfed3.sp

Excitations and loads at the ports:

 No.1 Port: Incident Wave =          1 (V) /          0 (deg.), Z0 = ( 50 , 0 ) (ohms)  Zc = 50 (ohms)

Excitations and loads at ports and independent module's terminals:

 No.1 Frequency =          2.66 (GHz):
  Port Information:
   No.1 Port: IncidentWave =          1 (V) /          0 (Deg)   ReflectedWave =      0.201786 (V) /     -141.4775 (Deg)   P = (    0.009592824,   -0.002513533 ) (W)
  Terminal Information:
   No. 1 Independent Module:
    No.1 Terminal: IncidentWave50-Ohms =          1 (V) /          0 (Deg)   ReflectedWave50-Ohms =      0.201786 (V) /     -141.4775 (Deg)   P = (    0.009592824,   -0.001256766 ) (W)
    No.2 Terminal: IncidentWave50-Ohms =     0.4130299 (V) /     -1.214398 (Deg)   ReflectedWave50-Ohms =     0.6508889 (V) /     -76.40099 (Deg)   P = (   -0.002530626,   -0.002599014 ) (W)
    No.3 Terminal: IncidentWave50-Ohms =     0.6147143 (V) /     -51.31195 (Deg)   ReflectedWave50-Ohms =      1.034365 (V) /     -121.8635 (Deg)   P = (   -0.006920369,   -0.005995586 ) (W)
   No. 2 Independent Module:
    No.1 Terminal: IncidentWave50-Ohms =     0.6508889 (V) /     -76.40099 (Deg)   ReflectedWave50-Ohms =     0.4130299 (V) /     -1.214398 (Deg)   P = (    0.002530626,    0.002599014 ) (W)
    No.2 Terminal: IncidentWave50-Ohms =      1.034365 (V) /     -121.8635 (Deg)   ReflectedWave50-Ohms =     0.6147143 (V) /     -51.31195 (Deg)   P = (    0.006920369,    0.005995586 ) (W)
```

Step 7 While we are displaying the current distribution on MGRID, select Options->Set Graph Parameters. Select Feed Network button. Select the excitation file we just created (.\ie3d\practice\output\pfed_ant.ect). The excitation information file is imported. The final excitation information is listed in the dialog (see Figure 7.36a). Those are the final excitations on the 2 ports of the antenna.

Step 8 Please select OK to continue. MGRID will display the current distribution with the exact excitation from the feed network (pfed_ant.ect) (see Figure 7.36b). Please note the excitation information listed at the bottom of the 3D view showing the current. They are the exact values saved into the .ect file. Please note that the excitations displayed are the ports of the antenna structure. No excitation information on the feed network is displayed. You will need to check the excitation information on the feed network from the .ect file.

By the way, in case you want to use Voltage Source to excite the antenna, before you save the .ect file on MODUA in Step 6, you should select Control->Change Excitation command on MODUA to change the excitation source to the appropriate one. Then, you can save the .ect file with the specified excitation source type. The excitation can not be changed after the .ect file is saved.



(a) Set Graph Parameters (b) Average and Vector Current Distribution Displayed

Figure 7.36 The Set Graph Parameters and displays with attached Feed Network file.

Step 9 Select Window->Display Radiation Pattern Properties to bring up the Pattern List dialog. Select Properties to bring up the General Pattern Properties dialog. You should see the excitations of the antenna are still the ones you defined in Simulation Setup for pfed3.geo.

Step 10 Select Feed Network File button. Select the file: .\ie3d\practice\output\pfed_ant.ect. The 2-port excitations of the antenna are updated with the data from the feed network file or .ect file.

Step 11 Select the downward arrow on the right hand side of the combo box at the middle to drop down the available options there (see Figure 7.37).

We have the options for: (1) Find Pattern of Structure with Specified Excitations; (2) Find Pattern of Antenna Including Feed Network with Specified Excitations; (3) Display Detailed Pattern Properties. For option (3), it is display the detailed pattern properties discussed. We would like to clarify the options (1) and (2).

For this antenna, we can consider the excitations are the 2-ports of the structure if we consider the feed network is not part of the system. This is the option (1). We can also consider the excitations are the 1-port of the design created on MODUA. In some sense, we consider the 2-port antenna and the feed network are parts of the system. The excitations are the 1-port of the final circuit or antenna. It is the option (2).

What are the differences between the two options? The radiation pattern's shape, the radiated power and the antenna directivity will be completely the same because the current on the structure is exactly the same for the two cases and the radiation from the feed network is neglected. However, the incident power, the input power, antenna gain, efficiency and any parameters related to the feed or excitations will be different because the excitations for the two options are not the same even though the working condition is the same.

Before IE3D 14, we only have the option (1) and we are unable to find the radiation pattern of the complete antenna including the feed network unless we calculate the radiation pattern with geometry modules on MODUA. Starting from IE3D 14, you have the option.

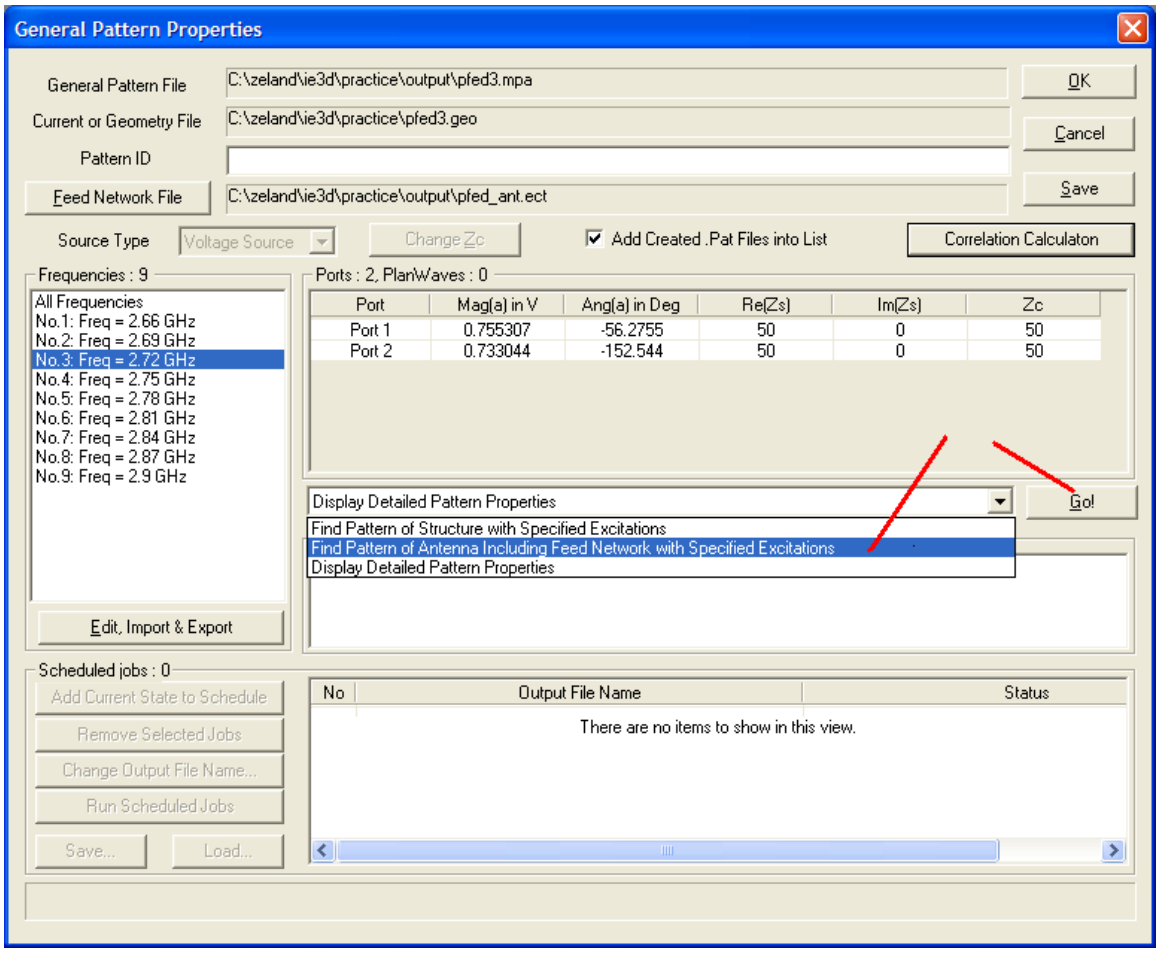


Figure 7.37 The current distribution with the feed network attached.

Step 12 Select the option (2) or "Find Pattern of Antenna Including Feed Network with Specified Excitations" and select "Go!". MGRID will prompt you the pattern file name to be saved. The default is "pfed3_ect.pat".

Select Save button and the pattern with the .ect excited will be saved. The created .pat file is added into a list in the dialog.

Step 13 Select the option (1) and select "Go!". Change the pattern file name as "pfed3_2ports.pat". Select Save button to save the pattern with the 2-ports as excitations into file. It will also be listed in the list.

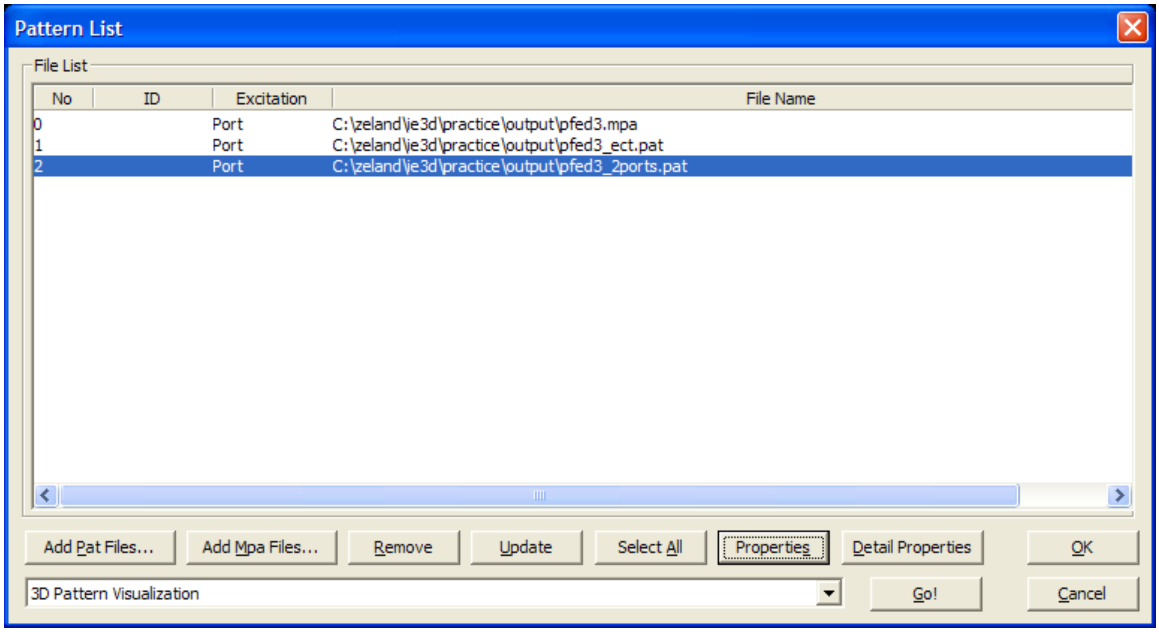


Figure 7.38 The Pattern List with three pattern files in the list.

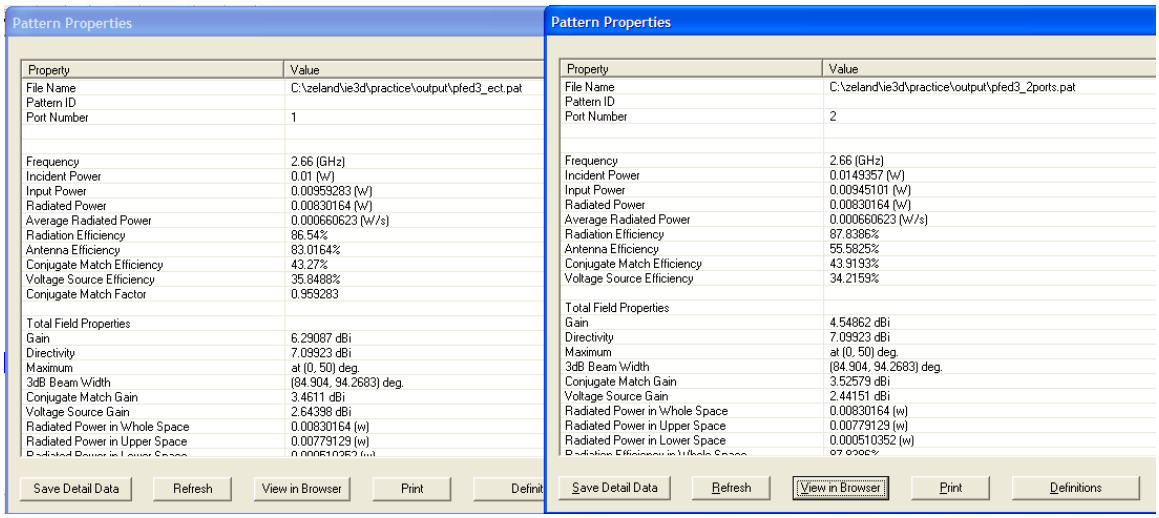


Figure 7.39 The comparison of the two generated patterns.

Step 14 Select OK to go back to Pattern List dialog (see Figure 7.38). There are three pattern files in the list. The No.0 file is the original general pattern file: pfed3.mpa for the 2-port structure. Its excitations are specified in the simulation setup while they are changeable in the Properties

dialog. The No.1 file is the 1-port pattern of the antenna including the feed network. The No.2 file is the 2-port pattern of the antenna with the excitations determined by the circuit connections defined in the pfed_ant.dsg file. The pattern properties of the No.1 and No.2 files are compared in Figure 7.39. They are for the same antenna while feeds at different locations are considered as the excitations of the antennas. Any parameters related to ports (input power, gain, etc.) are different while those parameters are not related to ports (radiated power, directivity, etc.) are the same.

Section 7.14 Simulation and Optimization of Radiation Pattern of a Circuit Consisting of Structures

In the last section, we demonstrate how we can find the current distribution and radiation pattern of a structure when it is fed by a feed-network. Starting from IE3D 14, we are able to find the radiation pattern of the antenna including the feed network as a system. Such a functionality was only available on MODUA with mixed EM and circuit co-simulation. However, there are still situations the new feature on IE3D 14 still can't do. For example, if we have multiple radiators fed by the feed network, what should we do? We will not be able to find the combined radiation pattern of the multiple sub-structures because the above way is only good for finding the radiation pattern of one structure with one or multiple lumped elements. With one radiator and multiple lumped elements, we are able to find the radiation pattern of the complete antenna using IE3D 14. However, we are still unable to do optimization of patterns involving structures and lumped elements. However, we are still able to use MODUA to perform mixed EM and circuit co-simulation and optimization. We will use the pfed3.geo and pfed_feed1.geo as our example. For example, we want to find the radiation pattern of the complete structure including pfed3.geo and pfed_feed1.geo. We will expect the radiation from pfed_feed1.geo to be very small and we will get the almost identical result what we got in Section 13. However, we just try to demonstrate how we can handle the case when we have multiple geometry files.

Step 1 Run MODUA and start a new design. Please select **File->Add Geometry Module** command. Please select the file: pfed_feed1.geo file to add it into MODUA.

Step 2 Please select **File->Add Geometry Module** and add the pfed3.geo file into MODUA too.

Step 3 Please connect the terminals of the geometry modules similar to what is shown in pfed_ant.dsg in Figure 7.34b except both files are not s-parameter files, but geometry files. In fact, you can replace the s-parameter file modules with the corresponding geometry modules in the pfed_ant.dsg. (1) Open the pfed_ant.dsg file on MODUA. (2) Click at the pfed3.sp module to select it. (3) Select Edit->Module Properties. (4) Select Replace Current Module with a Geometry Module and provide the pfed3.geo file. (5) Repeat (2)-(4) with the pfed_feed1.geo replacing pfed_feed1.sp module.

Step 4 Select **File->Save As** to save the design as: c:\ie3d\practice\pfed_ant2.dsg file.

Step 5 Select **Process->Simulate** command on MODUA. MODUA will prompt you for the frequency points. If your MODUA is from the previous run, the 9 frequency points should be still listed in the list box. Otherwise, please select Capture button to capture the frequency points from pfed3.sp file. Select OK to continue. MODUA will prompt you for the Simulation Setup dialog.

Step 6 Please un-check the AIF. Select Radiation Pattern File. MODUA will prompt you for the pattern angles. Please change the source from Wave Source to Voltage Source. Select the No.1 Port or the only port in the list. Change the Magnitude from 1 to 2. Click Update to update the listbox. Select OK to get back to Simulation Setup dialog. We should get the dialog shown in Figure 7.40.

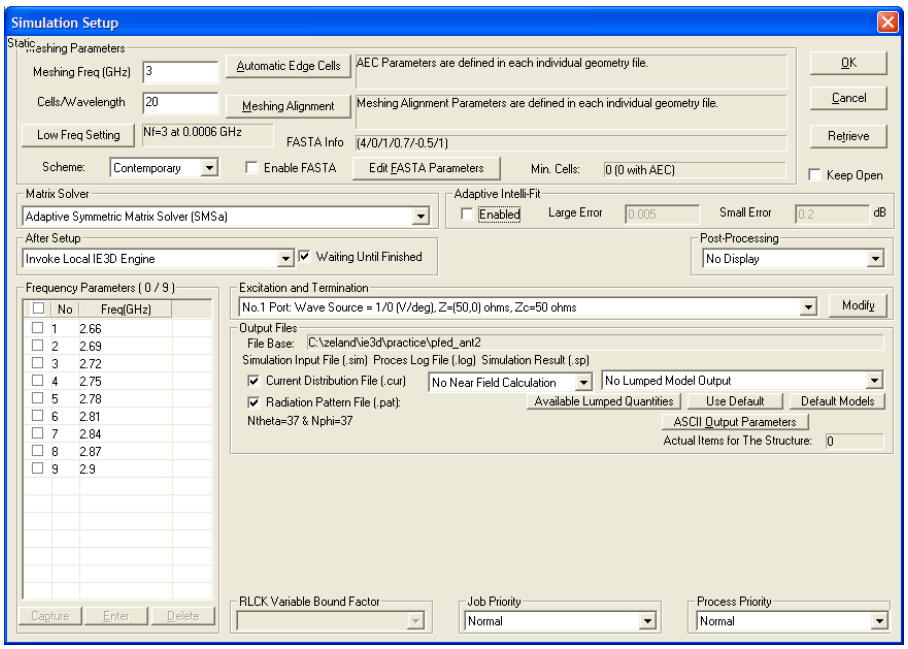


Figure 7.40 The Simulation Setup on MODUA for pattern calculation.

Step 7 Please select OK to continue. MODUA will invoke the IE3D engine to perform the simulation and pattern calculation. It will take a few minutes to finish it. Most of the time is spent on calculating the patterns at the 9 frequency points. After the simulation, the IE3D engine disappears. If you are interested in the intermediate simulation information, you can open the pfed_ant2.log file for it.

Table 7.6 The comparison of the two approaches at 2.72 GHz

Approach	Attach the .ect file from MODUA as feed-network to pfed3.cur	Build a circuit with geometry files pfed3.geo and pfed_feed1.geo on MODUA for complete simulation
File	Pfed3_ect.pat	Pfed_ant2.pat
Radiation Efficiency (%)	86.78	87.47
Antenna Efficiency (%)	86.46	87.15
Total Field Gain (dBi)	6.58	6.62
Total Field Directivity (dBi)	7.21	7.21
Total Field Maximum at	(0, 40) degrees	(0, 310) degrees
Total Field 3 dB Beam Width	(86.83, 88.95) degrees	(86.97, 88.75) degrees
RH Circular Gain (dBi)	6.56	6.59
Circular Directivity (dBi)	7.19	7.19
Circular Maximum at	(0, 220) degrees	(0, 310) degrees
3 dB Beam Width	(86.85, 88.27) degrees	(86.87, 88.07) degrees
Number of Ports	1	1

Step 8 We can use MGRID or PATTERNVIEW to visualize the pattern properties. To use PATTERNVIEW, Select **Process->Display Radiation Pattern** command. Select .\ie3d\practice\pfed_ant2.pat file. PATTERNVIEW will be invoked and the pfed_ant2.pat file is inserted into the list. This is a fast way to setup pattern display on PATTERNVIEW. We can run the PATTERNVIEW directly. However, we have to select **Edit->Add Pattern** command to add the pfed_ant2.pat file into the list. Select **Edit->Pattern Properties** command on

PATTERNVIEW to check the pattern properties. Certainly, you can always use Window->Display Radiation Pattern Properties to access the Pattern List dialog for displaying the pattern properties on MGRID. The pattern properties are compared against the data in pfed_ect.pat file in Table 7.6.

They are virtually the same. The only difference is that pfed_ant2.pat file contains the small radiation from the feed network which is negligible. Following are some more comments: (1). The Field Maximum = (0, 40) degrees means that the Field Maximum occurs at Theta = 0 and Phi = 40 degrees. When Theta = 0 or 180 degrees, Phi is not critical because Theta = 0 is the +z direction and Theta = 180 is the –z direction, independent of Phi value. We don't need to worry about the big difference in the Phi values when Theta = 0. (2). For the 3dB Beam Width parameter, we provided two values such as (86.83, 88.95). They are the minimum and maximum.

The feature discussed in this section is extremely powerful in designing antennas including feed network or multiple elements.

Section 7.15 Handling Structures with Differential Lumped Elements

We have demonstrated how we can simulate circuit and antenna structures with lumped elements. IE3D has implemented powerful schemes to solve tough EM and circuit co-simulation and optimization problems.

In Sections 1 to 10, we have demonstrated how we can simulate a structure with an amplifier. We try to replace the 2-port amplifier with two localized ports and performed EM and circuit co-simulation and optimization. It can yield accurate results.

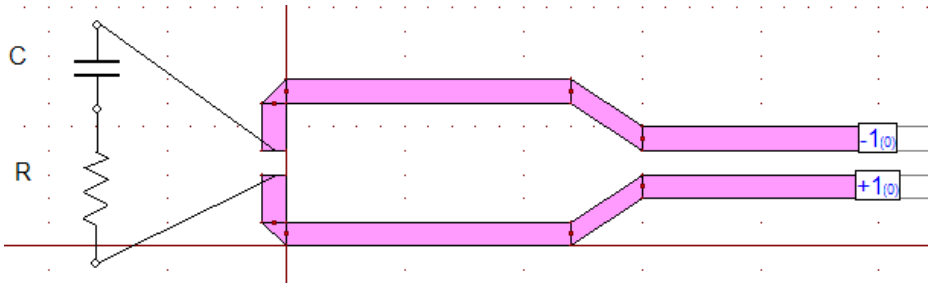


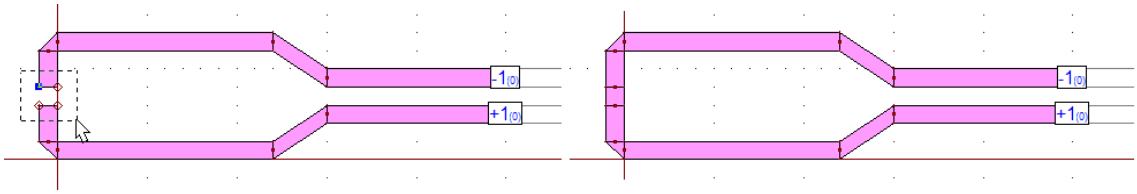
Figure 7.41 The structure with gap for series R and C lumped elements.

In some cases, we may not be able to define two localized ports for a 2-port lumped element. Saved in .\ie3d\samples\gap_for_diff_lumped.geo is a good example. It is illustrated in Figure 7.41 except there are no lumped elements in the layout. We would like to define some port(s) at the gap so that we can connect the series R and C into the circuit. Such case may happen much in surface mount technology (SMT) or other applications.

Can we define two ports on the structure as we did in Section 1? For this particular example, there is no ground plane. The traces are on one side of the substrate while the back side does not have a ground plane. We are unable to define the Localized MMIC port of Vertical Localized port for such a structure.

Is it necessary to define two ports at the gap? It is really not necessary. Please note the series R and C is basically a 1-port, 2-terminal lumped circuit when looking at the two terminals of the series R and C circuit. If we define a differential port across the gap on the structure, we will be able to connect the lumped

R and C elements to the differential port. We will demonstrate how we can achieve it and we assume the R = 10 ohms and C = 1 pF.



(a) Select the 4 vertices of the 2 edges (b) Fill the gap by Adv Edit->Build Connecting Path.

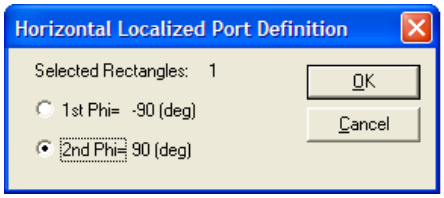
Figure 7.42 Build a connecting path to fill the gap for an H-localized port.

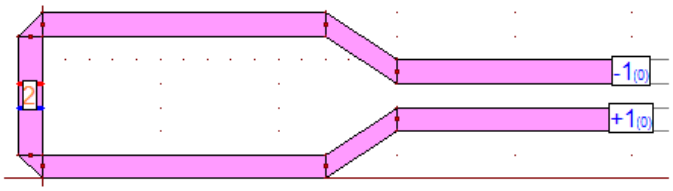
Step 1 Run MGRID and open .\ie3d\sample\gap_for_diff_lumped.geo. Save it into .\ie3d\practice directory. We will modify it for the port(s) in different way.

Step 2 Select Edit->Select Vertices. Focus selection on Z = 0.1 or where the strip is located. Window the two edges (or the 4 vertices) to select them (see Figure 7.42a). Select Adv Edit->Build Connecting Path to fill the gap. Please save the file as: .\ie3d\sample\filled_gap_for_diff_lumped.geo. We will use the intermediate geometry for later discussion.

Step 3 Select Port->Port for Edge Group. Check Horizontal Localized scheme. Select OK. Window the rectangle filling the gap. MGRID will prompt you for the Horizontal Localized Port Definition (see Figure 7.43a). You have two options for the polarity. For this example, it may not be critical. Just pick "2nd Phi = 90" and select OK and the port is defined (see Figure 7.43b).

You will see a red line indicating the positive terminal at the top of the port 2 and a blue line indicating the negative terminal at the bottom of port 2.





(a) Option for polarity (b) H-localized port defined.

Figure 7.43 Defining an H-localized port.

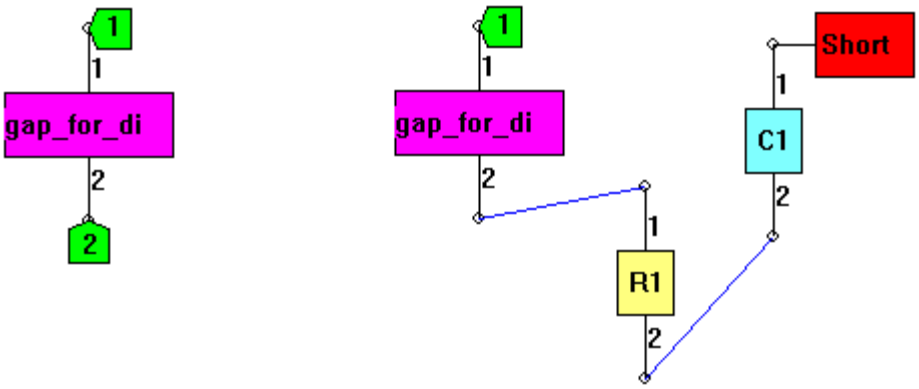
Step 4 Select Port->Exit Port. Save it as: .\ie3d\practice\gap_for_diff_lumped_1port.geo. Simulate it from 0.2 to 40 GHz and it will take seconds. If you have not chosen to invoke MODUA to display the s-parameters, please select Process->Display->Display S-Parameters on MGIRD and choose .\ie3d\practice\output\gap_for_diff_lumped_1port.sp to display the s-parameters. We are going to use MODUA to connect the lumped R and C to it.

Step 5 While MODUA is displaying the s-parameters; please select CONTROL->DISPLAY TOGGLE to get back to the schematic view (see Figure 7.44a). It is displaying a black box of 2-ports to you.

MODUA use one terminal to represent 1-port. We can only see one terminal on port 2. Howe can we connect the two terminals of the series R and C to port 2?

Step 6 Please click at the port 2 symbol on MODUA. Hit Delete button to delete it. Select Element->Capacitor. Enter "1" for 1-pF for the capacitance. Select OK and drop it somewhere on the window. Select Element->Resistor. Enter "10" for 10-ohms. Select OK and drop it somewhere on the window. Select Element->Connection. Click at the terminal 2 (or port 2) of the s-parameter

module. Click at the terminal 1 of R1 to connect them. Click at the terminal 2 of R1 and terminal 2 of C1 to connect them. Click at an empty spot on the window and select NO to confirm getting out of the Connection mode. Select Element->Short. Click the right mouse button once or multiple times to rotate the orientation of the Short on mouse. The orientation does not change the results but just the look. Click at the terminal 1 of C1 to connect the Short to it. Click at an empty spot and select NO to confirm getting out of the Short mode. We will get what is shown in Figure 7.44b.



(a) The 2-port black box for the structure. (b) The setup for connecting the series R and C.
Figure 7.44 The setup on MODUA to connect a 1-port lumped element.

Section 7.16 Matching of Single-Ended and Differential Ports

The way discussed in Section 14 seems to me a very simple way. It should yield excellent results for lumped elements with differential ports.

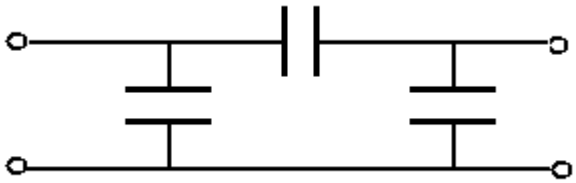


Figure 7.45 A PI-network of capacitors.

You may ask whether we can connect the 2-ports of an amplifier in Figure 7.1 in the same way as what we did in Figure 7.44. You may also ask whether we can connect the 2-port PI-network shown in Figure 7.45 in the same way as what we did in Figure 7.44. The answer to the questions is "No". What makes the difference is the number of terminals. Even though it is a 2-port s-parameters data, an amplifier actually contains three terminals. Same is true for a PI-network. We can't consider the 2-port s-parameters data of an amplifier or a PI-network as a 1-port, 2-terminal s-parameters data as we do for the series R and C. Whether the 2-ports can be considered as one differential port of 2-terminal is the only criteria on whether we can connect the lumped element in the way we discussed in Section 14.

The above discussion brings up another very important topic. Can we use circuit simulator freely without considering the port types? We can't. When we connect lumped elements together, it is the user's responsibility to make sure the ports match. If we connect port A of element X to port B of element Y, we need to make sure the port A and port B are either single-ended or differential ports simultaneous. We can't mix-use the single-ended port with differential port in a connection. Otherwise, we will end up with incorrect results without being notified because a circuit simulator does not know which port is a differential port and which port is a single-ended port.

Some circuit simulators use 2-terminals to represent 1-port while MODUA and MDSPICE use 1-terminal to represent a port. Both ways have advantages and disadvantages. If you don't use them properly, you will get incorrect results.

On MODUA, we allow a Short to be shared by elements with single ended s-parameters. However, if the elements are differential elements, you should be careful not to share a Short between elements. A Short represents connection to the negative terminal or return path of a differential port. Differential differential ports have different negative terminals and return paths and we should not connect them together.

Section 7.17 Single-Ended to Differential Conversion

When we have 2-ports, we are able to convert the 2-ports to a differential port. You can do it on MODUA or on the s-parameter processing utility on MGRID. When we have one differential port, can we convert the one differential port into two single-ended ports? On IE3D, there is no need to do it. The question comes from the fact that some circuit simulators may not be able to accept differential s-parameters. They use 2-terminals to represent 1-port. For the 2-port structure discussed in Section 14, they can't accept the one differential port to connect the two terminals of the series R and C to. They may require the differential port 2 to be explicitly represented as two single-ended ports (port 2 and port 3) in order to connect the series R and C to the two terminals. We can achieve it using either MODUA or MGRID.

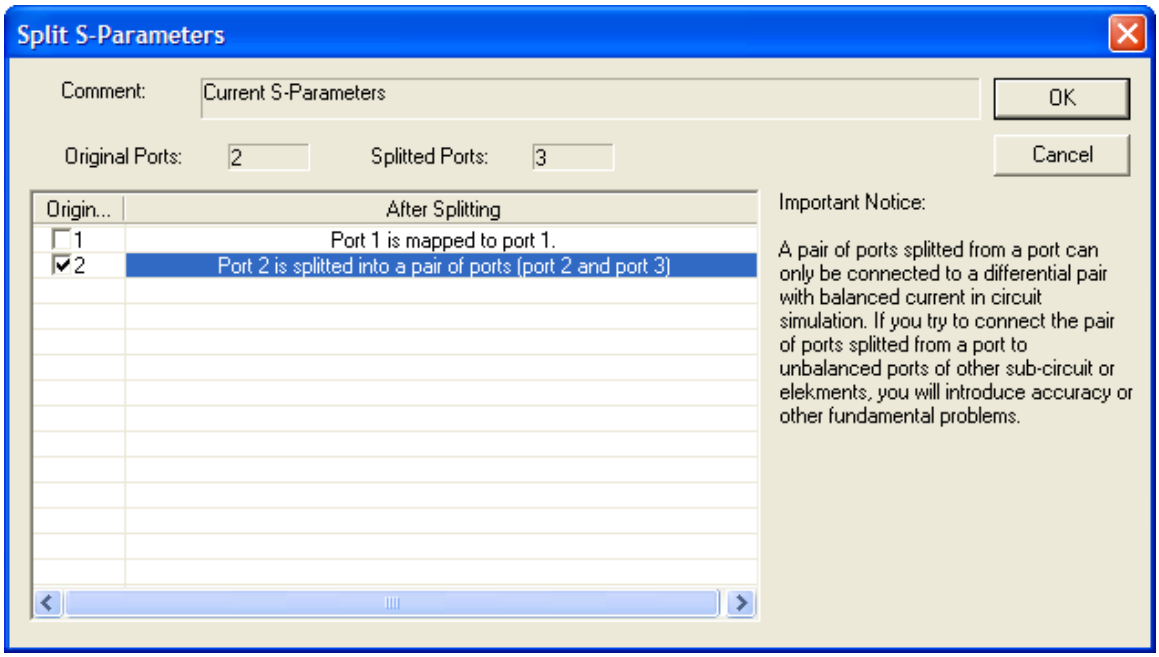


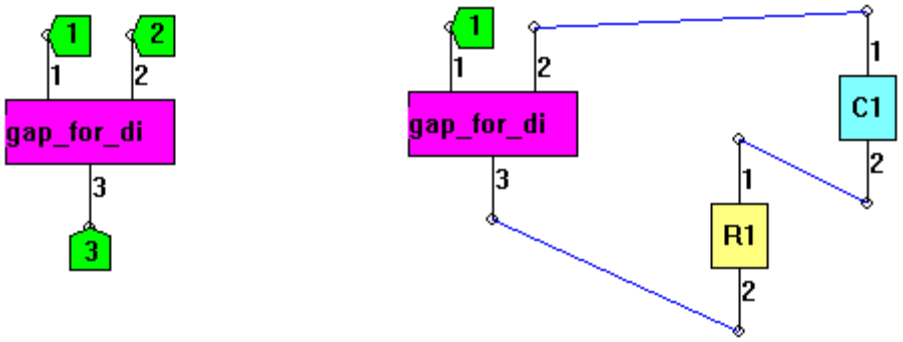
Figure 7.46 The differential to single-ended conversion setup.

Step 1 Run MODUA. Select File->Display Parameter Module. Select gap_for_diff_lumped_1port.sp or the 2-port s-parameters data created in Section 14. MODUA will display the s-parameters.

Step 2 Select Process->Split Differential Ports to bring up the "Split S-Parameters" dialog. Check the Original Port "2" (see Figure 7.46). The port 2 will be splitted into a pair of single-ended ports or port 2 and port 3 after the conversion. Select OK. MODUA will finish the conversion real-time. It prompts you to save the file into: gap_for_diff_lumped_1port_split.sp. Select Save to save the results into a 3-port s-parameters data.

Step 3 Select File->Display Parameter Module. Select gap_for_diff_lumped_1port_split.sp to display the results of the differential to single-ended conversion. Select Control->Display Toggle. You will see the 3-port schematic representation (see Figure 7.47a).

Step 4 Delete the ports 2 and 3. Connect the R1 and C1 as shown in Figure 7.47b. Save the design as: gap_for_diff_lumped_1port_split_connections.dsg. Select Process->Simulate to simulate the design. Select File->Save S-Parameters to save the results into file. Compare the results in gap_for_diff_lumped_1port_connections.sp, obtained from differential 1-port connection, and the results in gap_for_diff_lumped_1port_split_connection.sp, obtained from single-ended 2-port connection. You will see they yield identical results.

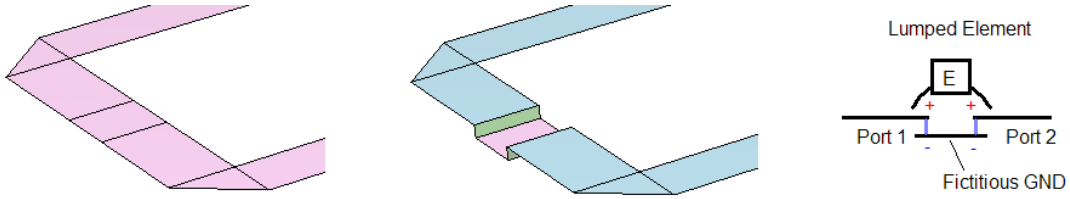


(a) The 3-port s-parameters (b) Connections on the R1 and C1

Figure 7.47 The MODUA setup for connecting the R1 and C1 to two single-ended ports.

Section 7.18 Using Fictitious Local Grounds for Structures with Lumped Elements

Another approach to connect the series R and C to the structure is to use fictitious ground. We demonstrated how to model the gap as a differential port and connect the series R and C to the differential port as a 1-port circuit. We also demonstrate how we can convert the differential port into two single-ended ports so that we can connect the series R and C as 2-port element to the two single-ended ports on the structure. We are going to demonstrate how we can model the structure as a 3-port structure with ports 2 and port 3 as the connection ports to the series R and C.



(a) Before elevation (b) After elevation (c) Schematic with two V-ports

Figure 7.48 Elevate the rectangle to prepare for two V-localized ports with a fictitious ground.

Step 1 Run MGRID. Open the intermediate file created or filled_gap_for_diff_lumped.geo.

Step 2 Select the rectangle filling the gap. Select Edit->Change Z-Coordinate and change it from 0.1 to 0.095 mm. Make sure "Keep Polygon Connection" is checked. Select OK. We will obtain the structure in Figure 7.48b.

Step 3 Select Port->Port for Edge Group. Choose Vertical Localized port. Select the Positive Level = 0.1 mm, and the Negative Level = 0.095 mm. Select OK. Make sure the Z = 0.095 and 0.1 layers are checked. Window the two vertical rectangles in Figure 7.48b to define two V-localized ports

on them (see Figure 7.48c). The elevated rectangle is serving as a fictitious local ground for the two V-localized ports. Select Port->Exit Port. Save the geometry as: gap_for_diff_lumped_2Vports.geo. Simulate it to get the 3-port s-parameters. Do the connection on MODUA as shown in Figure 7.47b. Simulate and save the resulting as: gap_for_diff_lumped_2Vports_connection.sp.

Step 4 We can also divide the filled rectangle into three rectangles and define two H-localized ports in .\ie3d\samples\gap_for_diff_lumped_2Hports.geo. Simulate the structure. Then, connect the lumped elements as shown in Figure 7.47b. Simulate the connected design and save the results into gap_for_diff_lumped_2Hports_connection.sp.

Step 5 Compare the resulting 1-port s-parameters at port 1 with the lumped elements connected using the four approaches discussed in Sections 14 to 16. We can see they yield almost identical results (see Figure 7.49).

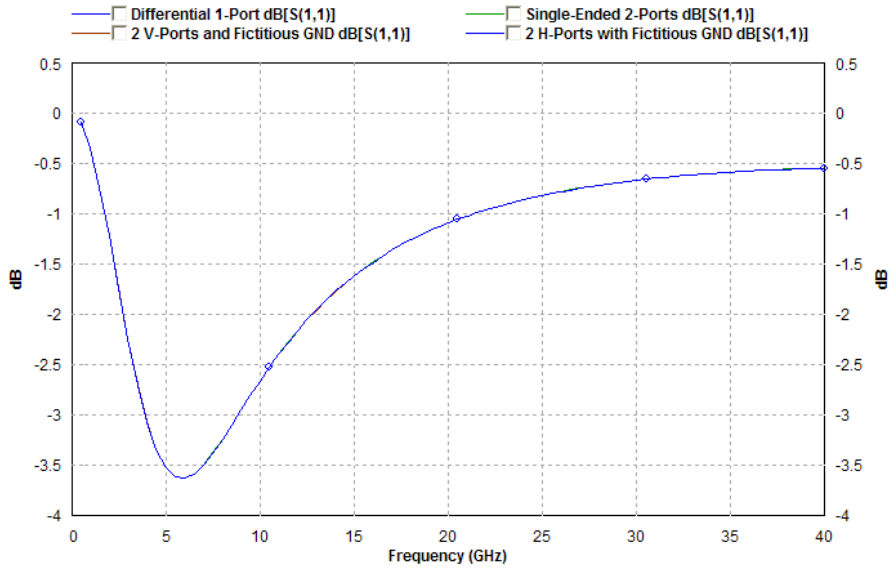


Figure 7.49 Comparison on the final 1-port s-parameters with different approaches.

We have discussed different ways of connecting 1-port, 2-terminal differential elements to the s-parameters of a structure to yield accurate results. The approaches are valid for N-port lumped element connections as long as 2-terminal in the lumped elements can be considered as a differential pair.

Are the any advantages for a specific approach over another approach? For the later three approaches, we have represented the structure as 3 ports with the ports 2 and 3 to be used for connections. Can you connect a general 2-port s-parameters data, such as an amplifier or a PI-network, to the ports 2 and port 3? If the answer is yes, the later three approaches should have an advantage over the differential port approach. Unfortunately, we can't use the ports 3 and 3 of the later three approaches freely. We still need to make sure we will connect the 2-terminals of a differential port of another lumped element to the ports 2 and 3 of the 3-port s-parameters data. Without meeting the requirements, the three-port s-parameters and any results derived mean nothing but incorrect results. In summary, all the different approaches yield the same results and they are equally flexible and they have the same limitations. You should understand the true meaning of connections in circuit simulations. When you are wiring up two ports from two different s-parameters blocks together, you should make sure the ports match, single-ended to single-ended and differential to differential. We will end this chapter here.

Chapter 8 Magnetic Current Modeling of Aperture Structures

So far, we have been discussing the modeling of electric current in electronic circuits. Basically, we are using polygons to represent the electric conductors and we are solving the current distribution on the electric conductors. There are circuits or antennas that utilizing slots in electric conductors to achieve specific performances. For such circuits or antennas, it would be better to model the tangential electric field across the slots instead of the current distribution on the electric conductors, because the area of slots is much smaller than the area of the conductors, and the computational effort will be significantly smaller. Theoretically, tangential electric field across a slot is equivalent to the magnetic current on the slot. We will call such a model as magnetic current model.

In this chapter, we will demonstrate how we can apply the magnetic current modeling to solve some problems efficiently. Magnetic current modeling is only available on IE3D PowerPack. For those users whose IE3D licenses do not have the magnetic current capability, you can choose to skip this chapter or you can follow it for practice. Following the examples can still help you to practice editing on MGRID. For those IE3D licenses without the magnetic current capability, no simulation involving magnetic current is allowed even you can still build it on MGRID. Therefore, you would not be able to create the current distribution file.

From the electromagnetic theory, the slot electric field is equivalent to the fictitious magnetic current distribution by Equation (8-1).

$$\boldsymbol{Jm} = \boldsymbol{E} \times \boldsymbol{n} \tag{8-1}$$

where $\boldsymbol{Jm}$ is the equivalent magnetic current density; $\boldsymbol{E}$ is the tangential electric field on a slot; $\boldsymbol{n}$ is the unit normal vector of the slot. A user should understand that the magnetic current model is based upon the assumption that the slot is on a flat, large and very good electric conductor. In reality, the electric conductor is always finite and its conductivity is always finite. The accuracy of the model will depend upon how big the ground plane is and how good the electric conductor is.

Section 8.1 Modeling a Slot Antenna with CPW-feed or Gap Port Feed

The first example for this chapter is a CPW fed antenna. It is published by H. S. Tsai and R. A. York, "Applications of planar multiple-slot antennas for impedance control, and analysis using FDTD with Berenger's PML method," *IEEE AP-Symposium Digest*, Newport Beach, CA, 1995, pp. 370-373. The structure is shown in Figure 8.1. The basic parameters are: Substrate Thickness = 0.813 mm, Permittivity = 2.2, Conductivity = 4.9×107 s/m, Slot Length = 78 mm, Width = 2 mm, Center Conductor Width = 2 mm. The colored polygon represents a slot on a ground plane even it looks like it is a metallic polygon.

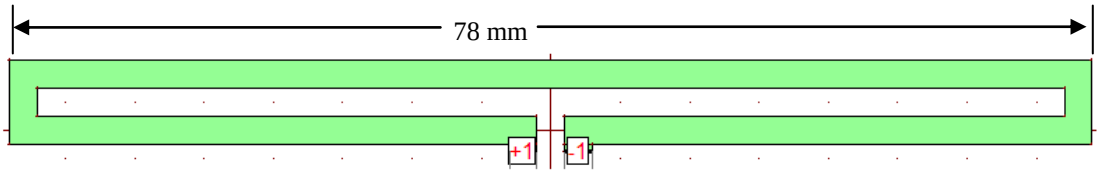


Figure 8.1 A CPW-fed slot antenna to be analyzed using magnetic current modeling.

Building the polygon to represent the slot should be simple with what you have learned. There are multiple ways to build it. You can build the slot as a path. You can also build the outer shape as a rectangle of 78 mm by 6 mm. Then, you build a smaller rectangle of 74 mm by 2 mm on another layer. You can use the smaller rectangle to build hole on the bigger rectangle for the shape in Figure 8.1. Certainly, you will need to do some extra work to define the ports 1 and 2 and its proximity. The structure is pre-built and saved in file: .\ie3d\samples\slot.geo. We will concentrate our discussion on the substrate setup and the

Layer Window. Figure 8.2 shows the layout, the Layer Window and the Basic Parameters. You may notice we have defined two substrate layers at Z = 0.813. Basically, the substrate setup means the following: From -∞ to 0 is air. From 0 to 0.813 mm is the substrate. From 0.813 to 0.813 mm is the ground plane. Then, from 0.813 mm to ∞ is air again.

It is not allowed to build a finite thick ground substrate layer on IE3D at this time. However, you can define one normal substrate layer between 2 infinitely thin ground planes to model the finite thick ground plane.

On the Layer Window, you see a horizontal dash line across the No.1 Layer at Z = 0.81300. It means that the polygon layer is an interface of ground plane. Any 2D polygon on it is assumed to be a slot on the ground. The polygons on the layout editor mean the area to be etched away from the ground plane.

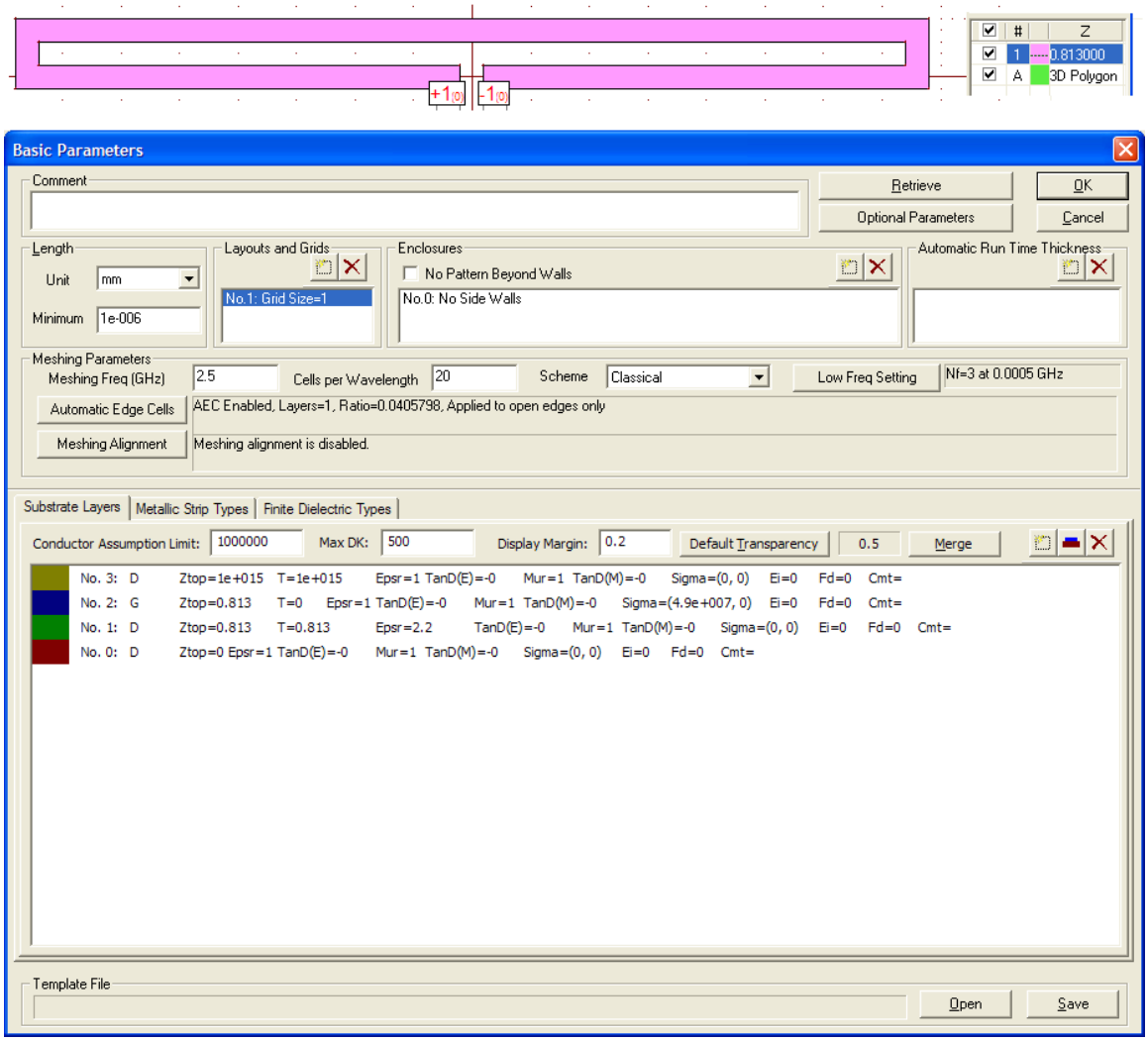


Figure 8.2 The structure, the Layer Window and the Basic Parameters dialog.

We have defined a differential pair of Extension for MMIC port on it. For magnetic current modeling Extension for MMIC scheme and the Advanced Extension scheme are the same. The question is: what does the differential pair mean? If the polygons are metallic polygons, the differential pair means it is a pair of

coupled strip. For magnetic current modeling, the differential pair means it is a CPW. We will explain the reason in Section 2.

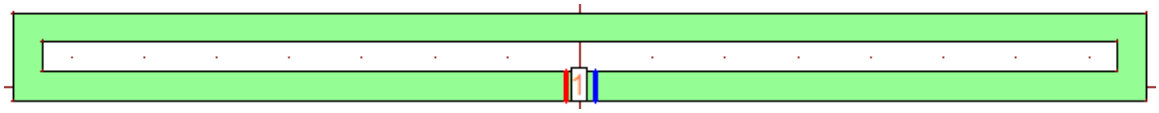


Figure 8.3 The slot antenna fed by a horizontal localized port.

Is this the only choice for a port on magnetic current modeling? It certainly is not. Saved in .\ie3d\samples\slot1.geo is the same geometry fed by a Horizontal Localized port.

A Horizontal Localized port is basically a gap port. If the polygons are metallic polygons, the structure in Figure 8.3 means a folded dipole with a gap port between the two terminals. For the magnetic current modeling, the polygons represent slots on the ground. The Horizontal Localized port means an excitation across the gap at the port location. More explanation can be found in Section 2.

How can we define the Horizontal Localized port on it? It is simple.

Step 1 Run MGRID. Open .\ie3d\samples\slot1.geo. Select Port->Delete All Ports. We will see the structure consists of two polygons (see Figure 8.4). Polygon 2 is the "folded dipole" while the polygon 1 is a rectangle with both ends connected to the ends of the "folded dipole".

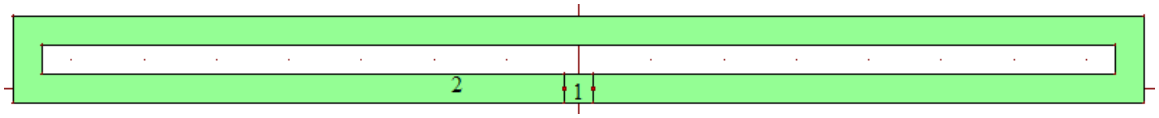


Figure 8.4 The slot1.geo without the port defined.

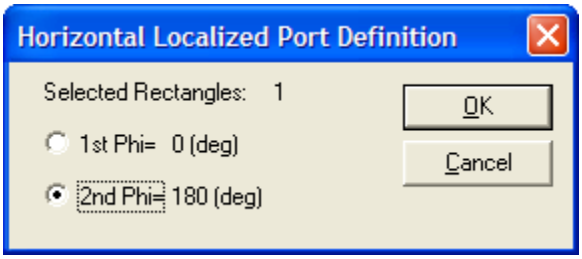


Figure 8.5 The Horizontal Localized Port definition dialog.

There are two ways we can define the Horizontal Localized port on it. We are going to show the both ways.

Step 2 Select Edit->Select Polygon Group. Window the rectangle 1 to select it. Select Port->Selected Rectangles for Horizontal Localized Port. MGRID will prompt you the Horizontal Localized Port definition dialog. It detects two possible directions for the port. We want the positive terminal to be at the left hand side of the gap and the negative terminal to be at the right hand side of the gap (see Figure 8.3). We should choose the 2nd Phi = 180 degree. Please note you may be prompted with "1st Phi = 180 and 2nd Phi = 0" in your run. You should choose the one corresponding toe the Phi = 180. Select OK to continue. We will get the port defined as shown in Figure 8.3.

If it is for electric current, the port defines a source to apply a voltage between the positive terminal and the negative terminal. For magnetic current modeling, there is an ambiguity for the

port direction. We will discuss this issue later in Section 2. Let's try the different way to define the Horizontal Localized Port.

Step 3 The Horizontal Localized Port scheme has been implemented a few years ago. The way to define a Horizontal Localized Port we just demonstrated was the only way. However, many users do not realize the existence of such a scheme because the way of defining it is different from other schemes. For this reason, we have implemented a way closer to defining ports in other schemes.

Please select Port->Delete All Ports to delete the port again. Select Port->Port for Edge Group. The De-Embedding Scheme dialog comes up (see Figure 8.6). Please choose Horizontal Localized. Select OK to continue.

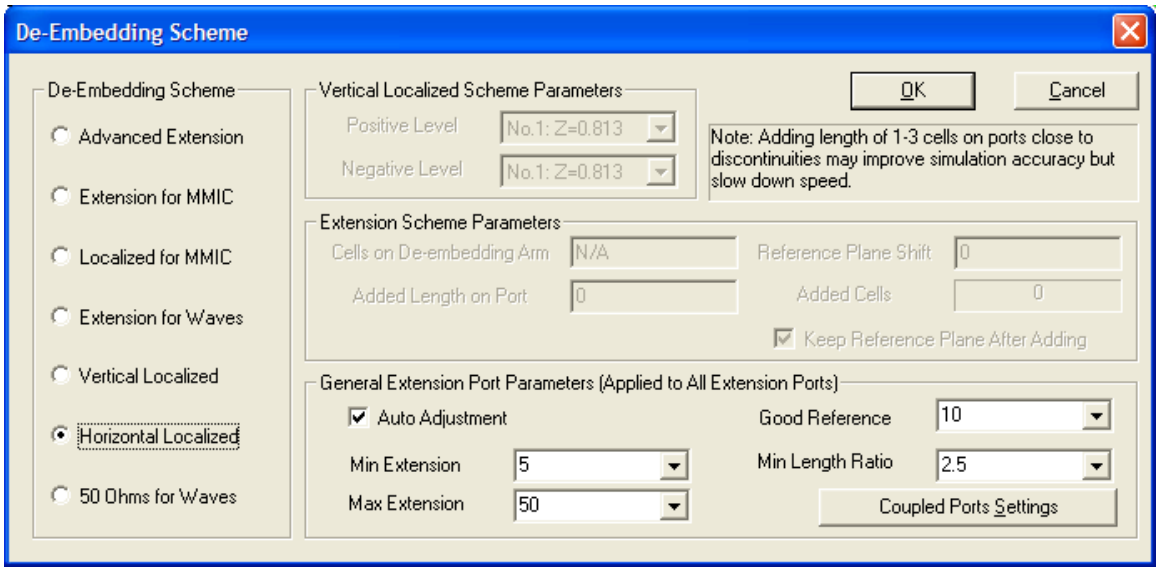


Figure 8.6 The De-Embedding Scheme dialog with Horizontal Localized scheme selected.

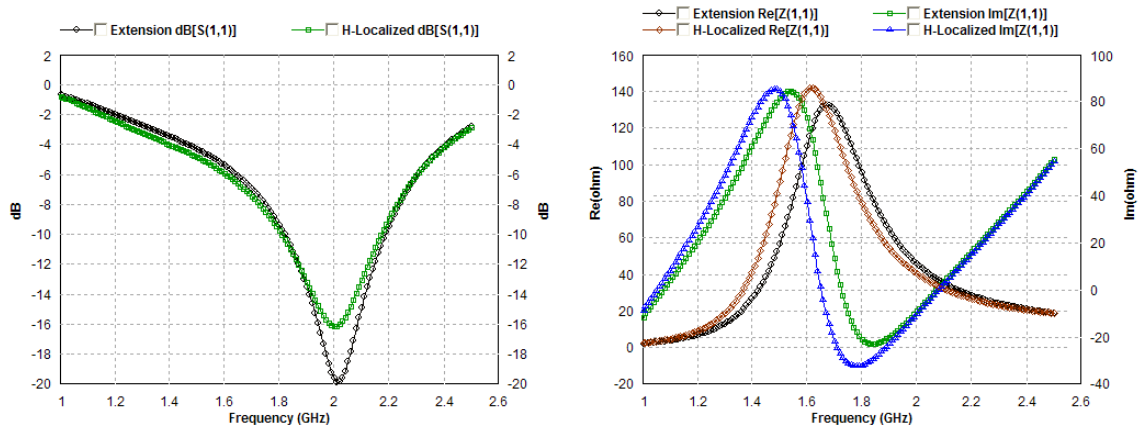


Figure 8.7 The comparison of the slot antenna with 2 different ports.

Step 4 Please window the polygon 1 in Figure 8.4. MGRID will prompt you for the Horizontal Localized Port Definition dialog in Figure 8.5. Select the 2nd Phi = 180 degrees and select OK. MGRID will define the Horizontal Localized port on slot1.geo. Select Port->Exit Port to exit the mode. Please simulate and compare the s- and z-parameters of the slot.geo and slot1.geo (see

Figure 8.7). The s-parameters are almost the same except there is some small difference at the dip. There is some slight shift in the z-parameter curves. It is due to the fact that the difference in the feed proximity causes some difference in the reactance part in the impedance.

Section 8.2 Discussion of Ports on Slot Structures

In Section 1, we have discussed how to model a slot antenna using differential feed schemes. However, what are the meanings of the schemes discussed. In this section, we will illustrate the different ports.

In the .\ie3d\practice\slot.geo, we are using differential Extension for MMIC port (see Figure 8.2). In fact, the "+1" port is applying a voltage between the metal on its right side to the metal on its left side. The "-1" port is applying a voltage between the metal on its left side to the metal on its right side. It is equivalent to applying a voltage between the center conductor and the two side conductors on the CPW structure (see Table 8.1). Basically, the two polygons are representing the two slots of CPW.

In the .\ie3d\practice\slot1.geo, we are using the Horizontal Localized port (see Table 8.1). What it means is that we are applying a voltage across the gap. It is equivalent to a gap port in the electric current model.

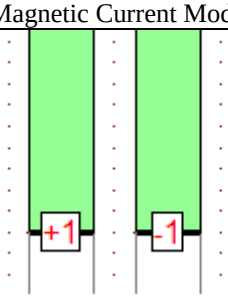
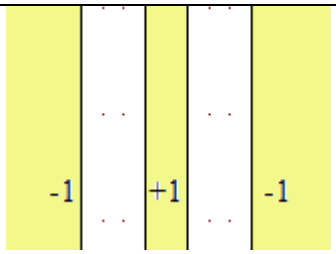
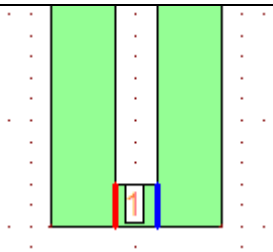
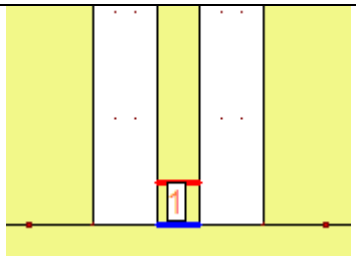
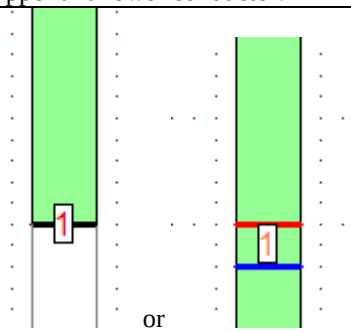
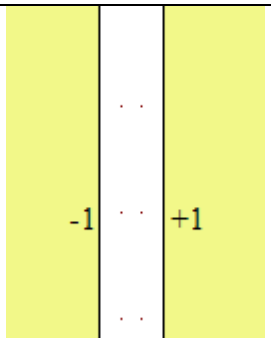
Certainly, the feed proximity is different between CPW feed and the Gap Port feed when we model the slot antenna. It is not surprising that they may yield slight different results in Figure 8.7.

Some users are considering whether we can define a single-ended port on a polygon for magnetic current modeling. They are considering that the single-ended port in "Excitation across a Gap" in Table 8.1 is equivalent to magnetic current Gap Port structure for the "Excitation across a Gap" in Table 8.1. In fact, they are not equivalent. The magnetic current Gap Port structure is equivalent to the eclectic current Gap Port structure in the "Excitation across a Gap" in Table 8.1.

Precisely, slotline of infinite ground plane should not be considered as a transmission line. We should not define a port on it like the single end port in "Excitation across a Gap" in Table 8.1. Defining a Horizontal Localized port in the "Excitation across a Gap" case in Table 8.1 is ok. It means that we are looking at the voltage and current across the slot. However, the single ended extension port is different. It implies that we are detecting the standing waves along the transmission line. Unfortunately, a slotline on an infinite ground is not a transmission line under the definition of waveguides. Interestingly, slot-lines of infinite ground are considered as a transmission lines in some textbooks in microwave theory. Some books even document the formulation for the characteristic impedance of slot-lines. In fact, it is misleading. From the field theory, we can prove that a slot-line with infinite ground plane cannot support the fundamental mode. Its modes are more like a spectrum. Following is the explanation.

We can consider a slot-line as two coupled strips. For a coupled strip structure, there is always a fundamental mode there. A slotline with infinite ground plane is basically a coupled strip structure with the strip width approaching infinity. However, when the strip width is increasing, the cut-off frequencies of the 2^{nd} order mode and other higher order modes of the coupled strips is decreasing. When the strip width is approaching infinity, the cut-off frequencies of the higher order modes are approaching 0. They are basically a spectrum. In the other word, a slotline with infinite ground plane cannot support a single fundamental mode. Microwave network theory is based upon the modes of the transmission lines. Apparently, microwave theory cannot be applied to slot-line structures. In the other word, if a slot-line structure is divided into 2 sections, we measure the s-parameter of each section. Then, we cascade the s-parameters of the 2 sections. The result will not agree with the measurement of the slot-line structure as one single structure.

Table 8.1 Comparison between electric current model and magnetic current model.

Port Structure	Magnetic Current Model	Electric Current Model
CPW Port	 The +1 excites a voltage between the CTR and the LHS. The -1 excites a voltage a voltage between the CTR and the RHS.	 A voltage is excited between the CTR and both sides.
Gap Port	 A voltage is excited between the upper and lower conductor.	 A voltage is excited between the upper and lower conductors.
Excitation across a Gap	 or The Gap Port (or H-Localized port) excites a voltage between the RHS and the LHS. However, the single ended port is invalid. It looks like it is exciting a voltage across the gap. However, it has some source at the end of the port extension (or slot extension).	 A voltage is excited across the gap from RHS to LHS.

Another proof for the fact that a slotline cannot support a fundamental mode is the following. Consider we have a single metallic strip in the air without any ground plane. We know we cannot use it as a transmission line because it does not support any mode since the ground plane is at infinity. Such a fact is quite obvious. Using the dual theory, we can find that the mathematical solution of a single slot on an infinite ground plane is the same as that of a strip in the air. It means that a single slot cannot support a fundamental mode.

Although we can use a slot-line as a waveguide to transmit high frequency energy, slotlines are used in microwave circuits much. A typical example is a slotline coupled to two microstrip lines (see Figure 8.8).

It is absolutely acceptable to define the two ports on the microstrip lines. However, if you want to break the slot-line at the center into 2 sections, so that you can analyze each section individually, you are wrong. You cannot break the slot-line at the center and define ports there. If you do it and connect the s-parameters of the two sections for the s-parameters of the whole structure, you will get incorrect results.

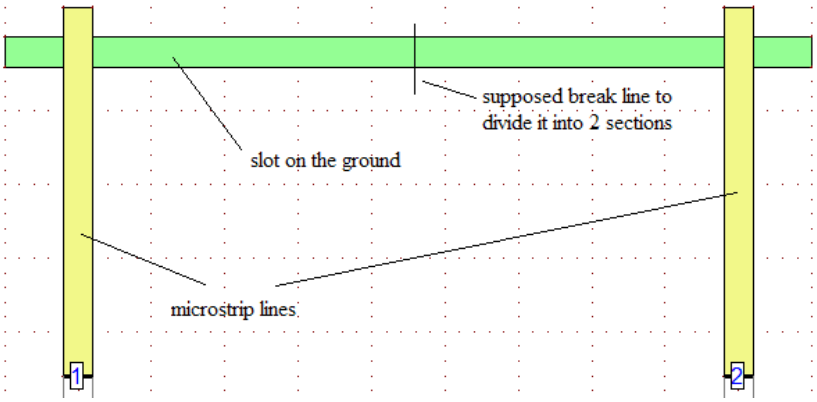


Figure 8.8 A slot-line coupled to two-microstripline structure.

Slot antennas and circuits normally yield wider bandwidth. However, accurate modeling of slot structures may need more attention. Discussion can be found in the last section of this chapter.

Section 8.3 Modeling of Aperture Coupled Fed Patch Antennas

We discussed modeling of CPW-fed slot antennas in the above sections. In the modeling of the CPW-fed slot antenna, we only used magnetic current modeling. In the following example, we will illustrate an example of modeling aperture coupled fed patch antenna using both electric and magnetic current modelings.

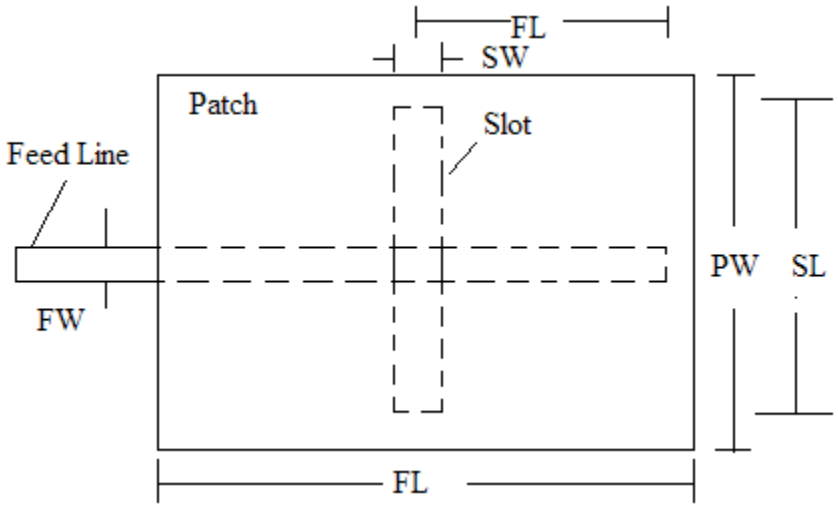


Figure 8.9 The aperture coupled fed antenna from literature.

The structure we are going to analyze is from: David M. Pozar, "A reciprocity method of analysis for printed slot and slot-coupled microstrip antennas", *IEEE Trans. Antennas Propaga*., Vol. AP-34, no. 12, pp. 1439-1446, Dec. 1986. Its dimensions are shown in Figure 8.9. The patch is on a substrate with εr = 2.54 and thickness of 0.16 cm. The substrate is on top of a ground plane. There is a slot on the ground plane. The slot and the patch have the common center. There is another substrate with the same parameters on the back of the ground plane. On the bottom of the back substrate is the feed line. The patch dimensions

re PL = 4 cm and PW = 3 cm. The slot sizes are SL = 1.12 cm, SW = 0.155 cm. The feed line dimensions are FL = 2 cm and FW = 0.442 cm. The reference plane is at the center of the slot. We will not show how to build it because it is quite straightforward.

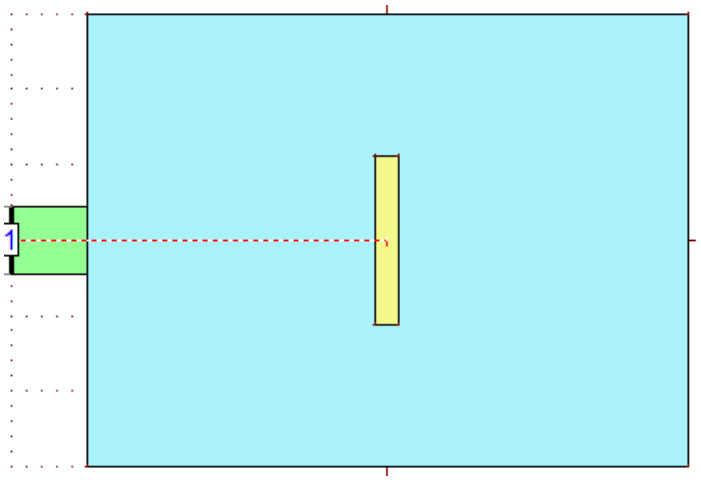


Figure 8.10 The aperture coupled fed antenna in .\ie3d\samples\aperpat.geo.

Step 1 Run MGRID. Open **.\ie3d\samples\aperpat.geo**. Save it into **.\ie3d\practice\aperpat.geo**. The structure is shown in Figure 8.10.

Please check the substrate parameters to see how we setup them. Please pay attention to a line from the port to the center of the slot. It indicates a shift of reference plane of the port. We will demonstrate how we define it.

Step 2 Select **Port->Delete All Ports** command. Select **Port->Port For Edge Group** command. Select the Advanced Extension scheme. Window the end of the feed line on the left to re-define it.

Port 1 is defined at the same location. However, there is no line from it to the slot center because the reference plane of port 1 is at where it is defined.

Step 3 Select **Port->Define Reference Plane** command. Move the mouse toward the center of the slot. Click the left mouse button. MGRID prompts you for the **Reference Plane to Port Distance**.

Step 4 Enter "-25" mm for the **Reference Plane to Port Distance**. Select **OK** to continue.

The distance is always negative toward the circuit and positive toward the source.

Step 5 Select **Port->Exit Port** command to exit. Simulate the structure from 2.15 to 2.30 GHz with 16 frequency points. Please enable the AEC, disable AIF and check "Current Distribution File" because we want to visualize the current distribution at all the frequency points. It takes a short time to finish the simulation. The results are saved as **.\ie3d\practice\output\aperpat.sp** and the current data will be saved into **.\ie3d\practice\output\aperpat.cur**. The s-parameters are shown Figure 8.11. The result is close to those presented in the literature.

Figure 8.12 shows the bottom view of the meshed structure. There are 3 layers in the structure. The No.1 layer is the feed line on Z = 0. The No.2 layer is the slot on Z = 1.6 and the No.3 layer is the patch on Z = 3.2. The Z = 0 and Z = 3.2 layers are the electric current layers. The Z = 1.6 is the magnetic current layer.

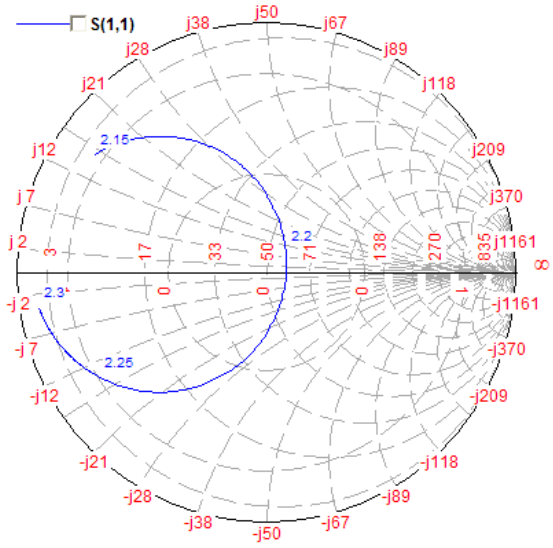


Figure 8.11 Simulation result on the aperture coupled fed antenna.

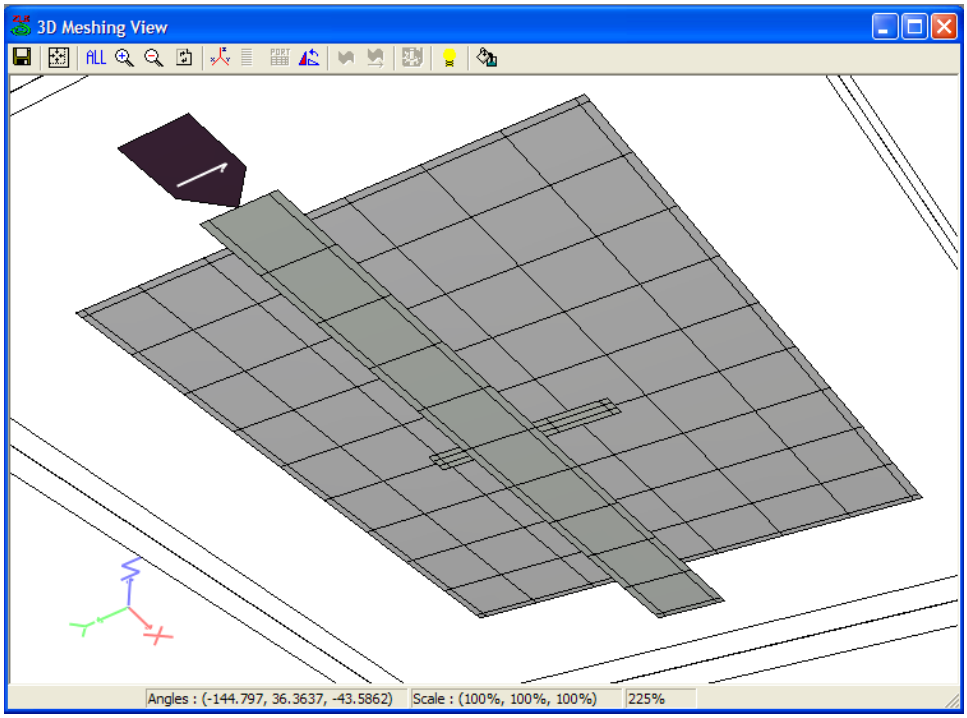


Figure 8.12 The 3 layers in the meshed structure on MGRID.

Step 5 On the MGRID, select Window->3D Current Distribution Display to open a window for the current distribution. Select Options->Set Graph Parameters to change the frequency and the Current Type (see Figure 8.13). By default, MGRID is display the E-Current or Electric Current (see Figure 8.14a). You can change it to M-Current or Magnetic Current (see Figure 8.14b) or Both (see Figure 8.14c). When you choose Both, both the max E-current and max M-current will be displayed on the color scale. The electric current density is in A/m, and the magnetic current density is in V/m. You can also choose different layers to be displayed (see Figure 8.13).

If you have done pattern calculation on the structure, you can also select Window->3D Radiation Pattern Display command. You are able to display the radiation pattern and the current distribution data simultaneously on IE3D V14 (see Figure 8.15).

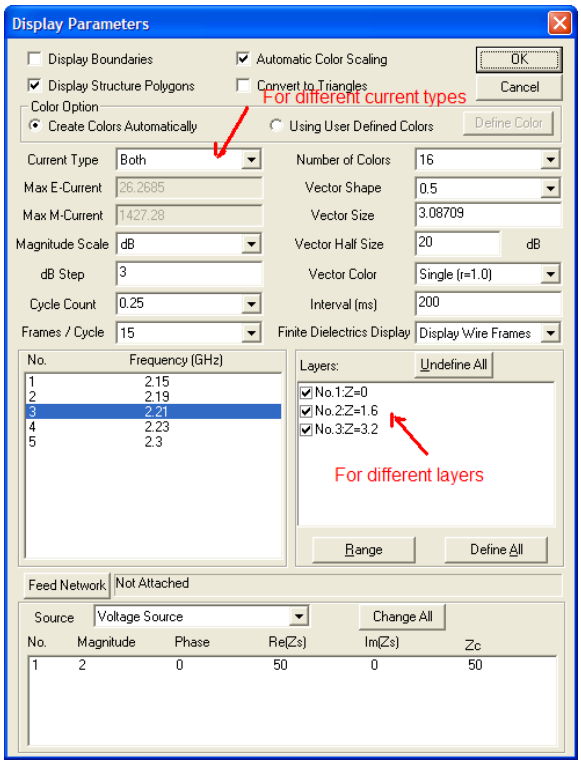
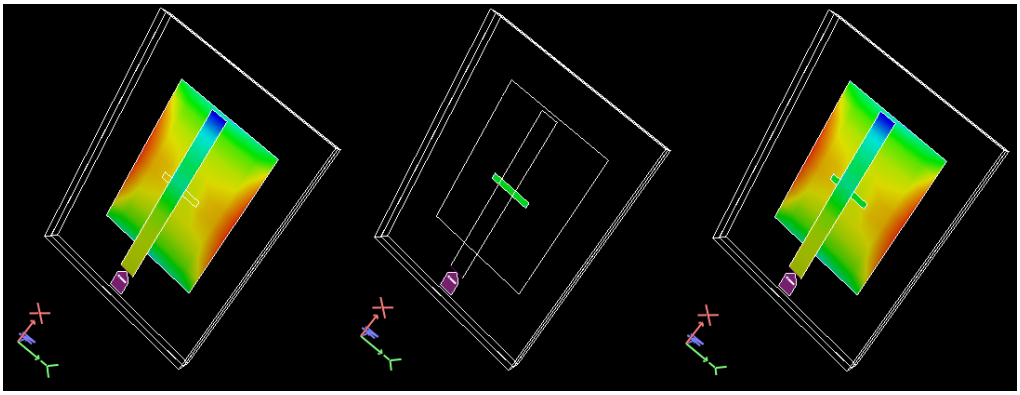


Figure 8.13 The Structure View Parameters dialog.



(a) Electric Current Only (b) Magnetic Current Only (c) Both.

Figure 8.14 The electric and magnetic current distribution displays.

Section 8.4 Construction and Simulation of 3D Interconnects Through Ground Planes

Our last example for this chapter is a via hole transition between two stacked striplines. The configuration is shown in Figure 8.16. In this example, we will show you how to build the via hole interconnect and how we build the hole in the ground plane.

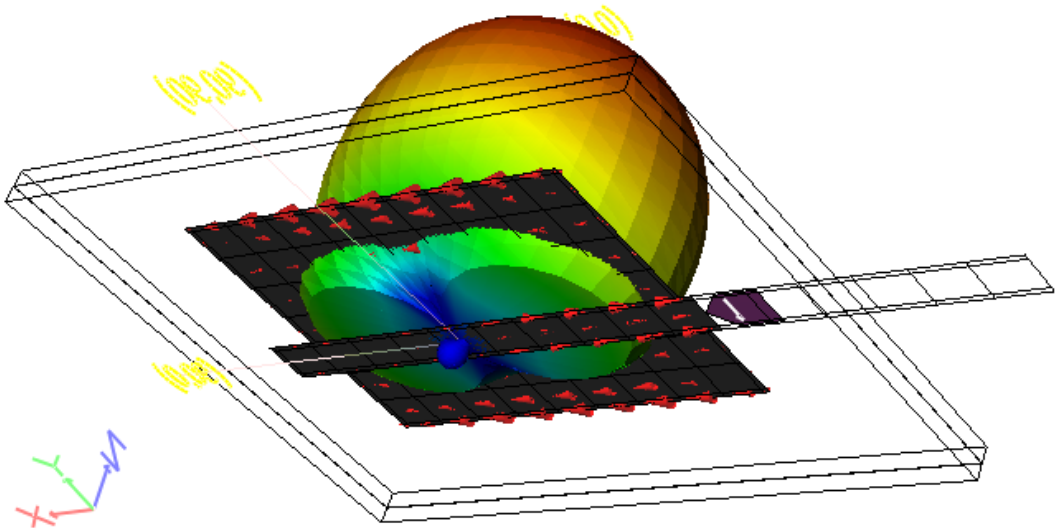


Figure 8.15 Radiation pattern and vector current are displayed simultaneously.

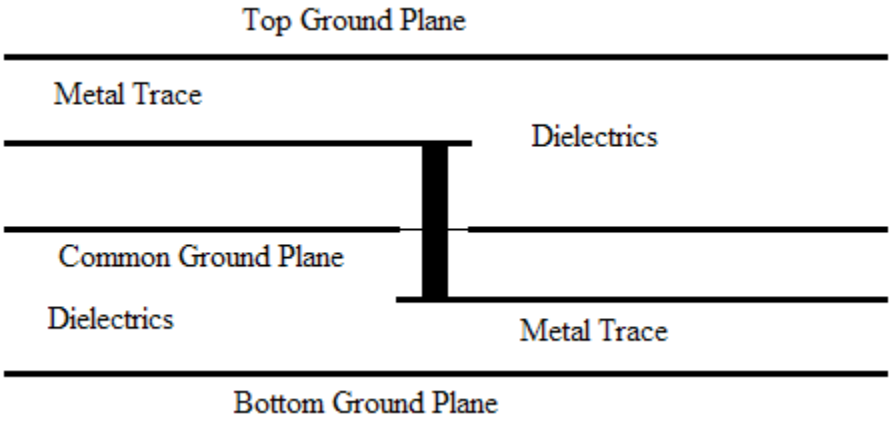


Figure 8.16 A strip line to strip line transition through a via hole.

Step 1 Run MGRID. Open .\ie3d\samples\intercon_sub.geo. Please select Param->Basic Parameters and check the substrates. They are:

No.4 Top Surface Z = 1e+15 mils, ε_r = 1, μ_r = 1, σ = (4.9e7,0) s/m
No.3 Top Surface Z = 60 mils, ε_r = 9.9, μ_r = 1, σ = (0,0) s/m
No.2 Top Surface Z = 30 mils, ε_r = 1, μ_r = 1, σ = (4.9e7,0) s/m
No.1 Top Surface Z = 30 mils, ε_r = 9.9, μ_r = 1, σ = (0,0) s/m
No.0 Top Surface Z = 0 mil , ε_r = 1, μ_r = 1, σ = (4.9e7,0) s/m

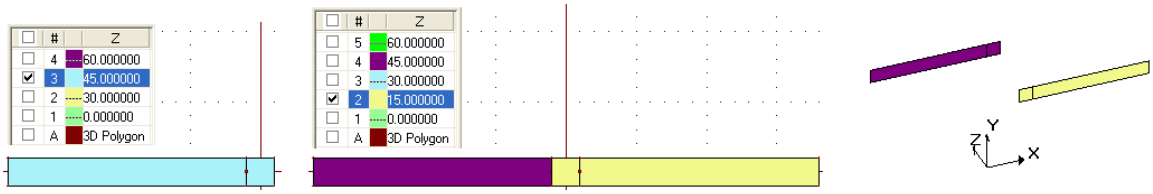
From -∞ to 0 is the bottom ground plane. From 0 to 30 mils is the dielectric for the first stripline. Located at 30 mils is the common ground plane of the stacked striplines. The second stripline is from 30 mils to 60 mils. From 60 mils to +∞ is the top ground plane.

Step 4 Close the Basic Parameters dialog. Select **Insert a Layer** on the Layer Window. Enter z = 45 mils. We are going to define the traces on Z = 45 mils. Type **Shift+A**. Enter x = 2.5 mils and y =

-2.5 mils. Select **Entity->Rectangle** command. Select the **Reference Point** as **Lower Right Corner**. Enter **Length = 5 mils** and **Width = 5 mils**. Select **OK** to accept all the other default values. A square of 5 by 5 is created at the origin. We will use it as the via pad.

In fact, we can enter a point at (x, y)=(0, 0) and use the default setting in Rectangle for it. We use the Lower Right Corner just because we want to demonstrate you the capability.

Step 5 Select **Input->Set To Closest Vertex** command. Click at Z = 45 mils to focus on the layer. Click at the upper left vertex of the square we just entered (x = -2.5 mils and y = 2.5 mils). Confirm the connection. Select **Entity->Rectangle** command. Select **Upper Right Corner** and enter **Length = 42.5 mils** and **Width = 5 mils**. Select **OK** to continue. The feed line for the upper level will be created. We will get Figure 8.17a.



(a) 2 rectangles on Z = 45 (b) The 4 rectangles on Z = 45 and 15 (c) 3D view of the 4 rectangles.

Figure 8.17 The two rectangles created.

We can build the via pad and the feed line in 1 shot. However, we do not want to do that. We will explain the reason later. We are going to build the via pad and the feed line on the lower level based upon the polygons for the upper level.

Step 6 Press down "Shift" and window the 2 polygons we just entered to select them. Select **Edit->Copy and Paste** command. Click it somewhere. Change the **X-offset = 0**, **Y-offset = 0** and **Z-offset = - 30 mils**. Select the option for **Only Pasted Objects Selected** in the **After Copy** group. Select **OK** to continue.

The polygons are duplicated at z = 15 mil level. You should understand that the copied polygons at Z = 15 mils are now selected, and the original polygons at Z = 45 mils are no longer being selected.

Step 7 Click at the Z = 15 mil on the Layer Window. Un-check all other layers, and make sure only the Z = 15 mil layer is checked for selection focus. Window the square only to de-select the via pad on the Z = 15 mil level. Only the feed line at the Z= 15 mils is still selected. We would like this feed line to be on the right hand side of the via pad.

Step 8 Select **Edit->Copy and Reflect** in command. Enter 0 degree for the Reflection Direction. MGRID is in the Copy and Reflect mode. If you move your mouse to the right hand side of the selected polygon, you will see the flying copied polygon.

Step 9 Click the left mouse button somewhere. Enter **5 mils** for the **Reflection Distance**. Select **Original Objects Remain Selected**. Select OK. The feed line is duplicated at the right hand side. The original at the left hand side is still being selected.

Step 10 Select **Edit->Delete** command to delete the polygon at the left hand side. Polygons on both layers are built. We still need to build the via. We will get the main window and 3D view in Figure 8.17b and Figure 8.17c.

Step 11 Select **Conical Via** in **Entity** menu. Enter the following parameters:

Number of Segments for Circle = 6
Starting Point = 0
Center X-coordinate = 0
Center Y-coordinate = 0
Start Z-coordinate = 15
Start Radius = 2
End Z-coordinate = 45
End Radius = 2
Closed Cap

Select **OK** to continue. A via is built from z = 15 to 45 mils. The via pads at Z = 15 and 45 are meshed to fit the via and the feed line at z = 15 and 45 mils are almost untouched. If we build a via pad and a feed line as one single polygon. The feed line will be meshed too, which may increase the number of cells in the simulation. Save the file as: .\ie3d\practice\intercon1.geo. The 3D view is shown in Figure 8.18, except there are not ports.

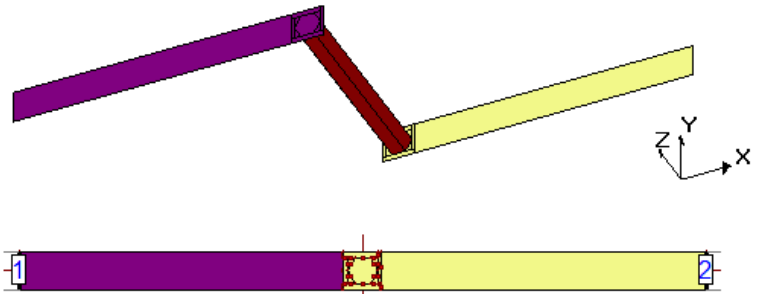


Figure 8.18 The structure by step 15.

Step 12 Select **Port->Port for Edge Group** command. Select the **Advanced Extension** scheme. Window the edge for port 1 (see Figure 8.18). Immediately after the selection, select **Define Reference Plane** in **Port** menu. Click the mouse somewhere. Enter the **Reference Plane Shift** as –45 mils. The reference planes should be shifted to the center of the via. Repeat the procedure to define the port 2 in Figure 8.18. Save the geometry as **.\ie3d\practice\intercon2.geo**. Simulate **intercon2.geo** from 0.1 to 15 GHz with AEC enabled. The results are saved into **.\ie3d\practice\output\intercon2.sp**.

The **Intercon2.geo** is a good approximation to the interconnect, even though we have not built the magnetic current aperture.

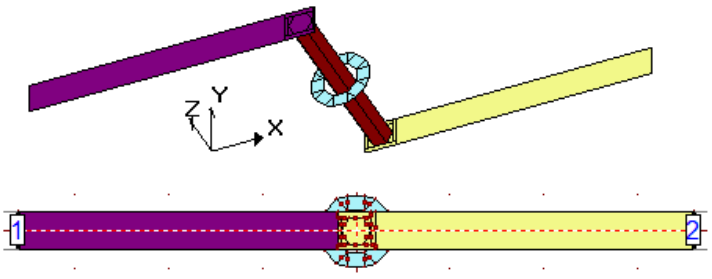


Figure 8.19 The interconnect with the magnetic current modeling of the aperture.

Step 13 Select **Ring** in **Entity** menu. Enter

Number of Segments for Circle = 8

Starting Point = 0
Center X-coordinate = 0
Center Y-coordinate = 0
Center Z-coordinate = 30 mils.
Inner Radius = 3 mils.
Outer Radius = 5 mils.

Select **OK** to continue. The magnetic current modeling for the aperture is built as shown in Figure 8.19.

Step 14 Save the geometry as c**:\ie3d\practice\intercon3.geo**. Simulate the **intercon3.geo** and save the result into **c:\ie3d\practice\output\intercon3.sp**. A comparison between **intercon2.sp** and **intercon3.sp** is shown in Figure 8.20. There is some slight difference between the 2 results, indicating the **intercon2.geo** is not bad an approximation. Also, you can see the dB[S(1,1)] for the intercon3.geo (magnetic ring) case has some uncommon trend. This is due to the numerical stability in handling vertical e-current and the m-current. When they are close together and the frequency is very low, it may cause stability problem. Further improvement will be done on it.

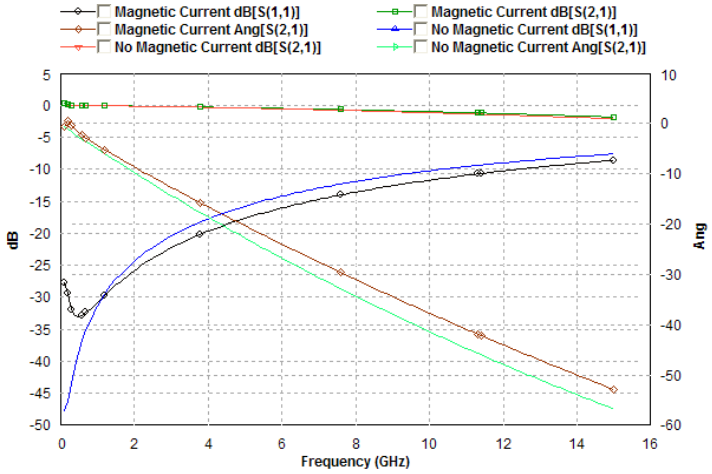


Figure 8.20 Comparison between the 2 models of the 3D interconnect.

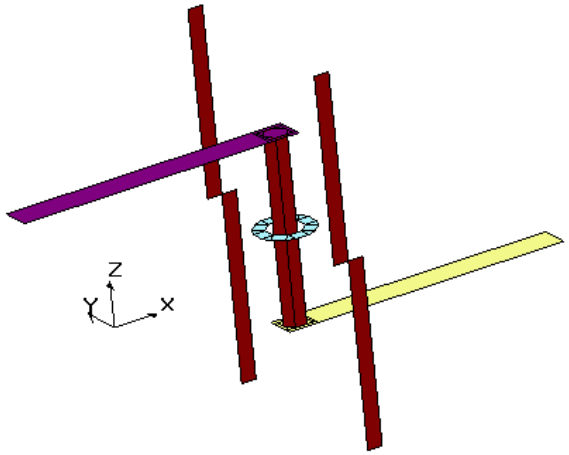


Figure 8.21 The structure with shorting pins.

It is indicated there is much radiation loss starting from 4 to 5 GHz. This is due to the fact that strong parallel plate mode is excited. In order to reduce the parallel plate mode excited, we need to build some shorting pin around the via and the aperture.

Step 15 Open structure c**:\ie3d\samples\intercon4.geo**. It is almost exact the same as the **intercon3.geo**. The difference is there are 4 shorting pins. 2 shorting pins are between the ground planes at z = 0 mil and 30 mils. The other 2 are between the ground planes of z = 30 mils and 60 mils. There is slight misalignment between the pins at the 2 layers in order to avoid the electrical connections between the shorting pins. If we align them to make the shorting pin from 0 to 30 connected to the shorting pin from 30 to 60, MGRID will consider the 2 shorting pins are connected together and they not connected to the common ground at Z = 30.

The simulation result of **intercon4.geo** is saved in **c:\ie3d\practice\intercon4.sp**. Comparison between **intercon3.sp** and **intercon4.sp** is shown in Table 8.2. The shorting pins reduce the power loss from the parallel plate mode significantly. At 15 GHz, the Intercon3.sp indicates 20.75% loss and the **Intercon4.sp** indicates only 9.96% loss.

Table 8.2 The comparison between the interconnects with and without shorting pins.

Frequency	**Intercon3.sp**: without shorting pins				**Intercon4.sp**: with shorting pins			
(GHz)	|S11|	|S21|	$\|S11\|^2+\|S21\|^2$	Loss(%)	|S11|	|S21|	$\|S11\|^2+\|S21\|^2$	Loss(%)
1	0.0288	0.9923	0.9855	1.4514	3.26E-02	0.9984	0.9979	0.2137
2	0.0510	0.9841	0.9711	2.8947	6.46E-02	0.9962	0.9966	0.3418
3	0.0766	0.9753	0.9571	4.2916	9.63E-02	0.9926	0.9945	0.5479
4	0.1035	0.9658	0.9435	5.6518	0.1265	0.988	0.9921	0.7854
5	0.1306	0.9555	0.9300	6.9963	0.1554	0.9822	0.9889	1.1134
6	0.1574	0.9443	0.9165	8.3523	0.1831	0.9754	0.9849	1.5069
7	0.1838	0.9324	0.9032	9.6848	0.2097	0.9675	0.9800	1.9970
8	0.2097	0.9195	0.8895	11.0546	0.2351	0.9586	0.9742	2.5814
9	0.2350	0.9059	0.8759	12.4120	0.2594	0.9486	0.9671	3.2870
10	0.2596	0.8913	0.8618	13.8192	0.2827	0.9376	0.9590	4.0987
11	0.2835	0.8761	0.8479	15.2077	0.305	0.9255	0.9496	5.0425
12	0.3067	0.8601	0.8338	16.6163	0.3264	0.9123	0.9388	6.1172
13	0.3291	0.8435	0.8198	18.0201	0.3471	0.8981	0.9271	7.2938
14	0.3508	0.8264	0.8060	19.4002	0.3668	0.883	0.9142	8.5769
15	0.3717	0.8089	0.7925	20.7520	0.3858	0.8669	0.9004	9.9643

Mixed-electric current and magnetic current modeling of multi-layered structures may not be the best according to accuracy. A better model is to use complete electric current modeling. Using the electric current modeling, some of the numerical stability problem between close coupling of vertical e-current and m-current is avoided. We are also able to model the finite ground plane and imperfect ground. Finite ground plane normally causes resonances which can not be captured using the mixed e-current and m-current model. The only disadvantage of finite ground plane model is that it will increase the size of the problem significantly. Such a model is discussed in chapter 5 and next chapter.

Section 8.5 Magnetic Current Modeling of CPW Structures

We have discussed magnetic current modeling of CPW structure in the section on the slot antenna. We will discuss more in this section.

Step 1 Open file **.\ie3d\samples\mcpws.geo**.

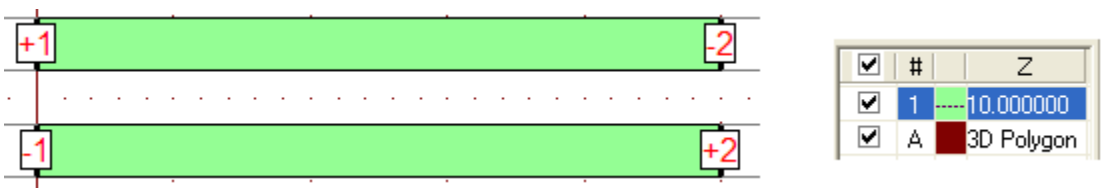


Figure 8.22 Two coupled magnetic current polygons or two slots.

The structure is shown in Figure 8.22. It looks like a coupled microstrip line. You should realize that there is a line on the colored strip z = 10 mils in the layer window. It means the layer z = 10 mils is a ground plane layer. Basically, the two coupled polygons in Figure 8.2 are on the ground plane. As we have discussed before, if an edge of a 3D polygon is touching the ground plane, it is considered as a grounded via. For a 2D polygon, it is different. Any 2D polygon on a ground plane is considered as a slot or a magnetic current polygon.

Step 2 Select **Basic Parameters** in **Parameters** menu.

MGRID shows the dielectric layers in Table 8.3. From z = -∞ to 0 is air. From z = 0 to 10 mils is dielectrics. There is an infinitely extended ground at z = 10 mils. Above z = 10 mils is air.

When electric current is involved, we always assume the z = -∞ to 0 is the whole ground plane when we define a high conductivity for the No.0 layer. When we define a layer with larger top surface z and a high conductivity for the last layer, we also assumed the upper space from the top surface z-coordinate of the second last layer is the whole ground plane. It is no longer true when we define magnetic current on the layer. Table 8.3 shows the dielectric layers for **.\ie3d\samples\mcpws1**.geo. Its dielectric configuration in the simulation is exactly the same as the one in **.\ie3d\samples\mcpws.geo**. Although we define the top surface z-coordinate of the No.2 layer to be a big number, we still consider air is filled from z = 10 mils to infinity and there is a ground plane at 10 mils. Another structure providing the same simulation result is the **.\ie3d\samples\mcpws2.geo**. It is also the same geometry except it is inverted. The dielectric layers of **.\ie3d\samples\mcpws2.geo** are shown in Table 8.4.

You should also be notified that the positive ports 1 and 2 are not on the same polygons. We must be very careful when we define port on magnetic current polygons. The voltage across the magnetic current polygon or a slot is always defined in the following way: The direction from the positive potential to negative potential is always from right to left when you are looking from the end of the port toward the circuit as shown in Figure 8.23. Negative port will make the direction of the voltage opposite. As you can see from Figure 8.23, when the direction for "+1" to "-1" is opposite to the direction for "+2" to "-2". The voltage directions are the same for Port 1 and Port 2.

Table 8.3 The dielectric layers in **.\ie3d\samples\mcpws.geo**

No of Layer	Layer Parameters
2	z=10 mils, EPSr = (1,0), Mur = (1,0), Sigma=(4.9e+7,0)
1	z=10 mils, EPSr = (24.5,0), Mur = (1,0), Sigma=(0,0)
0	z= 0 mil, EPSr = (1,0), Mur = (1,0), Sigma=(0,0)

Table 8.4 The dielectric layers in **.\ie3d\samples\mcpws1.geo**. It yields the same result as **.\ie3d\samples\mcpws.geo**.

No of Layer	Layer Parameters
2	z=1.0e+10 mils, EPSr = (1,0), Mur = (1,0), Sigma=(4.9e+7,0)
1	z=10 mils, EPSr = (24.5,0), Mur = (1,0), Sigma=(0,0)
0	z= 0 mil, EPSr = (1,0), Mur = (1,0), Sigma=(0,0)

Table 8.5 The dielectric layers in **.\ie3d\samples\mcpws2.geo**. It also yields the same result as **.\ie3d\samples\mcpws.geo** when the polygons are defined at z = 0 mils.

No of Layer	Layer Parameters
1	Z=10 mils, EPSr = (24.5,0), Mur = (1,0), Sigma=(0,0)
0	Z= 0 mil, EPSr = (1,0), Mur = (1,0), Sigma=(4.9e+7,0)

Step 3 Select **Meshing** in **Process** menu. Check the **AEC Ratio = 0.1.** Select **OK** to continue.

The meshed structure is shown in Figure 8.24.

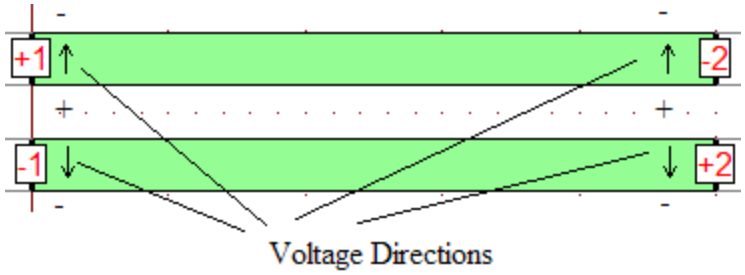


Figure 8.23 The voltage directions for an m-current model of CPW.

Step 4 Copy **mcpws.geo**, **mcpws1.geo** and **mcpws2.geo** from **c:\ie3d\samples** to **c:\ie3d\practice** and simulate them with the **Automatic Edge Cells** option enabled and the **Edge Cell Width** = 1 mils. We can find that the results of the three files are almost identical. Because of the difference in the dielectric layers, it affects the estimated waveguide wavelength for meshing. The 3 structures are not meshed the same and the simulation times are different for the 3 structures. The predicted Zc is about 43-Ohms at 10 GHz.

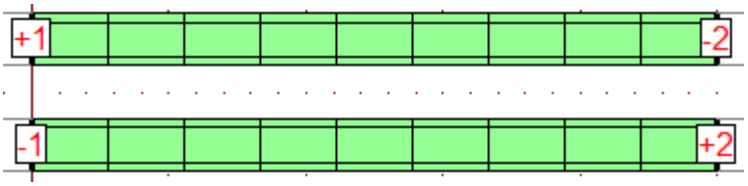


Figure 8.24

Section 8.6 Electric Current Modeling of CPW Structures with Finite Ground and Finite Thickness

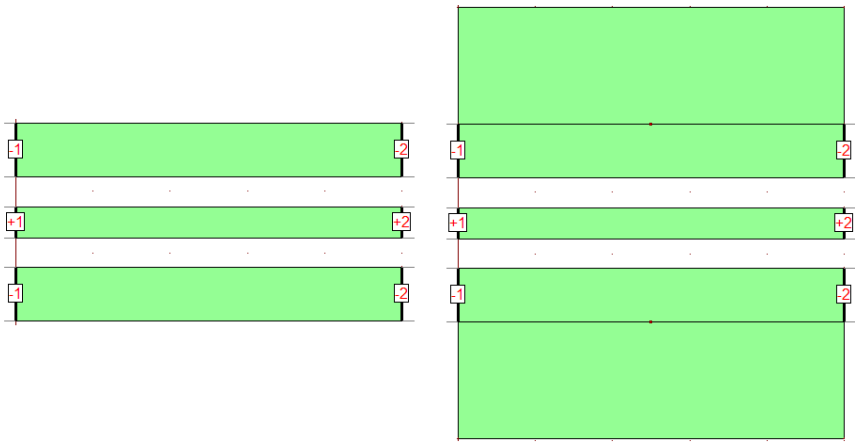
In practical applications, there are no infinite ground plane CPW structures. When the CPW ground plane is not very big. We may need to model finite ground plane CPW structures. For finite ground plane CPW structures, we need to use electric current modeling. Also, magnetic current modeling assumes no loss from the ground plane. In case we need to model the metallic loss in CPW structures, we have to use electrical current modeling. Another situation of CPW requires electric current modeling is the finite thickness of trace. It is more difficult to define finite thick traces or ground plane using magnetic current modeling we have to use electric current modeling for it.

The .\ie3d\samples\ecpws.geo is a typical electric current modeling of CPW with finite ground plane. Its predicted Zc is about 44.8-ohms. It is about 4% difference from the infinite ground plane model.

The .\ie3d\samples\ecpws1.geo is an electrical current modeling of CPW with finite wide ground plane and finite thick trace. It should yield high accuracy results for the thickness and loss effect.

When the width of the finite ground becomes bigger, it is suggested that the user should define a negative port on part of the edge only (see .\ie3d\examples\ecpws2.geo and Figure 8.25). Defining the negative port on the full edge may yield inaccurate results especially when there are discontinuities in the structure and when there are non-symmetrical structures connected to the CPW. The results in ecpws2.geo with wider ground plane are almost identical to the narrower ground plane structure in ecpws.geo.

When we use electric current to model the finite ground plane. Normally, you need to make sure the ground plane will be meshed into at last 2 segments in the transverse direction. It may create accuracy problem if a finite ground is meshed too coarsely in the transverse direction.



(a) The negative ports defined on the full range of the edges (b) The negative ports on parts of the edges

Figure 8.24 The ports on the edges of CPW circuits.

Section 8.7 Advantage and Disadvantage of Slot Structures and Magnetic Current Modeling

As it is mentioned earlier, slot structures normally yield wider bandwidth for both antenna and circuit structures. Why can they offer wider bandwidth? The reason is that the current is normally not highly concentrated at the proximity of the metal surrounding the slot and the field is not highly concentrated at the proximity of the slot. When energy is not highly confined in a small space, we will expect the structure has wider bandwidth.

When we have a slot in a large ground plane, we can model the field as magnetic current on the slot instead of the current on the ground plane. We can reduce the simulation domain significantly when the ground plane size is big. The results are normally quite accurate when the ground plane is big. However, there are situations we may want to avoid using magnetic current modeling.

As it is mentioned earlier, the current and field are not highly concentrated at the proximity of the slots. Therefore, the size of the ground plane can be very critical when it is not very big. Change in it may cause significant difference in the performance of the structure. Some antenna designers may be familiar with the book published by Prof. Kin-Lu Wong. It collects many good examples of slot antennas in his book. However, if you try to use IE3D to model them using magnetic current, you normally will not match the results. The reason is that the ground plane sizes are not very big in all those examples. They are very critical to the antenna performances. Users should try to use electrical current modeling for the slot antennas without a very big ground plane. Normally, microstrip antennas can be modeled as infinite ground plane structures even the ground planes are not very big. This is ok because the field and current for a microstrip antenna is confined underneath the patch.

Many modern circuits (PCBs and IC packaging) are multi-layer structures. An example is the PCB structure discussed in Chapter 5. They look like strip line structures connected with vias. We can try to use the mixed electric current and magnetic current model discussed in this chapter for it. However, users are strongly suggested to use the electric current for the finite ground plane structures whenever possible. Electrical current model will precisely capture the resonances of the finite ground and they are very important to the performance of a circuit. We will end this chapter here

Chapter 9 Lumped Element Extraction and Signal Integrity

This chapter will discuss how to use the IE3D to extract lumped element equivalent circuits from s-parameters. We will also discuss how we can model metallic strip thickness precisely and how we can build complicated packaging models. This chapter also includes discussion on accurate time transient analysis using the MDSPICE, an s-parameter based SPICE simulator.

In high speed digital circuits, interconnects or printed strips can no longer be considered as perfectly connecting elements because they are not extremely small compared to a wavelength. They may not be very long compared to a wavelength, and we may still model them as a set of cascading shunt capacitors and resistors, series inductors and resistors, and mutual inductors. From the viewpoint of accuracy, modeling printed strips or 3D interconnects using s-parameter frequency response is a better choice than modeling them as a set of lumped elements. The advantage of lumped element equivalence is the SPICE capability.

Traditional SPICE simulators are popular time-domain circuit simulators accepting linear and non-linear lumped elements. SPICE simulators are much faster than EM simulators normally. They can also provide flexible time-domain analysis. However, SPICE simulators do not have a precise modeling capability on distributed elements because they are not field solvers. Also, traditional SPICE simulators cannot accept the primary output of a full-wave EM solver. In order to make the IE3D simulator compatible with traditional SPICE simulators, we have implemented schemes to extract the RLC equivalent circuit of a structure. The extracted RLC equivalent circuit is saved in SPICE compatible format so that it can be directly imported into a traditional SPICE simulator.

As a related topic, MDSPICE is a mixed-frequency domain and time-domain SPICE simulator. In short, the advantage of the MDSPICE compared to traditional SPICE is that the MDSPICE can accept s-parameters modules accurately and efficiently. This advantage means much to the high speed and high frequency design societies: (1) The MDSPICE does not need to convert the s-parameters into RLC-equivalent circuits. We can obtain time-domain results directly from the IE3D solved s-parameters. (2) We can avoid accuracy and stability problems in converting the s-parameters to RLC-equivalent circuits. Conversion from s-parameters to RLC-equivalent circuits is a sophisticated process. It is normally available when a structure is small compared to wavelength. Even when the structure is small compared to wavelength, only the equivalent circuits of coupled interconnects are easy to extract. For general N-port s-parameters, the extraction of frequency independent RLC circuit from general s-parameters is much more sophisticated. Without an equivalent circuit accurate over the wide frequency range of interest, traditional SPICE simulators will not be able to provide accurate time domain results. (3) The MDSPICE can improve the efficiency significantly. Assume we have a complicated 2-port structure: 1-port is the input and the other is the output. We may have thousands or more RLC elements inside the circuit when the circuit it describes is electrically long. A traditional SPICE simulation will be time consuming because it needs to go through all the elements each step in the time-domain simulation. Using the MDSPICE, we can get the 2-port s-parameter frequency response and it contains limited data. Then, we can perform a simple time domain simulation on the 2-port s-parameter frequency response on MDSPICE. It will save much simulation time. The simulation is normally very fast even with electrically long structure.

It should worth mentioning that the MDSPICE 3 has also implemented a wide band frequency independent RLC extraction scheme from a general N-port s-parameter file. The resulting RLC circuit is SPICE compatible. Normally, the s-parameters based upon the wide band extracted RLC circuit can match the original s-parameters almost perfectly.

Section 9.1 Transmission Line Theory and RLC-Equivalence.

From the transmission line theory, we know the voltage v(z) and current i(z) along a transmission line are determined by the telegrapher equations in the frequency domain.

$$\partial v(z) / \partial z = - (R + j \omega L) \times i(z) \quad (9\text{-}1)$$
$$\partial i(z) / \partial z = - (G + j \omega C) \times v(z) \quad (9\text{-}2)$$

where R is the distributed series resistance per unit length, L is the distributed series inductance per unit length, G is the distributed shunt conductance per unit length, C is the distributed shunt capacitance per unit length.

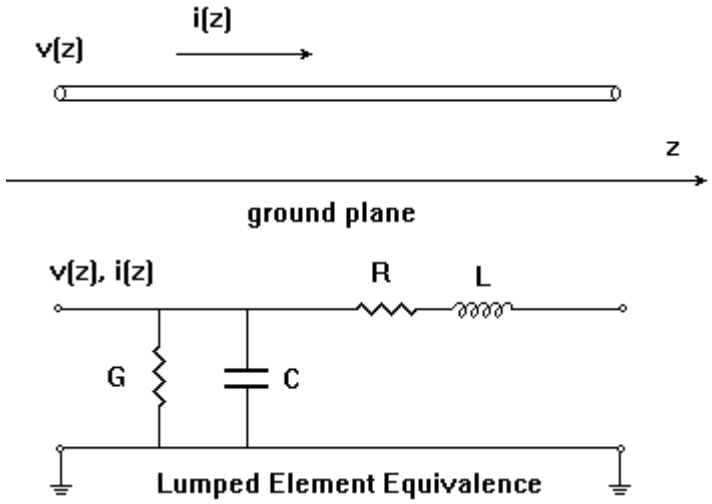


Figure 9.1 The lumped-element equivalence of a short transmission line.

Integrating both sides of equations (9-1) and (9-2) from z1 to z2 yields,

$$v(z2) - v(z1) = - (R + j \omega L) \times \int i(z)\, dz \quad (9\text{-}3)$$
$$i(z2) - i(z1) = - (G + j \omega C) \times \int v(z)\, dz \quad (9\text{-}4)$$

Assume the distance between z1 and z2 is small compared to a wavelength. There will be little variation in the current and voltage distribution along the transmission line. We can approximate equations (3) and (4) as:

$$v(z2) - v(z1) = - [(R + j \omega L) (z2 - z1)] \times i(z1) \quad (9\text{-}5)$$
$$i(z2) - i(z1) = - [(G + j \omega C) (z2 - z1)] \times v(z1) \quad (9\text{-}6)$$

Equations (9-5) and (9-6) suggest a lumped element equivalent circuit as shown in Figure 9.1. We call it an RLC-segment. Certainly, this equivalent circuit is conditional. It assumes the length (z2 – z1) is much smaller than a wavelength. If it is not too small compared to a wavelength, we will need to use multiple segments to approximate the circuit. In the other words, we can still approximate an electrically long transmission line using multiple RLC-segments accurately as long as each segment is electrically small. Equations (9-1) and (9-2), or equations (9-5) and (9-6) can be extended to coupled transmission line structures.

For a set of coupled interconnects as shown in Figure 9.2, we still can use the same kind of equivalent circuit to approximate the distributed structure. At low frequency, we can still use the equivalent circuit similar to the one in Figure 9.1 to approximate it. At high frequency, we can try to use multiple RLC-segments to approximate it.

The physical meaning of equations (9-1) and (9-2) in matrix form can easily be illustrated for a 3-conductor system as shown in Figure 9.3. As it is shown, L1_1, L2_2 and L3_3 are the self-inductances; K1_2 is the mutual inductance coefficient between L1_1 and L2_2. K2_3 is the mutual inductance coefficient between L2_2 and L3_3. K1_3 is the mutual inductance coefficient between L1_1 and L3_3. C1_1, C2_2 and C3_3 are the capacitances with respect to the ground. C1_2 is the capacitance between the

line 1 and line 2. C1_3 is the capacitance between line 1 and line 3. C2_3 is the capacitance between line 2 and line 3. Similarly, we can define RL1_1, RL2_2, RL3_3, RC1_1, RC2_2 and RC3_3. The users should be notified that there are two definitions for the self-capacitance for each line. The C1_1 here is only defined as the capacitance with respect to the ground. Another definition is defined as C1_1' = C1_1 + C1_2 + C1_3 + … + C1_N, where N is the number of transmission lines in the system.

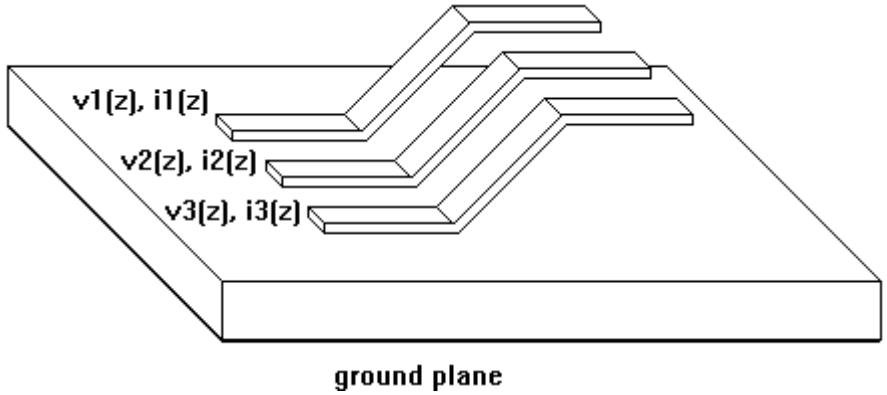


Figure 9.2 A multi-interconnect structure.

It shall be emphasized that the equivalent circuits in Figures 9.1 and 9.3 are valid only when the transmission lines or interconnects are much shorter than the wavelength of the application frequency. When the transmission lines are comparable to a wavelength, one has to break the transmission lines into multiple smaller sections. Each section will be approximated by an equivalent circuit in Figure 9.1 and Figure 9.3. Therefore, the net result of the equivalence for long transmission lines is the cascading of the equivalent circuits of those shown in Figure 9.1 and Figure 9.3.

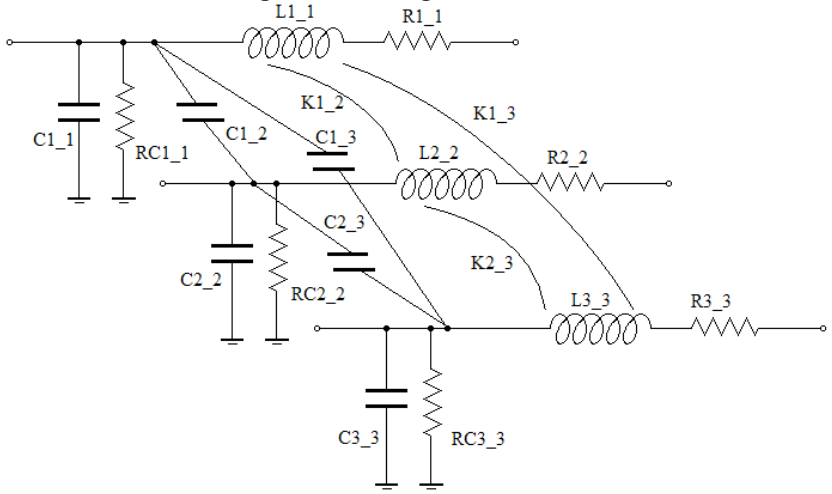


Figure 9.3 The equivalent circuit of a three-conductor system.

For printed strips, when the strip width is much larger than the substrate thickness, the capacitance effect is the dominant effect, and the series resistors and inductors can be neglected. Therefore, C-equivalent circuit can be used. When the strip width is much smaller than the substrate thickness, the inductance effect is the dominant effect, and the shunt conductance and capacitance can be neglected. Therefore the L-equivalent circuit is used. For most of the transmission line structures used in digital circuits, the inductance effect is the dominant effect.

On the IE3D, the RLC-equivalent circuit extraction from s-parameters is well implemented. The single frequency extracted RLC-equivalent circuit can be exported in the standard SPICE format. For

multiple frequency extraction, SPICE simulators do not accept frequency dependent RLC. The extracted RLC-equivalent circuit will be written in ASCII text readable by the users.

It should be noted that the IE3D extracted RLC-circuits are only good for low frequency when the structure size is smaller than 10-20% of the wavelength. For electrically large structures, the RLC-extraction requires much more sophisticated process. Such sophisticated process is available in the MDSPICE simulator (not part of IE3D). The MDSPICE allows a user to perform a time domain transient analysis on the s-parameters created by IE3D or other simulators. The MDSPICE also implements a sophisticated optimization scheme that can extract frequency independent RLC-equivalent circuit for wide band applications.

Section 9.2 Simulation of a Stripline Structure

Stripline has much less dispersion than microstrip structures. It may be more suitable for wide band applications. Our first example for this chapter is an offset stripline structure with the cross-section illustrated in Figure 9.4. The structure has 3 dielectric layers. The center one is the glue layer. The metallic strip of 1.4 mil thickness is embedded into the glue layer.

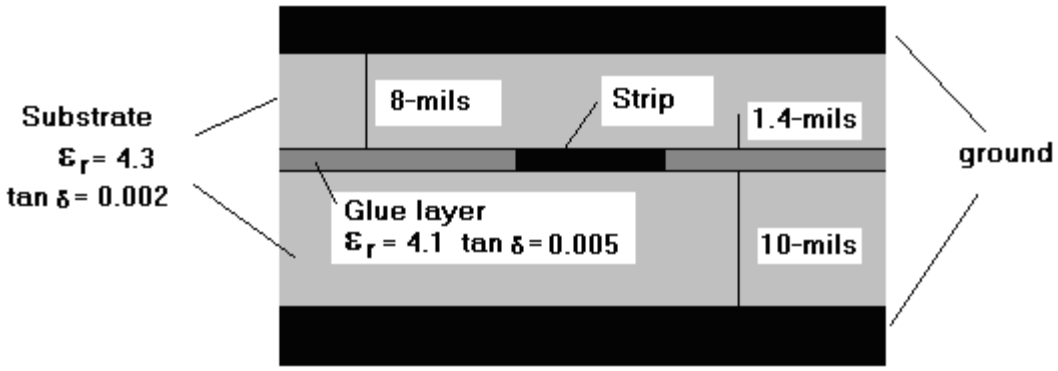


Figure 9.4 An offset stripline structure.

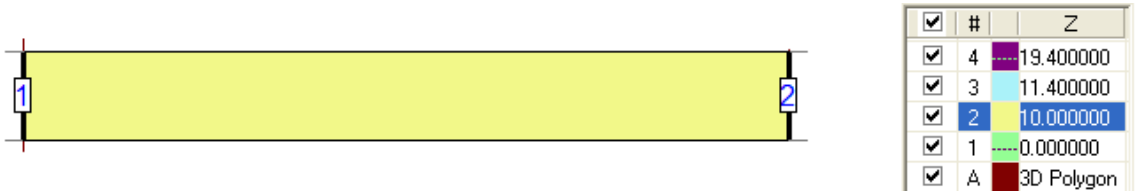
A uniform stripline structure of the cross-section is built in .\ie3d\samples\strip1.geo. We will start from the example.

Step 1 Run MGRID. Open **.\ie3d\samples\strip1.geo**. Save it into **.\ie3d\practice\strip1.geo**. Select **Basic Parameters** in **Parameters** menu to check the substrate setup and the metallic strip setup.

Both the substrate setup and metallic strip setup should be like what is described in Figure 9.4. The metallic strip thickness is 1.4 mils. On the main window, the strip should be in the color for Z = 10 mils, which is the bottom surface of the strip. Any polygon on the Z = 11.4 mils should be displayed in another color. However, we cannot see any polygons on the Z = 11.4 mils layer (see Figure 9.5). Basically, like all other MOM simulators, the strip thickness defined in the Basic Parameters is used to correct the loss effect of the strip only. It is the default mode on IE3D. The strip is still considered as infinitely thin in the default mode. The loss effect on the strip will be corrected based upon the thickness information defined in the Basic Parameters.
One very important feature of the IE3D is that it will allow the user to build the thickness as it is. It will include all the thickness effect of the structure when we use build the thickness. We will discuss it in the next section.

Step 2 Select **Process->Simulate** command to simulate the strip1.geo from 0.1 to 20 GHz for 200 frequency points. Please enable AEC Ratio = 0.07. Please also enable AIF for fast results. The

simulation will be done in couple seconds. MGRID will go into s-parameters processing dialog automatically.



(a) The polygon for the strip at Z = 10 mils (b) The layer window.

Figure 9.5 The single strip line without thickness modeled.

Section 9.3 Finding Transmission Line Parameters

A utility routine is implemented into MGRID and MODUA to extract the transmission line (TLN) parameters based upon the 2-port s-parameters of a uniform transmission line. The extracted parameters include the complex characteristic impedance, the waveguide wavelength, the attenuation and the complex effective dielectric constant. The requirement for the 2-port s-parameters is that its electrical length should be shorter than half of a waveguide wavelength. If it is longer than it, the calculated waveguide wavelength and the effective dielectric constant will be wrong due to the non-unique value in the Ang[S(2,1)] in the s-parameters.

Step 1 On S-Parameters processing dialog, select "Find TLN Parameters from 2-Port S-Parameters" in Processing S-Parameters section. Select "Go!". MGRID prompts you the Find Transmission Line parameters (see Figure 9.6). The TLN length value is automatically found by MGRID. In case it is not correct, you can correct the value. You need to enter the correct TLN length. Otherwise, the parameters related to the wavelength will not be correct.

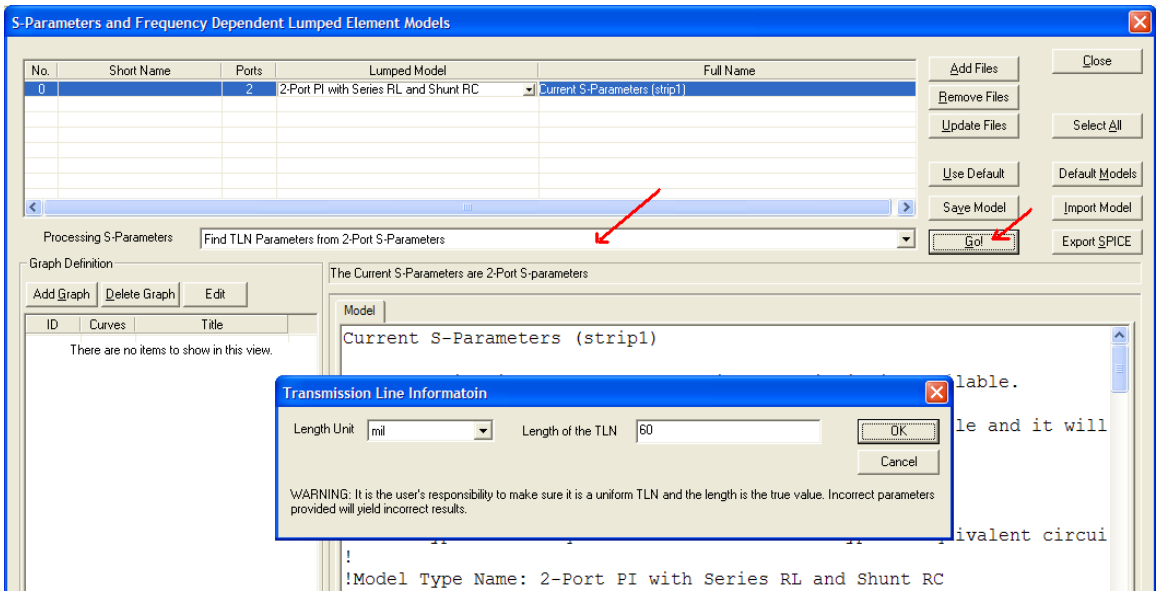


Figure 9.6 The Find Transmission Line Parameters utility on MGRID.

Step 2 Select OK. MGRID will find the TLN parameters and list them in a dialog. Select OK. MGRID will save the TLN data into a file and open it on Notepad. Part of the results is shown in Table 9.1.

Table 9.1 The TLN Parameters of the stripline without structure thickness effect.

Freq (GHz)	Re(Zc) ohms	Im(Zc) ohms	Lambda (mil)	Alpha (dB/mil)	Re(Ereff)	Im(Ereff)
0.1	59.9782	-1.159	55544.9	2.1487e-5	4.51313	-0.1974880
1	59.0287	-0.31426	5643.59	7.67631e-5	4.37357	-0.0694305
2	58.9267	-0.228772	2826.13	0.000124453	4.36024	-0.0562034
5	58.7587	-0.106889	1132.81	0.000218759	4.34223	-0.0394348
10	58.5316	-0.0385204	567.076	0.000350948	4.33198	-0.0315945
20	57.0863	0.031849	283.899	0.000554221	4.32099	-0.0249154

As you can see, the effective dielectric constant, Ereff, is about 4.3 at most of the frequency points. This is certainly correct because the top and bottom substrates of the strip line are of dielectric constant of 4.3.

You may notice that the Ereff is 4.513 at 0.1 GHz. It is obviously larger than 4.3. Some users may consider it not accurate. The reason for their concern is the following. According to waveguide theory, the Ereff can be estimated as:

$$\text{Ereff} = \sum \text{Ci Eri} \quad (9\text{-}7)$$

$$\sum \text{Ci} = 1 \quad (9\text{-}8)$$

$$0 \le \text{Ci} \le 1 \quad (9\text{-}9)$$

Where Eri is the dielectric constant of the i-th material in the TLN; Ci is the coefficient for the i-th material. It is related to how much field is in the i-th material. The more field is in the i-th material, the Ci will be larger. From (9-7) to (9-9), we can know that the Ereff ≤ maximum of Eri.

Some user may think that there are only 3 dielectrics in the stripline. Therefore, the Ereff should not exceed the maximum of Eri, and the maximum Eri is 4.3. Does it mean that the IE3D does not agree with the waveguide theory?

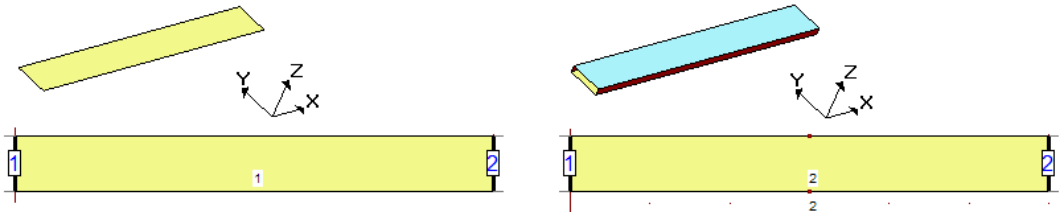
In fact, both the waveguide theory and the IE3D are correct. Please do not forget that both the ground plane and the stripline are lossy. Each ground plane and the metallic strip should be considered as a type of dielectrics in the equations (9-7) to (9-9). The coefficient should also be a complex value. The Eri for a metallic strip is a very large value. At low frequency, there is more field penetrating into the metal. Therefore, it has more effect to the Ereff at low frequency.

Section 9.4 Precise Modeling of Finite Trace Thickness

The Zc value calculated in strip1.geo and documented in Table 9.1 is in fact larger than the reality. The reason is that we have not included the effect of the trace thickness. As we discussed in Section 2, we are still modeling it as infinitely thin strip in the strip1.geo file. This is the way most MOM simulators handling the strip thickness. The IE3D is in fact one step ahead of some other MoM simulators in treating finite thick trace thickness. We can actually build the strip with thickness.

Step 1 Select Window->3D Geometry Display on MGRID. The only polygon is shown. It is a polygon without true thickness. Select Show Metal Type in View menu. MGRID will display the metal type 1 on the only polygon (see Figure 9.7a).

Step 2 Click at the Z = 10 mils on the Layer Window to focus the input at Z = 10 mils layer. Select **Edit->Layers->Grow Metal Thickness** command. MGRID will prompt you the Grow Metallic Thickness on Layer 2 dialog. Select the Thickness Grow Direction as Positive Z-Direction. Make sure the option Ports on All Edges is checked. Select OK to continue.



(a) The zero-thickness model (b) The finite-thickness model.

Figure 9.7 The comparison of zero-thickness model (strip1.geo) and finite-thickness model (stript1.geo).

. The polygon in the 3D view window is added with thickness (see Figure 9.7b). Basically, 4 polygons are created to replace the original polygon on Z= 10 mils. Each polygon is labeled with No.2 metallic strip type.

Please check the **Parameters->Basic Parameters** command. The No.2 metallic strip type has almost the same properties as the No.1 type. The only difference is that it has a factor F = 0.416667. It is a re-correction factor allowing the strip's loss effect to be modeled accurately with the thickness built. More discussion on modeling of finite thickness can be found in Appendix S.

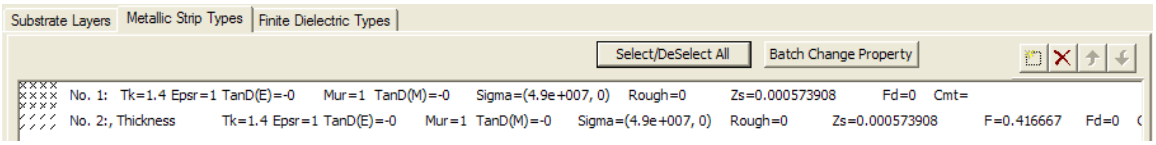


Figure 9.8 The Metallic Strip Types section of the Basic Parameters dialog.

Step 3 Save the structure as: .\ie3d\practice\strip1t.geo. It takes less than a minute to simulate it. The s-parameters are saved into: .\ie3d\practice\output\strip1t.sp. Use MODUA to display the s-parameters.

Table 9.2 The TLN Parameters of the stripline with structure thickness effect.

Freq (GHz)	Re(Zc) ohms	Im(Zc) ohms	Lambda (mil)	Alpha (dB/mil)	Re(Ereff)	Im(Ereff)
0.1	50.4965	-1.12823	55256.7	2.4344e-5	4.55974	-0.224914
1	49.5080	-0.267272	5637.28	7.41378e-5	4.38339	-0.0671398
2	49.4289	-0.200478	2822.52	0.000123073	4.37142	-0.0556515
5	49.3139	-0.124868	1130.64	0.000239619	4.35887	-0.0432779
10	49.1298	-0.065111	566.038	0.000374051	4.34787	-0.0337361
20	48.5686	-0.0137156	283.357	0.000585236	4.33752	-0.0263601

Step 4 Select "Find TLN Parameters from 2-Port S-Parameters" and select "Go!" in the s-parameters processing dialog to find the TLN parameters. The results are listed in Table 9.2.

Please compare the data in Tables 9.1 and 9.2. You will be surprised to see that the difference in Zc is about 12%. The Zc results in Table 9.2 should be very accurate. It means that the calculated Zc will be more than 12% off if we do not consider the structural effects of the

thickness. For our case, the thickness over width ratio (T/W) is 1.4 / 7 = 20%. Users are notified by the following facts: (1) The thickness effect in stripline structures and CPW structures is very critical. The reason will be explained in Chapter 12. Normally, we need to include thickness effect for high accuracy results. Thickness is normally less important in microstrip structures. (2) The thickness is affecting the Zc of stripline the most. It does not affect the Ereff much for stripline structures. For microstrip and CPW structures, it will affect the accuracy of both Zc and Ereff.

Section 9.5 RLC Equivalent Circuit Extraction

The TLN parameters extracted are very accurate. Unfortunately, SPICE circuit simulators may not accept frequency dependent TLN model. In order to use it, we may want to extract the RLC-equivalent circuit model. The RLC-equivalent circuit extraction is implemented into the MODUA. It is not only for a single uniform TLN. It is also for multiple coupled interconnects. We will demonstrate it on the strip1t.sp file.

Step 1 Run MODUA. Select **File->Display Parameter Module** command. Select strip1t.sp file to display it. Select Process->LC-Equivalent command. MODUA will prompt you that the simulation results will be abandoned. Select YES to continue. MODUA will prompt you for the frequency points. Please check all the frequency points. Select OK. MODUA will prompt you that the process is a multiple frequency LC-Equivalence. The results will not be in SPICE compatible format.

Step 2 Select OK and MODUA will prompt you for the Port Definition Style for the Equivalence dialog. Most of the options are grayed out. We will discuss this dialog later. Please select OK to continue. MODUA will perform the extraction. After it finishes, it will prompt you to save the RLCG Matrix.

There are a few options for the file format (see Figure 9.9): (1) Equivalent Circuit Format (Diagonal R); (2) Equivalent Circuit Format (Full R); (3) Coupled TLN Format. For the (1) and (2), they are format describing frequency dependent coupled interconnects. They can be imported into MDSPICE for time domain simulation. However, (1) and (2) store the distributed RLCG matrix per unit length. It requires the users to enter the Length of the structure the s-parameters are from. From the entered Length value, MODUA derives the RLCG values per unit length. In some senses, if we choose (1) and (2), we need to make sure we enter the Length in Figure 9.9 correctly. Otherwise, MODUA will yield incorrect RLCG values per unit length.

For our cases where we are interested in the total RLCG values for the structure, we should use the Coupled TLN Format.

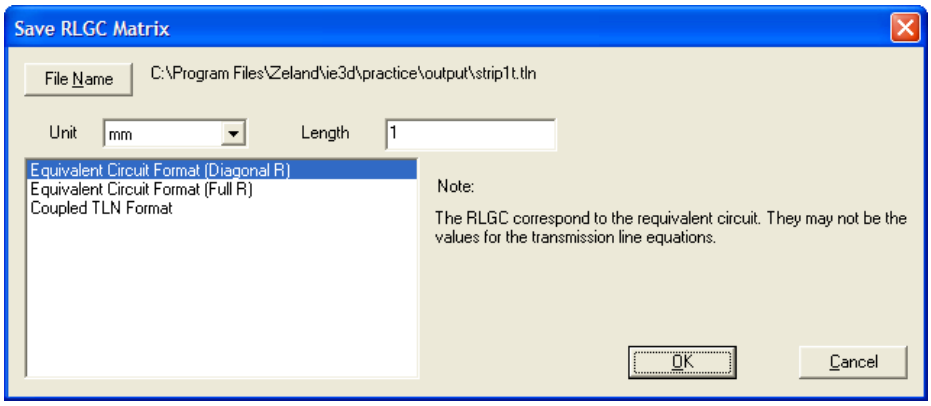


Figure 9.9 The Save RLGC Matrix dialog.

Step 3 Please choose Coupled TLN Format in the dialog. Then, select OK to continue. MODUA will inform you that the extracted RLCG matrix is saved in: strip1t.tln file. Also, it will inform you that the error function at 3.7 GHz and higher frequency exceed the limit of 0.25.

Step 4 We will discuss the error function in the next section. Please select OK to continue. The strip1t.tln file is opened in a nice tabular format on Notepad.

Section 9.6 Error Function in Predicting the Error Involved in RLC Equivalence

It is well known that we can use RLC equivalent circuit to approximate a distributed circuit when the electrical length of the structure is small compared to wavelength. What is small compared to wavelength? Normally, we require it to be less than 10% of a waveguide wavelength.

We work on the s-parameters directly when we are extracting the RLC from it. You may not know the true electrical length of the structure when you are performing the extraction. You are not sure whether the structure is too long and you want the simulator to inform you if the structure is too long.

For the above reason, we have introduced an Error Function to indicate whether a structure can be approximated by a RLC segment.

Let's take a uniform TLN as an example. We know the input impedance is determined as:

$$Zin = Zc [Zt + Zc \tanh(\gamma T)] / [Zc + Zt \tanh(\gamma T)] \quad (9\text{-}10)$$

Where Zt is the impedance terminated at output; Zc is the characteristic impedance of the TLN; T is the physical length of the TLN; $\gamma = \alpha + j\,\beta$, α is the attenuation constant and β is the propagating constant. $\beta = 2\,\pi / \lambda$ and λ is the waveguide wavelength.

For lossless TLN, we have,

$$Zin = Zc [Zt + Zc\, j \tan(\beta T)] / [Zc + Zt\, j \tan(\beta T)] \quad (9\text{-}11)$$

When T is small compared to waveguide wavelength, we have,

$$\tan(\beta T) \approx 2\,\pi\,(T / \lambda) \quad (9\text{-}12)$$

When the port 2 is terminated with open circuit, $Zt = \infty$ and the input impedance is,

$$Zopen = - j\, Zc / \tan(\beta T) \approx - j\, Zc / [2\,\pi\,(T / \lambda)] \quad (9\text{-}13)$$

When the port 2 is terminated with short circuit, Zt = 0 and the input impedance is,

$$Zshort = j\, Zc \tan(\beta T) \approx j\, Zc [2\,\pi\,(T / \lambda)] \quad (9\text{-}14)$$

We define the error function as:

$$E = \sqrt{}\,(| Zshort / Zopen |) \quad (9\text{-}15)$$

We will use the error function E as an indication how accurate a RLC-extraction is. For a uniform TLN, we have,

$$E \approx 2 \pi (T / \lambda) \quad (9\text{-}16)$$

When we use single LC segment to approximate the TLN (see Figure 9.1), we have the following: When port 1 is terminated with short circuit, the impedance at port 2 is,

$$Zshort = j \omega L \quad (9\text{-}17)$$

When port 1 is terminated with open circuit, the input impedance at port 2 is,

$$Zopen = j \omega L - j / (\omega C) \quad (9\text{-}18)$$

Therefore, we have the Zc predicted by the L and C as,

$$Z_{LC} = \sqrt{ (L / C)} = Zc \sqrt{ (1 - | Zshort / Zopen|)} = Zc \sqrt{ [(1 - E^2)} \quad (9\text{-}19)$$

Equation (9-17) is very meaningful. When the TLN length T is very small compared to the wavelength λ, the predicted characteristic impedance using the L and C is basically the same as the true value Zc. However, the Z_{LC} will deviate with increasing (T/λ). It is also interesting to note that Z_{LC} is always smaller than Zc. Similarly, the extracted L at higher frequency is normally smaller than the distributed L value, and the extracted C at higher frequency is normally larger than the distributed C value. Table 9.3 documents the relationship between the error of Z_{LC} compared to Zc vs. the value of (T/λ).

Table 9.3 The relative error in Z_{LC} vs. (T/λ).

(T/λ)	1%	2%	3%	4%	5%	10%
Relative Error in Z_{LC} (%)	0.2	0.8	1.8	3.2	5.1	22.2
Error Function, $E \approx 2 \pi (T / \lambda)$	0.063	0.126	0.188	0.251	0.314	0.628

We expand the concept of error function to RLC extraction for general coupled interconnect problems. We set the Error Function Limit at 0.25. When it exceeds 0.25, we will warn the users.

Section 9.7 Frequency Dependent and Independent RLC Equivalent Circuit Extraction

Let's get back to the discussion of the RLC extraction results. The results are saved in the strip1t.tln file and it is opened on the Notepad. The data is documented in Table 9.4.

Table 9.4 The frequency dependent RLC extracted results.

Freq (GHz)	Error Function	L (nH)	Series R (Ohm)	C (pF)	Shunt R (Ohm)	Q
0.1	0.006821	0.5481	0.01618	0.2150	3.212e+6	21.28
0.5	0.03353	0.5291	0.02865	0.2150	6.585e+5	57.95
1.0	0.06697	0.5254	0.04292	0.2153	3.238e+5	76.52
2.0	0.1344	0.5192	0.06694	0.2163	1.542e+5	95.34
3.0	0.2030	0.5107	0.08483	0.2180	9.689e+4	107.78
3.7	0.2520	0.5032	0.09372	0.2196	7.502e+4	115.11
5.0	0.3464	0.4855	0.1027	0.2235	5.051e+4	126.94
10	0.7858	0.3799	0.06159	0.2545	1.567e+4	150.7

As you can see, the extracted RLC values are changing with frequency. The L is decreasing with frequency and the decreasing is accelerated at higher frequency. The C is slightly increasing with frequency. It is believe the RLC equivalent circuit beyond 5 GHz is not accurate anymore.

The data in the strip1t.tln file can not be used by standard SPICE simulator because standard SPICE simulator accept frequency independent RLC. We have to perform the extraction at a specific frequency

where the L and C values are close to constant and we will use it the extracted RLC at low frequency. We certainly can take down the data in the strip1t.tln file and put it into SPICE compatible format. We do not need to do it manually because we have an automatic way to do it.

Step 1 Please select **Process->LC Equivalent** command on MODUA again. MODUA will prompt you for the frequency points. Please select Delete All button to un-check all the frequency points. Please check at the 0.2 GHz frequency point. Select OK to continue. MODUA will prompt you for the Port Definition Style for the Equivalence dialog. Again, we will delay the discussion of this dialog later.

Step 2 Please select OK to continue. MODUA will find the equivalent circuit and display its parameters: L = 0.5365 nH, C = 0.2150 pF, series R = 1.938e-2 ohms and Shunt R = 1.640e+6 ohms.

Step 3 Select File->Save SPICE File command on MODUA. MODUA will prompt you for the Settings for Saving SPICE.

The Cascading Factor is used to define how much of the LRCG you want to save into the file. For example, if you want 50% of the values to be saved into the SPICE equivalent circuit, you can specify the Cascading Factor = 0.5. The Capacitor Dropping Factor (C-Factor) determined when an off-diagonal capacitor is too small and should be neglected (The C_{ij} will be neglected if C_{ij} < C-Factor $\sqrt{C_{ii}C_{jj}}$. The Mutual Inductor Dropping Factor (L-Factor) determines when the mutual inductor should be neglected: The L_{ij} will be neglected if L_{ij} < L-Factor $\sqrt{L_{ii}L_{jj}}$. Other parameters are similarly defined. For this circuit, it is a single TLN. There are no off-diagonal elements.

Step 5 Select OK to accept the default setting. MODUA will save the equivalent circuit into the file: .\ie3d\practice\output\strip1t.lib. The file saves the equivalent circuit as a sub-circuit in standard SPICE format. You may want to see how close the equivalent circuit is to the original s-parameters.

Section 9.8 Comparing S-Parameters of the Extracted RLC to the Original S-Parameters

Step 1 Please keep the MODUA with the RLC extracted open. We will use it for later discussion. Start a new MODUA application.

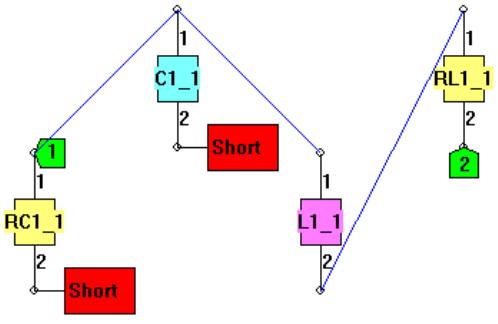


Figure 9.10 The imported SPICE file for the equivalent circuit of strip1t.sp.

Step 2 Select **File->Import SPICE File** command. Select strip1t.lib file to open it. The SPICE file will be imported (see Figure 9.10).

Step 3 Select **Process->Simulate** command on MODUA. MODUA prompts the user for the frequency points. Select Capture button. MODUA will prompt you for the name of the parameter file the user wants to capture the frequency points. Select .\ie3d\practice\output\strip1t.sp. The frequency points in strip1t.sp will be captured. Select OK. MODUA will simulate the circuit and display the results. Please select File->Save S-Parameters command. Please change the file name to: strip1t1.sp file (not to overwrite the original strip1t.sp file).

Step 4 Please use the **File->Display File Queue** command on MODUA to display the s-parameters in .\ie3d\practice\output\strip1t.sp simultaneously. Shown in Figure 9.11 is the comparison between the original s-parameters and those from the RLC-equivalent circuit. As you can see, the dB[S(2,1)] and Ang[S(2,1)] agree very well. They do not differ much up to 5 GHz. However, the agreement in dB[S(1,1)] and Ang[S(1,1)] is much worse. The dB[S(1,1)] is at about the same level at low frequency. However, the dB[S(1,1)] from RLC increases much faster than the original dB[S(1,1)]. The Ang[S(1,1)] is close at very low frequency only. The trend of Ang[S(1,1)] from RLC circuit is complete different from the trend of the original s-parameters. We can see obvious difference in dB[S(1,1)] and Ang[S(1,1)] at 1 GHz. However, as you can see, dB[S(1,1)] is smaller than –40 dB below 2 GHz. We can still consider the agreement in the s-parameters is very good below 2 GHz. In practical RLC-equivalent circuit, it is not easy to get very good agreement in S(1,1) over a wide frequency range. A user should also understand that any difference below –30 dB in dB[S(1,1)] will have little effect on the transient analysis on SPICE. We can say the RLC equivalent circuit is still good up to 3.7 GHz. Can we do a better job?

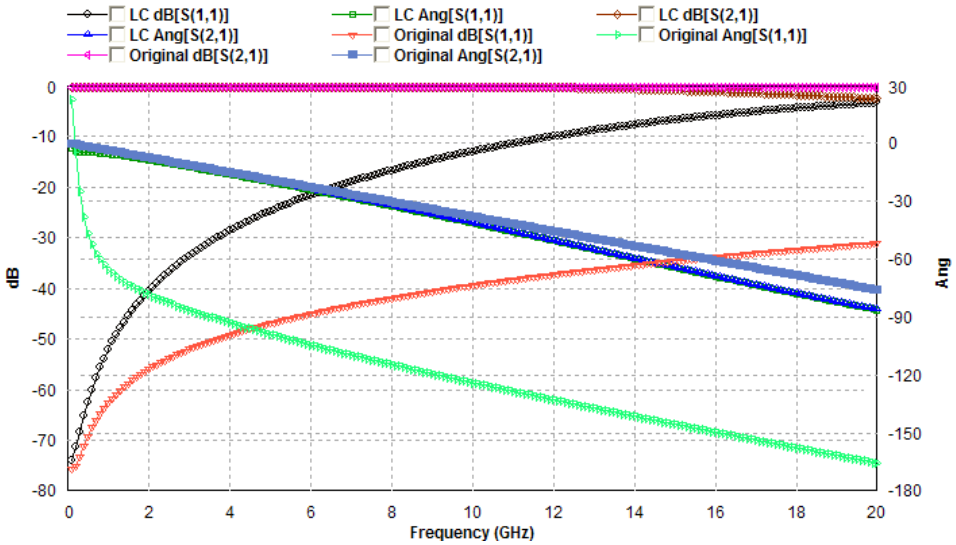


Figure 9.11 Comparison of original s-parameters and those from RLC.

Section 9.9 Multi-Segment RLC Equivalent Circuit Extraction in SPICE Format

In the above section, we discussed how we could use 1-single RLC-segment to approximate the original s-parameters. We expect the equivalent circuit will work in some limited bandwidth. To increase the application bandwidth, we can use multiple-RLC segments for it.

Step 1 On the MODUA with the extracted RLC values displayed on the window, please select **File->Save SPICE File** command. MODUA will prompt you for the Settings for Saving the SPICE.

Step 2 Please select the Change button to change the file name as: .\ie3d\practice\output\strip1t5.lib. Please change the LC-Segments from 1 to 5 (see Figure 9.12). Select OK. MODUA will save the 5 LC-Segments equivalent circuit into the strip1t5.lib file.

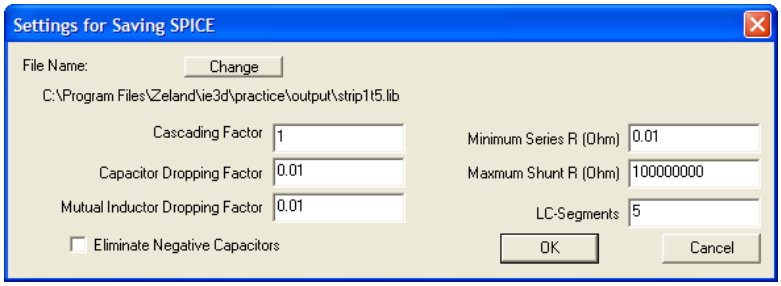
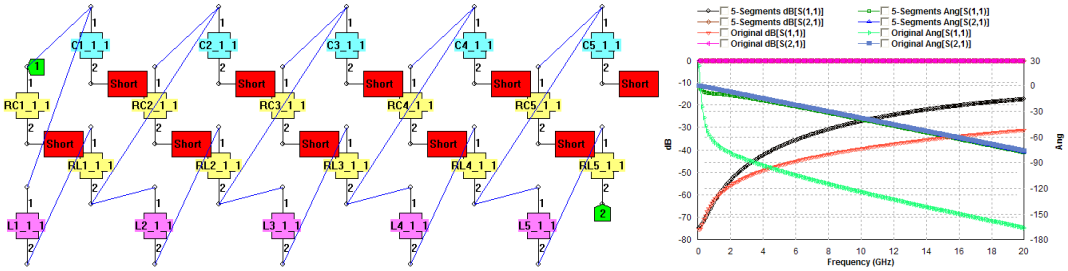


Figure 9.12 The Settings for Saving SPICE dialog.

Step 3 Select **File->New command** MODUA. Select NO to discard the change. Select **File->Import SPICE File** and select strip1t5.lib file. We will get the equivalent circuit shown in Figure 9.13a. It is a 5-segments LC equivalent circuit. We can compare its result with original s-parameters (see Figure 9.13b). As you can see, there is little difference in the Ang[S(2,1)] for the whole frequency range. The dB[S(1,1)] starts deviating at 2 GHz. Again, dB[S(1,1)] is very small. We can consider the equivalent circuit is accurate below 8 GHz where dB[S(1,1)] < - 30 dB. In practical applications, the power spectrum is concentrated at the low frequency. Slightly different dB[S(1,1)] at high frequency will not affect the waveforms significantly anyway.

Apparently, the multiple LC-segment equivalent circuit improves the valid bandwidth of the equivalent circuit substantially. However, we cannot indefinitely increase the bandwidth of the equivalent circuit by increasing the LC-Segments.



(a) 5 LC-Segment equivalent circuit (b) Comparison to original

Figure 9.13 The 5 LC-Segment equivalent circuit and its result.

Section 9.10 Wide Band RLC Equivalent Circuit Extraction

We have demonstrated how we can use multiple LC-segments to improve the valid frequency range of the RLC equivalent circuit. The extracted RLC values from the above 2 sections are compatible with SPICE and the elements have physical meanings at low frequency. Unfortunately, we can not expand the application frequency range of the RLC circuit indefinitely.

Is it possible we can match the s-parameters of the extracted RLC to the original one in the whole frequency range? The answer is yes. We have implemented a sophisticated wide-band RLC extraction scheme into the MDSPICE simulator. The feature is transferred to the s-parameters processing dialog of MGRID. We can get perfect match to the original s-parameters over a wide frequency range. However, we can not guarantee the extracted RLC components have physical meanings. Also, we can not guarantee the

R, L and C components have non-negative values even though they are frequency independent and the resulting RLC circuit is compatible with traditional SPICE.

We will not discuss the detail on using the MDSPICE to extract the wide band RLC circuit from s-parameters. It is the command Find RLC-Equivalent Circuit in Display->S-Parameters dialog. Figure 9.14 shows the comparison of the original s-parameters (in strip1t.sp) and the s-parameters from the wide band RLC circuit (in strip1t_wb.lib) extracted on MDSPICE. Perfect agreement over the complete frequency range is observed. The wide band SPICE file is saved in: .\ie3d\samples\output\strip1t_wb.lib file. You can import it onto MODUA and see how it looks like. It has tens of RLC elements and the values of the elements are in strange range. Please do not neglect those RLC elements with extreme values. Dropping them will affect the matching of the RLC circuit to the original s-parameters.

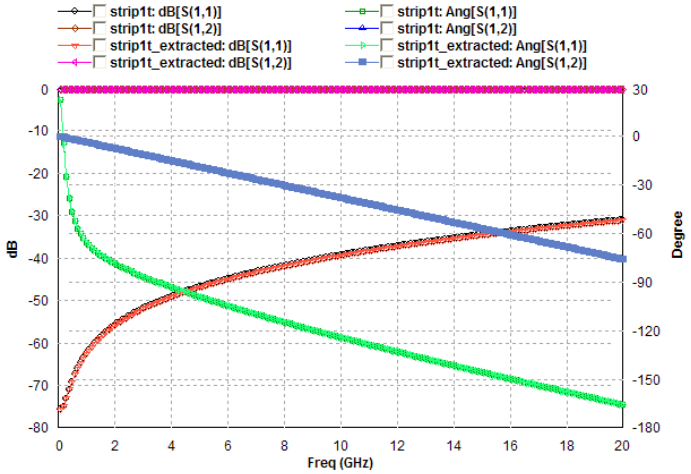


Figure 9.14 Perfect match between original s-parameters and the wide-band extraction results.

The wide band SPICE extraction is also implemented in the s-parameters processing dialog. Select "Extract Wide Band SPICE Model from S-Parameters" and select "Go!". MGRID will extract it for you.

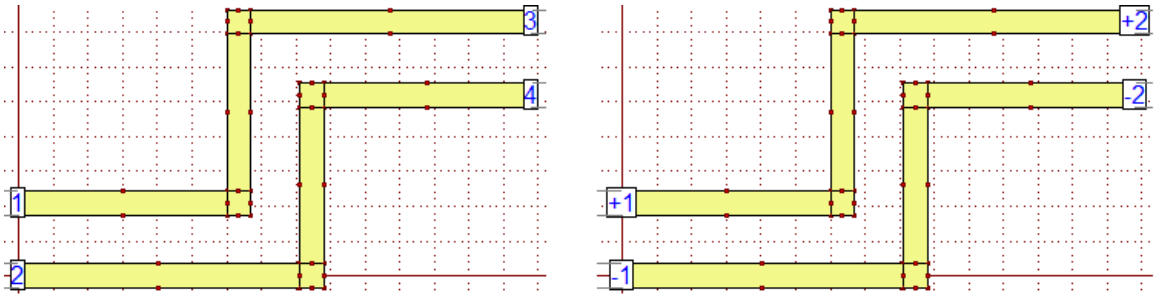
Section 9.11 Simulation of a Coupled Stripline Structure

Coupled stripline structures are widely used in modern high-speed digital circuits. A coupled stripline structure is created and saved in: .\ie3d\samples\strips2t.geo. The stripline has the same cross-sectional parameters as the one shown in Figure 9.4. The center-to-center distance between the two traces is 21-mils. It means that the gap between the traces is 14-mils. The strip2t.geo is shown in Figure 9.15a and it is a 4-port structure. The physical lengths of the two traces are identical (200 mils). In a typical digital circuit, the coupled stripline circuit is normally used as a differential pair. Saved in .\ie3d\samples\strips2t2.geo is the file for differential application. When we use the positive and negative ports for the structure, we imply that the sources for the ports can enforce the differential condition, which is that the current going out of the positive terminal is the same as the current going into the negative terminal. However, such a condition may not guarantee that the potential of the negative terminal is the opposite of that of the positive terminal when even mode of the circuit exists. If it is a pair of coupled strips, there exists no even mode in the structure. We have to model it as a differential pair shown in Figure 9.15b, and we should not model it as shown in Figure 9.15a.

The current structure is a coupled stripline structure. There exist 2 ground planes. Any non-symmetry in the structure will excite the even mode. Even though the lengths of the two traces are the same, we cannot guarantee no even mode is excited due to non-symmetry. It will be better to model the structure as a 4-port structure. Conversion of the resulting 4-port s-parameters to differential 2-port s-parameters is available on the Differential Conversion in the Process menu of MODUA.

The strips2t.geo is created by growing thickness on the strips2.geo. If we grow thickness on the Z = 10-mil layer on strips2.geo, we will get the exact structure as the strips2t.geo.

Step 1 Run MGRID. Open .\ie3d\samples\strips2t.geo. Save it as: .\ie3d\practice\strips2t.geo. Simulate the 4-port structure. Whether you enable the AEC or not, the mesh will be the same. No edge cells are created whenever you build thickness to the structure. Basically, the side strips in the thickness will take care of the edge effects efficiently. There is no need to have edge cells and the results are normally quite accurate. The resulting s-parameters are saved in: .\ie3d\practice\output\strips2t.sp after the simulation. MODUA will be invoked to display the s-parameters in strips2t.spt file if you enable AIF.

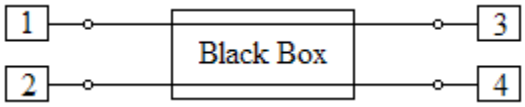
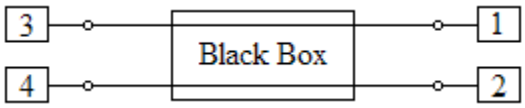


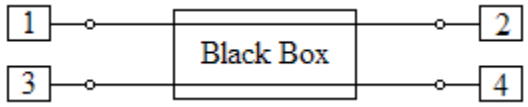
(a) Coupled striplines modeled as 4-port structure (b) Coupled striplines modeled as differential 2-port

Figure 9.15. The comparison of general 4-port and differential 2-port configurations.

Step 2 Use MODUA to display .\ie3d\practice\output\strips2t.sp, Select **Process->LC-Equivalent** command. MODUA will warn you of abandoning the simulation results. Please select OK to continue. MODUA will prompt you for the frequency points. Please select Delete All to delete all the frequency points. Then, check the No.2 f = 0.2 GHz. Select OK. MODUA will prompt you for the Port Definition Style for the Equivalence. There are 4-choices and their meanings are documented in Table 9.5.

Table 9.5 The different port definition styles for the equivalence.

Port Definition Style	Schematics
Open Circuit for Output Ports: We can connect the Open Circuits to some terminals to indicate they are the output terminals. In fact, the capacitance extraction for the MIM capacitors in Chapter 5 is a case. In that case, we tricked the MODUA to define a port and an open circuit to the same terminal.	
Last N/2 Ports for Output Ports: We can setup the design so that the No.1 and No. (N+1) port are for a trace, the No.2 and No. (N+2) ports are for another trace. The No. N and No. 2N ports are for the last trace.	
First N/2 Ports for Output Ports: We can setup the design so that the No.(N+1) and No.1 port are for a trace, the No. (N+2) and No. 2 ports are for another trace. The No. 2N and No. N ports are for the last trace.	

Even Ports for Output Ports: We can setup the design so that the No.1 and No.2 ports are for a trace. The No.3 and No.4 ports are for the 2nd trace. The No. (2N-1) and No. 2N ports are for the last trace.	

Step 3 For our case, we just select the default "Last N/2 Ports for Output Ports". Select OK to continue. The equivalent circuit is extracted at 0.2 GHz: L11 = 1.641 nH, L12 = 6.106e-2 nH, L22 = 1.641 nH, C11 = 0.6801 pF, C12 = 2.932e-2 pF, C22 = 0.6802 pF, Series R11 = 3.623e-2, Series R22 = 3.623e-2, etc. Save the equivalent circuit as: .\ie3d\practice\output\strips2t.lib.

Section 9.12 Time-Transient Analysis and Eye Diagram on MDSPICE

It is of much interest for signal integrity engineers to look at the time transient response and the eye-pattern of a circuit. Time-transient and eye pattern analysis using SPICE circuit and s-parameters are available on the MDSPICE. The MDSPICE model on strips2t.sp is created in c:\ie3d\samples\strips2t3.mds.

As an example, we may want to apply the 80-ps rise-time clock signal in Figure 9.16 to the port 1 of the structure in Figure 9.15a (strips2t.geo) while the ports 2, 3 and 4 are terminated with 50-ohms. For the 80-ps rise time clock signal, the ramp time and the falling time are equal and they are 100-ps. The period is 450-ps. The transient analysis on the strips2t.sp using the MDSPICE takes less than 1 second. The results are shown in Figure 9.17. As you can see, the output signal at port 3 has a clear delay to the input stimulus. The crosstalk between ports 2 and 4 is small.

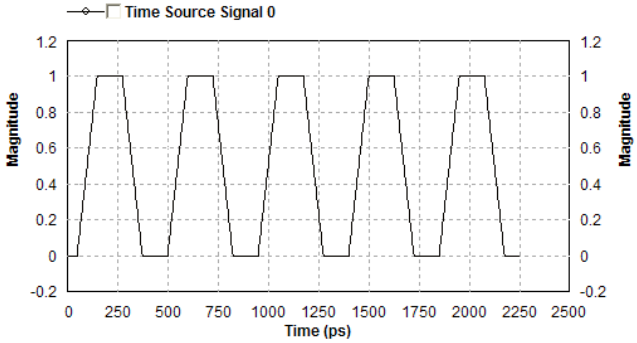


Figure 9.16 The clock-waveform to be applied to the port 1 of strips2t.sp.

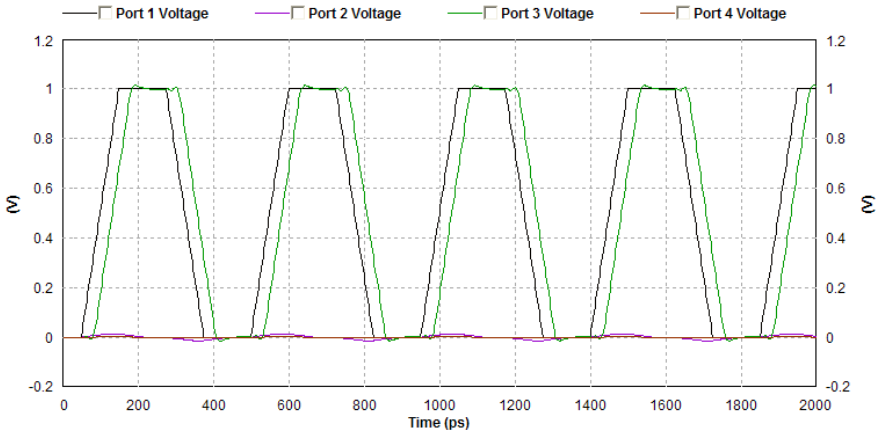


Figure 9.17 The transient analysis results on strips2t.sp.

Another display of great interest to signal integrity users is the eye diagram display. Figure 9.18 shows an eye pattern display on the circuit. The eye pattern basically shows the time transient response of the circuit to a series of random clock signals with random jitters. It is a way to judge how good a communication system is.

Section 9.13 Delay Analysis of Differential and Ultra Long Transmission Lines

Ultra long transmission lines have been widely used in high-speed networking back-planes. The length of the transmission line can be more than 1-foot. Accurate time domain simulation of ultra-long lossy transmission lines is an extremely difficult task. Using the MDSPICE, we will be able to perform accurate time domain simulation on ultra-long lossy transmission lines. We will use the strips2t.geo as an example. The length of strips2t.geo is 200-mils. Assume we have 64 of such structures cascaded together to form a transmission line of length 12.8-inches. Also, we want to excite the port 1 and port 2 using a differential signal, and terminate the ports 3 and 4 with 50-ohms on each end. We want to know the waveforms at the output ports 3 and 4. The MDSPICE file is saved in: .\ie3d\samples\strips2t3100.mds. A time domain simulation on it takes couple seconds. The input stimulus and output waveforms are shown in Figure 9.19. As you know, the strips2t.geo is not a complete symmetrical structure. When we cascade 64 such slightly non-symmetrical structures together, we will be concerned about the symmetry of the signals at the outputs. It is shown that the differential signals at output are still quite symmetrical in Figure 9.19. It is very interesting to see that the output signals are delayed by about 2187-ps. It is about 170 ps per inch.

We will end our discussion on the MDSPICE here. Interested users can contact us for the MDSPICE simulator.

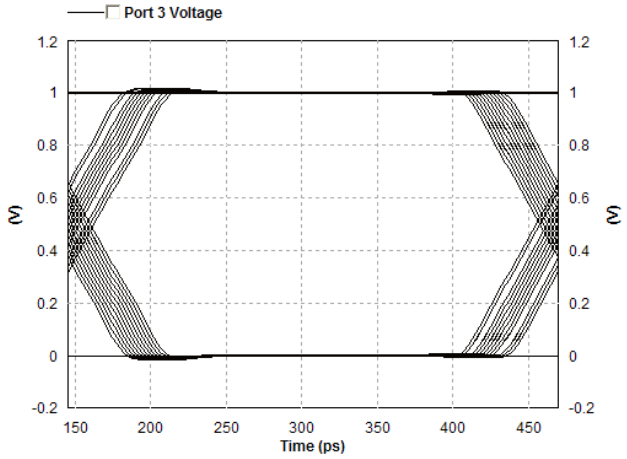


Figure 9.18 The eye-pattern display on the strips2t.sp on MDSPICE.

Section 9.14 Modeling Complicated Packages

High-speed digital circuit packages are complicated structures. We have implemented different schemes to make the modeling of the structures easier. In this section, we will discuss the key points in modeling them.

The structure we are going to discuss is saved in **.\ie3d\samples\frame.geo** and shown in Figure 9.20. It is a package with 10-leads. Normally, such a structure can be constructed in other format and it can be imported into IE3D. Saved in .**\ie3d\samples\frame1.geo** is the preliminary structure with the dielectrics and the metallic defined. More work is needed to clean it up. In the following steps, we will discuss the key procedures.

Step 1 Run MGRID. Open **.\ie3d\samples\frame1.geo**. Select **Param->Basic Parameters** command to check the basic parameters. The ground plane is a z = 0 mil. It is air from 0 to 7 mil. The substrate is from 7 to 30 mils. The polygons are 8-mil thick. The top surface of the lead frame will be at z = 38 mils. The bottom surface of the lead frame will be at z = 30 mils. The polygon at Z = 34 represents the die pad. We are going to connect some wire bonds from the lead frames to somewhere above the die pad. The die pad will be considered as the local ground plane for the chip. Each lead is 16-mils width. We need to consider the metallic thickness for it.

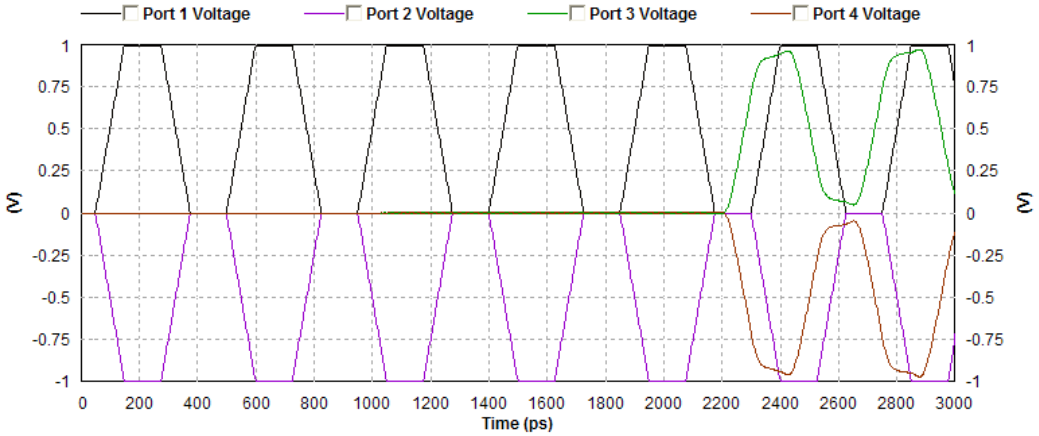
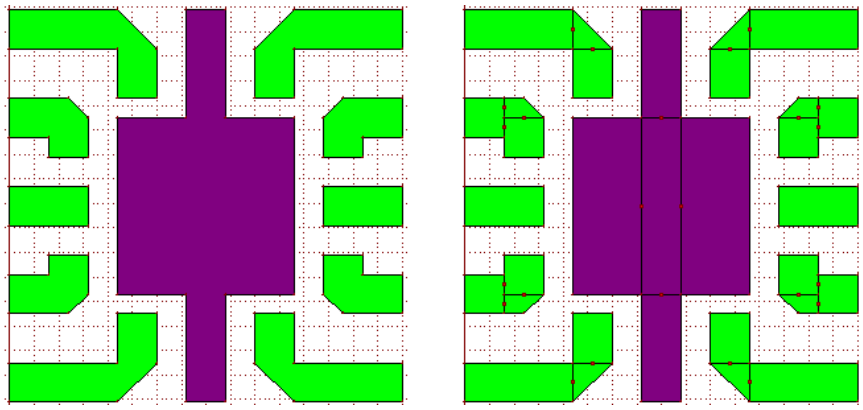


Figure 9.19 The differential stimulus at ports 1 and 2 and output waveforms at ports 3 and 4.



(a) (b)

Figure 9.20 (a) The preliminary structure saved in c:\ie3d\samples\frame1.geo.
(b) The structure after Rectanglization.

Step 2 Select **Adv Edit->Rectanglization** command to break the polygons into as many rectangles as possible as shown in Figure 9.20 (b). Sometimes, **Rectanglization** may miss some of the dividing. We can use the combination of **Input->Set to Closest Vertex**, **Input->Connect to Edge Perpendicularly** and **Edit->Divide Polygon** commands to divide those polygons manually. We will not do it here because polygons after Rectanglization are good enough (see Figure 9.20b).

Step 3 Click at the Z = 34 mil layer in the layer window. Select **Edit->Layer->Grow Metal Thickness on Layer** command. Select **Negative-Z** direction. Select OK. MGRID will build the thickness for the die pad. It will span from Z = 26 to Z = 34 because of the Negative-Z direction.

Step 4 Click at the Z = 38 mil layer in the layer window. Select **Edit->Layer->Grow Metal Thickness on Layer** command. Select **Negative-Z** direction. Select OK to accept the other default settings. MGRID will build the thickness for the leads. Save the result as .\ie3d\practice\frame2.geo.

The next step is to create the bond pads above the die pad. We are going to build 10 wire bonds. The start and end points of the wire bonds are listed in Table 9.6. The start points will be on the lead frame. We need to build the bond pads at the end point as small rectangles of size 4 by 4 mils.

Step 5 Select **Input->Create and Edit Vertices** command. Select **Import** button. Select the **.\ie3d\samples\frame2p.txt** file. Select OK to accept the default offset values of 0's. The vertices defined in **frame2p.txt** are imported into the list.

The **frame2p.txt** file contains the locations of all the end points. Each line contains 4 items: (x, y, z, b). The (x, y, z) are for the location of a vertex and "b" is for the bulge of the segment. It is for entering a curved segment. For a straight segment, b = 0. You can use an ASCII editor to open it for the format. Using the **Input->Create and Edit Vertices** command is a convenient way to create the vertex series. It is equivalent to using the mouse input or the keyboard input to create them. You can also export the entered vertices and their bulges to an ASCII text file.

Step 7 Select OK to continue. The entered vertices are created at the center of the bond-pad locations. They look just as you are entering the vertices for a polygon.

Table 9.6 The start and end locations of the wire bonds.

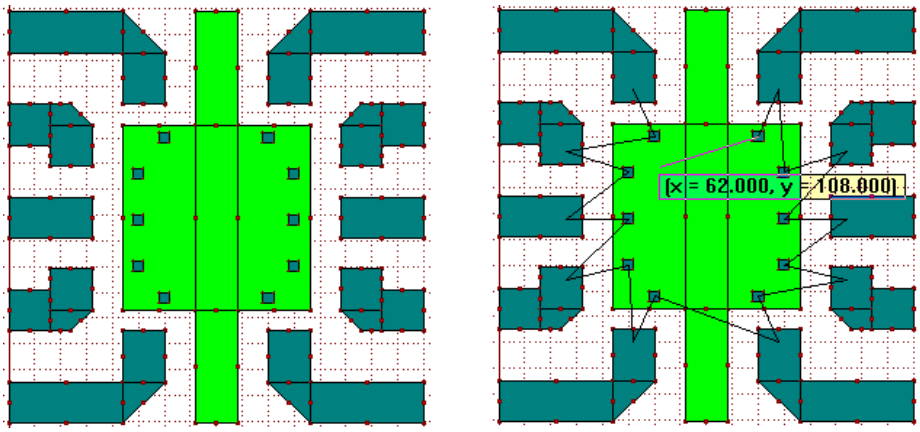
Wire Bond	Start Point			End Point		
	X	Y	Z	X	Y	Z
No.1	52	138	38	60	120	38
No.2	26	114	38	50	106	38
No.3	26	88	38	50	88	38
No.4	26	64	38	50	70	38
No.5	52	40	38	60	58	38
No.6	108	58	38	100	58	38
No.7	134	64	38	112	70	38
No.8	134	88	38	110	88	38
No.9	134	114	38	110	106	38
No.10	108	138	38	100	120	38

Step 8 Select **Entity->Rectangle** command. Change the **Length** and **Width** to 4 mils. Select OK to continue. The bond pads are created as shown in Figure 9.21a. Please save the file as: .\ie3d\practice\frame3.geo.

The next step is to build the bond wires. In order to build the wire bonds, we need to enter a series of vertices for the wire bond's location. If we want to build N-wire bonds in one shot, we should first enter the 2N vertices first. Each pair of the vertices will be for one wire bond. The order of the vertices is important. We have created the vertices in the text file: **.\ie3d\samples\frame2.txt**. We are going to use the file.

Section 9.15 Building a Set of Wire Bonds

Step 9 Select **Adv Edit->Create and Edit Vertices** command. Select **Import** button. Select **c:\ie3d\samples\frame2.txt** file. Select OK to accept the default offset values of 0's. Select OK to continue. The vertices will be imported and shown in Figure 9.21b.



(a) Structure after the bond pads built (b) Display after frame2.txt imported.

Figure 9.21 The display of the structure in 2 different steps.

Step 10 Select **Edit->Wire Bonding** command. MGRID will prompt you to change the default parameters. Please select style as: 3-Segments (JEDEC). Change the **Radius** to 1 mil. Change the **Wire Cross-Section** to 4 for square wire bonds. Square shape is normally very accurate for approximating the thin wire bonds. In case a wire bond is a ribbon or if we want to save some cells, we can **Wire Cross-Section** = 2). Click at **Z-Extremum**. Enter 50 mils. Select **Update All Wire Bonds** option to exit. MGRID will change the Z-Extremum value of each wire bond to 50 mils. Select H-Segment button. Change its value to 5 mils. Make sure **Change Beta** is selected. Select **Update All Wire Bonds** to exit. MGRID will change the H-Segment value of each wire bond to 5. Select OK to continue. It will take MGRID some seconds to build the 10 wire bonds.

Please save the geometry into an intermediate file **.\ie3d\practice\frame4.geo**. If you open the 3D View window, you will see the wire bonds, as shown in Figure 9.23. There are many layers created in the Layer Window. If you do not want to see the big list, you can un-check the List All Layers check box on the Layer Window. Only those layers with 2D polygons on them are listed.

We will build some ports on the lead frame and between the bond pads and the die pad.

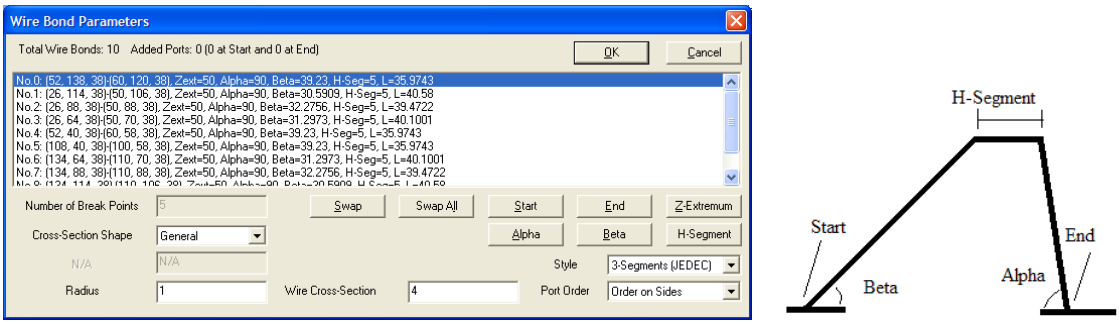


Figure 9.22 The Wire Bond Parameters dialog for building the wire bonds.

Section 9.16 Defining Ports on the Lead Frame and the Wire Bond Pads

Step 11 Please select Port->Port for Edge Group. Check all the layers including the 3D layer in the Layer Window. Window the starting points of the leads to define 10-ports on them sequentially (see Figure 9.24a).

You may notice from Figure 9.23 there are some vertical polygons covering the end of each lead. Some older versions of IE3D may have difficulty in defining an extension port on it with then end covered. The newer versions will not have such a limitation.

We are going to define vertical localized ports between the bond pads and the paddle. We will look at the edges for the No.11 port which is on the bond pad for the No.1 lead (see Figure 9.24b). You may see part of the polygons for the wire bonds are covered by the polygons on Z = 34. It is just due to the fact some layers are elevated in the top view.

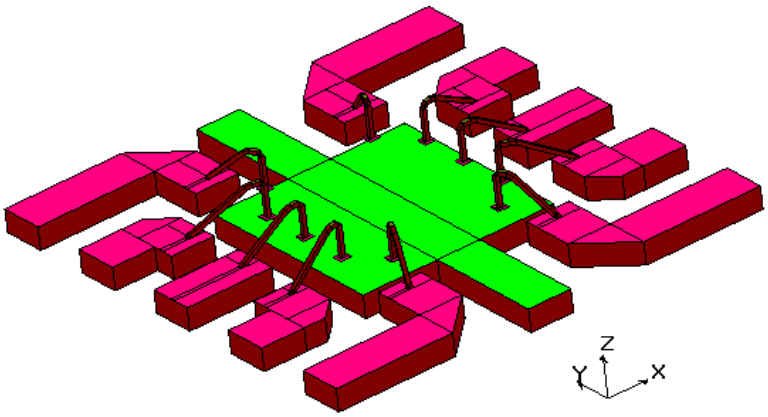
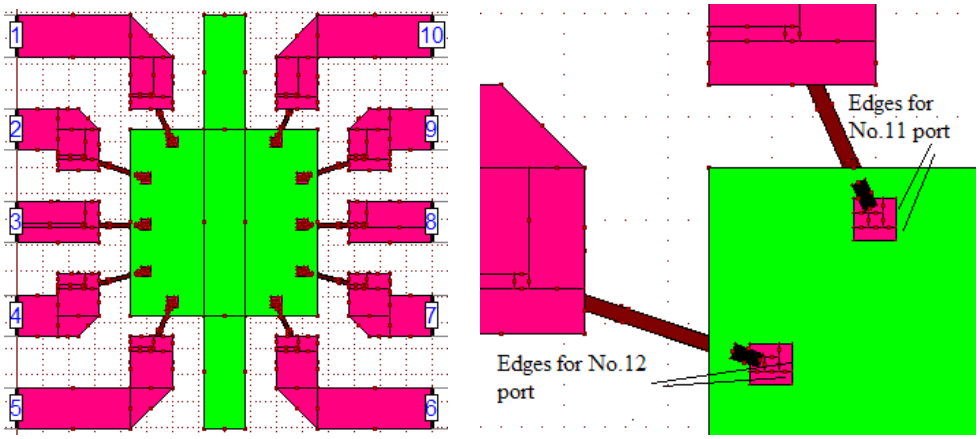
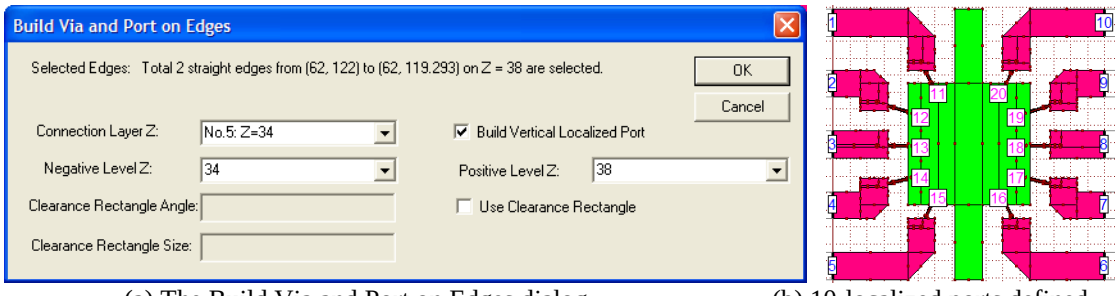


Figure 9.23 The 3D view after the wire bonds are built.



(a) The structure with 10-extension ports defined (b) The edges for No.11 and No.12 ports

Figure 9.24 The structure with 10 ports defined and the edges for the No.11 and No.12 ports.



(a) The Build Via and Port on Edges dialog (b) 10-localized ports defined.

Figure 9.25 The Build Via and Port on Edges dialog and the final structure with all ports defined.

Step 12 Un-check List All Layers in the Layer Window. Only those layers with 2D polygons are listed. Select **Edit->Select Vertices** command. Check No.9 layer at Z = 38 mils to focus the selection to this layer only. Window the two edges for the No.11 port in Figure 9.24b to select them.

Step 13 Select Adv Edit->Build Via Connection on Edges command. Select Connection Layer Z = No.5 Z = 34, Negative Level Z = 34 and Positive Level Z = 38. Check Build Vertical Localized Port. Select OK. MGRID will build the port 11 on the edges for bond pad 1 (see Figure 9.25b).

Step 14 Repeat Step 12 and Step 13 for edges for other nine ports. We will get the final structure in Figure 9.25b. Please save it as .\ie3d\practice\frame6.geo.

Please simulate it from 0.1 to 10 GHz for 100 frequency points. It may take about 15-30 minutes to get the s-parameters for the whole frequency range.

We can perform SPICE simulation based upon the s-parameters on the MDSPICE. We can also use MDSPICE to extract the wide band SPICE model from the 20-port s-parameters. However, you should realize that we could not use the **LC-Equivalent** command on MODUA to convert the 20-port s-parameters into the RLC-equivalent circuit.

If you check the current path, you will realize a good equivalent circuit for the structure in .\ie3d\practice\frame6.geo should be similar to the one shown in Figure 9.26. What is shown in Figure 9.26 is a simplified one for illustration. The actual one should include the series R, shunt R and shunt C for each of the 10 arms. There should be mutual inductors and mutual capacitors between arms. There should be series R and series L between ports 11 to port 12.

The Process->LC-Equivalent command in MODUA is specially for coupled interconnects. In practical application, we may use one or more leads frame to connect the paddle to the ground. At this time, the paddle, which serves as the reference for port 11 to port 20, is not grounded. We can not apply the Process->LC-Equivalent command on MODUA on the 20-port s-parameters to extract the equivalent circuit for it because the model does not fit the 20-port s-parameters.

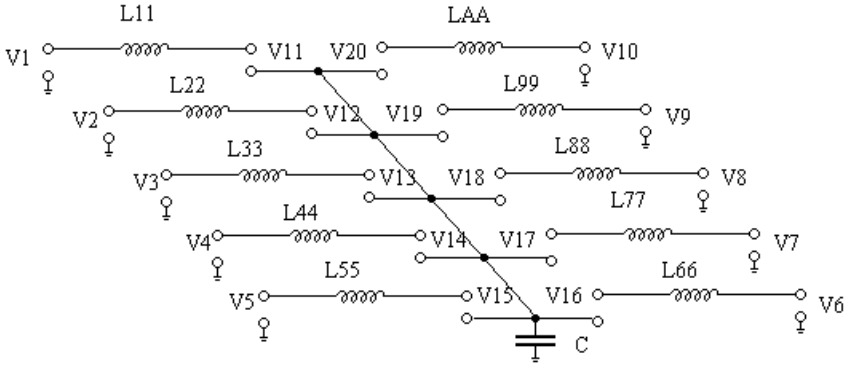


Figure 9.26 A good simplified equivalent circuit for the lead frame.

In case we know which leads are connecting the paddle to the ground, we will be able to use the Process->LC-Equivalent command to find the equivalent circuit. For example, if lead No.9 (from port 9 to port 19) and lead No.10 (from port 10 to port 20) are connected to the ground, we can use the MODUA setup in Figure 9.27 to find the 16-port s-parameters with the 4-port short circuited. Then, we can use Process->LC-Equivalent command on MODUA to find the SPICE equivalent circuit for the resulting circuit.

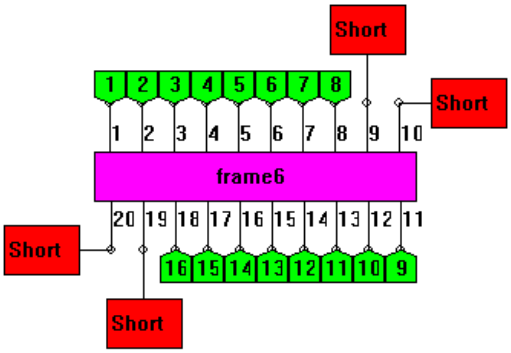


Figure 9.27 MODUA setup for 16-port s-parameters when the last 2 leads are grounded.

Section 9.17 Simplified Structure for the Lead Frame

In practical applications, the paddle or the common ground of the ports 11 to 20 will be connected to the ground through some leads. In such a case, the capacitance from the die pad to the ground (or the C in Figure 9.26) is not critical because its impedance is much higher than that of a lead. We can omit the C. One thing we can do is to short the common ground of ports 11 to 20 to the infinite ground at Z = 0. If we short the paddle to the ground, the C in Figure 9.27 will does not have any effect. We can replace the C as direct short circuit to the ground. We can find the equivalent circuit of the 20-port structure using the **Process->LC-Equivalent** command on MODUA.

We can select the edges on the square of the paddle at Z = 26. Then, use the Edit->Add Via on Edges command to build some shorting pin from Z = 26 to Z = 0 for the final structure in .\ie3d\samples\frame.geo.

Please simulate the structure in .\ie3d\samples\frame.geo. It is a medium size structure. It creates about 3000 unknowns and takes a few minutes per frequency point on a high performance PC. We can use **Process->LC-Equivalent** command on MODUA to extract the equivalent circuit of the structure. Table 9.7 shows the parameters of the equivalent circuit at 0.1 GHz. The lead 1 is the one from port 1 to port 11. The lead 2 is the one from port 2 to port 12, etc. The file is saved in .\ie3d\samples\output\frame.mat file. You can use the File->Open Matrix File on MODUA to open it and use the File->Save SPICE File to convert it to SPICE net-list. The shunt R is quite large and we can omit it. The shunt C's are also quite small. The series L is from 1.17 nH to 1.47 nH. We define the metal as gold. The series R is about 0.03-ohms.

Table 9.7 The parameters of the equivalent circuit of **frame.geo** at 0.1 GHz.

Series L (nH)	1	2	3	4	5	6	7	8	9	10
1	1.46784	0.33193	0.17333	0.10981	0.07631	0.03477	0.03748	0.04486	0.0583	0.09415
2	0.33193	1.2382	0.32663	0.17979	0.11139	0.03788	0.03716	0.04024	0.04543	0.0583
3	0.17333	0.32663	1.17246	0.3284	0.17684	0.04527	0.04029	0.03968	0.04024	0.04486
4	0.10981	0.17979	0.3284	1.22424	0.33855	0.05865	0.04558	0.04028	0.03717	0.03749
5	0.07631	0.11139	0.17684	0.33855	1.46222	0.0941	0.05867	0.04527	0.03788	0.03477
6	0.03477	0.03788	0.04527	0.05865	0.0941	1.45485	0.33872	0.17693	0.11137	0.07626
7	0.03748	0.03716	0.04029	0.04558	0.05867	0.33872	1.21305	0.32798	0.17959	0.10968
8	0.04486	0.04024	0.03968	0.04028	0.04527	0.17693	0.32798	1.16536	0.32725	0.17342
9	0.0583	0.04543	0.04024	0.03717	0.03788	0.11137	0.17959	0.32725	1.22056	0.33224
10	0.09415	0.0583	0.04486	0.03749	0.03477	0.07626	0.10968	0.17342	0.33224	1.46001
Series R (Ω)	0.02869	0.02949	0.02494	0.02700	0.02713	0.02629	0.02593	0.02315	0.02663	0.02693
Shunt C (pF)	1	2	3	4	5	6	7	8	9	10
1	0.14937	0.02198	-0.01273	-0.00017	0.00027	0	0.00013	-0.0002	0.00043	0.00188
2	0.02198	0.06794	0.00216	0.00736	-0.0007	0.00027	0.00014	0.00013	0.00015	0.00042

3	-0.01273	0.00216	0.08525	0.03942	-0.01513	-0.00026	0.00004	-0.00011	0.00013	-0.00022
4	-0.00017	0.00736	0.03942	0.03754	0.01634	0.00017	0.00005	0.00004	0.00013	0.00013
5	0.00027	-0.0007	-0.01513	0.01634	0.15841	0.00184	0.00018	-0.00024	0.00028	0
6	0	0.00027	-0.00026	0.00017	0.00184	0.15906	0.01633	-0.01558	-0.00069	0.00028
7	0.00013	0.00014	0.00004	0.00005	0.00018	0.01633	0.0366	0.03945	0.00837	-0.00024
8	-0.0002	0.00013	-0.00011	0.00004	-0.00024	-0.01558	0.03945	0.08838	0.00014	-0.01325
9	0.00043	0.00015	0.00013	0.00013	0.00028	-0.00069	0.00837	0.00014	0.06933	0.02237
10	0.00188	0.00042	-0.00022	0.00013	0	0.00028	-0.00024	-0.01325	0.02237	0.14982

Section 9.18 Simulation of Multilayer PCB, BGA and Wire Bonds

IE3D can be used to design very complicated interconnect problems. Our final example in this chapter will demonstrate such an example. Saved in .\ie3d\samples\bga.geo is the final geometry. We start from the initial geometry .\ie3d\samples\bga1.geo. It should be a geometry just imported from a GDSII or DXF file. The substrate and other basic parameters are already defined. In practical application, we should have built a template for importing it. After it is imported, we should have got the bga1.geo file. We are going to show to you how you can clean the structure to get the bga.geo file.

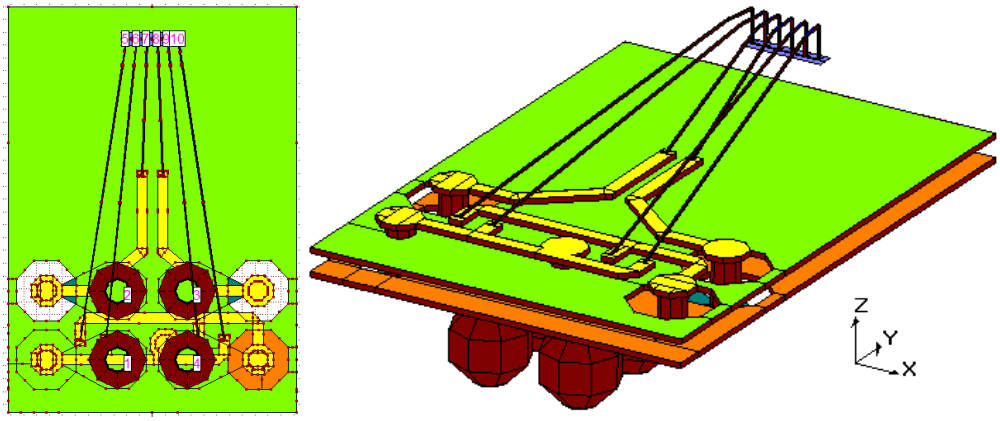


Figure 9.28 The 3D view of the final structure in bga.geo.

Assume all the metal layers have the same thickness of 40 microns. There are total 10 metal layers in bga1.geo. Most of the layers are shown in Figure 9.29. The relationship between the layers is documented in Table 9.8. The current paths are documented in Table 9.9.

Table 9.8 The metal layers in Figure 9.29.

Layer	Current Z-Coordinate	Description
M1	0	It is where the PCB ground is located.
M2	590	4 ball pads with 4 solder balls connected to the M1. The ball pads span from Z = 550 to 590. The ball pad B1 is for the ground plane. It is connected to the ground plane at M4 through the via V1.
M3	670	It is the power plane of the structure. It will span from Z = 670 to 710. It is connected to the ball pad B4 through via V4. It is not seen in Figure 9.29. It has the same size as M4 except the holes for the vias are different. It is connected to trace T4 through via V4.
M4	840	It is the ground plane of the structure. It spans from Z = 840 to 880. It is connected to the ball pad B1 through via V1. It is connected to trace T1 through via V1 and via V5.
M5	960	It consists of traces T1, T2, T3 and T4. It spans from Z = 960 to 1000. T1 is connected to V1 and V2 to PCB ground. It is connected to wire bond W1a and W1b.

M6	1250	It is the die ground serving as the common reference for the 6 ports to be defined at the upper ends of the 6 wire bonds.
M7	1300	It is not a metal layer in the final geometry. It defines the shapes to be used to build holes on the power plane and ground plane. H1 will build a hole on the power plane M3. H2 and H3 will build holes on both the power plane M3 and the ground plane M4. H3 will build a hole on the ground plane M4.
M8	1500	It is not a metal layer in the final geometry. It defines the shapes for the via pads. Since the shapes of the via pads are already built in M2 and M5. This layer is in fact not used. They will be removed.
M9	1700	It defines the shapes of the 5 vias. V1 connects B1 in M2 to M4, and M5 to the T1 in M5. V2 connects B2 in M2 to T2 in M5. V3 connects B3 in M2 to T3 in M5. V5 connects M4 to the T1 in M5.
M10	1900	It defines the shapes of the 6 wire bonds. The W1a and W1b are for the ground plane path. W2 is for the positive signal path. W3 is for the negative signal path. W4a and W4b re for the power plane path.

Table 9.9 The paths of the different signal nets.

Path of Current	Description
Ground Path	It starts from solder ball B1 to via V1. It connects to ground plane at M4. The V1 connects M4 to T1 in M5. The V5 connects M4 to T1 in M5 too. Then, the wire bond W1a and W1b connect T1 to the chip.
Power Path	It starts from solder ball B4 to via V4. It connects to power plane at M3. The V4 connects M3 to T4 in M5. The wire bonds W4a and W4b connect T4 to the chip.
Positive Path	It starts from solder ball B2 to via V2. It connects to T2 in M5. The wire bond B2 connects T2 to the chip.
Negative Path	It starts from solder ball B3 to via V3. It connects to T3 in M5. The wire bond B3 connects T3 to the chip.

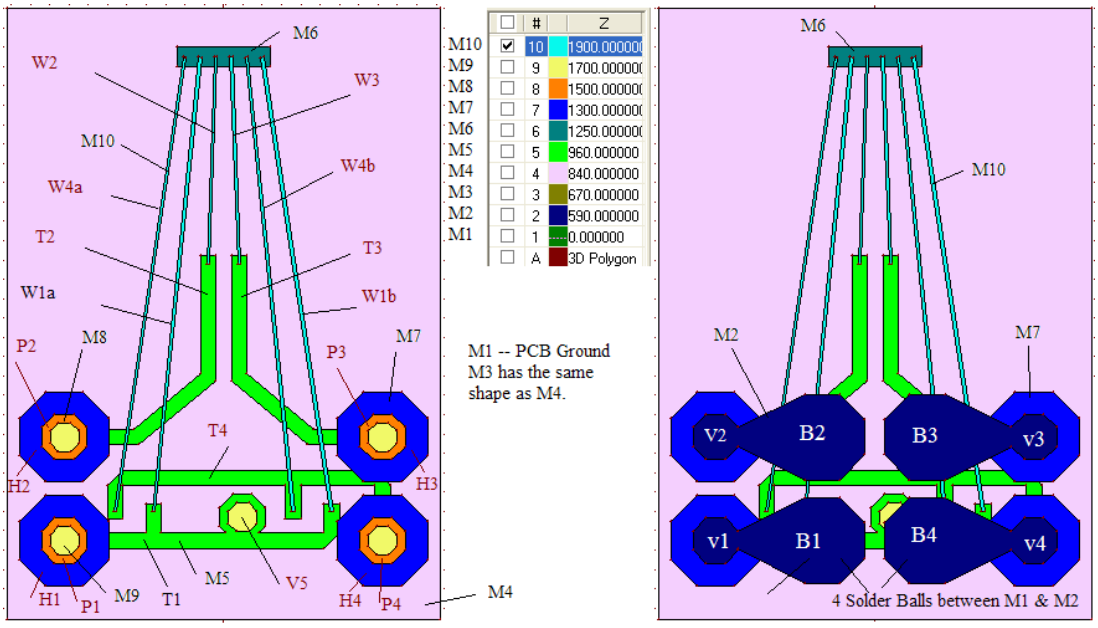


Figure 9.29 The 10 metal layers of the initial structure (bga1.geo).

The other parameters about the structure are listed here. The dielectric has permittivity of 3.6 and loss tangent of 0.005. It spans from Z = 590 to 960. Each solder ball has radius of 300 microns. The wire bonds are 20 microns in diameter. The wire bonds reach 1500 microns with the horizontal segment of length of 260 microns. The procedure is documented in Table 9.10. We will describe all the steps in the following.

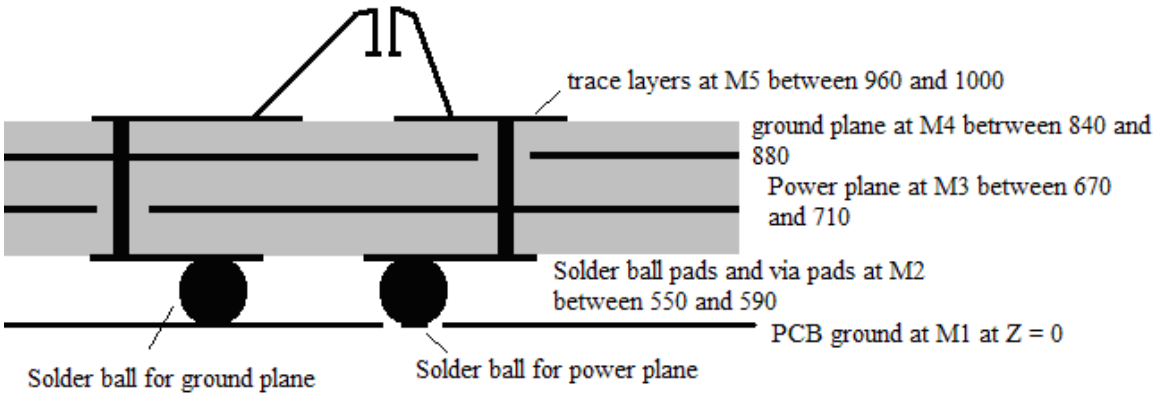


Figure 9.30 The cross-sectional illustration of the structure.

Table 9.10 Procedure in building the 3D model from the 2D model.

Task	Description
1	Try to use Adv Edit->Convert Polygons in Shape to convert polygons with many vertices representing circular shapes into fewer segments in the shape. It will reduce many unknowns in a simulation. Use Adv Edit->Remove Redundant Vertices to remove many vertices in curved structures.
2	Use the shapes in M7 to build holes in M3 and M4. H1 will build holes in M3. H2 and H3 will build holes in M3 and M4. H4 will build hole in M4.
3	Build metallic thickness in M2, M3, M4 and M5.
4	Build solder balls from M1 to M2.
5	Build via V1 connecting layers M2, M4 and M5. Build vias V2 and V3 connecting layers M2 and M5. Build via V4 connecting layers M2, M3 and M5.
6	Build wire bonds connecting M5 to M6.

Step 1 Run MGRID. Open .\ie3d\samples\bga1.geo.

We do not need to do the Task 1 (or convert polygons and remove redundant vertices). Normally, polygons representing vias and holes are round shapes and they may have many vertices on them. For this particular example, the shapes have 8 vertices and they are good enough.

Step 2 We are going for the Task 2 in Table 9.10. Select the polygon H1 at Z = 1300. Select Adv Edit->Build Holes and Vias from Selected Polygons. Select The Z of New Hole = 670. Select Add button to add it. Choose Clear the Hole and Remove Selected (see Figure 9.31). Select OK to build the hole. Save the file as: .\ie3d\practice\bga2.geo.

Step 3 Select polygons H2 and H3 at Z = 1300. Select Adv Edit->Build Holes and Vias from Selected Polygons. Add Z = 670 and 840 into the list box. Choose Clear the Hole and Remove Selected. Select OK to build the holes. Save the file as: .\ie3d\practice\bga3.geo.

Step 4 Select polygons H4 at Z = 1300. Select Adv Edit->Build Holes and Vias from Selected Polygons. Add Z = 840 into the list box. Choose Clear the Hole and Remove Selected. Select OK to build the holes. Save the file as: .\ie3d\practice\bga4.geo.

Step 5 We are going to do the Task 3. Click at Z = 590. Select Edit->Layer->Grow Metal Thickness on Layer. MGRID will prompt you for the parameters. Remember to choose Negative Z-Direction because we want the trace on this layer to span from Z = 550 to 590. Select OK and MGRID build the thickness for the layer. Save the file as: .\ie3d\practice\bga5.geo.

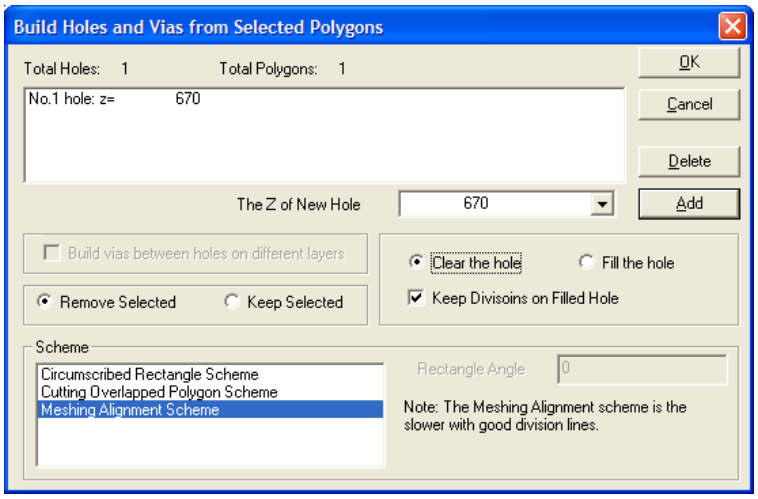


Figure 9.31 The Build Holes and Vias from Selected Polygons dialog for H1.

Step 6 Select Z = 670, 840 and 960 individually and perform the Edit->Layer->Grow Metal Thickness on Layer command. Please choose Positive Z-Direction for it. The end result is saved in .\ie3d\practice\bga6.geo with metal layers built for the M2, M3, M4 and M5. As a matter of fact, we may not need to build thickness for ground plane and power plane because they are big pieces of metal. The thickness may not affect it so much. If you want to save memory, you would not need to build thickness on them.

Step 7 Our next task (Task 4) is to build the solder balls. The basic skills in building wire bonds and solder balls are discussed in Appendix P. Basically, we need to know the locations of the solder balls. The locations of the solder balls are pre-measured and they are documented in Table 9.11.

Select Input->Create and Edit Vertices. Select Import and select the file: .\ie3d\samples\solder_balls.txt file. Select OK to accept the default without offsetting the locations. The locations of the 4 solder balls are imported. Select OK and you have the 4 vertices for the solder ball centers entered. You certainly can enter the vertices one by one by keyboard or mouse entry.

Table 9.11 The locations of solder balls.

No.	Ball 1	Ball 2	Ball 3	Ball4
(X, Y)-Location	(-340, 1560)	(-340, 2230)	(340, 2230)	(340, 1560)

Step 8 Select Entity->Solder Balls to bring up the dialog (see Figure 9.32). Change the Number of Segments in Phi = 8 from 6 (8 is consistent with the shape of the pads even the consistency is not required). Change the Radius = 300. Change the Vertex at Phi = 0 to Edge at Phi = 0. Change No Ports at Start Points to Ports at Start Points. We should have exactly what is shown in Figure 9.32 before we select OK. Select OK and MGRID will build the solder balls with the top view

and 3D view in Figure 9.33. You may not be able to see the shapes of the solder balls in the top view unless you focus the input to 3D Polygon layer. Also, you will see some ports are built for the solder balls. They are between the solder balls and the PCB ground at Z = 0. Save the file as .\ie3d\practice\bga7.geo.

Step 8 We will go for the Task 5 for the vias. You may notice that some polygons on Z = 1500 representing the shapes of via pads are automatically broken down by MGRID during the editing. It is possible the polygons for the vias are also broken down into multiple polygons. In such a case, you should select the polygons for the vias and merge them before you use them to build vias.

Select the V1 at Z = 1700. Select Adv Edit->Build Holes and Vias from Selected Polygons command. MGRID will prompt you for the parameters. Add Z = 590, 840, 880 and 960 into the list. Select Build Via Between Holes on Different Layers, Remove selected and Clear the Hole (see Figure 9.34). Select OK and we will have the via built (see Figure 9.34). For the 3D view in Figure 9.34, we have zoomed in the structure in Z-direction by a factor of 3. The command is in View->Transformation in the 3D View window. Save the file as .\ie3d\practice\bga8.geo.

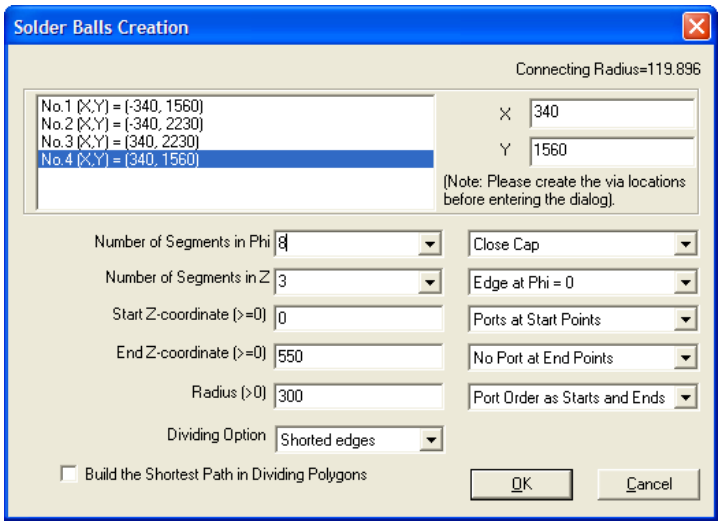


Figure 9.32 The Solder Ball dialog before selecting OK.

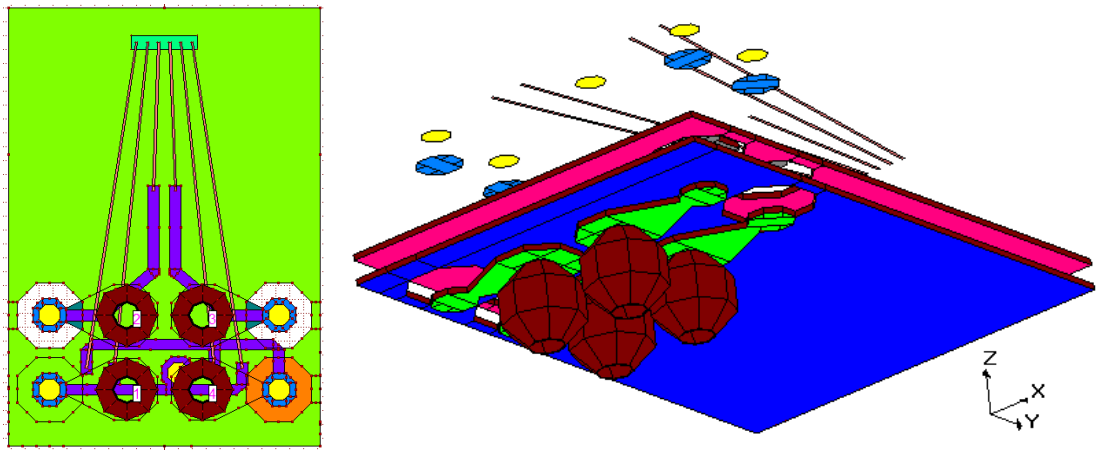


Figure 9.33 The top view and 3D view after the solder balls are built.

Step 9 Select V2 and V3 at Z = 1700. Build vias between Z = 590 and 960 with the same options in Figure 9.34. Select V4 at Z = 1700 and build a via between Z = 590, 670, 710 and 960. Select the V5 at Z = 1700 and build a via between Z = 880 and 960. Save the file as: .\ie3d\practice\bga9.geo.

Step 10 Our next task is to build the wire bonds. The wire bond locations are in Table 9.12. They can be easily measured and created from the structure. Detail in finding the vertices for the wire bond's vertex pairs is discussed in Appendix P. For this case, we just try to import them saved in a file: .\ie3d\samples\wire_bonds.txt.

Select Input->Create and Edit Vertices to import the 12 vertices representing the 6 wire bonds from wires_bonds.txt file.

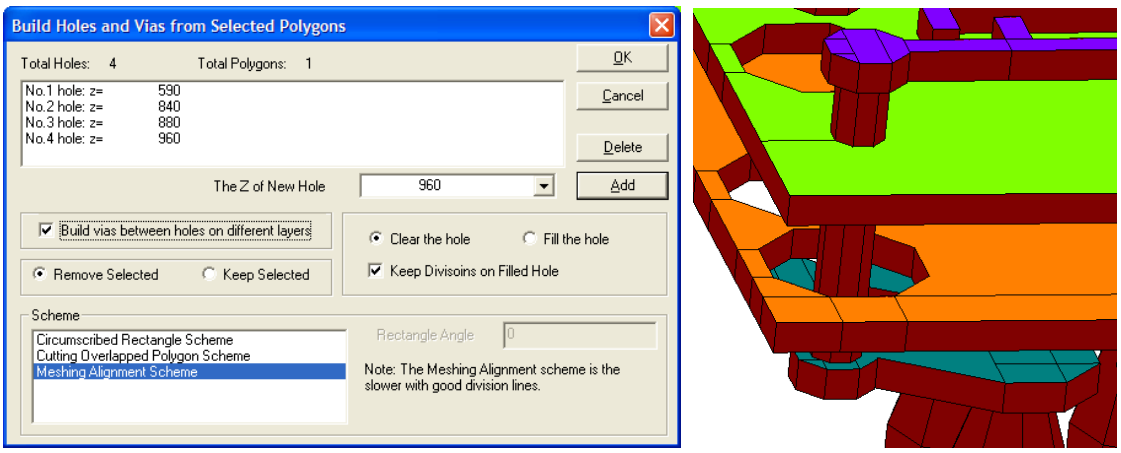


Figure 9.34 The Build Holes and Vias from Selected Polygons for V1 and the built via.

Step 11 Change the Radius to 10 (for diameter of 20). Make sure the Style is: 3-Segments (JEDEC). Select the Z-Extremum. Change its value to 1500. Select Update All Wire Bonds to change the Z-Extremum values of all the wire bonds. Select H-Segment. Change its value to 260. Select Update All Wire Bonds to change the H-Segment values for all the wire bonds. We will build ports at the ends of the wire bonds. Please select the End button. Change the Build Connection setting to Backward Directed Port (see figure 9.35). Select Update All Wire Bonds. We should have the dialog shown in Figure 9.36. Select OK. MGRID will take some seconds to build the wire bonds. At the ends of the wire bonds, it built 6 more ports. Please save the geometry as: .\ie3d\practice\bga10.geo.

Table 9.12 The wire bonds locations

No	Wire Bond	Start (X, Y, Z)	End (X, Y, Z)
0	W4a	(-700, 1750, 1000)	(-250, 4685, 1250)
1	W1a	(-450, 1750, 1000)	(-150, 4685, 1250)
2	W2	(-100, 3350, 1000)	(-50, 4685, 1250)
3	W3	(100, 3350, 1000)	(50, 4685, 1250)
4	W4b	(450, 1750, 1000)	(150, 4685, 1250)
5	W1b	(700, 1750, 1000)	(250, 4685, 1250)

Step 12 It seems to us the geometry is done. However, please don't forget to check it. We still have some polygons on Z = 1500 representing the shapes of the via pads, and some polygons on Z = 1900 representing the wire bonds. Please delete them and save the final geometry as: .\ie3d\practice\bga.geo.

For multi-layered structures, it may be hard to see the detail between layers when they are too close. You can choose View->Transformation in the 3D View to zoom it in the z-direction. Shown in Figure 9.38 is a 3D view zoomed in z-direction by a factor of 5.

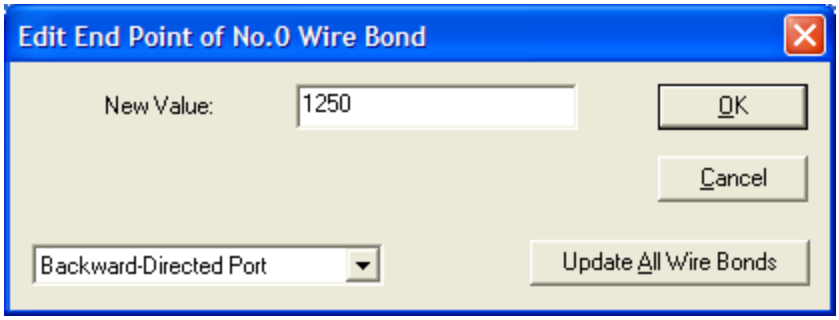


Figure 9.35 The End dialog for the wire bonds.

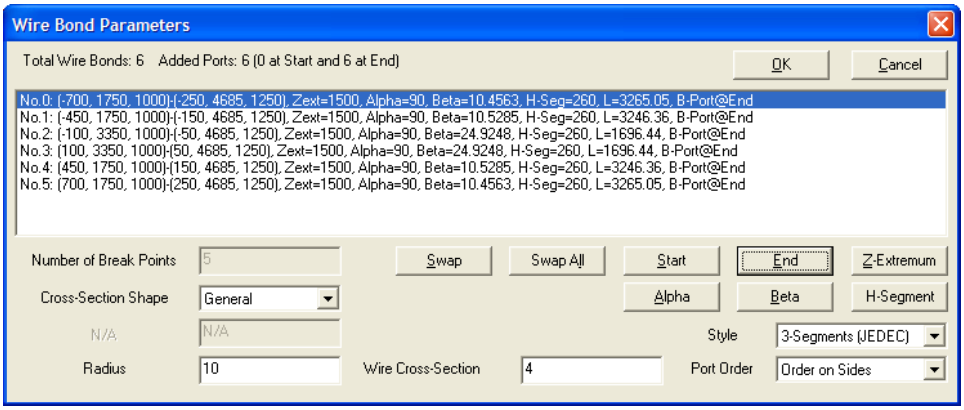


Figure 9.36 The dialog with all the parameters defined for the wire bonds.

Step 13 Please simulate the structure from 0.2 to 50 GHz for total 250 frequency points. It should take about 6 minutes per frequency point for a modern PC. However, it will need about 45 frequency points for AIF to converge due to the many resonances in the range. The frequency responses are displayed in Figure 9.37. You may wonder why the S(2,2) does not start from a very low value. Instead, it starts from about -13.4 dB.

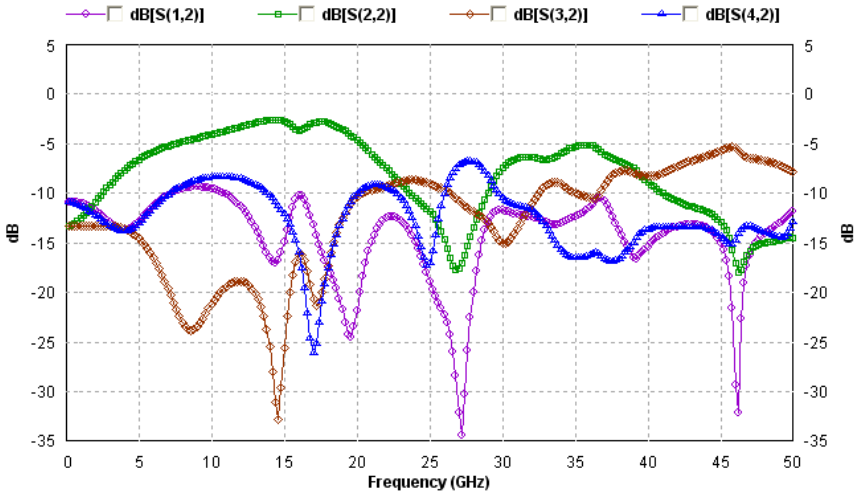


Figure 9.37 The s-parameters of the 10-port structure.

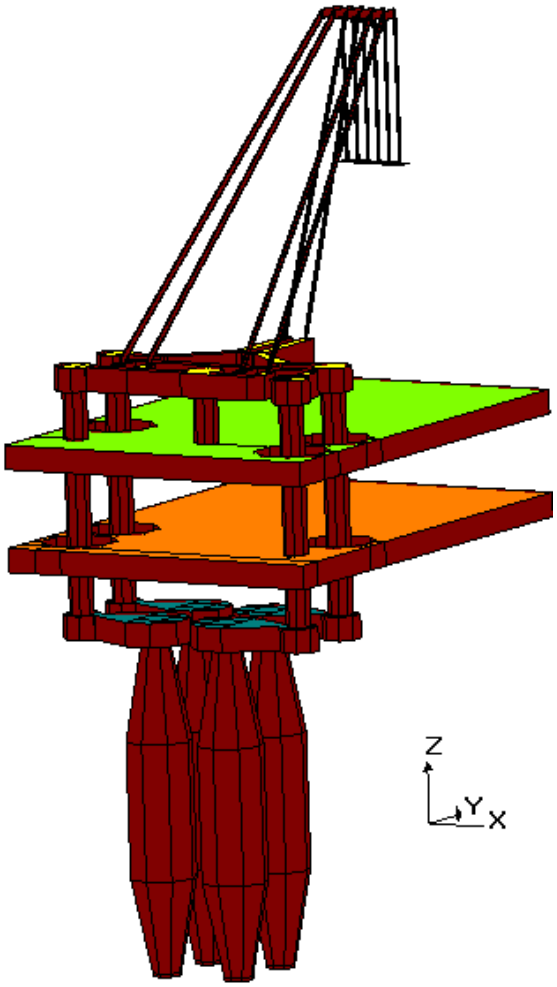


Figure 9.38 The 3D view with z-direction zoomed by a factor of 5.

The BGA model we built in this section is very complicated. It involves many steps and many advanced commands. The process is in fact automated in AGIF. The same structure can be built in one shot on AGIF. Interested users can read the AGIF User's Manual for the process.

Section 9.19 Post-Processing and Time Transient Analysis on the BGA Structure

We have simulated the structure as a 10-port structure. Again, we can not use the Process->LC-Equivalent command on MODUA to find the equivalent circuit of the structure, due to the fact that the die ground is still floating. Because of the floating die ground, the S(2,2) starts from about -13.4 dB instead of a very low value. To get the equivalent circuit of the structure, we need to short circuit ports 1, 6 and 10. We also need to combine the ports 5 and 9 together for one port (see Figure 9.39). The MODUA setup is saved in .\ie3d\samples\bga1a.dsg file. The resulting 6-port s-parameters represent the network for 3 current paths: (1) The positive path from port 1 to port 4 (or terminal 2 to terminal 7 of the BGA module in Figure 9.39); (2) The negative path from port 2 to port 5 (or terminal 3 to terminal 8 of the BGA module); (3) The power path from port 3 to port 6 (or terminal 4 to terminal 5 and 9 of the BGA module). The resulting 6-port s-parameters are saved in .\ie3d\samples\bga1a.sp shown in Figure 9.40. As you can see, the S(4,1) starts from almost 0 dB and declines with frequency. S(1,1) increases with frequency. There are many resonances beyond 5 GHz.

We can use MODUA to extract the RLC-equivalent circuit of the final 6-port circuit. The results are documented in Table 9.13. Certainly, the equivalent circuit should only be used at low frequency below 1 GHz.

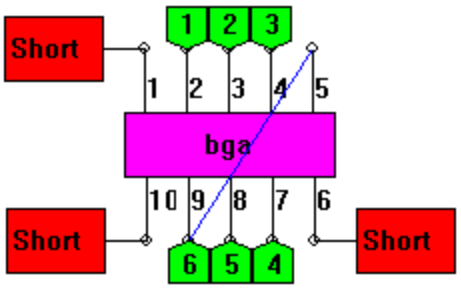


Figure 9.39 The MODUA setup to find the s-parameters for 3 current paths.

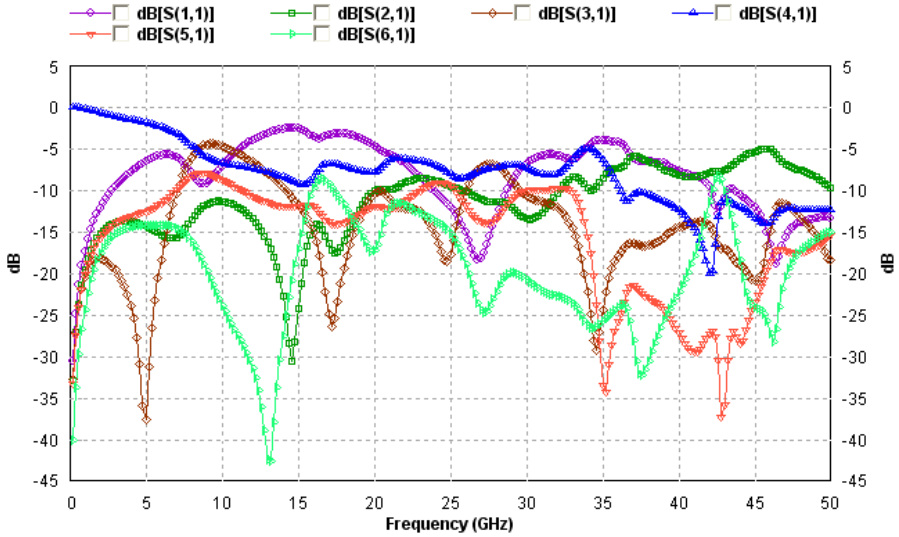


Figure 9.40 The s-parameters of the 6-port s-parameters for the 3 current paths.

Table 9.13 The values of the RLC equivalent circuits at different frequency points.

Freq (GHz)	Series L (nH)			Series R (Ohms)			Shunt C (pF)			Shunt R (Ohms)		
0.2	3.539	1.782	1.324	0.2056	N/A	N/A	0.2506	0.0169	0.2120	1.19e+6	4.11e+7	8.73e+5
	1.782	3.817	1.636	N/A	0.2113	N/A	0.0169	0.2340	0.228	4.11e+7	1.19e+6	8.78e+5
	1.324	1.636	3.125	N/A	N/A	0.1919	0.2120	0.228	3.514	8.73e+5	8.78e+5	4.90e+4
0.5	3.500	1.795	1.368	0.3356	N/A	N/A	0.2505	0.0167	0.2129	4.85e+5	1.48e+7	3.39e+5
	1.795	3.784	1.689	N/A	0.3473	N/A	0.0167	0.2350	0.2274	1.48e+7	4.70e+5	3.50e+5
	1.368	1.689	2.961	N/A	N/A	0.2895	0.2129	0.2274	3.575	3.39e+5	3.50e+5	1.75e+4
1.0	3.604	1.985	1.732	0.5472	N/A	N/A	0.2498	0.0160	0.2162	2.58e+5	5.08e+6	1.56e+5
	1.985	3.940	2.117	N/A	0.585	N/A	0.0160	0.2390	0.2261	5.08e+6	2.19e+5	1.76e+5
	1.732	2.117	2.587	N/A	N/A	0.3178	0.2162	0.2261	3.824	1.56e+5	1.76e+5	6.44e+3

Using MDSPICE, we can perform robust time transient analysis on the structure with different kinds of terminations and excitations. This is a differential structure. Assume we want to put a clock signal of rising time 24 ps (19.2 ps from 10% to 90%) at port 1 and the out of phase signal at port 2. The internal resistors of port 1 and port 2 are 100-ohms. Port 3 is the power supply and we can assume the termination at it is 0-ohms. Port 4 and port 5 are the loads at the die. We assume either one is terminated with 100-ohms.

Port 6 is the power supply at the die. We assume the termination is 3-ohms. Figure 9.41 shows the time-transient responses of the differential paths with the above excitations and terminations. Figure 9.42 shows the voltage fluctuation on the ground path at the die side due to the assumed 3-ohm termination between the power path and the ground path. Saved in .\ie3d\samples\bga1b is the MODUA design to convert the 6-port s-parameters into 4-port s-parameters (see Figure 9.43). The 4-port s-parameters can be displayed and converted into 2-port differential s-parameters on MODUA using the Process->Differential Conversion command. The differential s-parameters are shown in Figure 9.44. By default, the s-parameters are normalized to 50-ohms for each port. However, differential port is normally 100-ohms port. We can use Control->Terminating Impedance in MODUA to convert the 50-ohm normalized s-parameters into 100-ohms normalized s-parameters (see Figure 9.44).

IE3D combined with MDSPICE offers a complete tool to cover frequency domain and time domain analysis for signal integrity applications as well as other high frequency applications. The purpose of this chapter is just trying to demonstrate how we approach the problem. IE3D can be used to model much more sophisticated structures. You can try to explore it with the skills you have learned. We will end this chapter here.

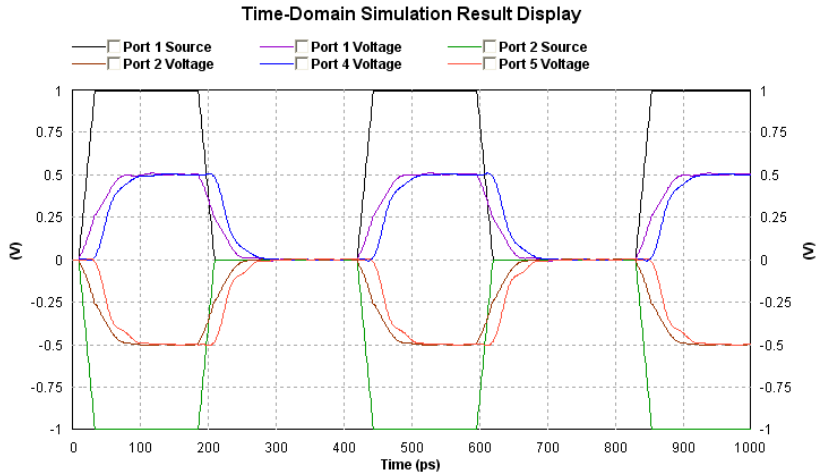


Figure 9.41 The time transient analysis with a differential clock signal applied to the input.

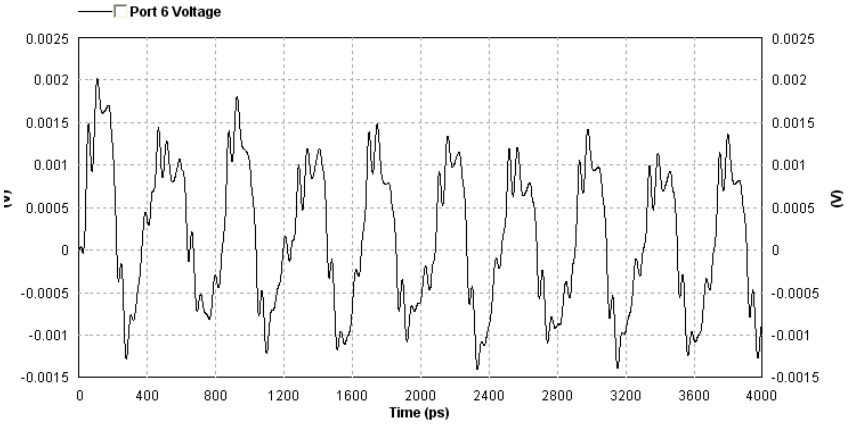


Figure 9.42 The voltage fluctuation at the power path at the die side.

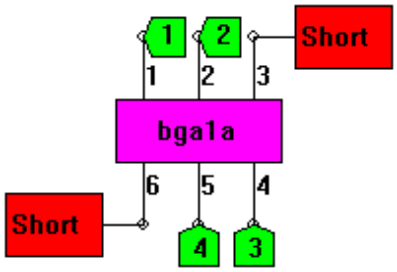


Figure 9.43 MODUA setup to convert the 6-port s-parameters into 4-port s-parameters.

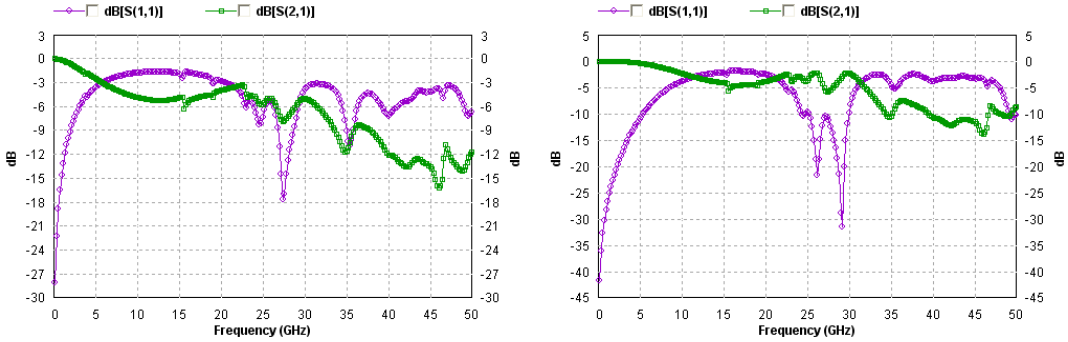


Figure 9.44 The 50-ohm (left) and 100-ohms normalized (right) s-parameters of the differential pair.

Chapter 10 Simulation of RF and Microwave Filters

Planar filters are important components in RF, wireless, and microwave circuits and systems. IE3D is a general-purpose EM simulator. It certainly can be applied efficiently to the analysis and design of different kinds of planar and 3D filters.

A filter normally consists of multiple closely coupled resonators. If we adjust the resonant frequencies of the resonators and the coupling between the resonators properly, we can achieve high performance filters. To design a filter, we can certainly start from the IE3D directly. However, no matter how fast it is, IE3D is still a full-wave EM simulator. It would be nice that we can start from some initial geometry. For this reason, we have developed the FilterSyn module for the Ie3dLibrary.

Section 10.1 Ie3dLibrary, the Parameterized and Object-Oriented 2nd Interface for IE3D

Ie3dLibrary is the 2nd interface developed for IE3D. We have been using the MGRID, the default interface for the IE3D in the previous chapters. As you can see, MGRID is a very powerful graphic interface. It allows you to construct and edit complicated structures with all details. The fundamental elements of MGRID are polygons and vertices. MGRID allows you to control the shape, size, and orientation of polygons and the locations of vertices. We can construct almost any kind of shapes easily. However, polygons and vertices are low level objects. Let's consider a filter. It may consist of many different polygons. There are relations between different polygons of a filter. On the MGRID, we can change the shape and size of a polygon easily. However, if some dimensions of an object are determined by many vertices or polygons, it may be tedious to change them by changing each polygon one by one. For this reason, we introduced the Ie3dLibrary. The basic elements of Ie3dLibrary are the objects of pre-defined shapes. The dimensions of the objects can be changed any time after it is created as a building block for a large circuit. For example, we can divide a circuit as objects of T-junctions, cross-junctions, Y-junctions and straight lines, etc. on the Ie3dLibrary. We can connect different objects using some default snapping scheme. After the circuit is built, we can still change each element's dimensions. When we change the dimensions of an object, Ie3dLibrary will preserve the connectivity between the objects. We can also define tuning variables with a dimension of an object described as a formula referencing the tuning variables. When we change the tuning variables, the dimension defined by the formula is updated automatically. Building a large structure with pre-defined object shapes becomes much simplified on the Ie3dLibrary.

IE3DLIBRAY is much improved in version 12 and version 14. Boolean objects and operations, void objects are introduced and they extend the geometry modeling capability of IE3DLIBRARY to a next level. We have also implemented the FastEM Design Kit discussed in Chapter 5 into IE3DLIBRARY. The combinations of equation-based geometry modeling, Boolean objects and operations and FastEM Tuning and Optimization make IE3D and IE3DLIBRARY much more capable than before. A separate document on using IE3DIBRARY is provided. Interested users please read the document for more information in using this great layout and parameterization tool.

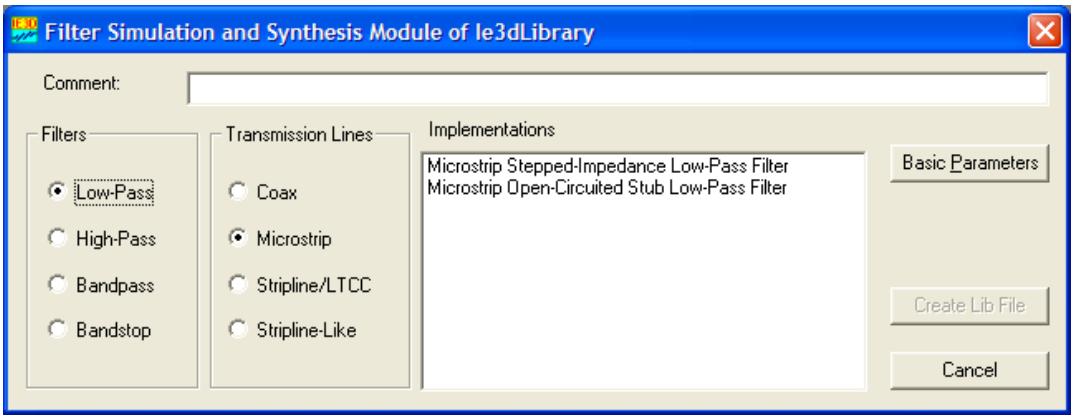
Section 10.2 Using FilterSyn Module for Ie3dLibrary for Initial Design

The FilterSyn module is basically a menu item on the Ie3dLibrary. The FilterSyn is an add-on to IE3D. We will demonstrate how to use the FilterSyn to get initial design in the examples in this chapter. Those users who did not purchase the FilterSyn will not be able to use the FilterSyn module to create the initial design. You can get the created geometry files from FilterSyn from the .\ie3d\samples directory. If you are interested in the license of FilterSyn module, please contact us.

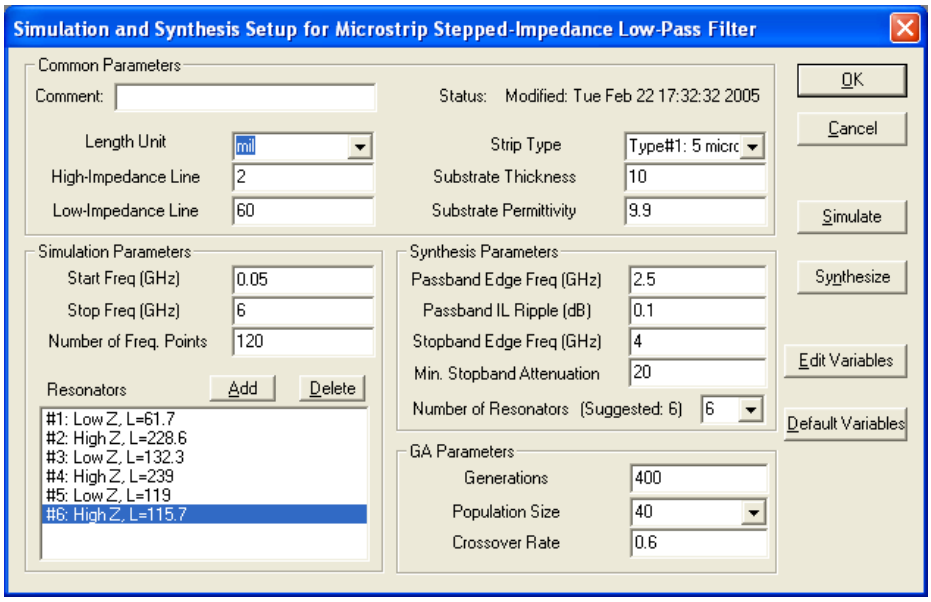
The FilterSyn deals with low-pass, bandpass, and bandstop filters. It covers coaxial, microstrip, stripline/LTCC, and stripline-like structures. It contains analytical models of the following filters: stepped impedance low-pass filters, open-circuited stub low-pass and bandstop filters, parallel-coupled bandpass filters, gap-coupled bandpass filters, and combline and interdigital bandpass filters. There are 4 types of

low-pass filters, 20 types of band-pass filters, and one type of band-stop filters. Each filter type can be used as an object of the Ie3dLibrary. We will demonstrate how to use it in the following.

Step 1 Run Ie3dLibrary. An empty Ie3dLibrary main window pops up. Select **File->New** command. Ie3dLibrary will prompt you for the New Project Wizard. Select Start a New FilterSyn Project, and select OK. The dialog "Filter Simulation and Synthesis Module for Ie3dLibrary" comes up.



(a). The FilterSyn dialog for Ie3dLibrary.



(b). The Simulation and Synthesis Setup dialog for the low-pass filter.

Figure 10.1 The FilterSyn dialog and the Low Pass Filter dialog.

Step 2 Check Low-Pass in Filters group and check Microstrip in Transmission Lines group (see Figure 10.1a). We will get the picture shown in Figure 10.1. There are 2 types of Low-Pass Microstrip Filters available.

In IE3D 14, you are able to access the Basic Parameters in the dialog. However, the substrates and some other parameters may not be useful for the FilterSyn synthesis because each filter type

may have its own substrate stack up. Also, the Meshing Frequency (Fmax) will not be used and it might be changed after a FilterSyn procedure due to some other considerations.

Step 3 Please select **Microstrip Stepped-Impedance Low-Pass Filter** on the list.

If you do not have the FilterSyn license, Ie3dLIbrary will give you a warning. Please skip this section and go to Section 3.

If you have the FilterSyn license, the setup dialog for the filter type comes up. The parameters for the object are divided into 3 groups: (1) Common Parameters, (2) Simulation Parameters, and (3) Synthesis Parameters. If we are simulating a microstrip stepped-impedance low-pass filter with pre-defined parameters, we can enter the corresponding parameters in the Common Parameters and Simulation Parameters groups. If we are synthesizing a filter with specified performance requirements, we should enter the parameters in the Common Parameters and Synthesis Parameters group. In our example, we are going to simulate the filter with the default parameters.

Step 4 Please make sure the length unit is in: mil (see Figure 10.1b). Click the Simulate button in the dialog. FilterSyn will finish the simulation in no time since it is based upon analytical formulas. Then, it will prompt you for the s-parameter file.

Step 5 Please type: .\ie3d\practice\output\lpfa.sp and select Save button. The resulting s-parameters are saved into the file and MODUA is invoked to display the s-parameters (see Figure 10.5).

You may wonder how accurate the results are. You may want to check the results using IE3D. We are going to create the IE3D model for it. We do not need to enter the polygons based upon the dimensions. We can create them automatically on the Ie3dLibrary.

Step 6 Select OK to close the Microstrip Stepped-Impedance Low-Pass Filter dialog. It will get back to the Filter Simulation and Synthesis Module for Ie3dLibrary dialog (see Figure 10.1a).

Step 7 While the Microstrip Stepped-Impedance Low-Pass Filter is still being selected in Figure 10.1a; please select the Create Lib File button. The response of IE3DLibrary will be different depending upon the real situation.

If you have created some other objects on IE3DLibrary, IE3DLibrary will create the filter object. However, it will not take the substrate parameters of the filter object. It will still use the Basic Parameters created before. Only the shape of the object is kept when you add it onto an IE3DLibrary object as an additional object. The performance of the filter may not be the same as in an IE3DLibrary project as it is inside the FilterSyn unless you already make sure they have the substrate substrates.

In case you have not created any objects on the IE3DLibrary project and this is the fresh started IE3DLibrary project from FilterSyn, IE3DLibrary will automatically created the Basic Parameters from the FilterSyn object and prompt you the Basic Parameters dialog (see Figure 10.2). The substrate configuration and the Meshing Frequency (Fmax) are automatically chosen from the filter's specifications.

Step 8 Select OK on the Basic Parameters dialog. The complete low-pass filter becomes an object of the Ie3dLibrary. It is following the mouse cursor for relocation.

Step 9 Click the mouse button somewhere in the Ie3dLibrary window. The object is dropped onto the window. However, it is still being selected with the corners highlighted with Red Colors (see Figure 10.3). You will see the filter object is checked on the Work Space, indicating the object is

still being selected. Starting from IE3D 14, you can select objects on the Work Space or by clicking at them on the layout window. Also, when the mouse is pointing at an object in the layout window or in the object list in the Work Space, the corners of the object will be highlighted in both locations. You can also select or do editing on the highlighted object by clicking at the right mouse button to bring a pop-up window.

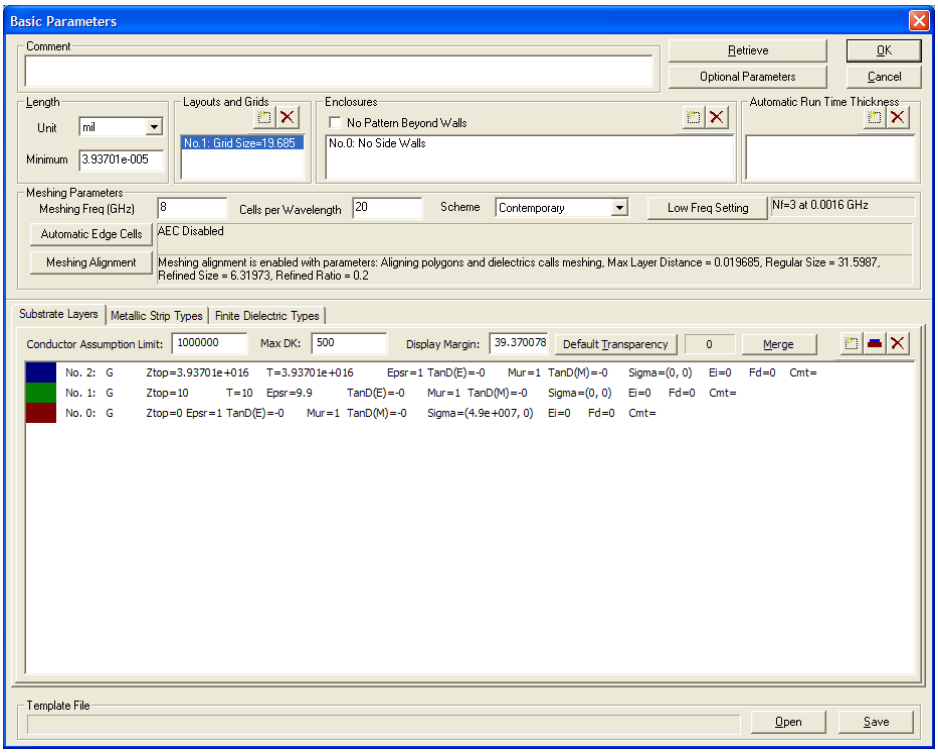


Figure 10.2 The Basic Parameters dialog when you select Create Lib File.

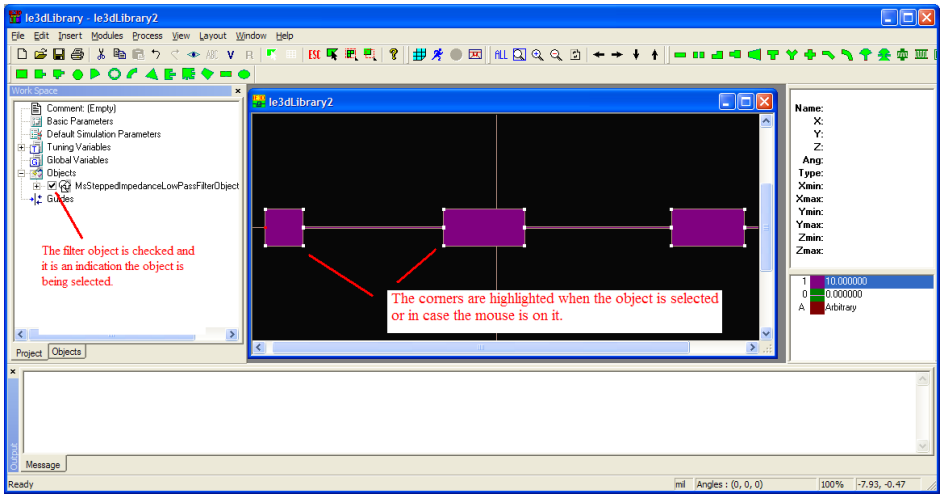


Figure 10.3 LPF1 dropped onto Ie3dLibrary.

Step 10 Click at an empty spot to de-select the object. You can also un-check the object from the Work Space to de-select it. Save it as: .\ie3d\practice\lpf1.ie3. The extension ".ie3" is for Ie3dLibrary.

You will see two red-dots on the two ends of the filter. They are the possible connection points for the object. You can connect other objects to the edges with red-dots only.

Section 10.3 Using LineGauge for Transmission Line Parameters

If you do not have the FilterSyn license and you have skipped the last section, please open the file: .\ie3d\samples\lpf1.ie3 on Ie3dLibrary. Save it as: .\ie3d\practice\lpf1.ie3. We are going to discuss the other features here.

We are going to add a 50-ohm line section to each end of the filter. What is the width of a 50-ohm microstrip line with a 10-mil-thick substrate of dielectric constant 9.9? It does not matter whether you know it, because you can use the **LineGauge Transmission Line Synthesis Tool** to do the synthesis. The basic edition of the LineGauge is free of charge. You can analyze and synthesize simple transmission lines such as microstrip, stripline, and coplanar waveguide using the basic edition of the LineGauge.

Step 1 Run LineGauge. Select Microstrip in the listbox. Select Length Unit = mil. Enter the Frequency = 3 GHz, Relative Permittivity = 9.9, Substrate Height h = 10 mils, Strip Thickness = 0.1. Then, enter Key Electrical Parameters as: Zc = 50 and Electrical Length = 30 degrees. Select the button **Electrical->Physical Parameters**. LineGauge will calculate the physical parameters based upon the common parameters and the electrical parameters and yields the Strip Width =9.49 mils and the Length = 128 mils (see Figure 10.4). The 30-degrees (or 128 mils) value is arbitrarily chosen. We will add a 50-mil-long section of transmission line to each end.

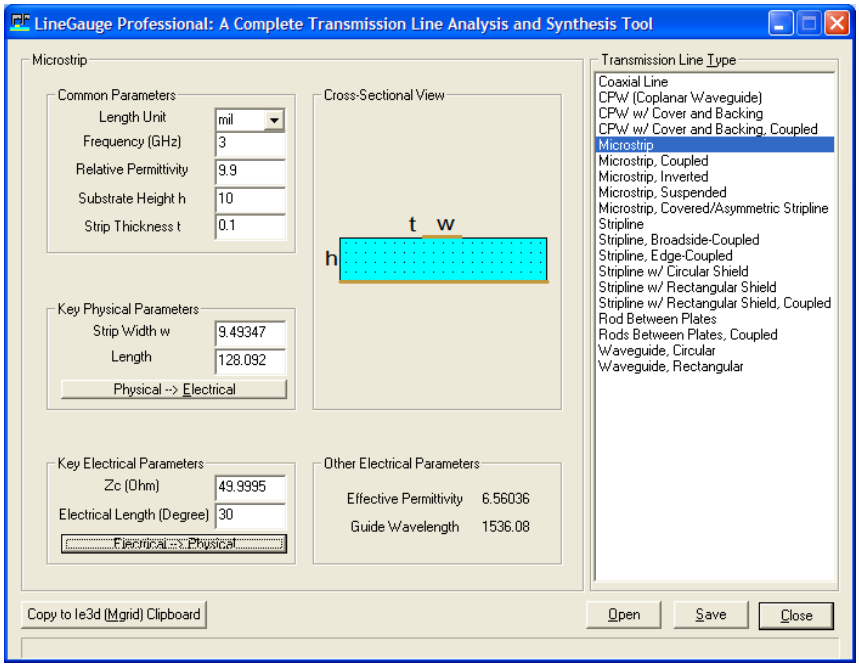


Figure 10.4 The LineGauge window.

Section 10.4 Connecting Objects on Ie3dLibrary

Step 1 On Ie3dLibrary, select **Insert->Strip Objects**. You will see a sub-menu with more than 10 menu items comes up. Each menu item corresponds to a parameterized object of pre-defined shape.

Please select **Insert->Strip Objects->Straight Strip** command. The dialog for Straight Strip object comes up (see Figure 10.5). Enter Length = 50 mils and the Width = 9.49 mils. Select the

Z = 10 mils. There is a parameter for Strip Type. It defines the metallic type of the object will use. We can use the default one or the only one.

Step 2 Select OK. The Straight Strip object is following the mouse cursor. If the Ie3dLibrary window is not at the right size, the user can use the "-" and "+" to zoom the objects on the window. When there is enough space on the left side of the filter, click at the left end of the filter. The Straight Strip object will be connected to the LPF object at the left end. The Straight Strip will also automatically adjust its orientation for the connection. The Straight Strip object is still being selected.

Step 4 Click anywhere to de-select the Straight Strip object. We are going to attach another feed line to the right side of the filter.

Step 5 Select **Insert->Strip Objects->Straight Strip** command. The dialog for Straight Strip object comes up. The parameters of the last Straight Strip object are used as the default. Please select OK. The Straight Strip object is following the mouse cursor. Click the cursor at the right end of the filter. The Straight Strip object is connected to the filter. Click anywhere to de-select the Straight Strip object. We will get the picture shown in Figure 10.6.

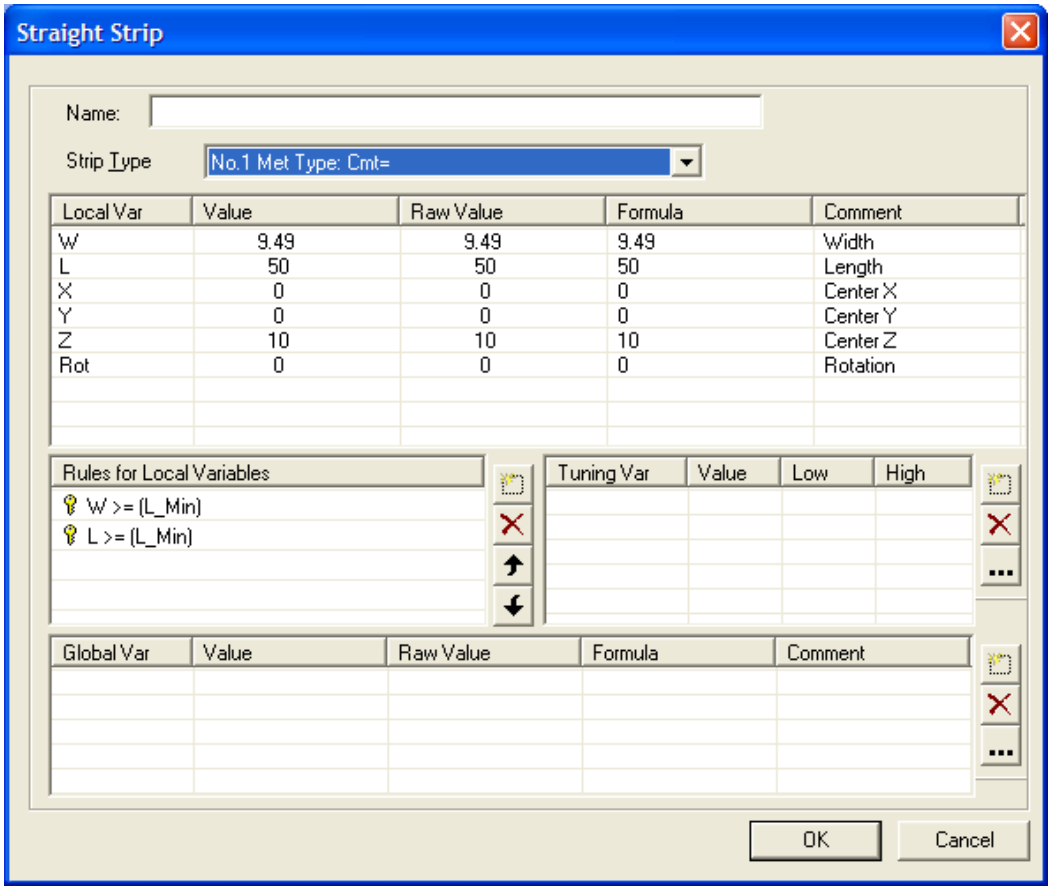


Figure 10.5 The Straight Strip object.

Step 6 Select **Insert->Dependent Objects->Extension Port** command. Please select or enter the Z of Terminal = 10. Scheme = Advanced Extension, Cells on De-embedding Arm = 5. Port Property 1. Display Size = 1 (see Figure 10.7). Please select OK to continue. The port object is following the mouse cursor.

Figure 10.6 The lowpass filter with two sections of 50-ohms lines connected.

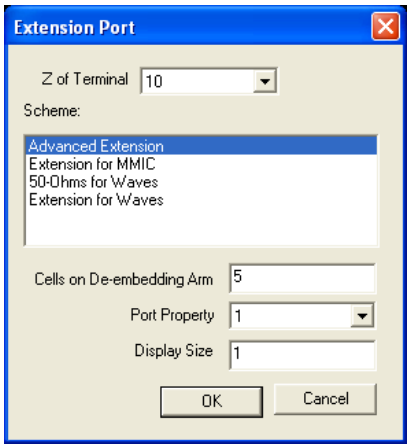


Figure 10.7 The Extension Port dialog.

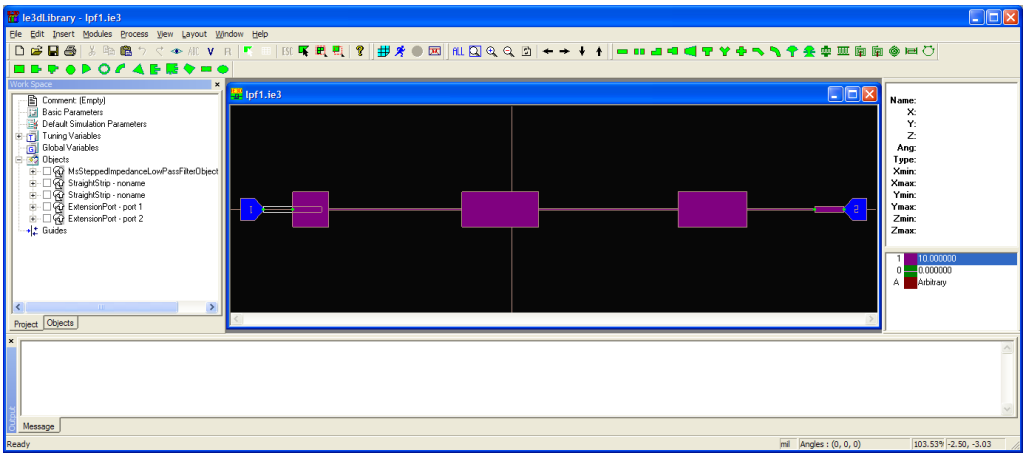


Figure 10.8 The filter with feed lines and ports defined.

Step 7 Please move the mouse to the left terminal of the left Straight Strip object, and click. The extension Port object is connected to the left terminal. The port object size is arbitrary chosen. In case it is too small or too big, please select it and change it to appropriate size. Its size does not affect the simulation results.

Step 8 Please select **Insert->Dependent Objects->Extension Port** command again. The parameters of the last extension port are used as default. Please select OK to continue. Please click at the right edge of the right Straight Strip object to define the port 2 on it. Please click anywhere to de-select the port 2 object. You will get the picture shown in Figure 10.8.

Step 9 Please save the file as: .\ie3d\practice\lpf2.ie3. The geometry for the filter should be ready for simulation.

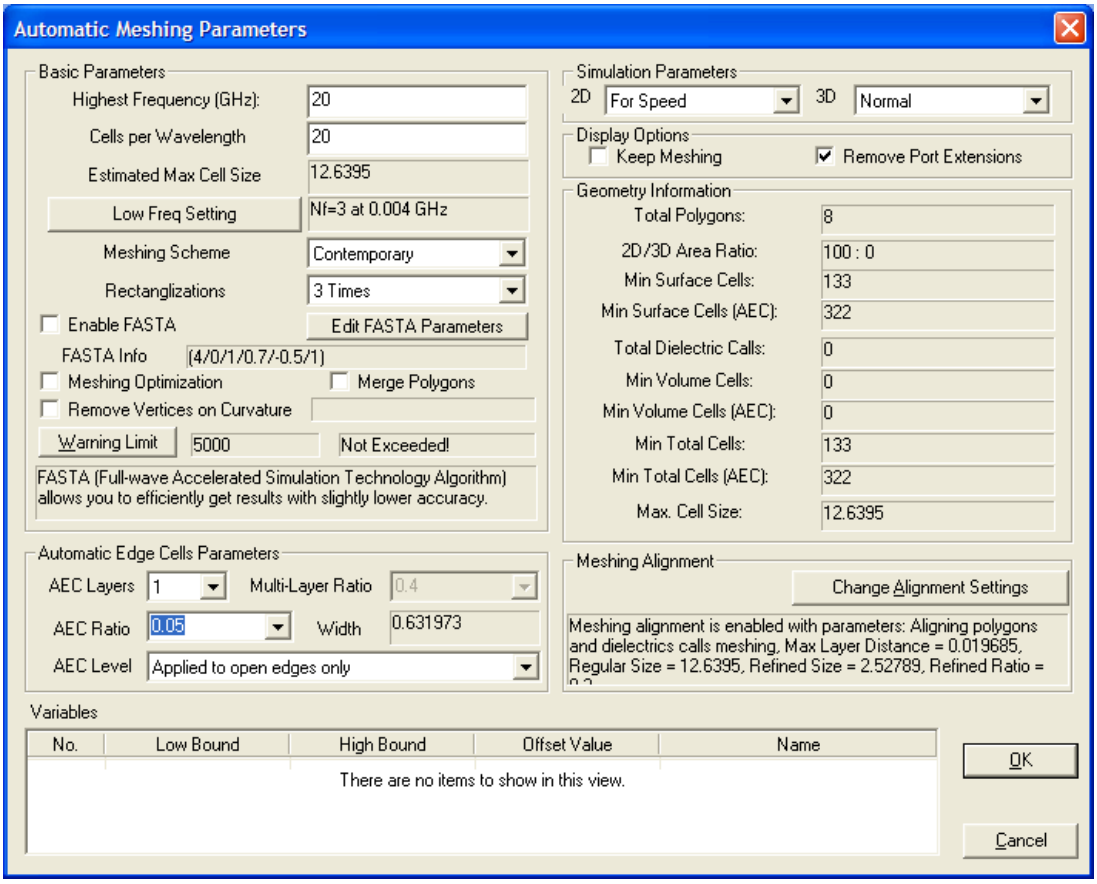


Figure 10.9 The Automatic Meshing Parameters dialog.

Section 10.5 Meshing and Simulation on Ie3dLibrary

We are able to mesh and simulate the circuit consisting of objects on the Ie3dLibrary. We can also export it to the MGRID as polygons and it will be ready for simulation if the objects are ready for simulation on the Ie3dLibrary at the time of exporting.

Step 1 Please select **Process->Display Meshing** on Ie3dLibrary. Enter the meshing parameters as shown in Figure 10.9. Remember to change Meshing Frequency to 20 GHz and enable the AEC Ratio = 0.05 (width = 0.6319 mils) for better meshing. In practical use, you should try to adjust the parameters for good meshing. Select OK to continue. IE3DLibrary will mesh the structure and prompt you. Select Continue. IE3DLibrary will open the Meshed View window to display the meshed structure. This process is just for you to check the meshing only. The meshing will not be the final one used in the simulation.

Step 2 Please select **Process->Simulate** command. The Simulation Setup dialog comes up. Select Capture button to capture the frequency points from .\ie3d\practice\lpfa.sp file (or .\ie3d\samples\lpfa.sp file). Select OK to simulate the filter. It may take tens of seconds because it is full-wave EM simulation on IE3D. Then, the s-parameter processing dialog corresponding to Process->S-Parameters and Lumped Equivalent Model command comes up (see Figure 10.10).

Step 3 Please change the Short Name to"IE3D". Add the s-parameters from FilterSyn in file ".\ie3d\practice\output\lpfa.sp" into the list and define its short name as "FilterSyn". Select Add Graph to define a graph to compare the results (see Figure 10.10).

IE3D predicts the maximum return loss is –17.5 dB and the 3dB bandwidth is about 2.75 GHz, while FilterSyn predicts the maximum return loss is –16.5 dB and the 3dB bandwidth is about 2.9 GHz. We will expect the IE3D should yield very accurate results. The results of the FilterSyn are good for this example. FilterSyn is based upon analytical formulas. We cannot expect it to yield perfect results. We can always use the synthesized filters from FilterSyn as the starting point for further full-wave EM optimizations on IE3D.

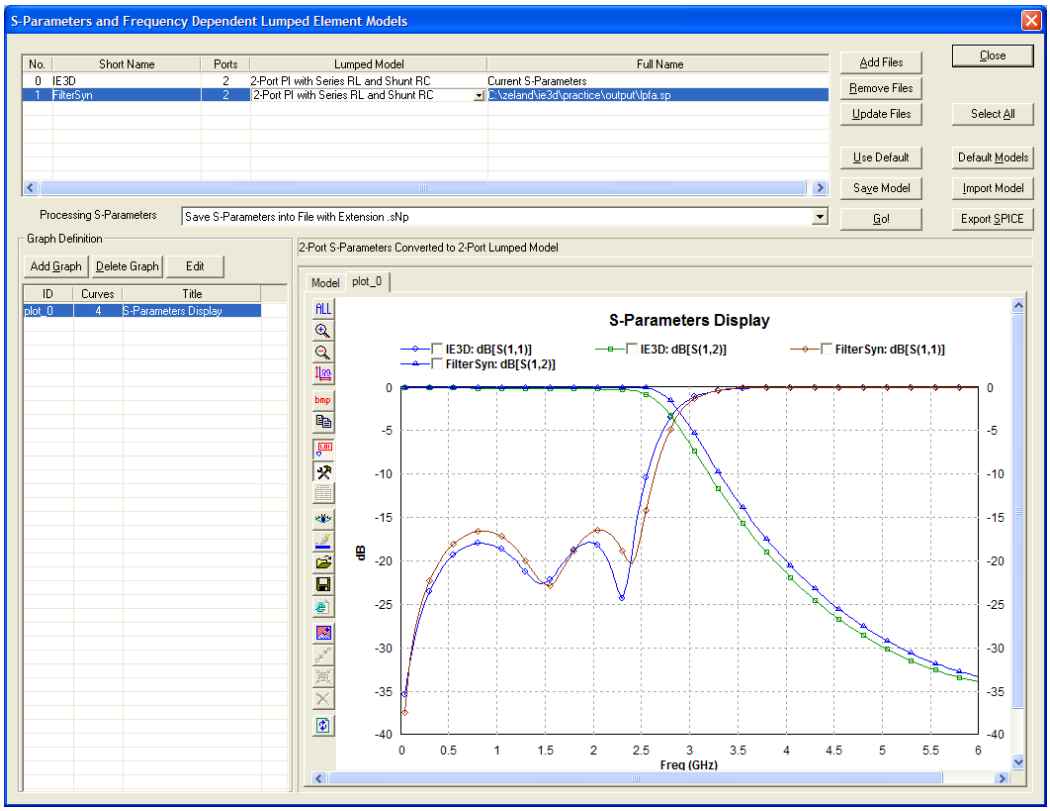


Figure 10.10 The comparison of IE3D and FilterSyn results on the LPF.

Section 10.6 Importing and Exporting Between Ie3dLibrary and MGRID

As you can see, MGRID is based upon polygons and vertices. IE3DLibrary is based upon parameterized objects of pre-defined shapes. They have their own advantages for different purposes. They are also interchangeable. We can import MGRID file (.geo) as an object into IE3DLibrary by selecting **File->Import .GEO File** command on the IE3DLibrary. Only the polygon parameters are used on the IE3DLibrary. The substrates and other parameters in the .geo file will be abandoned. We can also export a structure from IE3DLibrary to MGRID.

Step 1 While the lpf2.ie3 is still opened, please select **File->Save .GEO File** command. Please enter the file name as: lpf2a.geo. Select OK to continue.

The objects in lpf2.ie3 are converted into polygons and saved into lpf2a.geo file. MGRID is invoked automatically with the lpf2a.geo opened.

Step 2 You can select Process->Simulate command on MGRID to simulate the lpf2a.geo and you should get identical results to the lpf2.sp.

Section 10.7 Simulation of Bandpass Filters

We now try a bandpass filter. In this example, we will demonstrate the importance of the sidewalls and enclosures.

Step 1 Run IE3DLibrary again. Select Modules->Filters command. The FilterSyn dialog comes up. Select Bandpass Filter. Then, select Parallel-Coupled Microstrip BPF in the list. If you do not have the FilterSyn license, please skip Steps 2 to 4 and go to Step 5. If you have the FilterSyn license, FilterSyn will come up with the default bandpass filter dialog for you.

Step 2 Select Simulate button to simulate the filter from 10.5 to 13.5 GHz with 121 frequency points. FilterSyn will prompt you to save the s-parameters. Save it as .\ie3d\practice\output\bpfa.sp. MODUA is invoked to display the results. We will delay the discussion on the results later.

Step 3 Select OK on Ie3dLibrary. Ie3dLibrary will close the Parallel-Coupled Microstrip BPF dialog and it comes back to the FilterSyn main dialog. The item in the listbox is still being selected.

Step 4 Select Create LIB File button. The BPF object is following the cursor. Click it anywhere to drop it. Click an empty spot to de-select it. We will get the picture in Figure 10.11.

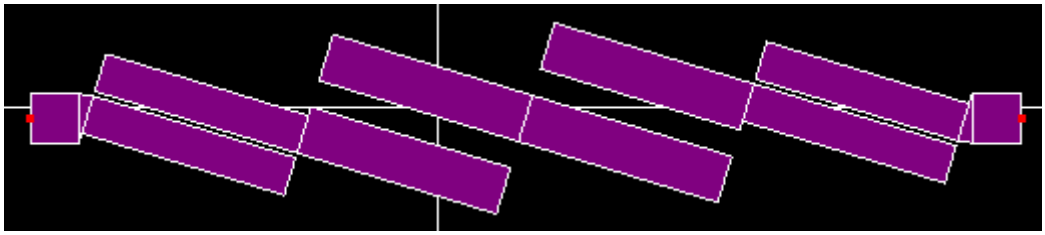


Figure 10.11 The BPF object on Ie3dLibrary.

Step 5 Select Insert->Dependent Objects->Extension Port command. The Extension Port dialog comes up. Please select Z of Terminal = 15, Scheme = Advanced Extension, Cells on De-embedding Arm = 5. Select OK to continue. Please click at the left terminal of the BPF filter to define the port 1 on it (see Figure 10.12).

Step 6 Repeat Step 5 to define the port 2 on the right terminal of the BPF filter. Please save the structure as: .\ie3d\practice\bpf.geo. We will get the structure shown in Figure 10.12.

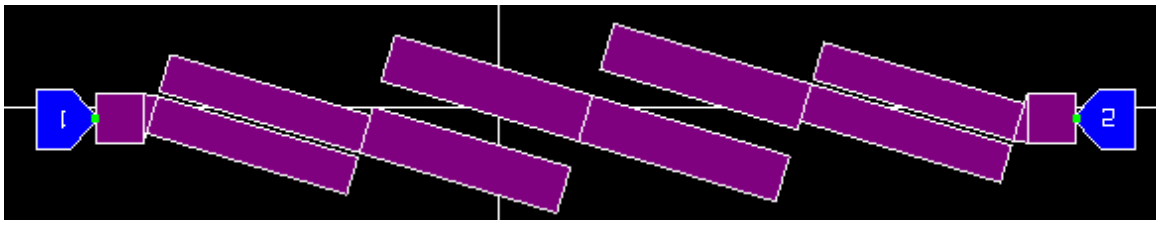


Figure 10.12 The BPF filter with 2 extension ports on it.

Step 7 The structure is ready for simulation. We can simulate it on either the Ie3dLibrary or the MGRID. We will do it on MGRID. Please Select **File->Save .GEO File** command on Ie3dLibrary and save the geometry as: .\ie3d\practice\bpf1a.geo. MGRID will be invoked with bpf1.geo opened. If you skip to this step from Step 1, please open the file: .\ie3d\samples\bpf1.geo and save it into: .\ie3d\practice\bpf1a.geo.

Step 8 Please select **Process->Simulate** command on MGRID. We will simulate the structure from 10.5 to 13.5 GHz with 121 frequency points. Please enable AEC Ratio = 0.1 (width = 2.73643 mils). Please also enable AIF. Select OK to continue. IE3D will finish the simulations in less than 20 seconds. The IE3D results and the FilterSyn results are shown in Figure 10.13. As you can see, the IE3D and FilterSyn agree very well at the high frequency end. At the low frequency end, there some slight difference. In the passband, IE3D predicts more loss than the FilterSyn. FilterSyn predicts a quite flat response over the passband. Where does the loss come in the passband predicted by the IE3D? It is basically due to the relatively thick substrate. A band-pass filter consists of a number of resonators. The default mode of the IE3D is for open structures. With thick substrate, the resonators are acting more like an antenna. Anyway, the loss in the IE3D predicted passband is due to the radiation loss. To remove the radiation, we can put the filter into an enclosure. How can we model the effects of the enclosure? We can do it using the option for boxed Green's functions on the IE3D.

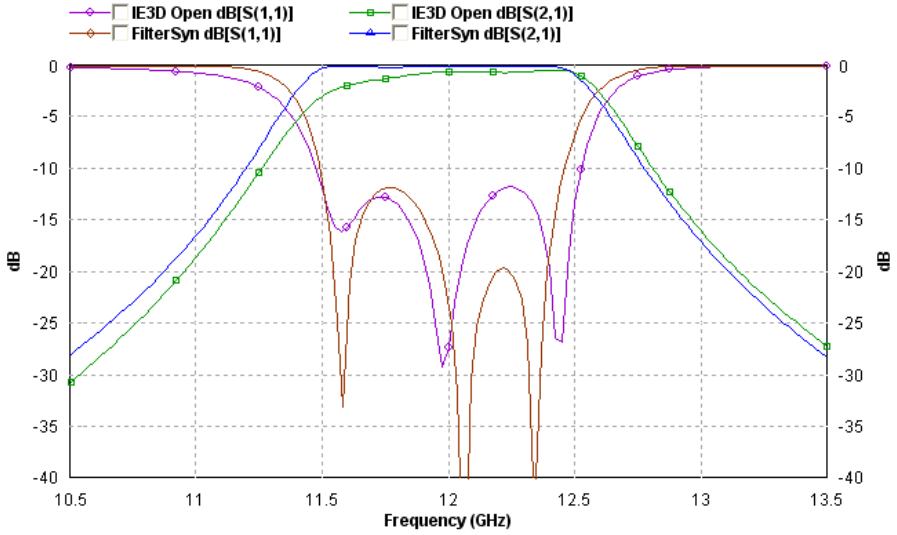


Figure 10.13 The comparison of the IE3D open formulation and FilterSyn.

Section 10.8 Relocating Structure to Specified Location

Step 1 Please open the .\ie3d\practice\bpf1.geo on MGRID. When we drop the object onto the Ie3dLibrary window, we did not specify the absolute location of the object. We want to re-locate it to some specific point so that we can use absolute coordinates to discuss the enclosure locations. Please click at the Z = 15 mils on the layer window to focus the input at Z = 15 mils. Select **Input->Set to Closest Vertex** command. Click at the vertex 1 in Figure 10.14 to snap to the vertex.

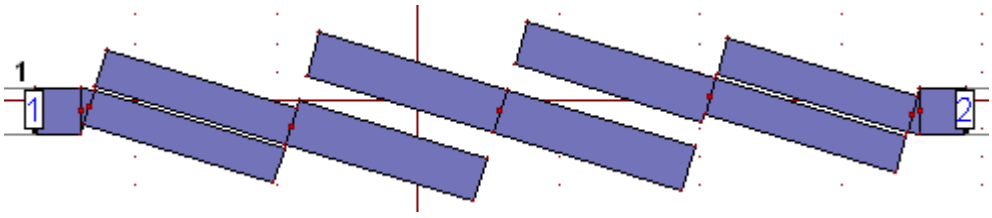


Figure 10.14 The BPF structure on MGRID.

Step 2 Select **Input->Info on Last Entry** command. MGRID will show the information on the entered vertex. It is located at (X,Y,Z) = (-266.404, 8.1685, 15) mils. We want to relocate the filter so that this vertex will be at (X, Y, Z) = (0, 0, 15). Certainly, if we move the object by dx = 266.404mils and dy = -8.1685 mils, we will be able to relocate the filter with the vertex at (X, Y, Z) = (0, 0, 15). We do not need to take down those long digits. There is a fast way to achieve it.

Step 3 Please select "Save 1st set" in the Info on Last Entry dialog. Then, select Close and Drop Vertices button. The dialog will be closed and the entered vertex is dropped.

Step 4 Press down "Shift" button and window the complete filter to select all the polygons. Select Edit->Move Objects command. The moved objects are following the mouse cursor. Click at anywhere the drop it. MGIRD will prompt you for the offset values (see Figure 10.16).

Step 5 You should notice that the saved (X, Y, Z) values are listed as Saved Information in the dialog (see Figure 10.16). Select Get Saved Values button in the dialog. The saved X, Y and Z values will be transferred to the X-offset, Y-offset and Z-offset values.

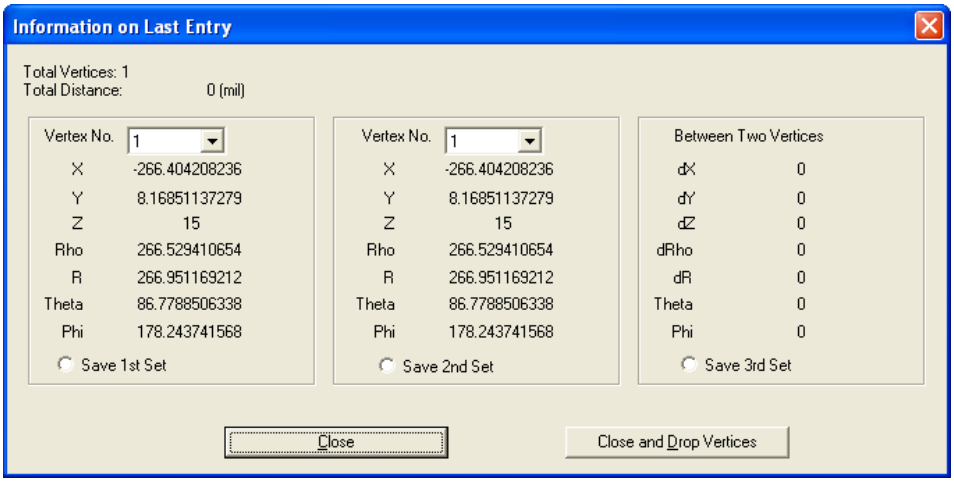


Figure 10.15 The Info on Last Entry dialog in Input menu of MGRID.

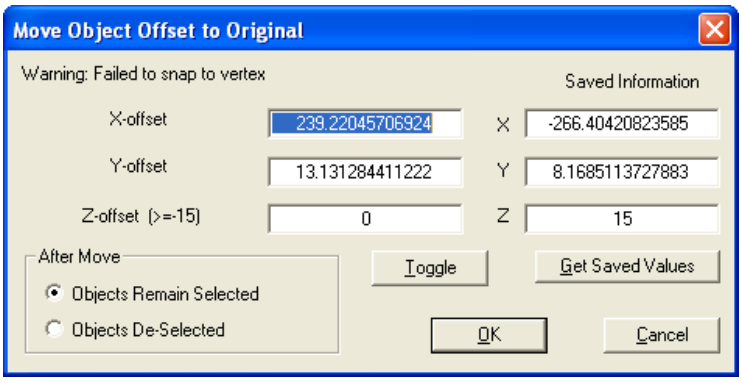


Figure 10.16 The Move Object Offset dialog.

Step 6 Change the sign of X-offset value and Y-offset value. We will get X-offset = 266.40420823585 and the Y-offset = -8.1685113727883. Change the Z-offset value to 0. The After Move will automatically shift to Objects De-Selected. Select OK to continue. The filter will be relocated such that the vertex 1 in Figure 10.8 is moved to (X, Y, Z) = (0, 0, 15) mils. MGRID is back to

drawing mode. Save the geometry as: .\ie3d\practice\bpf2a.geo. We are going to define the enclosure walls.

Step 7 Select **Param->Basic Parameters** dialog. Click at the **No. 0: No Side Walls** on the Enclosures section. MGRID will prompt you for the parameters. Enter the Xmin Wall = -180 mils, Xmax Wall = 830 mils, Ymin Wall = -102 mils and Ymax Wall = 70 mils. Enter the X-Direction: Spectral Terms = 30 and Y-Direction: Spectral Terms = 10 (see Figure 10.17). Select OK to continue. MGRID will get back to the Basic Parameters.

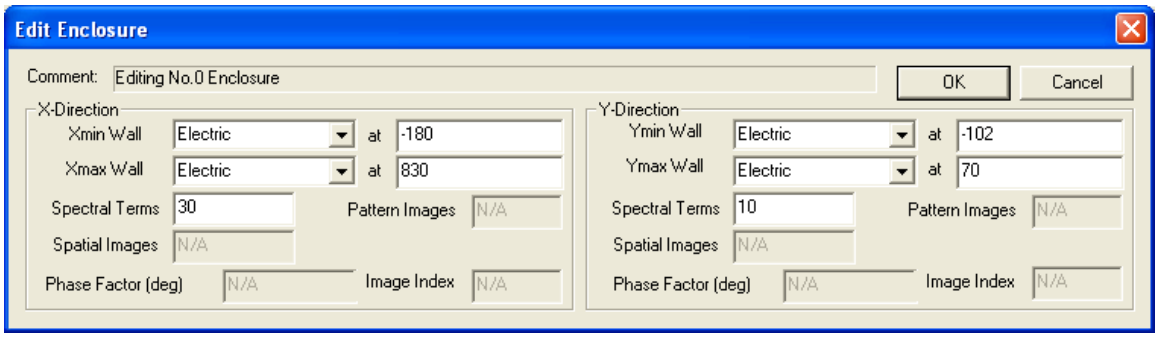


Figure 10.17 The Edit Enclosure dialog.

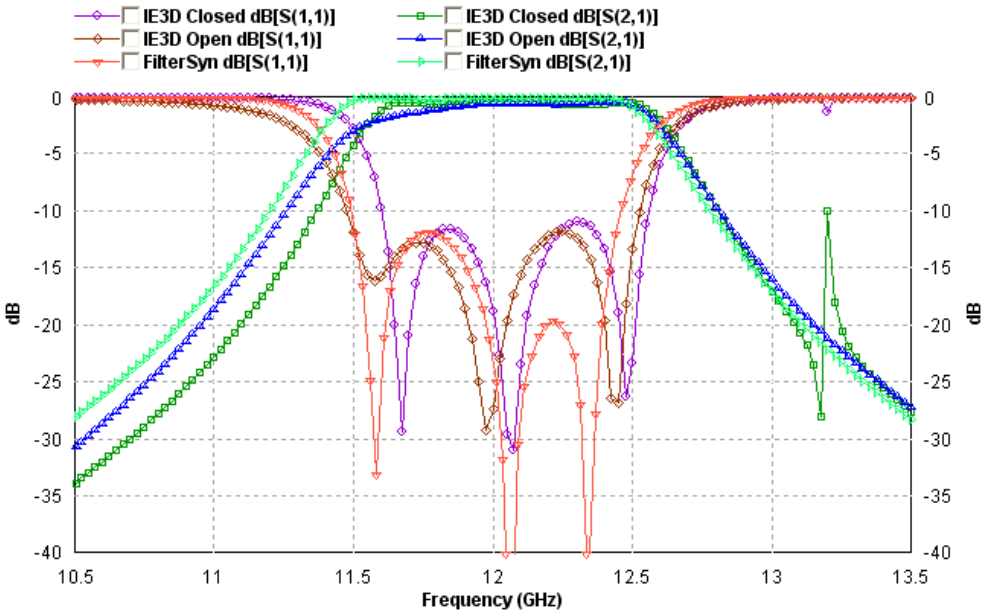


Figure 10.18 The comparison among open, closed IE3D models and FilterSyn model.

Step 8 Select check the **Substrates** in the **Basic Parameters** dialog. We have the substrates as:

No.2 Ztop = 3.937061e+16, Er = (1, 0), Mur = (1, 0), Sigma = (0, 0)
No.1 Ztop = 15, Er = (3.78, 0), Mur = (1, 0), Sigma = (0, 0)
No.0 Ztop = 0, Er = (1, 0), Mur = (1, 0), Sigm a= (4.9e+7, 0).

Change them to:

No.3 Ztop = 3.937061e+16, Er = (1, 0), Mur = (1, 0), Sigma = (4.9e7, 0)

No.2 Ztop = 500, Er = (1, 0), Mur = (1, 0), Sigma = (0, 0)
No.1 Ztop = 15, Er = (3.78, 0), Mur = (1, 0), Sigma = (0, 0)
No.0 Ztop = 0, Er = (1, 0), Mur = (1, 0), Sigm a= (4.9e+7, 0).

Basically, we added a metal cover at Z = 500 mils. Save the file as .\ie3d\practice\bpf3a.geo. Simulate it from 10.5 go 13.5 GHz with 121 frequency points. It takes some minutes to finish it. The comparison among the bpf1a.sp (IE3D open), bpf3a.sp (IE3D closed) and FilterSyn (bpf1.sp) is shown in Figure 10.18.

The IE3D results with enclosure (or IE3D Closed) predict flatter passband responses. The radiation is prevented by the enclosure. However, the IE3D Closed model predicts a narrower bandwidth than both IE3D Open model and the FilterSyn. We expect the bandwidth will be dependent upon the size of the enclosures and the height of the top wall. The IE3D should predict the real environment. On the other hand, the FilterSyn result is not bad. The FilterSyn uses analytical formulas for lumped elements and transmission line elements. No radiation and metallic losses are included. It will not be able to predict the effects from the enclosure or the open boundary. You may notice a sharp resonance at 13.2 GHz for the IE3D Closed model. It is from the resonance of the enclosure in the X-direction.

You should also realize that the IE3D simulation speed of enclosure model is much slower than the open box model. An IE3D simulation time is divided into 2 parts: (1) Filling the matrix and (2) Solving the matrix. For small and medium size problems, filling matrix is dominant. For medium and large size problems, solving matrix is dominant. The enclosure model may slow down the filling matrix part by a factor of more than 10.

Section 10.9 Modeling of Filters with Thick Substrates

In simulating high frequency microwave circuits, we need to be very careful on selecting the correct de-embedding schemes. Theoretically, there is no limitation on the simulation frequency for a full wave simulator like the IE3D. However, there is limitation on the de-embedding. The de-embedding of IE3D assumes single propagating mode. In practical applications, we always require single propagating mode at ports although we allow multiple propagating modes inside a circuit. In the other words, if we cannot guarantee single propagating mode, we will not be able to maintain unique network parameters because the network parameters become dependent upon the excitation.

Besides the single mode operation, we have further limitation on the **Extension for MMIC** scheme. Because the **Extension for MMIC** scheme uses the voltage concept, we cannot use it at very high frequency, where high dispersion occurs. We have discussion on the limitations of the different ports in Section 4 of Chapter 12, and we will not repeat here.

Saved in **.\ie3d\samples\lpfilt1.geo** is an example the **Extension for MMIC** scheme fails. The structure is shown in Figure 10.19. The low pass filter was published in the training course: "EM field simulator applications in practical circuit design," Besser Associates (Instructor: Dan Swanson) in January 1996. It was designed and proved to have a critical frequency at 19.5 GHz with good insertion loss in the pass-band. We assume 30 GHz is the highest application frequency or the **Fmax**. The substrate thickness **H** is 15 mils and the dielectric constant $\boldsymbol{\varepsilon_r}$ is 9.8. Let's check these parameters against the criteria in Table 12.5. The structure is simulated from 0.1 to 40 GHz for 400 frequency points using the **Advanced Extension** scheme. It takes a very short time for the IE3D to get the results using AIF. IE3D may issue some warning on thick substrate. Basically, there might be 2^{nd} order mode excited beyond 30 GHz for such a thick substrate. However, the Advanced Extension scheme should be ok for it even MGRID issues you a warning. The IE3D predicts the critical frequency exactly at 19.5 GHz (Figure 10.20). The simulation results also indicate the insertion loss in most of the pass band is below –30 dB. It is an excellent design example.

There is increasing abnormal loss beyond the 32 GHz. We can guess that we start getting the second propagating mode at about 32 GHz. The accuracy beyond 32 GHz cannot be guaranteed. Also, we should never use such a structure beyond 30 GHz.

If we apply the **Extension for MMIC** scheme on this example, we will observe high loss beyond 15 GHz. The loss is not actually there. It is just due to the limitation on the **Extension for MMIC** de-embedding scheme. This is a good example to demonstrate how the different de-embedding schemes work and what their valid ranges are.

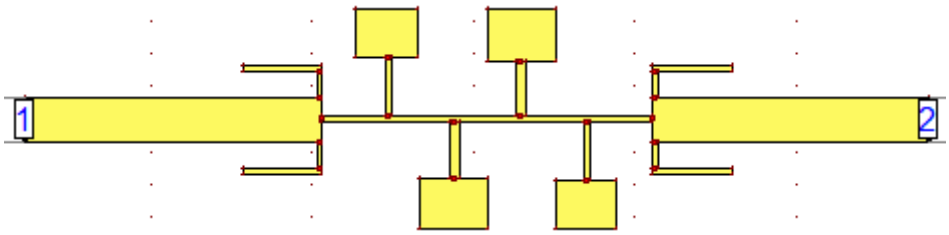


Figure 10.19 The layout of the lowpass filter.

In practical filter design, some users still use very thick substrates even though thick substrates are not good to the circuit performance and easy to introduce dispersion and higher order modes. Still good designs can be achieved using quite thick substrates. The trick is in the enclosure. For a filter using thick substrate, it has to be put into a box. The box will prevent the power from radiating out. The thick substrate may create high dispersion or even higher order modes. However, they may not create passband radiation from the filter because the input and the output are normally SMA connectors or other coaxial types of connectors. There might be higher order modes excited inside the box. However, the power of the higher order modes will be converted back to the TEM mode of the coaxial connectors at the input and output. As long as we can design the filter with specified output performance, the filter can be treated as a black box. For such a situation, the filter should be used in the same boxed environment. Changing the dimensions of the box will affect the performance of the filter inside. The box becomes an integrated part of the structure. In some sense, use of the enclosure can break the limit on the substrate thickness. The trade-off is that your circuit with thick substrate inside the box will no longer obey the microwave transmission line and network theory. All different parts of the circuit may have strong coupling between them even when they are far apart. We have to treat the whole structure including the box as an entity.

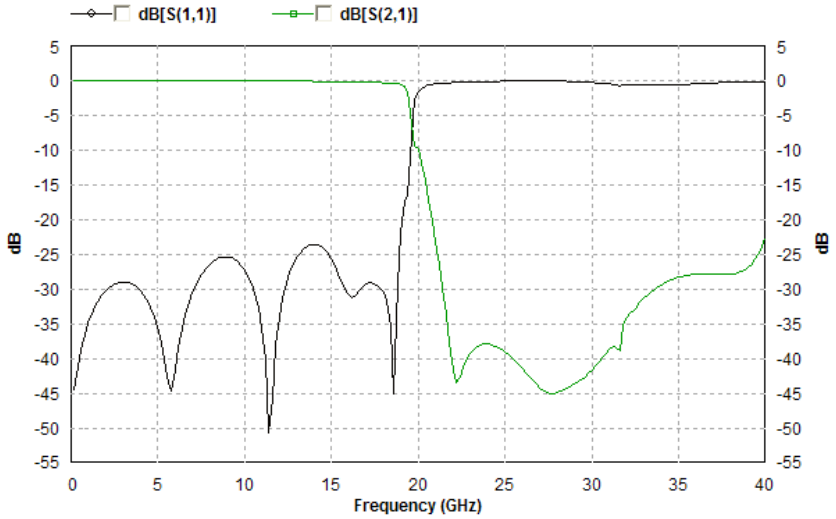


Figure 10.20 The simulation result using the **Advanced Extension** scheme.

Section 10.10 Electromagnetic Optimization in Filter Design

In our final example of this chapter, we will discuss the application of the genetic optimizer in filter design. We will not start from scratch. We will start from a well-designed filter documented in the IE3D benchmark example ---- a HTS hairpin filter from a user of the IE3D. The hairpin filter is well designed and manually tuned. We will de-tune the filter by changing some of its dimensions. Then, we will try to optimize it again using the Genetic optimizer feature in the IE3D.

Step 1 Open **.\ie3d\samples\hairpin.geo**. The structure is shown in Figure 10.21. Please save it into **.\ie3d\practice\hairpin.geo**. Simulate the structure from 0.85 to 0.91 GHz with 121 frequency points. Please enable the **AIF** scheme. Please un-check the **Current Distribution File** button. Please disable the **AEC** because the strips are quite thin and one cell may yield quite accurate results. The result is shown in Figure 10.24 labeled as "original". The frequency response is perfect. It also agrees very well with the measured data documented in the IE3D benchmark example brochure. We are going to de-tune the structure.

When you select OK, MGRID will inform you that "Minimum forced meshing size happens between (0, 0, 20) – (0.0787402, 0, 20) with size 0.0787402 (mil) (<0.209071). Are you sure you want to continue". This is a warning message for possible irregular meshing. Normally, when the size (0.0787402 mils in this example) is not at least 3 orders smaller than the limit (0.209071 mils in this example), we do not need to worry about it.

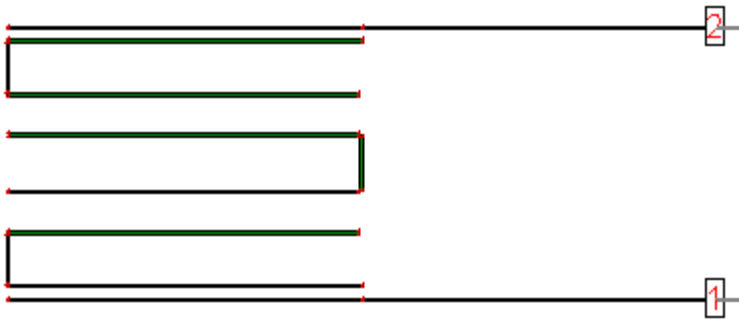


Figure 10.21 A manually tuned hairpin filter

Step 2 Press down "Shift" button and window the lower part of the structure to select it (the selection 1 shown in Figure 10.22). Select **Edit->Move Objects** command. Click somewhere and enter X-offset = 0 mil and Y-offset = -25 mils. Select OK to continue. The gap width between the lower folded pole and the center folded pole is increased by 25 mils.

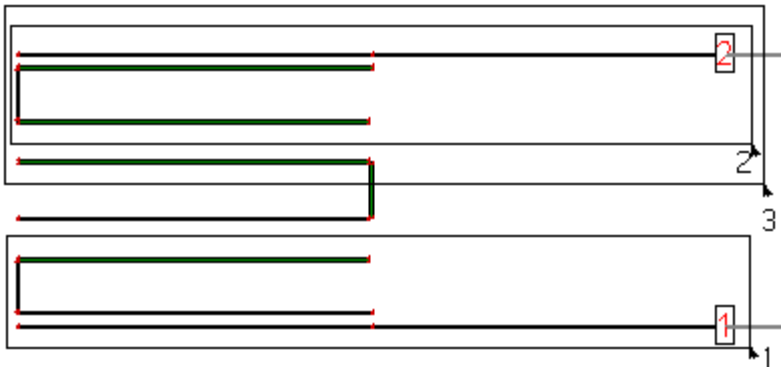


Figure 10.22 The selection windows for de-tuning the filter.

Step 3 Press down "Shift" button and window the upper part of the structure (the selection 2 shown in Figure 10.22). Select **Edit->Move Objects** command. Click somewhere and enter X-offset = 0 mil and Y-offset = 25 mils. Select OK to continue. The gap width between the upper folded pole and the center folded dipole is increased by 25 mils.

Step 4 Select **Edit->Select Vertices** command. Window the complete upper part of the structure (the selection 3 shown in Figure 10.22). Select **Edit->Move Objects** command. Click somewhere and enter X-offset = 0 mil and Y-offset = 20 mils. Check the **Objects De-Selected** and select OK to continue. The length of the center folded pole is increased by 20 mils.

We cannot use the "shift" plus windowing to select the vertices. If we do so, we will be selecting the polygons instead of the vertices.

Step 5 Save the geometry as "**.\ie3d\practice\hairpin1.geo**". Simulate it from 0.85 to 0.91 GHz with 121 frequency points using the **AIF** again. The result is shown in Figure 10.24 labeled as "de-tuned". The frequency response is completely different and we need to tune it again. It will not be manually. Instead, we will use the Adaptive EM Optimizer to optimize it.

Step 6 Press down "Shift" and window the portion for the selection 1 in Figure 10.22 again to select the lower part of the circuit. Select **Optim->Variable for Selected Objects** command. Select New Variable for the Vertices Mapped to. Enter the Tuning Angle = 90 degrees. Select OK. Move the mouse cursor somewhere toward the lower direction and click. Enter the **Low Bound** as -5 mils. Move the cursor somewhere toward the upper direction and click. Enter the **High Bound** as 35 mils. Select OK to continue. Select Continue without Action.

We have defined the gap width between the lower folded pole and the center folded pole as the No.1 optimization variable. MGRID will warn the user further geometry editing will cause removal of the optimization variable.

Step 7 Press down "Shift" and window the portion for the selection 2 in Figure 10.22. Select **Optim->Selected Objects to Variable** command. Select **Vertices Mapped to: No. 1 Variable**. Enter the Tuning Angle as –90 degrees, the Tuning Rate = 1. Select OK. Select Continue without Action. The selected vertices are added to the No.1 variable.

Since we want the structure to be symmetrical, we want to associate the change of the vertices with the change of the vertices defined as variable 1 in Step 6. We want them to change in opposite direction.

Step 8 Select **Edit->Select Vertices** command. Window the portion for the selection 3 in Figure 10.22 to select the vertices. Select **Optim->Variable for Selected Objects** command. Select **Vertices Mapped to: New Variable**. Enter the **Tuning Angle** as 90 degrees. Move the mouse cursor somewhere toward the lower direction and click. Enter the **Low Bound** as -30 mils. Move the cursor somewhere toward the upper direction and click. Enter the **High Bound** as 10 mils. Select OK. Select Continue without Action. The length of the center folded pole is defined as the variable 2.

Step 10 Save the structure as **.\ie3d\practice\hairpin2.geo**. Perform the meshing with some offset values to the variables for a few times. You will see how the optimization variables are controlling the gap between the poles and the length of the center pole. The structure should be kept symmetrical in the tuning.

Step 11 Select **Process->Optimize** command. The **Optimization Setup** dialog comes up. The frequencies defined in the last simulation should still be there. They should be from 0.85 to 0.91 GHz with 121 points.

Step 12 Select **Add** button. Enter **Start Freq** = 0.884 and **End Freq** = 0.888 GHz. Select **dB(S)** for **Parameter Type**. Select (1, 1) for the **1st Parameter**. Select **Optimization Quantity <= Objective1** for **Objective Type**. Enter "-25" for the **Objective1**. Select OK to define the 1st set of goals.

|S(1,1)| < -25 dB between 0.884 and 0.888 GHz is required. It is the pass band.

Step 13 Select **Add** button again. Enter **Start Freq** = 0.866 and **End Freq** = 0.876 GHz. Select **dB(S)** for **Parameter Type**. Select (2, 1) for the **1st Parameter**. Select **Optimization Quantity <= Objective1** for **Objective Type**. Enter "-40" for the **Objective1**. Select OK to define the 2nd set of goals.

|S(2,1)| < -40 dB between 0.866 and 0.876 GHz is required. It is the lower stop band.

Step 14 Select **Add** button again. Enter **Start Freq** = 0.896 and **End Freq** = 0.906 GHz. Select **dB(S)** for **Parameter Type**. Select (2, 1) for the **1st Parameter**. Select **Optimization Quantity <= Objective1** for **Objective Type**. Enter "-40" for the **Objective1**. Select OK to define the 3rd set of goals.

|S(2,1)| < -40 dB between 0.896 and 0.906 GHz is required. It is the upper stop band. We should get the display in Figure 10.23 for the **Optimization Setup** dialog. It is indicated that we have two variables and 51 objectives.

Step 15 Make sure the scheme is Adaptive EM Optimizer. Select OK to invoke the IE3D for the optimization of the structure.

It is time to take a break. You should let the Adaptive EM Optimizer of IE3D to run some minutes. You can terminate the optimization any time you want.

Starting from IE3D 8.0, you have the option to change the process priority in the Simulation Setup or Optimization Setup dialog. If you want to continue this tutorial without waiting for the results and want the computer to be more responsive, you can set the process priority to below normal.

You will see 3 values in the Residuals. The 1st one is the current residual. The 2nd one is the minimum residual and the 3rd one is the limit. The minimum residual is decreasing with more runs. It should not be easy for the optimizer to reach the Error Limit of 0.01. Very frequently, we may define very demanding goals and the optimizer may not be able to meet the goals. Then, you may ask how we should know the result might be good enough for us to terminate the optimization.

In fact, during the optimization, you can always open the intermediate file for the current best geometry. For this case, we open the **hairpin2m.geo**. We should save it into another file name and simulate it to see how good it is. If it is good enough, we can terminate the optimization and use the current **hairpin2m.geo** as our best optimization result.

Step 16 You can terminate the optimization after 5 minutes. You will get the best geometry saved in: .\ie3d\practice\hairpin2m.geo. The minimum residual is about 1.62 compared to the limit of 0.01. Basically, our default residual limit of 0.01 is too high a requirement and it is not achievable. Simulate the optimized file **hairpin2m.geo** from 0.85 to 0.91 GHz with 121 frequency points. Use MODUA to compare the results of the original geometry, de-tuned geometry and optimized geometry (see Figure 10.24). You can see the optimized structure has a

very good pass-band and stop-band properties. Its response is close to the original one which was manually optimized.

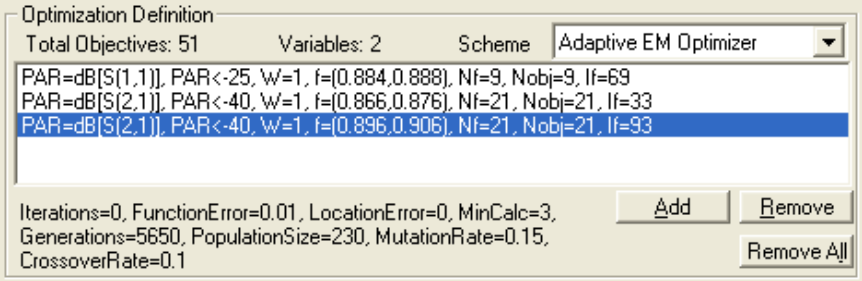


Figure 10.23 Portion of the Optimization Setup dialog after the objectives are defined.

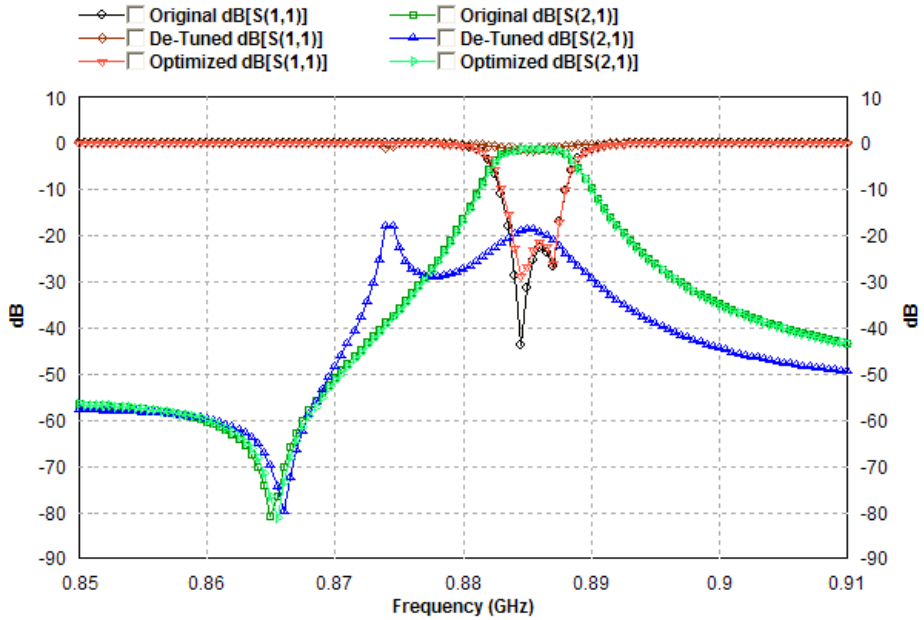


Figure 10.24 The comparison among original, de-tuned and optimized filters.

Section 10.11 Summary on IE3DLibrary, FilterSyn and Filter Designs

We have demonstrated how we can use IE3D including IE3DLibrary and MGRID for filter designs in this chapter. With the further improvement on the usability of IE3DLibrary, it makes IE3DLibrary and the FilterSyn much more powerful for synthesis of microwave components including filters and couplers. In the IE3DLibrary examples here, we have not demonstrated the features of equation-based geometry modeling with parameterization and Boolean operations. They are the most important and powerful features of IE3DLibrary. More documentations on using IE3DLibrary can be found from Appendix AH and the IE3DLibrary User's Manual (.\ie3d\manual\ie3dlibrary_manual.pdf).

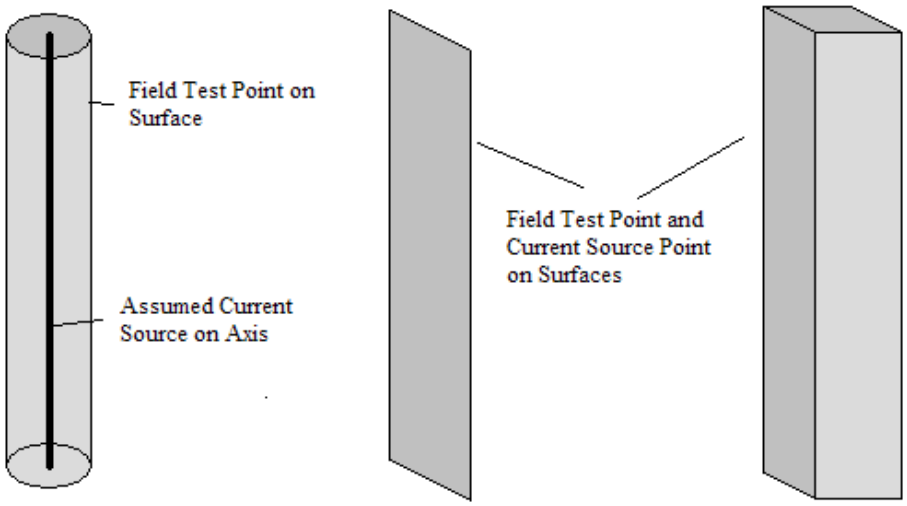
Chapter 11 Modeling of Wire Antennas and Other RF Antennas

IE3D is not only good for patch antennas but also good for wire antennas and other RF antennas. We have demonstrated how to model edge-fed antennas in Chapter 5, probe-fed patch antennas in Chapter 7 and aperture coupled-fed patch antennas in Chapter 8. In this chapter, we will discuss the modeling of wire antennas and other RF antennas.

Section 11.1 Discussion on Modeling Wire Antennas

Traditionally, a wire is modeled with separated source point and observation point or field test point (see Figure 11.1a). In order to avoid the singularity in numeral calculation, we assume the electric current is along the axis of the wire and the location where we test the field is on the surface of the wire. It is a good approximation when the wire is very thin compared to the length and wavelength. Normally, it can yield accurate real part of the input impedance for a typical half-wavelength dipole antenna. We call the model the thin wire approximation. The thin wire approximation certainly introduces some error, because we know the current is mainly flowing on the surface of the wire.

On the IE3D, we always model the surface current on metallic structures and we also test the field on the surface of metallic structures. Depending upon the accuracy requirement, we can model a thin wire as a metallic strip or a ribbon (see Figure 11.1b), a rectangular tube (see Figure 11.1c) or a tube with more sides.



(a) Thin wire model (b) Ribbon model on IE3D (c) Tube model on IE3D

Figure 11.1 Different models of wire antennas.

The metallic strip model in Figure 11.1b and the tube model in Figure 11.1c assume both the current source and the observation point to be on the surface. They are closer to the reality and they can yield much more accurate results than the thin wire model. Normally, the tube model should take more time to simulate.

Section 11.2 Discussion on Dipole Antennas

We start from a simple dipole antenna. The total length of the dipole is 15 mm. The radius of the dipole is 0.075 mm (0.5% of the length). We expect the resonant frequency to be slightly lower than 10 GHz. There are a few different ways to model such a dipole antenna depending upon which de-embedding scheme we use. We will try to use the different model on it. The 1st model we want to use is the differential **Advanced Extension** scheme. The file is saved in c:\ie3d\samples\dipole6.geo. We will re-build it for practice:

Step 1 Run MGRID. Open c:\ie3d\samples\dipole6.geo. Please select Basic Parameters to check the dielectric configuration.

No.0 Ztop = 0 mm, $\varepsilon_r = 1$, $\mu_r = 1$, $\sigma = 0$
No.1 Ztop = 1.0e+15 mm, $\varepsilon_r = 1$, $\mu_r = 1$, $\sigma = 0$

It is free space everywhere. Select OK to continue. MGRID is in the default drawing mode.

Step 2 Please select all the polygons on the window and delete them. We are going to rebuild the structure.

Select **Input->Create and Edit Vertices** command. Select the **Import** button. Select **OK** when prompted. Select **c:\ie3d\samples\dipole.txt**. MGRID will inform you that 6 vertices are imported and whether you want to give it some offset values. Select OK to continue. The 6 vertices are imported. The 6 vertices define the centerline of the dipole we are going to build. Select OK and MGRID resumes to drawing mode with the 6-vertices entered.

Step 3 Select **Adv Edit->Build Wire Path** command. Enter the **Number of Segments for Circle** = 6 and the **Radius** = 0.075. We will get the dialog shown in Figure 11.2. Select **OK** to accept the other default settings. MGRID will build a wire path with a small bend at the center (see Figure 11.3a). Select **View->Whole Circuit** command display the whole structure.

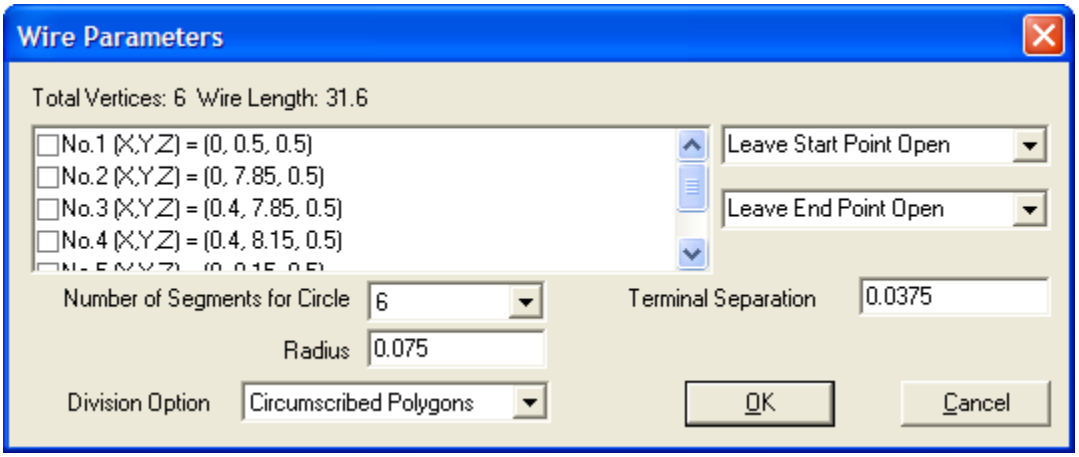
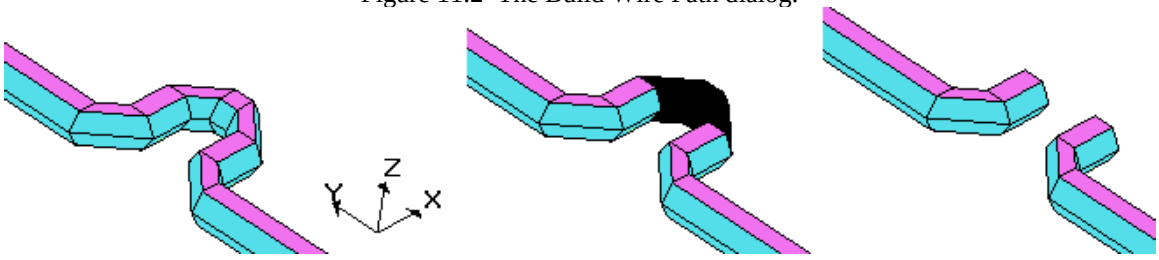


Figure 11.2 The Build Wire Path dialog.

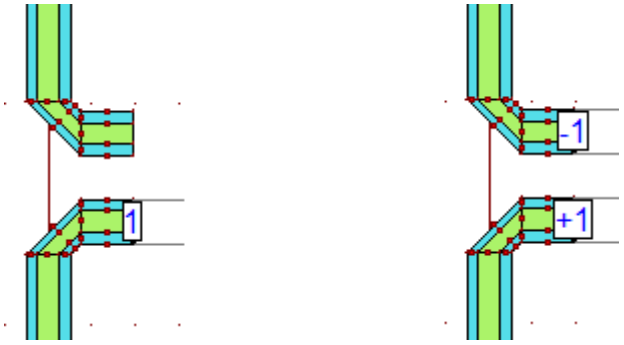


(a) Before the bends are deleted (b) The bends selected (c) After the bends are deleted.

Figure 11.3

Step 4 Press down "Shift" button and window the two right-angle bends at the center to select the polygons (see Figure 11.3b). Select **Edit->Delete** command to delete the polygons on the bends. We will get the picture in Figure 11.3c.

Step 5 Select **Port->Port For Edge Group** command. Select **Advanced Extension** scheme. Please make sure all the layers are checked on the Layer Window. Please window one of the terminals to define the port 1 on it. Please click the right mouse button to bring up the pop-up window. Select **Define Negative Port**. Then, window the edges of the other terminal to define the port –1 on it. The port 1 automatically becomes port +1. Select **Port->Exit Port** command to resume drawing mode. The final result at the feed location is shown in Figure 11.4.



(a) Port 1 is defined. (b) Port 1 changed to Port +1 after Port -1 defined.

Figure 11.4 The bends with ports defined.

Step 6 Save the structure as c:**\ie3d\practice\dipole6.geo**. Simulate the structure from 5 to 15 GHz with 101 frequency points. It takes couple seconds to finish it.

Step 7 Select all the polygons of the structure. Delete them. Repeat Step 2 to Step 5. In Step 3, you should enter the **Number of Segments for Circle** as 4. You will build the dipole with a rectangular tube. Save the structure as **c:\ie3d\practice\dipole4.geo** and the simulation result as **c:\ie3d\practice\dipole4.sp**.

Step 8 Select all the polygons of the structure. Delete them. Repeat Step 2 to Step 5. In Step 3, you should enter the **Number of Segments for Circle** as 2. You will build the dipole with a flat strip. Save the structure as **c:\ie3d\practice\dipole2.geo** and the simulation result as **c:\ie3d\practice\dipole2.sp**. Comparisons among the 3 cases are shown in Figure 11.5 and Table 11.1. Please note that the Re[Z(1,1)] is using the left vertical scale and the Im[Z(1,1)] is using the right vertical scale and they are different. The s-parameters look quite the same for the 3-models. There is some difference in the Im[Z(1,1)] of the resonance.

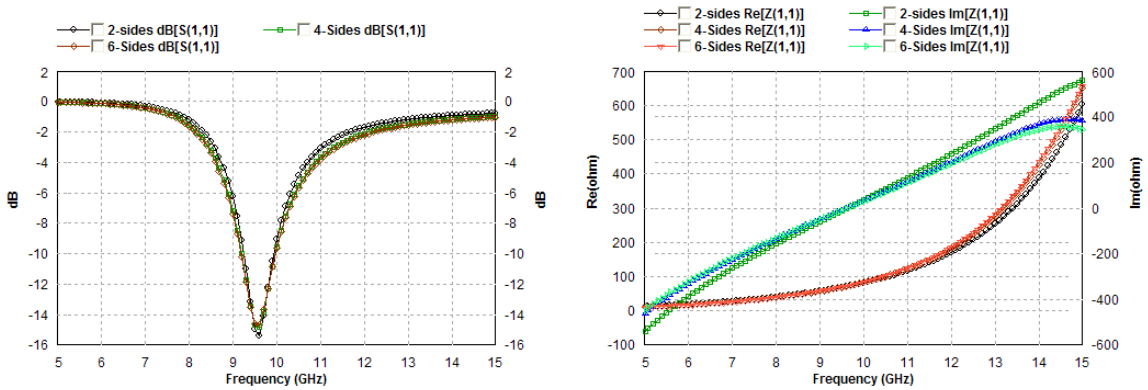


Figure 11.5 The s-parameters input impedance of the dipole for 3 different models.

Table 11.1 The comparison among the 3 models.

Wire Approximation	1-Side (or 2-Sides)	4-Sides	6-Sides
Total Time (seconds)	0	2	6
Resonant (Im[Zin] = 0)	≈9.617 GHz	≈9.629 GHz	≈9.628 GHz
Re[Zin] at Resonance	70.82 Ω	72.89 Ω	73.37 Ω
Directivity	2.157 dB	2.162	2.163
Efficiency	95.6%	95.8%	95.7%

As we can see, the differences among the 3 models are quite small. The flat strip model is reasonably accurate around the resonance for the thin wire antennas. However, when the radius of the wire becomes bigger or when it is getting farther from the 1st resonance, we will need the more accurate tube models.

Section 11.3 Modeling Dipole Antennas using Localized Ports.

We used the **Extension for MMIC** scheme for modeling of the above examples. In fact, we can do equally well or even better using the **Vertical Localized** and **Horizontal Localized** ports to model a vertical dipole.

Step 1 Open file: c**:\ie3d\practice\dipole6.geo**. Select all polygons and delete them.

Step 2 Select **Input->Create and Edit Vertices** command. Select the **Import** button. Select **OK** when prompted. Select **c:\ie3d\samples\dipolev.txt**. MGRID will inform you that 3 vertices are imported and whether you want to give it some offset values. Select OK to continue. The 3 vertices are imported and listed. The 3 vertices define the centerline of the dipole we are going to build. The three vertices are at: (X, Y, Z) = (0, 0.5, 0.5), (0, 0.5, 7.95) and (0, 0.5, 15.5). We purposely make the 3 vertices non-symmetrically. The distance from the 1st and 2nd vertices is 7.45 mm and the distance from the 2nd and 3rd vertices is 7.55 mm. The reason for it is that we will use the Adv Edit->Build Wire Path command to build the dipole with a vertical localized port on it automatically. The vertical localized port will be defined from Z = 7.95 to Z = 8.05. Then, the two arms will be of the same length.

Step 3 Select OK and MGRID resumes to drawing mode with the 3-vertices entered.

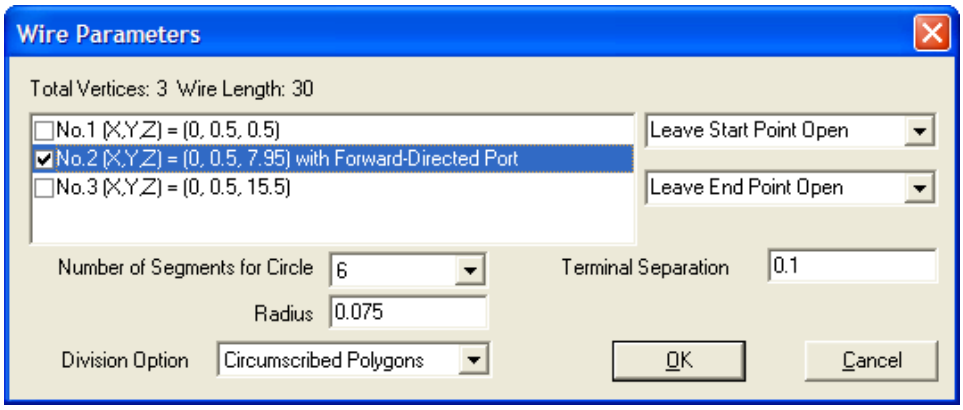


Figure 11.6 The dialog for building the wire path with a vertical localized port.

Step 4 Select **Adv Edit->Build Wire Path** command. Enter the **Number of Segments for Circle** = 6 and the **Radius** = 0.075. Please check the No.2 vertex in the list (see Figure 11.6). It defines MGRID to build a vertical localized port at the 2nd vertex. The Terminal Separation will be automatically set to 0.0375. Change the Terminal Separation to 0.1. We will get the dialog shown in Figure 11.6. Basically, we are trying to build a wire path as 3 segments. From the No.1

vertex at (0, 0.5, 0.5) to the No.2 vertex at (0, 0.5, 7.95) is the 1st segment. It is the 1st arm of the dipole. From the No.2 vertex at (0, 0.5, 7.95) to (0, 0.5, 7.95+0.1)=(0, 0.5, 8.05) as the 2nd segment. The 2nd segment will be defined as a vertical localized port automatically. The 3rd segment is from (0, 0.5, 8.05) to (0, 0.5, 15.5) and it is the 2nd arm of the dipole.

Step 5 Select OK to continue. MGRID will create a vertical dipole with a vertical localized port exactly at the center. Save the geometry file as: c:\ie3d\practice\diplole6v.geo. Simulate it from 5 to 15 GHz with 101 points.

Step 6 Select all polygons and delete them.

Step 7 Select **Input->Create and Edit Vertices** command. Select the **Import** button. Select **OK** when prompted. Select **c:\ie3d\samples\dipoleh.txt**. MGRID will inform you that 3 vertices are imported and whether you want to give it some offset values. Select OK to continue. The 3 vertices are imported and listed. The 3 vertices define the centerline of the dipole we are going to build. The three vertices are at: (X, Y, Z) = (0, 0.5, 0.5), (0, 7.95, 0.5) and (0, 15.5, 0.5). The vertices in dipoleh.txt are basically the same as those in dipolev.txt file with the Y and Z-coordinates swapped. Using the dipolev.txt, we built the vertical dipole. Using the dipoleh.txt file, we will build a horizontal localized port.

Step 4 Select **Adv Edit->Build Wire Path** in **Adv Edit** menu. Enter the **Number of Segments for Circle** = 6 and the **Radius** = 0.075. Check the No.2 vertex in the list. The Terminal Separation will be automatically set to 0.0375. Change the Terminal Separation to 0.1. We will get the dialog similar to the one shown in Figure 11.6. Only the vertices are different. Select OK to continue. MGRID will build a horizontal dipole with a horizontal localized port at the center. Save the file as c:\ie3d\practice\dipole6h.geo. Simulate it from 5 to 15 GHz with 101 points. The comparison among the 3 schemes, (1) Advanced Extension (Dipole6), (2) Vertical Localized (Dipole6v), (3) Horizontal Localized (Dipole6h), is shown in Figure 11.7. The Dipole6v and Dipole6h are almost identical. The Dipole6 is slightly off. The difference does not indicate which scheme is better in accuracy. All of them are accurate. However, the slight difference in the feed proximity is changing the properties of the antenna slightly. If your dipole is, in fact, a monopole over a large ground plane and fed by a coaxial, the vertical localized and the horizontal localized ports are closer to reality. If your dipole is fed by a parallel line, the Advanced Extension port is the better choice.

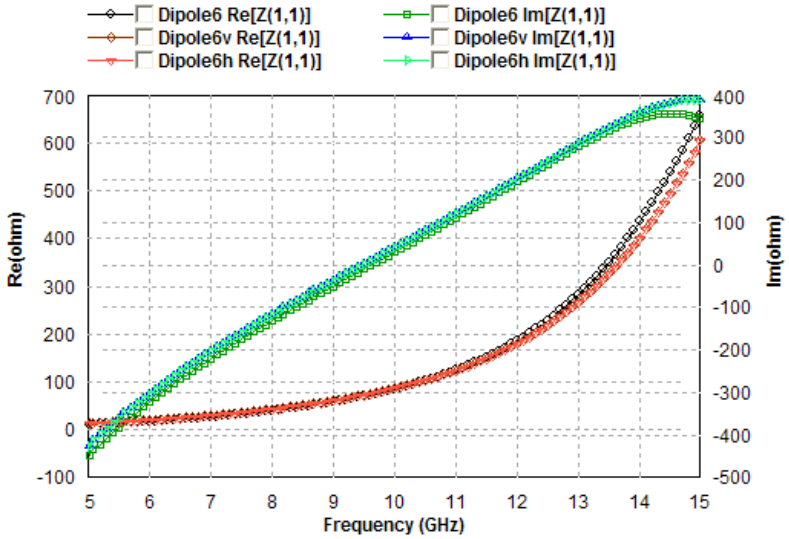


Figure 11.7 The comparison among the 3 different port schemes.

Section 11.4 Modeling of a Cylindrical Helix on a Handset

There is great interest in the modeling of a cellular handset. In this section, we will try to build a simple handset model. Assuming the handset's dimensions are 5 inches by 2 inches by 0.8 inches. The length of the helix portion is 600 mils. The radius of the helix is 200 mils. There are total 4 turns for the helix. The wire radius is 30 mils. The length of the wire before the helix is 150 mils. The length of the wire after the helix is 3.5 inches. We will neglect the plastic portion.

Step 1 Delete all the polygons on the window. Select Basic Parameters in Param menu. Change the Length Unit to mil. Define the **Metallic Strip Parameters** as:

No.1 Thickness = 4 mils, $\varepsilon_r = 1$, $\mu_r = 1$, $\sigma = 4.9e7$ s/m

Step 2 Define the **Meshing Parameters** as: **Fmax** = 1.2 GHz and **Ncell** = 15. Change the Grid Size for the layout as 100 mils. Select OK to continue.

Step 3 We are going to build the bottom of the handset first. Select Insert a Layer on the Layer Window. Enter Z = 0. Click the left button at x = 0 and y = 0 to enter the first vertex at (0, 0, 0). Type **Shift+R** and enter dx = 2000 mils and dy = 0 to enter the second vertex at (2000, 0, 0). Type **Shift+R** and enter dx = 0 and dy = 800 mils to enter the third vertex at (0, 800, 0). Type **Shift+F** to form the rectangle for the bottom of the handset.

Step 4 Select **Edit->Select Vertices** command. Window the 4 vertices of the rectangle to select them. Select **Edit->Add Via on Edges** command. Enter **End Z-Coordinate** as 5000 mils. The handset is built with the top cover opened (see Figure 11.8). We will not build the top cover first. We will build the helix monopole in the next step.

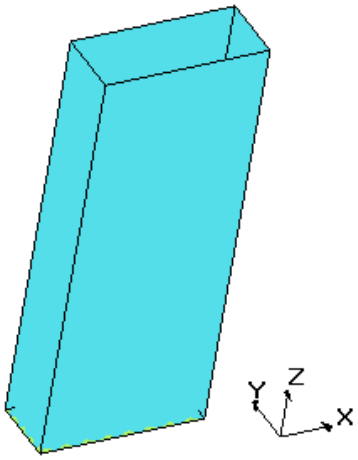


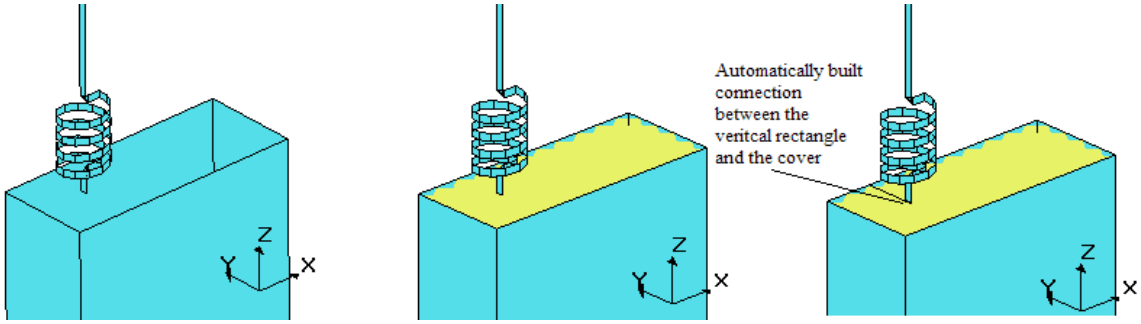
Figure 11.8 The handset is built with the top cover removed.

Step 5 Select **Entity->Cylindrical Helix** command. MGRID will prompt you to enter the parameters for the cylindrical helix. Enter the following parameters:

Wire Parameters:
Number of Segments for Circle = 2
Radius = 30 mils
Cylindrical Helix Parameters:
Segments each Turn = 12
Total Segments = 48
Helix Length = 600 mils
Radius = 200 mils

Start Vertical Wire Length = 150 mils
End Vertical Wire Length = 3500 mils
Start Angle = 0
Center X-Coordinate = 400 mils
Center Y-Coordinate = 400 mils
Center Z-Coordinate = 5010 mils

Select **OK** to continue. The cylindrical helix will be built. The feed proximity is shown in Figure 11.9. The bottom end of the helix is at Z = 5010 mils. It is 10-mils above the handset top cover, which is not built.



(a) The helix is built 10-mils above the handset (b) The cover is built (c) The connection is built

Figure 11.9 The helix is built 10-mils above the handset without cover.

Step 6 Select **Edit->Select Vertices** command. Try to find out the layer with z = 5010 mils on the layer window. Un-check all layers. Then, check the Z = 5010 layer to focus the selection on it.

Step 7 Window the whole structure. The 2 vertices at the start of the monopole at z = 5010 mils are selected.

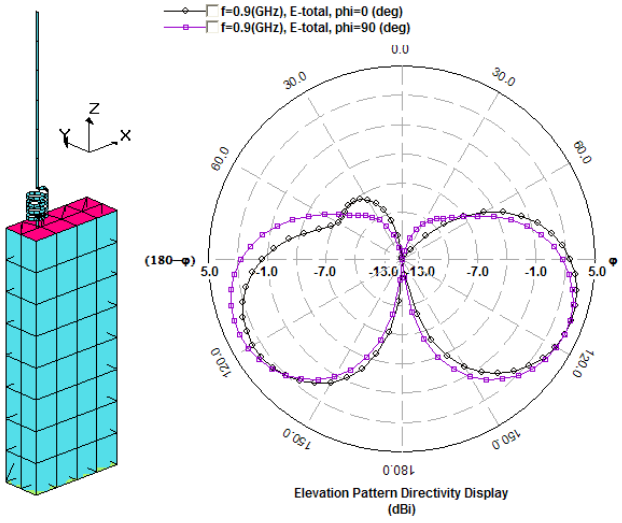


Figure 11.10 The meshed handset and the radiation pattern.

Step 8 Select **Edit->Add Via on Edges** command. Enter **End Z-Coordinate** as 5000 mils. A vertical rectangle should be built from Z = 5010 to Z = 5000. It will automatically create the connection at the top cover.

The vertical rectangle is at the start of the monopole from z = 5000 to 5010 mils. We will define it as the **Vertical Localized** port for differential feed.

Step 9 Select **Port->Port for Edge Group** in **command**. Select the **Vertical Localized** scheme. Enter 5010 mils for the **Positive Level** and 5000 mils for the **Negative Level**. Window the whole structure. Port 1 will be defined at the start of the monopole. Select **Port->Exit Port** command.

Step 15 Save the structure as **c:\ie3d\practice\handset.geo**. Simulate the structure and save the result in **c:\ie3d\practice\handset.sp**. It takes a fraction of a minute to simulate it. The meshed structure and the radiation pattern at 0.9 GHz are shown in Figure 11.10. Again, we would like you to understand that this is not a design. The dimensions are arbitrarily chosen and they are not optimized. We just demonstrate the concepts and approaches to model complicated structures.

Section 11.5 Modeling of Inverted F Antennas

Inverted F antennas or similar antennas are frequently used in RF applications due to their compact sizes. In this section, we will demonstrate how you can model an inverted F antenna. Shown in Figure 11.11 is an inverted F antenna in **c:\ie3d\samples\invf.geo**. The s-parameter result is saved into **c:\ie3d\samples\output\invf.sp**. Basically, it is a horizontal rectangle of length 2520 mils and width 900 mils. The horizontal rectangle is 700 mils above the infinite ground plane. On the left end, there is a via connecting the horizontal rectangle to the ground plane. The via connected to the port is of width 60 mils. The via for the port is 990 mils from the via on the edge and it is 180 mils from the closest edge,

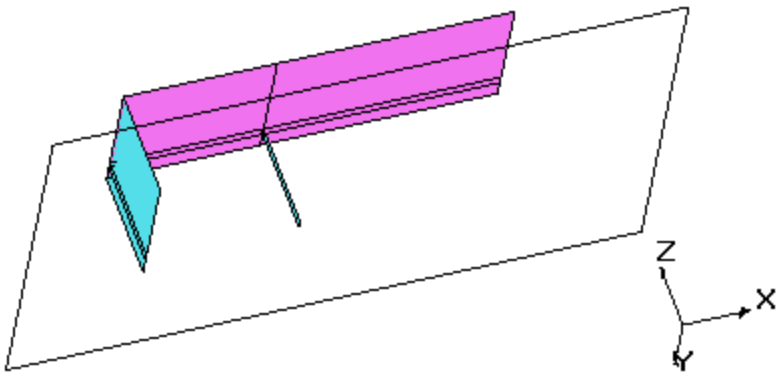


Figure 11.11 An inverted F antenna on an infinite ground (shown as a wire frame).

In practical application, 60-mil thick aluminum is normally used to make the antennas. The thickness of 60-mils is quite thick and we should model it with thickness in order to get high accuracy result. Also, the ground plane for the antenna may be small and we need to model the finite ground plane. The inverted F antenna with finite ground plane is saved in **c:\ie3d\practice\invf1.geo**. The finite ground plane is 500-mil larger than the antenna on each side. The finite ground plane and finite thick metallic strip model is saved in **c:\ie3d\practice\invf2.geo**.

The comparison of the 3 models is shown in Figure 11.12. The invf.geo is the "Infinite GND" model. The invf1.geo is the "Finite GND" model. The invf2.geo is the "Finite TK" model. The "Infinite GND model is quite different from the other two models. Inverted-F antenna is not a low profile antenna. The ground plane size does have significant effect to the performance of the antenna.

The locus of the "Finite GND" model is similar to the one for "Finite TK" model. However, the resonant frequency shifts a few percents. Certainly, 60-mil thick strips are quite significant. We cannot neglect it. We will not show how to build the models here. The examples serve to demonstrate the importance of the ground plane and finite thickness for non-low-profile antennas.

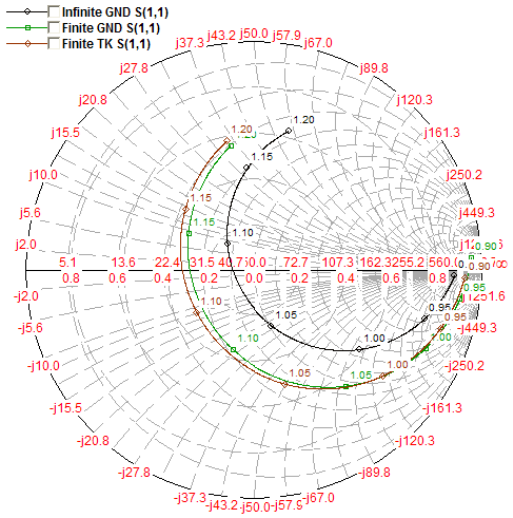


Figure 11.12 The comparison of different models on the inverted-F antenna.

Section 11.6 Simulations of RFID Antennas

RFID has been a hot topic in the last few years. IE3D is excellent for RFID applications due to its flexibility and high accuracy. There are normally two frequency bands for RFID applications: (1) Spiral-like antennas working at 13.56 MHz; (2) Half-wavelength long antennas working at about 868 MHz.

The design of RFID antennas should be similar to other types of antennas except there are special requirements for the input impedance for an RFID antenna. Normally, the antenna is connected to a RFID chip. The chip's internal impedance is of a complex value at the target frequency. To receive maximum efficiency, we should try to design an antenna such that the impedance of the antenna is a conjugate match to the chip internal impedance.

IE3D has excellent built-in optimizers. You are allowed to define some dimensions of an antenna as optimization variables and let IE3D to optimize them for specified goals. For RFID applications, you can specify the optimization goals as: Re[Z(1,1)] = Ri and Im[Z(1,1)] = - Xi, where Zi = Ri + j Xi is the internal impedance of the RFID chip at the target frequency.

On IE3D, we have defined two efficiency values for an antenna: (1) Radiation Efficiency; (2) Antenna Efficiency. The definitions can be found in Appendix AV. We also have defined Antenna Gain based upon the definition of Antenna Efficiency. The Antenna Efficiency and Antenna Gain are more defined for constant wave sources.

For RFID applications, we have defined Conjugate Match Factor for judging how good an RFID antenna is. Please read the Appendix BI for more discussion on RFID designs.

Section 11.7 Simulation of Ultra-Wide Band (UWB) Antennas

IE3D is an excellent tool for designing various antennas. We have implemented advanced features for UWB applications. In this section, we will not discuss how we can achieve wide band. We will just demonstrate how we can use IE3D and MDSPICE to perform simulation in both frequency and time domain.

Saved in .\ie3d\samples\oval_dipole.geo is a printed dipole antenna with arms of elliptical shapes. We would like to investigate how the waveform looks like if we have excite one Tx antenna and receive the

signal from an RX antenna. For the oval_dipole.geo, we have defined one port and a plane wave excitation for it.

Please open the file. Save it into .\ie3d\practice\oval_dipole.geo (see Figure 11.13). Simulate it at 0.1, 0.2, 0.5, 1.0, 1.5, 2.0, … 14.5 and 15 GHz. Please enable pattern calculation with the default setting (only the port is excited while the plane waves do not present). It will take a few minutes to finish because pattern calculation will take some time. Actually, we are not just interested in the radiation pattern with specified excitation. What is of interests to us is the General Pattern File (.mpa) created in the simulation. As it is discussed in Appendix AJ, we can find the 2-port s-parameters from the .mpa file of an antenna with 1-port and plane wave excitations defined simultaneously.

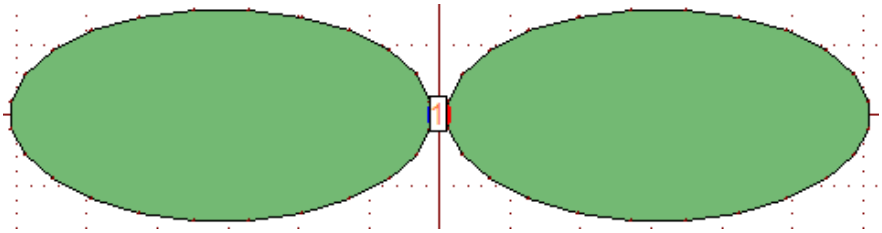


Figure 11.13 A printed oval shape dipole antenna.

Assume we would like to find the 2-port s-parameters of the two antennas with a separation of 3 feet. We can do it in the following.

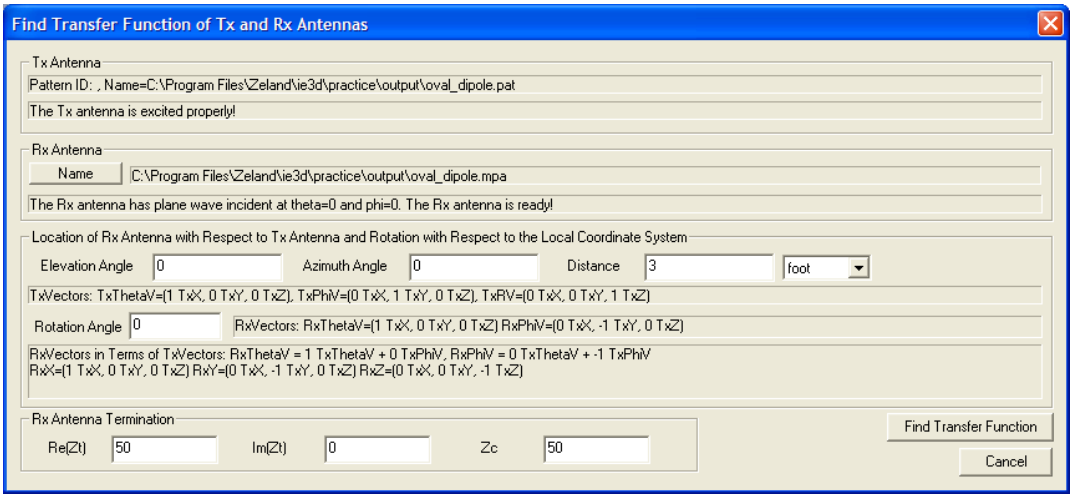


Figure 11.14 The definition of the Tx and Rx antenna's relative location.

After the IE3D simulation, the oval_dipole.mpa file should be accessible on MGRID for post-processing. You can also add the file oval_dipole.mpa file or the oval_dipole.pat file into the pattern list PATTERNVIEW for post processing. Please click at the oval_dipole.pat file in the list on PATTERNVIEW. Then, select Edit->Find TxRx Transfer Function. PATTERNVIEW will prompt you for the General Pattern File (.mpa). Select the .\ie3d\practice\oval_dipole.mpa file. PATTERNVIEW will load the file and prompt you to define the relative location between the Tx and the Rx antenna. Please define the parameters as shown in Figure 11.14. The dialog basically defines that the Tx and Rx antennas are facing each other with a distance of 3 feet. Please select Find Transfer Function button. PATTERNVIEW will prompt you for the file name. Please select OK to accept the default setting of TX_RX_0.TXT file name. PATTERNVIEW will prompt you for the Save Transfer Function Data dialog. You have the option to find different transfer functions and the 2-port s-parameter file. Please select 2-Port S-Parameters for TxRx.

PATTERNVIEW will automatically update the file name s: TX_RX_0.SP. Select OK. PATTERNVIEW will save the s-parameters into TX_RX_0.SP and open it on Notepad.

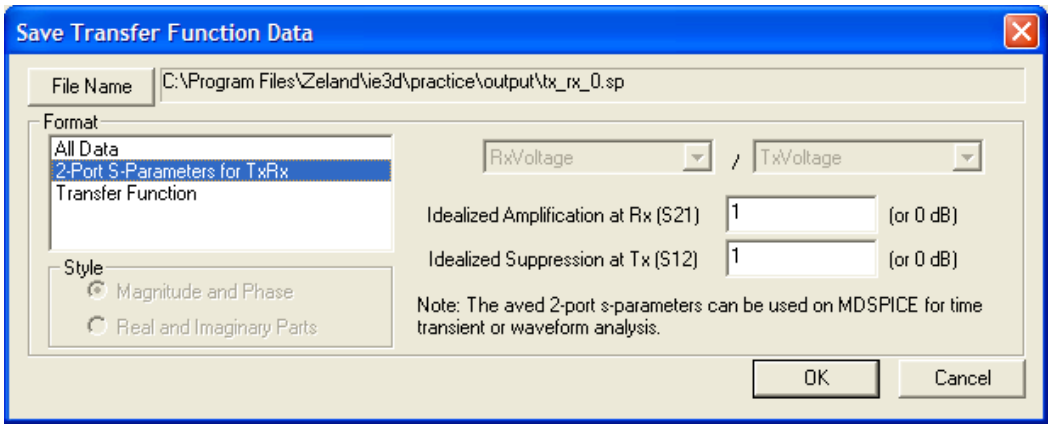


Figure 11.15 The Save Transfer Function Data dialog.

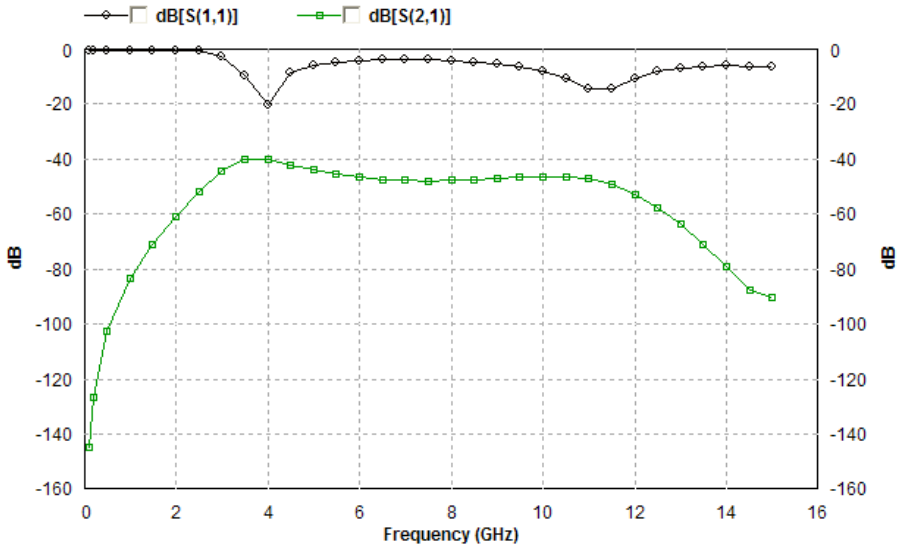
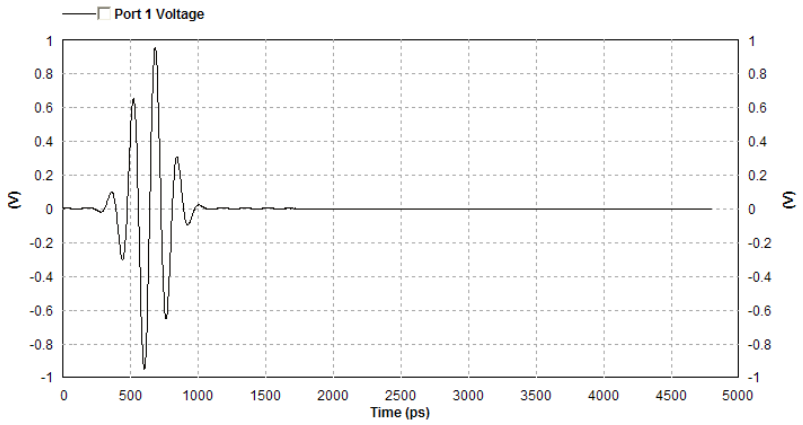


Figure 11.16 The 2-port s-parameters between the Tx and Rx antenna.

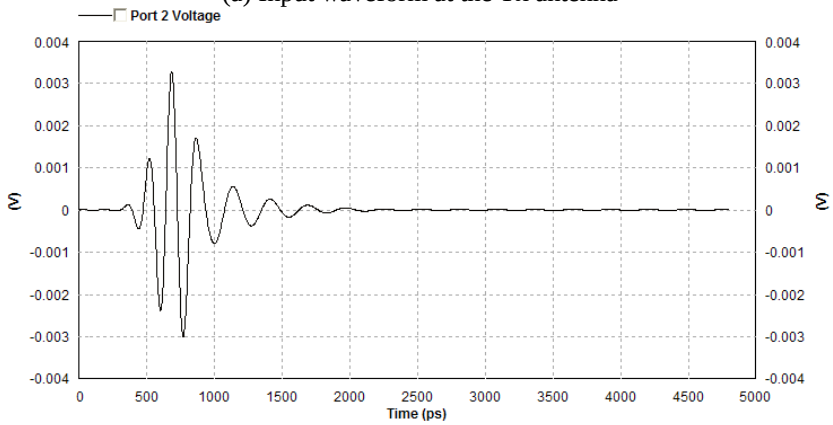
Please display the 2-port s-parameters saved in .\ie3d\practice\output\TX_RX_0.SP using MODUA. As you can see, the coupling between the 2-antennas is below -40 dB when they are separated by 3 feet. Please understand that the results are based upon the far-field radiation pattern. It should be reliable since the 2 antennas are far enough at the passband which is above 3 GHz.

We can setup MDSPICE to perform a time transient analysis on the TxRx transmission system. We try to put a stimulus of sine modulated with Gaussian centered at 8 GHz with 3 dB bandwidth of 1 GHz at the Tx antenna and observe the waveform at the Rx antenna with a 50-ohm termination. Figure 11.17 shows the comparison between the input waveform at the Tx antenna and the output waveform at the Rx antenna. We certainly should see some difference in the waveform because the transmission properties of the system are not flat over the frequency range of interest. We also see some time delay and ringing in the output waveform. Figure 11.18 shows the comparison in the spectrum between the source waveform and the output voltage waveform. As you can see, the difference between the 2 waveforms is at about 3.8 GHz and the result is consistent with the 2-port s-parameters.

IE3D is the most powerful and versatile antenna design tool. You can use it to do much more that we demonstrate here. We will end this chapter here.



(a) Input waveform at the Tx antenna



(b) Output waveform at the Rx antenna.

Figure 11.17 The Input Waveform at Tx antenna and Output Waveform at Rx antenna.

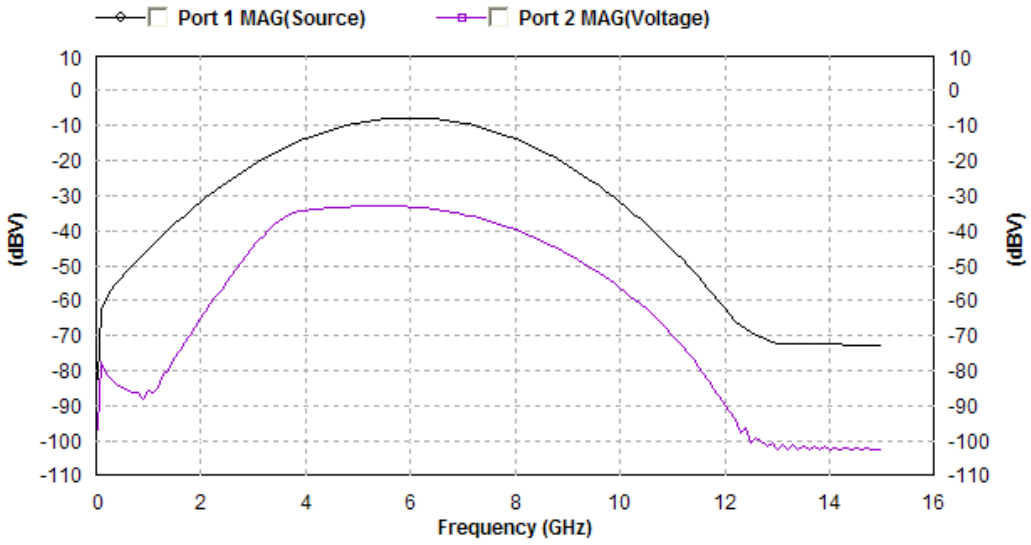


Figure 11.18 The spectrums of the source waveform and the output waveform.

Section 11.8 Parameterization of Complicated Structures

The elliptically shaped modified dipole exhibits excellent wide band characteristics. One question you may have is: Can we change the dimensions elliptical shapes. On MGRID, you may have to rebuild them. However, on IE3DLIBRARY, you are able to parameterize even more complicated structures. Saved in .\ie3d\ie3dlibrary_examples\ep4.ie3d is a more complicated structure on IE3DLIBRARY. Please use IE3DLIBRARY to open it and it is illustrated in Figure 11.19.

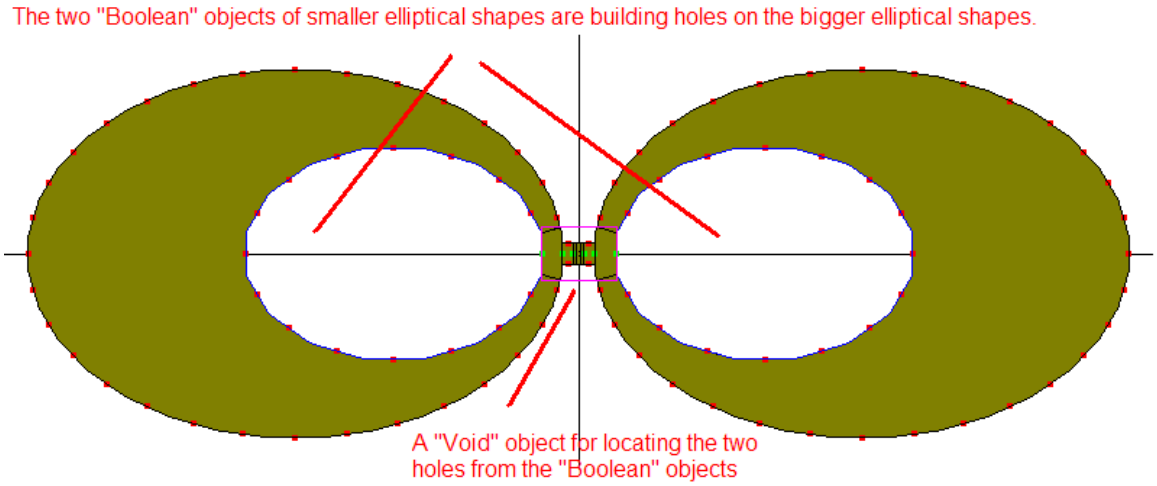


Figure 11.19 A tunable UWB antenna created using 4-elliptical shapes.

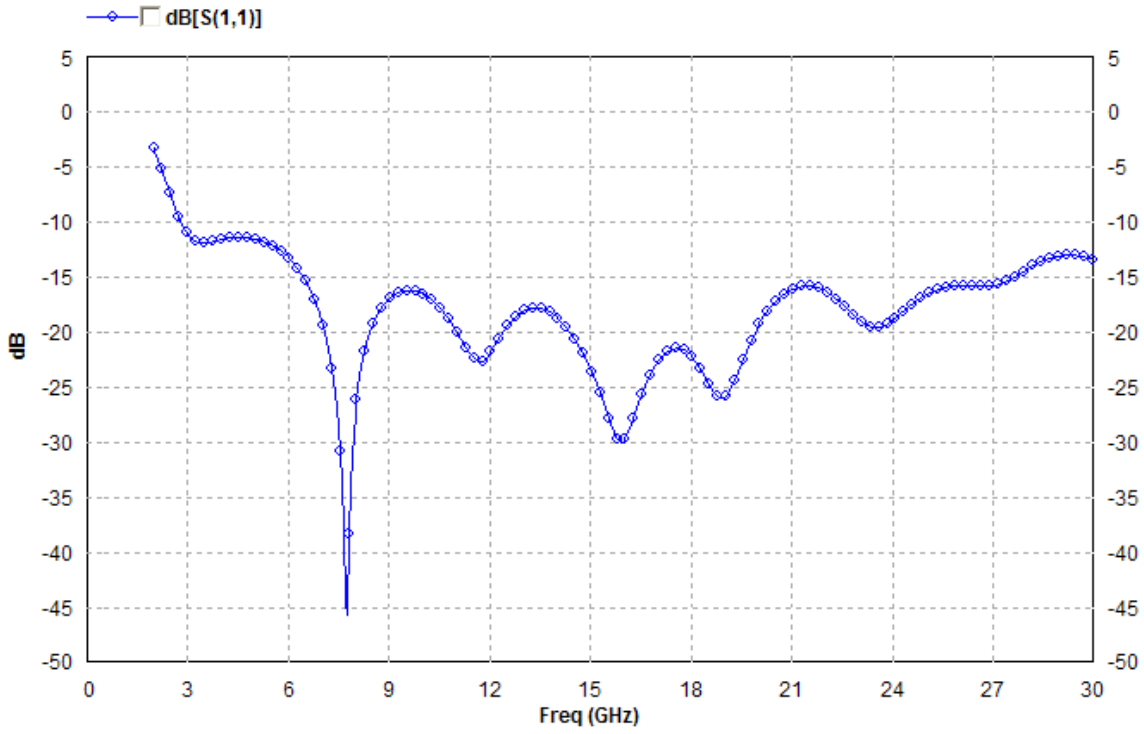


Figure 11.20 The frequency response of the UWB antenna.

The antenna is completely parameterized. It is created mainly from 4-elliptical shapes. The two smaller elliptical shapes are "Boolean" objects. They are building the holes on the larger elliptical shapes. The symmetry of the structure is guaranteed by connecting the two "Boolean" objects to a "Void" object. The "Void" object has shape but its shape will not be used for the simulation but just for locating other objects only. All the dimensions are parameterized and you can change the major and minor axis dimensions of both the larger and smaller elliptical shapes.

Creating of the parameterized structure is documented in the IE3D Library User's Manual (see .\ie3d\manual\ie3dlibrary_manual.pdf). You can read the manual and learn how to create parameterized IE3D geometry on IE3DLIBRARY.

The structure is FastEM prepared. You can select Process->Full-Wave EM Design Using Fast EM Design Kit command to do real-time EM tuning and optimization on it. The antenna yields the best results with normalization impedance of 120-ohms. We tried to optimize it for the goal of $|S(1,1)| < -10.75$ dB from 3.1 to 10.6 GHz with Zc = 120-ohms. It achieves the goal in seconds. The optimized geometry is saved in: .\ie3d\ie3dlibrary_examples\ep4_fastem.ie3 (for IE3DLibrary) and ep4_fastem.geo (for MGRID).

The example is arbitrarily chosen to demonstrate the FastEM feature of IE3D and it turned out to have extreme wide bandwidth. It is simulated from 2 to 30 GHz and the 10-dB bandwidth on Zc = 120-ohms from below 3 GHz to more than 30 GHz (see Figure 11.20).

Section 11.9 Modeling for Complicated Wire and Patch Structures with Connections

In Section 3, we have demonstrated how we can use the command Adv Edit->Build Wire Path command to build a general shaped wire structure along a series of vertices. You are allowed to build a port at every vertex of the wire path. For example, we may have a loaded wire antenna with lumped elements between segments. We are able to replace the lumped elements as ports on IE3D while we can perform mixed EM and circuit co-simulation and optimization on MODUA. We will build the wire antenna with many ports (see Figure 11.21). The wire structure with many ports in Figure 11.21 can be build in one shot using the Adv Edit->Build Wire Path, possibly with the importing vertices with the command Adv Edit->Create and Edit Vertices.

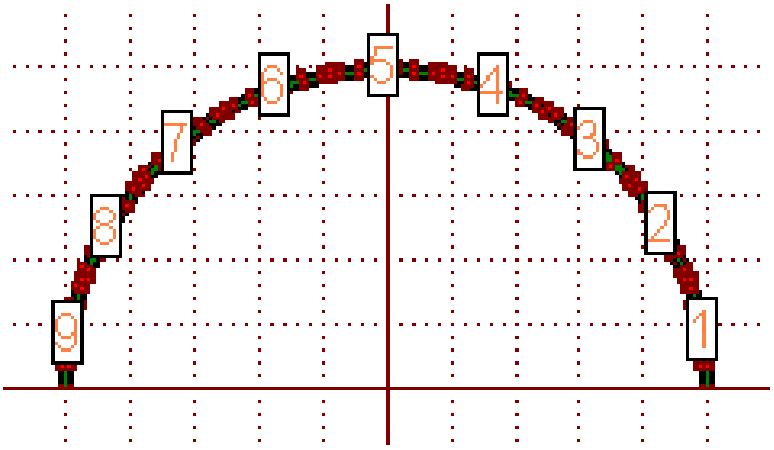
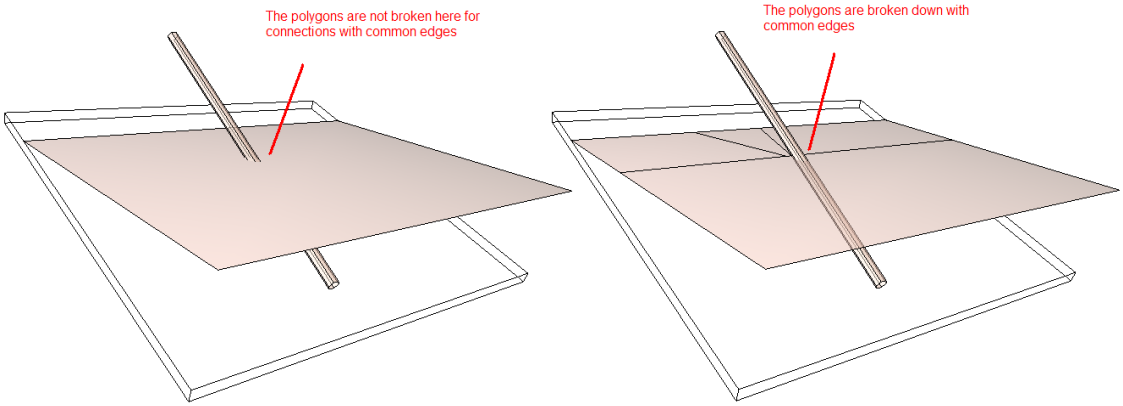


Figure 11.21 A Wire antenna with many ports for lumped element connections.

In reality, wire antennas can be more complicated. For example, a Yagi antenna may consist of multiple wire segments. We are not able to build the antenna as one single wire. We have to build them as multiple connected wires. Also, an antenna may contain plates and wires connected together. Before IE3D 14, building connections between wires and 3D polygons is not easy. We may be able to build multiple wires or 3D polygons in the same location and they may appear to be connected because they are physically overlapping each other (see Figure 11.22a). However, they are not considered to be physically connected by

IE3D due to the definition of connections on common edges (please refer to Chapter 4). As shown in Figure 11.22a, the polygons are not connected because there are no common edges between them. On IE3D V14, we have implemented some advance geometry modeling technology into the command Adv Edit->Connections->Adjust Geometry for Connection. The command is able to break 3D polygons and create the connections automatically for overlapped 3D polygons (see Figure 11.22). At the initial release, we would like to keep it as a manual command. You still need to select the command for the connection. If you don't select it, we will not build the connections for you. From our extensive testing, the command is quite reliable. Eventually, this command will be fully automated in IE3D's geometry modeling. Using this command, you are able to connect multiple wire segments or 3D polygons with guaranteed electrical connections easily.



(a) 3D polygons including wire not connected (b) 3D polygons connected on common edges.

Figure 11.22 3D polygon and wire connections on IE3D 14.

Chapter 12 Accuracy and Efficiency Enhancement

The IE3D employs an integral equation, method of moment formulation. Integral equation formulation itself is not an approximate method. However, the Galerkin's procedure and the finite truncation of the basis functions introduce approximation. Basically, we use finite dimensional functions to approximate the infinite dimensional current distribution on a circuit. Because it is a numerical procedure, many factors involved in the numerical calculation are affecting the accuracy of a simulation. Following are some of the major factors may affect an IE3D simulation.

(1). Meshing Frequency (or Fmax) ;
(2). Meshing Cell Size (Cells per Waveguide Wavelength, Ncell);
(3). De-embedding Schemes;
(4). The Number of Cells in De-embedding Region (De-embedding Cells);
(5). The Meshing Schemes (With or Without Edge Cells).
(6). The number of images used in the enclosed or periodic Green's functions, which we will discuss in the next chapters.

Usually, more cells in a simulation yield higher accuracy. Larger **Meshing Frequency (Fmax)** or **Cells per Wavelength (Ncells)** means more cells and means more accurate result. However, we cannot just try to increase the **Fmax** or **Ncells** indefinitely because we will run out of memory very soon. If we use the cells smart enough, we can achieve high accuracy with small number of cells.

In this chapter, we will concentrate on how we can improve the simulation accuracy by smartly constructing a structure.

Section 12.1 Adding Edge Vertices to Control Meshing

When we discuss our first example in Chapter 3, we have demonstrated how we can use Automatic Edge Cells (AEC) to improve simulation accuracy. Edge cells are certainly very critical in capturing the edge effects of microstrip structures, and use of edge cells can improve the simulation accuracy significantly especially for strongly coupled structures. We can claim that AEC guarantees high accuracy for novice users. In the IE3D 11, we define AEC based upon AEC Ratio or the ratio between the edge cell size and the regular cell size. In this way, it significantly reduces the chance for a user to use too small or too large cells accidentally because the edge cell width is automatically scaled with the meshing frequency.

In the IE3D 11, we also implemented the multi-layer AEC and AEC for thickness traces. Multi-layer AEC allows a user to define multiple edge cells along the edges to further improve the accuracy. It can yield extreme modeling accuracy for high accuracy requirements such as designing 40-dB directional couplers. AEC for thickness traces further improve the accuracy for modeling traces with true thickness, which normally do not need AEC unless extreme high accuracy is required.

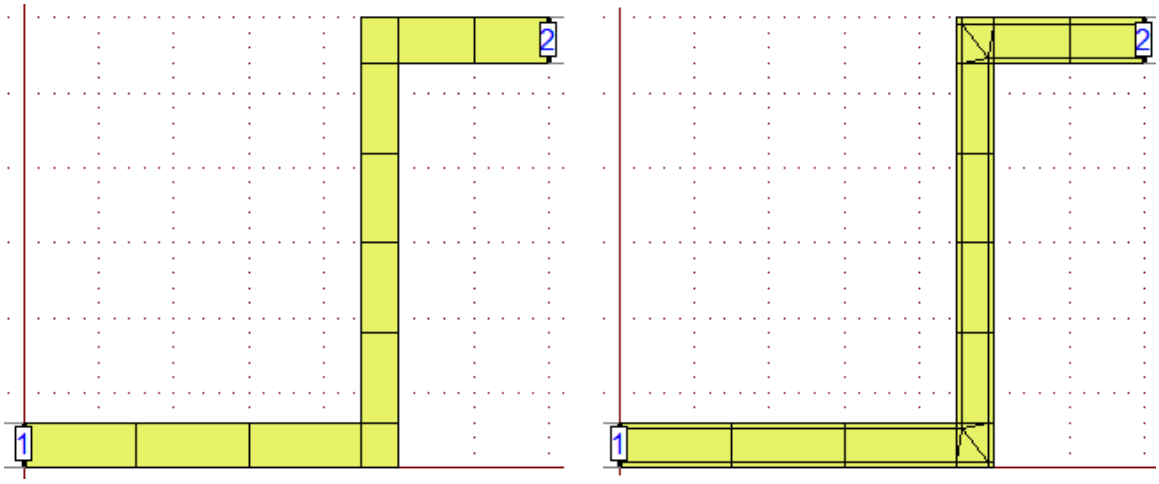
AEC is certainly very good for accuracy. For small and medium size problems, we normally should enable AEC. It will yield high accuracy without extra geometry modeling effort. However, AEC normally increases the problem size (number of unknowns) significantly. For a large structure, enabling AEC and without enabling AEC may make a big difference in the simulation time. It would be nice if we can get high accuracy results while we may not need to use edge cells everywhere. For some parts of a structure, we may get high accuracy results without edge cells. Normally, if two edges are not very close, edge cells may not be very important. If a trace is not a critical path for a transmission and the related S(i,i), as a small quantity, is not a very critical parameter, we can also omit the edge cells on it.

For experienced users who know about where edge cells might or might not be critical can disable AEC while they can create the edge cells by inserting vertices to control the meshing. We will demonstrate some examples on manually inserting vertices to control the meshing.

Our first example is the suspended strip line structure in **.\ie3d\samples\one_cell.geo**. It is the same one as the **.\ie3d\samples\connect.geo**. We have used this example to demonstrate the importance of polygon connection. We will use this example to demonstrate how we can insert vertices to control the meshing program to create edge cells.

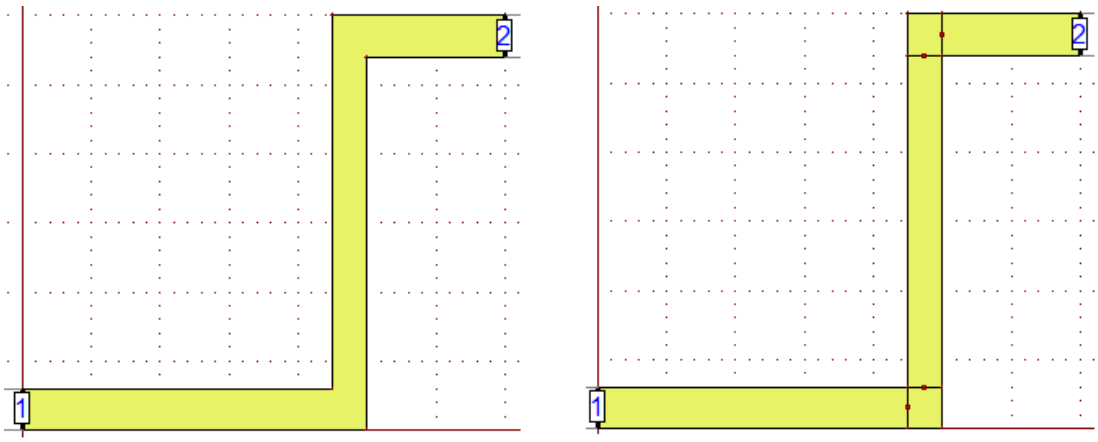
Step 1 Run MGRID. Open file **.\ie3d\samples\one_cell.geo**. Save it into **.\ie3d\practice\one_cell.geo**.

Step 2 Select **Process->Display Meshing** command. Try to do it with AEC disabled and AEC enabled with AEC Ratio = 0.05. The meshing results are shown in Figure 12.1.



(a) AEC disabled (b) AEC enabled with AEC Ratio = 0.05

Figure 12.1 The meshed structure of **one_cell.geo**.



(a) Before Rectanglization (b) After Rectanglization

Figure 12.2 The structure before and after Rectanglization.

Step 3 We are going to modify the geometry to force it to be meshed into 3-cells in the transverse direction. The command we are using is the Edit->Add Edge Vertices command. However, before we apply this command, we would like to rectanglize the structure first because the Edit->Add Edge Vertices can only control the meshing of rectangles effectively.

Please select **Adv Edit->Rectanglization** command. The structure is divided into rectangles only (see Figure 12.2).

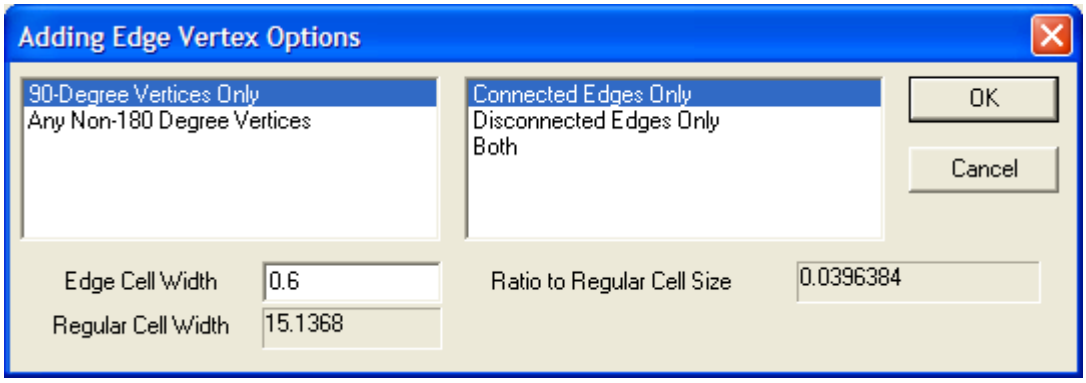
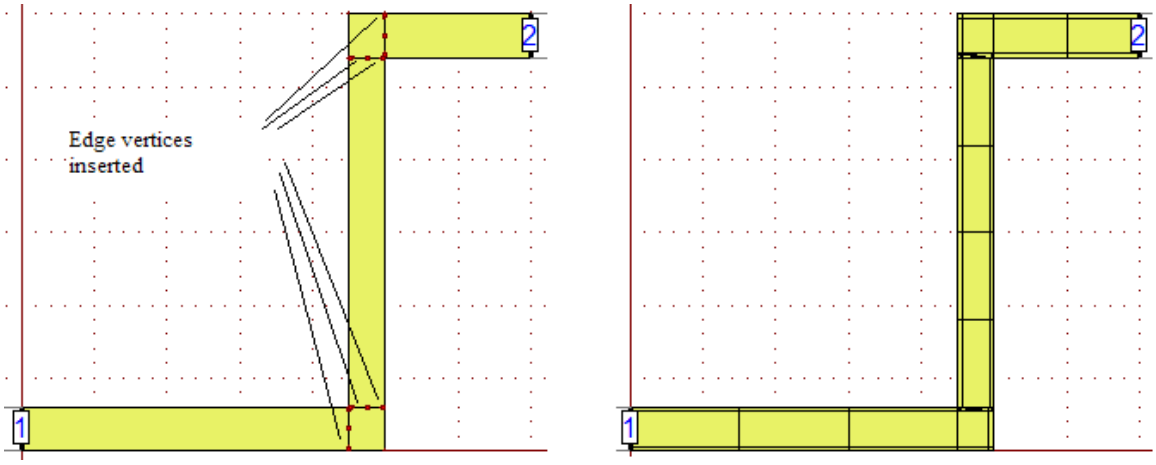


Figure 12.3 The Add Edge Vertex dialog.



(a) Structure with inserted vertices (b) Meshed structure without AEC.

Figure 12.4 The structure with edge vertices inserted.

Table 12.1 The different options in Add Edge Vertex dialog.

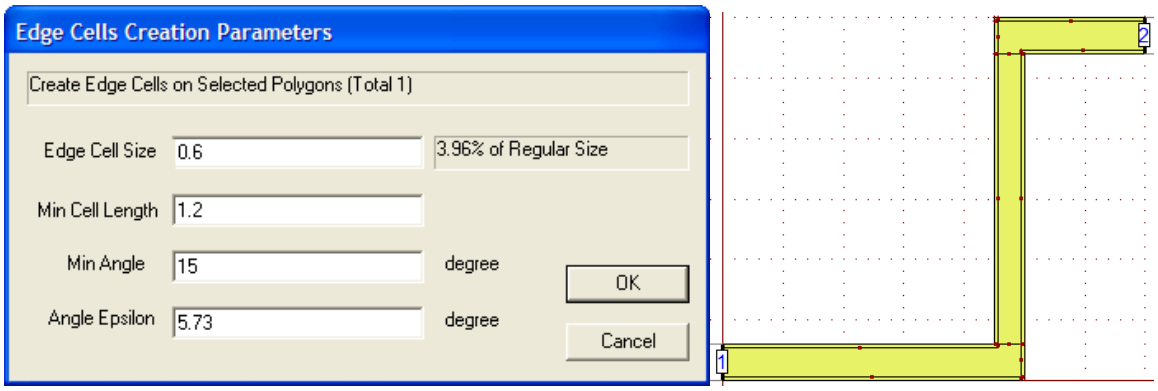
Option	Description
90-Degree Vertices Only	Insert vertices on the edges forming a right angle only.
Any Non-180 Degree Vertices	Insert vertices on two edges when they do not form a 180-degree angle.
Connected Edge Only	Insert vertices on those edges are common to multiple polygons or with ports defined on.
Disconnected Edges Only	Insert vertices on those edges are open edges only.

Step 4 Please select all the polygons in Figure 12.2b. Please select Edit->Add Edge Vertex command. The dialog comes up. Please enter the Edge Cell Width = 0.6. You will get the dialog shown in Figure 12.3. There are different options and they are documented in Table 12.1. Please select OK to continue. We will get the structure shown in Figure 12.4a. It is very close to the one shown in Figure 12.2b. However, there are some inserted vertices on the connected edges. The inserted edge vertices can control the meshing program to create edge cells in meshing even without AEC (see Figure 12.4b). Please save the file as: .\ie3d\practice\inserted_vertices_connected.geo.

Regarding to the two conditions in Figure 12.3, MGRID only adds vertices on those edges meeting the two conditions simultaneously. In fact, only the vertices added with the default settings in Figure 12.3 can effectively control the edge meshings. If you choose other options, you can still add vertices on the edges. However, the inserted vertices can not control IE3D to create edge vertices effectively.

Basically, the Edit->Add Edge Vertices command can only be effective in control the edge meshings for rectangular shaped structures only. It will not be useful for non-rectangular shaped polygons. What can we do for those non-rectangular shaped polygons? We certainly can use AEC for it. However, we have also implemented the command Adv Edit->Create Edge Cells for such a situation.

Step 5 Open .\ie3d\practice\one_cell.geo again. This time, please do not do the Adv Edit-> Rectanglization. Please select the polygon. Select Adv Edit->Create Edge Cells. Change the Edge Cell Size to 0.6 (see Figure 12.5a). Select OK and we will get the structure in Figure 12.5b. The edge cells are created. IE3D will mesh the structure with small cells along the edges no matter whether we enable AEC or not.



(a) The Create Edge Cells dialog (b) The structure with edge cells created.

Figure 12.5 The dialog and result for Creating Edge Cells.

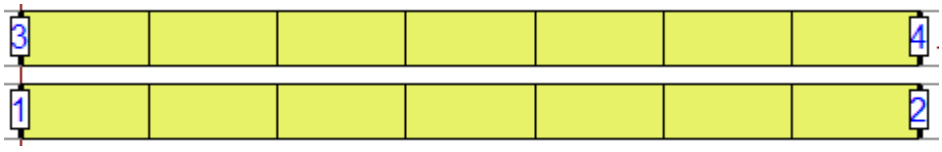
You may ask whether there is any advantage in creating the edge cells on MGRID instead of using AEC in simulation. Actually, there is no advantage if we take the resulting structure as it is. However, you can merge some of the created edge polygons with the interior polygons. You only leave those important edge polygons there. Then, you disable AEC in simulation. In this way, you can control where you want or don't want to have edge cells.

Section 12.2 Always Use Edge Cells for Closely Coupled Structures

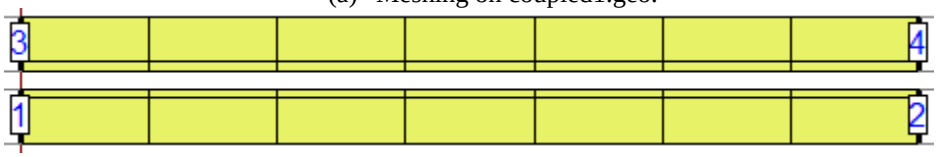
For strongly coupled structures, we really need edge cells for accurate modeling of the coupling effects. An example is the geometry saved in **.\ie3d\samples\coupled1.geo** and the one saved in **.\ie3d\samples\coupled3.geo** (see Figure 12.6). The two files describe the same geometry except we enforce AEC with AEC Ratio = 0.06 in **coupled3.geo**. It corresponds to AEC Width = 0.91 mils. Why do we choose AEC Ratio = 0.06 instead of the default value of 0.1. The reason is that the gap between the two strips is 2 mils. We would like to make the edge cell size smaller than 50% of the gap width. Normally, it is ok if the edge cell size is smaller than the gap width even it is better to be a little bit smaller.

Another structure is the one saved in .\ie3d\samples\coupled2.geo. It describes the same geometry except each strip is divided into 2 polygons (see Figure 12.7). Some narrow polygons of width of about 1 mil are created along the coupled edges. They are created using the Adv Edit->Create Edge Cells command.

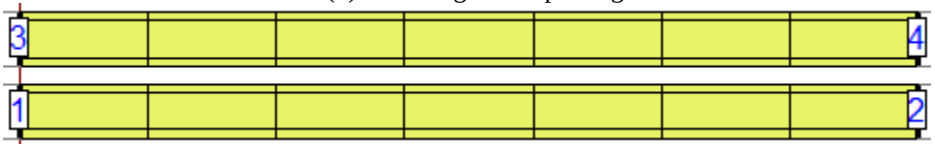
However, the divided polygons on the non-coupled edges are merged back to the bigger polygons after the process by using the command Adv Edit->Merge Selected Polygons.



(a) Meshing on coupled1.geo.



(b) Meshing on coupled2.geo.



(c) Meshing on coupled3.geo.

Figure 12.6 Comparison of meshing for different sfiles.

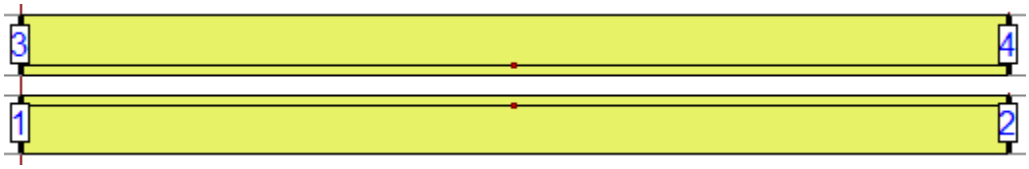


Figure 12.7 The coupled2.geo with narrow polygons along the coupled edges.

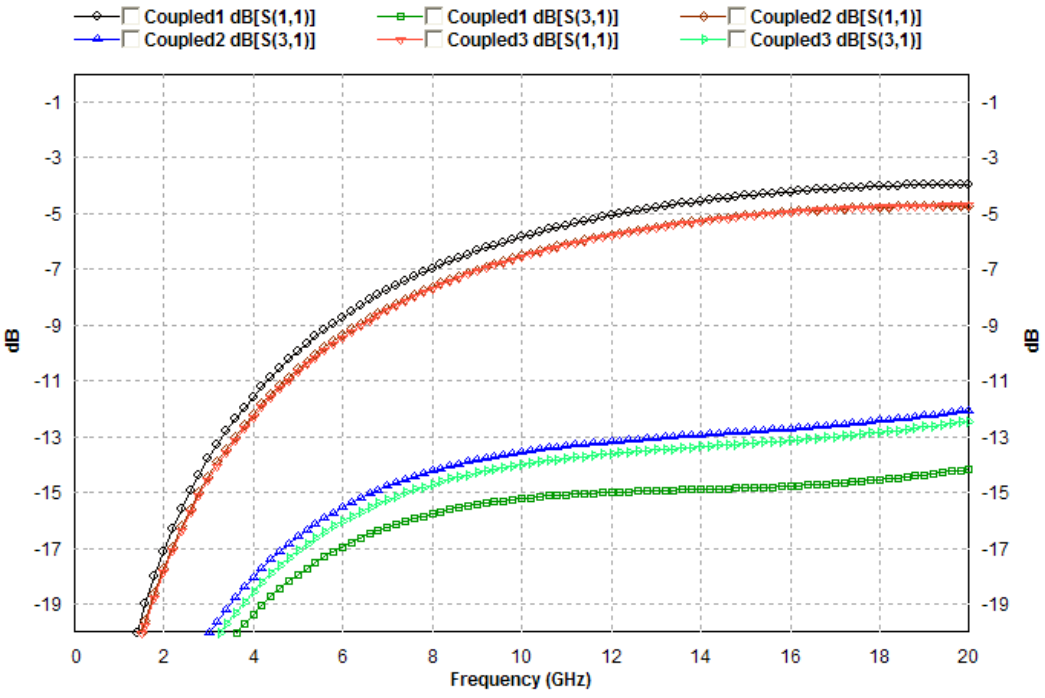


Figure 12.8 The comparison among the 3 models.

The comparison in the s-parameters between different meshings is shown in Figure 12.8. We just show the S(1,1) and S(3,1) which are some of the most affected parameters by edge cells. As you can see, the coupled2.geo (edge cells for coupled edges) and coupled3.geo (edge cells on both edges) yield very close result. There is about 0.4 dB difference out of -13 dB at 12 GHz. However, we can see about 1.4 dB difference out of -13 dB between coupled1.geo (no edge cells) and coupled3.geo. Apparently, without edge cells, the tight coupling can be under estimated. By adding the edge cells along the tightly coupled edges, we normally can capture the coupling while we do not require so many cells as using the AEC. For this simple structure, we see little difference in the simulation time. However, for large structures, applying edge cells along closely coupled edges may reduce the simulation time significantly (compared to AEC) while we can still capture the coupling precisely.

Section 12.3 Accurate Modeling of Thickness Effects

We can define metallic thickness in the **Param->Basic Parameters** command. The metallic strip thickness defined is used to correct the loss effect of the metallic strip only. In the actual simulation, we still assume the metal is infinitely thin. When the metallic strip thickness is not very small compared to the width, the thickness is affecting the transmission line properties very much and we need to model the actual thickness structure. We will denote the thickness over width ratio as t/w ratio in the following discussion.

Naturally, you may have the following question. What are the criteria for small t/w ratio? The criteria are different for different types of structures. In the following section, we will investigate how thickness is affecting transmission line properties.

Our first example is a microstrip line. The parameters of the microstrip line are:

Substrate Dielectric Constant = 12.9
Substrate Loss Tangent = 0.0005
Substrate Thickness = 100 mm
Metallic Strip Thickness = 4.5 mm
Metallic Strip Width = 75 mm
Metallic Bulk Conductivity = 4.9e+7 s/m

For this structure, the t/w ratio is 6%. We construct the structures using four different ways. The four geometry files are saved in the **.\ie3d\samples** directory.

nthick1.geo --- The finite-thickness metallic strip is modeled as infinitely thin strip and the loss effect is corrected using the surface impedance. Edge vertices are inserted to enhance accuracy.

nthick2.geo --- The finite-thickness metallic strip is modeled as two infinitely thin parallel strips. One is on the bottom surface and the other is on the top surface. AEC is applied to enhance accuracy. We have not discussed how we define multiple positive ports for the same port index. Basically, we need to use the Port->Multiple Positive Port to define ports of the same positive index. We can also use the Port->Change Port Order command to change port indices after port are defined. You can refer to Chapter 2 on discussion of this menu item.

wthick1.geo --- The finite-thickness metallic strip is modeled as 4 surfaces. One is on top and another is on the bottom. There are two vertical plates on the edges. There are not inserted vertices near the edge.

wthick2.geo --- Same as **wthick1.geo** except we apply AEC on thickness edges for even higher accuracy.

The four different geometry modeling schemes are shown in Figure 12.9. The four structures have been simulated and the corresponding s-parameter files are saved in the **c:\ie3d\samples** directory. We can simulate them and use the Process->Find TLN Parameters command on MODUA to extract the transmission line parameters. A comparison at 10 GHz among the four structures is shown in Table 12.2. We use the **wthick2.geo** as the standard. The Re(Zc) is about 3% higher for the **nthick1.geo**, 1.5% higher for the **nthick2.geo** and 1% higher for the **wthick1.geo**. For the waveguide wavelength λ, **nthick1.geo** is about 2% shorter, **nthick2.geo** is about 0.8% shorter and **wthick1.geo** is about 1% longer. As it is the nthick2.geo is not bad for the single strip. The advantage of the 4-plate model is more obvious for strongly coupled structure. The reason is that the edge walls are a major contribution to the coupling between closely coupled transmission lines.

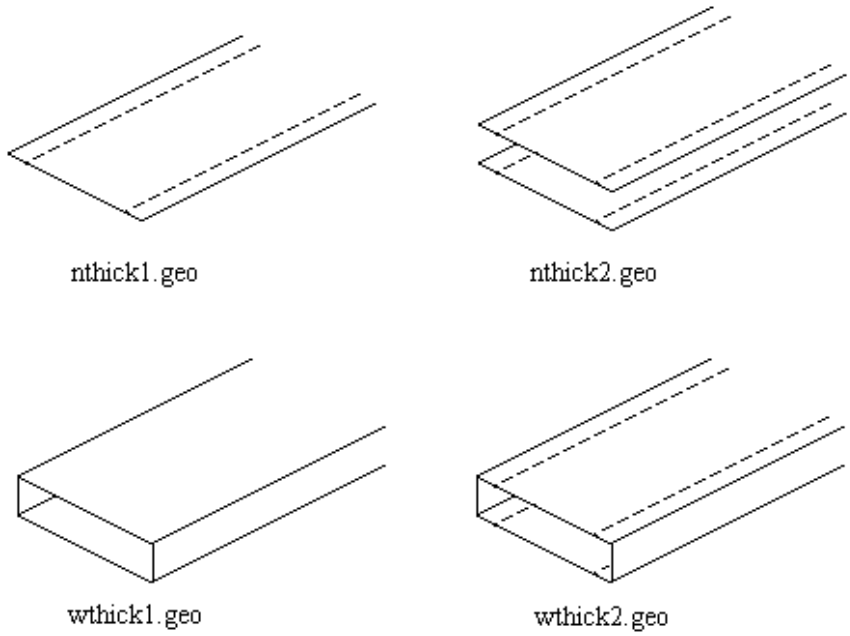


Figure 12.9 The differences between the 4 geometry files. The dash lines represent the controlled mesh near the edges.

From our experience, 4-plate approximation can improve the simulation accuracy of thick and closely coupled structures significantly. When we build the 4-plates for the thickness, we usually don't need to insert edge vertices because the cells on the edge walls will model the singular current density at the edge quite accurately. There are exceptions. For some thin structures requiring the modeling the thickness, we should use the 4-plate approximation and also the edge cells to enhance the accuracy. Also, when the substrate dielectric constant is close to 1, the thickness effect is more significant because the charge is less concentrated on the bottom surface.

You should also understand that characteristic impedance and waveguide wavelength are usually very sensitive. S-parameters are usually less sensitive. That is why we can get quite accurate result even using 1 to 3 cells in the transverse direction and without growing the thickness for many structures.

Table 12.2 Comparison of the 4 modeling schemes at 10 GHz.

F=10 GHz	Re(Zc) Ω	Im(Zc) Ω	λ (mm)	α (dB/mm)	Re(ε_{reff})	Im(ε_{reff})
nthick1.geo	51.06	-0.2050	10.27	-0.0253	8.523	-0.0811
nthick2.geo	50.03	-0.1100	10.42	-0.0155	8.270	-0.0490
wthick1.geo	49.83	-0.1540	10.60	-0.0202	8.000	-0.0628
wthick2.geo	49.27	-0.1294	10.50	-0.0178	8.145	-0.0559

Section 12.4 Accurate Modeling of Stripline Structures

We have shown that modeling thickness can improve simulation accuracy for microstrip structures with large t/w ratio. For stripline structures, we should add thickness to the structure even when t/w ratio is not very big. Before we discuss an example, we will explain the reason first.

The cross-sections of a typical microstrip structure and a stripline structure are shown in Figure 12.10. A typical microstrip line has one ground plane and one substrate with the dielectric constant e_r much larger than 1. Whereas, a typical stripline has two ground planes and the dielectrics is on both sides of the metallic strip are the same.

From electromagnetic theory, we know that the field is concentrated between the metallic strip and the ground plane in the microstrip structure, and the charge density is directly related to the normal component of the electric field on the metallic structure. Not only that, the charge density is also directly related to the dielectric constant of the material next to the metallic strip. For a microstrip structure, the electric field is concentrated at the bottom of the metallic strip and the bottom of the metallic strip is also a high dielectric constant material. Therefore, the charge concentration on the bottom surface of the metallic strip is much higher than that on the top surface. Electric current is basically the flow of charges. Therefore, we expect much higher current density on the bottom surface of the metallic strip in a microstrip structure. That is why we can approximate a microstrip structure using infinitely thin metallic plate on top of the substrate and we can still get very accurate result.

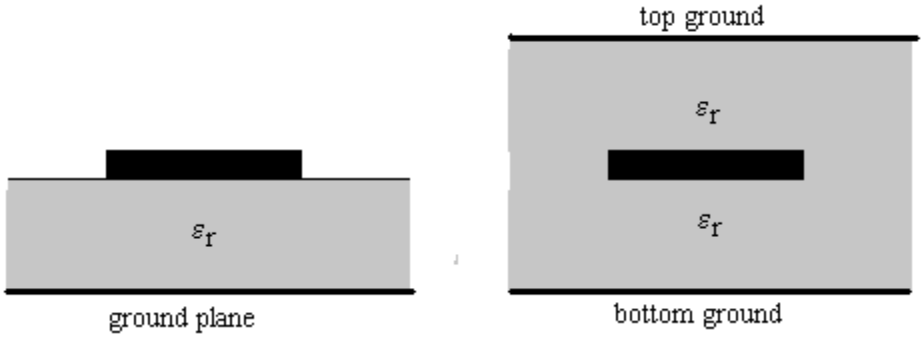


Figure 12.10

Some users may wonder whether our claim the current is concentrated on the bottom surface is true at low frequency when the skin depth is much bigger than the strip thickness. Our research shows that the current is still quite concentrated on the bottom surface even at low frequency. Certainly, there will be more volume current on the cross-section. We did try to model the current as uniform in the cross-section and we didn't get reasonable result. Modeling the metallic strip as an infinitely thin plate on top of the substrate yields much better result. It is an indirect proof that the current is not uniformly distributed across the cross-section.

For stripline circuits, charge and current will be equally divided onto the top and bottom surfaces of the metallic strips. Certainly, modeling both the top and bottom surfaces of the metallic strips is critical.

Example geometry files are **stripln1.geo**, **stripln2.geo**, **stripln3.geo** and **stripln4.geo** in **.\ie3d\samples** directory. The t/w ratio is also 6%. The parameters of the stripline are:

Substrate Dielectric Constant = 2.2
Substrate Loss Tangent = 0.001
Substrate Thickness = 30 mils (ground to ground)
Metallic Strip Thickness = 1.4 mils
Metallic Strip Width = 23 mils
Metallic Bulk Conductivity = 4.9e+7 s/m

stripln1.geo ---- The finite-thickness metallic strip is modeled as infinitely thin strip and the loss effect is corrected using the surface impedance. The infinitely thin strip is located at the middle of the dielectrics. Edge vertices are inserted to enhance accuracy.

stripln2.geo ---- The finite-thickness metallic strip is modeled as two parallel plates. One is on the top surface of the metal strip and the other is on the bottom surface of the metal strip. Edge vertices are inserted to enhance accuracy.

stripln3.geo --- The finite-thickness metallic strip is modeled as 4 surfaces. One is on top and another is on the bottom. There are two vertical plates on the edges. There are not inserted vertices near the edge. The bottom plate is at 14.3 mil level and the top plate is at 15.7 mil level.

stripln4.geo --- Same as **stripln3.geo** except AEC is applied to thickness polygons too.

The simulation results are saved in **stripln1.sp**, **stripln2.sp**, **stripln3.geo** and **stripln4.sp** in the **.\ie3d\samples** directory.

Table 12.3 shows the comparison among the four schemes. We can see the 4-plate models with and without the edge cells agree quite well. The parallel plate model is also very close. We see about 8% errors in the infinitely thin strip approximation. For the infinitely thin plate model with the same t/w ratio, we get much higher error for striplines than for microstrips.

Table 12.3 Comparison of the 4 single stripline models at 10 GHz.

F=10 GHz	Re(Zc) Ω	Im(Zc) Ω	λ (mm)	α (dB/mm)	Re(ε_{reff})	Im(ε_{reff})
Stripln1.geo	52.84	-0.0329	20.15	-0.0048	2.213	-0.0079
Stripln2.geo	48.83	-0.0113	20.16	-0.0039	2.211	-0.0063
Stripln3.geo	48.29	-0.0130	20.20	-0.0039	2.202	-0.0063
Stripln4.geo	48.04	-0.0047	20.21	-0.0035	2.201	-0.0058

Coupled stripline structures are available in **cstrip1.geo**, **cstrip2.geo**, **cstrip3.geo** and **cstrip4.geo** in **.\ie3d\samples** directory. Regarding the thickness modeling, they correspond to the **strip1.geo**, **strip2.geo**, **strip3.geo** and **strip4.geo**, respectively. The gap between two coupled lines is 9 mils. Their results are saved in **cstrip1.sp**, **cstrip2.sp**, **cstrip3.sp** and **cstrip4.sp** in the **.\ie3d\samples directory**. The comparison of the couplings from the 4 models is listed in Table 12.4.

Table 12.4 Comparison of the 4 coupled stripline models at 10 GHz.

F=10 GHz	Re(Zc) Ω	Im(Zc) Ω	λ (mm)	α (dB/mm)	Re(ε_{reff})	Im(ε_{reff})
Cstrip1.geo	94.71	-0.072	20.20	-0.0048	2.203	-0.0079
Cstrip2.geo	83.96	-0.033	20.21	-0.0038	2.201	-0.0063
Cstrip3.geo	83.83	-0.050	20.20	-0.0044	2.202	-0.0072
Cstrip4.geo	83.12	-0.031	20.21	-0.0038	2.201	-0.0062

We can see that modeling the finite thick strip with parallel plates certainly improves the accuracy in modeling the coupling between coupled strips. Adding the edge walls as what we did in **cstrip3.geo** and **cstrip4.geo** will further improve accuracy. The predicted Zc from **cstrip1.geo** is about 15% off the value from **cstrip4.geo**. The error certainly is not small. Characteristic impedance is a more sensitive parameter.

In the above discussion, we did not give any quantitative criteria when we should model structures with thickness. In fact, giving any t/w ratio as criteria may be misleading because the answer really depends on how accurate you want the result to be. If you want very accurate result, you may need to model the thickness even the t/w ratio is not big. You have to sacrifice the speed.

Section 12.5 Discussion on De-embedding Schemes

We have implemented several different de-embedding schemes on the IE3D to solve problems in different kinds of environment. More discussion can be found in Appendix AM.

The Advanced Extension scheme is normally the best when the applicable frequency range and stability is concerned. The Extension for MMiC is also a very stable scheme except it may fail for highly dispersive transmission lines. The Vertical Localized and Horizontal Localized schemes are extremely useful for highly packed structures and probe-fed structures.

Table 12.5 The file names and de-embedding schemes for the curves in Figure 12.11.

Notation	Geometry File	Scheme	Time (sec/freq)
Adv Ext	bend_adv_ext.geo	Advanced Extension	1.958
MMIC Ext	bend_mmic_ext.geo	Extension for MMIC	1.834
50-Ohms Waves	bend_50_ext.geo	50-Ohms for Waves	3.154
Waves	bend_waves_ext.geo	Extension for Waves	3.083

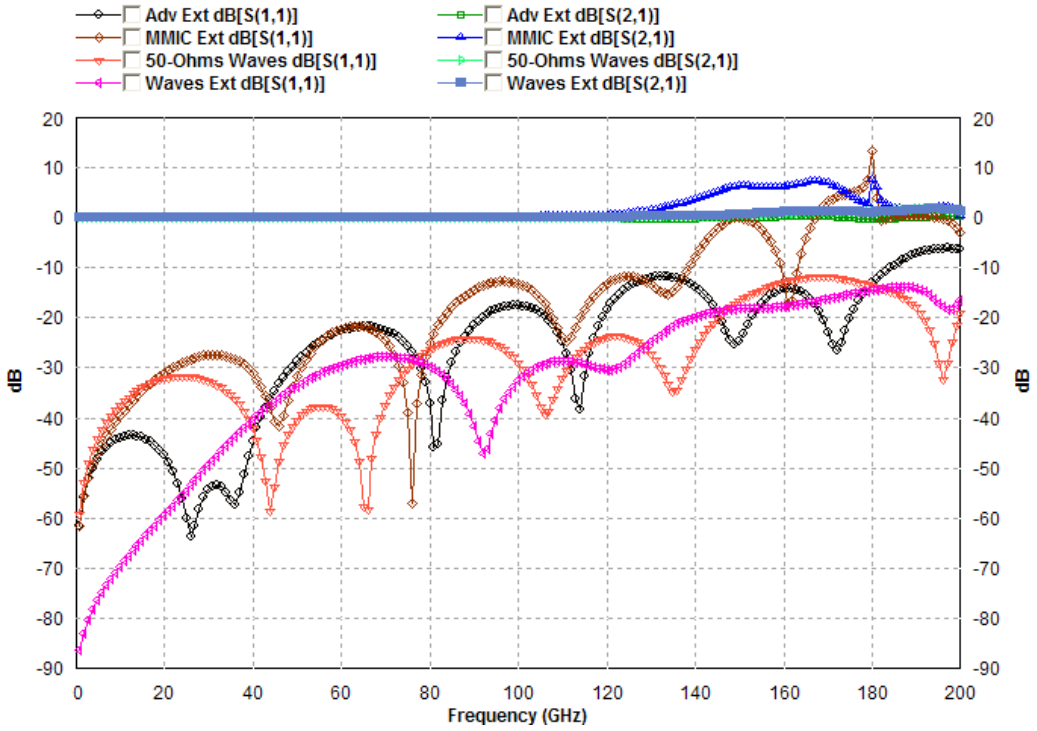


Figure 12.11 Comparison of the 3 extension de-embedding schemes.

We are going to provide an estimate on the applicable range of the different extension schemes. Before we do that, we would like to provide an example. Figure 12.11 shows the comparison of the IE3D simulation results on a microstrip bend structure using four different de-embedding schemes. The file names and the de-embedding schemes are shown in Table 12.5.

The microstrip bend structure is on a 100-micron or 4-mil GaAs (ε_r = 12.9) substrate. The width of the microstrip is 75 microns. At low frequency (5 GHz), the Zc is about 50-Ω.

If there is no loss involved, the dB(S11) from **Extension for Waves** should approach minus infinity. We can keep it below –40 dB below 40 GHz and –28 dB below 100 GHz. For 2-port structures with |S21|

close to 1. You may wonder why the dB(S11) does not go to -∝ because the s-parameters using the **Extension for Waves** scheme are normalized to the port impedance. The theory in textbooks tells you that the generalized dB[S11] should be -∝ for a uniform TLN. Basically, the difference is in the loss. If the TLN is lossless, we should expect the generalized dB[S11] = -∝. When the TLN is a lossy TLN, the generalized dB[S11] ≠ -∝ unless the TLN has a complete PEC boundary. In fact, waveguide theory is no longer exact if the lossy material goes to infinity on the cross-section. Such a conclusion can not be found from textbooks because textbooks are always dealing with idealized lossless cases.

The **Extension for MMIC** agrees quite well with the **50-Ohms for Waves** below 50-GHz. Its accuracy is affected by the dispersion beyond 50-GHz for the 100-micron substrate. The results of the **Advanced Extension** scheme are quite stable even at 100 GHz.

Many users asked the question. How high can IE3D go in terms of frequency? It is a very hard to answer question. It really depends upon what structures you are simulating. In some sense, the kernel of the IE3D should be good for any structures the Maxwell's equations can be applied. However, de-embedding is an important issue. We always assume there is only one single propagating mode in the transmission lines. It is in fact what designers want. Microwave network theory is hard to apply when multiple propagating modes exist. The existence of higher order propagating modes creates high dispersion and unexpected phenomenal result. Designers always want the transmission lines to operate at single propagating mode. When the cross-sectional dimensions of a transmission line become too big, higher-order modes will become propagating. In order to guarantee single mode, designers should try to limit the substrate height of a microstrip circuit. For a typical 100-micron substrate GaAs circuit, the 2^{nd} order mode becomes propagating at about 95 GHz. If you want to design your circuit to work close to 95 GHz or beyond 95 GHz, you should try to reduce the substrate height (or just the strip to ground distance). Otherwise, the circuits will become un-predictable.

The above discussion is only on the extension de-embedding schemes. Generally, the extension de-embedding schemes are the most accurate schemes for the IE3D, because the de-embedding arms really isolate the ports and the discontinuities. However, we do not always have enough rooms for the port extensions. We have to use localized ports in such a case.

As a summary, we list the normal criteria for the extension de-embedding schemes for the microstrip systems in Table 12.6. The above criteria are just estimated limits. In practical applications, you can check your structures using some simple through lines and bends against the above estimated limits. You should define no loss for the metals and substrates. Then, you detect the loss in the structure. When you start seeing abnormal loss (not from radiation), it would be the limit for the de-embedding scheme.

Table 12.6 The valid ranges for the extension de-embedding schemes for microstrip structures.

Valid Application Range	Criteria (Fmax: Highest Frequency; H: Substrate Thickness; ε_r: Dielectric Constant)
Advanced Extension	**Fmax · H / √(ε_r) ≤ 2.8-3.9 (GHz · mm) ≈ 112-156 (GHz · mil)**,
Extension for MMIC	**Fmax · H / √(ε_r) ≤ 1.4-1.9 (GHz · mm) ≈ 56-76 (GHz · mil)**,
Extension for Waves	**Fmax · H / √(ε_r) ≤ 2.8-3.9 (GHz · mm) ≈ 112-156 (GHz · mil)**,
50-Ohms for Waves	**Fmax · H / √(ε_r) ≤ 2.8-3.9 (GHz · mm) ≈ 112-156 (GHz · mil)**,

In the IE3D, we have 3 localized de-embedding schemes. As we discussed before, extension de-embedding schemes use de-embedding arms to isolate the excitation source and the port in order to get the best accuracy. However, there are cases we do not have space for the de-embedding arms, such as the amplifier example in Chapter 6. We have to use some localized de-embedding schemes.

The **Localized for MMIC** is the first localized scheme introduced in IE3D. It is easy to setup. However, it requires a ground plane at z = 0. It is only accurate when the port to ground distance is much smaller than waveguide wavelength and the dimensions of your circuit.

The **Vertical Localized** scheme is more general implementation. In fact, **Localized for MMIC** is only a special case of the **Vertical Localized** scheme. The **Vertical Localized** scheme is extremely flexible. It can be used almost everywhere for structures with or without ground plane. However, it has a very strict requirement on constructing the port proximity. Many users have experienced difficulty to get it work. Its requirements are:

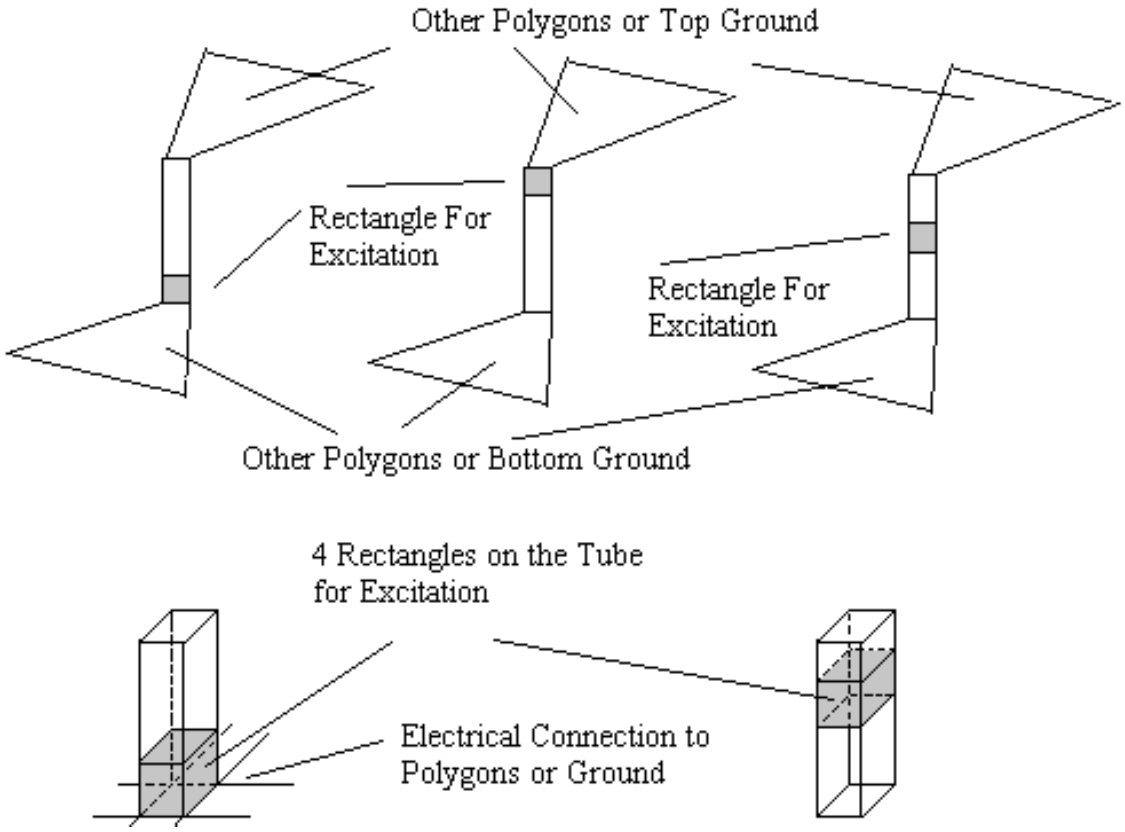


Figure 12.12 The good setup for **Vertical Localized** scheme.

(1) You need to build the excitation as a set of vertical rectangles. The vertical span of the vertical rectangles must be small when compared with wavelength (<5% of the waveguide wavelength).

(2) You need to define the vertical rectangles as the source. You also need to define the 2 z-coordinate levels of the vertical rectangles as the positive and negative levels.

(3) You need to make sure the vertical rectangles will connect to other polygons or the ground planes. Neither the positive level nor the negative level can be left open.

Figure 12.12 shows the good setup and Figure 12.13 shows the bad setup for the **Vertical Localized** scheme. Setting up a **Vertical Localized** scheme might be a tedious work for new users of the IE3D. Fortunately, we have implemented automatic scheme to define the **Vertical Localized** ports for applications such as probe-fed patch antennas. It should be mentioned that the **Localized for MMIC** and **Vertical Localized** schemes are best for probe-fed patch antennas.

For manual construction, the **Edit->Add Via on Edges** command is very useful in constructing **Vertical Localized** ports. As a general rule in defining a **Vertical Localized** port on 2 polygons in different layers as shown in Figure 12.14A, you should first select edge(s) of one polygon, use **Add Via on Edges** to

add some rectangles to the other layer. Then, connect the end of the vias with the other polygon (see Figure 12.14). In fact, the **Add Via on Edges** command is improved in IE3D 11. You can build the horizontal polygons first. The via will be connected to where it touches if there is any 2D polygons at the level. Another command we have implemented is the **Adv Edit->Build Via Connection on Edges**. This command allows you to build the via and the port in one shot.

In case the two polygons are on the same level, you should use the **Horizontal Localized** scheme, which is the corresponding localized scheme for horizontal structures.

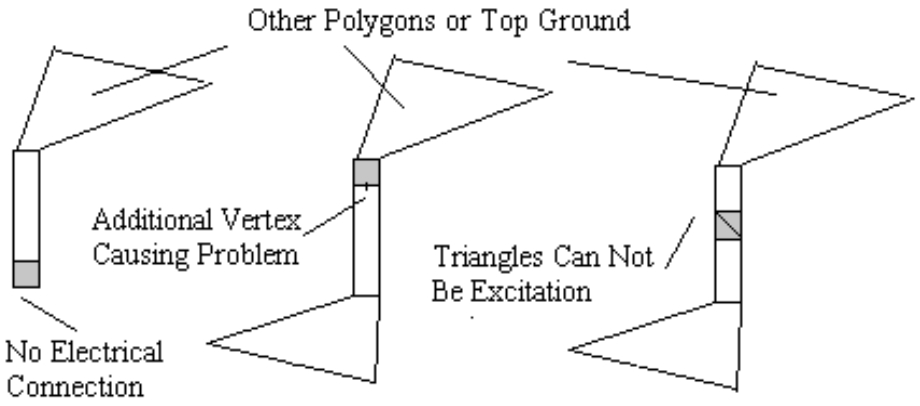
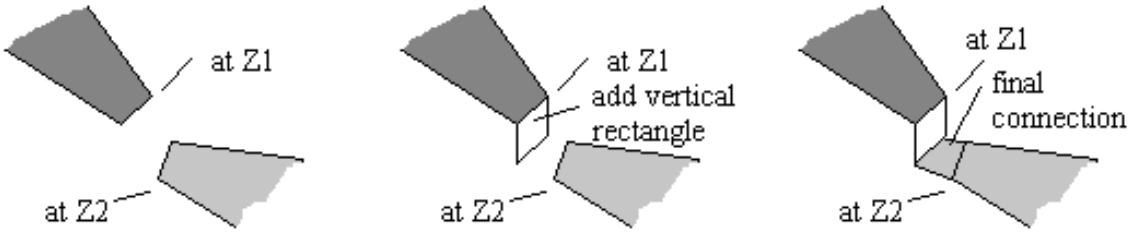


Figure 12.13 The bad setup for Vertical Localized scheme.



(A) (B) (C)

Figure 12.14 Illustration for the setup of a general **Vertical Localized** port.

We understand that the different de-embedding schemes may still be confusing to users who do not have much experience in electromagnetic modelings. Our conclusion is that we need to go with them. No single de-embedding scheme can do all the work for you. They have their suitability ranges. Just like you do a measurement, you need different kinds of schemes for different kinds of applications. For different circuits (microstrips or coaxials or waveguides, etc), you need different standards in your measurement in order to get high accuracy result. Same philosophy applies here.

Table 12.7 are the suggestions on using the de-embedding schemes. It should give you roughly an idea which scheme you should use for a specific kind of applications. This is a general rule. You should try to practice more so that you can understand more about the advantage and disadvantage of each de-embedding scheme.

Section 12.6 Reducing Number of Cells for Better Efficiency

Efficiency and accuracy are usually contradictory. In order to improve efficiency, we need to reduce the accuracy. However, if you are an experienced user of the IE3D, you will be able to improve the efficiency significantly with little degrading in accuracy, or improve the accuracy significantly with little degrading in efficiency.

Table 12.7 Suggestions on de-embedding schemes

Scheme	Advanced Extension	Extension for MMIC	Localized for MMIC	Vertical Localized	Extension for Waves	50-Ohms for Waves	Horizontal Localized
Single Layer Substrate MMIC and RF Circuits	Best	Best	Good	Good	Best	Best	Good
Multi-Layer Substrate MMIC and RF Circuits	Best	Best	Fair	Good	Best	Best	Good
Shielded Circuits	Best	Best	Fair	Good	Best	Best	Good
Highly Packed MMIC and RF Circuits	Possibly No Room	Possibly No Room	Good	Good	Possibly No Room	Possibly No Room	Good
Modeling Lumped Elements in Circuits	Possibly No Room	Possibly No Room	Best	Best	No	Possibly No Room	Best
Very High Frequency Microwave Application	Best	Good	Fair	Fair	Best	Best	Fair
50-Ω Normalization	Yes	Yes	Yes	Yes	No	Yes	Yes
High ε_r Substrate RF Circuits	Best	Best	Fair	Fair	Best	Best	Fair
Thick Substrate	Best	Good	No	Best for Diff. Feed	Best	Best	Best for Diff. Feed
Differential Port	Best	Best	Good	Best	No	No	Best
Digital Circuits with Large Port to Ground Distance	Best	Best	No	Good	No	Good	Good
Digital Circuits with Small Port to Ground Distance	Best	Best	Good	Good	No	Good	Good
Structure without Ground Plane	Best	Best	No	Best	No	No	Best
Edge-Fed Microstrip Antennas	Best	Best	Good	Good	Best	Best	No
Edge-Fed Microstrip Antennas with Finite Ground Plane	Good	Good	No	Best	No	No	No
Probe-Fed Antennas with Infinite Ground Plane	Fair	Fair	Best	Best	No	No	No
Probe-Fed Antennas with Finite Ground Plane	Fair	Fair	No	Best	No	No	No
Probe-Fed Antennas with Infinite Ground Plane Not at Z = 0	Fair	Fair	No	Best	No	No	No
Coaxial Structures	Best	Best	No	No	No	No	No
Coupled Ports		Best	Good	Good	No	No	Good
Magnetic Current Ports (Diff. Ports)	Best	Best	No	No	No	No	No
True Definition of S-Parameters	No	No	No	No	Yes	No	No
Circuits with Enclosures	Best With Enough m	Best With Enough Rm	Good	Good	Best With Enough Rm	Best With Enough Rm	Best With Walls

Ultra Thick Substrate Circuits with Enclosures	Best	No	No	No	No	No	Yes

In all the previous examples we discussed in this manual, we always use at least 15 for Cells per Waveguide Wavelength (Ncell) for meshing. Usually, simulation accuracy is guaranteed with about 20 cells per waveguide wavelength (Ncell=20). Ncell=15 is also a very safe number.

Before we simulate a structure, we really do not know what the waveguide wavelength is. We need to estimate it before we mesh the structure. For microstrip structures with single dielectric substrate, we will estimate the effective dielectric constant by taking a weighted average with $\varepsilon_{reff} = [\ C\ \varepsilon_r + (\ 2 - C\ )\ ] / 2$, where ε_r is the dielectric constant of the substrate; C is a constant between 1 and 1.6. Then, we estimate the waveguide wavelength as $\lambda = \lambda_0 / \sqrt{\varepsilon_{reff}}$. Obviously, the accuracy of the estimated λ is dependent upon the C value. For different structures, the best C value is different. For narrow microstrip lines, the best C value should be slightly larger than 1. For wide microstrip lines or patch antennas, the C value should be close to 1.6. For simplicity, we just take a value for C for all different kinds of structures. Therefore, the estimated λ can be different by 10% or even 20%. However, this kind of error is acceptable because the estimated λ is used for meshing only. The final calculated λ will be much more accurate, with accuracy about 0.1-2%.

You should understand that the estimated λ can be off much especially for multiple substrate structures. You should try different settings to check the convergences.

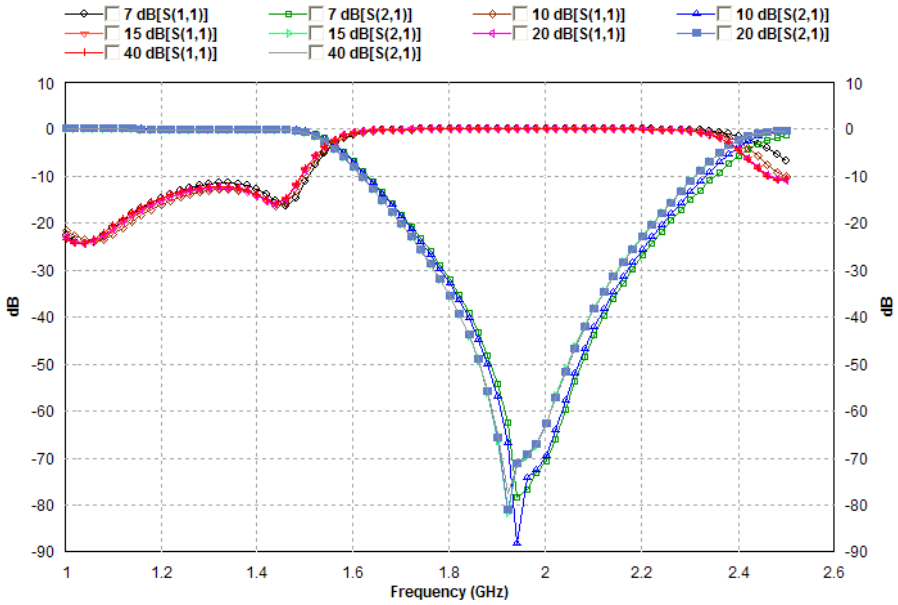


Figure 12.15 Influence of Ncell = 7, 10, 15, 20 and 40.

Sometimes, we would like to reduce the Ncell. We may want to get fast result, or we may want to simulate larger structures with fixed amount of RAM. What would be the consideration we should take? We will explore the effect of reduced Ncell.

Saved in **.\ie3d\samples\mwlp.geo** is a filter from a course note of Besser Associates. We simulate the structure with Ncell = 7 (**mwlp7.sp**), Ncell = 10 (**mwlp10.sp**), Ncell = 15 (**mwlp15.sp**), Ncell = 20 (**mwlp20.sp**) and Ncell = 40 (**mwlp40.sp**). The comparison is shown in Figure 12.15. Surprisingly, very good result is obtained even with Ncell = 7. This example demonstrates that it is possible to obtain good result for some structures using very small Ncell = 10 or even 7.

Saved in **.\ie3d\samples\lex8.geo** is an edge fed patch antenna with 8 cells per wavelength or Ncell = 8. You can create the **alex12.geo** for Ncell = 12, **alex15.geo** and Ncell = 15 and **alex20.geo** for Ncell = 20. This structure was published by S.-C. Wu, N. G. Alexopoulos, and O. Fordham, "Feeding structure contribution to radiation by patch antenna with rectangular boundaries," *IEEE Trans. Antennas Propagat*., Vol. AP-40, Oct. 1992, pp.1245-1249. Good agreement between the IE3D result and the measured result is observed. The comparison is documented in the IE3D Benchmark Examples.

The meshed structure is shown in Figure 12.16. The structure is simulated from 4 to 20 GHz with Ncell = 8, 12 and 15 at Fmax = 20 GHz. The results are compared in Figure 12.17. There is little difference between the results for Ncell=12 and Ncell=15. The result for Ncell=8 starts deviating at 16 GHz. Ncell = 8 at Fmax = 20 GHz is basically the same as Ncell = 10 at Fmax = 16 GHz. We can see that good result can be obtained with Ncell = 10 for some structures. A comparison on computational resource among the 4 different cases is shown in Table 12.8.

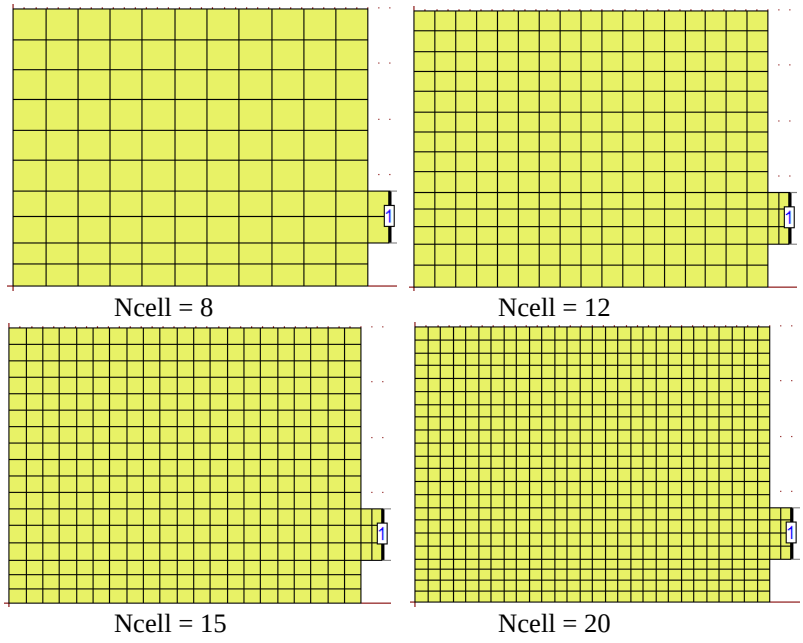


Figure 12.16 Comparison of the different meshings at Fmax = 20 GHz.

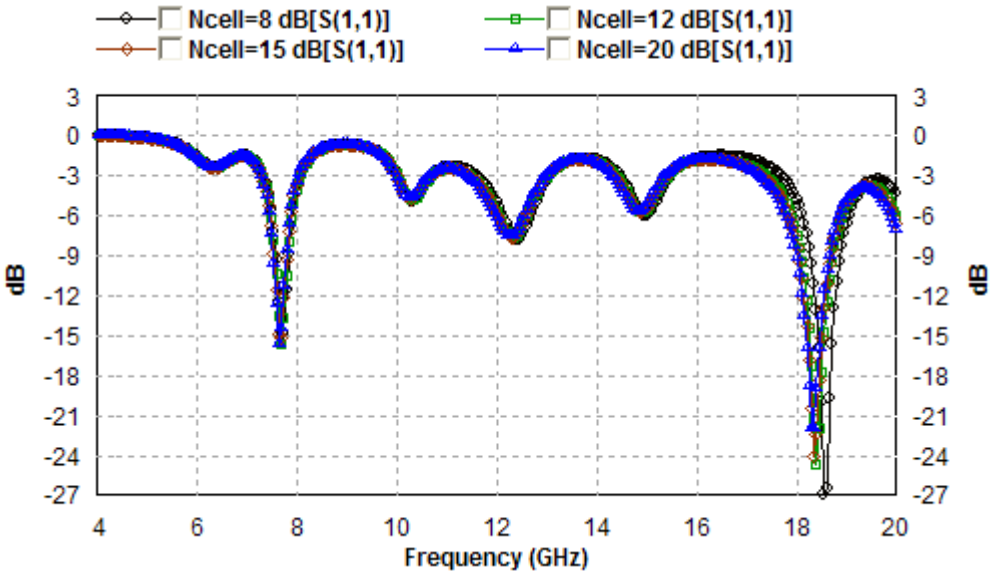


Figure 12.17 The frequency response of an edge-fed rectangular patch with different meshes (Ncell=8, 12, 15, 20)

Big difference is observed in the computational resource for the 4 different meshes. For medium and large structures, when we double the Ncell, the number of unknowns N may be quadruple. The simulation time, which at least is proportional to N^2, will become about 16 to 32 times. The required memory will be about 16 times using the symmetric matrix solvers.

The above 2 examples demonstrated that reducing the Cells per Waveguide Wavelength (Ncell) in the meshing is the most significant way to improve efficiency.

One final remark we would like to make is that you need to be more careful when you push the limit of Ncell down to 10. There are some cases you cannot use too small Ncell. For some special cases, even Ncell = 20 cannot guarantee complete convergence. It is suggested that you check a simulation result with 2 different meshes whenever it is possible or whenever you are not so sure about the results.

Table 12.8 Comparison of the 4 different meshes for the patch antenna.

File	alex8.geo	alex12.geo	alex15.geo	alex20
Ncell	8	12	15	20
Total Number of Cells	120	256	375	640
Total Number of Unknowns	216	478	709	1228
Simulation Time (sec/freq)	0.5	1	3	12

Section 12.7 Using Matrix Solvers Wisely

We have implemented a few different matrix solvers.On IE3D 15.0, we have removed PMS, IMS, AIMS, Adaptive AIMS, AIMS II and AIMS III due to the fact they are not as good as the GEMS solvers. We have: Full Matrix Solver (FMS), Advanced Symmetric Matrix Solver (SMSi), Adaptive Symmetric Matrix Solver (SMSa) and General Efficient Matrix Solver GEMS-F and GEMS-I

Until now, we only discussed using SMSi and SMSa. The FMS, SMSi and SMSa are the matrix solvers without special requirement. They can be applied to any structures. Starting from IE3D 12, FMS and SMSa are parallel matrix solvers. They are much faster than SMSi no matter whether you have more than one CPU. SMSi is a symmetric matrix solver and it requires 50% of the memory FMS requires. SMSa may require the same RAM as FMS for high speed. However, it may adaptively reduce the RAM

requirement in case no enough RAM is available. When it uses less RAM, it will slow down. Normally, we should use SMSa as our default matrix solver. Complete discussion on different matrix solvers is documented in Table 12.9.

Table 12.9 The different matrix solvers.

Matrix Solver	Description
Full Matrix Solver (FMS)	It is multi-threaded and multi-CPU supported. It is very fast. Assume its RAM requirement is R0, we have: T0 = C0 N^3, R0 = 16 N^2 bytes, where C0 is a constant and N is the number of unknowns.
Advanced Symmetric Matrix Solver (SMSi)	It is a very solid symmetric matrix solver and very efficient. However, it does not have multi-CPU support. It saves 50% of the RAM or its RAM requirement is R1 = 0.5 R0 = 8 N^2 bytes.
Adaptive Symmetric Matrix Solver (SMSa)	It is the default matrix solver in IE3D 12. It is an extremely efficient symmetric matrix solver. It can be a few times faster than SMSi especially with multi-CPU support. Its RAM requirement ranges from that for SMSi to that for FMS. It is faster with more RAM while it can adaptively choose the best scheme. Its speed can hardly be beaten by even iterative matrix solvers (AIMS II, AIMS III, GEMS-Fand GEMS-I) unless it is extremely big structures.
Generally Efficient Matrix Solver (GEMS)	GEMS matrix solver can be divided into GEMS-F and GEMS-I. It is the only iterative matrix solver supported in IE3D 15.0. GEMS-F can be much faster and require much less RAM than SMSa for large structures with infinite ground. GEMS-I may require the least RAM. However, it can be extremely slow and it should be the last resort for simulating large structures.

Setting up default matrix solver is discussed in Appendix A. In Appendix A, we also discuss advantage and disadvantage of the a few matrix solvers. We will not repeat the discussion here.

Section 12.8 Dividing Circuit into Sub-Circuits Wisely

There are structures that the IE3D is difficult to simulate or inefficient to simulate. For such cases, we need to divide a large circuit into small sub-circuits.

What are the criteria in dividing circuits? The most important criterion is that we need to make sure the sub-networks after division are only weakly coupled or not coupled. On the IE3D, we also need to be careful in planning the s-parameter extraction or de-embedding.

Saved in .\ie3d\samples\cplines1.geo is 3-port structure (see Figure 12.18). Certainly, we can simulate it using the IE3D. The full model predicts that the |S(1,1)| = - 9.618 dB and |S(2,1)| = -3.535 dB at 0.5 GHz. The results should be quite accurate. Apparently, for a loss-less case, the |S(1,1)| = - 9.542 dB and |S(3,1)| = -3.522 dB.

Assume we want to simulate the structure as two portions. We need to divide the circuit somewhere at the middle. We may have different schemes to divide it. Figure 12.19 and Figure 12.20 show two different kinds of divisions: (1) Aligned Division: The structure is cut somewhere on the coupled lines. The cuts on the two lines are aligned. The left sub-circuit is cplines1a1.geo and the right one is cplines1a2.geo. The final s-parameters are calculated on MODUA and saved in .\ie3d\samples\cplines1a.sp; (2) Unaligned Division: The structure is cut somewhere on the coupled lines. The cuts on the two lines are not aligned. The left sub-circuit cplines1b1.geo and the right one is cplines1b2.geo. The final s-parameters are calculated on MODUA and saved in .\ie3d\samples\cplines1b.sp. The comparison between the Completely (cplines1.sp), Aligned (cplines1a.sp), Un-aligned (cplines1b.sp) is shown in Figure 12.21. Interestingly, the Aligned model creates almost identical s-parameters with the complete structure while the Un-Aligned model creates deviation at both low frequency and high frequency ends.

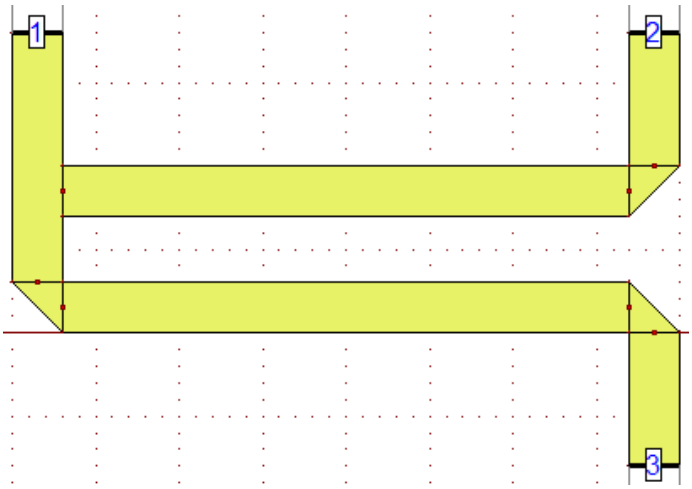
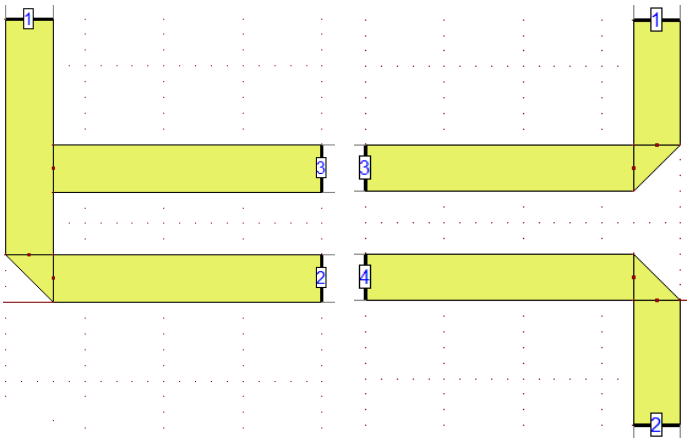


Figure 12.18 A 3-port structure.

The problem for the un-aligned case is that the port 3 is affected by the port extension on port 2 for the structure cplines1b1.geo. If you display the 3-port s-parameters in cplines1b1.sp, you will see the similar rapid change at the low frequency. Interestingly, the ports 3 and 4 are also not aligned for the cplines1b2.geo. However, we do not see the rapid change in the s-parameters at the low frequency ends.

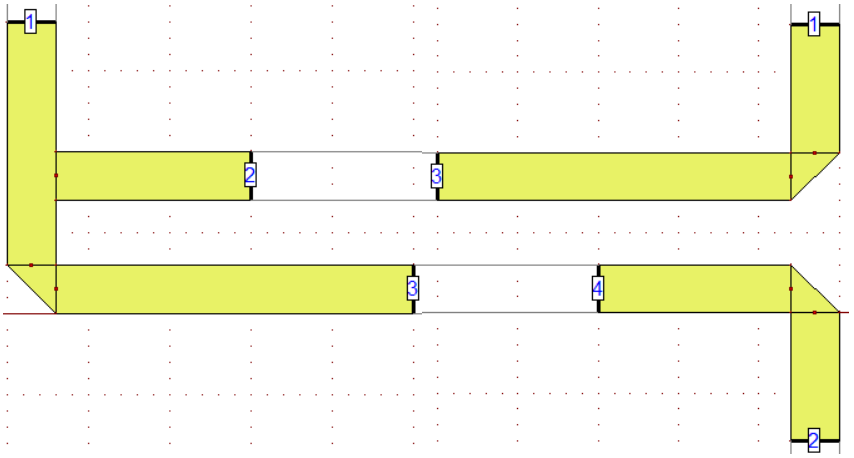
The example shows that we should be careful when we divide a circuit into multiple sub-circuits. We should try to align the ports. It is not necessary to exactly aligned. However, we should not allow the cases with two ports are obviously not aligned.



(a) cplines1a1.geo (b) cplines1a2.geo

Figure 12.19 The aligned division of circuit (cplines1a.spm)

In practical applications, we may also encounter structure similar to the one in Figure 12.22. What should be the best port scheme we should use for it? Shown in Figure 12.22a is the one using localized scheme. Shown in Figure 12.22b is the one using extension port scheme. No matter what, we should avoid using extension port scheme for the port 4. Its port extension is coupled with the rest of the structure and the de-embedding will be affected.



(a) cplines1b1.geo (b) cplines1b2.geo

Figure 12.20 The un-aligned division of circuit (cplines1b.spm)

Is the one using the localized scheme a good model? It really depends upon the reality. If we will connect a lumped element between port 4 and the ground, the localized port model is a good one. In case there is a continuous trace beyond the port 4 in your real circuit, what is the best scheme for it? It looks like the extension port scheme might be good for it because it adds a line on it anyway. However, that is not true. We should not use the extension port. We can use the localized port to approximate it. However, it may not be the best case. In fact, we should not model it like what is shown in Figure 12.22. We should add the extra line beyond the port 4 in case the reality is like that. We should try not to cut a circuit at where strong coupling is happening.

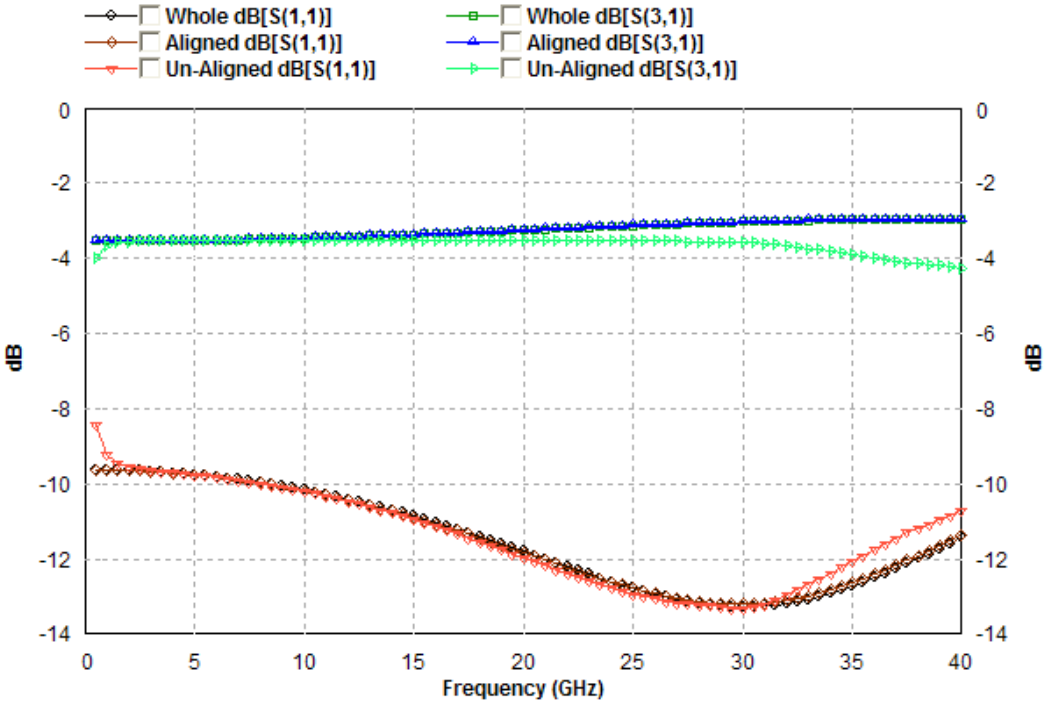
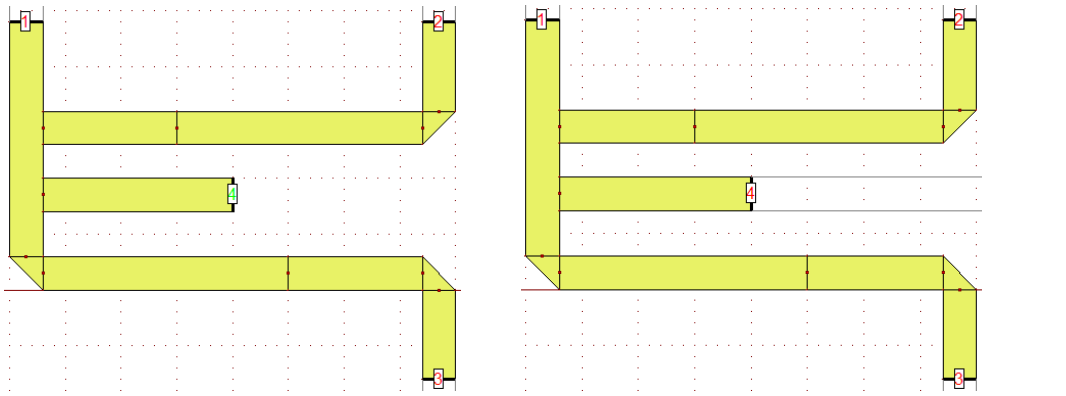


Figure 12.21 The comparison between the 3 models.



(a) Localized port for port 4 (cplines2.geo) (b) Extension port for port 4 (cplines2b.geo)

Figure 12.22 Different ports schemes for a coupled structure.

Section 12.9 Other Techniques to Improve Accuracy

There are many other ways to improve accuracy. For example, meshing alignment between coupled plates and printed strips on finite dielectrics. We have discussed this topic other chapters. We will not repeat them here. We will end this chapter here.

Chapter 13 Simulation of Antenna Arrays

We have demonstrated how to use IE3D to simulate antenna elements. Certainly, we can use the IE3D to simulate antenna array. We have touched this topic in Chapter 12 on matrix solvers. In this chapter, we will concentrate on antenna array simulation and design. IE3D can accept arbitrarily shaped structures with any number of ports. There is no difficulty for a user to enter an antenna array layout and simulate it. However, there are some points we may need to take into consideration. Depending upon the nature of an array or how accurate we want to model an array, we can use different approaches. Table 13.1 lists the 4 most commonly used approaches. We have successfully implemented the periodic Green's function formulation for precise modeling of planar phase array. We will also use it to simulate the following linear phase array.

This chapter is virtually not updated on IE3D 12 and IE3D 14. The newly implemented technology such as the GEMS solvers, IE3D FASTA and multiple CPU support may make the simulations of larger arrays become simple jobs. Also, many new commands will make construction of the structures much simpler. We would like to have you informed about the above fact.

Table 13.1 The common 4 approaches in array simulation and design in IE3D.

Approach	Features
Array Factor (or No Coupling)	An array is simplified as an element plus an array factor. No element coupling and feed network radiation is included. This is the fastest scheme. There is no limitation on the size of the structure. However, it is the least accurate model. The radiation from the feed network is normally not included. It is suitable for loosely coupled and equally spaced array elements with well-shielded feed network.
Periodic Wall	An array is considered as an element with duplications from the periodic walls. The major couplings between elements are included from the periodic wall condition. The radiation from the feed network is normally not included. This is also a fast scheme. It is suitable for large phased arrays with identical and equally spaced elements. The excitation phase can be tempered. One-dimensional periodic array is an approximation using finite number of images. More discussion can be found from Appendix H. Starting from the IE3D 8.0, we are able to model the 2D periodic array precisely. We can also use it to model 1D periodic array accurately because we can always define much larger element-to-element distance in the other direction. It seems that the 2d periodic array modeling is a better approach than the finite images one.
Separation	The whole structure is divided into 2 portions: all the antenna elements and the feed network. All the antenna elements are modeled in 1 simulation. The feed network is simulated in another simulation. The radiation from the elements and the feed network will be merged (you can optionally neglect the radiation from the feed network). This approach includes the coupling among elements, and the coupling within feed network. The simulation time and resource requirements are moderate. One advantage using this scheme is that you can access the power, voltage and current distribution in the whole array. Such a feature may provide very important information in array design. This new feature is also available to the previous 2 schemes. The new feature in IE3D 7.0 allows you to check the radiation from different part.
Complete Array	The complete array including elements and feed network is modeled in one single simulation. The coupling and radiation from all components are included. This certainly is the most accurate scheme. However, it requires the most computational resources. The power, voltage and current information at ports is not available for this scheme. For relative large arrays, the iterative matrix solver (IMS) or the partial matrix solver (PMS) is required to reduce the simulation time and RAM requirement.

In this chapter, we will concentrate on a 16-element linear array design. The same procedures can be applied to planar array design. The layout of the 16-element linear array is shown in Figure 13.1. We would like to optimize it for the resonant frequency at 1.88 GHz. We also would like the array to be uniformly distributed.

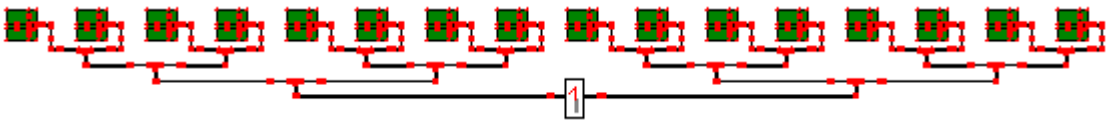
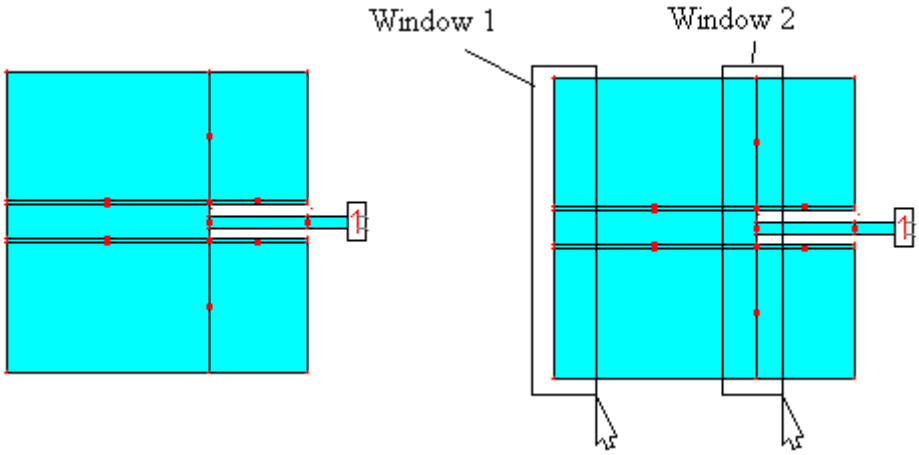


Figure 13.1 The 16-element linear array to be modeled.

Section 13.1 Element Analysis and Optimization

An array design starts from element analysis. The initial array element is saved in **.\ie3d\samples\ar_elem.geo**. The initial array element can be from some simpler tool or formula.

Step 1 Run MGRID and open geometry file **.\ie3d\samples\ar_elem.geo**. The element layout is shown in Figure 13.2. The feed line of the element was optimized for 50-ohms. Please save it into **.\ie3d\practice\ar_elem.geo**. Simulate the structure from 1.85 to 1.9 GHz without edge cells. The resonant frequency may not be predicted very precisely without the edge cells. We disable **AEC** for simplicity here. In practical design, you may consider using it for higher accuracy. The simulation result is displayed in Figure 13.3.



(a) (b)

Figure 13.2 The initial array element.

As you can see, the resonance is slightly off. We would like to optimize the resonant frequency at 1.88 GHz with perfect match.

Step 2 Select **Edit->Select Vertices** command. Select the vertices using Windows 1 in Figure 13.2b.

We would like to control the length of the patch by moving its vertices at the left end in the φ = 0 angle.

Step 3 Select **Optim->Variable for Selected Objects** command. MGRID prompts you to define the optimization variable.

Step 4 Select **Vertices Mapped to: New Variable**. Enter the **Tuning Angle = 0** degree and select **OK**. We would like to change the x-coordinates of the selected vertices.

Step 5 Move the mouse to somewhere on the left hand side of the selected vertices. Click the left button. Enter "-200" mils for the **Low Bound** of the variable.

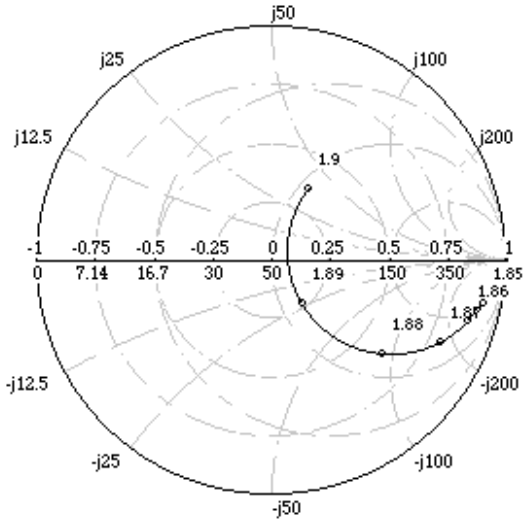


Figure 13.3 The Smith Chart display of the initial design.

Step 6 Move the mouse to somewhere on the right hand side of the selected vertices. Click the left button. Enter "200" mils for the **High Bound** of the variable.

The optimization variable for the length is defined. We can use it to tune the patch's resonant frequency. However, we cannot guarantee the match. We would like to change the length of the insets simultaneously for the perfect match.

Step 7 Select **Edit->Select Vertices** command again. Select the vertices using the Window 2 in Figure 13.2b. We would like to control the length of the insets by moving the vertices at the middle.

Step 8 Select **Optim->Variable for Selected Objects** command. MGIRD will prompt you there are total 22 vertices and 8 distinctive vertices selected.

Step 9 Select **Vertices Mapped to: New Variable**. Enter the **Tuning Angle = 0** and select **OK**. We would like to change the x-coordinates of the selected vertices.

Step 10 Move the mouse to somewhere on the left hand side of the selected vertices. Click the left button. Enter "-200" mils for the **Low Bound** of the variable.

Step 11 Move the mouse to somewhere on the right hand side of the selected vertices. Click the left button. Enter "200" mils for the **High Bound** of the variable. The optimization variable for the length of the insets is also defined. We can use it to tune the patch's matching.

Step 12 Save the geometry as **.\ie3d\practice\ar_ele1.geo**. We will optimize Re[S(1,1)] = Im[S(1,1)] = 0 at 1.88 GHz. Select Set Optimization in Process menu. Enter 1 frequency at 1.88. Select Add button. Define the **Parameter Type** as Re(S), **1st Parameter** as (1, 1). **Objective Type** as **Optimization Quantity** = Objective1, **Objective1** = 0. Select OK to add the 1st optimization goal. Select Add button again. Define the **Parameter Type** as Im(S), **1st Parameter** as (1, 1). **Objective Type** as **Optimization Quantity** = Objective1, **Objective1** = 0. Select OK to add the

2nd optimization goal. Select **Powell** for the optimization scheme. Select OK to start the optimization.

The default generation number is 565 times. Interestingly, the optimizer achieves the goals in only tens of simulations (You can check the **.\ie3d\practice\output\ar_ele1.log** file for the optimization information). The best result is saved **in .\ie3d\practice\ar_ele1m.geo**.

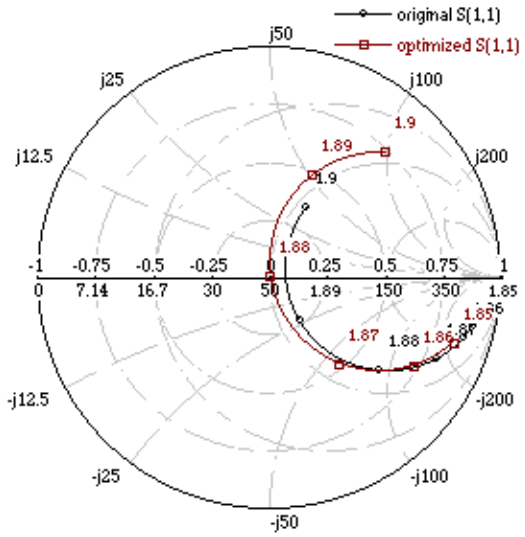
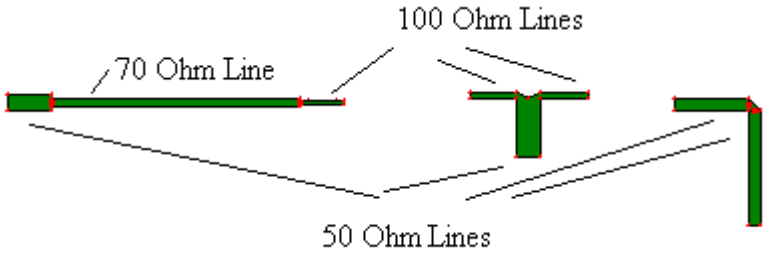


Figure 13.4 The comparison between results before and after optimization.

Step 13 Open the optimized geometry **.\ie3d\practice\ar_ele1m.geo**. Simulate it from 1.85 to 1.9 GHz. We will use it later. Display the result before and after the optimization using MODUA. You will see the optimizer really did the job (see Figure 13.4).

Section 13.2 The "Complete Array" Approach

For convenience reason, we will discuss the "Complete Array" approach in the array analysis. We would like the element-to-element distance from center to center to be 3,500 mils. This distance is a little bit more than half of a wavelength in free space (about 3,140 mils). No grating lobe is created because the element pattern will suppress it. The input impedance for each element is 50-ohms. We will combine each pair of the elements. Before we combine them, we need to transform the 50-ohms to 100-ohms. Therefore, we need a transformer from 50-ohms to 100-ohms and we need a power combiner from two 100-ohm lines back to 50-ohm line. The geometry file for the transformer is saved in **.\ie3d\samples\ar_trans.geo** (see Figure 13.5a). The power combiner is saved in **.\ie3d\samples\ar_comb.geo** (see Figure 13.5b). The **ar_trans.geo** is a quarter wavelength power transformer with the middle section of Zc = 70-ohms and quarter of a wavelength. The total length for **ar_trans.geo** is 1,200 mils. The total length for the **ar_comb.geo** is 300 mils. We can use the Symmetrical T-Junction and Symmetrical Step in Entity of MGRID 4.16 to build the 2 structures easily. Another small geometry we want to be handy is a 50-ohm bend. The geometry is saved in **.\ie3d\samples\ar_bend.geo**. The horizontal length of the bend is 400 mils and the vertical length of the bend is 600 mils. The 2 dimensions, especially the horizontal length, are planned for the element-to-element separation.



(a) Transformer (b) Power Combiner (c) Bend

Figure 13.5 The 3 small structures need to be handy.

Step 1 Open **.\ie3d\practice\ar_ele1m.geo**. Drop the optimization variables in the geometry file. Save it into **.\ie3d\practice\ar_comp.geo**. Select **Port->Delete All Ports** command to delete all the ports.

Step 2 Run another MGRID. This MGRID will be referenced as the 2nd MGRID. Open **.\ie3d\samples\ar_bend.geo**. Press down "shift" and window all the polygons in **ar_bend.geo**. Select Copy in Edit menu. We are going to copy the bend from ar_bend.geo to ar_ele1m.geo.

Step 3 Go the 1st MGRID. Select **Edit->Paste** command. The bend is moving with the cursor. Move the cursor to the feed line. The bend will be snapped to the feed line. Click the left button to locate the pasted bend. MGRID will prompt you for the offset values for the coordinates. Accidentally, they are 0's. Normally, you do not need to change them. MGRID automatically snap vertices. Select OK to continue. The bend will be copied and connected to the feed of the patch (see Figure 13.6).

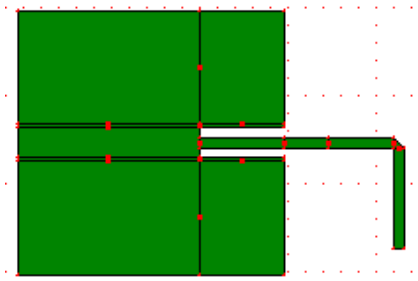


Figure 13.6 The patch with the bend connected.

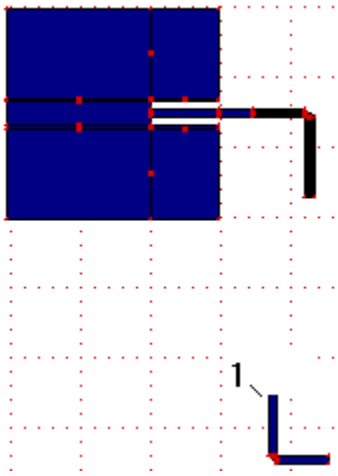


Figure 13.7 The patch, the original bend and the copied and rotated bend.

Step 4 Press down "Shift" button and window the copied bend to select it. Select **Edit->Copy and Rotate** command. Enter Object Rotation Angle = 180 degrees. Enter the Reference Point: X-Coordinate = -4000 and Y-Coordinate = 4000. Select Copied Objects Selected for After Rotation. Select OK. The bend is copied with 180-degree rotation (see Figure 13.7). The copied and rotated bend is still being selected. Please note that the Reference Point coordinates are arbitrarily chosen.

Step 5 Select **Edit->Move Objects** command. The copied and rotated bend is moving with the cursor. One of the vertices is the moving reference and it is at the location of the cursor. It may not be the vertex 1 in Figure 13.7. Click the right mouse button multiple times until the vertex 1 becomes the moving reference. Move the bend toward the end of the original bend until it snaps to it and connect. Click the left mouse button to locate the 2nd bend. MGRID will prompt you to change the default offset values. Select Objects De-Selected for the After Move option. Select OK to continue. The 2nd bend is connected to the end of the 1st bend (see Figure 13.8). We will save the change into ar_comb.geo. What we need to do is to copy the ar_trans.geo and the ar_comb.geo to connect them as shown in Figure 13.8.

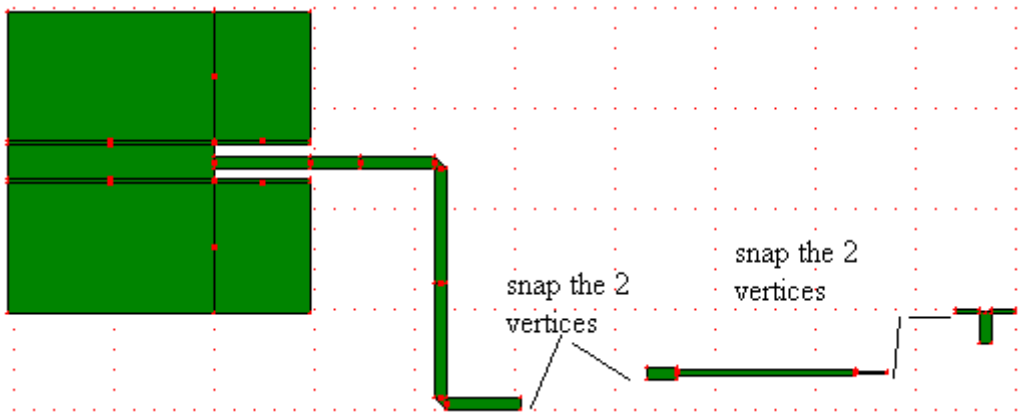


Figure 13.8 The illustration of the different components.

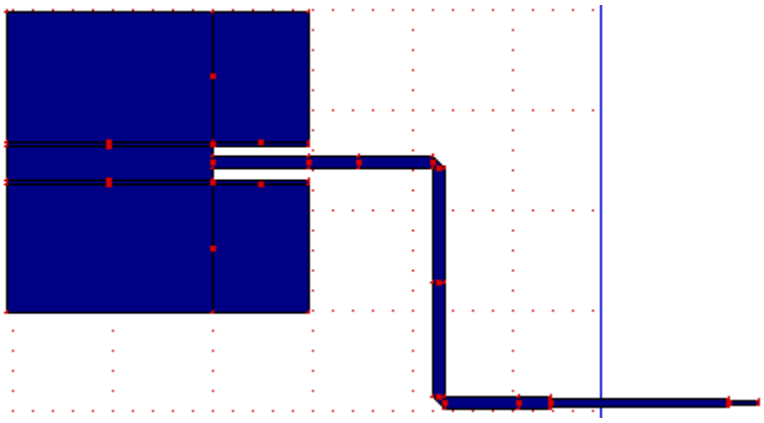


Figure 13.9 The ar_trans.geo is connected to the feed line.

Step 6 On the 2nd MGRID window containing the ar_bend.geo. Select **File->Open** command and open ar_trans.geo. Select **Port->Delete All Ports** command to delete the two ports. Press down "Shift" and window all the polygons. Select Copy in Edit menu.

Step 7 On the 1st MGRID window containing ar_comp.geo, select Paste in Edit menu. The ar_trans.geo is following the mouse. One of the vertices is chosen as the moving reference. Click the left t mouse multiple times until one of the two vertices on the left end of the ar_trans.geo becomes the moving reference. Move it toward the end of the feed-line of the patch antenna, until it snaps. Then, click the left mouse button to locate it. The x-offset should be –3500 and the y-offset should be 757.5. Select OK to continue. The ar_trans.geo is connected to the end of the feed line of the patch.

Step 7 On the 2nd MGRID window, open .\ie3d\samples\ar_comb.geo. Press down "Shift" button and window all the polygons to select them. Select **Edit->Copy** command.

Step 8 Go to the 1st MGRID. Select **Edit->Paste** command. The polygons from ar_comb.geo are following the cursor. Click the right mouse button multiple times until one of the two vertices on the left edge of ar_comb.geo becomes the moving reference. Move it until the ar_comb snaps to the end of the transformer. Click the left mouse button. MGRID will prompt you to change the offset values. Accidentally, they are 0's. Select OK to continue. We will get the picture in Figure 13.10. The next step is to copy and reflect the 2nd bend and the transformer to the right hand side to get the picture in Figure 13.11.

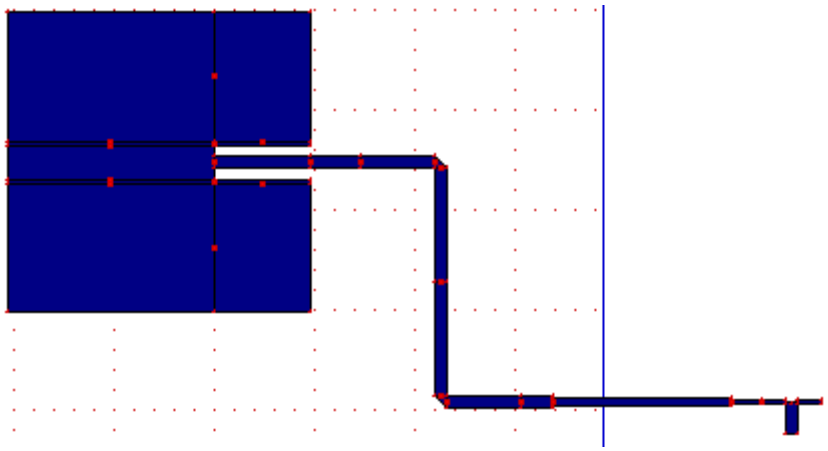


Figure 13.10 The ar_comb.geo is connected to the transformer.

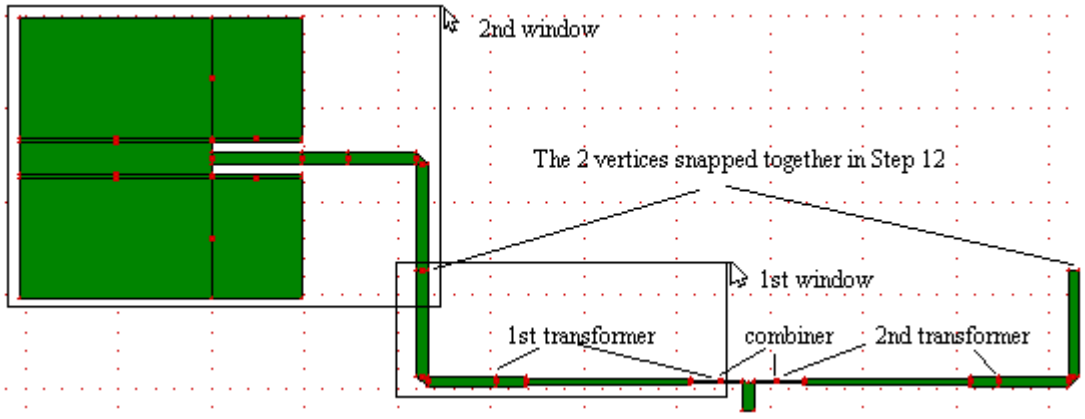


Figure 13.11 The second transformer is copied and connected to the combiner.

Step 9 Press down "Shift" and windows the 2nd bend and the transformer to select them. Select **Edit->Copy and Reflect** command. MGRID will prompt you for the Object Reflection Angle. Enter

0 for the angle. Select OK to continue. The copied and reflected polygons are following the mouse cursor. Click the left mouse somewhere. MGRID will prompt you for the distance. Enter the distance as 300 mils, which is the length of the power combiner. Check All Objects De-Selected. A transformer and a bend are created on the right hand side of the power combiner. The connection between the power combiner and the second transformer is guaranteed, because we have entered the exact length value of 300 mils. The picture is shown in Figure 13.11. Save the file as: .\ie3d\practice\ar_comp1.geo.

Step 10 Press down "Shift" button and window the patch and the 1st bend (see the 2nd window in Figure 13.11) to select them. We will duplicate them for the second patch. Select **Edit->Copy** command. The **Moving Reference Vertex** for the copied polygons is far from the lower right corner of the bend, which we want to be the **Moving Reference Vertex**. In the MGRID 8.X or earlier version, we had the command **Input->Set Moving Reference Vertex** command to select the **Moving Reference Vertex** quickly. This command is no longer valid after we implemented the Paste command in MGRID 9.0. Instead, we implemented a new command called **Input->Shift Moving Reference Vertex** command. If the pasted/moved polygons have more than 1 polygon, the **Shift Moving Reference Vertex** will shift the **Moving Reference Vertex** to the next polygon. Otherwise, it will shift it to the next vertex. The accelerator for the **Input->Shift Moving Reference Vertex** command is the "tab" key.

Step 11 Click the "tab" key several times until you see the 1st vertex in the polygon of the lower right of the 2nd bend is the Moving Reference Vertex. Then, click the right mouse button until one of the two vertices at the lower right end becomes the Moving Reference Vertex. Move the pasted polygons to snap to the right end of the structure until they snap. Click the left mouse button to locate it. MGRID should prompt for the X-offset and Y-offset values. Make sure the X-offset value is 3500 mils and the Y-offset value is 0. Select OK to continue. We will get the structure in Figure 13.12. Save the file as: .\ie3d\practice\ar_comp2.geo.

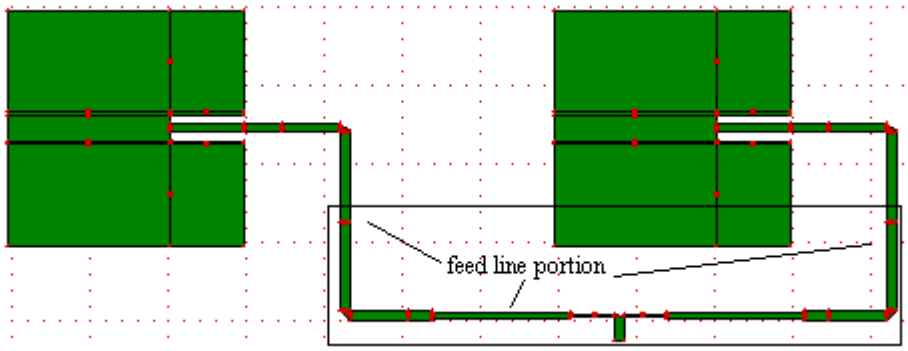


Figure 13.12 The 2 patches in the linear array.

Step 12 From here to the 16-element array is easy. Please select **Edit->Select Polygon Group** command. Window all the feed line portion of the structure, as shown in Figure 13.12. The feed line portion of the structure includes the power divider, the 2 transformers and the 2 bends, each connected to the transformer. In fact, it is the whole structure without the 2 patches and the 2 bends. Copy, Paste and the "tab" button to connect them as shown in Figure 13.13. The X-offset = 1750 mils and the Y-offset = -757.5 mils.

The copied feed line, which spans about 3,500 mils, is too short. We need to double its length, in order to accommodate the 4 patches.

Step 13 Select **Edit->Select Vertices** command. Window the vertices as shown in Figure 13.13 (the window for the 1st selection). Then, select **Edit->Move Objects** command. Click somewhere and enter the X-offset = 1,750 mils = 3,500 / 2 mils, and Y-offset = 0 mil. Uncheck the **Object**

De-Selected before selecting **OK** to exit the selection mode. The left arm of the second feed network is stretched.

Step 14 Select **Edit->Select Vertices** command. Define the second window in Figure 13.13 to select the vertices. Then, select **Move Objects** in **Edit** menu. Click somewhere and enter the X-offset = 1,750 mils = 3,500 / 2 mils, and Y-offset = 0 mil. Uncheck the **Object De-Selected** before selecting **OK** to exit the selection mode. The right arm of the second feed network is stretched. (see Figure 13.14). Save the file as: .\ie3d\practice\ar_comp3.geo.

Step 15 Select the 2 patches and the feed network in Figure 13.12 and use the Copy, Paste and "tab" commands to create the 4-patch array in Figure 13.15. The X-offset = 7000 and the Y-offset = 0. Save the file as: .\ie3d\practice\ar_comp4.geo. You should also use the **Whole Circuit** in **View** menu or the 4 extend layout commands in **Param** menu to expand the view size in order to keep up with the expanded layout.

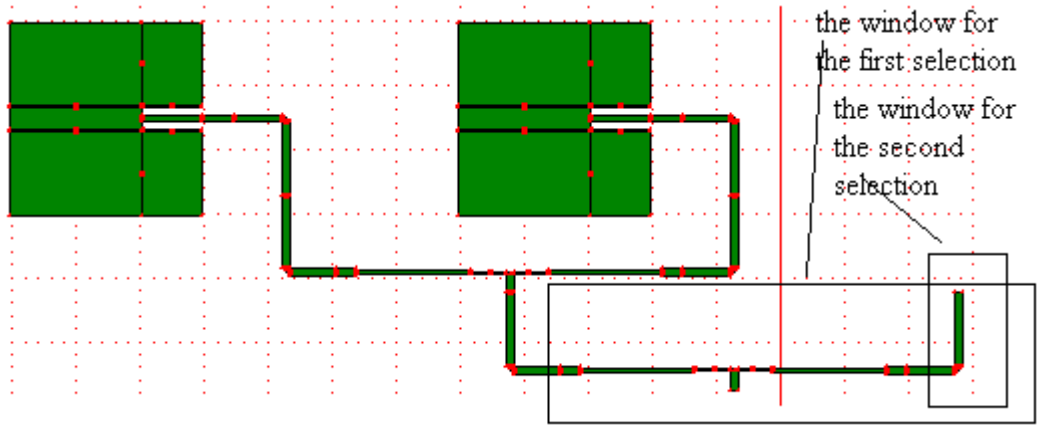


Figure 13.13 The copied feed line connected to the original feed line.

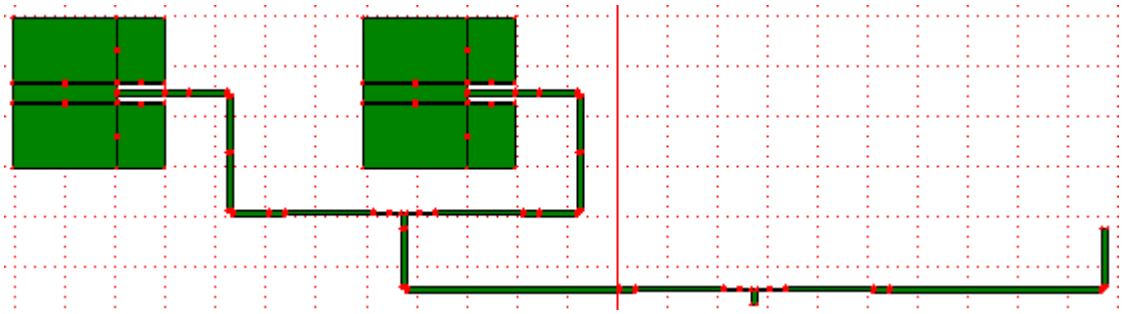


Figure 13.14 The second feed line is stretched for the 4 patches.

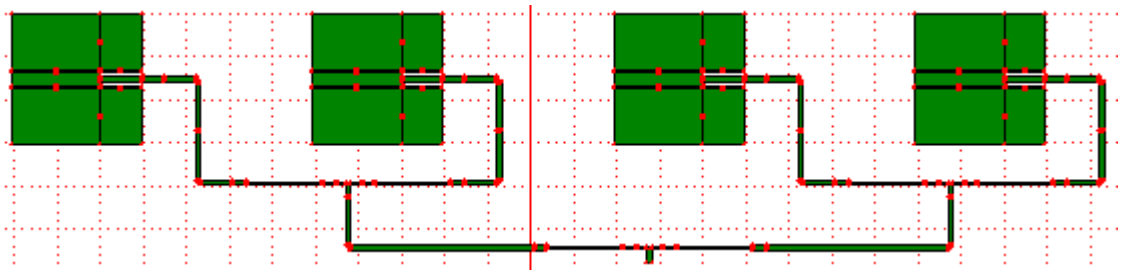


Figure 13.15 The 4 patch array is created.

Step 16 For a few more steps, you should be able to create the 16-element linear array shown in Figure 13.1. The steps will be similar to those from 2-patch array to 4-patch array. The 8-element array is saved as **.\ie3d\practice\ar_comp4.geo** and the 16-element array with 1-port defined is saved as: **.\ie3d\practice\ar_comp5.geo**.

By this step, we have created the complete antenna array. Simulation of the complete structure takes 2857 cells and 4041 unknowns. It requires about 136 MB RAM using the SMSi solver. The simulation time is about 185 seconds per frequency point on Pentium 4 2.8 GHz. Only about 33 seconds is used to mesh the geometry and fill the matrix. Using the AIMS III with Separation Distance = 10 Cells and Pre-Conditioner = "2 0.0007", the simulation time is reduced to 97 seconds. The matrix fullness is about 2.32% and the RAM requirement is about 33 MB only. You should understand that the AIMS II and AIMS III yield the accurate result without neglecting the far coupling.

In the simulation, you should enable **Radiation Pattern File** in the simulation setup. IE3D will simulate the structure and find the radiation pattern automatically for you. When you enable **Radiation Pattern File**, MGRID will prompt you for the pattern calculation parameters. Please enter total 73 data point from 0 to 180 degrees for the Theta angle, and total 73 data points from 0 to 360 degrees for the Phi angle. The radiation pattern calculation may take couple extra minutes.

Section 13.3 The "Separation" Approach

As you can see from Section 2, the complete approach is a 4,039 unknown. It is the most accurate modeling. The result is the 1-port s-parameter or the input impedance of the array. We cannot get any information about the power distribution on the feed network. It would be nice if we can know the power, voltage and current distribution on the feed network. We certainly can do it using the "**Separation**" approach as we discuss in the following.

Step 1 Open .\ie3d**practice\ar_comp5.geo** (or **.\ie3d\samples\ar_comp5.geo**). Delete the 16-patches and the 16-bends connected to the patches. Define the ports on the terminals. The 17-port feed network is created and shown in Figure 13.16. Please save it as **.\ie3d\practice\ar_feed.geo**.

Some people may be curious about the return loss, the power distribution and the radiation of the feed network. We will show how we can get the information.

Step 2 Open .\ie3d**practice\ar_comp5.geo**. Delete the feed network except the 16-patches and the 16-bends connected to the patches. Define the ports on the terminals. The 16-port feed network is created as shown in Figure 13.17. Please save the geometry as **.\ie3d\practice\ar_ps.geo**.

We are going to perform separate electromagnetic simulations on both the **ar_feed.geo** and the **ar_ps.geo**. Certainly, the simulations cannot include the coupling between the portion of elements and the feed network, even though the coupling between elements and the coupling inside the feed network will be included.

On the IE3D 6.0, we can simulate the two structures separately. Then, we connect the s-parameters of the two separated structures using a similar design setup shown in Figure 3.18. We can use the **Process->Simulate and Find Excitation** command in MODUA to find and save the voltage and current at each port when the elements are fed by the feed network. Finally, we can use MGRID or CURVIEW to calculate the radiation pattern from the elements using the excitation file saved from MODUA. The radiation from the feed network will be neglected. It is a valuable feature of the IE3D because it provides a sophisticated tool to fine tune the array. However, the process is a multi-step process and it neglects the radiation from the feed network. Also, the final radiation pattern created does not include the feed-network as part of the

structure. Information such as return loss, gain and efficiency does not include the effect of the feed network.

Starting from the IE3D7.0, we are able to automate the process. We are able to merge the patterns of the elements and the feed network automatically, in a one step process. Every element in the whole design will be considered as part of the system.

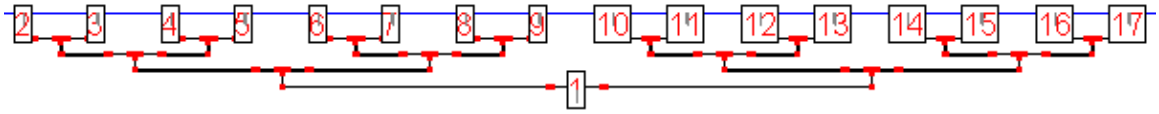


Figure 13.16 The 17-port feed network without the elements.

Figure 3.17 The 16-port structure of array elements without feed network.

Section 13.4 Pattern Merging and Simulation of Large Arrays

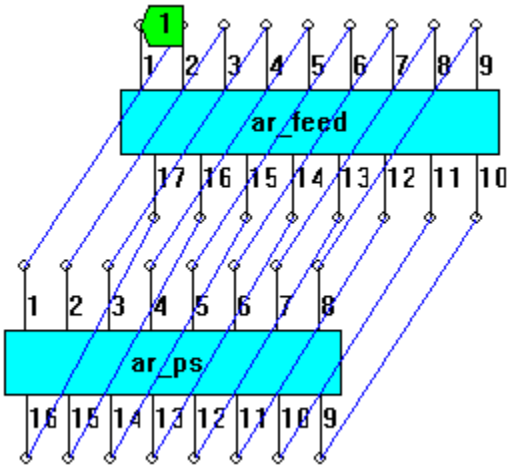


Figure 13.18 The connection between the elements and the feed network.

Step 1 Run MODUA. Select **Add Geometry Module** in **File** menu. Select the file **.\ie3d\practice\ar_feed.geo**. Select OK twice and click at a spot to drop it. Click an empty spot to de-select it. Select **Add Geometry Module** in **File** menu again. Select the file **.\ie3d\practice\ar_ps.geo**. Select OK twice and click at a spot to drop it. Click an empty spot to de-select it. Select **Connection** in **Element** menu to define the connections one by one shown in Figure 3.18. Select **Exit Element** in **Element** menu to exit from the mode. Select **Define All Ports** in **Element** menu to define the port 1 in Figure 13.18. Save the design file as: **.\ie3d\practice\ar_fd_ps.dsg**.

Step 2 Select **Simulate** in **Process** menu of MODUA. Enter 1 frequency point at 1.88 GHz. Select OK to continue. The simulation setup dialog similar to the one in MGRID comes up.

Step 3 Check the **Radiation Pattern File** checkbox. The **Current Distribution File** checkbox will be automatically checked after you check the **Radiation Pattern File**. The **Radiation and Excitation Parameters** dialog comes up.

Step 4 Enter **Start Angle** = 0, **End Angle** = 180, **Number of Angles** = 73. Select **Add Theta** to add the angles to the elevation angles. Enter **Start Angle** = 0, **End Angle** = 180, **Number of Angles**

= 73. Select **Add Phi** to add the angles to the azimuth angles. Select OK to finish defining the radiation and excitation parameters.

Step 5 Since this structure has many ports, it might be better to use the AIMS II instead of the AIMS III. Select the AIMS II matrix solver. MGRID will prompt you for the parameters. Select the Separation Distance = 10 cells. However, the actual SD = 2950 mils (different from the 1796 mils in **ar_comp.geo**). It is due to the fact that MODUA use the cell size in free space as the criteria. We can change the Separation Distance in Cells to 7 and it is about 2000 mils.

Step 6 Select OK to invoke the IE3D simulation on the complete design.

The IE3D will simulate the 2 structures sequentially. The feed line takes about 1127 cells and 1143 unknowns. It takes about 22 seconds (2-iterations) per frequency point for the feed line. The patch creates 1856 cells and 3040 unknowns. It takes about 247 seconds (5-iterations) for each frequency point. Totally, it takes about 270 seconds per frequency point for the s-parameters. However, the pattern calculation takes about 150 seconds per frequency point. You may notice that the "Separate" approach takes much longer time than the "Complete Array" approach. It is due to the 5-iterations in simulating the ar_ps.geo. It is also due to the fact it requires many more calculation for the pattern due to the larger number of ports.

The final results of the whole simulation include: the s-parameter files (*.sp), the current distribution files (*.cur), the radiation pattern files (*.pat) of the **ar_feed.geo** and **ar_ps.geo** structures. For the design combining the two structures, we will have the **ar_fd_ps.spm** file for the s-parameters, and the **ar_fd_ps.pat** for the radiation pattern, and the **ar_fd_ps.ect** for the excitation information for the complete structures. The intermediate result files from **ar_feed**, **ar_ps** and **ar_fd_ps** can be used for more detail information for the complete structures. For example, you can open **ar_ps.cur** file and attach the **ar_fd_ps.ect** file to calculate the radiation pattern of the elements assuming the radiation from the feed network (**ar_feed.geo**) is negligible. You can also open **ar_feed.cur** file and attach the **ar_fd_ps.ect** file to investigate the radiation from the feed network. You should understand that such an approach on pattern post-processing is an approximation. The radiated power of the complete structure will be different from the sum of powers of the two structures when they exist individually. More discussion on this issue can be found in Section 7 of Chapter 8.

Step 7 Use an ASCII editor to open the **ar_fd_ps.ect** file. We will get all the information about the excitation and termination at the ports of the feed network and the antenna elements.

As you can see from the above procedure, we can use this approach to model very large arrays with non-identical elements. As we will see in the next section, we can use array factor to model arrays with uniformly spaced identical elements. However, the array factor approach cannot be applied to any of the following situations: (1) The elements are not identical, (2) The rotation or orientation of each element is not the same, and (3) The elements are not uniformly spaced. Instead, the "separation" approach allows more flexibility in modeling general arrays. Both approaches do not include coupling between elements (or simulation blocks).

Section 13.5 The "Array Factor" Approach

We have not created the models for the "Array Factor" and "Periodical Walls" approaches. Creating the single element from **ar_ps.geo** is simple. You can just delete all but 1-element. The remaining element will be the single element. The single element file is saved as **.\ie3d\samples\ar_p1.geo**. Simulating **ar_p1.geo** takes almost no time. To get the array radiation pattern is simple. We can calculate the element pattern on either IE3D (invoked by MGRID) or by the Pattern Calculation on MGRID as post-processor. Then, we can use PATTERNVIEW to add the array factor.

Run MGRID. Open **ar_p1.geo**. Simulate it again at 1.88 GHz. Please remember to enable Radiation Pattern File. The IE3D will be invoked to simulate it and find the single element pattern in .\ie3d\practice\ar_p1.pat file. The PATTERNVIEW will be invoked to display the pattern. The ar_p1.pat file is added into the pattern list automatically.

On the PATTERNVIEW, select **Edit->Array Pattern Calculation** command. The **Antenna Array Parameters** dialog comes up (see Figure 13.20). Please select **Add Theta** button to define 73 angles from 0 to 180 degrees. Please select **Add Phi** button to define 73 angles from 0 to 360 degrees. Then, select the **Add** button. The Add Elements dialog comes up. On the X-Location column, enter From = 0, To = 52500, Number = 16, Phase Step = 0. Please do not change the default values in Y-Location column and Z-Location column (From =0, To = 1, Number = 1, Phase Step = 0). Please enter the common parameters as: Magnitude = 1, Start Phase = 0, Element Index = 0, 1st Phi = 0, 1st Theta = 0 and 2nd Phi = 0.

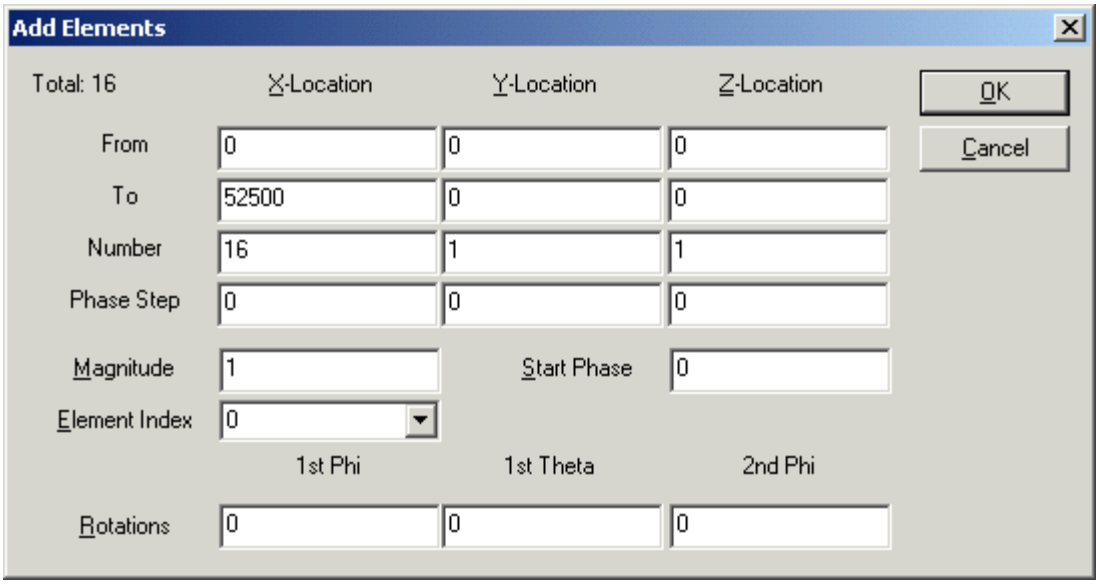


Figure 13.19 The Add Elements dialog after we entered the parameters.

What do the parameters mean? For the X-Location, we want to add 16-elements in the X-direction. The 1st element will be located at X = 0 and the last element will be located at X = 52500. The distance between adjacent elements is 52500/(16-1)= 3500 mils. The Phase Step = 0 means that there is no tapered phase in the elements. The meanings of the Y-Location and Z-Location are similar to the X-Location. All the elements have the magnitude of 1. The Start Phase defined the No. (1, 1, 1) element's phase. The phase of any element (ix, iy, iz) can be calculated as: Phase = Start_Phase + X_Phase_Step * (ix-1) + Y_Phase_Step * (iy-1) + Z_Phase_Step * (iz-1). The Rotations parameters define how the elements are rotated.

There is an additional parameter: Element Index. It is for the index of a specific pattern in the pattern list window. Basically, PATTERNVIEW allows a user to combine the radiation patterns of different elements with different rotations. It is extremely flexible. In our example, we only added one element into the pattern list. We want to find the pattern of a uniform array of the only element. However, the user should understand that you could achieve much more on the PATTERNVIEW.

After you select OK in the Add Elements dialog. The 16-elements are added into the list in the Antenna Array Parameters dialog (see Figure 13.20). Each element is assigned with amplitude, phase, rotation parameters, offset parameters and element index. Old IE3D users should be very careful on the number of elements. In the CURVIEW program, which is out phased and still available on IE3D 9.0, the number of elements added is not the total number of elements. The total elements are the added elements

plus the original one. On the PATTERNVIEW, the number of elements added is the total number of elements of the array.

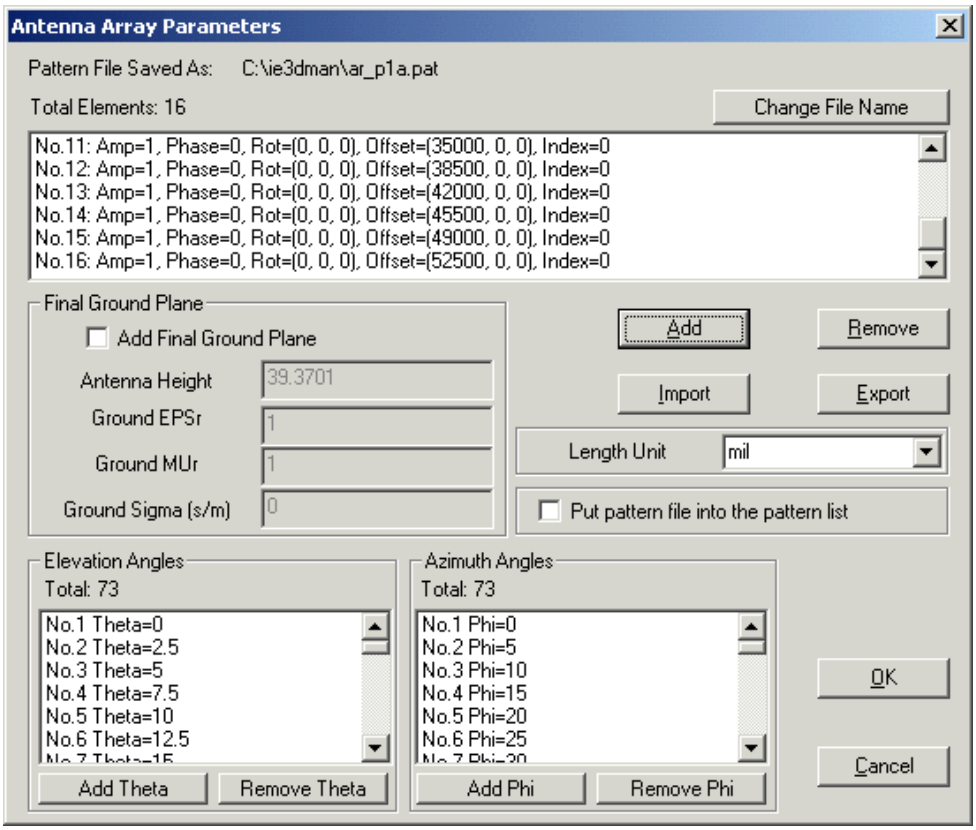


Figure 13.20 The Antenna Array Parameters dialog after the parameters are entered.

If you need to add more elements, you can select Add button again. No only that, a user also has the option to Add Final Ground Plane. This option allows the user to add the true ground plane's effect into the pattern when the ground plane is in the far-field zone.

Please check the Put Pattern File into the Pattern List. It allows PATTERNVIEW to save the array pattern file into the default name: ar_p1a.pat file and add the file into the pattern list automatically. Select OK to continue. PATTERNVIEW will perform the pattern calculation for you. It takes a second to finish it. Then, the resulting ar_p1a.pat file will be added into the list. You can display the pattern parameters on ar_p1.pat and ar_p1a.pat. We will delay the discussion on them to later sections.

Section 13.6 The "Periodic Walls" Approach

There are two ways to model the periodic walls: (1) Model it as a 1-dimensional infinite array using 1 pair of periodic walls; (2) Model it as a 2-dimensional infinite array using 2 pairs of periodic walls. For a large planar phased array, we should undoubtedly choose the 2-dimensional infinite array approach. For a 1-dimensional linear array, the 1-dimensional infinite array is supposed to be the appropriate one. However, it seems to us that it is better to choose the 2-dimensional infinite array approach on the IE3D 8.0 even for 1-dimensional linear array.

On the IE3D 8.0, we have implemented the periodic Green's functions for the 2D planar array case. Its simulation result is precise. For the 1D linear array, we use a pair of parallel periodic walls to approximate it. Unfortunately, we have not implemented the exact Green's functions for the one pair of

parallel periodic walls for it. We also do not have the intention to implement it in the near future. We are using finite number of images to approximate it. On the other hand, it might be equally well or even be better if we use the 2D periodic Green's function to approximate it. The 2D periodic Green's function uses two pairs of parallel periodic walls for the modeling. An infinite number of images are included and the results are quite stable. If we make the 2nd pair of parallel walls to be far enough, the results should approach the one for 1 pair of parallel walls.

Creating the files for the "Periodic Walls" approaches is quite simple. We open the single element file **.\ie3d\samples\ar_p1.geo**. Then, we will define the periodic walls. We select the **Param->Basic Parameters** command. Double click at the **No. 0: No Side Walls** in the Enclosure section to edit it (see Figure 13.21). Select **Xmin Wall** as **Periodical** at –6,600 mils. Select **Xmax Wall** as **Periodical** at –3100 mils. For the X-Direction, please enter the **Pattern Images** = 15 (the number of images used in the pattern calculation), the **Spatial Images** = 4 (the coupling included in the simulation), the Phase Factor = 0 degree (the phase difference between adjacent cells), the **Image Index = 7**. This is a finite array. The Image Index = 7 means that the element will be considered as the No.7 element in the 16-element array. We will get the dialog in Figure 13.21. Totally, we have 16-elements in consideration. Please save the file as: **ar_pp.geo**.

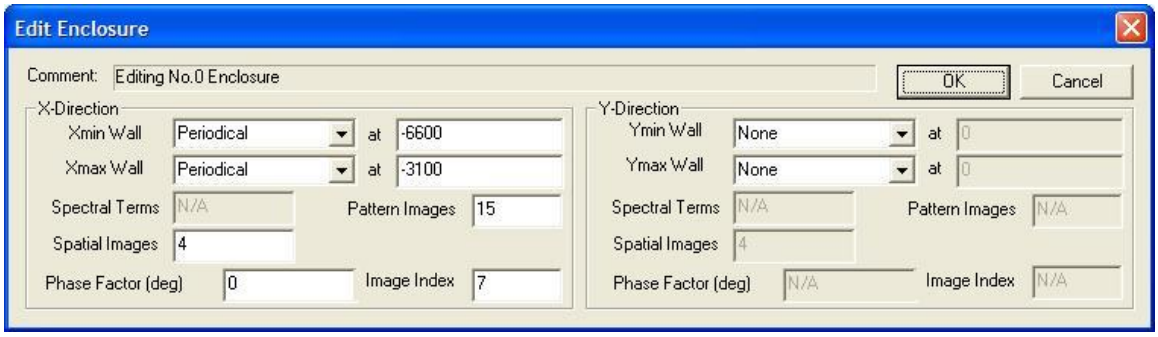


Figure 13.21 The Edit Enclosure dialog for ar_pp.geo.

Please select **Param->Basic Parameters** command and edit the **No.0 : No Side Walls** in the Enclosure section again. Please select the **Ymin Wall** as **Periodical** at 0 and the **Ymax Wall** as **Periodical** at 10000 mils. Please define the Y-Direction Parameters as: Pattern Images = 0 (No image in the y-direction).

After we define all the 4 side walls as **Periodical** walls, the Spectral Terms in both X-Direction and Y-Direction are active. They are controlling the accuracy for the matrix elements in the simulation. Normally, their default values are 15. Please define X-Direction: Spectral Terms = 15 (default) and the Y-Direction Spectral Terms = 30 (larger value may improve accuracy for larger span between the walls) (see Figure 13.22). Select OK and we will get the 2D periodic walls defined as shown in Figure 13.23. Save the geometry file as: **ar_pp2.geo**. The **ar_pp2.geo** is the 2D periodic wall approximation to the 1D array.

Please simulate the **ar_pp.geo** and **ar_pp2.geo** at 1.88 GHz. Please enable the **Radiation Pattern File** in the simulation setup. IE3D will calculate the pattern and save the pattern into **ar_pp.pat** and **ar_pp2.pat** files, respectively.

Section 13.7 The Comparison of Different Approaches

The comparison on the radiation parameters between the 5 schemes is shown in Table 13.3. We have listed three efficiencies: Element Efficiency (dB), the Antenna Efficiency with Feed Network (dB) and the Antenna Efficiency without Feed Network (dB). For the "Array Factor" and the 2 "Periodical Walls" approaches, the efficiency in the radiation parameters is the Element Efficiency. It is the efficiency of the

elements without the feed network. The Efficiency (dB)'s for the two cases are estimated values based upon the results of the "Separation" case.

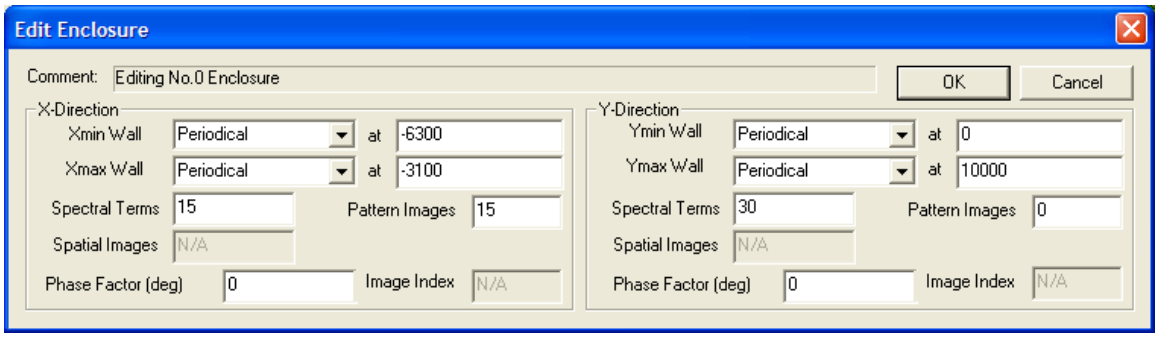


Figure 13.22 The Edit Enclosure dialog for ar_pp2.geo.

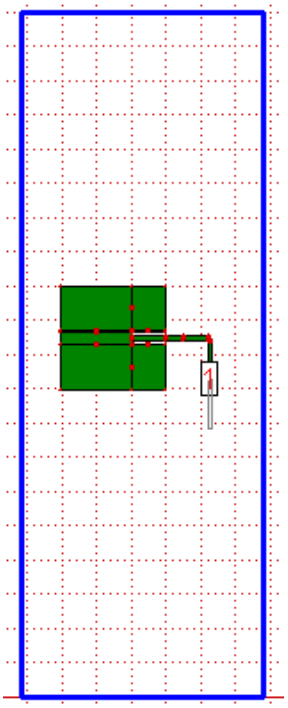


Figure 13.23 The 2 pairs of parallel periodic walls for precise modeling of planar phase array. Large distance between the top and bottom walls is purposely defined to approximate linear phase array here.

For the "Separation" case, the feed network loss is displayed as –2.02 dB. It is calculated based upon the input power at port 1 and the output power at port 2 to port 17 of the ar_feed.geo (listed in Table 13.2). We also use it for the feed loss of the 1st three cases. For the 1st four schemes, we can use the IE3D to simulate the feed network. In fact, we can also use the **Process->Create TLN S-Parameter File** command on MODUA to create the s-parameters for the feed network. For the "Separation" case, the Antenna Efficiency with Feed Network (dB) is the radiation efficiency in the **ar_fd_ps.pat** file.

Certainly, the most accurate method is the "Complete Array" approach. The "Separation" should be the 2nd most accurate approach. The "Periodical Walls" approaches are the 3rd. The "Array Factor" is the simplest approach and the fastest approach. The rule is the more accurate approach normally requires more

computational effort. For this case, the "Separation" approach takes more time. However, "Complete Array" approach will be slower for bigger arrays.

It is not surprising to see that even the "Array Factor" approach yields quite accurate result for this array. The reason is that the element-to-element coupling for this array is low enough. For those arrays with much stronger element-to-element coupling (>> -20 dB), you will see the difference.

Section 13.8 Powerful and Efficient Pattern Calculation and Handling on PATTERNVIEW.

We have demonstrated different ways to analyze antenna array. We can use the PATTERNVIEW to find the radiation pattern of an array based upon the element pattern. The antenna array is not limited to single type of basic elements. It can have different types of basic elements. Each element of the array can be of different shape, different orientation and different excitation. We can also add the final ground' effects.

On the PATTERNVIEW, we can also estimate the field propagation and distribution in some specific locations in the far field zone based upon the far field pattern.

A user can also merge different patterns from different radiators or antennas. We will not demonstrate all the features here. However, interested users can explore the power of the PATTERNVIEW by himself. We will conclude this chapter here.

Table 13.3 The comparison between the 4 approaches at 1.88 GHz.

	"Array Factor" (ar_p1.geo)	"Periodic Walls" (ar_pp.geo)	"Periodic Walls" (ar_pp2.geo)	"Separation" (ar_ps.geo/ ar_feed.geo)	"Complete Array" (ar_comp.geo)
Directivity (dB)	18.95	18.52	18.38	18.73	18.48
Antenna Gain with Feed Network (dB)	N/A	13.08*	13.03*	12.43	11.39
Antenna Gain without Feed Network (dB)	13.55	N/A	N/A	15.10	N/A
Mismatch (dB)	N/A	N/A	N/A	-0.095*	-0.068
Element Efficiency (dB)	47.1%	N/A	N/A	N/A	N/A
Loss in Feed Network (dB)	-2.02*	-2.02*	-2.02*	-2.02*	N/A
Antenna Efficiency with Feed Network (dB)	N/A	N/A	28.2%	23.47%	19.57%
Antenna Efficiency without Feed Network (dB)	40.0%	44.3%	44.8%	39.1%	N/A
3 dB Beam Width (deg)	5.55	5.88	6.41	5.87	5.93
Unknowns	190	190	190	3040	4039
Seconds/Freq(Structure)	0.1	4	21	247 (Aims II)	97 (Aims III)
Seconds/Freq (Feed Network)	22	22	22	22	N/A
Seconds/Freq (Pattern Calc)	1	5	2	150	18
RAM Requirement (MB) (with Feed Network)	1 (N/A)	1 (10)	1 (10)	30 (Aims II)	20 (Aims III)

The "*" 's indicate the values are not direct output from the software. It needs calculation based upon information in the .pat files and .ect files.

Chapter 14 Modeling of Structures in an Enclosure

Traditionally, non-uniform meshing is linked to open Green's function formulations. Uniform meshing is linked to close Green's function formulations. Most of the examples we have discussed are for open Green's function formulations. The IE3D featured with non-uniform meshing and the open Green's functions is very flexible and powerful. However, some microwave circuits are packaged into some enclosures. In fact, enclosure cover can be modeled easily using the open Green's function formulation because the enclosure cover can be considered as dielectric layer. A typical example is the suspended stripline structure in the Chapter 4. The difficulty in modeling enclosure using the IE3D earlier versions is in the modeling of enclosure sidewalls. In the IE3D 7.0 and earlier versions, the sidewalls are approximated using the images. In reality, there should be infinite number of images. However, we could not include infinite number of images using the open Green's function formulation. We have to truncate the infinite image series and the truncation of image series introduces error. In fact, it is found that the finite image approximation using the open Green's functions normally may not be able to model all the effects of the enclosure.

The open Green's function formulations predict the surface wave effect in the horizontal direction. The thicker the substrate is, the more serious the surface wave is. The enclosure walls eliminate the surface wave. Certainly, the difference between the Green's functions for the two cases will become more and more evident with increasing substrate thickness.

The 2nd effect of enclosure is the enclosure resonance. On one hand, the enclosure prevents the radiation from the circuit. On the other hand, the enclosure may introduce unexpected resonances which may change the performance of the circuit completely. Certainly, the open Green's function formulations cannot predict the frequency responses of the circuit close to the resonance of the enclosure. Fortunately, it is a common practice to avoid a circuit to be operating at the frequency range with the enclosure resonances. In this sense, open Green's function formulations are still very useful for circuit simulations especially when the size of the enclosure is big or for the case without an enclosure. However, it certainly is good to have the capability to predict where the enclosure resonances are and how the enclosures will affect the performance of a circuit.

The IE3D features both open Green's function formulation and close Green's function formulation. Compared to other close Green's function formulations, the advantage of the close Green's function implementation on the IE3D is that it still uses the non-uniform meshing scheme. Due to the limitation of the FFT scheme used in other close Green's function formulations, they have to use the uniform meshing. On the IE3D, the non-uniform meshing is kept for the close Green's function formulation. An advanced integration scheme is used in place of the FFT to speed up the calculation of the moment integrals. Thanks to the non-uniform meshing scheme, the close Green's function formulation in the IE3D is as flexible as the open Green's function formulation in the IE3D, even though it is normally significantly slower.

We will discuss different simulation examples comparing the open Green's function formulation and the close Green's function formulation. They will give a guideline on when the open or close Green's function formulation should be used.

Due to the fact that the Advanced Extension scheme is not implemented for the closed and periodical Green's functions in the IE3D 10.0. We will use the Extension for MMIC scheme exclusively in this chapter. This chapter is virtually not updated in IE3D 12 and IE3D 14.

Section 14.1 Convergence Study on the Closed-Boundary Green's Function Formulation

The open Green's function formulation of the IE3D has been used for years. Its accuracy and convergence have been proved by numerous practical examples. We will not discuss its convergence here. What we will be discussing in this section is the convergence study on the closed Green's function

formulation. We will use the chamfered bend example with AEC as an example. The open geometry is saved in **.\ie3d\samples\c_bend3.geo**. The same structure in enclosure is modeled in 3 different files: **c_bend3be5.geo**, **c_bend3be10.geo** and **c_bend3be15.geo** in the .\ie3d\samples directory. They are different from the number of terms used in the simulation.

Step 1 Run MGRID and open the file **.\ie3d\samples\c_bend3be10.geo**. The structure is shown in Figure 14.1. It is the same structure as the one shown in chapter 3, except there is box with red lines enclosing the bend. The box basically represents the side walls of the metallic enclosure. You can open the Substrate dialog from the Param menu. Its dielectric configuration is exactly the same as the open case in **.\ie3d\samples\c_bend3.geo**. Apparently, there is no top cover for the **c_bend3be10.geo**. What are the dimensions of the enclosure walls?

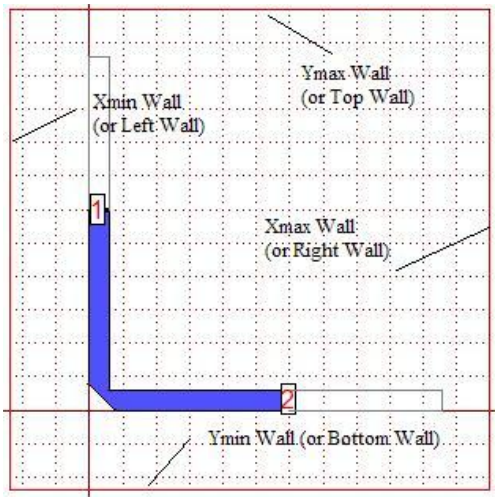


Figure 14.1 The chamfered bend enclosed by a box.

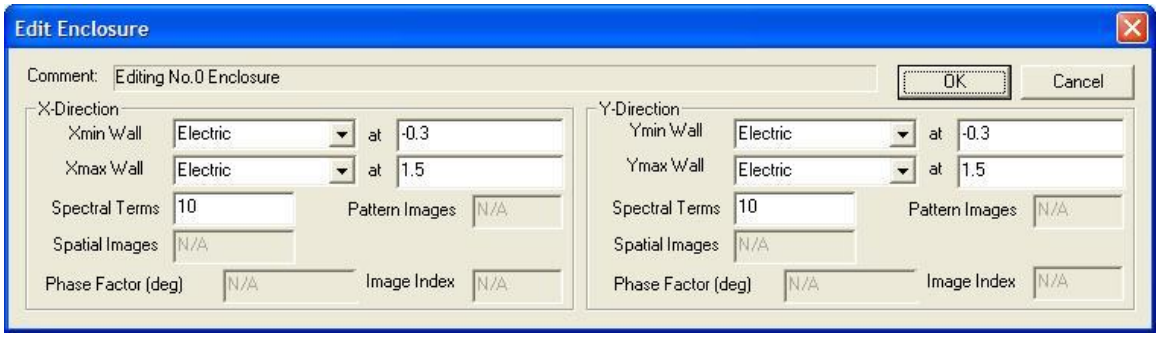


Figure 14.2 The dialog for the enclosure walls.

Step 2 Select **Param->Basic Parameters** command. Double click at the "No.0 X-Wall…" in the Enclosure section. We will see the dialog shown in Figure 14.2.

It shows the parameters of the enclosure wall. The meanings of the Xmin, Xmax, Ymin and Ymax walls are shown in Figure 14.1. The Top Wall and Bottom Wall in Figure 14.1 do not mean the top and bottom covers of the enclosure. The top and bottom covers of the enclosure are defined in the **Substrate** section of the **Basic Parameters** dialog. The Xmin Wall is at X = -0.3 mm. The left edge of the bend is at X = 0 mm. Therefore, the minimum distance from the bend to the walls is 0.3 mm comparing to the substrate thickness of 0.075 mm.

The other important parameters are the **X-Direction: Spectral Terms (Nx) = 10**, and **Y-Direction: Spectral Terms (Ny) = 10.** Before the close Green's function formulation is implemented into the IE3D, The Nx and Ny are used to identify how many finite images in the X- and Y-directions we will use to approximate the side walls. After the close Green's function formulation is implemented, they have different meanings. They represent how many terms we include in calculating the double infinite series in the moment matrix element (or the Zij). Certainly, the more terms we use, the more accurate the Zij is. Without special treatment, the double infinite series in the Zij may converge until Nx and Ny approach 50 to 1000. The convergence of double infinite series is accelerated using some advanced scheme in the IE3D. From our experience, the Zij can be accurately calculated using Nx = Ny = 10 in the IE3D. We may increase the Nx or Ny appropriately with large box dimensions in the X- or Y-direction. For this example, we use Nx = Ny = 5 for the **c_bend3be5.geo**, Nx = Ny = 10 for the **c_bend3be10.geo** and **c_bend3be15.geo** for the Nx = Ny = 15.

Step 3 Select OK multiple times to close the dialogs. Simulate the 3 structures at 80 frequency points from 0.5 to 40 GHz with AEC enabled with edge width = 0.01 mm. Table 14.1 and Figure 14.3 show their comparison on dB[S(1,1)] and Ang[S(2,1)].

Table 14.1 The convergence comparison on the boxed chamfered bend.

File	c_bend3be5.geo	c_bend3be10.geo	c_bend3be15.geo
Nx and Ny	5 and 5	10 and 10	15 and 15
Simulation Time (sec./freq)	4.5	5	5.5
dB[S(1,1)] at 0.5 GHz	-52.43	-52.37	-52.36
Ang[S(2,1)] at 0.5 GHz	-2.374	-2.372	-2.372
dB[S(1,1)] at 40 GHz	-45.99	-45.75	-45.71
Ang[S(2,1)] at 40 GHz	168.6	168.5	168.5

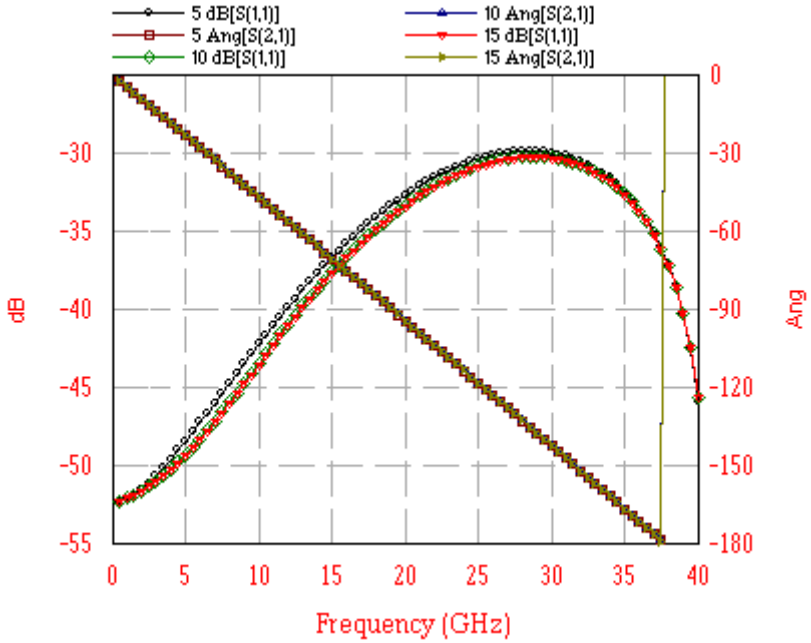


Figure 14.3 The comparison on boxed chamfered bend with different terms.

As you can see, the difference between the 3 cases is very small. The results of the **c_bend3be10.geo** and **c_bend3be15.geo** are almost identical. In fact, there is little difference in the simulation time. For such a small structure, the Nx and Ny are not very important to the

simulation time. The reason is due to the fact that the matrix filling time can be represented roughly as: $Tf = C1 + C2\ Nx\ Ny\ Nc^2$, where C1 and C2 are constants, Nc is the number of cells used in the simulation. For the chamfered bend structure, the Nc = 69 and it is small. Therefore, the 1st term (or the C1) in the formula is much more important than the 2nd term (or the C2 Nx Ny Nc^2) in the formula. When the Nc becomes bigger, reducing the Nx and Ny will significantly reduce the matrix filling time, which is one of the two most time consuming processes in a MOM simulation. The other most time consuming process in a MOM simulation is the matrix inversion, which is even more important. Efficient matrix inversion has been discussed in Chapter 13. The above example demonstrates that Nx = Ny = 10 normally on the IE3D close Green's function formulation normally can yield high accuracy results.

Section 14.2 The Effect of Sidewalls and Cover on Regular Structures.

The last section has demonstrated the fast convergence of the IE3D close Green's function formulation. In this section, we will discuss the effect of side walls and cover for the enclosure. For the example in **c_bend3be10.geo**, there is no cover. The side walls are at least 0.3 mm away from the edge of the traces. An important factor to measure how close the traces to the side walls is the distance over substrate thickness. We will call this ratio as D/T ratio. For the example we are discussing, the substrate thickness (T) = 0.075 mm. Therefore, the D/T = 4 for the **c_bend3be10.geo**. We move the **Xmin Wall** and **Ymin Wall** away to make D/T = 8 for the **c_bend3be10a.geo** and D/T = 12 for the **c_bend3be10b.geo**. A comparison between the 3 cases and the open case (c_bend3.geo discussed in Chapter 3) is shown in Table 14.2 and Figure 14.4. The 1st thing you may notice is that the open Green's function formulation is much faster than the close Green's function formulation. As you can see, the difference among the 4 cases is very small. Apparently, even when the D/T = 4, the open Green's function is a good approximation to such a structure. Such a situation is normally true for non-resonant structures. The 1st conclusion we can make for this chapter is that the open Green's function formulation is normally much faster than the close Green's function formulation, and it should be used most of the time unless it is necessary to consider the effect of the enclosure.

Table 14.2 The comparison on different sidewall distance over substrate thickness (or D/T) ratio.

File	c_bend3be10	c_bend3be10a	c_bend3be10b	c_bend3e
Min. Side Wall Distance (mm)	0.3	0.6	0.9	infinite
D/T Ratio	4	8	12	infinite
Symbol in the Graph	4	8	12	Infinite
Simulation Time (sec/freq)	5	5	5	0.3
dB[S(1,1)] at 0.5 GHz	-52.37	-52.29	-52.28	-52.24
Ang[S(2,1)] at 0.5 GHz	-2.372	-2.382	-2.383	-2.383
dB[S(1,1)] at 40 GHz	-45.74	-38.56	-36.44	-38.04
Ang[S(2,1)] at 40 GHz	168.5	167.1	166.7	166.2

The next comparison we would like to make is on the enclosure cover. The effect is certainly related to the separation from the top cover to the strip. We will investigate 4 cases with different top cover distance (S) to the substrate thickness (T): (1) c_bend3.geo: no top cover; (2) c_bend3c12.geo: The top cover is located at Z = 1 mm, the separation between the top cover and the strip (S) is 0.9 mm. S/T = 12; (3) c_bend3c8.geo: The top cover is located at Z = 0.7 mm and the S/T = 8; (4) c_bend3c4.geo: The top cover is located at Z = 0.4 mm and the S/T = 4. The comparison is shown in Table 14.3 and Figure 14.5. There is obvious difference in the dB[S(1,1)] for the S/T = 4 case even though the dB[S(1,1)] is below –24 dB. Other cases are quite close to each other. Comparing the Figure 14.4 and Figure 14.5, we can see the S/T = 4 case is not as close as the D/T = 4 case to the complete open case (c_bend3.geo). This seems quite reasonable because the S/T = 4 case experiences more coupling from the top cover than the D/T = 4 case's coupling from the side walls.

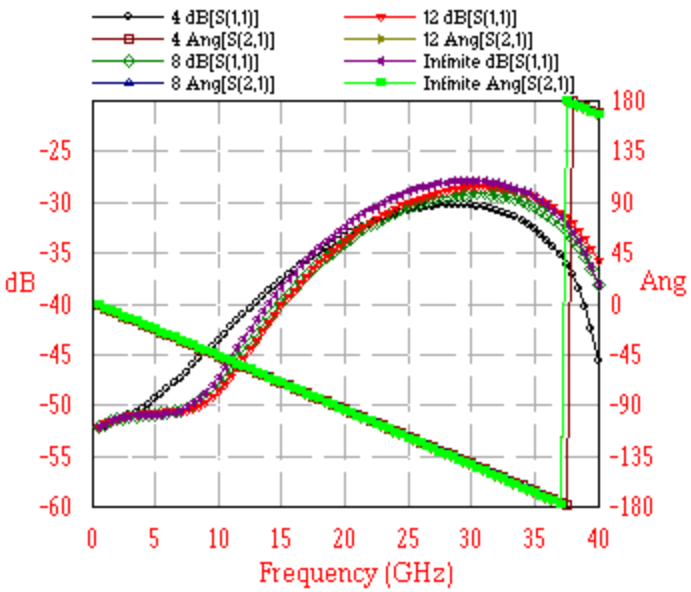


Figure 14.4 The effect of the sidewalls.

Table 14.3 The comparison on the cases with different top cover distance.

File	c_bend3c4	c_bend3c8	c_bend3c12	c_bend3e
Top Cover Distance (mm)	0.3	0.6	0.9	infinite
S/T Ratio	4	8	12	infinite
Symbol in the Graph	4	8	12	Infinite
Simulation Time (sec/freq)	0.4	0.4	0.4	0.3
dB[S(1,1)] at 0.5 GHz	-52.30	-52.31	-52.28	-52.24
Ang[S(2,1)] at 0.5 GHz	-2.350	-2.377	-2.380	-2.383
dB[S(1,1)] at 40 GHz	-39.15	-41.37	-40.90	-38.04
Ang[S(2,1)] at 40 GHz	166.5	166.4	166.4	166.2

Table 14.4 The comparison on the cases with different D/T ratio and S/T ratio.

File	c_bend3c4b	c_bend3c8ba	c_bend3c12bb	c_bend3e
Top Cover Distance (mm)	0.3	0.6	0.9	Infinite
S/T Ratio	4	8	12	Infinite
Min. Side Wall Distance (mm)	0.3	0.6	0.9	Infinite
D/T Ratio	4	8	12	Infinite
Symbol in the Graph	4	8	12	Infinite
Simulation Time (sec/freq)	5	5	5	0.3
dB[S(1,1)] at 0.5 GHz	-52.28	-52.35	-52.32	-52.24
Ang[S(2,1)] at 0.5 GHz	-2.347	-2.375	-2.380	-2.383
dB[S(1,1)] at 40 GHz	-64.57	-39.91	-36.78	-38.04
Ang[S(2,1)] at 40 GHz	169.7	167.3	166.8	166.2

We will compare the cases with top cover and side walls. Table 14.4 and Figure 14.6 are the case with different D/T ratio and S/T ratio. As expected, the larger the D/T ratio and S/T ratio are, the closer to the open case the results are. Significant difference is observed for the case with D/T = S/T = 4 in the dB[S(1,1)], whereas the dB[S(1,1)] is below –28 dB. Our 2nd conclusion for this chapter is that the enclosure (or box) does not have strong effects to many components when the D/T ratio and S/T ratio are not too small. However, there are exceptions and we will discuss the exception cases in Sections 5 and 6.

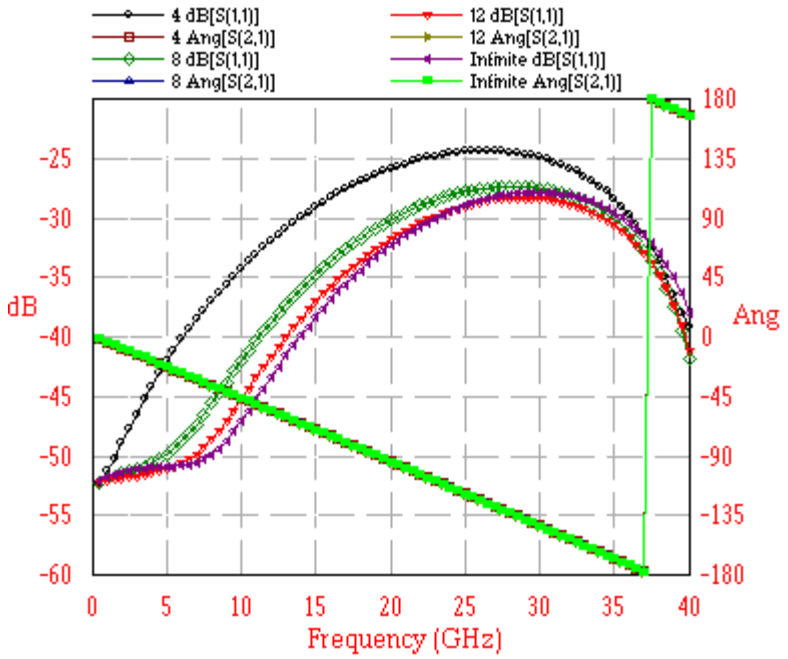


Figure 14.5 The comparison on the cases with different top cover distances.

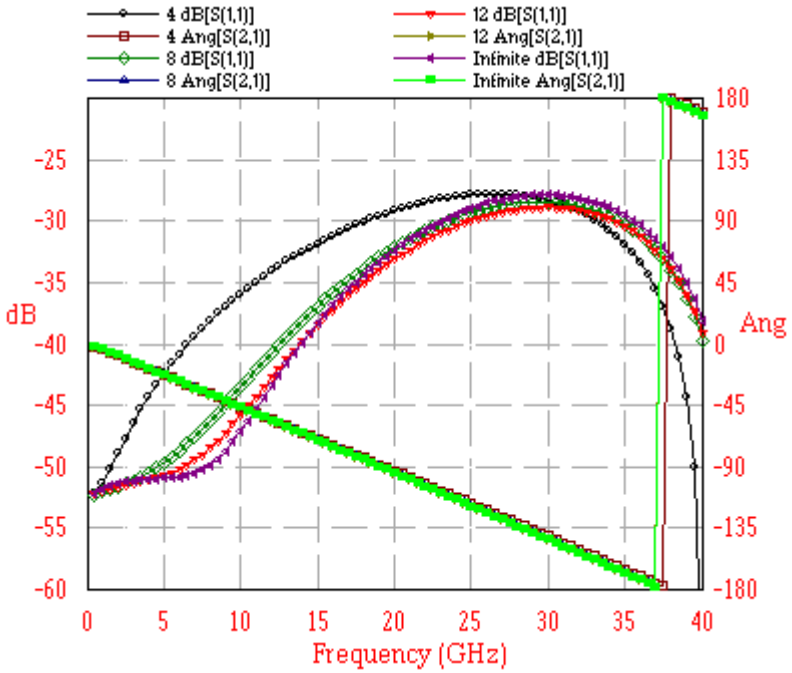


Figure 14.6 The comparison on the cases with different D/T ratio and S/T ratio.

Section 14.3 Horizontal Localized Port in Modeling of Enclosed Structures

For the above cases with enclosure side walls, we have been using the Extension for MMIC de-embedding scheme. We have to define a large box in order to have enough space for the extensions of the Extension for MMIC scheme. Certainly, we can use the Localized for MMIC scheme and the Vertical Localized scheme. However, we have to sacrificing loss of accuracy due to the higher order modes. In the reality of boxed circuits, the traces are normally fed by coaxial launch. Basically, the traces will extend to

the side walls and connect to the coaxial launch. The Extension for MMIC scheme cannot model such a situation accurately. The best scheme for modeling such a situation is using the Horizontal Localized Scheme.

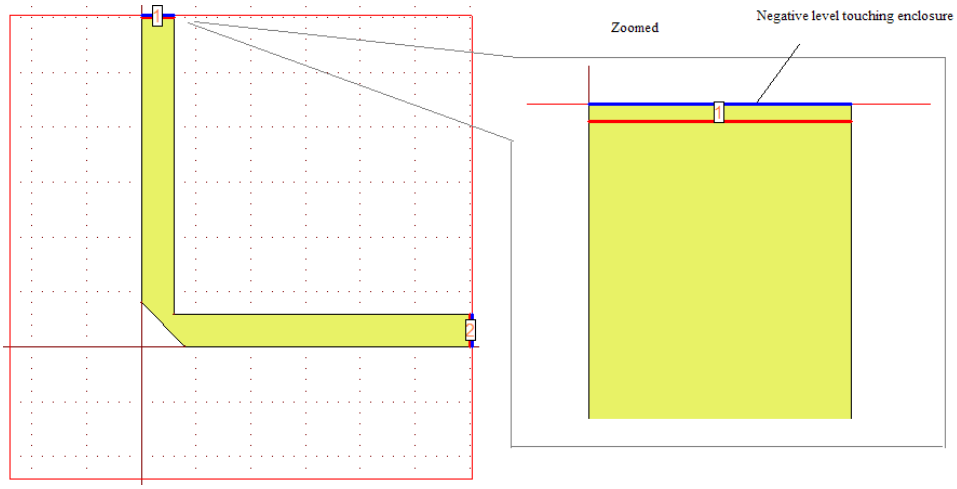


Figure 14.7 The Horizontal Localized Port in modeling enclosed circuits.

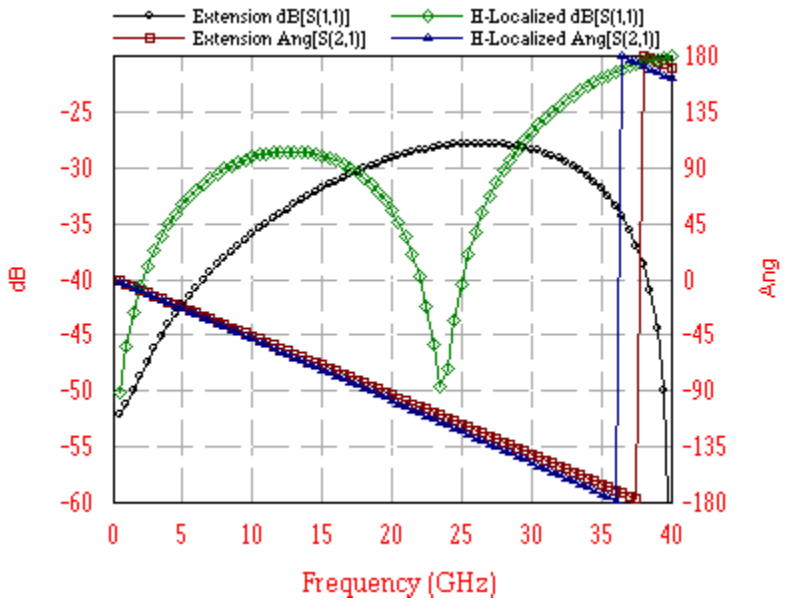


Figure 14.8 The comparison of the Extension for MMIC scheme (bigger box) and Horizontal Localized Scheme (smaller box).

The **.\ie3d\samples\c_bend3c4bh.geo** is the case implementing Horizontal Localized Scheme. Its minimum side wall distance and top cover distance are the same as the **.\ie3d\samples\c_bend3c4b.geo**. The top view of **c_bend3c4bh.geo** is shown in Figure 14.7. As you can see, the traces are touching the side walls. There are 2 small rectangles at the end of each trace. The 1st rectangle is attached to the side wall. The 2nd one is defined with a Horizontal Localized Port with the positive terminal connected to the trace and the negative terminal connected to the rectangle attaching the side wall. Ideally, we should be able to use one rectangle for such a modeling with the negative terminal of the port touching the wall. Due to the limitation on the implementation, we have to do it using 2 rectangles in the IE3D 8.0. It is expected that we will be able to eliminate the 1st rectangle in future release for better modeling. Such a model should be very

close to the coaxial launch case in reality. A comparison between the **c_bend3c4bh.geo** and **c_bend3c4b.geo** is shown in Figure 14.8. We can see some difference in the dB[S(1,1)]. The difference is more due to the difference in the enclosure sizes.

Section 14.4 The Evidence and Effect of Metallic Loss

This chapter is mainly discussing the structures with enclosure. We would like to discuss some side topic encountered in this chapter. If you pay attention to the low frequency end of the frequency responses in all the curves shown in this chapter, you will notice that the dB[S(1,1)] seems to be approaching a limit around –52.5 dB, no matter how big the enclosure is, how high the top cover is and how many terms we use in the simulation. Similar situations happen when the IE3D is used to model spiral inductor or long and thin traces. Many users may expect that the dB[S(1,1)] of a TLN to be approaching -∞ when the frequency is approaching DC.

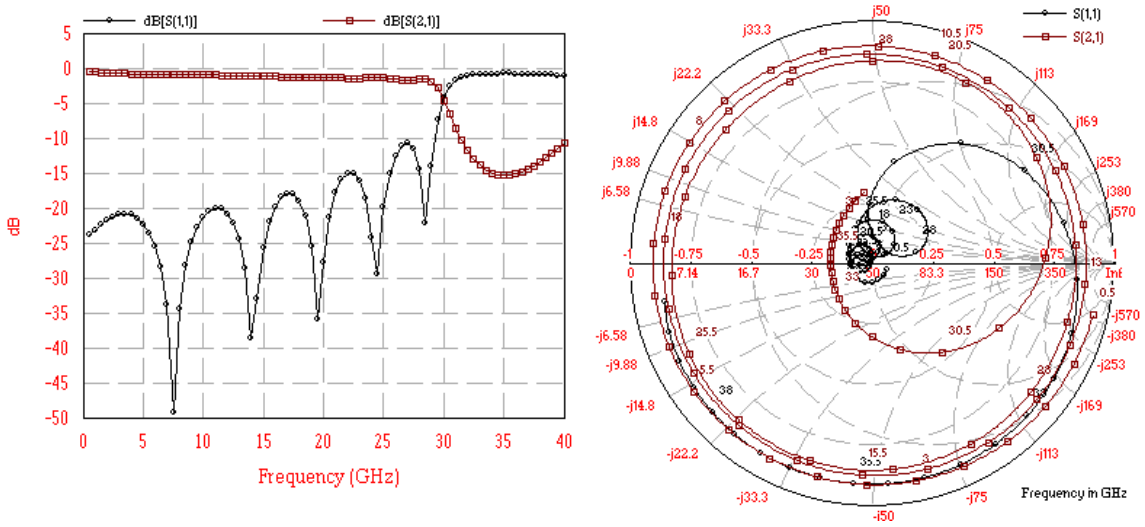


Figure 14.9 The frequency response of a long and thin trace in **.\ie3d\samples\long_trace.geo**.

Shown in Figure 14.9 is the frequency response of a long trace with low metallic conductivity. As you can see, the S(1,1) does not approach the center of the Smith Chart and the S(2,1) does not approach 1, as expected by many designers. Some users were concerned about the accuracy of the simulation. It turns out that the real situation is like what is shown in Figure 14.9. The reason for the above phenomenon is the metallic loss. In many idealized or low loss simulations, we do not consider the metallic loss or the metallic loss is very small. The S(1,1) is approaching the center of the Smith Chart (dB[S(1,1)] approaches the –∞). When the metallic loss is apparently, the S(1,1) will be approaching a positive real constant smaller than 1, or the dB[S(1,1)] will be approaching a finite value. Assuming the resistance of the strip at DC frequency is Zs. Then, the S(1,1)=Zs / (2 Zc + Zs), where Zc is the normalization impedance (50-Ω in the Extension for MMIC scheme). Re-arrange the formula and it will yield Zs = 2 Zc S(1,1) / [1 – S(1,1)]. When the dB[S(1,1)] = -52.5 dB at DC, it means that the S(1,1) = 2.37e-3. Therefore, we can get the Zs = 0.237Ω. In the other word, it is the Zs = 0.237 Ω causing the dB[S(1,1)] = –52.5 dB at DC frequency.

Interestingly, all our simulations, with or without the enclosure, indicate that the dB[S(1,1)] is approaching some similar constant when the shape and the parameters of the trace are given. As a matter of fact, the Green's function formulations are quite different for all the cases. The consistency in the dB[S(1,1)] limit at DC frequency is some indirect proof of the accuracy of the formulations. The shape and the parameters of the trace determine the resistance of the trace at DC frequency. The dimensions of the enclosure can only affect the coupling effects. The coupling effects are basically the inductance and

capacitance existing in the structure. At DC, the inductance and capacitance of the structure do not affect its performance and we will expect the dB[S(1,1)] to be the same for different enclosure parameters. The consistency in dB[S(1,1)] in the different IE3D modelings in the above discussion is a pronounced proof of the IE3D's accuracy for different cases.

Section 14.5 The Effect of Sidewalls and Cover on Resonant Structures

We come back to the topic of effect of sidewalls and cover. In Section 2, we have demonstrated that the effect of the enclosure (side walls and cover) is almost invisible when D/T ratio (the side wall to trace distance over the substrate thickness) and the S/T ratio (the cover to trace distance over substrate thickness) are about 8. The conclusion is good for many closed microstrip structures. However, there are situations that the enclosure may have significant effects to the properties of the structure even with much larger D/T and S/T ratios. A patch antenna will be a good example to demonstrate such an example.

We have 3 pre-built files: **notch1.geo**, **notch1b1.geo** and **notch1b2.geo** in **.\ie3d\samples**. They have the exact same shape in the polygons, however with different enclosure configuration. Figure 4.10 shows the top view of **notch1b2.geo**. The substrate thickness is 0.794 mm (about 31.26 mils). It is an open structure without enclosure for the **notch1.geo**. The **notch1b1.geo** has an enclosure with the side walls and the top cover about 6 mm away from the patch (D/T=S/T=7). The **notch1b2.geo** has an enclosure which is about 6-mm larger than the one in notch1b1.geo in each direction (D/T=S/T=14).

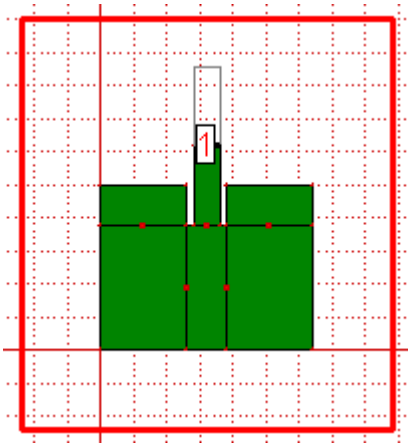


Figure 14.10 The notched patch with a sidewall.

Please simulate them. Again, you will notice the open formulation is much faster. The frequency responses of the 3 cases: open structure (notch1.geo), small box (notch1b1.geo) and large box (notch1b2.geo), are compared in Figure 14.11. Please note that the 2 pictures use different scales and they are quite different. The notch1.geo predicts a very good resonance with strong radiation at about 7.85 GHz. The notch1b1.geo predicts a weak resonance at about 8 GHz. The notch1b2.geo predicts 2 weak resonances at about 7.6 GHz and 8.1 GHz.

If we apply the rule we found in Section 2, the simulation results for the boxed cases do look suspicious. However, they are the real situations. The notch1.geo is an open structure. The open radiation boundary condition allows the patch to radiate efficiently at 8 GHz.

When the patch is put into a box, the situation is changed completely. From the view point of time domain, the power is delivered to the patch from the feed line. The patch radiates the power output when it is at resonance. However, the radiated power will hit the box and return back. After many radiating out and returning back process, the majority of the power is still returned back to the source along the feed line.

Only small percentage of the power is consumed due to the metallic and dielectric loss. That is the reason why we see the weak resonances for the boxed cases. Apparently, the box prevents the power to be radiated out and it may change the properties of the structure completely.

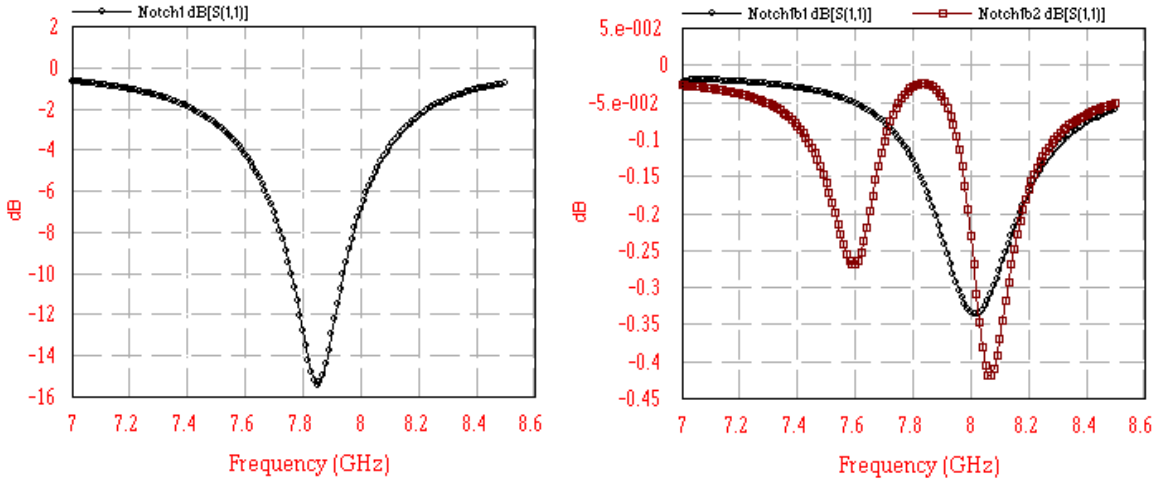


Figure 14.11 The comparison between the open and closed patches.

You may ask why there are 2 resonances for the larger box case (notch1b2.geo). It turns out that the other big effect the box may introduce is the box resonances. Basically, the box may introduce resonances at some frequency points. The properties of the structure are completely changed close to the resonant frequency points of the structure. For the notch1b2.geo, the 8.1-GHz resonance is caused by the patch, and the 7.6-GHz resonance is caused by the box resonance.

Some users may ask the following questions. When we keep increasing the size of the box, will the frequency response of the structure approaches some kind of limit? If we keep increasing the box size, will it approach the open structure case? In a lossless case, the frequency response should not approach some kind of limit. Think about it in the time domain. No matter how big the box is, the radiated power will come back completely. The time delay for the coming back will be different for different sized boxes. The frequency response is the steady state of the time signals. The steady state will be different for different sized boxes. Mathematically, one thing is for sure: the |S(1,1)| will be always 1 for the loss-less case no matter how big the box is as long as it is still finite. When the box is infinitely large, it becomes the open case and we will get strong radiation for the open case. In reality, the air has loss. When the box becomes larger and larger, the radiated power will be absorbed more and more by the air and the reflected back power will become weaker and weaker. Eventually, the frequency response of the structure will approach the open case. In antenna measurement range, people built microwave chambers to absorb the radiated power to reduce the reflected back power so that the measured input impedance of the antenna is as close as the reality.

The above example demonstrated that the enclosure might have significant impact to the properties of the circuits when the circuits are at resonances. However, the above example is an extreme example. In practical circuits design, we may see the effects of enclosure when the circuit is at resonance. However, they may not be able to change the circuit performance completely.

Let's use the hairpin filter discussed in Chapter 6 for more discussion on the enclosure effect to resonant structures. Antennas are making use of resonances to radiate out the power. Filters are making use of resonances to transmit power in a transmission line system at selected frequency ranges. Both of them are making use of resonances.

All the files will be discussed are in the .\ie3d\samples directory. Files for different cases are documented in Table 14.5. For the boxed structures, we use Nx = Ny = 20. It seems to us the Nx = Ny = 10

is not accurate enough. Interested users can compare the difference between the Nx = Ny = 20 and Nx = Ny = 10 cases. The Nx = Ny = 20 case takes about 27 seconds per frequency point and the Nx = Ny = 10 takes about 17 seconds per frequency point. As the 1st release on the boxed case, we have not implemented an automatic way to find the suitable Nx and Ny values for the users automatically. We will implement such a feature in the next releases. The frequency responses of the different cases are shown in Figure 14.12. They actually compare very well in the passband. The notch in the lower stopband is shifted with covered and boxed cases. The notch is caused by far coupling between the poles. The presence of the cover and the sidewalls change the far coupling more.

Table 14.5 The comparison of the hairpin filter with or without the enclosure.

File	Symbol	Description
Hairpin.geo	Open	The filter is in an open environment with the upper space filled with dielectrics of εr=1.4. It takes about 2 seconds per frequency point.
Hairpin_cover.geo	Cover	The filter has a cover at z = 200 mils and has not side walls. It takes about 2 seconds per frequency point.
Hairpin_short_wall.geo	Small Box	The filter has a cover at z = 200 mils and side walls about 200 mils away from the traces. It takes about 27 seconds per frequency point.
Hairpin_short_wall12.geo	Large Box	The filter has a cover at z = 200 mils and side walls about 400 mils away from the traces. It takes about 27 seconds per frequency point.

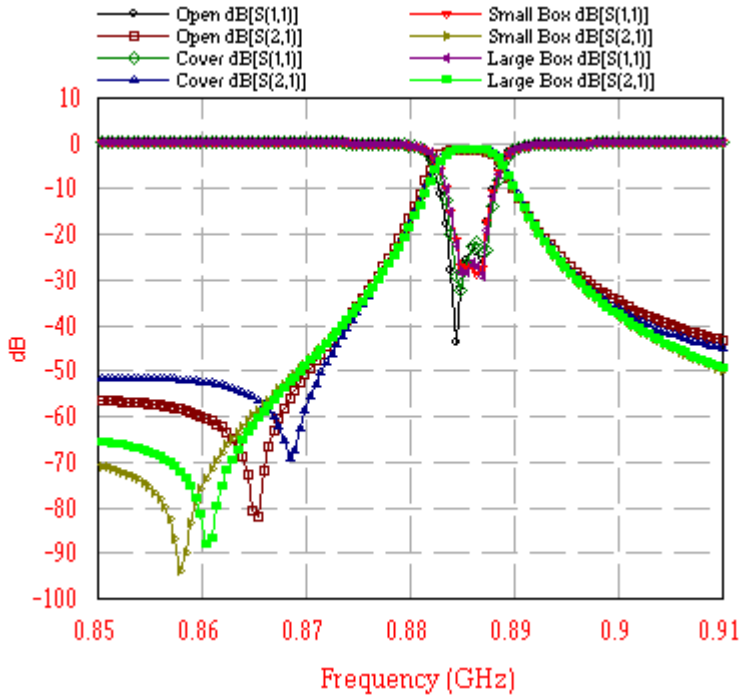


Figure 14.12 The frequency responses of the hairpin filter for different enclosures.

This filter is well designed to prevent radiation even at the resonant frequency. That is the reason why the enclosure does not change the properties of the filter at the passband. The key to use prevent radiation is to keep the electrical dimension of the substrate thickness small. Another important point is to avoid vias. In practical microwave filter design, it is possible that the designers may choose very thick substrate and vias. Thick substrate millimeter wave circuits are easier to manufacture. Vias are used to reduce the lengths of the poles to half. Thick substrate will allow the power to be easier to radiate out. Vias

have strong radiation properties. For a filter with vias and thick substrate, much power may be radiated out at the resonant frequency if no enclosure is used. For such a case, it is very important to include the size of the box accurately. If you do not include the enclosure in an IE3D simulation for thick substrate filters with vias, it will predict much radiation loss which may not be there in practical use. We will not give an example here.

Section 14.6 Resonances Resulting from an Enclosure.

We have mentioned that there exist enclosure resonances in a boxed structure in Section 5. The notch1b2.geo is the case. The resonance at 7.6 GHz in Figure 14.11 is basically a resonance from the enclosure. We will use the box of the same dimensions as our example in the discussion of this section. What we want to investigate is gap at the middle of a microstrip line. The strip width is 2 mm (same width as the feed line in notch1b2.geo). The structure is saved in **.\ie3d\samples\boxed_gap.geo** and shown in Figure 14.13. Two horizontal localized ports are defined between the ends of the strip and the enclosure walls. For high accuracy results, we will use edge cells to model the strip. However, the users should be informed on the fact that the Automatic Edge Cells is not implemented into the Horizontal Localized port. If you use Automatic Edge Cells for a simulation involving the Horizontal Localized port, there will not be edge cells created in the region of the Horizontal Localized ports. Feature improvement on implementation will remove such a small limitation. In order to make sure there will be edge cells in the Horizontal Localized ports, we purposely create the Horizontal Localized ports with edge cells.

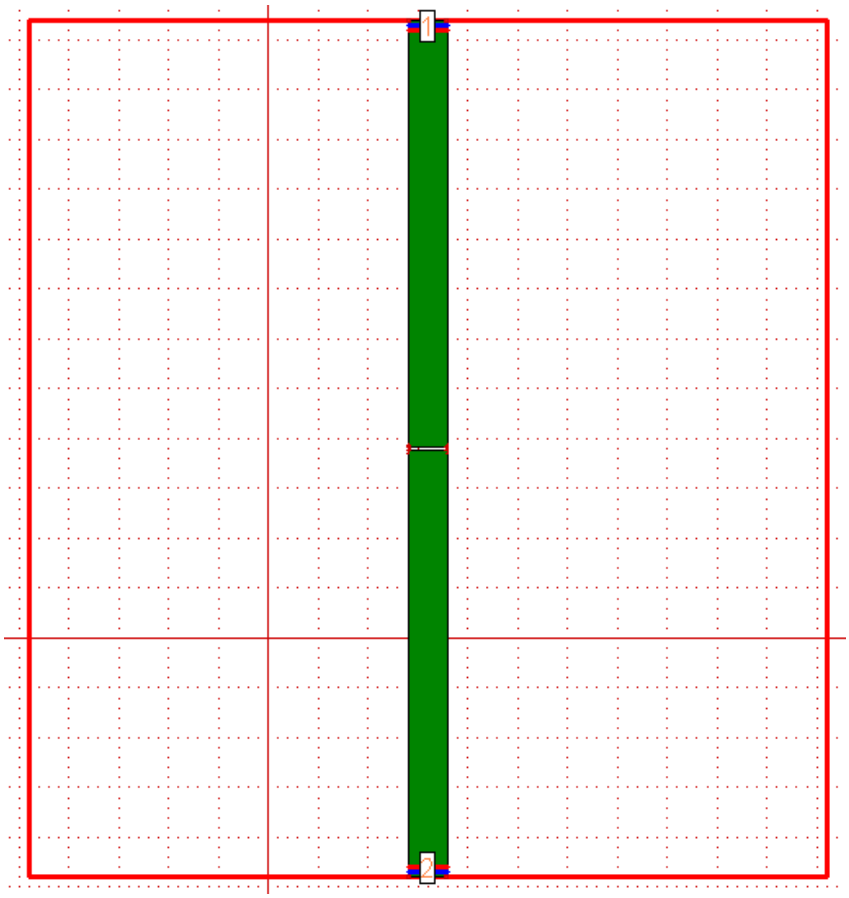


Figure 14.13 The microstrip gap structure inside an enclosure.

The same microstrip in open environment is saved in .\ie3d\samples\open_gap.geo. Figure 14.14 shows the comparison between the two situations. Perfect agreement is observed below 4 GHz. The

agreement is still very good except close to the resonances at 5 GHz and 7.6 GHz. Apparently, the 7.75-GHz resonance is causing the notch at 7.6 GHz in the frequency response of **notch1b2.geo**. At the box resonances, we can see the dB[S(2,1)] is quite large and the gap circuit becomes a bandpass filter at the resonant frequencies. Both the open and close Green's function formulations yield accurate results for you under the conditions of their formulations. In the design of microwave circuits, the designers normally should try to re-size the box size to avoid the resonances to be located at the range of interest.

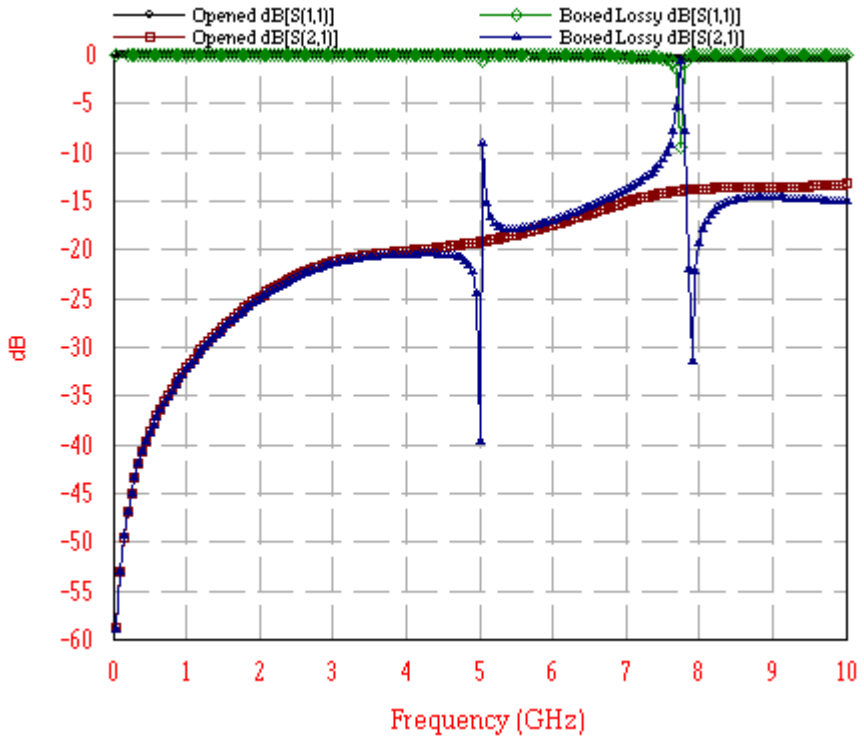


Figure 14.14 The comparison on the gap structure for the open and closed cases.

It is good to have a tool like the IE3D to be able to compare the different situations. The combined open and closed Green's function formulation on the IE3D makes the IE3D equally powerful for microwave circuits and antenna applications. We can make our conclusions for this chapter here: (1) IE3D is equally powerful in modeling opened and closed structures. (2) The open Green's function formulation is normally significantly faster than the close Green's function form formulation due to its nature. (3) Circuit enclosure walls normally do not have significant impact to the circuits except strong resonances or the circuits are too close to the walls. (4) Enclosures will affect the performance of the circuits at strong resonances. (5) The enclosure resonances may change the properties of the circuits completely.

Chapter 15 Modeling of Differential Structures

Differential structures have become more and more important due to the recent development in the wireless communications and high-speed digital circuits. In electromagnetic simulation, we also call differential structures as finite ground plane structures. For wireless applications such as a handset, we may not have a large ground plane to be approximated as an infinite ground plane. We have to use the differential feed to model such a structure. Differential lines are used in high speed networking extensively in the semiconductor industry. Compared to the traditional structures with large (or infinite) ground planes, differential structures have their unique features. In this chapter, we will discuss some topics involved in the differential feed.

Section 15.1 The Odd Mode and Even Mode

When differential structure is discussed, odd mode and even mode are normally discussed. For symmetrically structures, a differential structure means that we are exciting the odd mode of the structure. In fact, differential structure means more than that. On the IE3D, we can apply differential feed to symmetrical or non-symmetrical structures.

A typical differential structure is a pair of coupled metallic strips as shown in Figure 15.1. Every person knows that we can feed it differentially to excite the odd mode on it. We apply a voltage between the two traces and the power of the odd mode will be delivered along the transmission line efficiently. Some people may ask the following questions: What is its even mode? What is the coupling between the 2 traces?

Figure 15.1 A pair of coupled strips for differential excitation.

Naturally, people may consider the even mode as the one with the two traces of the same potential. In fact, this is misleading. Before we can answer what is its even mode, we need to know whether there is the 3rd conductor on this transmission line. In the discussion of de-embedding schemes in Section 4 of Chapter 12, we have discussed about modes. In the microwave theory, a mode is basically some kind of field configuration on the cross-section of a transmission line structure. Ideally, it should allow the power to be delivered with little loss in the longitudinal direction and no power radiated out in the cross-sectional direction.

For a 2-conductor system, there exists only one fundamental mode. The fundamental mode is the differential mode of the coupled strip. There exists no even mode for the coupled strip. If you apply the same potential to the 2 traces, most of the input power will be radiated out by the coupled trace if the traces are long. If the traces are short, you just cannot deliver any power into the traces. Also, we cannot apply the microwave network theory to the coupled traces with co-phase excitation because microwave network theory is based upon the mode theory. In other words, if we have two sections of the evenly driven coupled traces, we cannot use the s-parameters of the two sections cascaded together to get the s-parameters of the overall structure. Precisely, s-parameters are also defined based upon mode. However, Z-parameters and Y-parameters are not. Even though there is no mode in the structure, you can still get its Z-parameters or Y-parameters. Then, you can convert the Z-parameters or Y-parameters to the 50-ohms normalized s-parameters.

For the coupled strip structure, we also cannot define the coupling between the two traces. Basically, the two traces become a system. They are inseparable. To define the coupling between the traces, we have to use the so-called partial inductance concept. The partial inductance concept is pure mathematically

defined. It is an elegant way to define the mutual magnetic coupling at DC frequency. However, it does not have the physical meaning in the microwave theory. For full wave simulators, it is impossible to find it. Using full wave simulators, we can only find the loop inductance. The two traces form a loop and we can find the inductance for the two traces (when they are driven differentially). However, we cannot find the coupling between the two traces because each of the traces is only part of the system.

Section 15.2 Modeling of Slotline Structures

Quite some users try to use the IE3D to model slotline structures (see Figure 15.2). However, they could not find the right way. In fact, for a single slot on an infinite ground plane, we just cannot define the s-parameters because there is no fundamental mode on a single slot line on an infinite ground plane. As it is mentioned in multiple locations, there exists one fundamental mode for a two-conductor system. For a single slot on an infinitely large ground plane, we can consider it as two conductors separated by the slot. Why can we not find the fundamental mode on it? If we drive the two semi-infinite grounds, should we build up a differential mode on it? The answer is No. The problem is that the two conductors are infinitely large. When it is infinitely large, the situation is completely different.

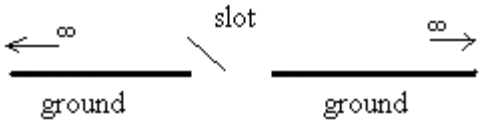


Figure 15.2 A typical slot on an infinite ground plane.

On the IE3D, we can model the slot as a polygon on a ground plane (see Figure 15.3). It just looks like a printed strip structure. In fact, mathematically, a slot on an infinite ground is the dual problem of a strip in the air without a ground plane. Can we feed a single strip without a ground plane? The answer is obviously no, because there is no negative terminal for the excitation. With only one terminal, you cannot deliver the power to it.

Figure 15.3 A polygon on an infinite ground plane to model a slot on the IE3D's magnetic current modeling.

Antenna designers know they can differentially feed two strips to get a dipole antenna. However, if we have only one strip, it will not become an antenna.

From the viewpoint of low frequency circuit theory, we try to apply some voltage to the single strip. However, we will not be able to get any current on it because there is no return path. For the single slot on an infinite ground, we try to inject some current into it. However, we will not be able to build up the voltage because it is basically short-circuited.

From the dual theory, we can conclude that we cannot support a fundamental mode on the strip line structure. If we look at the problem in the microwave transmission line theory, we can also have the same conclusion. If we consider the slot has two finite ground planes, it is basically the coupled strip structure (see Figure 15.4). Certainly, we will have the differential mode on the coupled strip structure. However, if we try to increase the width of the finite ground plane, at some point, some higher order modes will become propagating. In the other word, if we keep increasing the ground width, the cut-off frequency of the higher order modes will keep decreasing. When the ground width approaches infinity, the cut-off frequency of many higher order modes will approach 0. Eventually, it becomes a spectrum. It means that there is no fundamental mode in it. In fact, for the single strip in the air, we can get the same conclusion. We can consider a single strip in the air as a special case of coupled strips. When we move the two coupled strips

apart, we start reducing the cut-off frequencies of the higher order modes. When the distance between the two strips approaches infinity, the coupled strips becomes a single strip and the cut-off frequencies for many higher order modes becomes 0. The above analysis will explain why we cannot define a port on a slot line with infinite ground plane on the IE3D. Certainly, if we have finite ground planes for the slot, the slot line becomes coupled strip and we can still define a port on it, either in electric current or magnetic current modeling. If we have coupled slots, the structure is basically a CPW structure and we have demonstrated how to simulate it in Chapter 6.

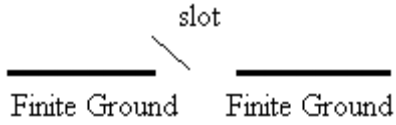


Figure 15.4 A slot with finite ground planes.

Section 15.3 Measurement of Differentially Fed Structures

Some antenna designers find difficulty to measure wireless antennas. A typical example is shown in Figure 15.5. A coaxial is feeding an antenna through a finite ground plane. They may found that the bending of the far-end of the coaxial will affect the input impedance and the radiation pattern of the antenna. They wonder whether an electromagnetic simulator like the IE3D or the FIDELITY can predict why.

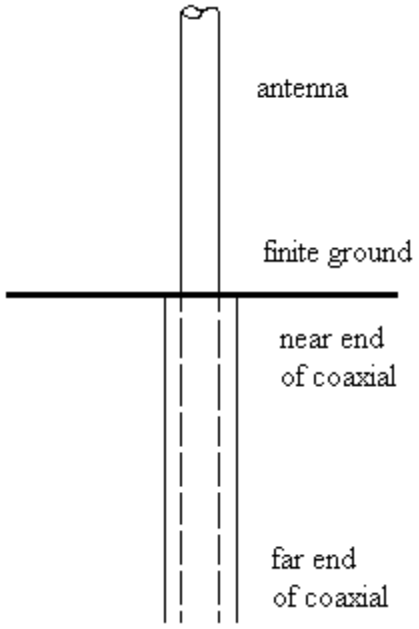


Figure 15.5 A measurement setup for a wireless antenna.

In fact, the answer is very simple and it is related to the differential feed. Basically, the coaxial is feeding the antenna. The voltage between the inner conductor and the outer conductor is applied to the antenna between the wire and the finite ground plane. Unfortunately, this antenna is not well balanced. Some current will go onto the outer surface of the outer conductor of the coaxial. The power carried by the current on the outer surface of the outer conductor will radiate out eventually. When you are bending the coaxial, you are changing the current on the outer surface of the outer conductor and it is changing the input impedance and the radiation pattern. If we want to get consistent results, we should design a good balun at the feed point or we should be able to enforce balance of current at the feed point internally in the amplifier. Otherwise, we have to include the long coaxial lines and consider it as part of the system in the measurement or simulation.

Section 15.4 Strategy in Modeling Differential Ports

When we are modeling symmetrical differential structures, everything is quite straightforward. If we are using the Extension for MMIC port scheme, we just need to define the + terminals and the – terminals as shown in Figure 15.6. To define a negative terminal is quite simple. We just use the right mouse button to select the edges immediately after we define the positive terminal. We can define multiple negative terminals consecutively for 1-positive terminals by using the right mouse button to select the edges.

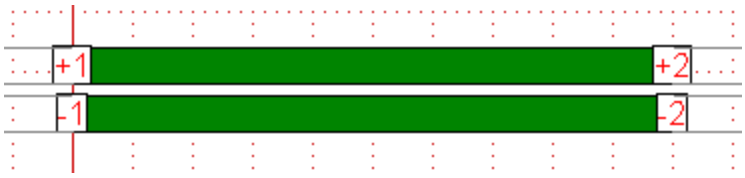


Figure 15.6 The positive and negative terminals for a symmetrical differential structure.

We can cascade different symmetrical differential structures together to find the s-parameter of a larger structure (see Figure 15.7). It is suggested that you should match the port configuration and keep the same strip widths for the + and – terminals. It is also suggested that you should have extra uniform sections before the ports (see Figure 15.7).

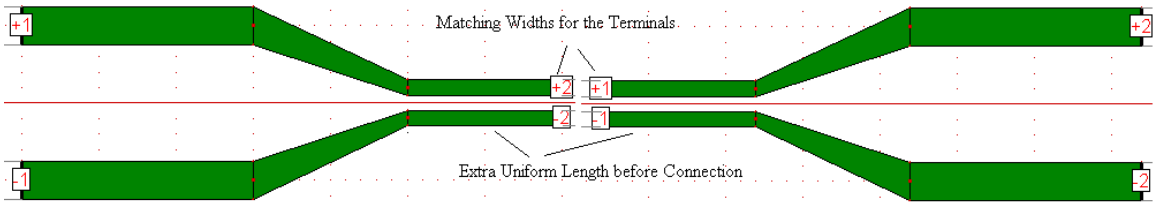


Figure 15.7 Requirements for high accuracy network connection on MODUA.

For non-symmetrical structures, we need to be careful in defining the ports. For the coupled strip structure in Figure 15.8, we need to be careful how we can use the results. We cannot use it anywhere. If we are able to enforce the differential condition at the port 1 and port 2, the modeling in Figure 15.8 is absolutely good. When we mean enforce the differential condition, we mean that the amplifier (or driver) at port 1 will eliminate the even mode or even mode like current distribution, and the load at the port 2 will also eliminate the even mode like current distribution. For example, a resistor, a capacitor or an inductor connected across the 2 terminals of port 2 will enforce the differential condition (see Figure 15.8).

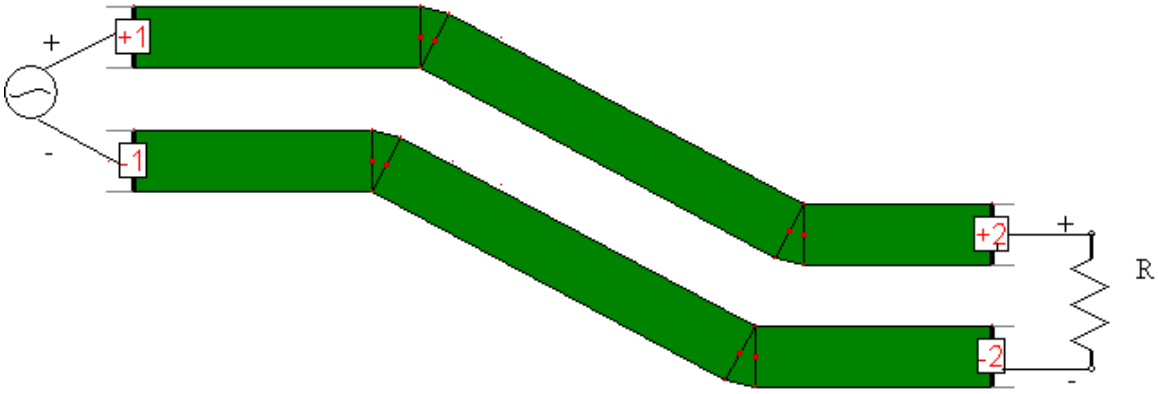


Figure 15.8 Differential ports on non-symmetrical structures.

If we want to get the final s-parameters of the coupled strip structure C consisting of two non-symmetrical structures A and B, or at least one of the two structures is non-symmetrical (see Figure 15.9), we need to be very careful in order not to lose accuracy.

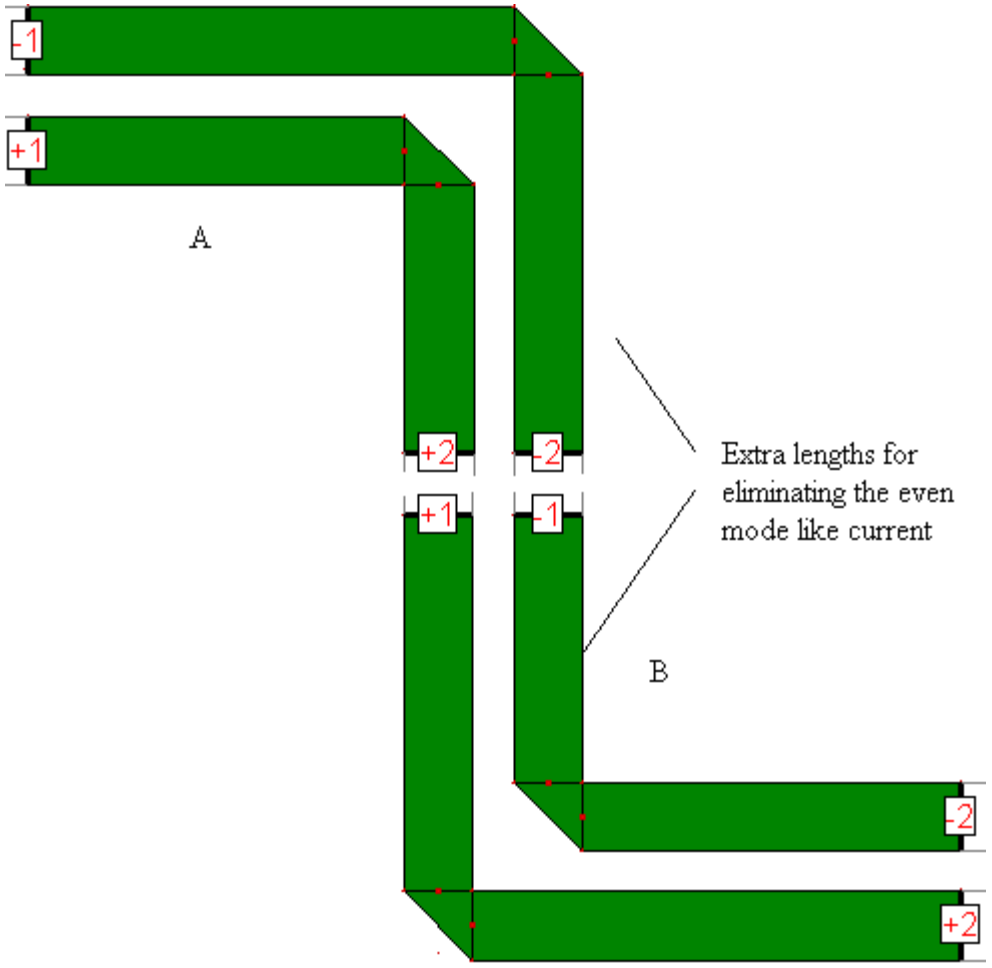


Figure 15.9 The port configuration for network connection of non-symmetrical structure without infinite ground plane.

If the coupled strips do not have the infinite ground plane, we have to simulate the structure as 2-port differential structure. Certainly, the junction in the non-uniform structure of Figure 15.9 will excite the even mode like current distribution. We have to make sure there is a long enough feed line before the ports. The even mode like current distribution eventually will die out due to the fact it cannot propagate along the transmission line. Its power may be converted back to the differential mode or radiated out. Then, cascading the s-parameters of the A and B in Figure 15.9 will still be valid. However, if the feed lines in the port 2 of A or the port 1 of B are not long enough, the cascading of the differential s-parameters of A and B may not yield accurate results because the even mode like current distribution created at the junction A may reach and interact with the junction B. The interaction may create new phenomenon that cannot be captured by the differential s-parameters of A and B. For such a situation, we would rather simulate the complete structure C (see Figure 15.9) as one single structure.

If the coupled strips have an infinite ground plane, non-symmetrical discontinuities such as bends will certainly create the even mode. The even mode will not disappear automatically because the even mode is a propagating mode. We have to enforce the differential condition. Unfortunately, enforcing differential condition is not simple on a coupled strip structure with the ground plane. For such a case, we would rather

simulate structures A and B as 4-port structures (see Figure 15.10). We can connect the s-parameters of A and B to get the 4-port s-parameters of C. We can cascade the s-parameters of C with more such differential s-parameters without much limitation. Eventually, we will get the final 4-port s-parameters and can convert it into 2-port differential port s-parameters using the Differential Conversion feature in the Process menu of MODUA. When we convert the 4-port s-parameters into the 2-port differential s-parameters, we assume that the external amplifier, driver or load will enforce the differential conditions. If we cannot guarantee it, we cannot convert the 4-port s-parameters into 2-port s-parameters because it may lose information about the transmission line or interconnect.

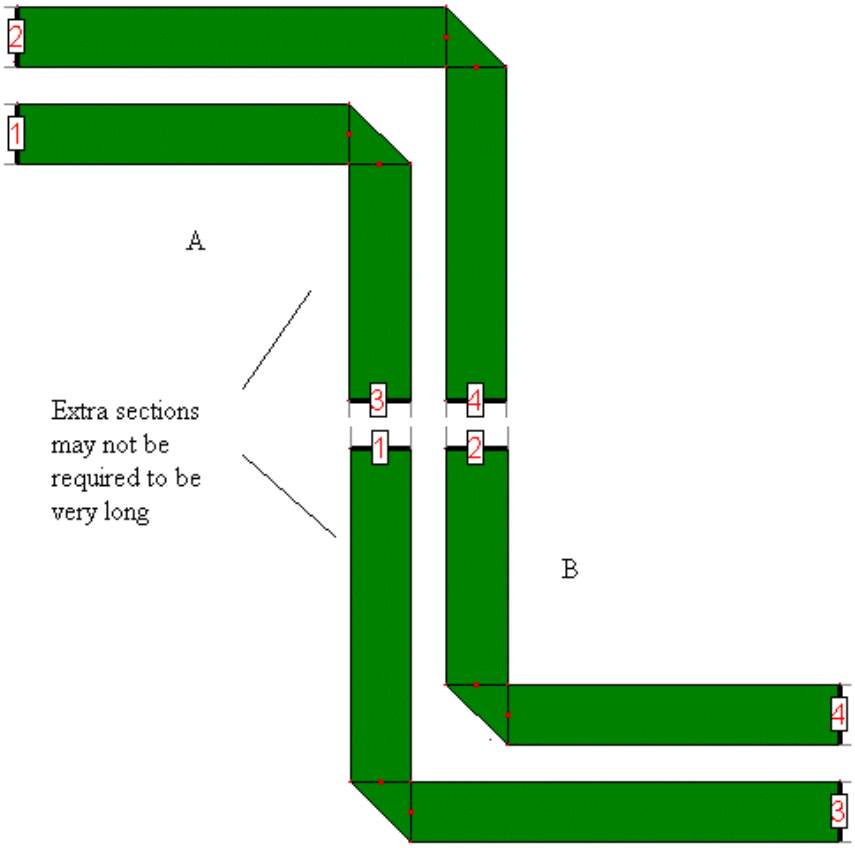


Figure 15.10 The best port configuration for modeling differential structures.

In fact, the port configuration in Figure 15.10 should be the best no matter whether the structure has a finite or infinite ground plane. However, if you use the port configuration in Figure 15.10 for structure without a ground plane, you need to know that the 4-port s-parameters for A, B and C may not have physical meaning. We need to use the Differential Conversion feature on MODUA to convert the 4-port s-parameters into the 2-port differential s-parameters. Interestingly, when we use the port configuration in Figure 15.10, we do not need to suppress the even mode like current distribution, and we even do not need the extra lengths for the ports to be very long.

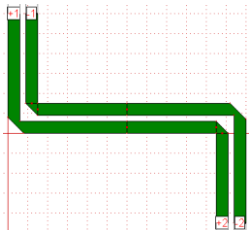
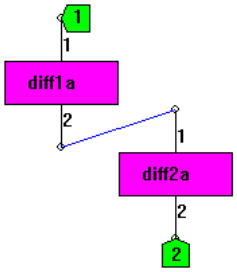
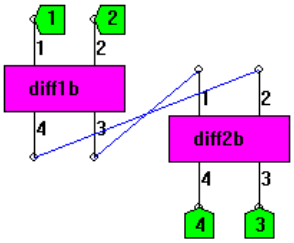
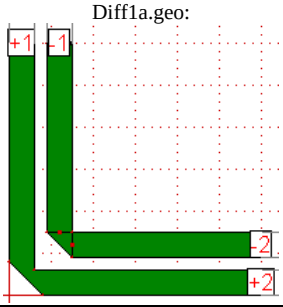
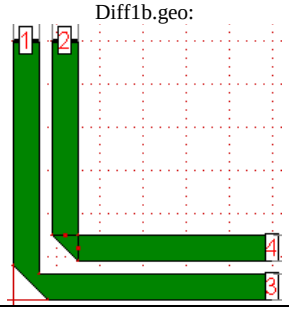
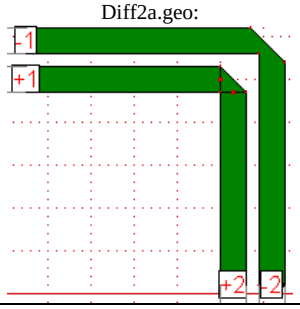
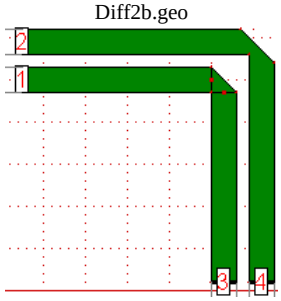
Section 15.5 Differential Conversion

Differential conversion can be achieved on both MGRID and MODUA in IE3D 14. The feature is transferred from MODUA into the s-parameter processing on MGRID. We will still discuss how we can do it on MODUA. MODUA is a powerful utility program for the IE3D. On the IE3D, we can simulate a differential structure as non-differential structure first. We can use the MODUA to connect the non-differential s-parameters without much limitation. After we get the final s-parameters, we can always

convert it into the differential s-parameters. Modeling structure in this way, we can always preserve the accuracy. In this section, we will use an example to demonstrate the process:

Assuming we want to simulate a differential structure c:\ie3d\samples\diff1c.geo in Table 15.1, we can simulate it as a whole structure. Assume we want to simulate it as 2 sections and we use the s-parameters of the 2-sections to get the final s-parameters of the total structure. We have two approaches, and the 2 approaches may yield results of different accuracy.

Table 15.1 The approaches in simulating differential structures using sub-network.

Approach	Complete Structure	Differential All the Way	Differential at Last
Final Results	S-parameters for Diff1c.geo	S-parameters for Diff1c.geo	S-parameters for Diff1c.geo
Scheme	Modeling Diff1c.geo as a whole structure on MGRID (Diff1c.geo). The simulation result in Diff1c.sp is the final result: 	Modeling Diff1a.geo & Diff2a.geo. Connect their s-parameters on MODUA (Diff3a.dsg). The simulation result in Diff3a.spm is the final result: 	Modeling Diff1b.geo & Diff2b.geo as non-differential structures. Connect their s-parameters on MODUA (Diff3b.dsg). The simulation result in Diff3b.spm is the intermediate result: 
1st sub-geometry	N/A	Diff1a.geo: 	Diff1b.geo: 
2nd sub-geometry	N/A	Diff2a.geo: 	Diff2b.geo 
Differential Conversion	N/A	N/A	The 4-port intermediate result Diff3b.sp from the Diff3b.dsg needs to be converted into 2-port result in Diff3b1.sp.
Final Result	Diff1c.sp	Diff3a.sp	Diff3b1.sp

After we simulate the Diff3b.dsg file on MODUA, we should select **File->Save S-Parameters** command to save the 4-port intermediate s-parameters. We still need to convert the 4-port s-parameters into 2-port differential s-parameters. We first select **File->Display Parameter Module** command of MODUA.

Select the Diff3b.spm and it will be displayed by MODUA. Then, we select **Control->Display Toggle** command of MODUA. We will see the black box representation of Diff3b.sp. Select **Process->Differential Conversion** command. Select Diff. Port button. MODUA will prompt you for the Positive Terminal Mapped from Port 1, and Negative Terminal Mapped from Port 2. Select OK to continue. The No.1 Differential Port will be listed and its positive and negative terminals are mapped from the ports 1 and 2 of the Diff3b.spm, respectively. Select Diff. Port button again. MODUA will prompt you for the Positive Terminal Mapped from Port 3, and Negative Terminal Mapped from Port 4. Select OK to continue. The No.2 Differential Port will be listed and its positive and negative terminals are mapped from the ports 3 and 4 of the Diff3b.spm, respectively. Select OK to continue. MODUA will convert the s-parameters of the 4-port non-differential s-parameters into the 2-port differential s-parameters. The results will be displayed on MODUA. Select **File->Save S-Parameters** command on MODUA and save it as .\ie3d\samples\output\Diff3b1.sp.

Diff1c.sp, Diff3a.spm and Diff3b1.spm are the final results for each approach. The "Complete Structure" result in Diff1c.sp is certainly the accurate one. The question is which one of Diff3a.spm and Diff3b1.spm is more accurate. The results are compared in Figure 15.11. As you can see, the "Differential at Last" result in Diff1b1.spm agrees perfectly with the "Complete Structure" result in Diff1c.sp. The result in "Differential All the Way" in Diff3a.spm agrees very well with the "Complete Structure" result in Diff1c.sp below 20 GHz. However, we can see they start depart at 20 GHz. The difference is basically from the fact that the "Differential All the Way" approach neglects the possible even mode created in the intermediate results.

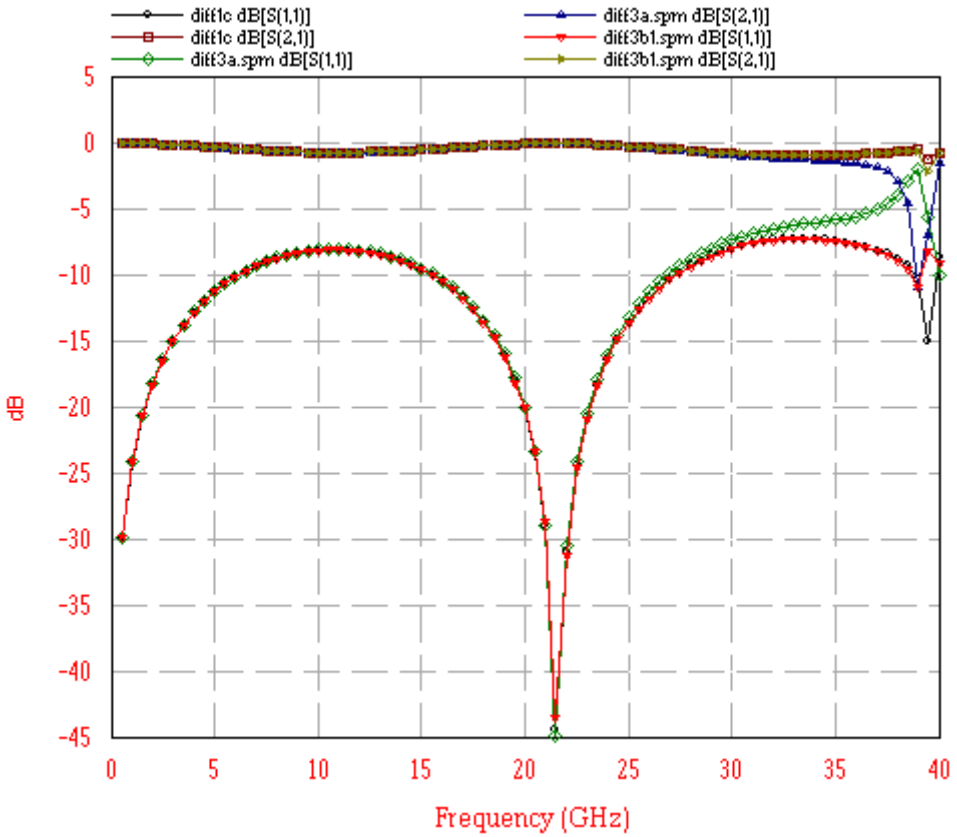


Figure 15.11 The comparison of the 3 approaches in modeling differential structures.

What we have shown above is a differential port structure with infinite ground plane. However, the same procedure can be applied to differential structures with finite ground planes. The users should realize

that, when no infinite ground plane is defined, the 4-port s-parameters in Diff1b.sp, Diff2b.sp and Diff3b.sp might not be reasonable results. However, the differential converted 2-port s-parameters in Diff3b1.sp will be reasonable results.

Section 15.6 Modeling Odd- and Even-Mode Structures on IE3D

We have used IE3D to model differential structures in the previous chapters. For a symmetrical structure, differential feed means odd mode. In this chapter, we will give a summary on how to model the odd or even mode structures in the following. Saved in **.\ie3d\samples\coupled3.geo** is a differential structure consisting of two coupled microstrip lines with an infinite ground plane. The question is how we can re-define the ports to get the even and odd mode characteristic impedance (Zeven and Zodd). The odd mode configuration is in fact the differential feed configuration. We will brief the steps in the following. We will do the Zodd (or differential feed) configuration first.

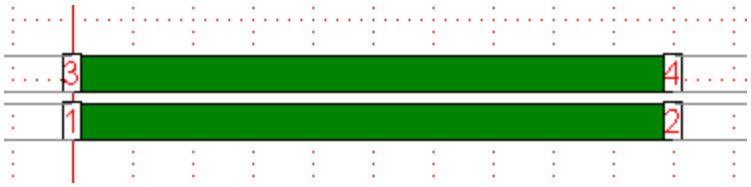


Figure 15.12 The coupled microstrip line: coupled3.geo

Step 1 Run MGRID. Open **coupled3.geo** file. Select **Port->Delete All Ports** to delete all the ports.

Step 2 Re-define the ports as shown in Figure 15.13. Remember to use right mouse button to select the edges when you define the negative port. You can also use Port->Change Port Order for it. Save the structure as **coupled3_diff.geo**. Simulate it.

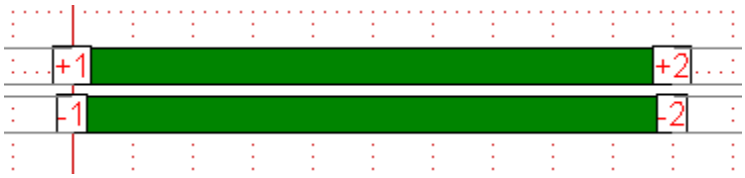


Figure 15.13 The Zodd configuration for the coupled lines: **coupled3_diff.geo**.

Step 3 Use MODUA to display the 2-port S-parameters of the differential feed structure. Then, use the **Process->Find Transmission Line Parameters** command on MODUA to find the odd mode characteristic impedance of the coupled transmission line (Zodd). If you do not know how to find the characteristic impedance of a transmission line using the 2-port s-parameters of a uniform transmission line, please read the Section 5 of Chapter 6. For the **coupled3_diff.geo**, Zc = 2Zodd = 141.656 – j 0.4127 Ω at 10 GHz. So, Zodd = Zc/2 = 70.828-j0.206 Ω. The reason is that IE3D gives in a general way the characteristic impedance of the transmission line formed by one strip at +1V and the other at –1V with the ground present while the odd-mode impedance is the one due to one of the strips and the ground under odd-mode excitation.

Step 4 Re-define the ports as shown in Figure 15.14. You may notice that there are two port 1's. This is so-called Multiple Positive Port on MGRID. Immediately after you define a port, if you want to make it multiple positive, you need to select the **Port->Multiple Positive Port** command on MGRID. Then, the next port you define will have the same index as the previous port.

Step 5 Simulate the structure. Use MODUA to find the characteristic impedance of the even mode configuration. We will get the characteristic impedance of the whole transmission line, **coupled3_even.geo**, as Zc = Zeven/2 = 82.6864 – j 0.105226 Ω. Therefore, Zeven = 2Zc = 185.5728 – j 0.210452 Ω. The reason is that IE3D gives in a general way the characteristic

impedance of the transmission line formed by the two equipotential strips and the ground while the even-mode impedance is the one due to one of the strips and the ground under even-mode excitation.

Figure 15.14 The even mode configuration of the coupled strips: **coupled3_even.geo**.

As we have demonstrated above, in order to get the Zodd and Zeven of a coupled transmission line, we have to do two simulations, one for the odd mode and one for the even mode. In fact, we can use the differential conversion on MODUA for it.

Step 1 Simulate the **coupled3.geo** file as a 4-port structure.

Step 2 Build the design shown in Figure 15.15 on MODUA. Basically, we define the terminals 1 and 2 as ports1 and 2. We define a connection to connect terminals 1 and 3, another connection to connect terminals 2 and 4.

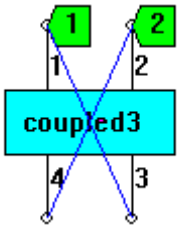


Figure 15.15 The MODUA setup for converting 4-port s-parameters into even-mode 2-port s-parameters.

Step 3 Simulate the design shown in Figure 15.15 on MODUA. The resulting s-parameters will be the same as the result from **coupled3_even.geo**. Select **File->Save S-Parameters** command on MODUA to save the result as **coupled3_cvt_even.sp**. Display the s-parameters in **coupled3_cvt_even.sp** on MODUA and we will be able to find the **Zeven_d2** as we did on **coupled3_even.sp**.

Step 4 Build the design shown in Figure 15.16 on MODUA. In fact, you can use **file->Display Parameter Module** command to create the design automatically.

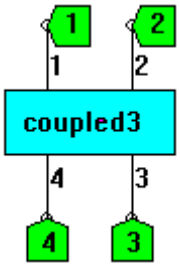


Figure 15.16 The MODUA setup for converting the 4-port s-parameters into odd-mode 2-port s-parameters.

Step 5 Select **Differential Conversion** in **Process** menu. Click at **Diff. Port** button. Choose the Port 1 and Port 3 as a pair of differential ports. Select OK. Click at **Diff. Port** button again. Choose the Port 2 and Port 4 as another pair. We will get what is shown in Figure 15.17. Select OK. The 4-

port s-parameters will be converted into 2-port s-parameters. MODUA is ready for you to do the display. Please select **File->Save S-Parameters** command to save it as **coupled3_cvt_odd.sp**. The **coupled3_cvt_odd.sp** file should contain the s-parameters for the odd mode 2-port s-parameters. You can use MODUA to get the Zodd using the method described above.

In fact, we can also use the Odd and Even Mode Conversion command in Process menu on MODUA to solve the s-parameters based upon the odd and even modes. If we apply such a command to the 4-port s-parameters, we will also get the 4-port s-parameters. The ports 1 and 2 after conversion are the odd and even modes of the original ports 1 and 2, respectively. The ports 3 and 4 after conversion are the odd and even modes of the original ports 3 and 4. The Odd and Even Mode Conversion command is also implemented into MGRID in s-parameter processing.

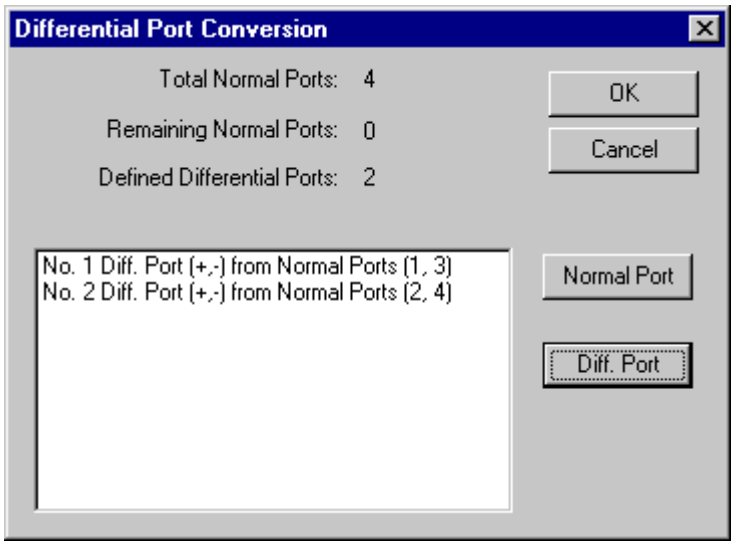


Figure 15.17 The dialog as we define the 4-ports as 2-differential ports.

Chapter 16 Modeling of Finite Dielectric Structures

IE3D has been a full-wave EM simulator capable of modeling planar and 3D metallic structures in layered dielectrics. Most modern planar circuits fall into this category. However, there are structures that can not fit into such a category. A typical example is a patch antenna with finite substrate.

Practically, a patch antenna always has finite substrate or its substrate is finite size in the horizontal direction. However, normally the substrate size is significantly larger than the patch size. We can consider it as a patch on top of a horizontally infinitely extended substrate. Such a model can yield very accurate results normally.

Why do we want to assume infinitely extended substrate horizontally? The reason is in the Green's functions. When we assume the substrates are layered, we are able to solve the Green's functions analytically. The only unknown is the current density on the patch only. The field continuities inside the substrates (or dielectrics including air) will be guaranteed in the solution even we do not solve the field. Such a model is very accurate and reduces the computational effort significantly.

When the substrates are finite, we can not find the Green's functions matching the field continuities conditions on the surface of the substrates. If we want to solve the problem precisely, we have to solve the field distribution in the substrates (or dielectrics), as well as the current density distribution on the patch. Such a model costs much more in computational effort. Starting from the IE3D 10.1, we are able to solve structures with finite substrates. Basically, we define the finite substrates as finite dielectric blocks. We still allow users to define substrates, which will be considered as infinitely extended. In the initial implementation, the IE3D 10.1can only accept 2 cases: (1) The whole space is filled with air; (2) The upper space is filled with air and the lower space is a ground plane. You can define as many finite dielectric blocks as you like. Also, the finite dielectric blocks are limited in z-directed cylinders with cross-sections of arbitrary shapes.

We have made a more general implementation on IE3D 11: Finite dielectric blocks in infinitely extended layered substrates. This general implementation will make IE3D more powerful in handling finite dielectric structures.

Section 16.1 Modeling of a Patch on Finite Ground and Infinite Substrate

For comparison reason, we will first model a patch antenna with finite ground and infinite substrate. The file is saved in: .\ie3d\samples\finite_ground_infinite_substrate.geo. The structure is pre-defined as 2 polygons: one rectangle of size 16 mm by 12.5 mm on z = 0.794 mm as the patch, and another rectangle of size 25 mm by 20 mm on z = 0 serving as the finite ground. The substrate is from z = 0 to 0.794 mm with permittivity of 2.2 and it is extended to infinity. We are going to define a probe-feed at y = -3 mm.

Step 1 Run MGRID. Please open .\ie3d\samples\finite_ground_infinite_substrate.geo. We will get the picture shown in Figure 16.1.

Step 2 Click at the No.2 layer (Z=0.794) on the Layer Window to focus the input at z = 0.794. Type Shift+A (equivalent to selecting Input->KeyIn Absolute Location command). MGRID will prompt you the dialog of Keyboard Input Absolute Location. Enter X = 0 and Y = -3. Select OK to enter the vertex on the patch.

Step 3 Select Entity->Probe Feed to Patch command. MGRID will prompt you for the Probe-Feed to Patch dialog. The entered vertex at (X, Y) = (0, -3) is used for the probe location.

Step 4 Change the Radius to 0.25 mm (see Figure 16.2). Select OK to continue. MGRID will create the probe fed patch antenna (see Figure 16.3a). It is a good model. However, we can make the

polygons better for meshing. Please select Adv Edit->Rectanglization. Please change the Minimum Division Size = 0.2 (default 0.4…). The reason we want it to be 0.2 is because the probe radius = 0.25. If we choose a value larger than 0.25, we may not get better result than it is in Figure 16.3a. Select Ok and we will get the result in Figure 16.3b. It is a better model because the meshing will create fewer cells even though the results should be the same.

Please save the geometry as: .\ie3d\practice\finite_ground_infinite_substrate_pfed.geo. Please simulate it from 7 to 8.5 GHz with 151 frequency points and AEC enabled (edge cell width = 0.2 mm). The total cells and unknowns are about 455 and 855. The simulation takes about 6 seconds per frequency point and less than 36 seconds for the whole frequency range. You will get the s-parameters shown in Figure 16.10. We will discuss the results later when we compare the results of the infinite substrate and the finite substrate.

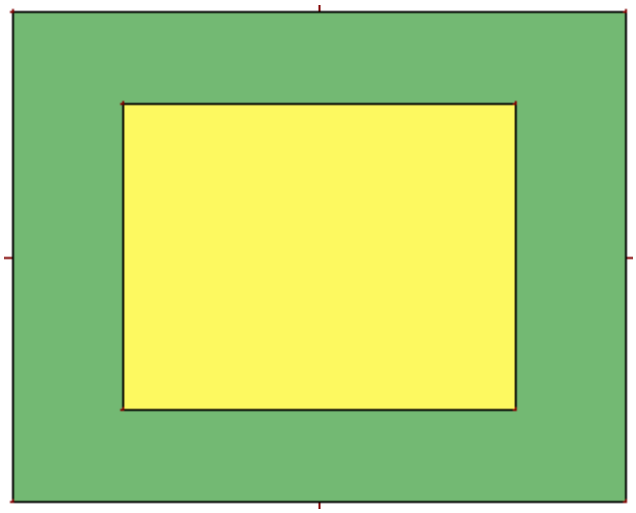


Figure 16.1 The 2 patches in finite_ground_infinite_substrate.geo.

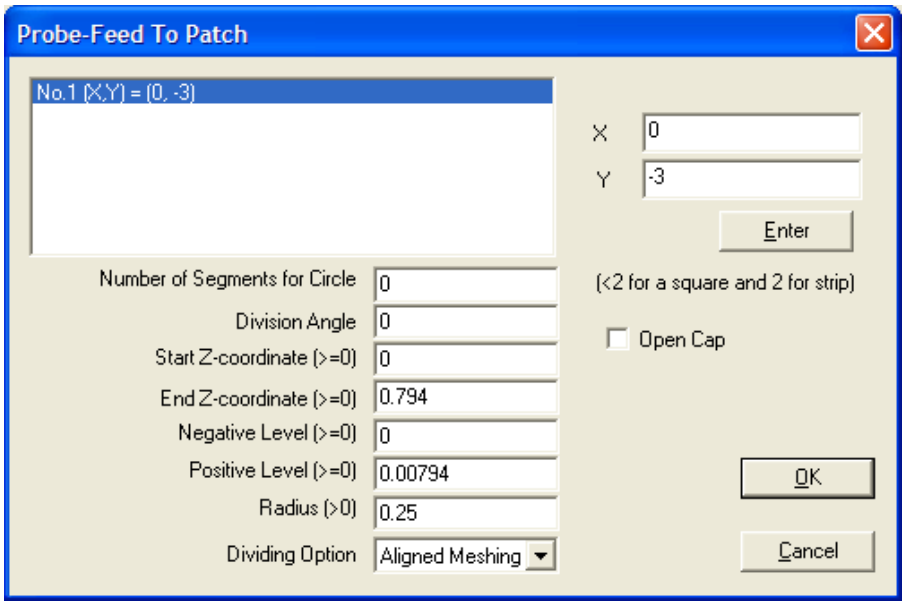
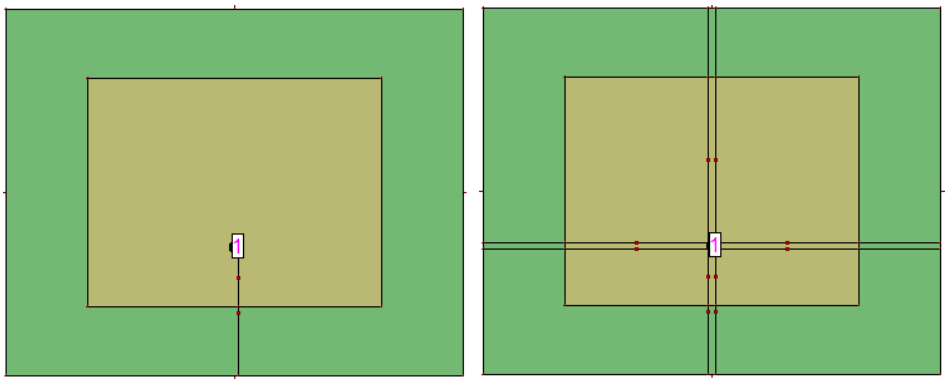


Figure 16.2 The Probe-Feed to Patch dialog after the Radius is changed to 0.25.



(a) After the probe feed built (b) After Rectanglization.

Figure 16.3 The probe-fed patch on infinite substrate built.

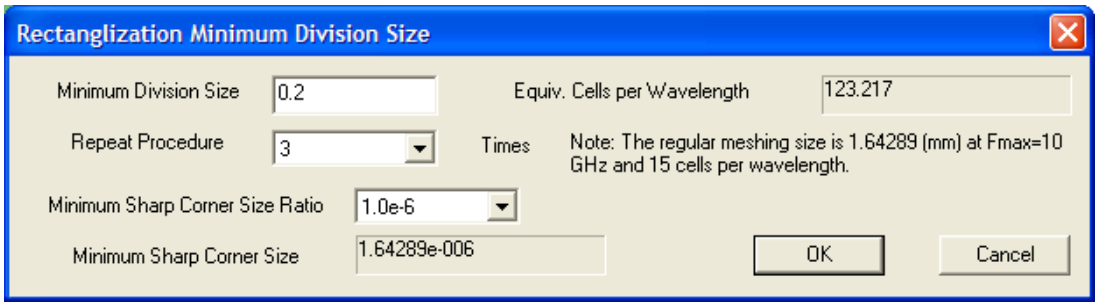


Figure 6.4 The Rectanglization dialog with the Minimum Division Size = 0.2.

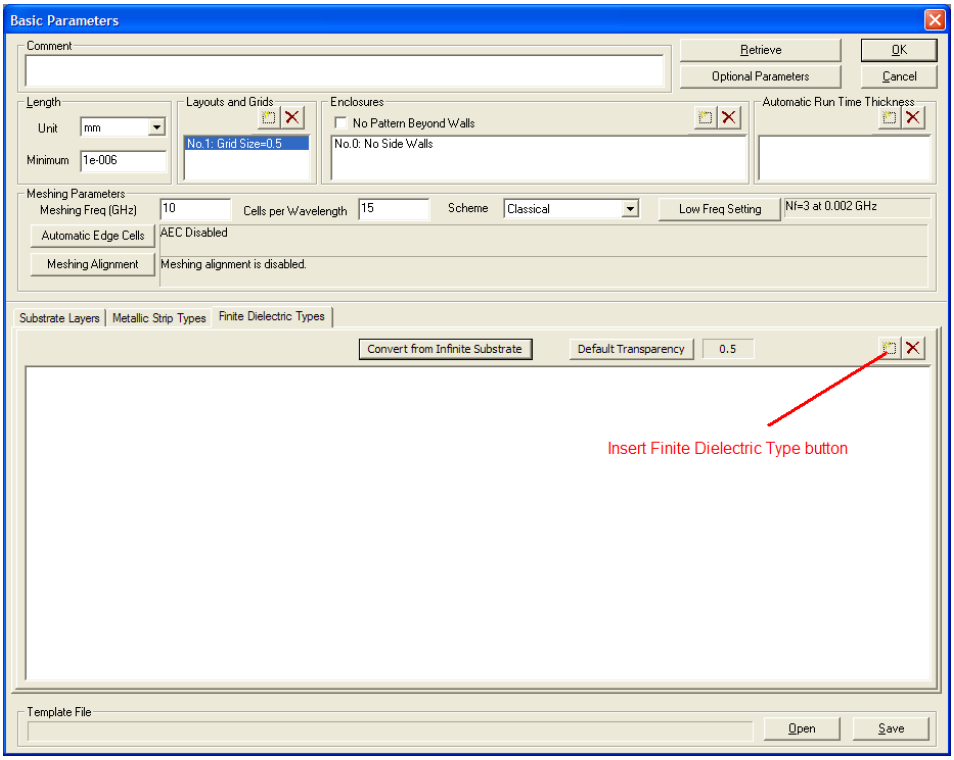


Figure 16.5 The Basic Parameters dialog.

Section 16.2 Modeling of a Patch on Finite Ground and Finite Substrate

Now it comes to the most important topic of this chapter: How can we model the patch with finite substrate?

Step 1 While the .\ie3d\samples\finite_ground_infinite_substrate_pfed.geo file is still opened. Please select Param->Basic Parameters command. The Basic Parameters dialog comes up. Please select the Insert button at the top right corner of the Dielectric Types list box (see Figure 16.5). The Add New Dielectrics dialog comes up. Please change the Dielectric Constant, Epsr from 1 to 2.2 (see Figure 16.6). Select OK and the defined dielectric type is created and entered into the list of the Dielectric Types in the Basic Parameters dialog. Please select the No.1 substrate (with Ztop = 0.794) in the Substrate Layers list box (see Figure 16.5). Please select the Delete button on the top right of the Substrate Layers to delete the infinite substrate from the list box. We will have No.0 substrate with Ztop = 0 and the No.1 substrate (original No.2 substrate) with Ztop = 1.0e+15 left. Both of them are air substrates. In means that everything is in free space right now.

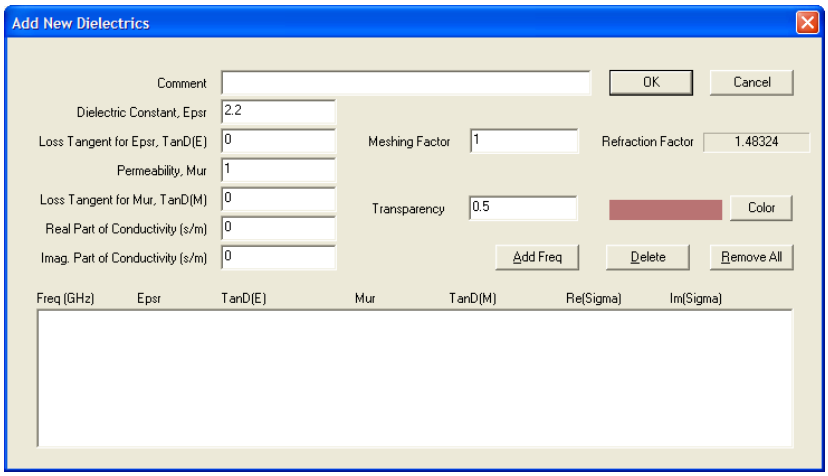


Figure 16.6 The Add New Dielectrics dialog after the Epsr is changed to 2.2.

Step 2 Select OK to get out of the Basic Parameters. The geometry is not changed. However, right now, the infinite substrate is gone and it is a probe-fed patch antenna with air substrate. We need to define the finite substrate for the structure. Assume we want the size of the finite substrate to be the same as the size of the finite ground plane. We are going to define the finite substrate based upon the polygons for the finite ground plane.

Step 3 Please select Edit->Select Polygon Group command. MGRID will be in the Select Polygon Group mode. By the default of default, the select focus should be on all the layers. However, you should understand that the default selection mode can be configured in the Param->Optional Parameters dialog. We can configure the default focus layer to be on the layer of our input focus. Anyway, please check the check box for the Z = 0 (No.0 layer) and un-check all the other layers. We want to focus the selection to the Z = 0 layer only.

Step 4 Please window the polygons for the finite ground. The polygons become blackened indicating they are selected. We are using the polygons for the finite ground plane to define the shape of the finite substrate.

Step 5 Select Adv Edit->Define Dielectrics Call. The Define Selected Polygons as Dielectrics Call dialog comes up. Please click the No.0 Dielectric Type. Please enter the Comment as: Finite

Substrate. Define the Z1 = 0 and Z2 = 0.794. Remember to check the "Keep original polygons after defining dielectrics". This option is critical for our current structure. Basically, we are using the polygons for the finite ground to define the finite substrate. After we define the selected polygons as a finite dielectric block, the polygons will be copied internally for the dielectrics call. We have the option to "Keep the original polygons after defining dielectrics". If we check the option, the original polygons will be retained. If we un-check it (by default), the polygons will be removed. If we keep the polygons, we also have the option whether we will keep them selected after the procedure. If we check Keep Polygons Selected, the retained polygons will be still selected after we select OK. For our case, the original polygons are for the finite ground plane. We want to keep it. However, we do not need to select them after OK. Therefore, please check the option for "Keep original polygons after defining dielectrics", and un-check the option for "Keep Polygons Selected". Please select OK. The finite dielectric is defined.

However, you do not see any change in the main window of the layout editor. Basically, the finite dielectric is displayed as wire frame on the top view. Right now, the shape of the finite dielectric is the same as the finite ground plane and the frame display is not visually clear on it on the top view.

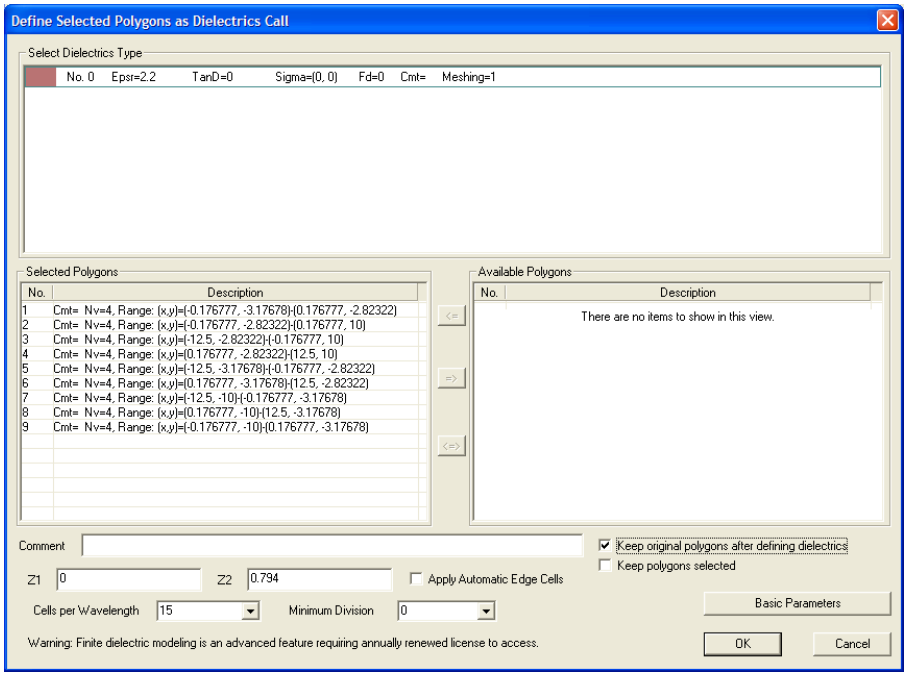
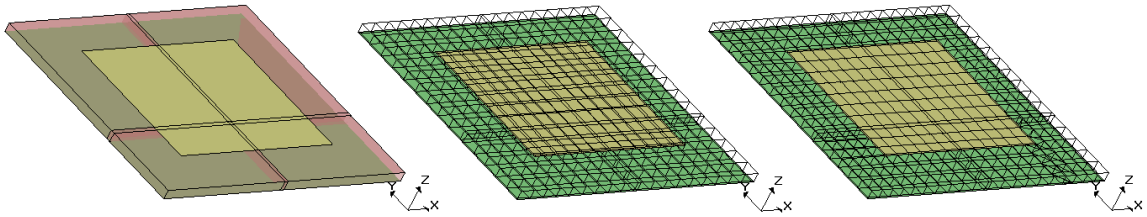


Figure 16.7 The Define Dielectrics Call dialog.

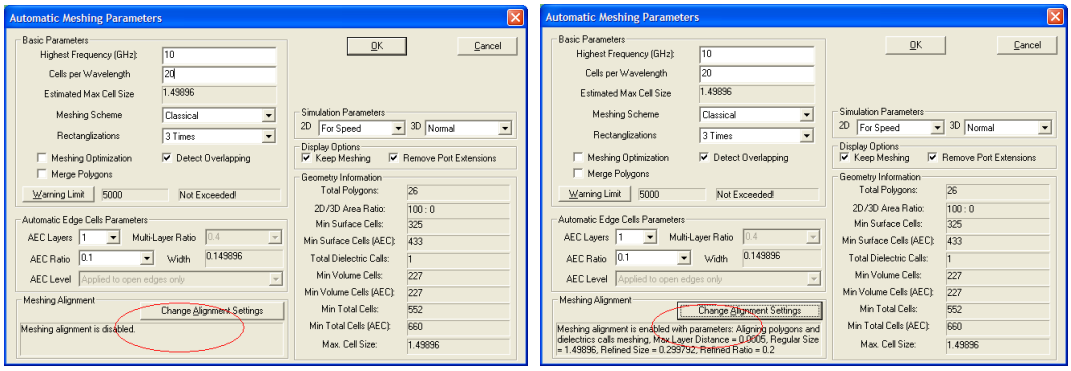
Step 6 Please select View->3D View command on MGRID. The 3D view comes up. You will see the finite dielectric is shown transparently (see Figure 16.8a).

Step 7 Please select Process->Display Meshing command. Please change the Cells per Wavelength, Ncells = 15 to 20. Why we want to do it? Basically, when we model the structure on infinite substrate, we can estimate the waveguide wavelength based upon the substrate configuration. For finite dielectric modeling, it is not so simple to estimate the waveguide wavelength. The meshing will be based upon the wavelength in free space. In order to make the meshing comparable to the infinite substrate case, we should increase the Cells per Wavelength appropriately so that the final meshing density will be similar. We can not expect the meshing density to be the same for the 2 cases.



(a) Finite dielectrics displayed transparently (b) Un-aligned meshing (c) Aligned meshing.

Figure 16.8 The finite dielectric in 3D view



(a) Meshing Alignment Disabled (b) Meshing Alignment Enabled.

Figure 16.9 The Automatic Meshing Parameters dialogs.

Step 8 Please also pay attention to the Meshing Alignment section in Figure 16.9a. Please make sure it is disabled. Select OK. We will get non-aligned meshing between the dielectrics and the patch (see Figure 16.8b). Save the file as: .\ie3d\practice\finite_ground_finite_substrate_pfed1.geo

Step Please do the meshing again and enable "Align and Mesh Polygons and Dielectrics". We will get the picture in Figure 16.9b. Select OK. We will get the aligned meshing between the finite dielectrics and the patch (see Figure 16.8c). Please save the file as: .\ie3d\practice\finite_ground_finite_substrate_pfed2.geo

Why did we want to do the above procedure? The reason is that meshing alignment between the finite dielectrics and the patch is extremely critical to the simulation results. We want to demonstrate it to you.

Step 6 Please simulate both structures from 7 to 8.5 GHz with 151 frequency points. The unaligned structure creates 560 surface cells, 285 volume cells and 2232 unknowns while the aligned structure creates 644 surface cells, 359 volume cells and 2683 unknowns. The un-aligned structure takes about 96 seconds per frequency point (less than 10 minutes for the frequency range) while the aligned structure takes about 124 seconds per frequency point (about 12 minutes for the whole frequency range). Certainly, with finite dielectrics, the simulation speed is significantly slower.

Comparison among the 3 cases (infinite substrates, finite substrate with un-aligned meshing, and finite substrate with aligned meshing) is shown in Figure 16.10. As you can see, the curves in Smith Chart are almost identically. From the Cartesian display, there are some shifts in the resonant frequency. The "infinite" model predicts the resonance at 7.64 GHz. The "un-aligned"

model predicts the resonance at about 7.87 GHz (about 3% off "infinite"). The "aligned" model predicts the resonance at about 7.75 GHz (about 1.5% off "infinite").

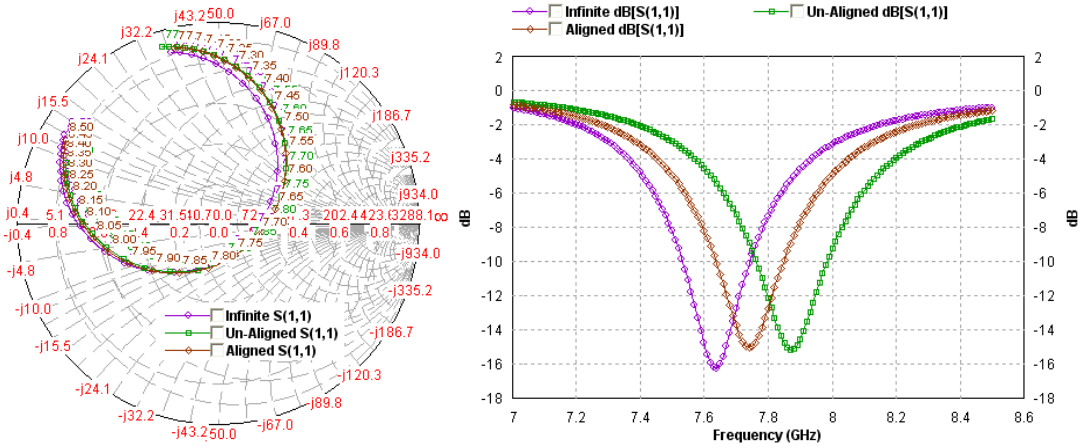


Figure 16.10 The comparison of infinite substrate and finite substrate.

From our experience, aligned meshing between the polygons and the finite dielectrics normally is much more accurate than the case with un-aligned meshing. It is also the reason for us to implement the meshing alignment in the IE3D 11.

What should be the resonance precisely at? It should be somewhere between the "infinite" model at 7.64 GHz and the "aligned" model at 7.75 GHz. Finite dielectric modeling certainly allows us to model the boundary conditions on the dielectric interfaces of the truncation portion more accurately. However, it will be less accurate on modeling the boundary conditions on the layer interfaces better. Infinite substrate model guarantees the boundary conditions analytically through the Green's function while finite substrate model numerically enforces it there. Both approaches have their strong points and weak points.

Is the result accurate enough? You can increase the meshing density and apply AEC (enabled for each finite dielectric call). You will see little difference in the results. We found that edge cells are not critical to finite dielectrics. Therefore, by default, we do not apply edge cells to dielectrics calls. However, we do want to keep the option. For each dielectrics call, we allow you to define whether you want to apply AEC on the dielectrics call (see Figure 16.7). If you check the "Apply Automatic Edge Cells" on the dielectrics call, the meshing program will create edge cells on the call if the AEC option is checked in the Simulation Setup. Otherwise, the edge cells will not be created on the dielectrics call.

Section 16.3 Precise Modeling of Radiation at θ = 90˚ Direction

IE3D's infinite substrate model has been very good for many different kinds of antennas. Normally, it will predict very accurate input impedance. For radiation patterns, it is also very accurate for microstrip antennas. However, as you can see from the finite ground patch antenna example in Chapter 7, the infinite substrate model always predicts a notch in the radiation pattern at θ = 90 degrees. This notch is created by surface waves due to the infinite extended substrates and it can not be eliminated with the infinite substrate model. Normally the notch will not affect the properties of an antenna if its main radiation is at the broadside (θ = 90 degrees). However, if it is an end-fire antenna, you will see strange radiation pattern at θ = 90 degrees even though the input impedance can be quite accurate. If we model it using finite substrate, we will be able to calculate the pattern at θ = 90 degrees accurately.

Figure 16.11 shows the comparison in the radiation patterns between the "infinite" and "aligned" models. They agree very well except at the θ = 90 degrees. The detailed antenna properties are shown in Table 16.1. They agree very well except the "Infinite" model predicts lower efficiency. This is due to the surface wave. Surface wave is considered as losses. For the finite substrate model, the surface wave will be bounced back or radiated out as space wave and the radiation efficiency should be approaching 100% if there are no metallic and substrate losses.

Please note that we provide two values for the 3 dB beam width. The 1st one is the minimum value and the 2nd one is the maximum value from the oval cone shape.

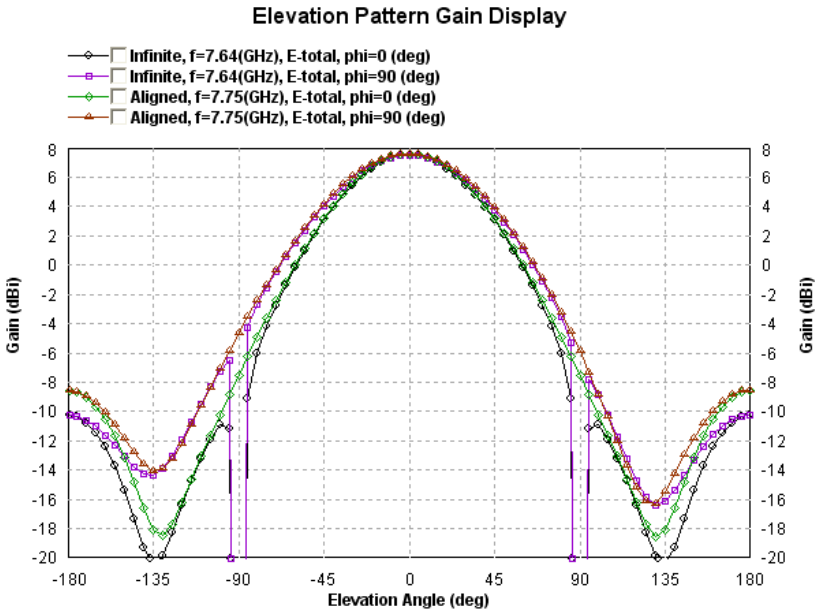


Figure 16.11 The comparison in pattern between infinite and finite substrate models.

Table 16.1 The comparison in pattern properties.

Property	Infinite Model	Aligned Finite Model
Resonant Frequency (GHz)	2.64	2.75
Radiation Efficiency	92.17%	98.49%
Antenna Efficiency	89.97%	95.35%
Conjugate Match Efficiency	46.08%	49.24%
Gain	7.505 dBi	7.624 dBi
Directivity	7.964 dBi	7.31 dBi
3 dB Beam Width	(73.58, 81.90) Degrees	(73.41, 82.08) Degrees
Conjugate Match Gain	4.600 dBi	4.754 dBi

Section 16.4 Meshing Alignment for Higher Accuracy

We have pointed out that meshing alignment between the finite dielectrics and the metal traces on them is extremely critical to accurate modeling of finite dielectric structures. For this reason, we have implemented the automatic meshing alignment on IE3D 11 for it. The procedure is an extremely sophisticated one due to the fact there are countless different situations. We have been testing and improving the scheme for a very long period. However, occasionally, it may still cause difficulty. In this initial release of IE3D 11, we would like to suggest you to check the meshing with meshing alignment before you perform a simulation to make sure there is no meshing problem involved. In case you see something strange, you may want to use the command Adv Edit->Meshing Alignment on MGRID to check

the result of meshing alignment. It is basically using the same kernel as the one in the meshing scheme for display and simulation. Another way is to use the Adv Edit->Build Holes and Vias from Selected Polygons command to align the divisions between the polygons for the traces (or patches) and the polygons to be defined as a dielectric call before you define the call (see Appendix AE). You can also use the Adv Edit->Separate Polygons command to divide polygons along specified lines. Those commands are extremely robust and they can do much advanced work for you.

Section 16.5 Modeling of Wire Bonds in Inhomogeneous Dielectrics

The above sections have demonstrated the capability of the IE3D in modeling finite dielectric structures. The procedure is quite straight-forward.

Another type of structures the finite dielectric modeling capability may help much are wire bonds in inhomogeneous dielectrics. Wire bonds are used to connect different IC modules together. Normally, each module has its own dielectric layers. In the other word, the 2 sides of a wire bond may have different dielectric configuration (see Figure 16.12). Precise modeling of such a structure on the IE3D 10.0 or earlier editions is not possible. Normally, we have to assume the same dielectric layers for both sides of the wire bond. Now, we are able to model such true 3D structures with the finite dielectric modeling capability.

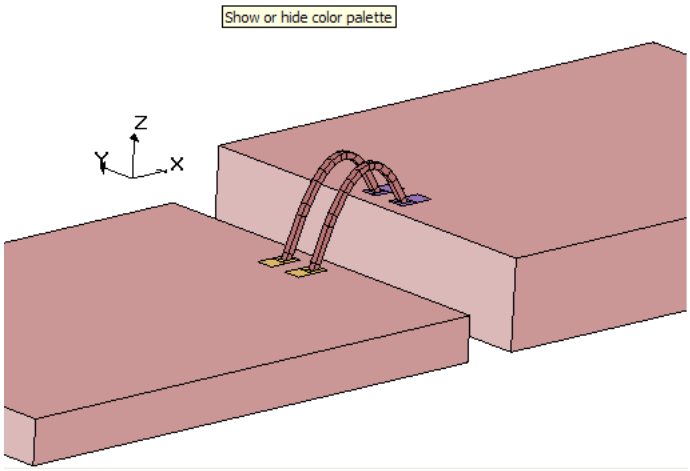


Figure 16.12 Two coupled wire bonds between two dielectric blocks.

Saved in .\ie3d\samples\wire_bonds_gap.geo is a coupled wire bond structure. The structure is shown in Figure 16.12. The left side of the wire bonds is on a substrate of thickness 0.5 mm, while the right side of the wire bonds is on a substrate of thickness 1.0 mm. Each wire bond spans about 1.2 mm center-to-center horizontally. The total length of each wire bond along the center line is about 2.05 mm. There is extra strip of length 0.3 mm on each side. The total physical length between 2 ports is about 2.65 mm. The radius of the wire bond is about 0.05 mm. The gap between the dielectric blocks is 0.6 mm. Certainly, we can ignore the dielectrics and assume the wire bonds are in air. The model is in \ie3d\samples\wire_bonds_air.geo. The structure can be built easily using the Edit->Wire Bonding command on MGRID. Simulating the wire_bonds_air.geo takes a few seconds for the whole frequency range. Simulating the finite dielectric model (or wire_bonds_gap.geo) takes about 153 seconds per frequency with meshing alignment. The s-parameters of the 2 models are compared in Figure 16.13, showing little difference between the 2 models.

Using MODUA, we are able to extract the RLC equivalent circuit of the coupled wire bonds. The comparison between the two models is shown in Table 16.2. It is indicated that the finite dielectric blocks reduce the self and mutual inductances slightly even though we can not see the difference in the corresponding s-parameters.

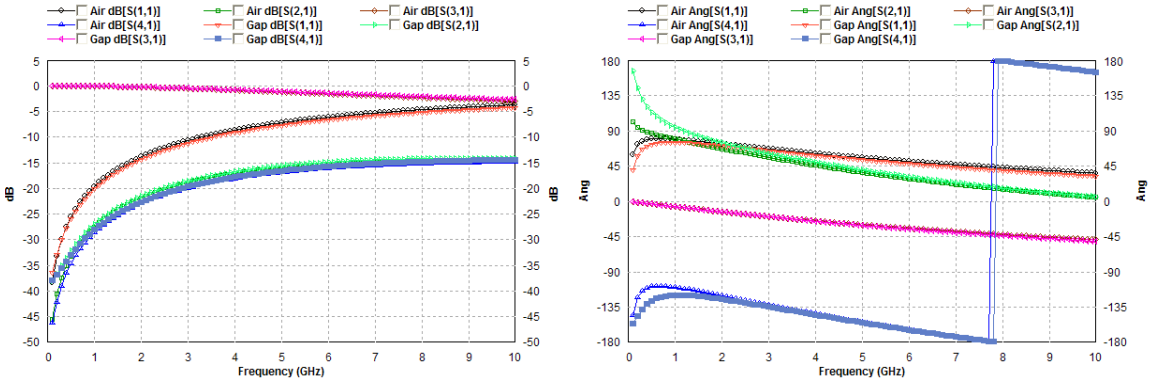


Figure 16.13 The comparison of air dielectric model and finite dielectric model.

Table 16.2 The extracted inductance at 0.1 GHz.

Model	Air Substrate Model	Finite Dielectric Model
File Name	Wire_bonds_air.geo	Wire_bond_gap.geo
L(1,1) (nH)	1.800	1.761
L(1,2) (nH)	0.637	0.628

Construction of the finite dielectric model (wire_bonds_gap.geo) is quite straight-forward. For the wire bonds, you build the 4 bond pads first. Then, we can enter 4 vertices at the locations where we want the wire bonds to connect to the bond pads. The order of the vertices is critical. We should have the 4 vertices in the order of 1, 3, 2 and 4 where the number indicates the bond pad where we will define the ports (see Figure 16.14). After we enter the 4 vertices, we can use the Edit->Wire Bonding command to build the wire bonds. For this structure, we choose the Symmetric Style with Z-Extremum at 1.5 mm. Then, we will build 2 polygons to be used to define the finite dielectric blocks (see the wire_bonds_for_gap.geo). Finally, we can select each extra polygon and define the polygon as finite dielectrics in the Adv Edit->Define Dielectrics Call command.

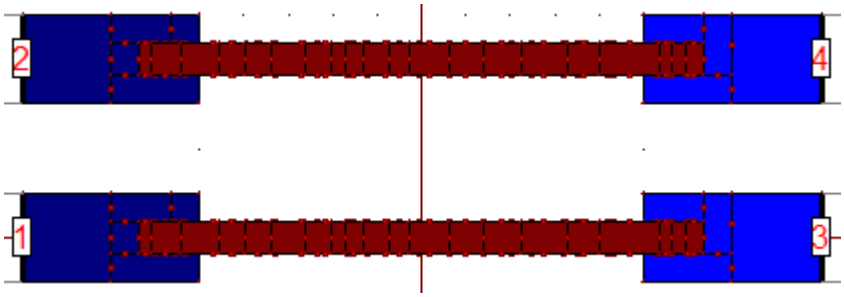


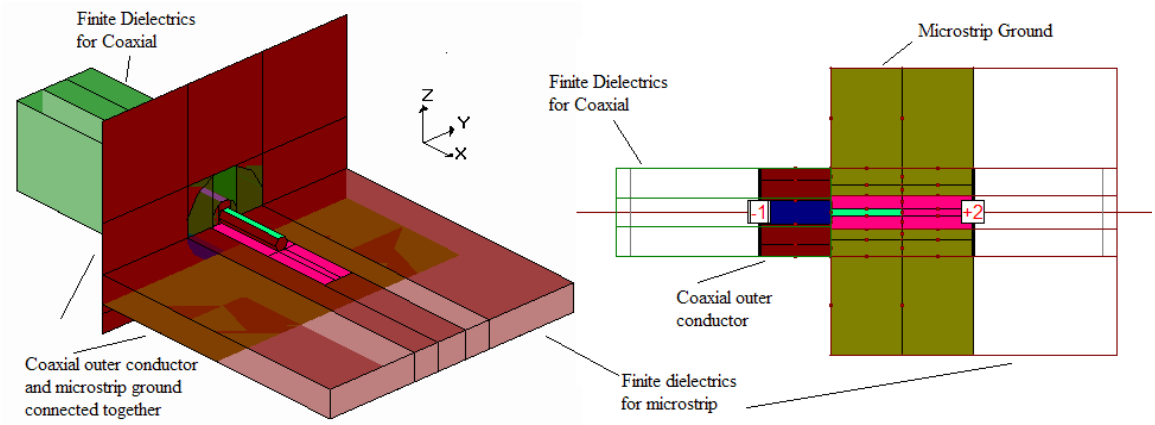
Figure 16.14 The top view of the wire_bond_air.geo structure.

Section 16.6 Modeling of SMA Connector to Microstrip Transition

Another type of frequently encountered problems requiring 3D dielectric modeling capability are the problems of coaxial connectors to microstrip structures. The coaxial takes one type of dielectrics and the microstrip types another type. Precise modeling of the structure using layered dielectric configuration is not available. With the 3D finite dielectric modeling capability, we are able to simulate the structure precisely.

Figure 16.15 is a typical example. The diameter of the coaxial outer conductor is 63.5 mils. The diameter of the coaxial inner conductor is 20 mils. The dielectric permittivity for the coaxial filling is 2.05. The probe from the coaxial inner conductor is of length 50 mils and diameter 10 mils. The probe is pressed against the microstrip with a good electrical connection. The microstrip is 23 mil wide. The dielectric

constant of the microstrip substrate is 9.8. As it is required by the IE3D simulator, we have to add extra length on both the coaxial line and the microstrip line. The extra coaxial line is 50-mil long, and the extra microstrip line is 100-mil long from the junction.



(a) 3D view of the connector (b) Top view of the connector

Figure 16.15 A SMA to microstrip transition connector.

Construction of the connector is tedious. To simulate it, we need to build the coaxial line and the microstrip line horizontally in order to define extension ports on them. However, it is easier to handle the coaxial part with it directing in the z-direction. We will discuss the basic idea in the following. The lengthy detailed procedures are not documented here.

We build the coaxial inner and outer conductors in: connector_p1.geo. The outer conductor is from z = 0 to 50 mils. The inner conductor is from 20 to 50 mils. We purposely leave a difference of 20 mils in the length in consideration of defining the ports later. We build a cap on the inner conductor in order to build a probe from it. Also, we make sure some polygons have constant x-values. Those polygons will be used to define the ports later when we swap the X and Z-coordinates later.

To build a probe on it, we can build a 6-sided polygons on z = 100 mils. Then, we can select it and use the Adv Edit->Build Holes and Vias on Selected Polygons command to build the probe from z = 50 to 100 mils. Please check either the Keep Selected or Fill the Hole option in order to remain the polygon on z = 100 mils. The result is shown saved in connector_p2.geo.

The next step is to build the microstrip connected to the probe. Please note that the microstrip is not only connected to the probe. It is also connected to the inner conductor at the end. To do it, we need to divide the cap of the inner conductor. Saved in connector_p3.geo is the example.

The next step is to build the finite ground for the microstrip line. The geometry with the finite ground plane built is saved in: connector_p4.geo.

Now, the basic coaxial connector is built. We can select all the polygons and use the Adv Edit->Flip Coordinates command to swap X and Z. MGRID will prompt you to provide a value for the New Z Offset to Old X. it is required to be at least 44.33012702 mils. Please enter the value for it (see Figure 16.16). We will get the connector_p5.geo. After the rotation, the microstrip is on z = 40 mils and the finite ground is on z = 20 mils.

We can try to build the other parts on the structure. The other parts should be simpler. Also, for the coaxial ports, we need to define +1 on the inner conductor and the –1 on the outer conductor. After we

define the ports, we can select the vertices of the inner conductor at X = 20 mils (at Z = 20 mils before flipping). Then, we can move them to X = 0 to align with the outer conductor. The final geometry is saved in: connector_p6.geo.

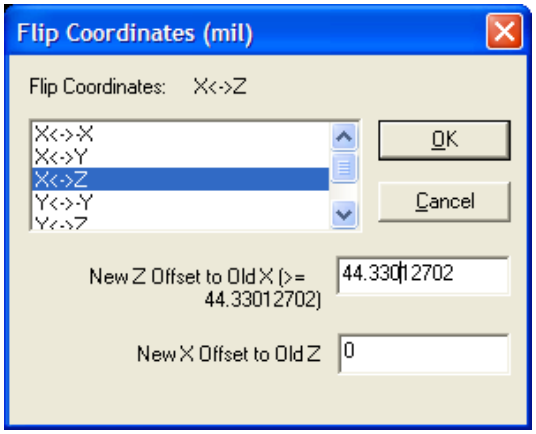


Figure 16.16 The Flip Coordinates command.

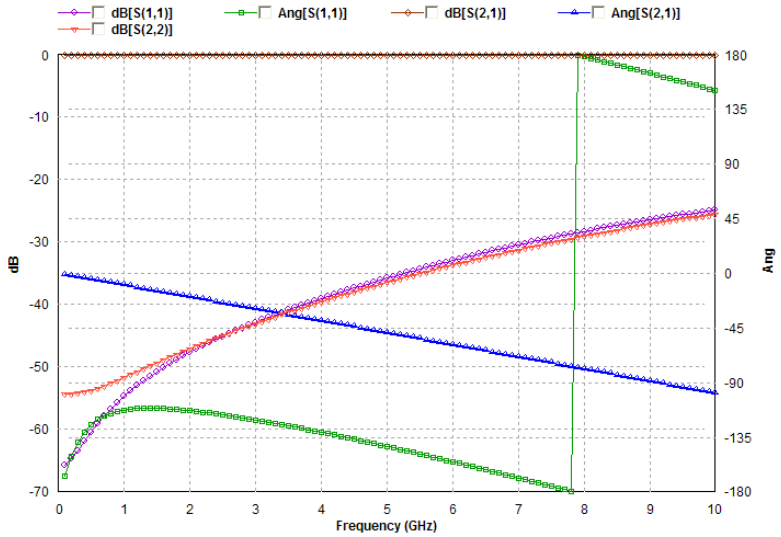


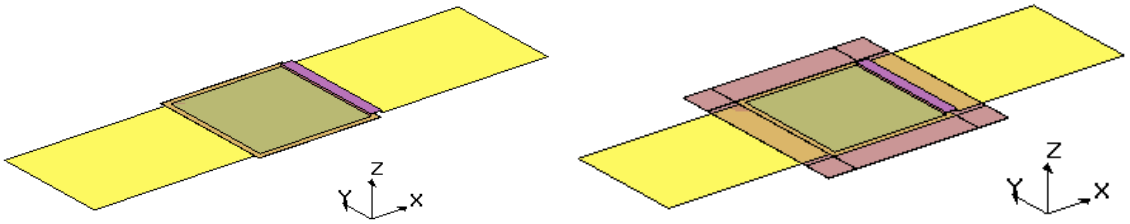
Figure 16.17 The simulation results of the SMA connector to microstrip transition.

Due to the fact that the meshing of the finite dielectrics will not match the one on the coaxial connectors, the simulation results do not converge very fast. The users are suggested to use finer meshing for better results. If possible, the user should try to reduce the length of the coaxial line for better accuracy. For this particular example, we use Fmax = 70 GHz and Ncells = 15 cells per wavelength. The finite dielectric block for the coaxial part is meshed into 6 by 6 cells in the cross-sectional directions. The finite dielectric block for the microstrip substrate is meshed into 2 cells in the z-direction. The microstrip part should be fine enough. However, we should be more careful with the meshing on the coaxial part. Due to the fact that the current IE3D 10.1 will not yield matched meshing for the coaxial conductors and the filling, the convergence is normally much slower than the microstrip part.

The simulation results on the connector_p6.geo are shown in Figure 16.17. The S(1,1) is below –35 dB below 5 GHz and it is below –25 dB below 10 GHz. This is a standard design and the results are quite good. Changing the meshing may change the S(1,1) slightly. However, if the meshing density is not fine enough, the results may not be good.

Section 16.7 Mixed Infinite and Finite Dielectrics Modeling

One of the most significant implementations in the IE3D 11 is the mixed infinite and finite substrate modeling. Finite dielectric was introduced in IE3D 10.1 in early 2004. It opens a door for IE3D to model 3D structures involving finite dielectrics. However, in the IE3D 10.1 implementation, we can only have either infinite substrates or finite dielectrics. We can not have the mixed cases. In many applications, a structure mainly has very large substrates which can be modeled by infinite substrates very accurately while it may have very limited area with finite dielectrics. In such a case, modeling everything with finite dielectrics is very inefficient. A typical example is an MIM capacitor or a structure with holes in the substrates. The 3D dielectric area is very small. We can use finite dielectrics to model the limited thin layers or holes while we can still use infinite substrates to model the large substrates on IE3D 11 efficiently and accurately. We will take the MIM capacitor we modeled in Chapter 6 as an example. The structure can be simulated efficiently and accurately using infinite extended substrates in .\ie3d\samples\mimcap1_refined.geo. We have built two finite dielectrics models for it: (1) One model with the finite dielectrics of the same size as the bottom plate of the MIM capacitor saved in .\ie3d\samples\mimcap1_finite_same.geo; (2) The other with the finite dielectrics larger than the size of the bottom plate of the MIM capacitor saved in .\ie3d\samples\mimcap1_finite.geo. The structures are compared in Figure 16.18.



(a) Same size finite dielectrics (b) Larger size finite dielectrics

Figure 16.18 The two finite dielectric models for the MIM capacitor.

The simulation results are compared in Figure 16.19 against the infinite substrate model saved in .\ie3d\samples\mimcap1_refined.geo. It is interesting to note that the results from mimcap1_finite_same.geo (Same Finite Size) agree with the mimcap1_refined.geo (Infinite) perfectly, while the dB[S(1,1)] from mimcap1_finite.geo (Larger Finite Size) is slightly off. Table 16.3 shows more comparison between them. There are very slight differences in the parameters of the equivalent circuit, which are much more sensitive than s-parameters, between the different models. The agreement between the infinite model and the finite models demonstrate the accuracy and robustness of the IE3D. Both thin substrate modeling and finite substrate modeling are extremely difficult and sophisticated. IE3D can do extremely well on both modeling schemes.

Table 16.3 The comparison between the finite dielectrics models and the infinite substrate model.

Parameter	Parallel Plate	Infinite	Same Finite Size	Larger Finite Size
File Name	N/A	Mimcap1_refined.geo	Mimcap1_finite_same.geo	Mimcap1_finite.geo
Cells/Volumes/Unknowns	N/A	212/0/366	238/83/723	238/100/793
Seconds / Frequency	N/A	29	55	76
C(1,1) (pF) at 1 GHz	N/A	0.05782	0.05730	0.05791
C(2,1) (pF) at 1 GHz	2.97	3.070	3.075	3.009

Section 16.8 Summary on Finite Dielectric Modeling

We have discussed the modeling of finite dielectrics on IE3D. In the following, we would like to summarize the key points in using this newly implemented advanced feature.

(a) Finite dielectric modeling is an effective way to model true 3D structures.
(b) The volume meshing and the surface meshing are required to be dependent upon each other.
(c) However, matched meshing between the finite dielectrics and the metallic strips may speed up the convergence speed significantly.
(d) AEC is normally not required for finite dielectrics. Also, the meshing density is not very critical when the meshing of the metallic plates and the finite dielectrics is matched. Normally, one cell in the thickness direction is good enough for modeling finite dielectric patch antennas.

The finite dielectric modeling capability greatly enhances the modeling capability of the IE3D. The IE3D 11 allows modeling of finite dielectric blocks inside infinite substrates. It makes it more versatile. The current implementation does require the finite dielectric blocks to z-directed cylinders of arbitrary cross-sections. Further improvement is being carried on to expand the full 3D modeling capability of the IE3D. We should see much more improvement in the near future.

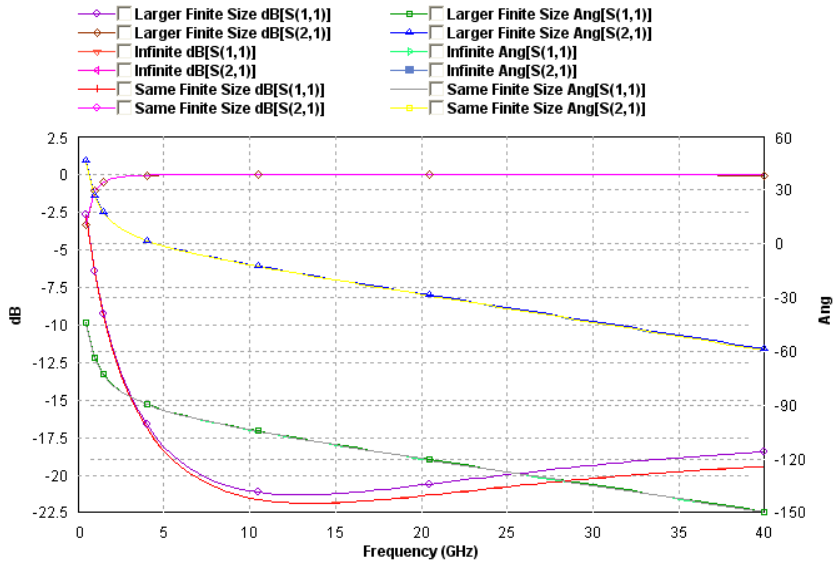


Figure 16.19 The comparison between the finite dielectric models and infinite substrate model.

Appendix A. Optional Parameters and Setup of IE3D

The optional parameters are configured in the **Parameters->Optional Parameters** dialog on MGRID. You can configure the optional parameters for the current geometry or for the application. When a new IE3D geometry is started, it will use the optional parameters for the application by default.

Correct setup of the Optional Parameters will make your use of the IE3D more convenient, efficient and accurate. The dialog for the optional parameters is shown in Figure A.1. In the Optional Parameters dialog, you can configure many things for the MGRID and IE3D. They become the default settings for your parameters when you use the MGRID and IE3D. We will discuss the major optional parameters in the following.

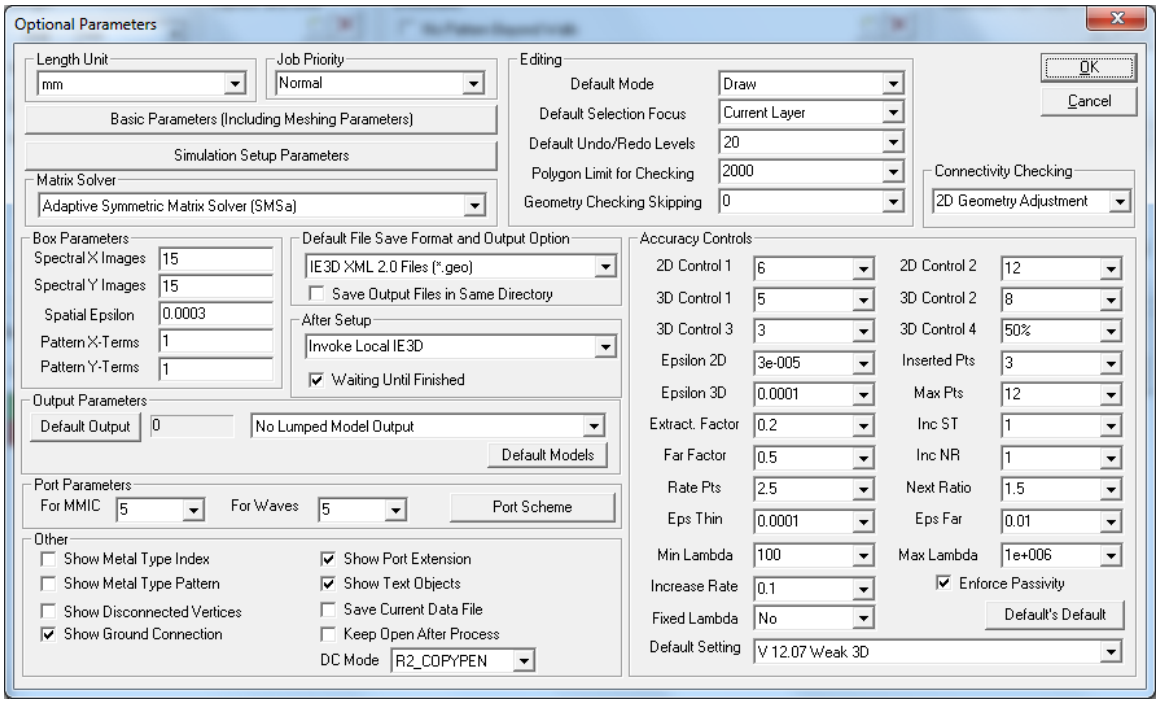


Figure A.1 The Optional Parameters dialog

1. Basic Parameters:

We have been defining Basic Parameters for each IE3D geometry. The IE3D 11 allows us to use many of the defined basic parameters for a new IE3D structure we create.

2. Length Unit:

Length Unit is one of the Basic Parameters. It allows a user to choose the length unit he is most familiar for layout editing.

3. Conductor Assumption Limit (CAL):

CAL is one of the Basic Parameters. We define the infinite ground plane as a substrate with high conductivity. The CAL is used to determine whether a substrate is an infinite ground plane or a lossy substrate. Internally, they are treated differently. Normally, we should set CAL to 1.0e+6.

From (3) of Chapter 3, we know that the conductivity σ is really related to the ε_{rc} in the equation. For HTS superconductor, the σ in the formula is small. However, $|\mathrm{Im}(\varepsilon_r)|$ is a big number. Therefore, our criteria for infinite ground plane is actually big $|\varepsilon_{rc}|$. Due to the contribution of σ to ε_{rc} is frequency dependent. We will use the $|\varepsilon_{rc}|$ at 1 GHz for our criteria. We define the CAL in this way: When $|\varepsilon_{rc}|$ at 1 GHz < CAL, we consider the dielectrics as normal dielectrics. When $|\varepsilon_{rc}|$ at 1 GHz ≥ CAL, we consider the dielectrics as ground plane. In the Edit Substrate dialog (see Figure 3.6), there is a parameter Factor. It is basically the $|\varepsilon_{rc}|$ at 1 GHz.

Because we compare the CAL to $|\varepsilon_{rc}|$ at 1GHz), we would not know directly whether a lossy substrate with finite σ is considered as a ground plane or just a normal lossy substrate. For most circuits with the σ of the ground plane about 10^7 (s/m), we really want to consider it as a ground plane. For some semi-conductor substrate, there might be some dielectric layer with the σ about 1000 (s/m). This dielectric layer will be on top of the true ground plane with the σ of the ground plane about 10^7 (s/m). In such a case, you may want to model the effect of the semi-conductor substrate. You should define the right CAL until you see the horizontal line on the metal layer on the interface of the semi-conductor substrate disappears.

Saved in **.\ie3d\samples\cal1.geo** is an example. The dielectric setup for the structure is shown in Figure A.2. When you open the file on MGRID, you will see 3 polygon layers in the Layer Window. No.0 layer at Z = 0 mm., No.2 layer at Z = 0.25 mm and No.2 layer at Z = 0.5 mm. You will see a line across the No.1 layer at Z = 0 mm, meaning it is an infinite ground plane. When you set the CAL in the **Param->Basic Parameters** dialog to 10000, you will see there is a line across the No.3 layer at Z = 0.5 mm. It means that the interface at z = 0.5 mm is the interface of a high conductivity conductor. For our case, the dielectrics from z > 0.5 mm is considered as the top ground plane.

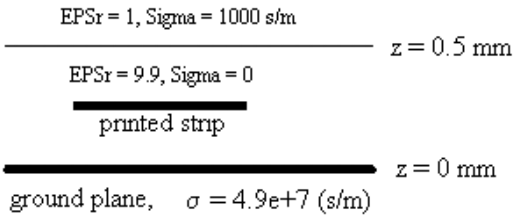


Figure A.2 The dielectric setup for **.\ie3d\samples\cal1.geo**.

4. Max DK:

When IE3D does a meshing, it will use the estimated effective dielectric constant Ereff to find the meshing size. The estimated Ereff is from the substrate configuration. Normally, every regular substrate (except ground plane layer) is used to do the estimation. However, some substrate with large value of permittivity (Er) and permeability (Mur) may cause IE3D to over-estimate the Ereff. Over estimated Ereff may cause IE3D to mesh a structure into fine cells and it may slow down the simulation. The Max DK allows us to exclude those substrates with large value of (Er · Mur) from the list for the estimation of the Ereff.

5. Matrix Solver

There are quite some matrix solvers available on the IE3D. Interested users should read Chapters 12 and 13 for more information on different matrix solvers. We would like to give more comments on the Separation Distance (SD) in the following paragraph. The SD is used in the partial matrix solvers and iterative matrix solvers.

For IE3D, the matrix is always a full matrix. However, it is a diagonal dominant matrix. Many off-diagonal elements are insignificant especially for large structures. Each matrix element is corresponds to the coupling between 2 cells as shown in Figure A.3. The coupling between the cell 1 and cell 3 is much

weaker than the coupling between cell 1 and cell 2. The coupling between the cell 1 and cell 4 is even much weaker. The difference might be a few orders in the corresponding matrix elements. Because the coupling between cell 1 and cell 4 is so weak compared with other couplings, we may not need to consider it because it can be negligible. If we neglect the coupling between 2 far away cells, we will obtain a sparse matrix. Based upon the sparse matrix solver, we implemented the GEMS-F and GEMS-I iterative solver. We introduce a parameter called Separation Distance (SD) to identify which coupling is negligible. As it is shown in Figure A.3, for any cell with a distance to cell 1 smaller than the SD, the coupling between this cell and cell 1 will be considered. When a cell with a distance to cell 1 is larger than the SD, the coupling between the cell and cell 1 will be neglected at the 1st stage. Same rule applies to any cell in the layout. SD is defined as number of cells. Normally, we recommend users to choose the SD = 5 to 15 cells.

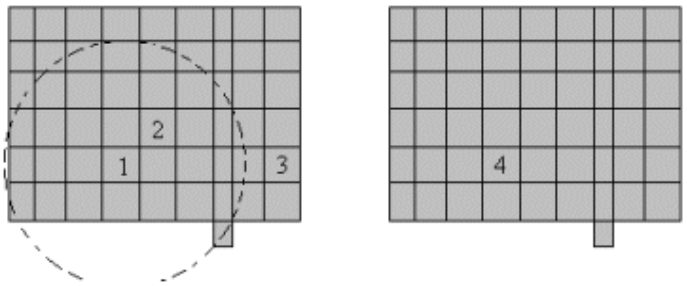


Figure A.3 A 2-element patch antenna array.

The selection of SD is very critical to the convergence of the GEMS solvers. There is no way to guarantee a SD with converged GEMS solution. Apparently, when we choose the SD to be 0, the GEMS will always diverge. When we choose the SD to be the largest distance between cells, the GEMS is basically FMS and the iteration always converges, and it also defeats the purpose of iterative solvers. We need to choose the right SD so that we can fast convergence with the least memory and time.

The Iteration Relative Error, Iteration Absolute Error, Maximum Iterations are for controlling the convergence of the iterative matrix solvers. They should not be changed normally. The Buffer Size should not be changed either.

For planar circuits and antennas, a good suggestion for SD is about 5-10 times of the substrate thickness. For 3D structures, the SD should be chosen to be larger because the off-diagonal terms do not decay so fast.

Normally, default SMSa is a very efficient matrix solver. It should be always used except for very large structures. Iterative matrix solvers GEMS-F and GEMS-I may be able to solve loosely coupled large structures much easier. When you choose FMS, SMSi, SMSa or GEMS solvers, IE3D will prompt you for the Matrix Solver Parameters. Please pay attention to the FMS TO SMSI SWITCH, SMSI TO GEMS-F SWITCH and GEMS-F TO GEMS-I SWITCH. If you set FMS TO SMSI SWITCH below 1000, IE3D will automatically choose the right solver for you when you choose SMSa. The SMSI TO GEMS-F SWITCH will allow IE3D engine to switch to GEMS-F solver from SMSi when no enough RAM is available. The GEMS-F TO GEMS-I SWITCH will do the same thing between GEMS-F TO GEMS-I. GEMS-I should not be used whenever RAM is enough because it can be very slow.

You have also implemented the FASTA option on SMSa for large structures. It is discussed in other appendix of the manual. It can help reducing RAM and it may speed up simulation.

6. Automatic Edge Cells (AEC) and Meshing Parameters:

AEC and Meshing Parameters are part of the Basic Parameters. The AEC is further improved. Please read Appendix AL for more information.

7. Cells on De-Embedding Arms:

The Cells on De-Embedding Arms (Ncdea) control how many cells we want to put on the de-embedding arm of an extension port. The larger the Ncdea is, the better accuracy. However, it will take more cells for a simulation. Normally, Ncede should be at least 3. The default is 5. Starting from IE3D 11, we can configure Automatic Adjustment for port extensions. IE3D will automatically detect whether a port extension is long enough based upon some user defined criteria. It will automatically add extra cells in case a port extension is too short.

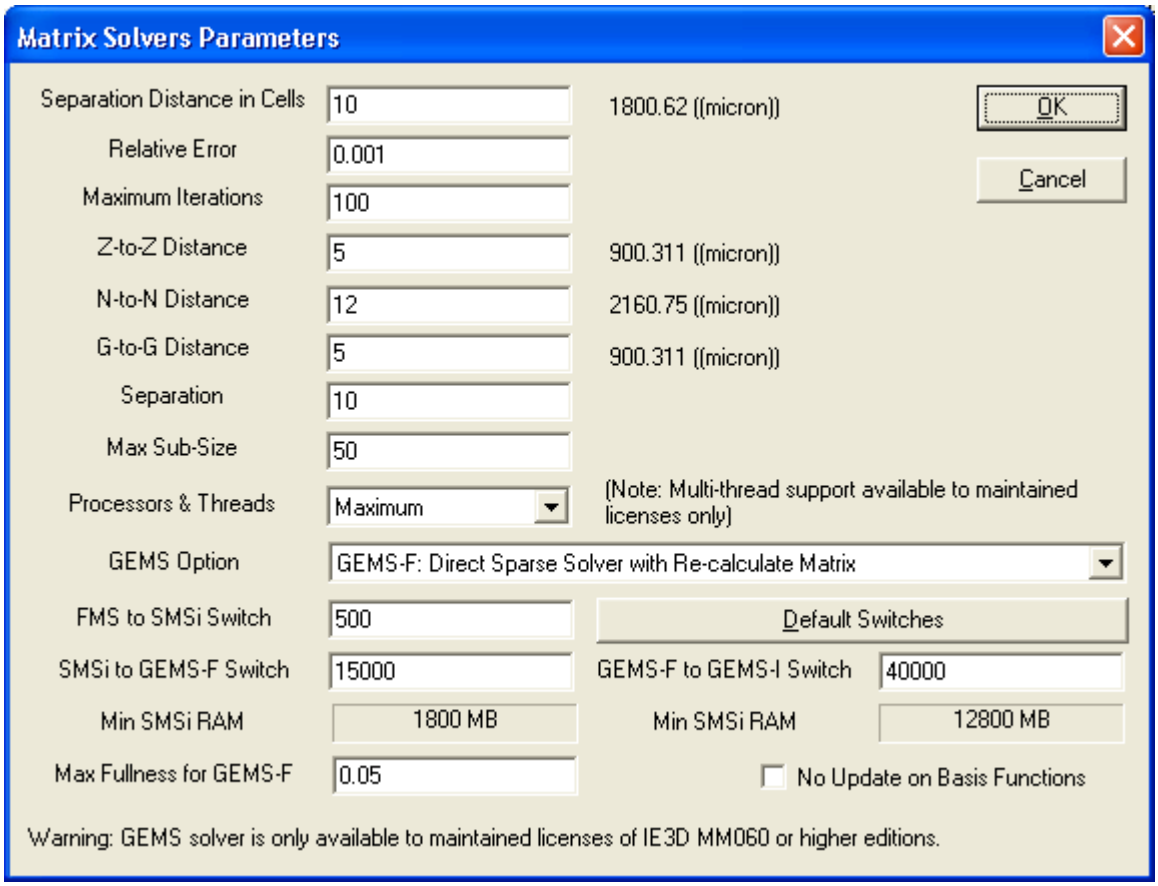


Figure A.4 Matrix Solver Parameters dialog.

8. Default Output Parameters:

The Default Output Parameters option is a newly introduced option for the IE3D 10.0. The purpose of introducing the Default Output Parameters is to make the simulation results in more readable format for post-processing.

As you know, the primary simulation results are the s-parameters in Agilent/EEsof Touchstone format. The Touchstone format is adopted by most microwave and RF simulators. A file in original Touchstone format limits an s-parameter value to 4 digits. It is quite easy to read. However, it introduces very serious accuracy problem, especially at low frequency, due to loss of digits. For this reason, we always keep at least 10 digit accuracy for the IE3D.

Some users want to get the s-parameters in more readable formats. Some users want to get parameters other than the s-parameters. Although we can always get any other parameters on MODUA after

a simulation, it would be nice to let the IE3D engine to output different parameters of interests in a nice tabular format easy to be read by the users. The parameters can be used by users to automate the post-processing.

For the above reason, we introduced an optional output file called Indexed Parameter Array file with extension .IPA. We will refer it as IPA file in the discussion below.

Table A.1 Parameters output in the IPA file.

Parameter	Format	Description
S-Parameters S(i,j)	dB, Mag, Ang Re, Im	The s-parameters are not necessary to be normalized to characteristic impedance Zci = 50 ohms. The users can modify the Zci to other value in the Modify button on the Simulation Setup dialog.
Y-Parameters Y(i,j)	Mag, Ang Re, Im	The Y-parameters are defined in standard textbooks.
Z-Parameters Z(i,j)	Mag, Ang Re, Im	The Z-Parameters are defined in standard textbooks.
Z for S-Parameters Zsp(i,j)	Mag, Ang Re, Im	The Zsp(i,j) is not defined in any textbook. It is defined as: $Zsp(i,j) = \sqrt{(Zci \cdot Zcj)}\,[1 + S(i,j)] / [1 - S(i,j)]$. Please understand that Zsp(1,1) = Z(1,1) for 1-port structure. However, for a general N-port structure, $Zsp(i,j) \neq Z(i,j)$.

The IPA file is nicely formatted. It allows the users to output combinations of the parameters documented in Table A.1.

On the **Parameters->Optional Parameters** dialog, you can configure which groups of parameters will be written into the IPA file as default (see Figure A.5). However, in the Simulation Setup dialog (see Figure 3.16), you can still decide which detail item you want to output. If you select no group, the .IPA file will not be created.

Shown in Figure A.4 is the dialog with 7 groups of parameters selected. However, the number of items written to an IPA file may be different from this number.

9. Default File Format:

The most popular IE3D geometry file formats are IE3D 9.0, IE3D XML and IE3D XML 2.0 file formats. They are discussed in Appendix B. Please read Appendix B for more information.

10. After Setup:

After a simulation setup, you can allow MGRID to invoke the IE3D engine for simulation. You can also allow MGRID to create the .sim file for batch simulation later. For those IE3D licenses without the engines, the users can choose to send jobs to the ZDS so that the simulation job can be handled by other IE3D licenses with engines or by the ZDM distributed simulation engines.

11. Default Edit Mode:

MGRID's default editing mode is Draw Mode. Some users may want it to be configured as Selection mode. You can choose the default here.

12. Default Selection Focus:

On MGRID, when you go into a selection mode, MGRID will set the selection focus to all layers by default. However, you can configure MGRID to focus on the current updated layer or the checked layers when you select the selection mode.

13. Polygon Limit for Checking and Geometry Checking Skipping:

Geometry checking after each editing command may take long time when the number of polygons involved is big. The Polygon Limit for Checking parameter allows users to skip some length checking process when the number of polygons exceeds the limit. The Geometry Checking Skipping allows users to define how frequent MGRID remindS the users of the reduced checking. If Geometry Checking Skipping is 0, it means MGRID will remind users of the reduced geometry checking for every geometry editing command.

14. Connectivity Checking:

IE3D has been able to detect and correct polygon connections automatically for layered polygons when they overlap. However, when 3D polygons are intersecting, earlier versions of IE3D would not do the connection checking and corrections automatically. In IE3D V15, we have implemented the feature to detect and correct 3D polygon connections in Adv Edit->Connection->Adjust Geometry for Connection. By default, we do not take 3D polygon connection checking every command. This correspond to Connectivity Checking as “2D Geometry Adjustment”. In case a user wants MGRID to perform 3D geometry connection checking for every geometry command, the user can change the selection to “3D Geometry Connection”. It should be pointed out that “3D Geometry Connection” does require more time for checking while it helps you to build connectivity between 3D polygons automatically.

15. Accuracy Controls:

Accuracy control parameters are used internally. Please try not to change them unless you are instructed to.

16. Other Parameters:

The parameters in the Other group are used to control how MGRID displays the information of the structure.

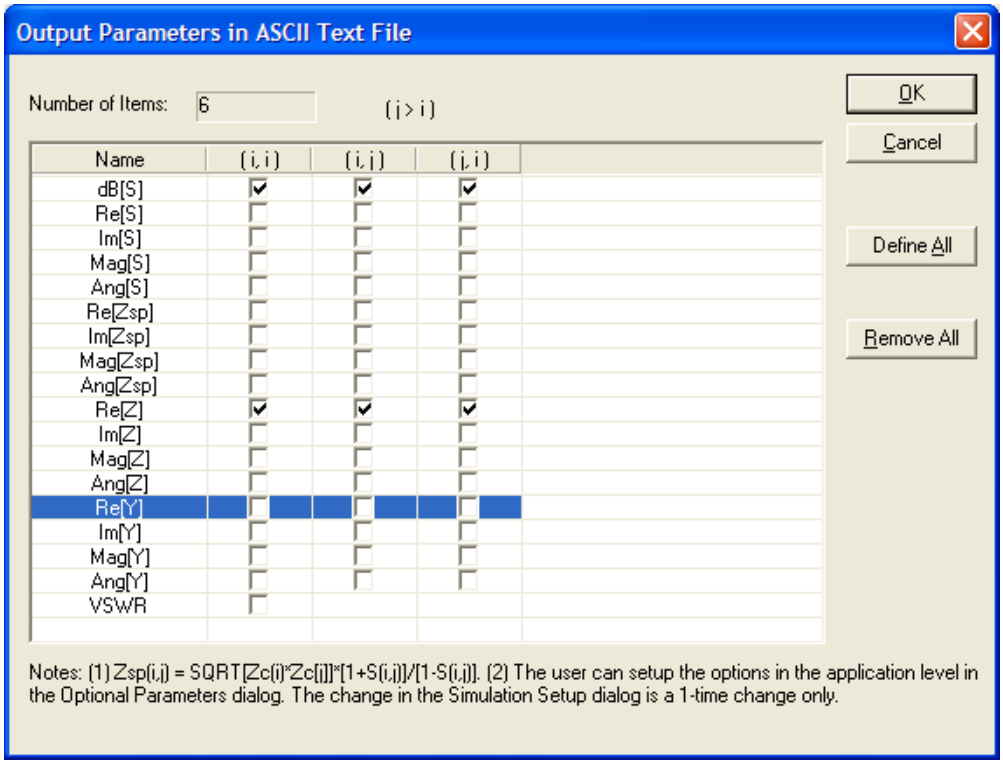


Figure A.5 The Default Output Parameters dialog allowing a user to configure the group of parameters.

Appendix AA. Simulation Input File (.SIM) Format and Using External Optimizers

In an IE3D simulation, the geometry is saved in the .geo file. The input and control parameters are saved in the .sim file. When we setup a simulation, MGRID will save the geometry into .geo file, and the input and control parameters in the .sim file. Then, MGRID will call the IE3D using the .sim file as the command line argument. The .sim file will tell IE3D where to find the .geo file, which frequency points a simulation should do, and where the save the output files. To perform an IE3D simulation, we need both the .sim file and the .geo file.

On the Simulation Setup, we have the option to save the .sim file only. Also, MGRID always saves the .geo file after the simulation setup. Anytime if you want to perform an IE3D simulation, in case you save the .sim file only or terminate a simulation invoked by MGRID, you can invoke the IE3D to perform a simulation by entering:

"C:\MentorGraphics\<latest_release>IE3D\SDD_HOME\IE3D\exe\ie3dpm.exe"
"C:\MentorGraphics\<latest_release>IE3D\SDD_HOME\IE3D\ie3d\samples\ u_3ports.sim "

Where the "C:\MentorGraphics\<latest_release>IE3D\SDD_HOME\IE3D\ie3d\samples\u_3ports.sim" is the .sim file name for the "C:\MentorGraphics\<latest_release>IE3D\SDD_HOME\IE3D\ie3d\samples \u_3ports.geo" which is documented in Appendix B.

1. The Simulation Input File (.SIM) for IE3D 11:

Tables AA-1 and AA-2 document the .sim file in IE3D XML 2.0 file format.

Table AA-1. The Simulation Input File in .\ie3d\samples\u_3ports.sim".

Content	Section
<?xml version="1.0" ?>	1
<ZlsDoc type="SimOptimParameters xml file" version="2.0" copyright=" Mentor Graphics Corp." >	2b
<SimOptimParameters unit="micron" SimType="1" ProcessPriority="4" SaveNetPort="0" WriteOrAppend="0" AppendLog="0"	3b
FileBase=" C:\MentorGraphics\<latest_release>IE3D\SDD_HOME\IE3D\ie3d\practice \u_3ports" AfterSetup="0" >	
<!-- SimOptimParameters base type begin -->	Comment
<MeshingParametersBase ereff="5.450002474999382" fmax="20" ncells="20" warning_limit="4000" aec="0" aec_level="0"	4b
aec_ratio="0.1" multi_aec_ratio="0.4" meshing_optim="1" detect_overlapping="1" meshing_scheme="0" align_meshing="4"	
max_layer_distance="0.0005" cmax_regular="0.3210423766983152" refined_ratio="0.2" rectanglizations="3" merge_polygons="0"	
option_2d="2" option_3d="2" >	
</MeshingParametersBase>	4e
<!-- SimOptimParameters base type end -->	
<PrintOptionStruct id="print_option" print_geosim="1" print_error="1" print_variables="1" print_designsim="1" >	5b
</PrintOptionStruct>	5e
<MatrixSolverParameters Type="5" SeparationDistanceInCells="10" SeparationDistance="3210.423766983152"	6
SD_z2z="1605.211883491576" SD_n2n="4815.635650474727" SD_g2g_far="1605.211883491576" SD_no_update="0"	
Processors="1" AbsoluteEps="0" RelativeEps="0.025" SecondRelativeEpsr="0.001" NumberOfIterations="20" BufferSize="1"	
NumberOfExternFiles="5" DroppingCriteria="0.0004" MinimumUnknowns="1000" AlternativeType="5" />	
<AccuracyControllParameters ptype="3" pformat="1" epsr_2d="3e-005" epsa_2d="3e-005" epsr_3d="0.0001" epsa_3d="0.0001"	7b
max_lambda="1000000" max_factor="2.5" extract_factor="0.2" far_factor="0.5" max_images="100000"	

max_pts="12"	
rate_pts="2.5" ratio_next="1.5" insert_points="3" >	
<ValueList id="AccuracyControls">6,12,5,8,3,5,4</ValueList>	
</AccuracyControllParameters>	7e
<AdaptiveIntellifit Enabled="1" LargeValueEps="0.005" SmallValueEps="0.2" />	8
<AfterSimOptions InvokeModua="1" Keep_Open="0" />	9
<PatternParameters Enabled="0" ImagesAlongX="1" ImagesAlongY="1" EnforceConservation="1" >	10b
<AngleArray id="angle_array" >	11b
<ValueList id="phi_array">0,10,20,30,40,50,60,70,80,90,100,110,120,130,140,150,160,170,180,190,200,	12
210,220,230,240,250,260,270,280,290,300,310,320,330,340,350,360</ValueList>	
<ValueList	13
id="theta_array">0,5,10,15,20,25,30,35,40,45,50,55,60,65,70,75,80,85,90,95,100,105,110,115,120,125,	
130,135,140,145,150,155,160,165,170,175,180</ValueList>	
</AngleArray>	11e
</PatternParameters>	10e
<CoordinateTransforms id="rotations" m_1st_phi="0" m_1st_theta="0" m_2nd_phi="0" xoffset="0" yoffset="0"	14b
zoffset="0" >	
</CoordinateTransforms>	14e
<SecondGround id="second_ground" m_2nd_ground="0" epsr="1" mur="1" sigma="1000" height="1500000" >	15b
</SecondGround>	15e
<ExcitationArray id="excite_array" ports="3" planewave="0" source="1" connect="0" connect_file="" >	16b
<ObjList id="this">	
<Excitation zc="50" re_zi="50" im_zi="0" magnitude="1" phase="0" >	16.1b
</Excitation>	16.1e
<Excitation zc="50" re_zi="50" im_zi="0" magnitude="1" phase="0" >	16.2b
</Excitation>	16.2e
<Excitation zc="50" re_zi="50" im_zi="0" magnitude="1" phase="0" >	16.3b
</Excitation>	16.3e
</ObjList>	
</ExcitationArray>	16e
<CurrentParameters Enabled="0" />	17
<ValueList	18
id="FrequencyParameters">0.5,1,1.5,2,2.5,3,3.5,4,4.5,5,5.5,6,6.5,7,7.5,8,8.5,9,9.5,10,10.5,11,11.5,12,12.5,13,13.5,	
14,14.5,15,15.5,16,16.5,17,17.5,18,18.5,19,19.5,20</ValueList>	
<OptimVariables id="opt" name_convention="1" >	19b
<ObjList id="this">	
</ObjList>	
</OptimVariables>	19e
<IndexedParameterOutput>	20b
<IndexedParameterArray id="ipa" >	
<ObjList id="this">	
</ObjList>	
</IndexedParameterArray>	20e
</IndexedParameterOutput>	3e
</SimOptimParameters>	2e
</ZlsDoc>	

Table AA-2. The explanation on the .sim file in .\ie3d\samples\u_3ports.sim.

Section	Explanation
1	It is required by XML and it indicates this is an XML file.
2b	This the beginning section for Mentor Graphics Document XML file. The version=”2.0” indicates it is the IE3D XML 2.0 file format.
3b	It is the beginning of the SimOptimParameters object. The SimOptimParameters object is

	basically all the data saved in a .sim file. SimType="1" means that it is a structure simulation from MGRID etc. You may encounter cases with the SimType of other values. The situation is documented in Table AA.3.
Comment	The line "<!....>" means it is a comment.
4	This section control the meshing parameters. Again, the same section is in the associated .geo file for a geometry simulation or optimization.
5	This section is used internally. You can omit it.
6	This section saves the matrix solver information. By default, Type="5" indicates it is SMSi matrix solver or the default one. Other parameters are only used by iterative matrix solvers.
7	This section saves the accuracy control parameters. Please do not change them.
8	This section controls the Adaptive Intelli-Fit (AIF). Enabled="1" means it is enabled. Please use the default LargeValueRes and SmallValueRes.
9	This section describes whether you want to invoke MODUA to display s-parameters and whether you want to keep the IE3D dialog open after the simulation.
10	This section describes whether you want IE3D to perform pattern calculation. The ImageAlongX indicates how many mirror images will be included in case it has two parallel walls for the enclosure at constant X (theoretically infinite number of images). The ImageAlongY is for the y-direction. They are not used here.
11b	It indicates the start point for the theta array and phi array for pattern calculation.
12	It saves the phi array for pattern calculation. The phi values must be in ascending order. They can be non-uniform. However, phi=0 must be the 1st value and phi=360 must be the last value.
13	It saves the theta array for the pattern calculation. The theta values must be in ascending order. They can be non-uniform. However, theta=0 must be the 1st value and theta=180 must be the last value.
14	You can apply coordinate transform (rotation and shift) to the pattern. This section is for it.
15	You can define a second ground for the pattern. This is for base station application. Your antenna's ground plane may not be the only ground. You may rotate your antenna and its pattern will be affected by the true ground in the base station. This feature allows you to add the effects of the true ground into the pattern.
16	It describes the excitation and termination array for the ports and the planewave excitation if there is any. The ports="3" means that this structure has total 3 net ports. Please remember this value must be consistent with the total net port number in the .geo file in case you need to calculate pattern. This structure has 4 global ports (+1, -1, 2 and 3) while it has only 3 net ports (+1 and -1 forming the net port 1). The source="1" means we are using constant wave source (see Appendix R). The connect="0" means that it does not have an feed network or .ECT file to feed it. In case it does have one, you need to provide the .ECT file name in connect_file="the_ect_file.ect". The planewave="0" means that there is no plane wave. In case, there is plane wave excitation, you will have planewave="1" and the last two excitation sections in this section will be for the E-plane and H-plane excitations of the plane wave.
16.1	It describes the excitation and termination for the No.1 port. The zc="50" means that the normalization impedance is 50-ohms. You can change it to other positive value and it is effective. The source impedance is re_zi="50{ and im_zi="0". The magnitude="1" and phase="" mean the constant wave source is 1 v with 0 degree phase.
16.2	It describes the excitation and termination for the No.2 port.
16.3	It describes the excitation and termination for the No.3 port.
17	It saves whether you want to save the .cur file for the current distribution visualization. When pattern calculation is enabled, the .cur file option must be enabled.
18	It saves the series of frequency points you want to simulate the structure. In case AIF is enabled, it will simulate it at selected frequency points while it will yield accurate results at all the frequency points.
19	It saves the Optimization Variables with the number of offset values and ranges of the variables when EM Tuning is performed. Please make sure the number of Optimization Variables matches the one in the .geo file for EM Tuning. Also, please make sure the ranges of the variables will not be out of bound defined in the .geo file for each variable. The above

	statements are also true for Design Tuning procedure. In case it is an EM simulation with Optimization Variables defined, the Start value of each Optimization Variable in the range will be used to control the shape of the geometry for simulation.
20	This section saves which parameters you want to be included in the ASCII output file (.ipa) after a simulation. You can choose some selected items for streamlined simulation output.

Table AA-3. The different values of "SimType" and their corresponding procedures.

SimType	Procedure	Explanation
1	Geometry Simulation	It instructs IE3D to perform an EM simulation described in the .sim file and the associated .geo file.
2	Design Simulation	It instructs IE3D to perform a mixed EM and circuit simulation described in the .sim file and the associated .dsg file which is created by MODUA. A .dsg file may contain one or more structures (.geo files) as well as some lumped elements as R, L, C and K, and s-parameters files.
3	Geometry Tuning	Perform a series of EM simulation on the same .geo file with different shapes controlled by the offset values to the Optimization Variables. The range and number of offset values for each Optimization Variable are saved in the .sim file.
4	Design Tuning	Perform a series of mixed EM and circuit simulation on the same .dsg file with different offset values to the Optimization Variables. The range and number of offset values for each Optimization Variable are saved in the .sim file. The Optimization Variables can be from the .geo files and/or R, L and C-elements.
11	Geometry Optimization	It instructs IE3D to perform an EM optimization described in the .sim file and the associated .geo file.
12	Design Optimization	It instructs IE3D to perform a mixed EM and circuit optimization described in the .sim file and the associated .dsg file.

2. The Simulation Input File (.SIM) in ASCII format:

Table AA-4 and AA-5 document the .sim file ASCII format. The example file is saved in .\ie3d\samples\u_3ports_9.sim. It saves the same data as the u_3ports.sim file. The .sim file ASCII format is the one before we introduced the XML file format into IE3D 10.1. Some new features such as Contemporary Meshing and Multi-Layer AEC will not be available using the ASCII .sim file format because the corresponding variables can not be saved in the old format.

Table AA-4. The ASCII .sim file in .\ie3d\samples\u_3ports_9.sim.

Content	Section
8 1	1
1 1 1 1	2
3 3 0 1	3
5 10 3.21050453291106e+3 0.0 2.5e-2 1.0e-3 20 1.0 5 4.0e-4 1000 5	4
3 1 2 2 1 4	5
6 5 3 12 8 5 4	6
1 5.0e-3 2.0e-1	7
1 1 1	8
0 0 0 0	9
C:\MentorGraphics\<latest_release>IE3D\SDD_HOME\IE3D\ie3d\samples\u_3ports_9.geo	10

C:\MentorGraphics\<latest_release>IE3D\SDD_HOME\IE3D\ie3d\samples\u_3ports_9.sp	11
C:\MentorGraphics\<latest_release>IE3D\SDD_HOME\IE3D\ie3d\samples\u_3ports_9.spt	12
C:\MentorGraphics\<latest_release>IE3D\SDD_HOME\IE3D\ie3d\samples\u_3ports_9.log	13
C:\MentorGraphics\<latest_release>IE3D\SDD_HOME\IE3D\ie3d\samples\u_3ports_9.cur	14
C:\MentorGraphics\<latest_release>IE3D\SDD_HOME\IE3D\ie3d\samples\u_3ports_9.pat	15
C:\MentorGraphics\<latest_release>IE3D\SDD_HOME\IE3D\ie3d\samples\u_3ports_9.sts	16
40	17
5.0e-001 1.0 1.5 2.0 2.5 3.0 3.5 4.0 4.5 5.0	18
5.5 6.0 6.5 7.0 7.5 8.0 8.5 9.0 9.5 1.0e+1	
1.05e+1 1.1e+1 1.15e+1 1.2e+1 1.25e+1 1.3e+1 1.35e+1 1.4e+1 1.45e+1 1.5e+1	
1.55e+1 1.6e+1 1.65e+1 1.7e+1 1.75e+1 1.8e+1 1.85e+1 1.9e+1 1.95e+1 2.0e+1	
0	19
0 0	20
0.0 0.0 0.0 0.0 0.0 0.0	21
0 1.0 1.0 1.0e+3 1.5e+6	22
0	23
1 5.0e+1 5.0e+1 0.0 1.0 0.0	24
2 5.0e+1 5.0e+1 0.0 1.0 0.0	
3 5.0e+1 5.0e+1 0.0 1.0 0.0	

Table AA-5 The description of the sections in .\ie3d\samples\u_3ports_9.sim

Section No	Description
1	The "8" means that the format is in version 8 of sim file format. The "1" means that it is a simulation .sim file. For optimization, it is 11.
2	Please set this line as "1 1 1 1".
3	The 1st number "3" means that the geometry file uses micron as length unit ("1" means mm and "2" means mil, etc. It is in the same order as you see in the Basic Parameters). The 2nd number "3" is the number of net ports of the structure. The 3rd number "0" means there is no plane-wave excitation. The 4th number "1" means that it is using the wave source ("2" for voltage source and "3" for current source). The source type is only used for pattern calculation and optimization
4	The 1st number "5" means using SMSi, the default matrix solver. The 2nd number "10" is the SD in Cells. The 3rd number "3.21050453291106e+3" is the SD in microns. The 2nd and 3rd numbers and the rest 10 numbers on the line are not useful for SMSi and other direct solvers.
5	Please leave the "3 1 2 2" as they are. The 5th number "1" means that the IE3D will invoke MODUA after simulation ("0" means no invoking). The last number "4" is for the process priority.
6	Please leave the numbers as they are.
7	The 1st number "1" means AIF is enabled. ("0" means disabled). Please leave the "5e-3 2.0e-1" as they are.
8	The 3 numbers are for pattern calculation. Please leave them as they are.
9	Please leave the 1st two "0"s as they are. The 3rd "0" means that the .cur file is not saved. The 4th "0" means that the .pat file is not saved.
10	The geometry file name if it is using the IE3D 10.2 and earlier versions. Starting from IE3D 11, the file names in the whole file are not critical. IE3D will capture the .sim file name when it is invoked with the .sim file name as command line argument. It will automatically derive all the related file names from the .sim file name.
11	The s-parameter file name.
12	The original s-parameter file name when AIF is enabled.
13	The log file name for intermediate information
14	The .cur file name for current distribution.

15	The .pat file name for radiation pattern.
16	The status file for the simulation. It may not be used in version 10.0.
17	The “40” means there are 40 frequency points.
18	There are 40 floating point values for the 40 frequency points.
19	The “0’ means there is no optimization variable in the geometry file.
20	Please leave the line as it is.
21	Please leave the line as it is.
22	Please leave the line as it is.
23	Please leave the line as it is.
24	The No.1 port’s excitation and termination information. The 1st number “1” is for port 1. The 2nd number “50” is for the Zc. The 3rd number “50” and the 4th number “0” are for the real and imaginary parts of the terminating impedance. The 5th number “1” is the magnitude of the excitation. The 6th number “0” is for the phase of the excitation in degree.

Table AA-6. The Simulation Input File in .\ie3d\sample\rpatch2m.sim.

Content	Section
<?xml version="1.0" ?> <ZlsDoc type="SimOptimParameters xml file" version="2.0" copyright="Mentor Graphics Corp." > <SimOptimParameters unit="mil" SimType="1" ProcessPriority="4" SaveNetPort="0" WriteOrAppend="0" AppendLog="0" FileBase=" C:\MentorGraphics\<latest_release>IE3D\SDD_HOME\IE3D\ie3d\practice \rpatch2m" AfterSetup="2" > <!-- SimOptimParameters base type begin --> <MeshingParametersBase ereff="2.7" fmax="2" ncells="15" warning_limit="4000" aec="1" aec_level="0" aec_ratio="0.05" multi_aec_ratio="0.4" meshing_optim="0" detect_overlapping="1" meshing_scheme="1" align_meshing="4" max_layer_distance="0.0005" cmax_regular="6.081595993204448" refined_ratio="0.2" rectanglizations="3" merge_polygons="0" option_2d="2" option_3d="2" > </MeshingParametersBase> <!-- SimOptimParameters base type end --> <PrintOptionStruct id="print_option" print_geosim="1" print_error="1" print_variables="1" print_designsim="1" > </PrintOptionStruct> <MatrixSolverParameters Type="5" SeparationDistanceInCells="10" SeparationDistance="2394.329131182853" SD_z2z="1197.164565591427" SD_n2n="3591.49369677428" SD_g2g_far="1197.164565591427" SD_no_update="0" Processors="1" AbsoluteEps="0" RelativeEps="0.025" SecondRelativeEpsr="0.001" NumberOfIterations="20" BufferSize="1" NumberOfExternFiles="5" DroppingCriteria="0.0004" MinimumUnknowns="1000" AlternativeType="5" /> <AccuracyControllParameters ptype="3" pformat="1" epsr_2d="3e-005" epsa_2d="3e-005" epsr_3d="0.0001" epsa_3d="0.0001" max_lambda="1000000" max_factor="2.5" extract_factor="0.2" far_factor="0.5" max_images="100000" max_pts="12" rate_pts="2.5" ratio_next="1.5" insert_points="3" > <ValueList id="AccuracyControls">6,12,5,8,3,5,4</ValueList> </AccuracyControllParameters> <AdaptiveIntellifit Enabled="1" LargeValueEps="0.005" SmallValueEps="0.2" /> <AfterSimOptions InvokeModua="0" Keep_Open="0" /> <PatternParameters Enabled="0" ImagesAlongX="1" ImagesAlongY="1" EnforceConservation="1" > <AngleArray id="angle_array" > <ValueList id="phi_array">0,10,20,29.999999999999996,40,50,59.999999999999993,70,80,90,100,109.99999999999999,119.99999999999999,130,140,149.99999999999997,160,170,180,190,200,210.00000000000003,219.99999999999997,230,239.99999999999997,250.00000000000003,260,270,280,290,299.99999999999994,310,320,330.00000000000006,340,350,360</ValueList> <ValueList id="theta_array">0,5,10,14.999999999999998,20,25,29.999999999999996,35,40,45,50,54.999999999999993,59.999999999999993,65,70,74.999999999999986,80,85,90,95,100,105.00000000000001,109.99999999999999,115	

,119.99999999999999,125.00000000000001,130,135,140,145,149.99999999999997,155,160,165.000000000000 03,170,175,180</ValueList> </AngleArray> </PatternParameters> <CoordinateTransforms id="rotations" m_1st_phi="0" m_1st_theta="0" m_2nd_phi="0" xoffset="0" yoffset="0" zoffset="0" > </CoordinateTransforms> <SecondGround id="second_ground" m_2nd_ground="0" epsr="1" mur="1" sigma="1000" height="59055.11811023622" > </SecondGround> <ExcitationArray id="excite_array" ports="1" planewave="0" source="1" connect="0" connect_file="" > <ObjList id="this"> <Excitation zc="50" re_zi="50" im_zi="0" magnitude="1" phase="0" > </Excitation> </ObjList> </ExcitationArray> <CurrentParameters Enabled="0" /> <ValueList id="FrequencyParameters">1.8,1.802,1.804,1.806,1.8080000000000001,1.8100000000000001,1.8120000000000 001,1.8140000000000001,1.8160000000000001,1.8180000000000001,1.8200000000000001,1.8220000000000 01,1.8240000000000001,1.8260000000000001,1.8280000000000001,1.8300000000000001,1.832000000000000 1,1.8340000000000001,1.8360000000000001,1.8380000000000001,1.8400000000000001,1.8420000000000001 ,1.8440000000000001,1.8460000000000001,1.8480000000000001,1.8500000000000001,1.8520000000000001, 1.8540000000000001,1.8560000000000001,1.8580000000000001,1.8600000000000001,1.8620000000000001,1 .8640000000000001,1.8660000000000001,1.8680000000000001,1.8700000000000001,1.8720000000000001,1. 8740000000000001,1.8760000000000001,1.8780000000000001,1.8800000000000001,1.8820000000000001,1.8 840000000000001,1.8860000000000001,1.8880000000000001,1.8900000000000001,1.8920000000000001,1.89 40000000000001,1.8960000000000001,1.8980000000000001,1.9000000000000001,1.9020000000000001,1.904 0000000000001,1.9060000000000001,1.9080000000000001,1.9100000000000001,1.9120000000000001,1.9140 000000000001,1.9160000000000001,1.9180000000000001,1.9199999999999999,1.9220000000000002,1.92399 99999999999,1.9260000000000002,1.9279999999999999,1.9300000000000002,1.9319999999999999,1.934000 0000000002,1.9359999999999999,1.9380000000000002,1.9399999999999999,1.9420000000000002,1.944,1.94 6,1.948,1.95,1.952,1.954,1.956,1.958,1.96,1.962,1.964,1.966,1.968,1.97,1.972,1.974,1.976,1.978,1.98,1.982,1.98 4,1.986,1.988,1.99,1.992,1.994,1.996,1.998,2</ValueList>	19b
<OptimVariables id="opt" name_convention="1" > <ObjList id="this">	19.1b
<OptimVariable comment="" End="0" HighBound="112.2306035532842" LowBound="-87.76939644671582" Noffset="1" Start="0" StepDeriv="0" >	19.1e
</OptimVariable>	19.2b
<OptimVariable comment="" End="0" HighBound="367.861287009065" LowBound="-32.13871299093501" Noffset="1" Start="0" StepDeriv="0" >	19.2e
</OptimVariable> </ObjList>	19e
</OptimVariables> <IndexedParameterOutput> <IndexedParameterArray id="ipa" > <ObjList id="this"> </ObjList> </IndexedParameterArray> </IndexedParameterOutput> </SimOptimParameters> </ZlsDoc>	

3. Using External Optimizers:

IE3D has implemented excellent optimizers. However, some users may want to use their own optimizers to drive the IE3D engine. It can be done in the following procedures.

Let's take the .\ie3d\samples\rpatch2m.geo file discussed in Chapter 5 for EM tuning as an example. Please open it on MGRID. MGRID will prompt you that optimization variables exist and whether you want to remove them. Please select No to proceed. Please select Process->Simulate. Enter total 101 frequency points from 1.8 to 2.0 GHz. Please don't define any more points for the Tuning Setting. Please select "Create .sim File Only" in After Setup. Select OK and MGRID will save the setting data into .\ie3d\samples\rpatch2m.sim file which is documented in Table AA-6.

Please pay attention to the Section 19 in Table AA-6. It defines the setting of the optimization variables. The Section 19.1 is for the No.1 variable and the Section 19.2 is for the No.2 variable. You can invoke IE3DOS.EXE to simulate this structure:

C:>"\MentorGraphics\<latest_release>IE3D\SDD_HOME\IE3D\exe\ie3ddos"
"\MentorGraphics\<latest_release>IE3D\SDD_HOME\IE3D\ie3d\samples \rpatch2m.sim"

When IE3DOS.EXE is invoked using the above command on Command Prompt, IE3D will simulate the .\ie3d\samples\rpatch2m.geo at the frequency points specified in the .sim file. It will also use the Start="0" in Section 19.1b for the No.1 variable and the Start="0" in Section 19.2b for the No.2 variable. If you can use your own optimizer to change the Start="…" values in the Section 19.1b and Section 19.2b in the .sim file and call the IE3D engine using the above command, and parse the IE3D simulation results (.sp, .cur and .pat) files inside your optimizer, you will be able to use the IE3D engine to drive your own optimizers.

Appendix AB. Physical Component Compiler Library (PCCL)

PCCL is part of the IE3D-SI license. It is a parameterized component library enabling automatic IE3D geometry generations for some commonly used components in PCB layout. High speed PCB goes beyond GHz range. At high frequency, the discontinuities and coupling effects need to be modeled precisely in order to design high quality PCB. Creating an IE3D model on a typical PCB component such as via pair based upon the physical dimensions can be a challenging job. PCCL enables PCB designers the capability to generate the correct IE3D EM model automatically.

At the time of this writing, PCCL is implemented into the IE3D EM Design System (MGRID). However, only IE3D-SI users can use the functionality.

One of the most important components on PCB is differential via pair (see Figure AB.1). We will demonstrate how we use PCCL to create a differential via model in the following procedure.

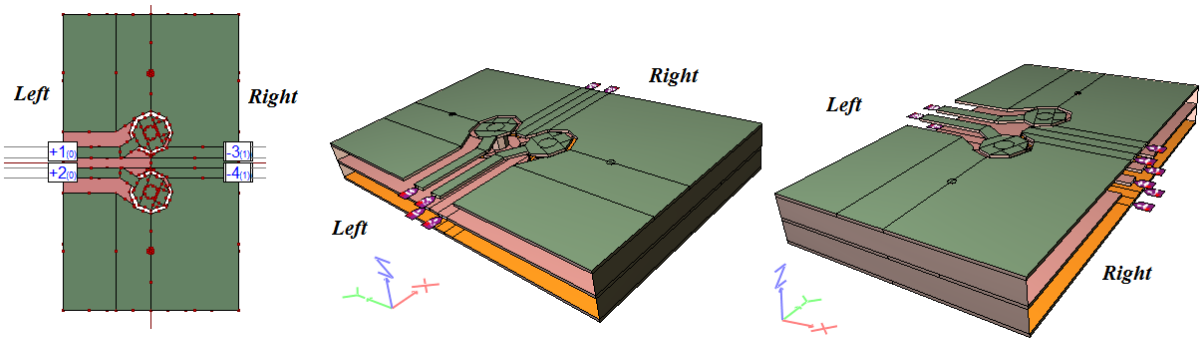


Figure AB.1

Step 1 Run MGRID. Select File->IE3D-SI Physical Component Compiler Library. IE3D-SI Physical Component Compiler Library window will come up (see Figure AB.2). Click on 'Add' button to see 'Select a PCB Element Type' window (see Figure AB.3). Select "Differential Via Pair" type in this window and click OK.

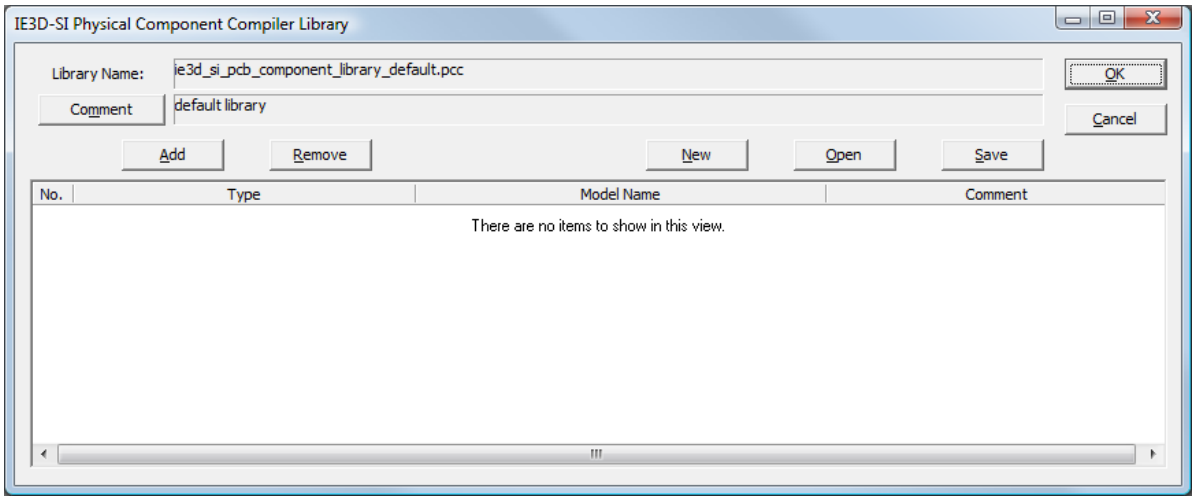


Figure AB.2

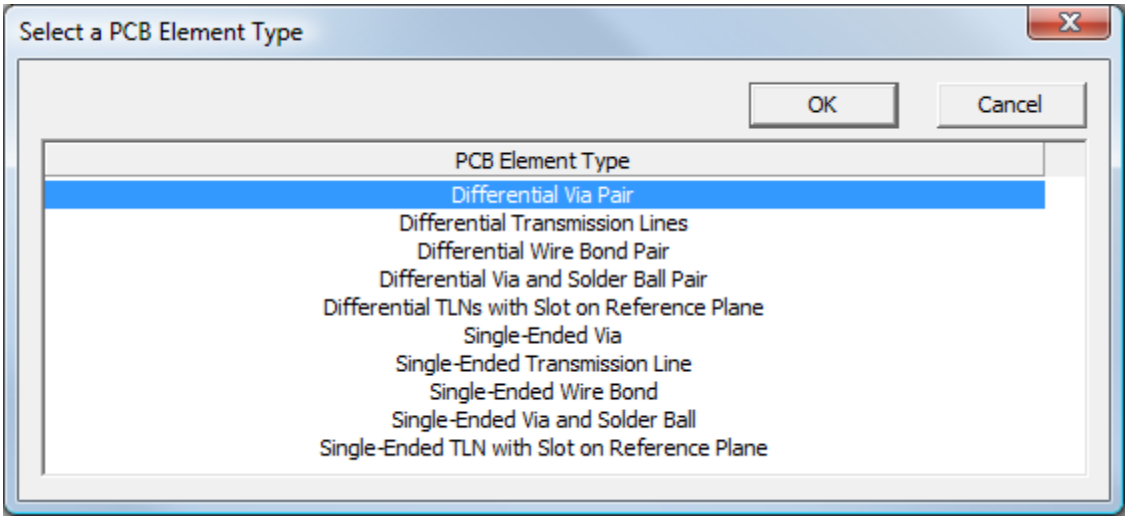


Figure AB.3

The IE3D-SI Differential Via Modeler comes up (see Figure AB.4) with default parameters. It is a differential via pair structure with 3-metal layers. You can easily add or remove layer by selecting Add Layer button or Remove Layer button on the dialog in case you have different stackup.

It is a model for 4-single ended ports. Port 1 and port 2 are on the left of metal layer M1. Port 3 and port 4 are on the right of metal layer M2. Each port has one or two negative ports on M1, M2 and/or M3 as references. The properties of port 1, port 2 and the feed lines on the left are defined in First Diff Port. The properties of port 3, port 4 and the feed lines on the right are defined in Second Diff Port. You have the option to build 2-differential port model automatically by select "2-Differential Ports (Extension Ports without Additional Reference)" option in Ports field (see Figure AB.4). For such a case, each side of the structure is a pair of differential port. The final s-parameters are the 2-port differential s-parameters.

Other parameters such as Via Radius, Via Pad Radius can be changed in the Structure Parameters section.

IE3D simulation related parameters such as Meshing Frequency, Cells/Wavelength, and simulation frequencies are defined in IE3D Simulation Parameters dialog.

You can change the Model Name as you want. The default name is: ie3d_si_differential_vias_pair.geo.

Step 2 Click at Build Model button. PCCL will create an IE3D geometry and save it into file ie3d_si_differential_vias_pair.geo. The file will be opened on another instance of MGRID. You will get the top view and 3D view similar to Figure AB.1. You can view, modify and simulate the structure on the new MGRID instance.

Step 3 If you don't need to check the IE3D geometry file, you can select Simulate button on PCCL to simulate the structure with the default parameters. For the 3-metal layer differential via pair structure, IE3D engine can finish the simulation sweeping from 0.01 GHz to 20 GHz in a few minutes on a high performance computer. The s-parameters will be saved into ie3d_si_differential_vias_pair.s4p file.

As you can see, using PCCL to create a differential vias pair is quite straight-forward. You just enter the parameters and PCCL will create a 3D model and simulate it for you.

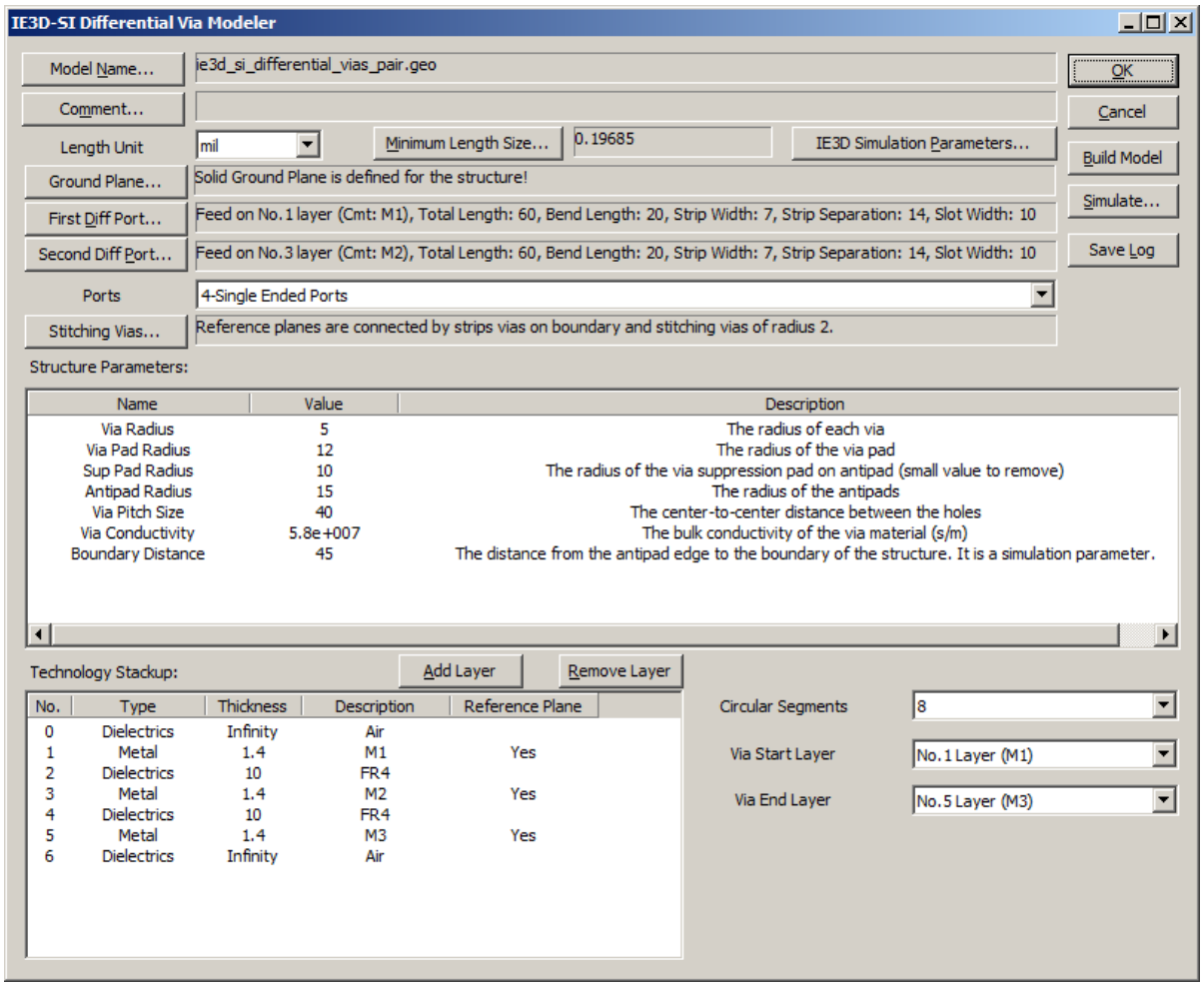


Figure AB.4 The differential vias pair dialog on PCCL.

Appendix AC. Extracting L and Q-Values from S-Parameters

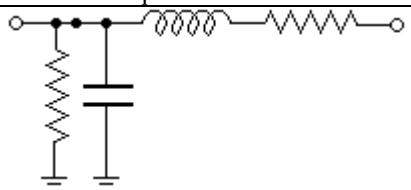
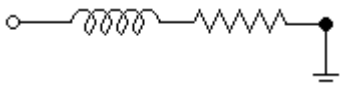
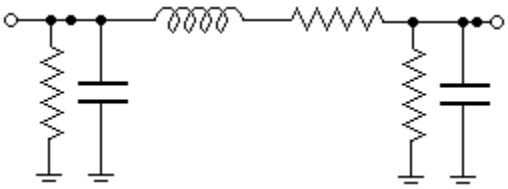
On the IE3D, we have implemented multiple ways to extract the RLC equivalent circuit from the s-parameters. A few of the schemes allow us to extract the frequency dependent RLC equivalent circuits and Q-values from the s-parameters. The L-values and Q-values are extremely important for RFIC modeling.

At low frequency, the L and Q values are kind of quite consistent. It may not be very consistent at high frequency and it is especially true for the Q-values. Basically, the circuit is represented as s-parameters. A lumped element equivalent circuit is only an approximation to the s-parameters. Depending upon how we define the lumped element equivalent circuit, the extracted L and especially the Q can be different at high frequency. In this appendix, we will explain how we extract the L and Q-values and explore how they differ.

Shown in Table AC.1 are the different schemes on the MODUA of the IE3D for extraction of RLC-equivalent circuit and the Q-values from a 2-port or 1-port s-parameter file.

Basically, the 3 models yield the close L and Q values at low frequency where the C's are no longer important. However, at high frequency, the C's are important. Fitting them to different models will yield different L-values and Q-values. The LC-Equivalent and the Pi-Network should yield same Q-values but not the L-values at high frequency.

Table AC.1 Different ways of extracting RLC and Q-values from s-parameters on MODUA.

Command	Ports	Equivalent Circuit	Description
LC-Equivalent	2		The command is good for any coupled N-TLN or interconnect problems. However, if you apply it to the single trace case with 2-ports, it will yield the Q value as: $Q = - Im(Y11) / Re(Y11)$
1-Port Equiv.	1		This command is good for multiple models. If the user choose series R and L, MODUA will find the L and R values. Then, it will calculate the Q as: $Q = \omega L / R$
Pi-Network	2		The command is good for multiple models. If the user chooses series R and L, and shunt R and C, MODUA will yield the Q value as: $Q = - Im(Y11) / Re(Y11)$

Appendix AD. Frequency Dependent Substrates and Metallic Types

Starting from the IE3D 9.2, we allow you to define frequency dependent substrates and metallic strip types. To define a substrate with frequency dependency is quite simple. Figure AD.1 shows an example with frequency dependent substrate. Basically, it defines a substrate with the Epsr = 2.5 at 0 GHz, 2.4 at 2 GHz, 2.0 at 6 GHz, and 1.8 at 10 GHz and beyond. For other frequency between any of the 2 points, we will use linear interpolation. For example, the Epsr at 0.5 GHz is (1.5×2.5+0.5×2.4)/2 = 2.475. We can define frequency dependent metallic strip types in the same way too.

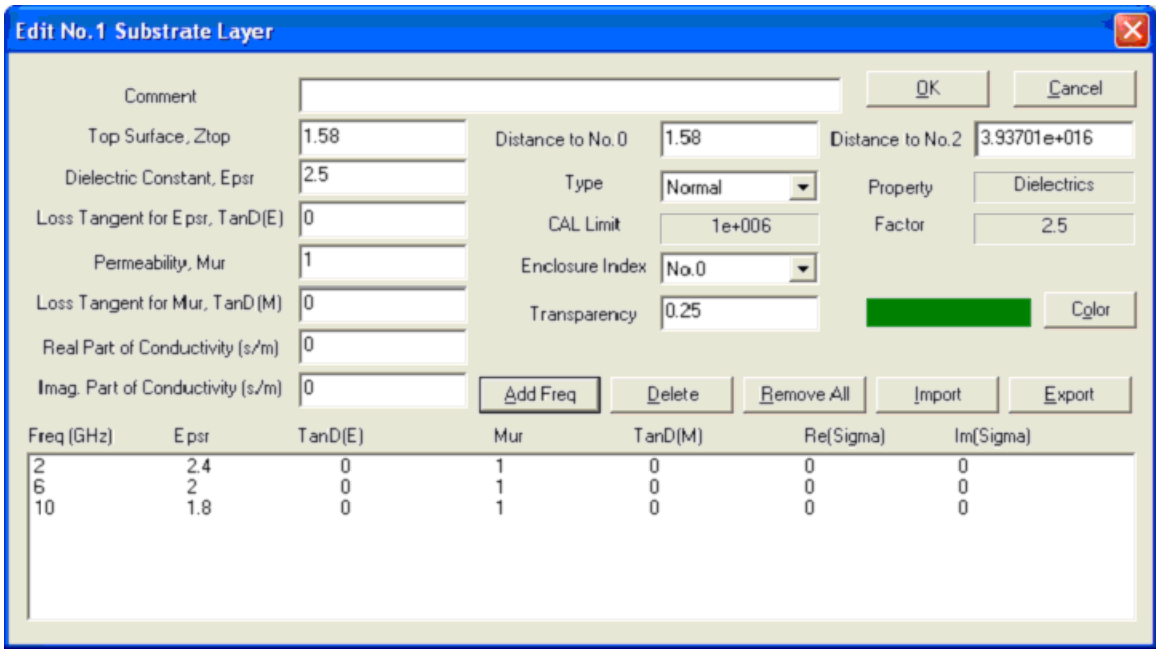


Figure AD.1 The Edit Substrate dialog with frequency dependent substrate.

Appendix AE. Meshing Alignment for Tightly Coupled Structures

It is noted in Chapter 5 that meshing alignment and refined meshing are extremely critical to accurate modeling of tightly coupled structures such as MIM capacitors. Meshing alignment is also necessarily needed for metallic structures on finite dielectrics such as patch antennas on finite substrates. We will need to align the meshing of the patches and the finite substrates even though we do not need to use refined cells. For patch antennas with infinite substrates, we may also need meshing alignment when the substrate is thin and the ground plane is finite. Finite ground plane is modeled as polygons. When the patch is too close to the finite ground, we will need to align the meshing between the patch and the ground for accurate modeling of the resonant frequency. In such a case, it is a high Q-resonance. Slight shift in resonant frequency due to mis-aligned meshing may cause obvious shift in the predicted resonant frequency.

In the IE3D 11, we have implemented the Automatic Meshing Alignment and Refinement. It automatically detects the tightly coupled polygons and finite substrate and perform necessary meshing alignment and refinement at the simulation time. Such a feature greatly reduces the pre-processing time for modeling such a structure and makes it possible for EM optimization of such structures.

However, it is still possible you may occasionally use manual meshing alignment for some special structures. Also, the Automatic Meshing Alignment scheme is relatively new. Even though we have tested it much, however it is a very complicated procedure and there are numerous possibilities. It is still possible it may fail for some special cases in the next couple years after the release of IE3D 11. We would like to document manual meshing alignment in this appendix in case you may need it. We will take the file .\ie3d\practice\viacap2.geo created in Section 12 of Chapter 6 as our starting point. We have created the bridges and reduced vertices on polygon 3 in Figure AE.1 representing the shape of the via. We are going to build the via from Z = 0 to Z = 100 to connect to the bottom plate (polygon 1). We need to alignment the meshing between the bottom plate (polygon 1) at Z = 100 and top plate (polygon 2) at Z = 100.2.

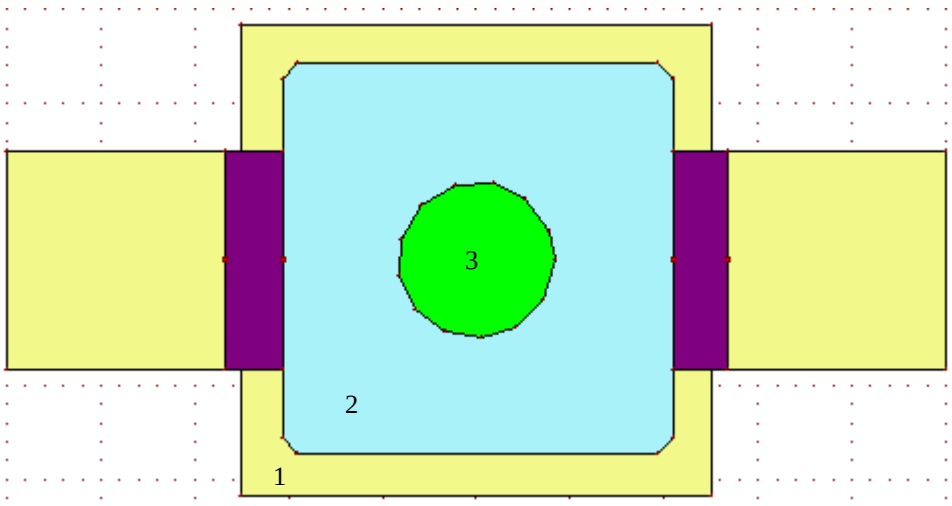


Figure AE.1 The .\ie3d\practice\viacap2.geo from Chapter 12.

Step 1 Select Edit->Select Polygon. Click at No.3 Layer at Z = 100.2 to focus on the layer. Click at polygon 2 to select it. We are going to use it to build a hole at Z = 100.

Step 2 Select Adv Edit->Build Holes and Vias from Selected Polygons command. MGRID prompts you for the parameters. Please select Z = 100 and select Add button to add it into the list. Remember to check Clear The Hole and Keep Selected (see Figure AE.2). Select OK. MGRID will use the polygon at Z = 100.2 to build a hole at Z = 100 while it still keeps the polygon at Z = 100.2.

Step 3 Select Edit->Select Polygon. Click at No.5 Layer at Z = 150 to focus on the layer.. Click at polygon 3 to select it. We are going to use it to build the division lines on Z = 100.2 for aligned meshing.

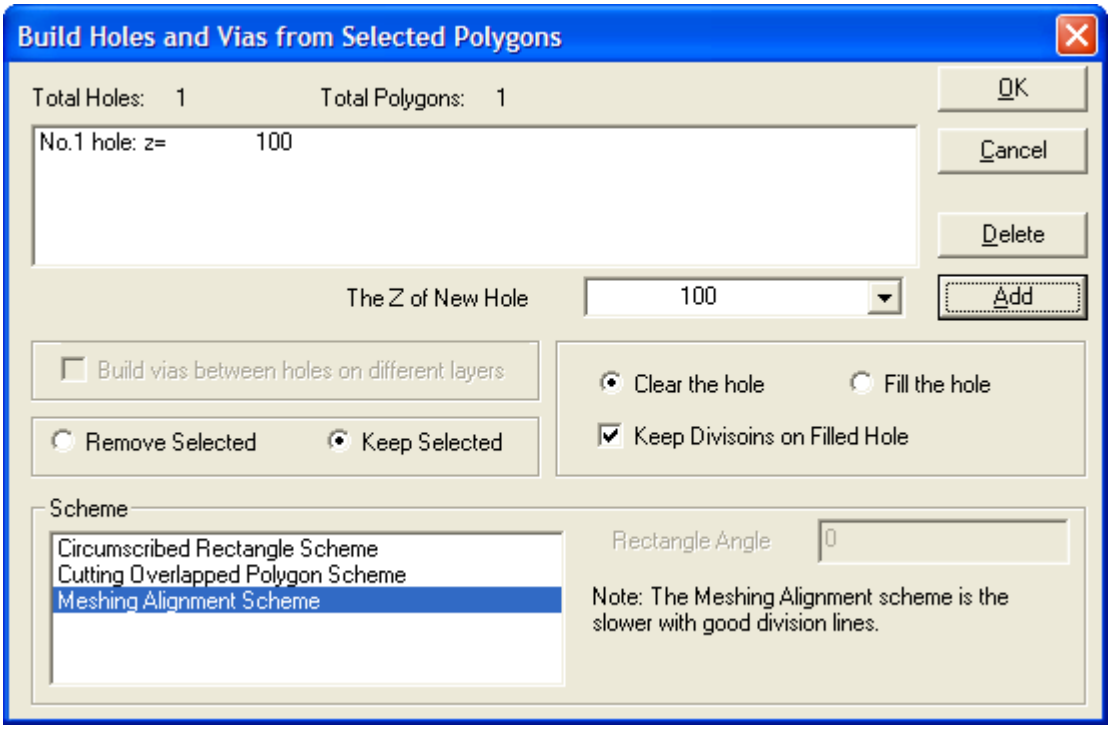


Figure AE.2 The dialog when you are ready to select OK.

Step 4 Select Adv Edit->Build Holes and Vias from Selected Polygons command. MGRID prompts you for the parameters. Please select Z = 100.2 and select Add button to add it into the list. This time, remember to check Fill The Hole and Keep Selected. Select OK. MGRID will duplicate the shape of the polygon on Z = 150 at the polygon on Z = 100.2 while it still keeps the polygon on Z = 150. Please save this file as: .\ie3d\practice\viacap4.geo. We are going to fine meshing the plate at Z = 100.2. Before we do it, we would like to save the change first. In case the refined meshing is not good enough or we need to change it, we can still open the saved file.

Step 5 Select Edit->Select Polygon. Click at No.5 Layer at Z = 150 to focus on the layer.. Click at polygon 3 to select it. We are going to use it to build the via from Z = 0 to Z = 100. Select Adv Edit->Build Holes and Vias from Selected Polygons command. Select Z = 0 and select Add button to add it into the list. Select Z = 100 and select Add button to add the 2nd layer into the list. Check Remove Selected and Clear The Hole. Check Build Vias Between Holes on Different Layers. Select OK to continue. MGRID will build the via from Z = 0 to Z = 100. There is no polygon at the location of the via on Z = 100. It is disconnected there right now.

Step 5 Select Edit->Select Polygon Group. Click at No.3 Layer at Z = 100.2. Window the top plate to select the polygons. Select Adv Edit->Mesh Selected Polygons. MGRID will prompt you for the parameters (see Figure AE.3). Change the Mesh Size to 20. It is equivalent to approximately 142.665 cells per wavelength at 40 GHz. Select OK. MGRID will mesh the top plate into small cells (see Figure AE.4).

Step 6 Select Edit->Select Polygon Group. Click at No.3 Layer at Z = 100.2. Window the meshed top plate to select all the cells. Select Edit->Copy command. Select Edit->Paste command. Click anywhere on the window. Enter X-Offset = 0, Y-Offset = 0 and Z-Offset = -0.2. Select OK. MGRID will duplicate the shape of the top plate onto the bottom plate. It will guarantee the electrical connection between the bottom plate and the via because they are aligned. MGRID may complain about overlapped polygons.

You can select Clean them or even ignore it. It is due to the sensitive detection program. Define the two ports on it. Save the file as: .\ie3d\practice\viacap5.geo. We have the top plate and bottom plate displayed in Figure AE.5. You can see they are meshed into small cells and their cells are aligned. You can also see the shape of the via from the division lines. You can simulate it with Meshing Alignment disabled. You will get similar results as the one in .\ie3d\practice\viacap3.geo using automatic alignment.

Manual alignment and refinement need to follow a special procedure. It is powerful. However, you can not swap the orders of the commands. Normally, you should not use it unless for special purposes.

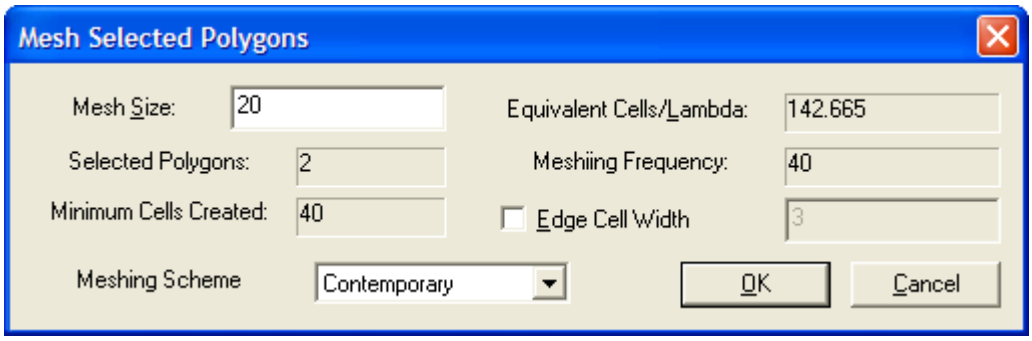


Figure AE.3 the Mesh Selected Polygons dialog.

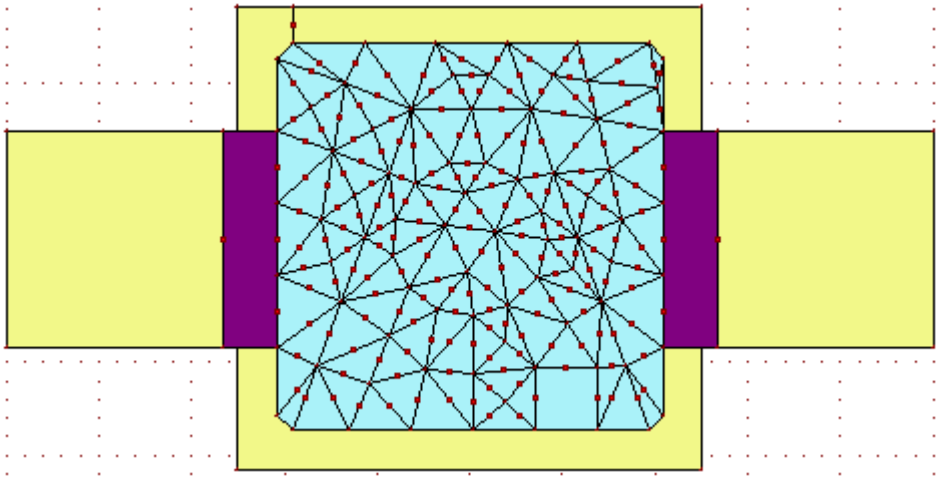
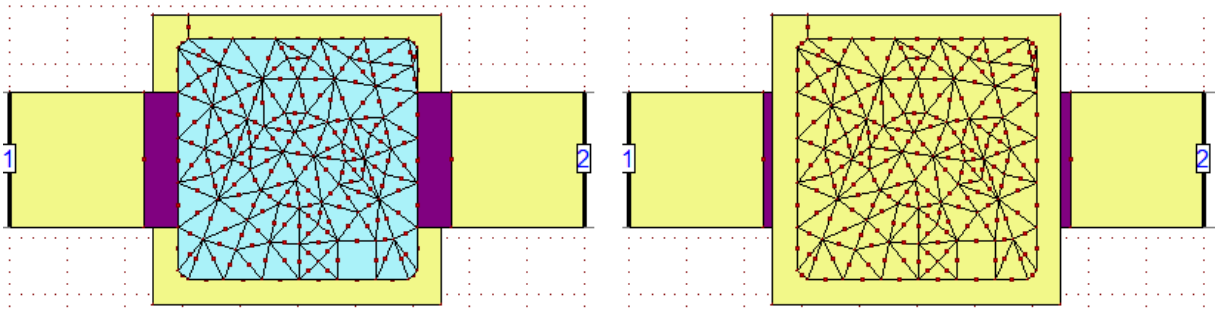


Figure AE.4 The meshed top plate in the structure.



(a) The meshed top plate (b) The meshed bottom plate

Figure AE.5 The manually aligned and meshed structure in .\ie3d\practice\vicap5.geo.

Appendix AF Near Field Calculation and Visualization

One of the major features of the IE3D 10.0 is the implementation of accurate, robust and complete near field calculation and visualization.

Near field was implemented before the IE3D 9.0. However, the implementation was not a robust one. Also, it only calculated the E-field and it is not complete. When the IE3D 9.0 was released, we did not have enough time to implement robust near field calculation and visualization into the IE3D 9.0. The project is delayed. On the IE3D 10.0, we have implemented robust and accurate near field calculation and visualization. Also, the near field visualization in the IE3D 10.0 includes most of the field quantities of interests. The quantities can be displayed are documented in Table AF.1. In our following discussion, any bold italic symbol means that it is a vector.

Table AF.1 The near field quantities can be displayed on the IE3D 10.0.

Name	Symbol in the Manual	Symbol in MGRID
Horizontal Scalar Potential	φ_h, Phi-h	Phi-h
Vertical Scalar Potential	φ_z, Phi-v	Phi-z
Horizontal Vector Potentials	Ax, Ay or $\boldsymbol{A}$t	Ax or Ay
Vertical Vector Potential	Az	Az
Electric Field Components	Ex, Ey, Ez or $\boldsymbol{E}$t	Ex, Ey or Ez
Magnetic Field Components	Hx, Hy, Hz or $\boldsymbol{H}$t	Hx, Hy or Hz
Poynting Vector Components	Px, Py or Pz	Px, Py or Pz

The primary field quantities are the φ_h, φ_z, $\boldsymbol{A}t = Ax\,\boldsymbol{x} + Ay\,\boldsymbol{y}$ and Az. The other field components are defined as:

$$\boldsymbol{E} = \boldsymbol{E}t + Ez\,\boldsymbol{z} \tag{AF-1}$$

$$\boldsymbol{E}t = Ex\,\boldsymbol{x} + Ey\,\boldsymbol{y} = -j\,\omega\,\boldsymbol{A}t - \nabla_t\varphi_h = -j\,\omega\,\boldsymbol{A}t - \left(\partial\varphi_h/\partial x\,\boldsymbol{x} + \partial\varphi_h/\partial y\,\boldsymbol{y}\right) \tag{AF-2}$$

$$Ez = -j\,\omega\,Az - \partial\varphi_z/\partial z \tag{AF-3}$$

$$\boldsymbol{H} = 1/\mu\,\nabla \times \boldsymbol{A}t \tag{AF-4}$$

$$\boldsymbol{P} = \boldsymbol{E} \times \boldsymbol{H}^* \tag{AF-5}$$

Where "*" means complex conjugate.

On the IE3D 10.0, we are able to calculate and display all the quantities in Table AF.1. However, we calculate the primary field quantities first. We use the equations (AF-1) to (AF-5) to calculate the other field components. Generally, speaking, the field primary field quantities are more accurate. The secondary field components (***E***, ***H*** and ***P***) based upon numerical differentiations in the equations.

We will use the .\ie3d\samples\c_bend3nf.geo as our example. Please simulate it at 5 and 40 GHz for 8 frequency points with the Current Distribution Data saved on c_bend3nf.cur file and AIF disabled. Please select **Process->Near Field Calculation** command on MGRID. MGRID will prompt you for the **Near Field Calculation Parameters** dialog (see Figure AF.2). It needs you to define which frequency points you want to calculate the near field (top right list box). It shows you the dielectric interfaces on the top left list box. The dialog allows you to enter some grid points in the X-, Y- and Z-directions. Please select the Insert button at the top right corner of the X-Grids list box to insert some X-grid points: Start = -0.2, End = 0.2 and Number = 17 (see Figure AF.1). You can see there is another parameter called Differential Sign.

We are using numerical differentiations to calculate the secondary field components. The Differential Sign controls how we calculate the numerical differentiations. If you choose the Differential Sign as "+", it means that we will find an potentials at $x2 = x1 + \Delta x$ ($\Delta x > 0.0$) to calculate the numerical differentiation as:

d φ / dx ≈ (φ_2 - φ_1) / Δx. If you choose the Differential Sign as "-", you will have the Δx < 0.0 at the grid points. The |Δx| is defined as the Differential Step in Figure AF.2.

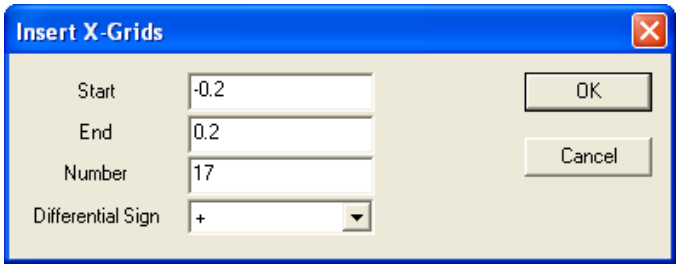


Figure AF. 1 The dialog for inserting some points for near field calculation.

Please define the following grid points for Y: Start = -0.2, End = 0.2 and Number = 17. For the z-direction, we need to be more considerate. We have a dielectric interface at Z = 0.1. Please define the Z-grids as: Start = 0, End = 0.3 and Number = 13. MGRID will create grid points at 0, 0.025, 0.050, 0.075, 0.1, 0.1, 0.125, 0.150, 0.175, 0.200, 0.225, 0.250, 0.275 and 0.3. You may notice there are two grids at 0.1. However, the signs are different. It is for consideration of that some field components can be discontinuous at Z = 0.1.

In case, you have created some near field data files (.fld) before and you want to get the field grid points from the file, you can use the Capture Grids from File button to capture the grid points from an existing .fld file.

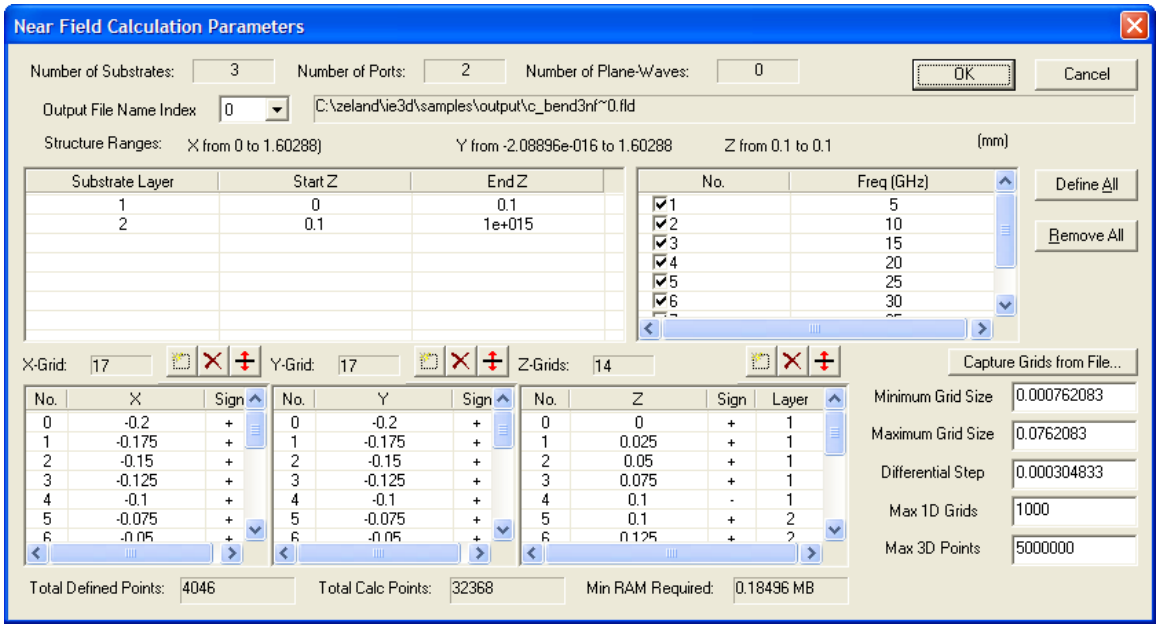


Figure AF.2 The dialog after all the field points are defined.

You may notice that there are some parameters below the grid point list boxes. It will estimate how much RAM is required to do the near field calculation. In case you define too many grid points, it may require large amount the memory. MGRID will issue you a warning. Basically, the grid points we define in the list boxes determine the volume of the near field visualization. The bigger space you define, you will be able to view the near field in a larger range. However, the calculation time will be increased. When the

space is fixed, the more points you put in the space, the higher resolution you will get. However, again, the calculation time will be longer.

The near field data is saved into the c_bend3nf.fld file in the output sub-directory. The near field data can be opened for display later. We have not defined the excitations for the structure yet. Basically, we do not need to define the excitations at the time of the calculation. MGRID will allow us to define the excitations at the time of displaying the near field. After you enter all the parameters as shown in Figure AF.2, please select OK to continue. MGRID will start the near field calculation process. It will take couple minutes for it. After it finishes, it will show you the dialog in Figure AF.3.

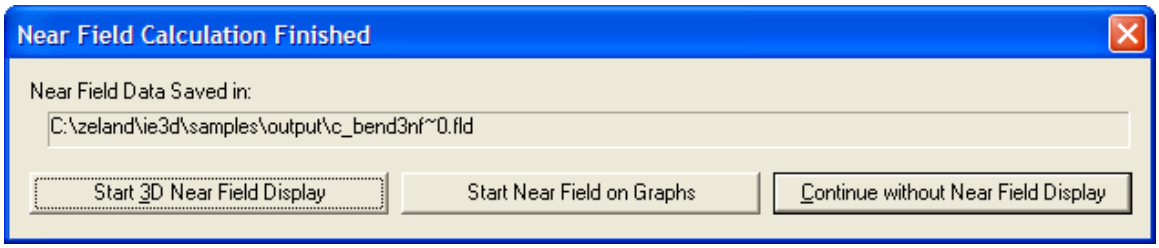


Figure AF.3 The dialog after the near field calculation is finished.

You have 3 options: (1) Start 3D Near Field Display; (2) Start Near Field on GUI; (3) Continue without Near Field Display. If you choose (1), MGRID will start the 3D display. If you choose (2), MGRID will start the 2D display on graphs. If you choose Continue Without Near Field Display, MGRID will resume the display we were at when you start the near field calculation. Anytime, we want to display the near field saved into a .fld file, we can open the geometry file or the current distribution file. Then, we can select the **Window->3D Near Field Display or 2D Near Field Plots** commands.

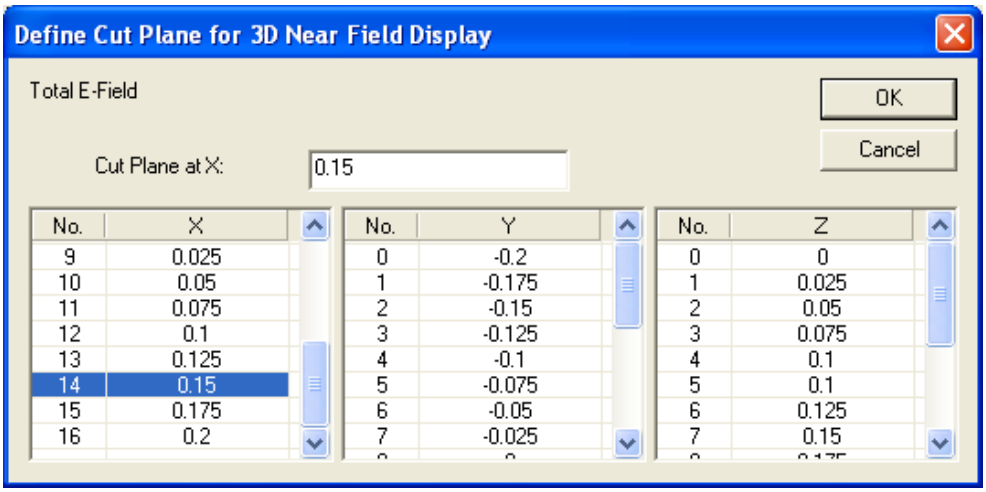


Figure AF. 4 The dialog for defining a cut plane for display.

Please select Start 3D Near Field Display button. MGRID will load the file and show you the Near Field 3D Display dialog (see Figure AF.5). Please select the No.1 frequency, Scalar Potential for Field Quantity, and Phi-h for the Component. Then, please select the Insert (or Add) button above the right most list box. MGRID will prompt you for the coordinate of the cut plane. You can define one at any direction. Please select the No.14 X = 0.15. You will see the Cut Plane at X: 0.15. In fact, the coordinate is not necessary to be at a grid point. You can define any point within the minimum and the maximum. Please select OK to continue. A cut plane is added into the Cut list box. Please select the Add button above the Cut

list box again. Please define the Cut Plane at Y: 0.15. Select OK. The 2nd cut plane is added into the list box. You should get the picture shown in Figure AF.5.

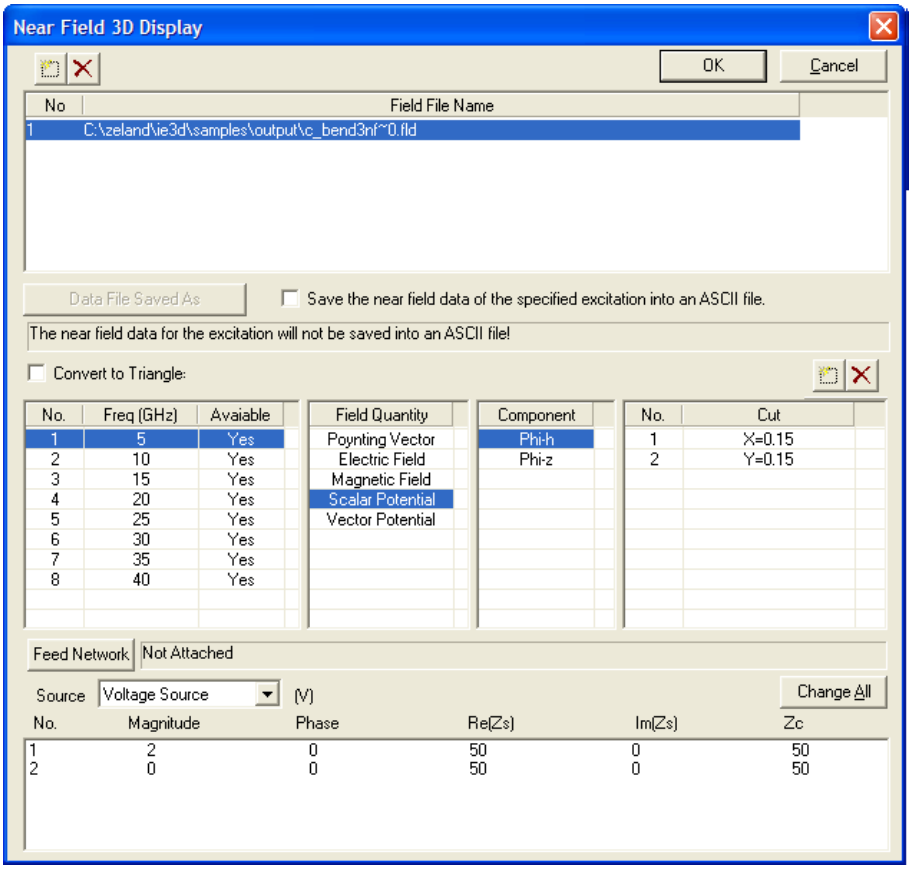


Figure AF. 5 The Near Field 3D Display dialog after 2 cut planes are defined.

As you can see from Figure AF.5, you are allowed to define the excitation. You are even allowed to attach a Feed Network (or .ect file) to it.

Please select OK to continue. MGRID will display the 3D near field together with the structure (see Figure AF.6). You can select the **Options->Set Graph Parameters** to change the settings of the graph.

What is displayed on the Near Field 3D Display is the field distribution on the structure. If we like, we can also display the near field as curves in Cartesian coordinate system. Please select **Window->2D Near Field Plots->Define 2D Near Field Plot** command. MGRID will prompt you the Near Field 2D Display dialog (see Figure AF.9).

Please select the No.1 Frequency = 5 GHz. Then, please select the Add (or Insert) button to define a graph. MGRID will prompt you for the display quantity (see Figure AF.8). Please select Phi-Potential, Magnitude/Angle and Z-Direction (see Figure AF.8). Select OK to continue. The Select Items for Near Field Graph dialog comes up (see Figure AF.9). Please select the Add (or Insert) button on the list box. The Add Scalar Potential Display Item dialog comes up (see Figure AF.7).

Please select The X = 0.05 and Y = 0.05. Pleas select Mag(Phi-h) in the Field Component list box (see Figure AF.7). Select OK. The item is added into the Select Items for Near Field Graph dialog (see Figure AF.9).

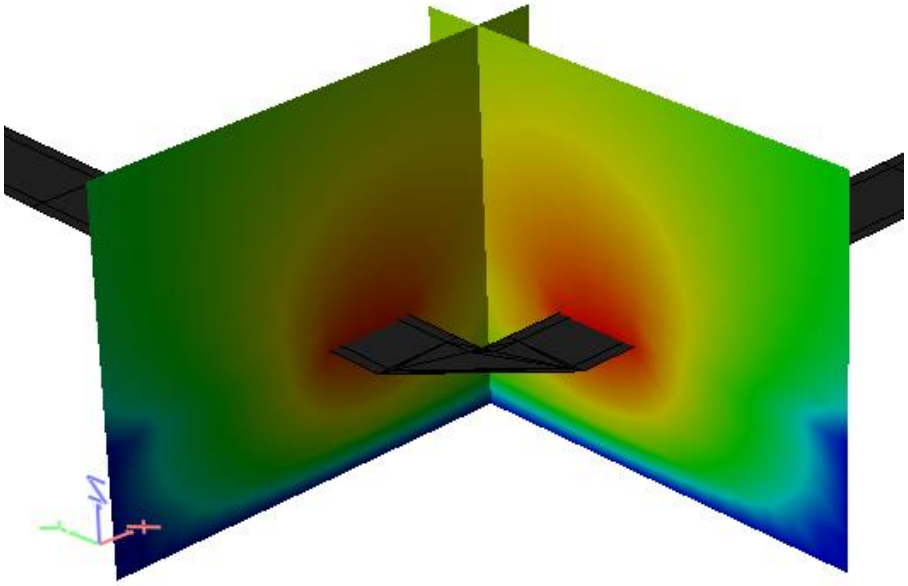


Figure AF.6 The 3D near field display of the Phi-h on the bend.

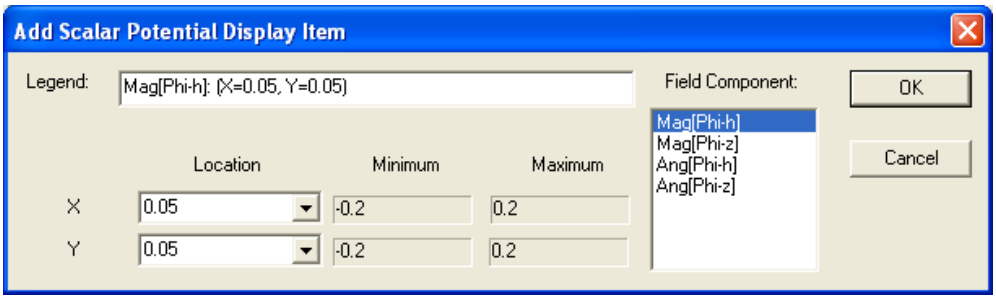


Figure AF.7 The Add Scalar Potential Display Item dialog.

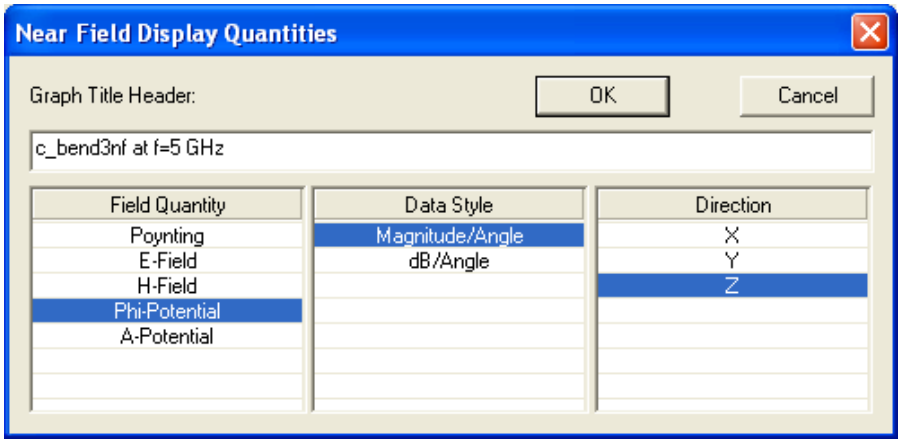


Figure AF.8 The Near Field Display Quantities dialog.

We can add more items (or curves) into the graph if we like. Please select OK to continue. The graph with 1-curve is added into the list box (see Figure AF.10). We can also add more graphs if we want. You have the option to save the near field data of the specified excitations into an ASCII file. Please select OK

to continue. The near field at (X, Y) = (0.05, 0.05) is displayed in Figure AF.11. As you can see, the maximum is at Z = 0.1 mm or the location of the strip.

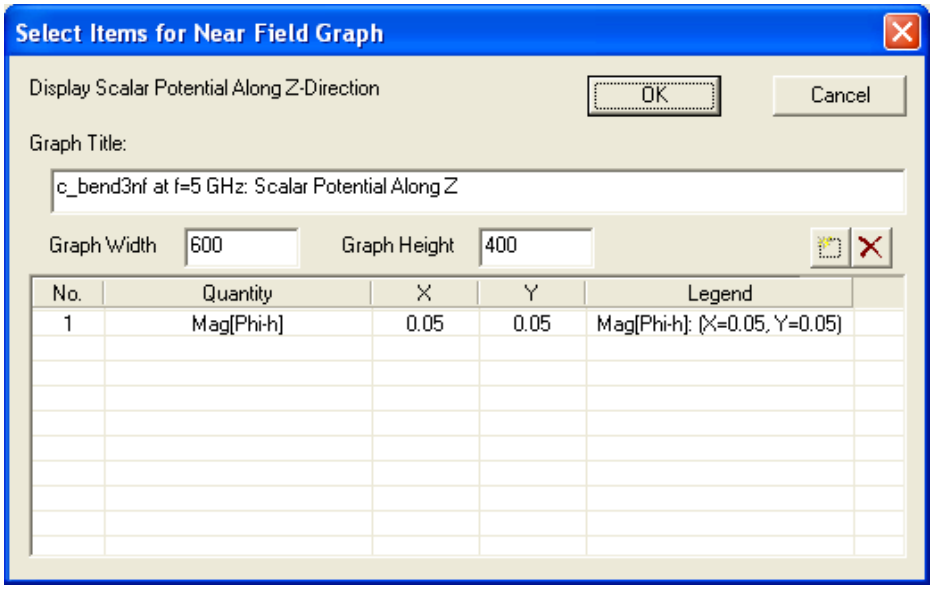


Figure AF.9 The Select Items for Near Field Graph dialog.

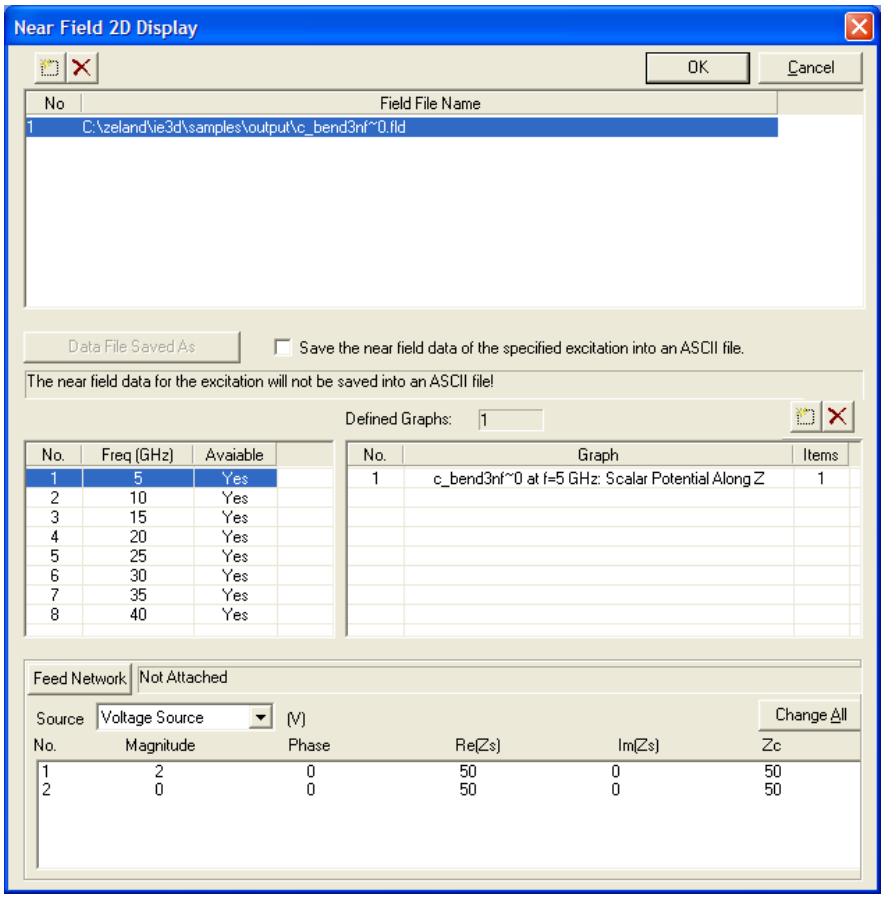


Figure AF.10 The Near Field 2D Display dialog.

There are many different quantities you can display on the Near Field Visualization on the IE3D. We will not demonstrate all the features here. Interested users can explore them.

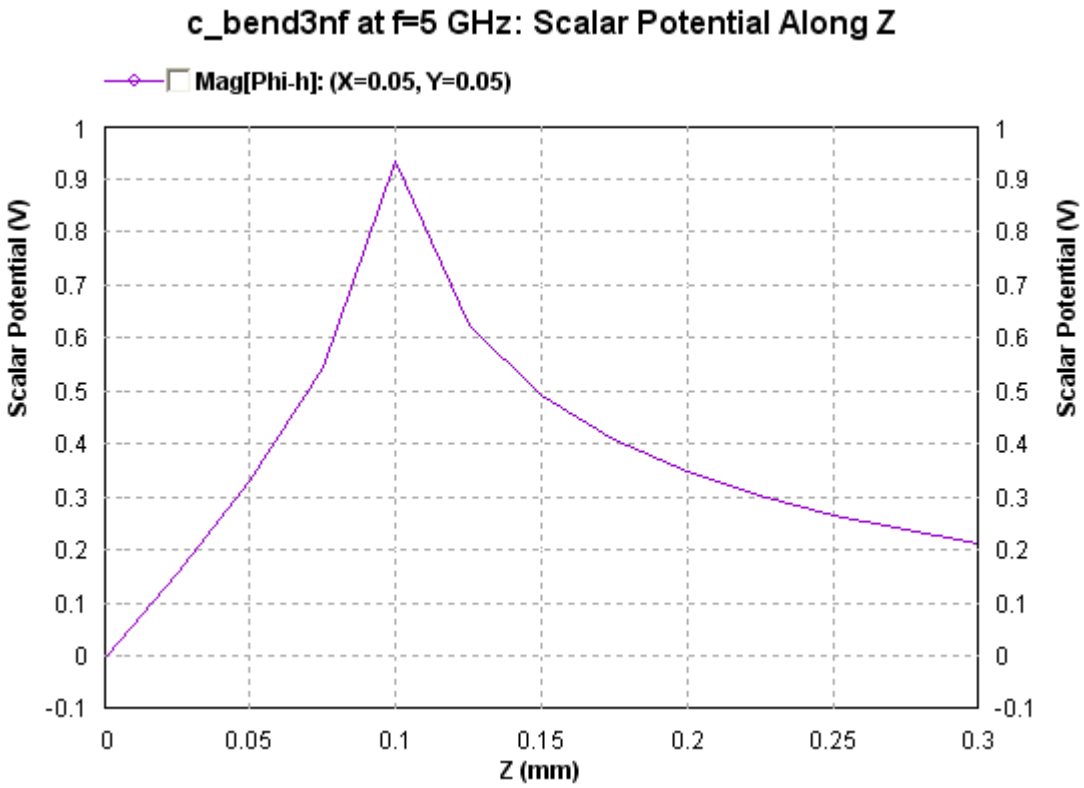


Figure AF.11 The near field distribution along Z at (X, Y)=(0.05, 0.05).

Appendix AG. User Programmable IE3D Objects for Ultimate Flexibility In Full-Wave EM Optimization

As you can see, MGRID and Ie3dLibrary are quite powerful layout editors. They allow users to create complicated structures. They also allow users to define optimization variables on structures to make the structure parameterized.

However, there are cases you may want to have more control over the parameterized geometry. For example, you have some programs to general the shapes of a geometry based upon a few parameters. Building the parameterized geometry as objects in the Ie3dLibrary may be a very tedious job. You would like to be able to link your geometry generator to our software so that the IE3D can optimize your geometry through the few parameters. In such a case, you may want to use the: User Defined Object in the Ie3dLibrary.

Basically, on the Ie3dLibrary, we allow you to create a User Defined Object as a link to an external executable. When you define the object, you just need to tell Ie3dLibrary the parameters specified in Figure AG.1 and Table AG.1. Internally, at the time of geometry creation which happens at a display, a simulation or a specific iteration of an optimization), Ie3dLIbrary will try to call the External Program to create the structure. The External Program is provided by you, the user. The External Program will read the information in the User Defined Control (or UDC) file, which is provided by the Ie3dLibrary as the last argument in the call to the External Program. The content of the UDC file is documented in Table AG.2 to Table AG.6.

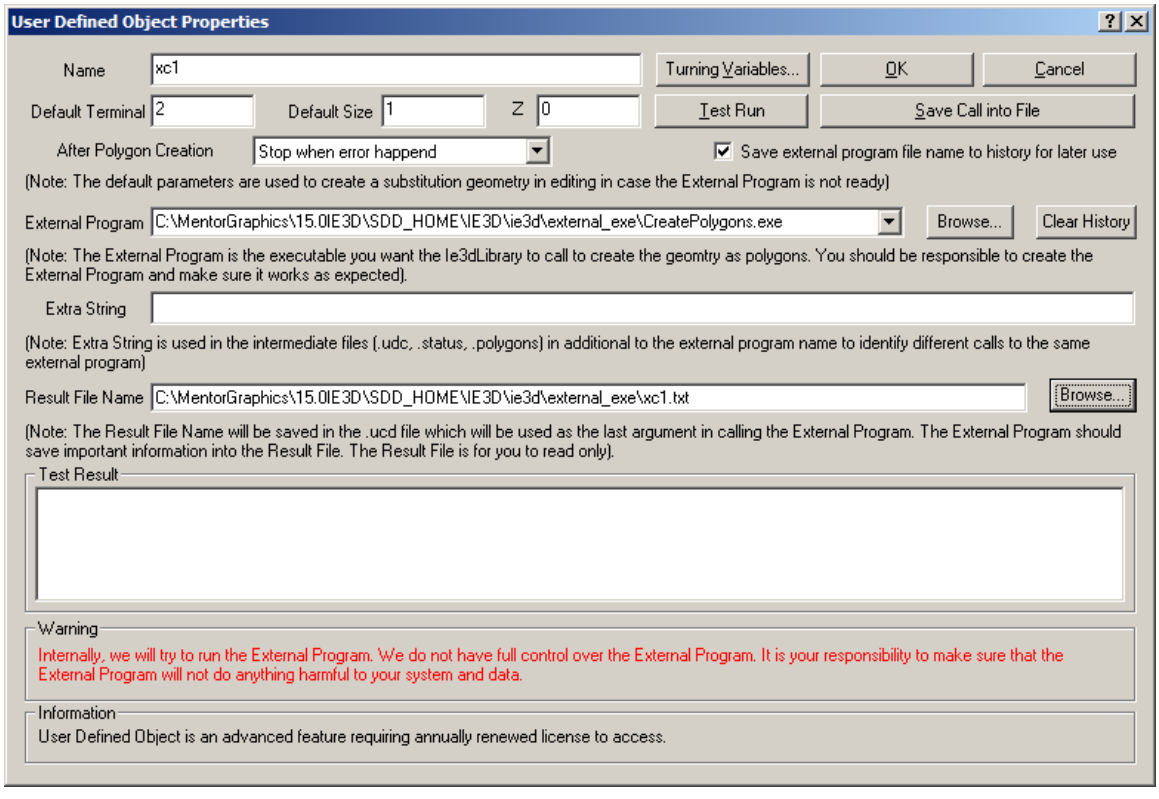


Figure AG.1 The User Defined Object Properties dialog on Ie3dLibrary.

Table AG. 1 The parameters of the User Defined Object in Ie3dLibrary.

Parameter	Description

Name	A user can use a name to define the object.
External Program	It is the executable including the command line arguments a user wants the Ie3dLibrary to call to create the polygons. For example, you can enter it as: "c:\my_exe_dir\my_exe" "c:\my_data_dir\mydata1.txt" for the command. Please try to use double quotation marks because it will avoid any possible ambiguities caused by spaces in your path names. On the Ie3dLibrary, it will call command as: "c:\my_exe_dir\my_exe" "c:\my_data_dir\mydata1.txt" "temp_file.udc" The "c:\my_exe_dir\my_exe" is the command you use to create the geometry. The "c:\my_data_dir\mydata1.txt" is a possible file for your command. The "temp_file.udc" file is a file created by Ie3dLibrary at run time. It contains the information such as the .sts file name for communication between your executable and the Ie3dLibrary, the polygon file name for your executable to store the polygon data, and the offset values for the optimization variables defined in the Ie3dLibrary.
Number of Connections	It tells the number of connections the user wants his automatically created geometry to the other objects on the Ie3dLibrary.
Default Z-Coordinate	In case, the executable is not available or failed at the time of geometry editing, Ie3dLibrary will assume a default shape to represent the object. The Default Z-Coordinate is the z-coordinate of the polygon representing the default shape of the object in case the command failed.
Default Size	The size of the default shape representing the object in case the command failed.

Table AG.2 The content of the example UDC file for the bend.ie3 file.

Content	Se n #
<?xml version="1.0" encoding="utf-8" ?> <udc_data> <version_number>10.0</version_number> <user_defined_command>C:\MentorGraphics\<latest_release>IE3D\SDD_HOME\IE3D\ie3d\external_exe\CreatePolygons.exe</user_defined_command> <status_file_name>C:\DOCUME~1\COMPAQ~1\LOCALS~1\Temp\sta2962.tmp</status_file_name> <polygons_file_name>C:\DOCUME~1\COMPAQ~1\LOCALS~1\Temp\pol2963.tmp</polygons_file_name> <result_file_name>C:\MentorGraphics\<latest_release>IE3D\SDD_HOME\IE3D\ie3d\external_exe\xc1.txt</result_file_name> <number_of_mets>2</number_of_mets> <number_of_terminals>2</number_of_terminals> <length_unit> <name>(mm)</name> <value>1.000000e-003</value> </length_unit> <optim_variable> <name>length</name> <value>0.000000e+000</value> <low_bound>-1.000000e+000</low_bound> <high_bound>1.000000e+000</high_bound> <accumulation>0.000000e+000</accumulation> </optim_variable> <optim_variable> <name>width</name> <value>0.000000e+000</value> <low_bound>-1.000000e+000</low_bound> <high_bound>1.000000e+000</high_bound>	

<accumulation>0.000000e+000</accumulation> </optim_variable> </udc_data>

Table AG.3 The explanation of the UDC file.

Section #	Description
1	XML file header indicating it is an XML file. It is the same for every XML file.
2	Beginning root tag for UDC file. Anything between the beginning root tag and the end tag belongs to the UDC file.
3	UDC file format number and it is 10 now.
4	The External Program name defined in the User Defined Object in Ie3dLibrary
5	It is the status file name. Anytime, you can write two parameters as one line in this file. The 1st parameter is an integer referred as CODE in the next. The 2nd parameter is a string. If the CODE = 0, it means that the polygons are created and saved successfully. If the CODE ≠ 0, the message will be detected by Ie3dLibrary and displayed to you. In case, you encounter any problem in the External Program, you should write a line with CODE ≠ 0 and a string following it. Remember, only the 1st line of the code will be read. If you write multiple lines into the file, Ie3dLibrary will not read the 2nd and the other lines. When you write to the file, remember to overwrite the old file.
6	It is the polygon file name. The polygon file should be in XML format. Each polygon should have at least 3 vertices. Each vertex has (X, Y, Z) coordinates. It also has an integer following the (X, Y, Z) coordinates. The integer is used to denote whether the edge formed by the vertex and the next vertex is defined as a terminal. For a terminal, the Z-coordinate of the 2 vertices must be the same. The number of terminals in the file should match the Number of Terminals in the Section #9. Each terminal must be and can only be defined on one edge only. The format of the polygon file is documented in Table AG.4
7	It is the result file name. The result file is used to store the important information for the user. It is not read by the Ie3dLibrary and other software. You should make it easy to understand for yourself. For our bend.ie3 example, we store the LENGTH and WIDTH information. From the LENGTH and WIDTH example, you will know the exact dimensions of your structure. If it is for simulation, the result file will store the information of the last simulation. If it is for optimization, the result file will store the results of the optimized geometry.
8	It is the number of metallic strip types Nmets. For this case, Nments = 2. Each of the created polygons must be assigned with a metallic type Imet. It is used in modeling the loss effect of the structure. The Imet must be between 1 and Nmets.
9	It is the number of terminals. It is the maximum number of connections you want the object to be connected to other objects in the design.
10	It is the length unit for the object. You should store the vertices of the polygon using the length unit specified here: The length unit has a symbol and a value. The value is the value of the unit in meter. The length unit pair takes: (1) millimeter: symbol = (mm) and value = 1.0e-3; (2) mil (1/1000 inch): symbol = (mil), value = 2.54e-6; (3) micron: symbol = (micron), value = 1.0e-6; (4) meter: symbol = (m), value = 1; (5) inch: symbol = (inch), value = 0.0254; (6) foot: symbol = (foot), value = 0.3048; (7) cm (1/100 m): symbol = (cm), value = 0.01.
11	It defines a variable as: “name”, “value”, “low bound”, “high bound” and “accumulation”. You can use “name” to identify which variable it is. For internal objects, the value is the current offset value of the variable. For External Program, you are suggested to use the sum of “value” and “accumulation” as the offset value of the variable for more consistency. If you do not include accumulation, Ie3dLibrary may not yield the right geometry for you in updating the design. There may be multiple variables in the file. For this example, there are 2 of them. The Section #12 defines the 2nd variable.
12	It defines the 2nd variable for the bend example.
13	The end tag of the UDC file section.

Table AG. 4 The polygon file format.

Content	Section #
<PolygonList>	1
<Ie3dPolygon mets="1">	2
<NodeAndSideList>	3
<NodeAndSide>-3.750000e-002,-3.750000e-002,1.000000e-001,0</NodeAndSide>	4

<NodeAndSide>-3.750000e-002,3.750000e-002,1.000000e-001,0</NodeAndSide>	5
<NodeAndSide>3.750000e-002,3.750000e-002,1.000000e-001,0</NodeAndSide>	6
<NodeAndSide>3.750000e-002,-3.750000e-002,1.000000e-001,0</NodeAndSide>	7
</NodeAndSideList>	8
</Ie3dPolygon>	9
<Ie3dPolygon mets="1">	10
<NodeAndSideList>	11
<NodeAndSide>-3.750000e-002,3.750000e-002,1.000000e-001,0</NodeAndSide>	12
<NodeAndSide>-3.750000e-002,5.375000e-001,1.000000e-001,1</NodeAndSide>	13
<NodeAndSide>3.750000e-002,5.375000e-001,1.000000e-001,0</NodeAndSide>	14
<NodeAndSide>3.750000e-002,3.750000e-002,1.000000e-001,0</NodeAndSide>	15
</NodeAndSideList>	16
</Ie3dPolygon>	17
<Ie3dPolygon mets="1">	18
<NodeAndSideList>	19
<NodeAndSide>3.750000e-002,-3.750000e-002,1.000000e-001,0</NodeAndSide>	20
<NodeAndSide>3.750000e-002,3.750000e-002,1.000000e-001,0</NodeAndSide>	21
<NodeAndSide>5.375000e-001,3.750000e-002,1.000000e-001,2</NodeAndSide>	22
<NodeAndSide>5.375000e-001,-3.750000e-002,1.000000e-001,0</NodeAndSide>	23
</NodeAndSideList>	24
</Ie3dPolygon>	25
</PolygonList>	26

Table AG.5 The explanation of the polygon file.

Section #	Description
1	The start tag of the polygon list.
2	The start tag of a polygon. The “mets” indicates the polygon is of metallic type 1.
3	The start tag of the NodeAndSide list.
4	The location of the vertex and the edge index. If it is not a connection terminal, the edge index should be 0. Please check the Section #13, for the 2nd vertex of the 2nd polygon defined in Section #13, the edge index is 1. It means that that edge is the No.1 connection terminal of the object. The 3rd vertex of the 3rd polygon defined in Section #22 has the edge index = 2 and it is the 2nd connection terminal of the object. For the edge index, it is always formed by the current vertex and the next one. If the current vertex is the last vertex of the polygon, the next vertex is the 1st vertex of the polygon. If it is a connection terminal, the 2 vertices should have the same z-coordinate.
5, 6, 7	The other 3 vertices of the polygon. This polygon has 4 vertices. A polygon should have at least 3 polygons. Also, no 2 adjacent vertices should be at the same location, otherwise they will be merged into 1 vertex. Ie3dLibrary will check the polygons too.
8	The end tag of the NodeAndSide list.
9-25	It defines 2 more polygons. There are total 3 polygons defined in the file. Please note that whether two polygons are connected depends upon whether the two polygons have a pair of matching edge. There is not indication for it. Ie3dLibrary will detect their physical locations to see whether the 2 edges are matching. If the 2 edges are matching, it will be considered as connected. For example, the 2nd vertex of the 1st polygon (Section #5) is at the same location as the 1st vertex of the 2nd polygon (Section #11), and the 3rd vertex of the 1st polygon (Section #6) is at the same location as the 4th vertex of the 2nd polygon (Section #15). Therefore, the 1st polygon is connected to the 2nd polygon at the 2nd edge of the 1st polygon (2nd and 3rd vertices) and the 4th edge of the 2nd polygon (4th and the 1st vertices).
26	The end tag of the polygon list.

Table AG. 6 The procedure the Ie3dLibrary to create polygons from a User Defined Object.

Steps	Description
1	It is at the start of an IE3D simulation (or some other procedure requiring creation of the polygons from the objects) on the Ie3dLibrary.
2	Ie3dLibrary will set the offset value of each optimization variable.

3	Ie3dLibrary will call the CreatePolygons command of each object to create the polygons with the offset values of the optimization variables defined: (1) If an object is an internal object, Ie3dLibrary will call the internal routine to create the polygons. (2) If the object is a User Defined Object, Ie3dLibrary will create a UDC file. It will call the External Program as a command specified by the user with the UDC file as the last command line argument. Then, Ie3dLibrary will wait for the external command to finish the job.
4	The user is supposed to create an executable for the External Program. The external executable will get the information from the UDC file. It will read the .sts file name, the polygon file name, the number of connections, and the offset values it. The external executable should check and make sure the number of connections matches the expectation. From the offset values of the optimization variables, the external command should create the polygons with proper connection information. The polygons should be saved into the polygon file in XML file format before the external command stops. In case there is any error, the external command should write an error code (non-zero value) and a message into the .sts file. In case, the external command wants to pass some information to the Ie3dLibrary, it should write a code (non-zero value) and a message into the .sts file. Anyway, only the 1st line of the .sts file count. When it finishes, it should write a code = 0 to the file.
5	When Ie3dLibrary detects the external command finishes, it will read the polygon file for the polygons. In case it encounters error code from the .sts file, it will report to the user. If it is in the editing mode, it will use the default parameters to create some polygons to show the users.
6	Ie3dLibrary loops through all the objects to create the polygons.
7	In case no error is detected, it will start the Ie3d simulation.
8	After a simulation, it will check whether performance of the circuit. If it does not meet the optimization requirement, it will try to find the offset values of the optimization variables for the next iteration. It will go back to step 1.

We will show 2 examples in the following. Our example External Program is saved in C:\MentorGraphics\<latest_release>IE3D\SDD_HOME\IE3D\ie3d\external_exe\CreatePolygons.exe. It is created using the VC++ 6.0 project saved in C:\MentorGraphics\<latest_release>IE3D\SDD_HOME \IE3D\ie3d \external_exe\CreatePolygons_vc6. The corresponding VB 7.0 project is saved in C:\MentorGraphics \<latest_release>IE3D\SDD_HOME\IE3D\ie3d\external_exe\CreatePolygons_vb7. You can use them as examples to build your own External Programs for the User Defined Objects. It does not matter what language or program environment you are in. You are required to read the UDC file and create the polygons in XML format with the requirements documented in this appendix.

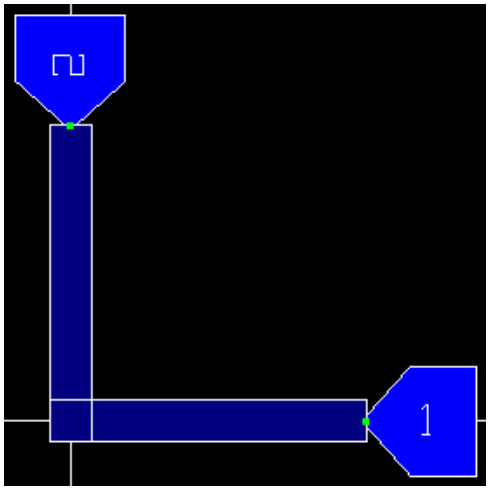


Figure AG-2.

Please run the Ie3dLibrary and open the file: . \ie3d\external_exe\bend.ie3 file. You will see the bend with 2 ports (see Figure AG.2). This bend is basically defined by the External Program: .\ie3d\external_exe\CreatePolygons.exe. The arm length of the bend is controlled by the No. 1 variable "length" in the bend.ie3 project. Please select Edit->Tuning Variable command of Ie3dLibrary (see Figure AG.3). The width of the arm of the bend is controlled by the No.2 variable "width" in the bend.ie3 project (see Figure AG.3).

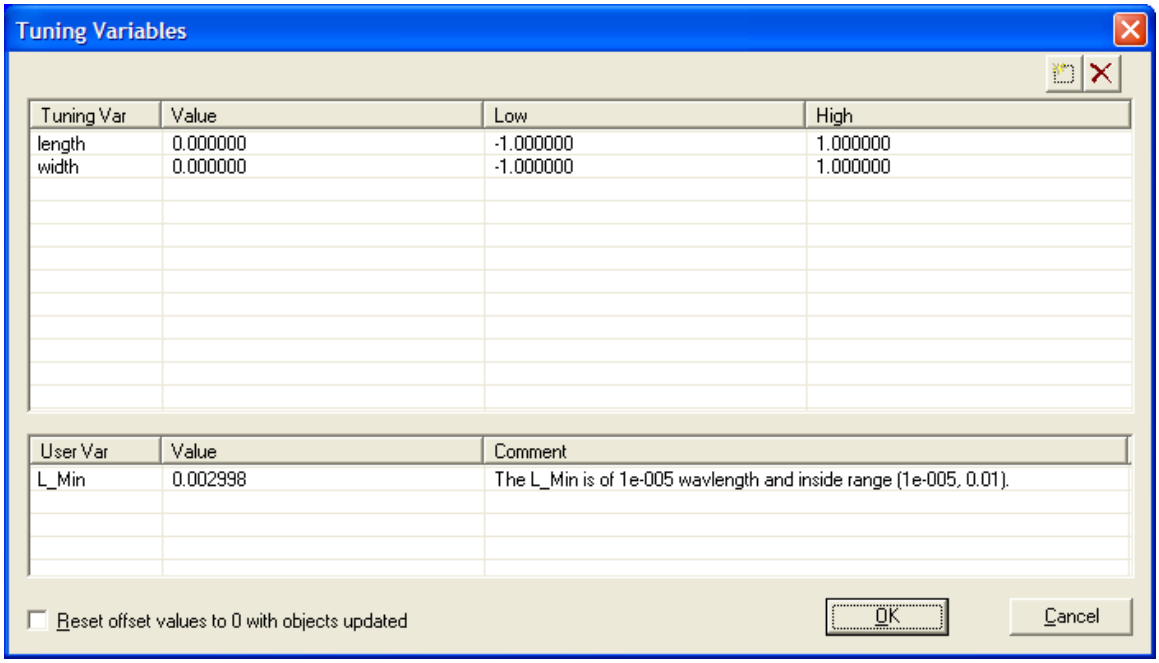


Figure AG.3 The Tuning Variable Array dialog for Edit->Tuning Variables command.

As you can see, each Tuning Variable has "name", "value", "low (bound)", "high (bound)" and "Accumulation". Anytime, we can assign a "value" to a tuning variable. Then, the "value" in the tuning variable will control the shapes of the structure if some dimensions of the internal IE3D objects or User Defined Objects are associated with the variable. Anytime, you can reset the "value" of each variable to 0 by check the Check Box for "Reset Offset Values to 0 With Objects Updated". When you select OK with the Check Box checked, Ie3dLibrary will take the following actions: (1) new "Low" = old "Low" – old "Value"; (2) new "High" = old "High" – old "Value"; (3) new "Accumulation" = old "Accumulation" + old "Value". (4) Then, it will set the dimensions of all the objects associated with this variable to the value corresponding to the old "Value". (5) Finally, it will set the "Value" to 0. In such a way, we can make sure all the dimensions of the objects of the structure are consistent. For example, after an optimization, we will get the "value" of each Tuning Variable as non-zero. We want to reset it to 0. Taking the about procedure automatically, Ie3dLibrary will make sure your structure is the same one as the one with the specified old "value" for each variable while the new "value" of the variable is set to 0.

In case we have User Defined Objects in the project, the above procedure may have problem to make sure the geometry is consistent before resetting the "value" for each variable. If you make sure you use the variable as: "final value" = "value" + "accumulation" in the External Program for a User Defined Object, you will be able to make sure the geometry is consistent before and after resetting the "value" to 0. If you use the "final value" = "value" in the External Program for a User Defined Object, you will get a different geometry after you reset the "value" to 0.

In the CreatePolygons_vc6 project (or CreatePolygons_vb7 project), we describes the User Defined Objects as 3 polygons: The corner square of W by W (W is the width) and 2 rectangles of W by L (L is the length). By changing the "value" of the variables "length" and/or "width", we are able to control the shape of the bend. In the CreatePolygons_vc6 project, we defines:

$$L = L0 + dL * (\text{value} + \text{accumulation of variable "length"}) \quad \text{AG-1}$$

$$W = W0 + dW * (\text{value} + \text{accumulation of variable "width"}) \quad \text{AG-2}$$

Where L0 = 0.5 mm, dL = 0.4 mm, W0 = 0.075 mm, dW = 0.025 mm. Because the range of the "length" variable is from –1 to 1 and the range of the "width" variable is from –1 to 1, the range of the L is from 0.1 mm to 0.9 mm and the range of the W is from 0.05 mm to 0.1 mm.

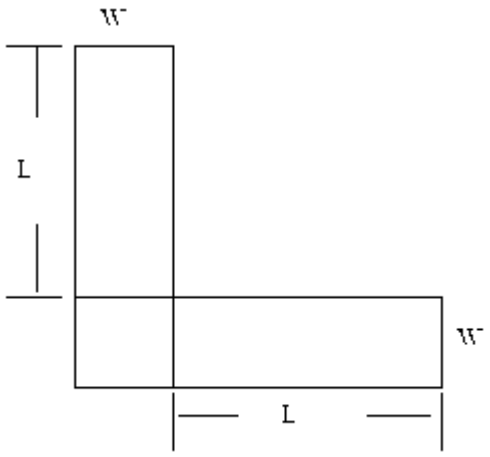


Figure AG.4.

We can set some "value" to each variable in the Edit->Tuning Variables dialog and un-check the "Reset the Offset Values to 0 with Objects Updated". For example, we can set the "value" of variable "length" as 0.5 and the value of variable "width" as –0.5. We select OK. Then, we select Process->Simulate command to simulate the structure. Ie3dLibrary will simulate the structure as: L = 0.5 + 0.4 * 0.5 = 0.7 mm, and W = 0.075 + 0.025 * (-0.5) = 0.0675 mm. Please note that the CreatePolygons example is written in such a way that the L, W and Z are independent of the length unit value saved in the UDC file. Their values in meter are always the same no matter what length unit it is.

After the simulation, Ie3dLibrary will display the s-parameters. The information on the final L and W will not be displayed on the Ie3dLibrary. In fact, we can not know it from the Ie3dLibrary. You should open and check the file: .\ie3d\external_exe\xc1.txt (the result file specified in Figure AG.1) for it. It is very important that the you should make sure the result file save all the important information about your geometry. In case it is optimization, the result file will save the dimensions of optimized (best) geometry.

Please note that there is a check box "Save information to history for later use" in Figure AG.1. When you check it, Ie3dLibrary will save the names of the External Program into a list. You can recall the name later when you are defining the same object in the same or other design.

The User Defined Object in the Ie3dLibrary allows you to program your own geometry generator. It is even more powerful and flexible than script language based geometry generators. It will make the IE3D more useful to the users.

Appendix AH Electromagnetic Synthesis of Filters on Ie3dLibrary

Ie3dLibrary promises a significant boost to IE3D. With lots of pre-defined circuit elements in actual layout, Ie3dLibrary allows a user to generate complicated circuitry with a few mouse clicks. The effective use of an optional module of Ie3dLibrary, FilterSyn, will demonstrate the electromagnetic simulation based synthesis of filters in parallel with equivalent circuit based synthesis.

Start Ie3dLibrary by Clicking Ie3dLibrary from IE3D-SI menu in IE3D Program Manager and select **New** from the **File** menu. We'll have a window like Figure AH.1 below.

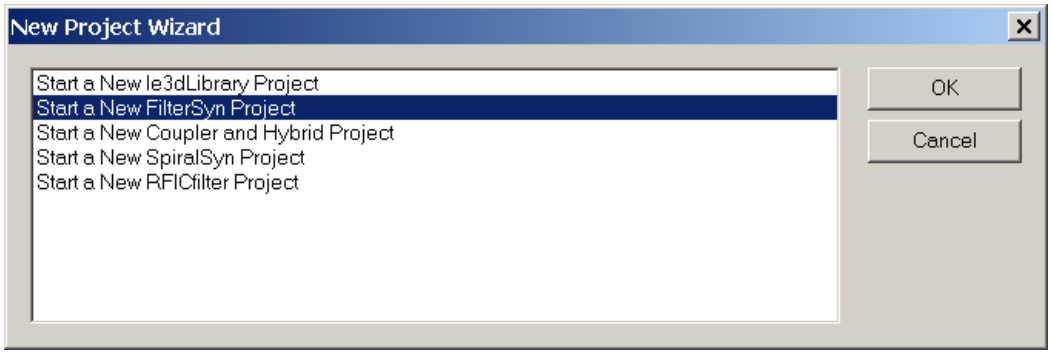


Figure AH.1 New Ie3dLibrary project window.

Selecting **Start a New FilterSyn Project** from the list and clicking OK will present a window like Figure AH.2. The same window will appear also if we select **Start a New Ie3dLibrary Project** and then select **Filters** from the **Modules** menu.

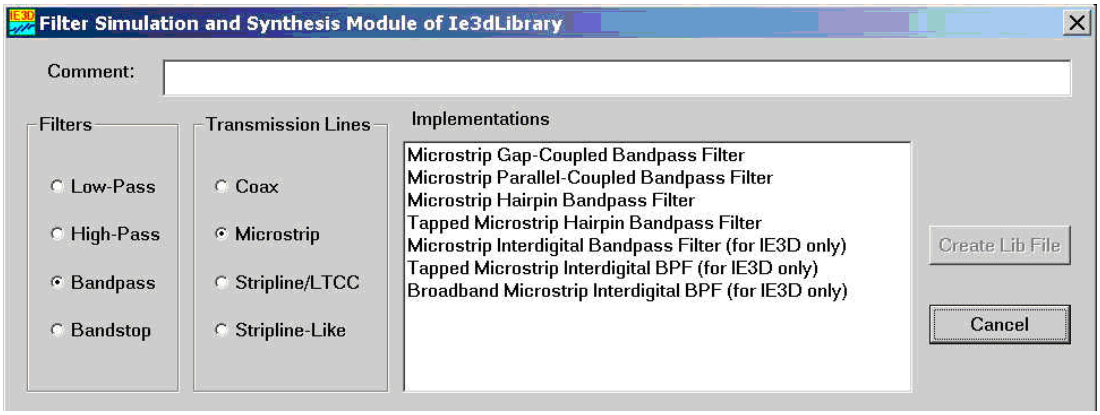


Figure AH.2 FilterSyn startup window.

We are going to synthesize a LTCC bandpass filter using FilterSyn and IE3D. As we go through the process, we will see the respective role of FilterSyn and IE3D. FilterSyn serves as a quick but approximate design generator and IE3D as a slower but accurate tool. Without FilterSyn, IE3D can still be used to produce a good filter design, but that would require more work and knowledge from a designer. If we check Bandpass for Filters and Stripline/LTCC for Transmission Lines and select LTCC Combined Broadside/Parallel-Coupled BPF, we will have a dialog like Figure AH.3.

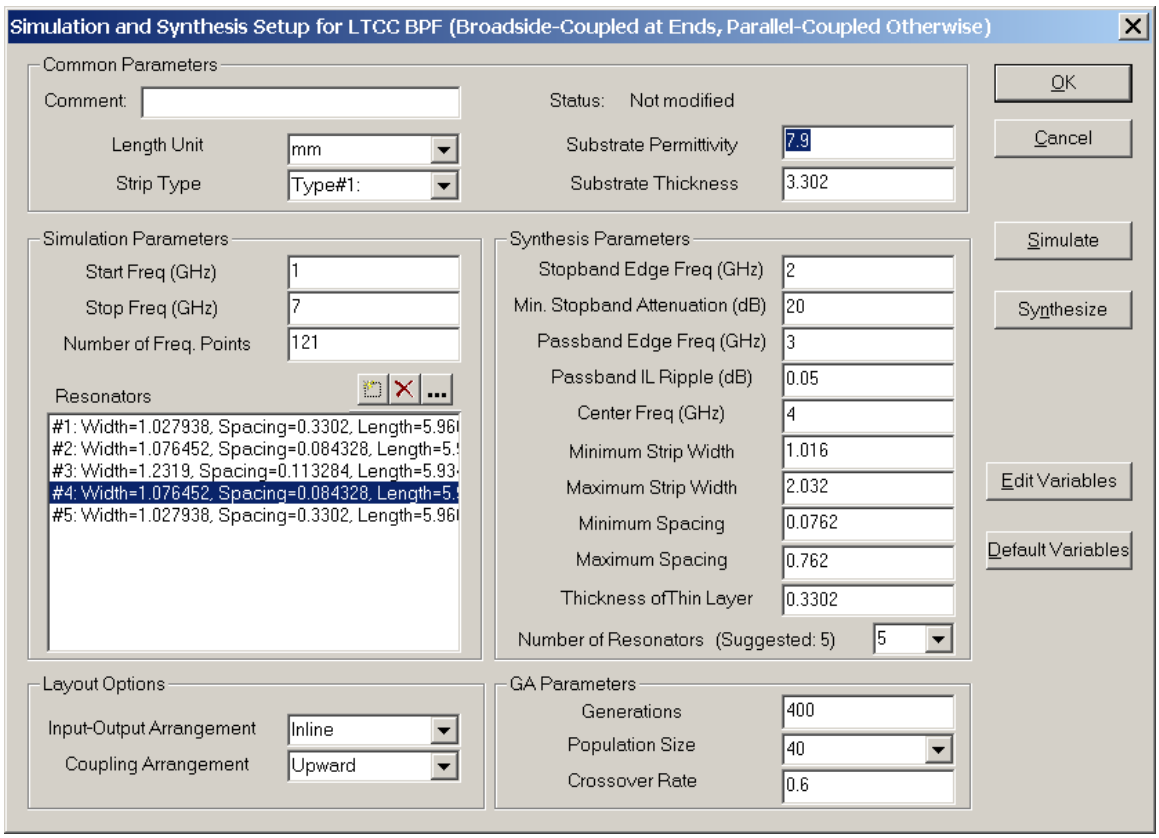


Figure AH.3 Simulation and synthesis setup for a LTCC bandpass filter.

After defining the specifications for the LTCC bandpass filter and choosing genetic algorithm parameters for the optimization, we simply click **Synthesize** to let FilterSyn do a quick synthesis. Once the equivalent circuit based synthesis is completed, synthesized dimensions of resonators will be shown in the list box and the simulated frequency response of the filter will be plotted on Modua.

With the equivalent circuit based synthesized dimensions as an initial design, we can now set up optimization variables and goals for IE3D and let IE3D produce an improved design.

After clicking … button above the resonator dimensions on the dialog in Figure AH.3, we will have a window like Figure AH.4 for us to define optimization variables.

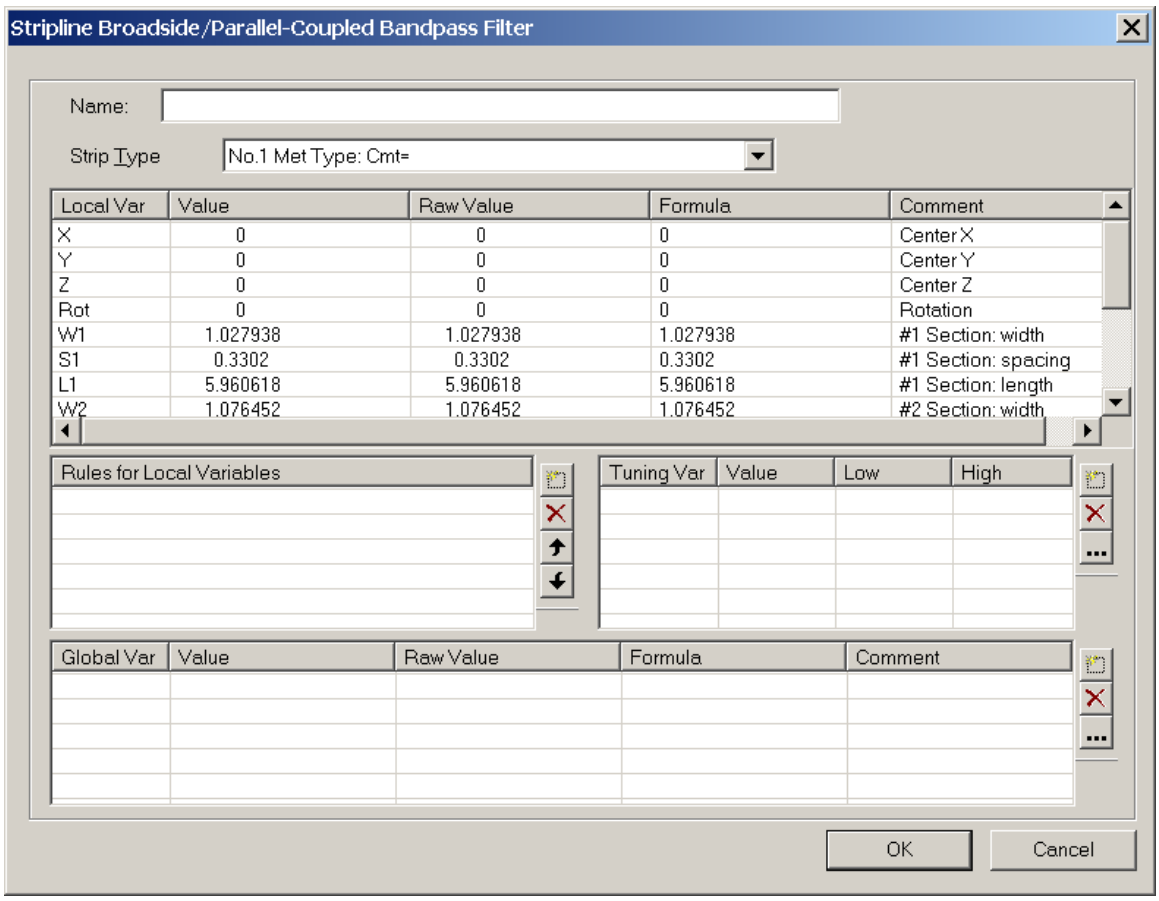


Figure AH.4 Defining tuning (optimization) variables.

Initially, the lower three tables in Figure AH.4 is empty, unless there were variables defined previously. If the tables are empty, we can click the small “Create” icon for creating a row. Otherwise, we can double click an existing row to make changes. Of course, we can also use the “Delete” icon to delete a row.

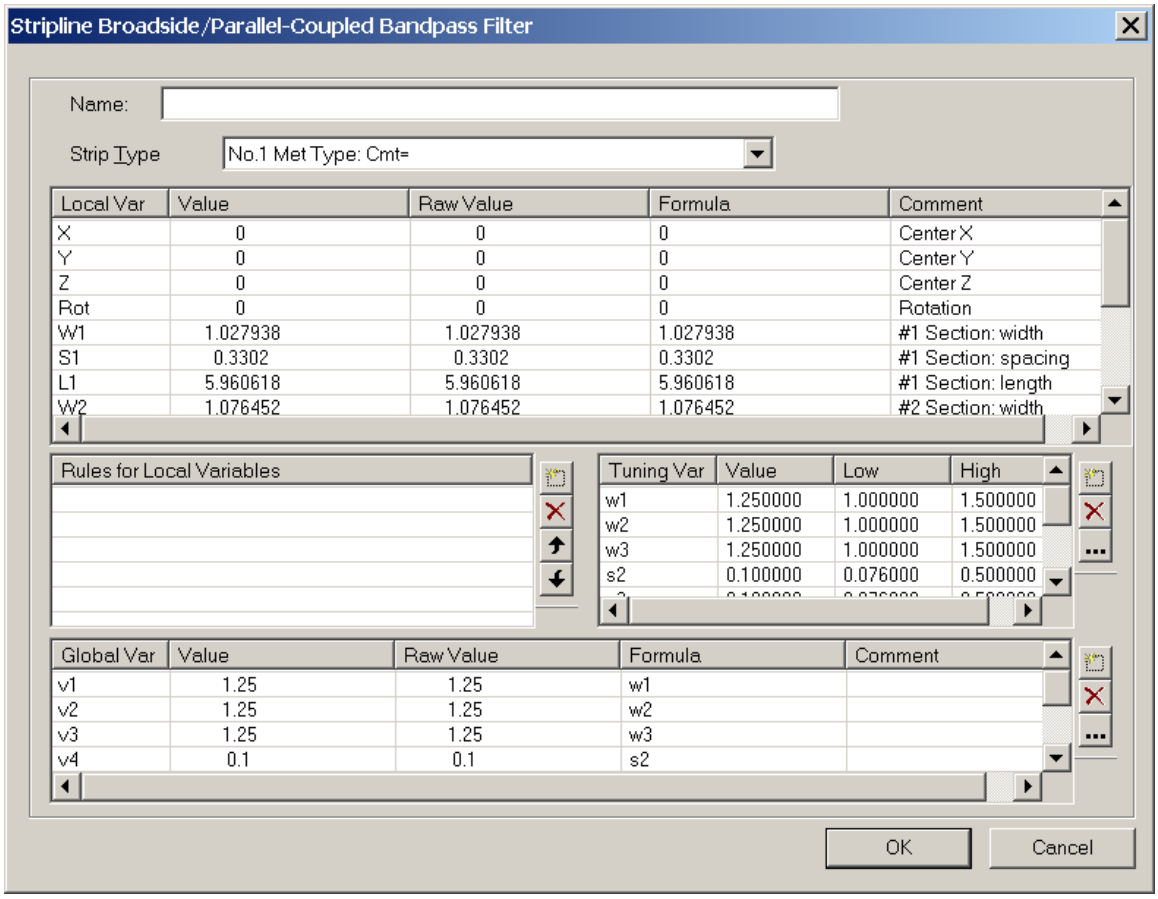


Figure AH.5 Tuning and global (optimization) variables defined.

We are now ready to generate the layout of the filter by clicking OK button in Figure AH.5 and then clicking **OK** in Figure AH.3 and **Create Lib File** button in a subsequent window. Once we click the Ie3dLibrary layout window, we will have a filter layout like Figure AH.6.

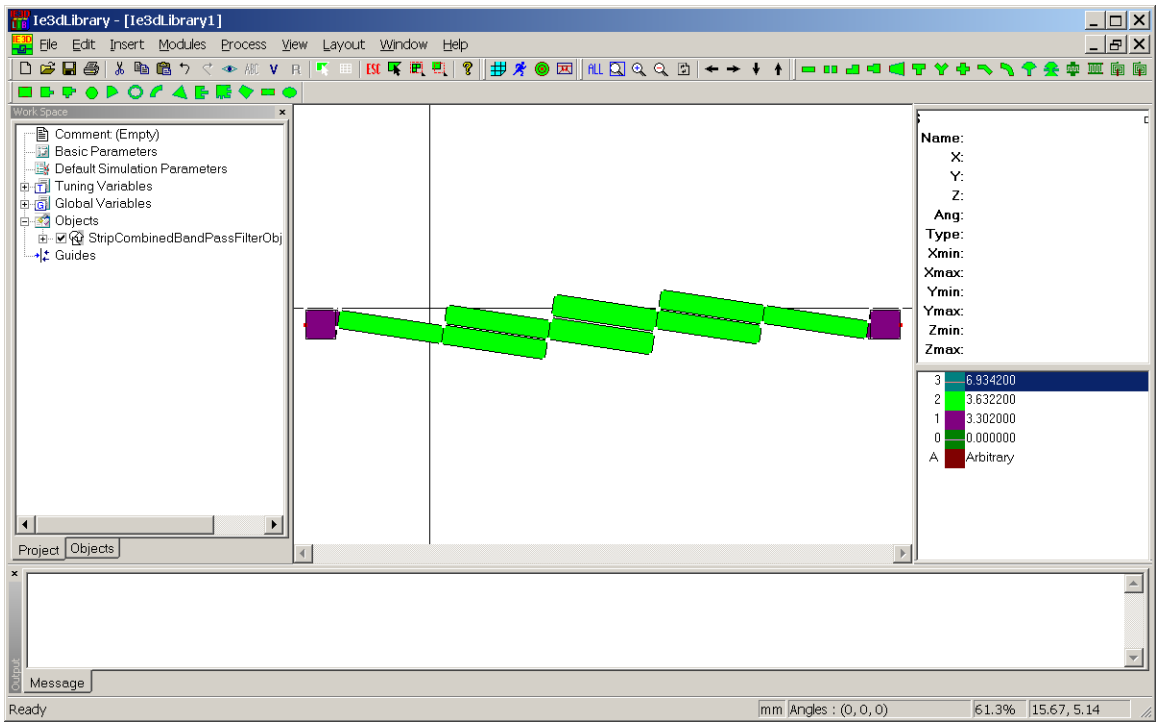


Figure AH.6 Filter layout generated by FilterSyn.

Before we optimize the filter, we need to define input port and output port for the layout. To do this, click Insert menu and select Dependent Objects… and Extension Port… as shown in Figure AH.7.

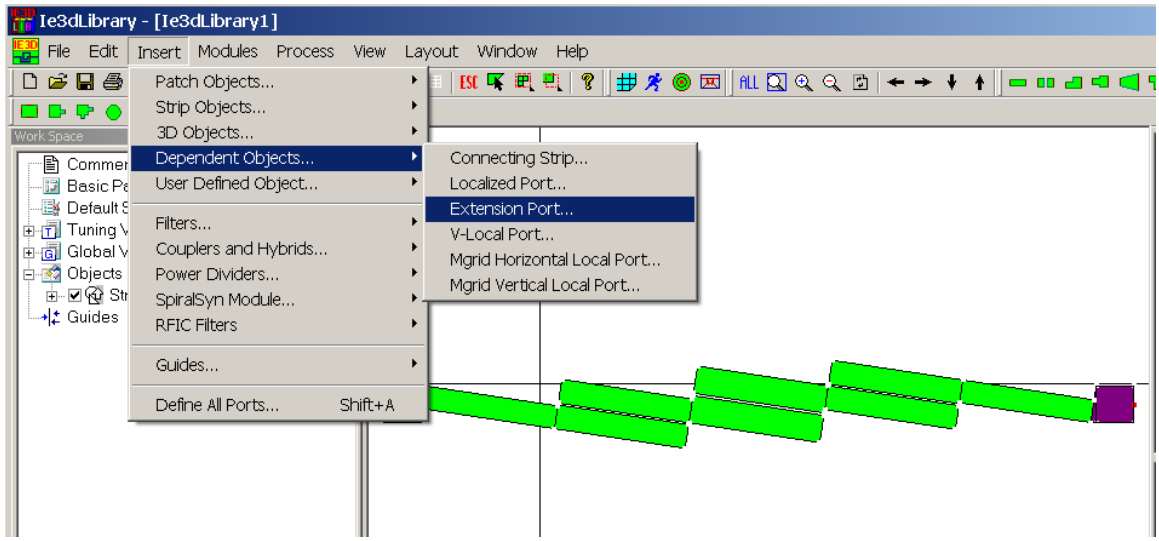


Figure AH.7 Defining ports for the filter.

After defining the ports and save the project as LTCC_BPF.ie3, we can now set up the optimization by clicking Process menu and select Optimize as in Figure AH.8.

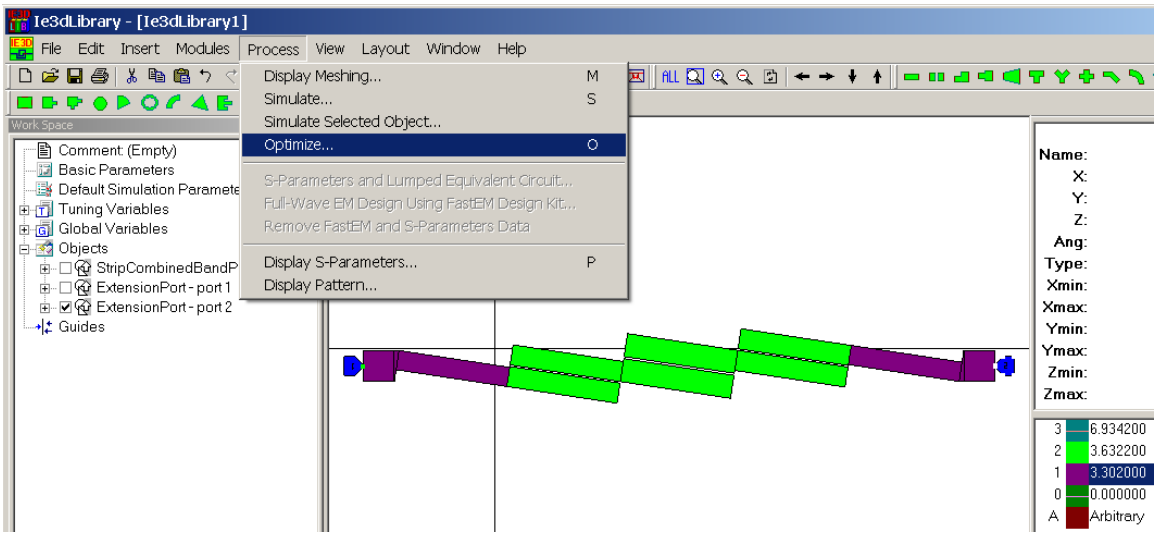


Figure AH.8 Setting up the optimization.

The optimization setup dialog is shown in Figure AH.9. In that dialog, 21 frequency points from 1 GHz to 7 GHz are set, SMSi solver is chosen, optimization goals involving |s21| and |s11| are defined, and Adaptive EM Optimizer is selected. Clicking OK button will start the optimization. Upon the completion of the optimization, an optimized filter layout will be created in a .geo file with the same name as the project but with "m" appended. It is noted that the time the optimization takes depends on the number of frequency points and unknowns, type of matrix solver, optimization goals, and optimization algorithm. For the current problem, it took about 15 minutes on a Pentium 4 PC with 1.5 GHz clock frequency.

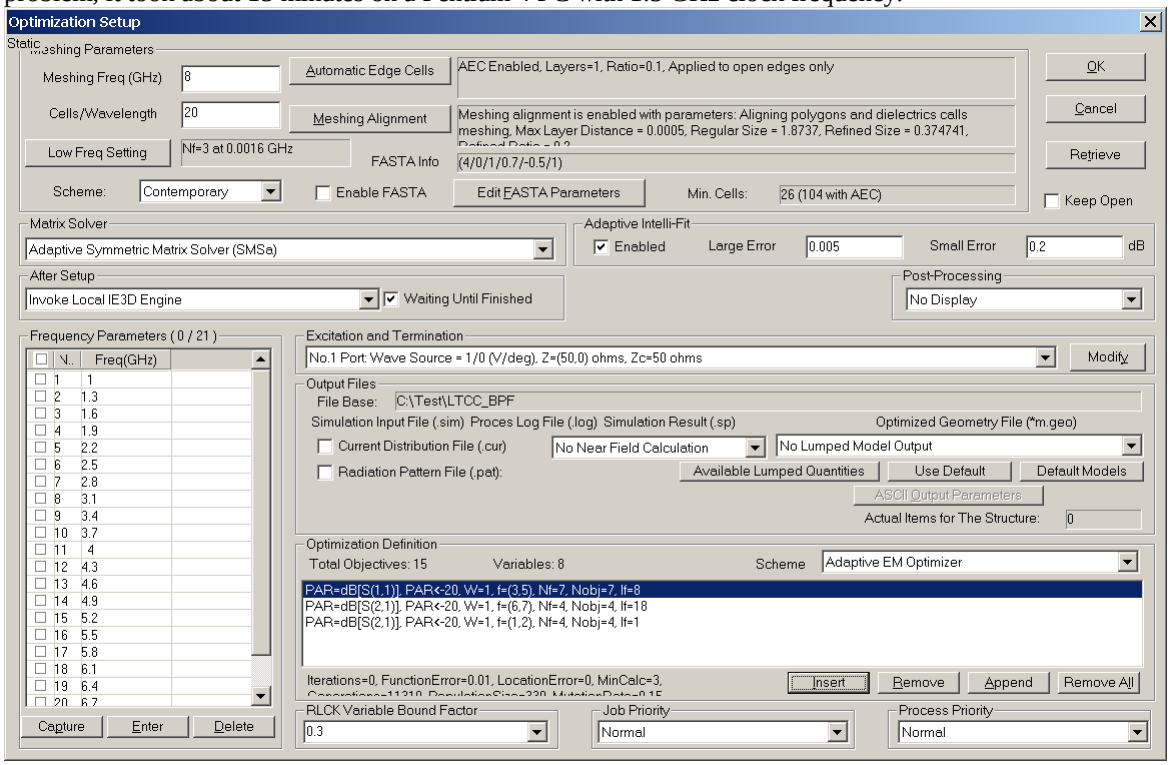


Figure AH.9 Optimization setup dialog.

When the optimization is complete, a dialog like Figure AH. 10 pops up.

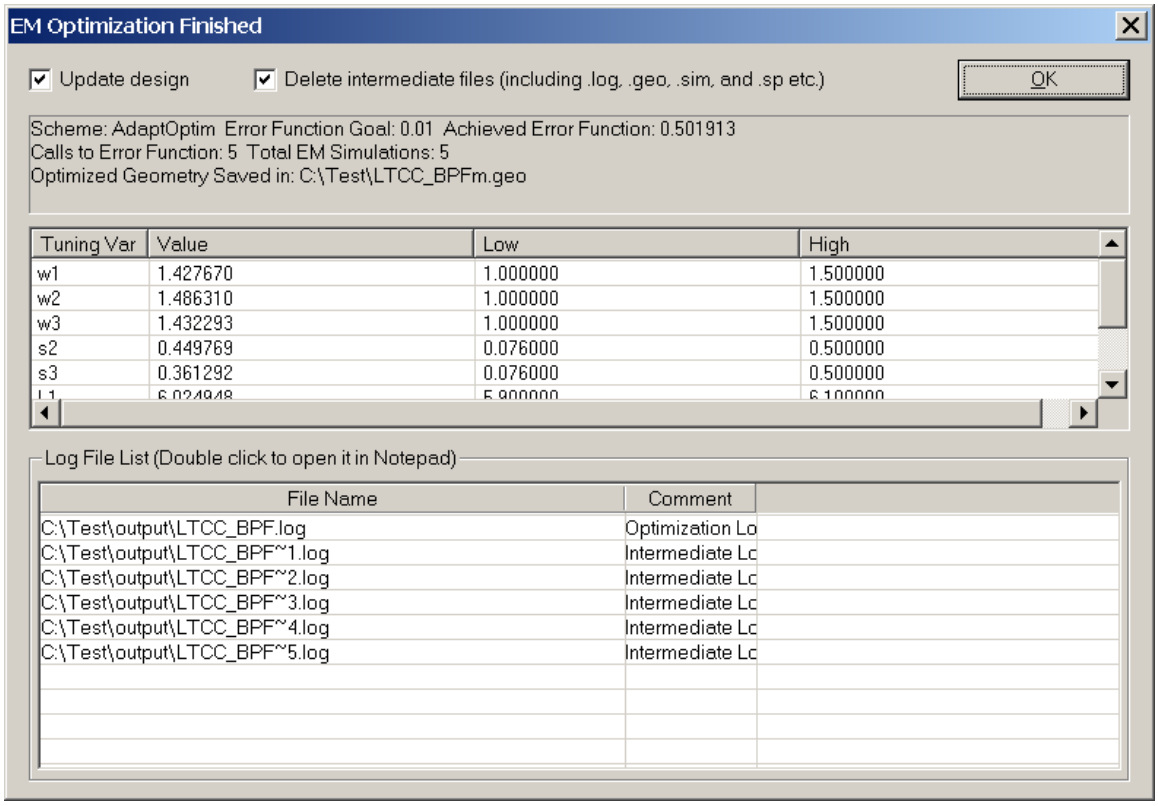


Figure AH.10 Optimization finish dialog.

The performance of the initial LTCC bandpass filter before the IE3D optimization is compared with that of the final filter after the optimization in Figure AH.11. We see clearly the improvement on the return loss in the passband and the bandwidth against the specifications.

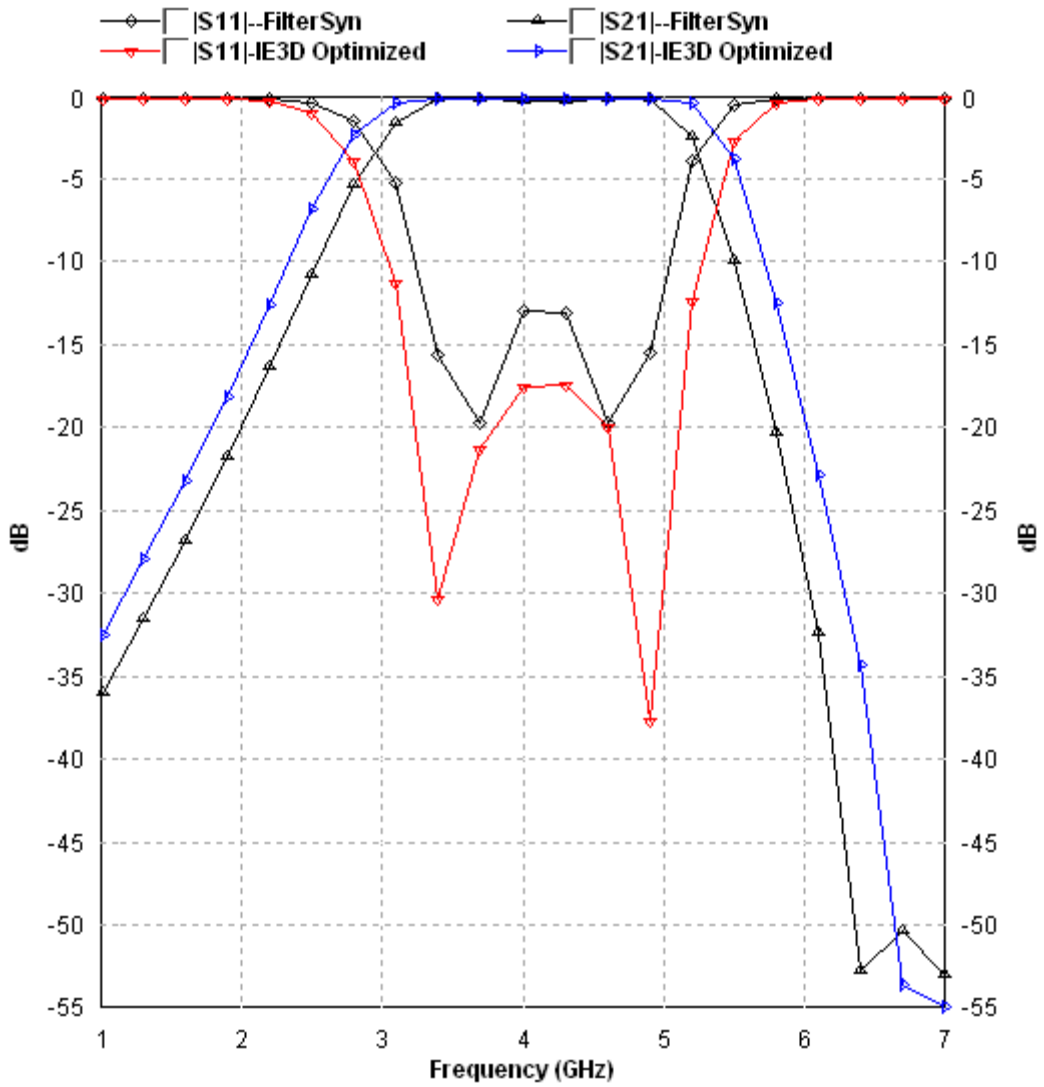


Figure AH.11 Comparison of the performance of a LTCC BPF before and after IE3D optimization.

Appendix AI. Special Notes on Modeling Spiral Inductors

Spiral inductors are frequently encountered in MMIC, RFIC and other types of applications. Accurate modeling of spiral inductors is very critical in the design of different high frequency IC circuits. Many advanced schemes have been implemented into the IE3D to guarantee high accuracy simulation on spiral inductors. However, we do need to take some special steps in order to get high accuracy results.

A typical spiral inductor in RFIC is small in size. For example, the size of a typical spiral inductor is about 300 microns by 300 microns. There are normally a few turns. The current is changing quite rapidly along the turns. If we use the normal rule of thumb for meshing, or 20 cells per wavelength, we may not be able to achieve very high accuracy. In fact, the fundamental rule for high accuracy meshing is that we should try to mesh the structure at those regions where current change very fast. For a spiral inductor, its size is very small; however, its current is changing quite fast. We may need to use finer meshing to achieve high accuracy.

Saved in: .\ie3d\samples\rect_spiral1.geo is an example. The structure is shown in Figure AI.1. The rectangular spiral is about 250 by 250 microns. It is on 2 substrates: 100 micron thick with permittivity = 100 microns and 0.2 micron thick with permittivity = 4. The trace width is 10 microns and the gap width is 5 microns. In practical cases, you may have lossy substrate with significant conductivity. We would like to simulate the spiral up to 10 GHz.

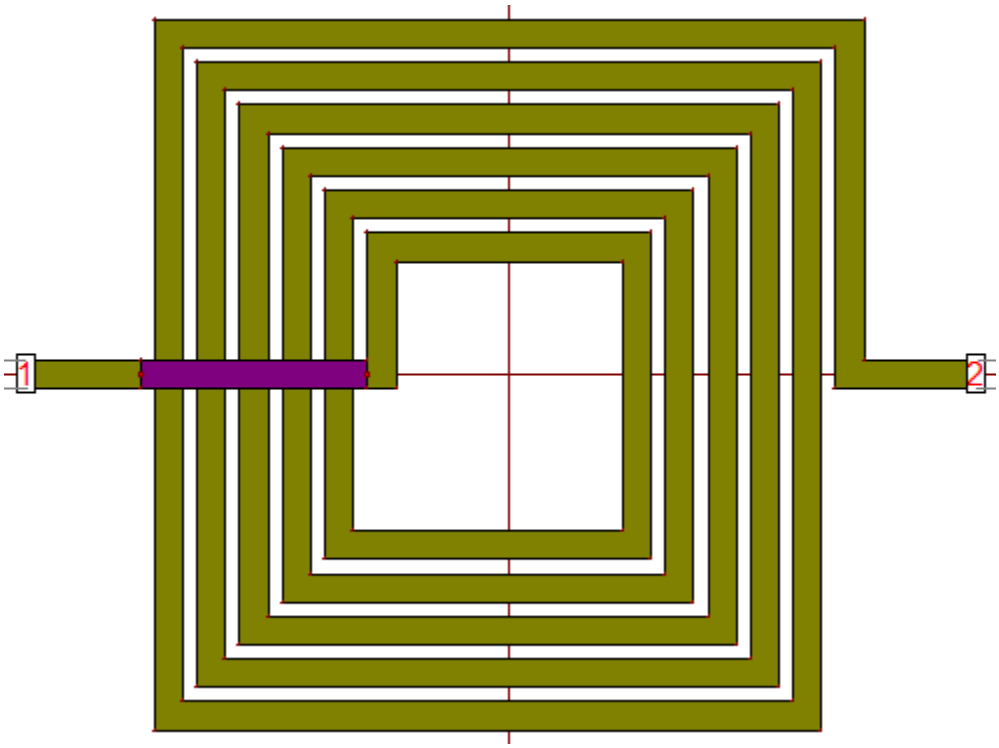


Figure AI.1 A rectangular spiral inductor.

IE3D is well implemented to handle spirals on semi-conductor substrates and thin substrates. However, we do need to pay attention to the meshing. For the rect_spiral1.geo, we use Fmax = 10 GHz and Ncells = 15 cells / wavelength. We disable the Automatic Edge Cells (AEC). The simulation result is shown in Figure AI.2. As it is, the result is not bad. It is quite accurate. However, you are reminded to check the meshing every time you simulate spiral inductor. Please select Process->Display Meshing and select OK to accept the default values (Fmax = 10 GHz and Ncells = 15). We will get the meshing result shown in Figure

AI.3 (a). Then, please mesh it again with the Fmax to 50 GHz. We will get the meshing result shown in Figure AI.3 (b). Change the Fmax = 100 GHz and we will get the meshing result shown in Figure AI.3 (c).

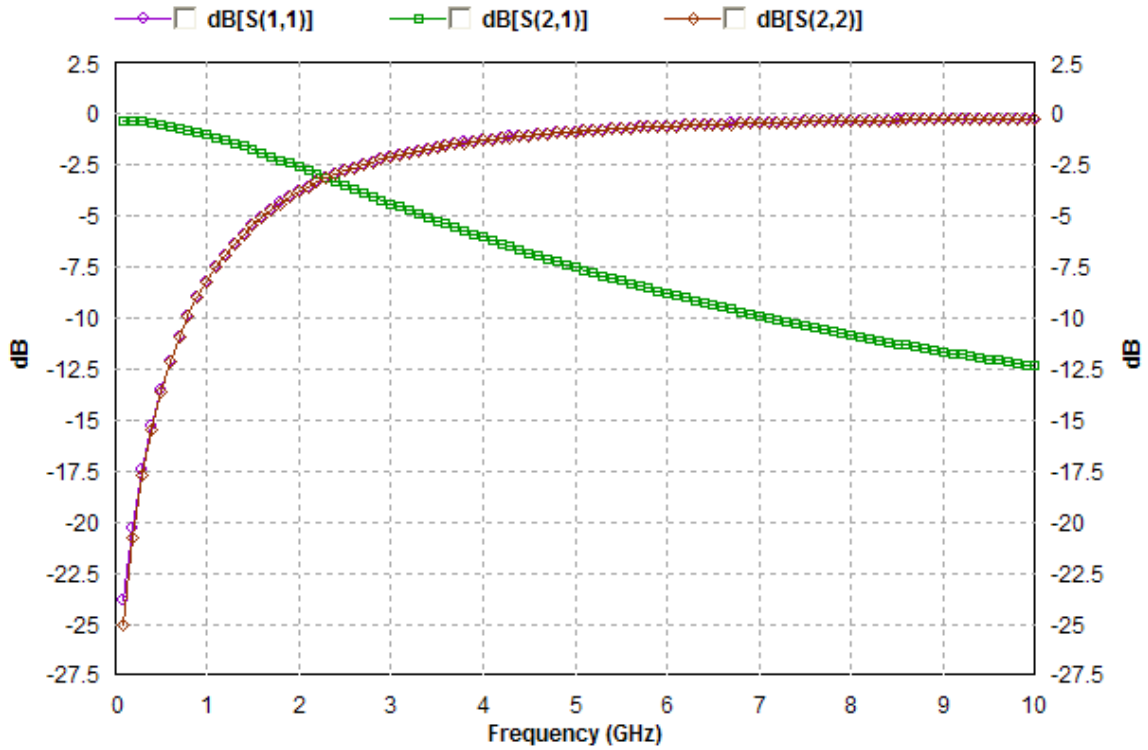
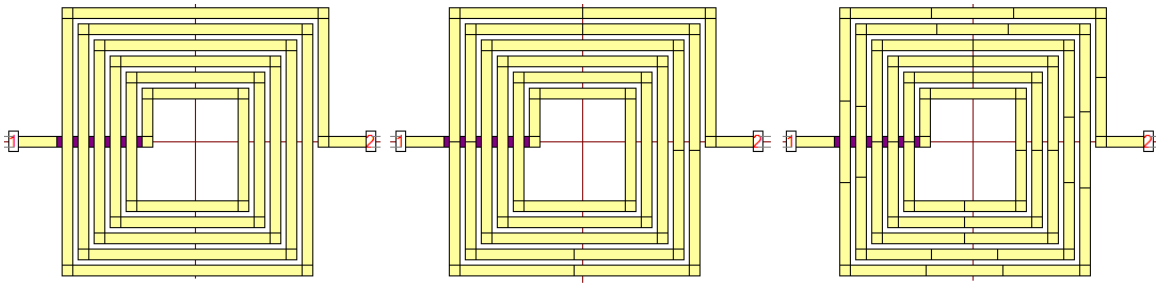


Figure AI.2 The results of rect_spiral1.geo with Fmax = 10 GHz and Ncells = 15 and no AEC.



(a) Fmax = 10 GHz (b) Fmax = 50 GHz (c) Fmax = 100 GHz

Figure AI.3 The meshed spiral with different Fmax values.

As you can see, the Fmax = 10 GHz yield one cell on each quarter of the turn. Normally, such a meshing may not be fine enough. For the Fmax = 50 GHz case, we start seeing that each segment (quarter a turn) for the outer turns is meshed into 2 cells. Such a meshing density is ok. For the Fmax = 100 GHz, we see that the inner most segment is meshed into 2 cells. From our experience, when we start seeing the inner most segment is meshed into 2 cells, we should normally get quite converged results. This should be the rule of thumb for the meshing of spiral inductor.

As we increase the meshing density, the extension arm of a port is reducing in the length. When the length of the extension arm is too short, it may also affect the simulation accuracy. Normally, we would like to make sure the extension arm length to be at least two substrate thickness. For the current structure, the substrate is about 100 micron thick. The outer segment of the spiral is divided into 3 cells when the length of the segment is about 250 microns. The cell size should be about 80 micron long. By the default, the port

extension is 5 cells and it is about 300 microns long. It should be long enough. We do not need to add extra cells for the port extension. If it is necessary, you can select Port->Change Scheme menu and select the Port Extension button to set a higher value for all the ports. Certainly, you can change individual port's port extension by double clicking the port listed in the dialog of Change Scheme menu.

The structure with Fmax = 100 GHz is saved in c:\ie3d\samples\rect_spiral2.geo. The simulation results of different meshing schemes: (1) Fmax = 10 GHz, No AEC in rect_spiral1.geo; (2) Fmax = 100 GHz, No AEC in rect_spiral2.geo; (3) Fmax = 10 GHz, AEC in rect_spiral1_aec.geo; (4) Fmax = 100 GHz, AEC in rect_spiral2_aec.geo are compared in Figure AI.3.

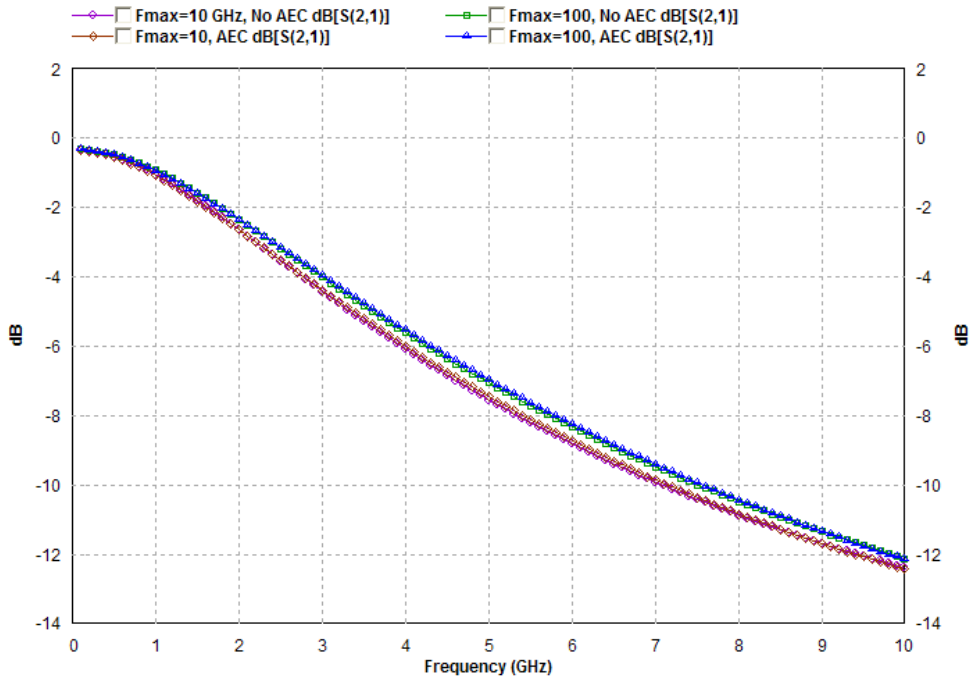
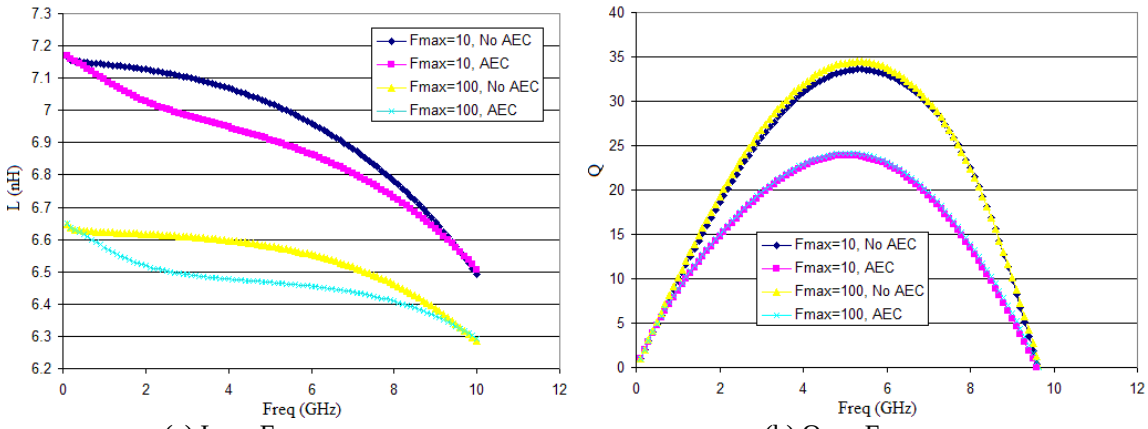


Figure AI.3 The comparison of results using different meshing.



(a) L vs. Frequency (b) Q vs. Frequency

Figure AI.4 The L-values and Q-values vs. frequency for different meshing schemes.

As you can see, there is some difference between the Fmax = 10 GHz and Fmax = 100 GHz cases. However, the difference in the s-parameters between AEC and NO AEC is very small. It is almost invisible from the graph. This does not mean that AEC is not important in modeling spiral inductors. Shown in

Figure AI.4 are the L vs. frequency and Q vs. frequency plots. Given a 2-port s-parameters, we can get the L and Q values vs. frequency using the Process->PI-Network Equivalent command on MODUA. The Q-value is defined as: Q = - Im[Y(1,1)] / Re[Y(1,1)] in this command.

It is interesting to see the following: (1) AEC predicts lower Q-values. In the other word, without AEC, a simulation normally under estimates the loss. (2) The L-values do not differ much with and without AEC. (3) The L-value is sensitive to the meshing density while the Q-values are not sensitive to the meshing density as long as AEC is enabled. (4) Even though the s-parameters with and without AEC are almost the same, the Q-values may differ significantly.

Trace thickness may also be very important. For our particular example, the strip thickness is 2 microns, compared to trace width of 10 microns and gap width of 5 microns. The strip thickness is not too small. For the rect_spiral1.geo and rect_spiral2.geo, the traces are modeled as infinitely thin strip with the loss effect of thickness included. However, the structure effect is not included. In order to get more accurate results, we need to use the "Grow Thickness" model for precise modeling of thickness. Saved in: c:\ie3d\samples\rect_spiral3.geo is the spiral inductor with thickness precisely modeled. Before we build the thickness using the command Edit->Layers->Grow Metal Thickness on Layer or the command Edit->Grow Metal Thickness on Polygons, it is good to do the Adv Edit->Rectanglization command first. The Rectanglization command allows a user to divide the polygons into more rectangular shapes. The Minimum Division Size in the command is a critical parameter (see Figure AI.5). It controls how MGRID is dividing the polygons for rectangles. For our particular example, the trace width is 10 mils. We can enter Minimum Division Size = 9 for it. If we enter the Minimum Division Size larger than 10, the polygons may not divided into rectangles as good (see Figure AI.6). Interested users can give it a try to see the difference.

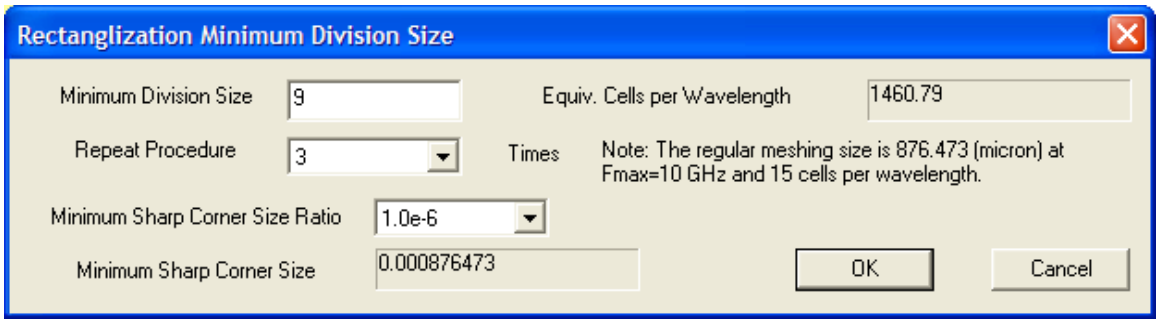
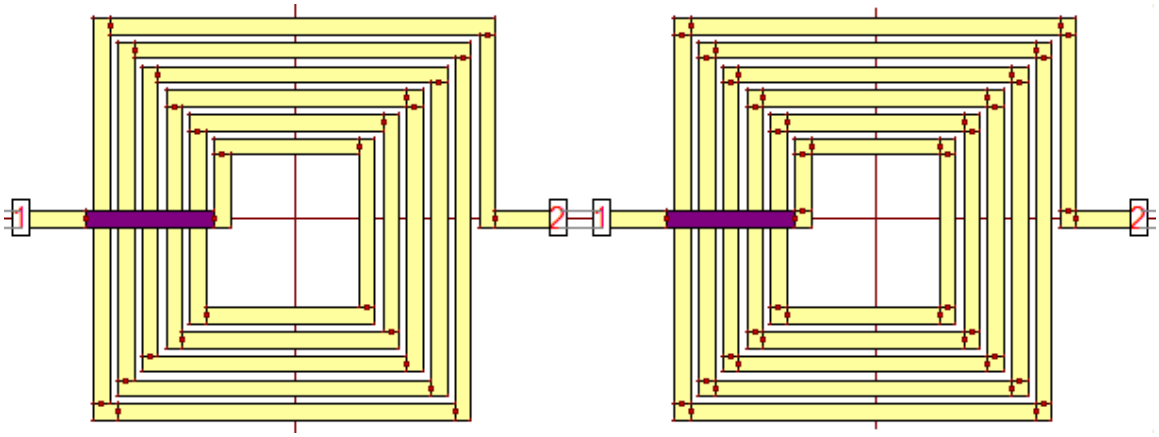


Figure AI. 5 The Adv Edit->Rectanglization dialog.



(a) Minimum Division Size = 11 (b) Minimum Division Size = 9

Figure AI.6 The different results with different Minimum Division Size.

As it is, the Rectanglization command is automatically executed when we choose the Surface Impedance Correction Scheme as, "2nd Degree Compensation with Pre-Processing" (see Figure AI.7). The Minimum Division Size is automatically chosen. For this particular example, the Minimum Division Size is automatically chosen to be larger than 10. The end result is ok. However, the meshing program may create more (triangular) cells and it may slow down the simulation. For the rect_spiral3.geo file, we did the Adv Edit->Rectanglization with Minimum Division Size = 9. Then, we grow the thickness on the Z = 100.2 mil layer where the traces are on, and the Z = 104.2 mil layer where the air-bridge is on.

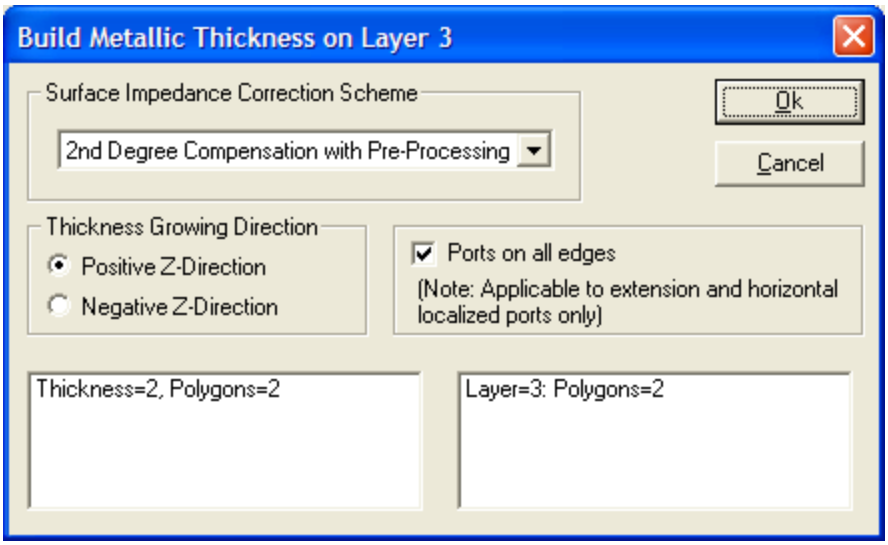


Figure AI.7 The Edit->Layers->Grow Metal Thickness on Layer dialog.

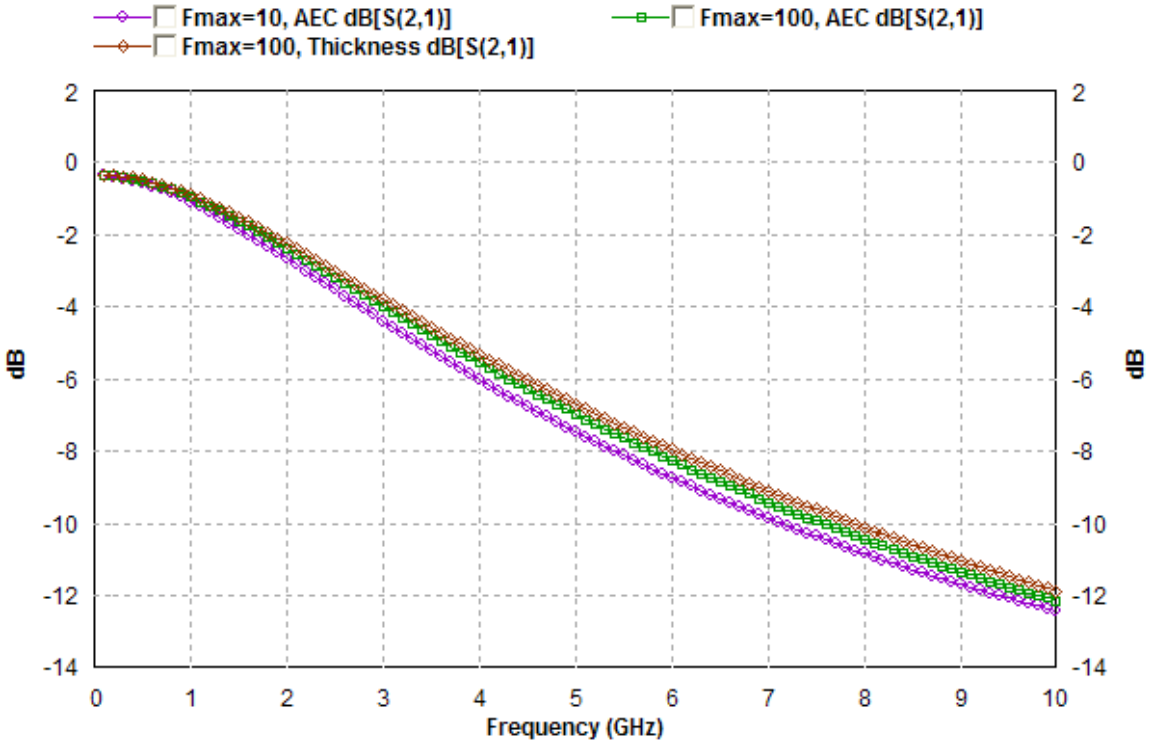
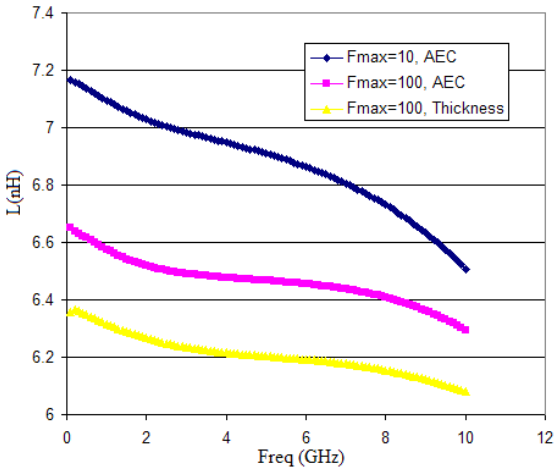
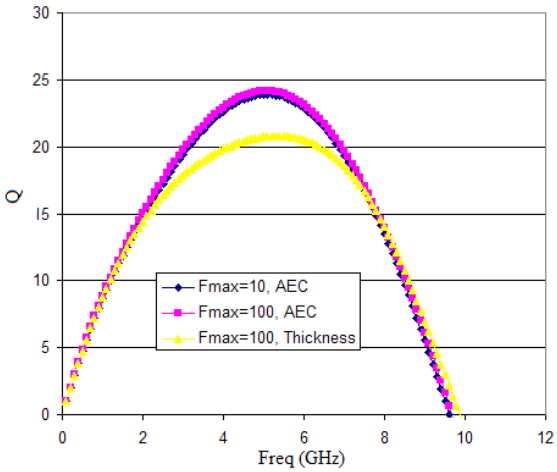


Figure AI.8 The s-parameters of different meshings with AEC and thickness.

Figure AI.8 shows the comparison in s-parameters among 3 different cases: (1) Fmax=10, AEC for rect_spiral1_aec.geo; (2) Fmax=100, AEC for rect_spiral2_aec.geo; (3) Fmax=100, Thickness for rect_spiral3.geo. As you can see, there is little difference among the 3 cases. No AEC is applied for the thickness model or rect_spiral3.geo case. It is not necessary to use edge cells because the thickness model automatically takes care of the edge effects.



(a) L vs. Frequency



(b) Q vs. Frequency

The L and Q-values vs. frequency is shown in Figure AI.9. As you can see, the thickness model predicts small L values and Q values. Normally, the thickness model should be of the best accuracy.

Appendix AJ. Finding Transfer Function Between TX and RX Antennas

In the IE3D 10.2, we have implemented 2 very important features for pattern handling: (1) The File->Save Detail Formatted Pattern Data on PatternView; (2) The Edit->Find Tx Rx Transfer Function in PATTERNVIEW.

We have noticed that quite some users are interested in the format of the .pat files created by MGRID/IE3D. As .pat files contain much data and they are not well formatted, parsing a .pat file is not a simple task. For such a reason, we implemented the command File->Save Detail Formatted Pattern Data onto PatternView. The feature allows a user to access all the detailed pattern data (directivity, gain, axial ratio, etc at specified angles, etc.) in nicely formatted ASCII files.

Another important feature is finding Tx/Rx Transfer function. UWB antenna design has received much attention in recent years. UWB antenna design involves finding the transfer function between a Tx antenna and a RX antenna over a wide frequency range. Designers may also be interested in finding the waveform distortion after the transmission with specified excitation conditions at the Tx antenna and termination conditions at the Rx antenna. The Edit->find Tx Rx Transfer Function serves the purpose of finding the transfer function in frequency domain. To perform a waveform analysis, users are suggested to use the optional MDSPICE for robust time transient analysis. If you do not have the MDSPICE license, please contact us for it.

The Find Tx Rx Transfer Function involves 2 antennas. One is a Tx antenna and one is an Rx antenna. The requirements for such a function are the following: (1) The Tx antenna must be a 1-port antenna. (2) The Rx antenna must be a 1-port antenna with plane wave excitation saved in General Pattern (.mpa) format.

You can create a General Pattern (.mpa) file from the current distribution on MGRID . For the Tx Rx Transfer function, you can use the same antenna as the Tx and Rx antenna. For such a case, you can simulate the antenna with 1 port and plane wave excitation and find its current distribution. Then, you can use MGRID to open the .cur file. Select Pattern Calculation. After the Pattern Calculation is finished, MGRID will prompt you whether you want to save the General Pattern (.mpa file) or define the excitation for the pattern (.pat file) with specified excitation. You should select Save General Pattern to save it as the Rx antenna. If you want to use the same antenna as the Tx antenna, you can then select Define Excitation to define the excitation on the only port. You should not define any excitation for the E-plane and H-plane plane wave excitations. Otherwise, you will not be able to use the antenna as the Tx antenna. Certainly, you can use different antennas for Tx and Rx antennas as long as they meet the requirements mentioned above.

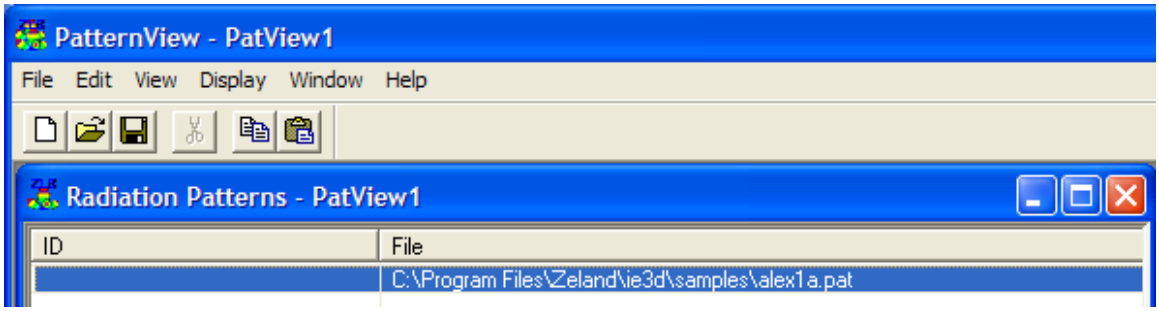


Figure AJ.1 The alex1a.pat file added into the list on PatternView.

After you create the .pat file for the Tx antenna and the .mpa file for the Rx antenna. You can run the PatternView for the Tx and Rx Transfer function. Assume we have the Tx antenna as: alex1a.pat (pattern with specified excitation at the port) and the Rx antennas as: alex1a.mpa (General Pattern). They are

created from the same geometry: alex1a.geo with 1-port and plane wave excitations. Please select Edit->Add Pattern in PatternView to add the alex1a.pat fie into the list (see Figure AJ.1).

Please select the file in the list. Then, select Edit->Find Tx Rx Transfer Function. PatternView will prompt you for the Rx antenna file name. Please select the file: alex1a.mpa. PatternView will load the .mpa file and prompt you a dialog similar to the Figure AJ.2. There are a few parameters you can change. When you change the parameters, many parameters in the dialog may be updated simultaneously. Shown in Figure AJ.2 is the dialog after we change the Elevation Angle to 40 degrees and the Azimuth Angle to 30 degrees.

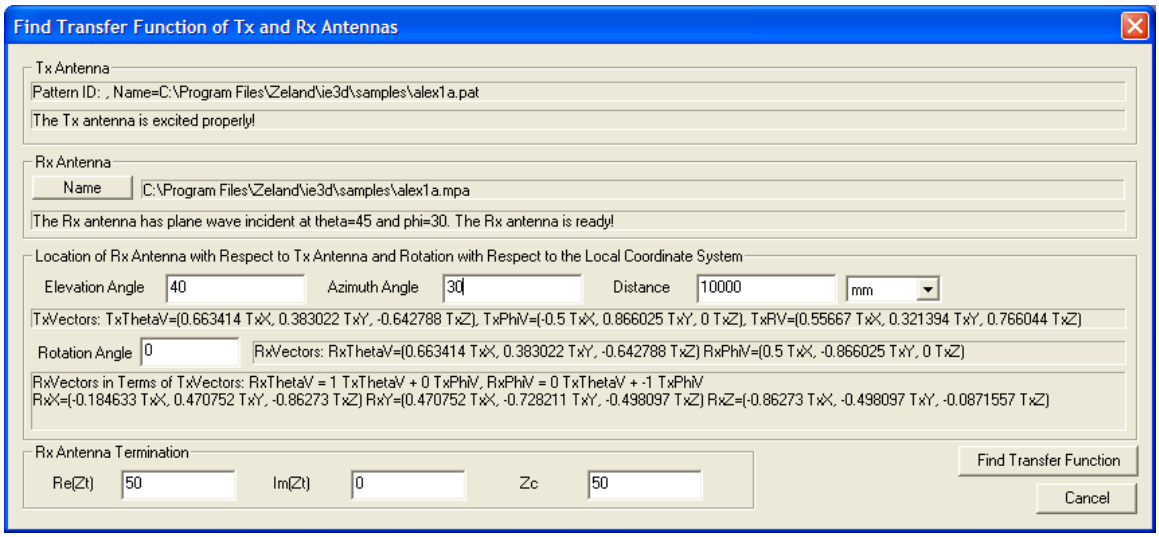


Figure AJ.2 The dialog for finding Transfer Function of Tx and Rx Antennas.

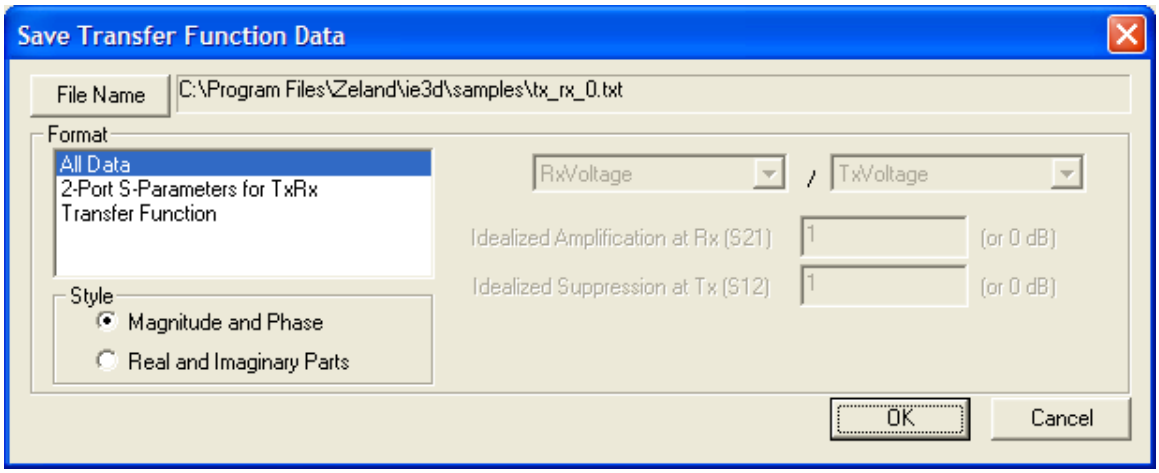


Figure AJ.3 The option for saving Transfer Function data.

After the parameters are specified, you can select Find Transfer Function button to find some specific transfer function of your interests (see Figure AJ.3). You have the option to save the 2-port s-parameters or a specific combination for the transfer function. You can find all the different combinations of transfer functions by selecting the All Data option. You can post-process the transfer function data using other software packages such as MS-Excel. The s-parameters are saved in Agilent/EEsof Touchstone format. You can perform further frequency domain analysis or display on MODUA or MDSPICE. In case you want to do waveform analysis, you should choose the MDSPICE for it. MDSPICE allows you to define a general waveform and see what the output waveform looks like. It also allows you to add other circuit elements

including non-linear elements into your design for time domain circuit simulation. Detail will not be demonstrated here.

There are many parameters involved in the Tx and Rx transfer function calculation (see Figure AJ.2). The meanings of the parameters are documented in Table AJ.1. They are very complicated and they are not easy to be illustrated. Please read their definitions carefully before you use them.

Table AJ.1 The different parameters involved in Tx and Rx function calculation

Elevation Angle Azimuth Angle Distances	The parameters define the location of the Rx antenna with respect to the Tx antenna in the spherical coordinate system.
TxX, TxY, TxZ	TxX is the unit x-vector, TxY is the unit y-vector and TxZ is the unit z-vector in the Tx coordinate system.
TxThetaV, TxPhiV, TxRV	TxThetaV is the unit theta vector, TxPhiV is the unit phi vector, TxRV is the unit R vector at the location of the Rx antenna.
RxThetaV, RxPhiV	RxThetaV is the unit theta vector and RxPhiV is the unit phi vector in the Rx antenna's coordinate system when it is looking at the Tx antenna.
Rotation Angle	It is related to how the Rx antenna is rotated against the theta angle and the phi angle of the plane wave. For the Rx antenna, at which angles the plane wave is coming in is already defined in the geometry. The user can not change it. However, the polarization of the plane wave can be changed in finding Tx and Rx transfer function. The Rotation Angle defines the angle of the Tx theta vector (TxThetaV) in the local plane formed by the 2 vectors: RxThetaV and –RxPhiV. If Rotation Angle is 0, the TxThetaV is at the same direction as the RxThetaV. If Rotation Angle is 30 degrees, then the TxThetaV and the RxThetaV will form an angle of 30 degrees while the TxThetaV and the RxPhiV will form an angle of 120 degrees.
RxX, RxY, RxZ	The x, y, and z-unit vectors in the Rx antenna's local coordinate system.

The Rx antenna termination will affect the transfer functions only. It will not affect the s-parameters between the Tx and Rx antennas.

Appendix AK. Transformation from Near Fields to Far Fields

The capability of computing far-zone fields and pattern properties from near-zone fields on the surfaces of a box is added to PatternView for IE3D. The box can be arbitrarily shaped, but it should enclose all excitation sources and induced currents on relevant geometry for a specific problem. The XML file format and "mnf" extension should be used to create a file for storing data of near-zone fields on the surfaces of a box. When a *.mnf file is loaded into PatternView using the command "Near Field to Far Field Transformation" in the menu "Edit" and numbers of elevation theta and azimuth phi angles are entered in the next-coming popup dialog, far-zone fields and pattern properties will be computed and stored in a *.pat file.

To create a near-field (*.mnf) file, the following information is needed: length unit, port excitation, operating frequency, multilayer substrate, near–field nodes and near –field cells. A sample file is created at: .\ie3d\samples\XmlFileFormatForDataOFNearFields.mnf.

A length unit should be provided using the attribute - UnitName of the tag <NearFieldNodesAndCells>. Any following of length units can be used in the near-field file: mm, mil, micron, m, inch, foot, and cm. For example, if mm is selected to the length unit, UnitName="mm" should be put in the section of attributes of the tag <NearFieldNodesAndCells>.

It is assumed that only one port is used for source excitation. The port voltage, port current, characteristic impedance and source impedance are needed to compute some pattern properties such as radiation efficiency and gain. Magnitude and phase (in degree) of port voltage and port current, and real and imaginary parts of characteristic impedance and source impedance should be provided using the attributes - Mag_V, Ang_V, Mag_I, Ang_I, Re_zc, Im_zc, Re_zt, Im_zt of the tag <NearFieldNodesAndCells>.

The operating frequency in the unit of GHz should be provided using the attribute - Frequency of the tag <NearFieldNodesAndCells>.

A multilayer planar substrate should be provided using the tag <Substrates>. The easiest way to provide the substrate information correctly is to create a geo file in The XML file format for the specific substrate using Mgrid and to copy the part for the tag <Substrates> in the geo file to a near-field (*.mnf) file. Note that the free space can be represented by using same dielectric properties in all substrate layers.

A near–field node is represented by a position and electric and magnetic fields at the position. The x, y and z coordinates of the position vector must be given in terms of the above-specified length unit. Electric and magnetic vector fields have three x, y and z components that are complex numbers. Each node must be assigned to a unique node number. Numbers of all near–field nodes start at zero and are consecutive. All the information for all near–field nodes should be provided using the tag <NearFieldNodes>. Each row for the tag consists of 16 numbers (first is an integer and others are real) and is used to represent a near–field node. The meaning of each number in a row is as follows:

The 1st number is node number

2nd x coordinate of the position
3rd y coordinate of the position
4th z coordinate of the position

5th	magnitude (V/m) of x component of electric field
6th	phase (degree) of x component of electric field
7th	magnitude (V /m) of y component of electric field
8th	phase (degree) of y component of electric field
9th	magnitude (V /m) of z component of electric field
10th	phase (degree) of z component of electric field
11th	magnitude (A/m) of x component of magnetic field
12th	phase (degree) of x component of magnetic field
13th	magnitude (A /m) of y component of magnetic field
14th	phase (degree) of y component of magnetic field
15th	magnitude (A /m) of z component of magnetic field
16th	phase (degree) of z component of magnetic field

Note that the node number is always 0 for the first row and n -1 for row n. As required in IE3D, z coordinate of all nodes must be greater than or equal to zero.

A near–field cell consists of 3 or 4 near–field nodes and is used to represent a surface element on the surfaces of the box. The nodes of a near –field cell are located at corners of the surface element. The order of local node numbers of nodes of a near–field cell must be in the counterclockwise sense with respect to the outward normal of the corresponding surface. A near–field cell is represented by 3 or 4 global node numbers of the nodes on it. Each cell must be assigned to a unique cell number. Numbers of all near–field cells start at zero and are consecutive. All the information for all near–field cells should be provided using the tag <NearFieldCells>. Each row for the tag consists of 4 or 5 integer numbers and is used to represent a near–field cell. The meaning of each number in a row is as follows:

The 1st number is cell number

2nd	global node number of the 1st local node
3rd	global node number of the 2nd local node
4th	global node number of the 3rd local node
5th	global node number of the 4th local node if it exists

Cell number must be always 0 for the first row and n -1 for row n. Near–field cells should not be overlapped. If near-fields are insignificant on some parts of the surface of the box, near–field nodes and cells on these parts do not need to be created.

Appendix AL Contemporary Meshing Scheme and Meshing Parameters

The IE3D meshing scheme is improved significantly in the IE3D 11. A new meshing scheme "Contemporary" is implemented in the IE3D 11. The meshing scheme used before IE3D 11 is renamed as "Classical" meshing scheme. In fact, the Classical meshing scheme in IE3D 11 is not completely the same as the IE3D 10 meshing scheme. Some of the new improvement is integrated into the Classical meshing scheme. We provide the Automatic Meshing Parameters dialog in the Basic Parameters dialog (at button Automatic Edge Cells) for users to change many of the controls for meshing. The Automatic Meshing Parameters dialog is shown in Figure AL.1.

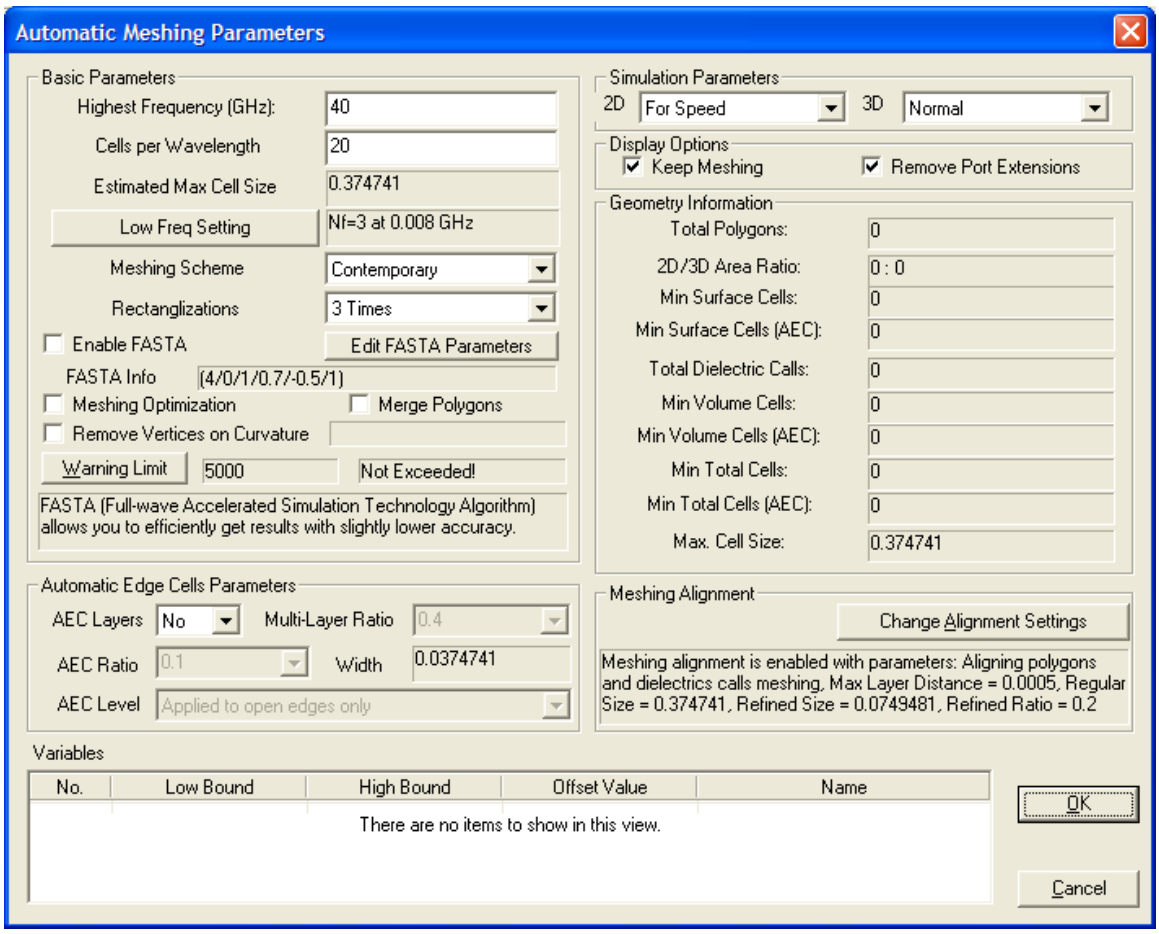


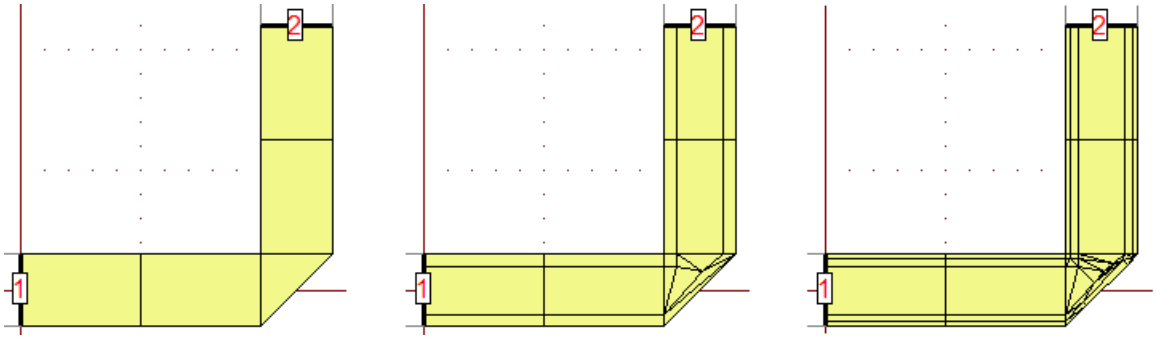
Figure AL.1 The Automatic Meshing Parameters dialog in Basic Parameters dialog.

There are a few new and important features implemented into the Contemporary meshing scheme.

1. Multi-Layer Automatic Edge Cells (AEC) and AEC Level:

If you have tried IE3D, you will know that edge cells are very important in yielding high accuracy EM simulation results. IE3D has implemented the Automatic Edge Cells (AEC) many years ago. Enabling AEC will allow IE3D to yield results of guaranteed accuracy for most structures. AEC automatically adds one layer of narrow cells along the open edges of a structure (see Figure AL.2). The narrow cells on the open edge will allow IE3D to accurately model the edge current singularity effects (see Figure 1.4). When we mention "open edge" above, it means a boundary of a polygon which is not shared with other polygons.

For some structures with extremely high accuracy requirement, you may want to try more than one layer of edge cells. The Classical meshing scheme does not support multi-layer AEC. Manual operation is required for the IE3D 10 to create multiple layers of edge cells, and the process is tedious. For example, some user has reported the following situation. When he tries to design a directional coupler with -20 dB coupling, the user is able to get single pass design using the AEC in IE3D 10. When he tries to design -40 dB coupler, he can no longer get single pass design using the AEC in IE3D 10. He has to go for multiple iterations. However, if he manually creates 2 layers of edge cells, he will be able to get single pass and reduce the design cycle.



(a) No edge cells (b) 1 layer edge cells (AEC Layers=1) (c) 2 layer edge cells (AEC Layers=2)

Figure AL.2 The comparison of different settings for AEC layers.

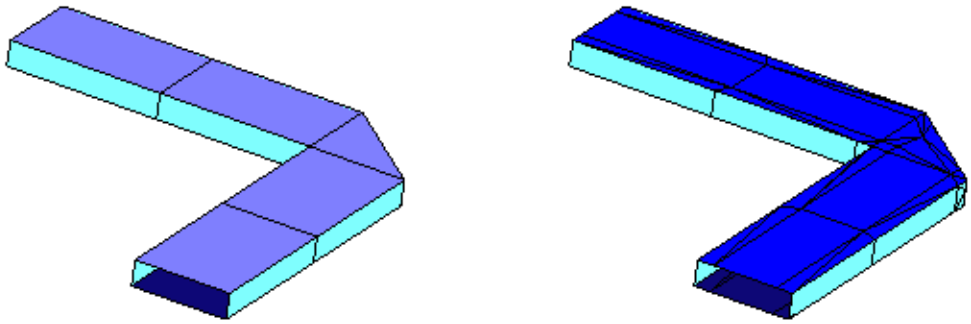
For the above reason, we have implemented multi-layer AEC. You can select "AEC Layers: No" for no AEC (see Figure AL.2a), "AEC Layers: 1" for 1-layer of AEC (see Figure AL.2b), and "AEC Layers: 2" for 2-layers of AEC (see Figure AL.2c), and so on. Normally, the more layers you define, you will get higher accuracy. However, you will create more unknowns and the simulation time will be much slower. From our experience, No AEC is normally ok for isolated trace without strong coupling if you do not need very high accuracy. One layer of AEC normally can guarantee high accuracy for most structures. Two or more layers of AEC should only be used when you require extreme accuracy. For most structures, you really do not need more than 1 layer of AEC. Please note multi-level AEC is only supported when you choose Contemporary meshing scheme.

Enabling AEC is only one aspect of the story. Another important factor is the "AEC Width" or the "Edge Width". Before IE3D 11, we require a user to define the AEC Width as an absolute value. The suggested value is about 5-20% of the regular cell size. When the AEC Width is too big, it will not improve the accuracy. When it is too small, it may not have the proper effect either. We noticed that quite some users do not know how to define the AEC Width even though IE3D does give them warning if the defined AEC Width is in some extreme range. For this reason, we have changed IE3D 11 to allow a user to define the AEC Ratio instead of AEC Width. AEC Ratio is the ratio between the AEC Width and the regular cell size. By default, the AEC Ratio is 0.1. It corresponds to that the AEC Width is 10% of the regular cell size. Using AEC Ratio will make it easier for novice users to have the correct AEC Width with the default value.

Another important parameter for multi-layer AEC is the Multi-Layer Ratio for AEC. The default value is 0.4. If you define AEC Layers = 2 and AEC Ratio = 0.1 and Multi-Layer Ratio = 0.4, you will end up with the following parameters: Assuming the regular cell size is 0.3 mm. Then, the 1st layer AEC Width is about 10% of 0.3 mm or 0.03 mm. The 2nd layer AEC Width is 0.4 of the 1st layer AEC Width or 0.012 mm. Multi-Layer Ratio is only used when AEC Layers is larger than 1.

Another AEC introduction is the AEC Level (see Figure AL.1). This is related to AEC for a structure with thickness grown. When you build thickness to a trace, IE3D will have polygons for the 4 sides of a trace for accurate modeling of strip thickness (see Figure AL.3). The vertical edges of a thickness trace normally are able to include the edge singularity effects. It is normally not necessary to add edge cells when

a trace is modeled with thickness option on the IE3D (growing thickness on polygons). For this reason, we do not add edge cells on the edges of thickness polygons in IE3D 10. For the Contemporary meshing scheme in IE3D 11, we provide an option to users. You can choose "AEC Level: Applied to open edges only" and "AEC Level: Applied to thickness edges too" (see Figure AL.3). The "AEC Level: Applied to thickness edges too" normally can further improve accuracy. However, unless it is necessary, we do not suggest users to enable it because it will increase the problem size significantly. Nevertheless, there are cases you may want to enable it. When the strip thickness is very small and you grow thickness, the vertical edge polygons become very narrow. In this case, the vertical edge polygons may not be enough to capture all the edge effects. In fact, such a case normally should no happen because you should use the thin strip model when the thickness is too small. Another reason to enable "AEC Level: Applied to thickness edges too" is for the case you want to have higher accuracy results.



(a) AEC Level: Applied to open edges only (b) AEC Level: Applied to thickness edges too.

Figure AL.3 The meshed thickness trace with different AEC Level settings.

2. The settings for "Rectanglization":

"Rectanglization" has been implemented into MGRID as a command to clean structures. It will try to break the polygons into as many rectangles as possible. IE3D uses non-uniform mixed rectangular and triangular meshing. Rectangular cells are more efficient than triangular cells. Normally, each rectangular cell is at least equivalent to two triangular cells. For the same structure, using rectangular cells can save at least 50% of unknowns. Saving 50% unknowns means that we can reduce the simulation time by a factor of 4 to 8. Therefore, if we can use as many rectangles as possible, we will be able to reduce the problem size significantly. The "Rectanglization" procedure tries to divide an irregular polygon some rectangular cells until it can not divide further. After the "Rectanglization" procedure, normally we can reduce the number of unknowns. Also, it may require multiple time of "rectanglization" in order to find all the possible rectangular shapes. You can specify how many "rectanglizations" you want IE3D to perform in the meshing.

IE3D tries to mesh the entered polygons into small rectangular cells and triangular cells. When multiple polygons put side by side, we have a rule to check whether they are electrically connected or disconnected. Whether two polygons are electrically connected or disconnected makes big difference. In case two polygons are overlapped together, IE3D 14 will try to merge them together automatically.

3. Meshing Alignment:

The general rule of thumb for meshing is that Ncells = 15-20 cells per waveguide wavelength at the highest frequency. However, this is not the fundamental rule of thumb. The fundamental rule of thumb is that we need to mesh the structure fine enough so that it can capture the change of current distribution on a structure. Also, depending upon what parameters we focus on, we will have different accuracy

requirements. For example, we need to have much finer meshing for modeling spiral inductors in RFIC and MMIC. They are normally electrically small. However, the current changes very fast in a spiral even its electrical size is small. Also, we are interested in the L and Q values of a spiral inductor. The numerical errors involved in L and Q are much more sensitive to meshing than in s-parameters. For this reason, we normally need to increase the meshing density for spirals in RFIC and MMIC.

Another kind of structures we need to pay attention for special meshing is MIM capacitors. An MIM capacitor consists of two metallic plates with a very small separation by an insulation layer. The thickness of the layer is normally about 0.1 to 0.2 microns. When two plates are separated by such a small distance, there is a strong capacitive coupling between them. The current changes very fast in a small region of the MIM capacitor. We normally need to use finer and aligned meshing between the coupled plates in order to obtain high accuracy in the extracted C value.

Another situation requiring meshing alignment is the case of finite dielectrics. When we define some finite dielectric blocks and there are polygons on the surface of the finite dielectrics, we want the meshing of the finite dielectrics and the polygons are matching each other in order to obtain the best accuracy.

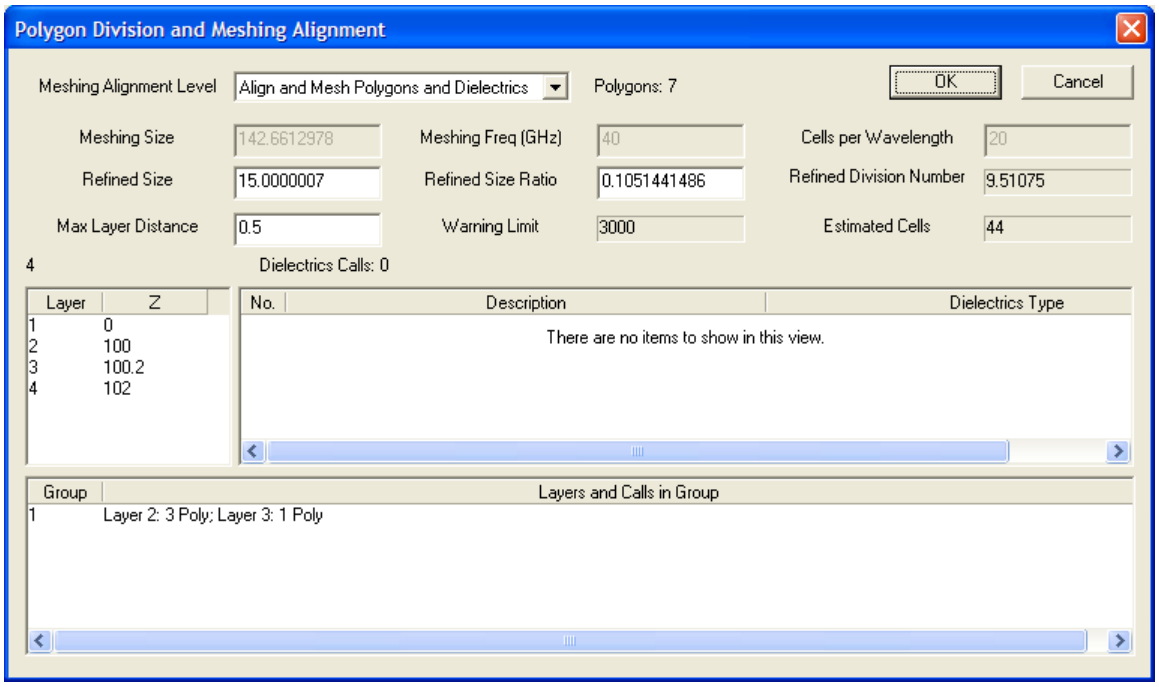


Figure AL.4 The meshing alignment control parameters.

The meshing alignment control parameters can be defined in Basic Parameters, Adv Edit->Display Meshing command, and the Simulation Setup dialog. Figure AL.4 shows a typical example in .\ie3d\samples\mimcap1_refined.geo. You have a few options in Meshing Alignment Level. Table AL.1 documents the different situation.

Table AL.1 The different options for Meshing Alignment Level.

Option	Description
No Alignment	No alignment is performed in the meshing.
Align Polygon Divisions	Just align the division lines of closed coupled polygons. We will not align the meshing. The meshing can be different even with same divisions due to some other consideration. This option is not useful. It is kept just as an

	option.
Align Polygons and Dielectrics Divisions	Align the division lines of closed coupled polygons and finite dielectrics. There is no alignment for the meshing. It is similar the last option.
Align and Mesh Polygons	Align the divisions and meshing of closed coupled polygons.
Align and Mesh Polygons and Dielectrics	Align the divisions and meshing of closed coupled polygons and finite dielectrics. This should be the option to use for alignment.

The Max Layer Distance = 0.5 (micron) means that when two layers of polygons are within 0.5 microns, we will consider they are tightly coupled. We will align the meshing between them.

Refined Size Ratio = 0.105 means that we will sub-divide the tightly coupled portion of the polygons into smaller cells about 10.5% of the regular cell size. The Refined Size = 15 microns while the regular Meshing Size is 142.67 microns. We can change either the Refined Size or the Refined Size Ratio while the other parameters will be updated.

There is more information in the Figure AL.4. The Layer list box shows the layers in the structure. The Dielectrics Calls list box shows the finite dielectrics here. There are none for this particular example. The list box at the bottom shows what MGRID is detecting. "Layer 2: 3 Poly; Layer 3: 1 Poly". It means that three polygons on layer 2 are tightly coupled with one polygon on layer 3. On the meshing, IE3D will align the divisions of the tightly coupled polygons and sub-divide them into small cells about 10.5% of the regular cell size.

The Meshing Alignment feature on IE3D allows us to align the meshing between closely coupled plates and the meshing between polygons and the finite dielectrics where the polygons are sticking to.

Appendix AM. Discussion on Different De-Embedding Schemes

Six different s-parameter extraction or de-embedding schemes are implemented into the IE3D. They are for different purposes. Ports are the most difficult to understand issues in an EM simulation. This appendix concentrates on the different port schemes and the scopes of their applications.

The ports in IE3D are divided into two groups: (1) Extension ports; (2) Localized ports. Extension ports come from the concepts of traditional microwave network theory.

A traditional microwave network is shown in Figure AM.1. A microwave circuit consists of a number of discontinuities connected by uniform TLN segments. The circuit normally meets the following conditions: (1) There is no coupling between any two discontinuities through other routes than the TLNs; (2) Each TLN segment can support only one propagation mode; (3) Every TLN segment is long enough so that the higher order evanescent modes are decayed so much there is virtually no coupling from the higher order modes between two discontinuities through the TLN.

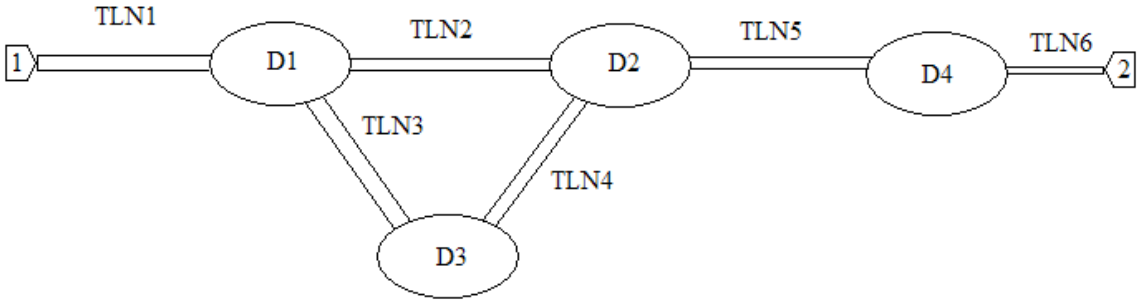


Figure AM.1 The illustration of a typical microwave network.

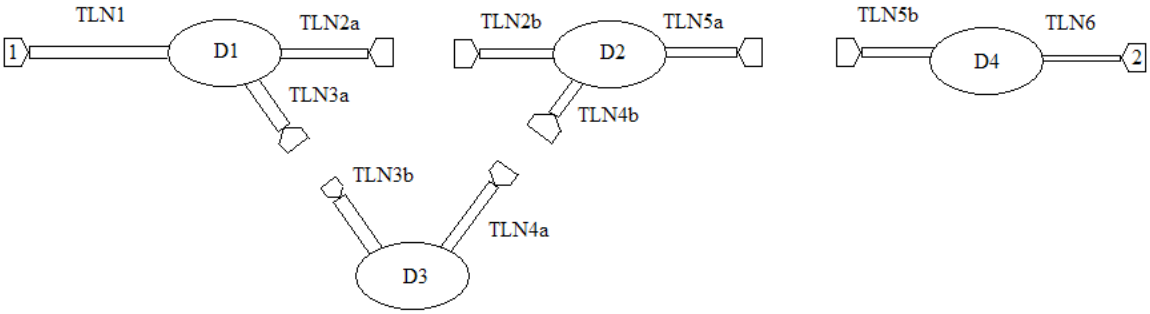
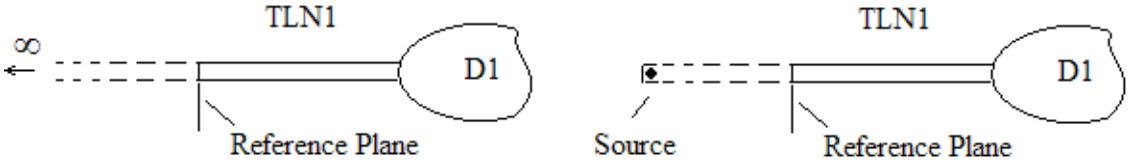


Figure AM.2 The illustration of cascaded sub-circuits.

Under the above conditions, we are able to divide the microwave network into multiple sub-circuits. Each sub-circuit contains a discontinuities and some TLN segments (see Figure AM.2). Each sub-circuit can be modeled as the s-parameters of the sub-circuit. By cascading the s-parameters of the sub-circuits together, we can yield the s-parameters of the complete circuit. In case the conditions mentioned above are met, the final s-parameters will be precise.

For each sub-circuit in Figure AM.2, we assume every TLN segment is infinitely long with the reference plane at the division plane. When we simulate a sub-circuit, we assume an incident wave of the fundamental mode is propagating along the TLN toward the discontinuities. It will hit the discontinuities. Part of the wave will be reflected back. Part of it will propagate to other parts of the sub-circuit. There might be many reflections and transmissions going on inside the sub-circuit. Eventually, it reaches a steady state with fixed reflection and transmission at each port. From this fixed reflection and transmission waves, we are able to find the s-parameters of the sub-circuit.

For each port, it should be a semi-infinite TLN based upon the definition (see Figure AM.3a). However, numerically, we can not model a semi-infinite long TLN. We have to truncate it so that it is a finite long TLN (see Figure AM.3b). At the end of the finite long TLN, we put some source there. The source will excite the standing waves along the TLN, from the standing waves, we are able to solve the s-parameters. As long as the source and the discontinuities are far from the detection point, we are able to have the standing waves of pure fundamental mode at the detection point. The solved s-parameters will be precise. On IE3D, an extension port basically is to emulate the finite long TLN excitation (see Figure AM.3b). The basic requirements for an extension port on IE3D are: (1) The port should be far enough from the discontinuities; (2) The de-embedding arm, or the distance from the port location to the source, should be long enough. The port location is where we detect the standing waves. Basically, we want the detection point to be free of higher order modes from the discontinuities and the source.



(a) Semi-infinite TLN extension (b) Truncated finite length TLN extension

Figure AM.3 An extension port with infinite extension and finite extension.

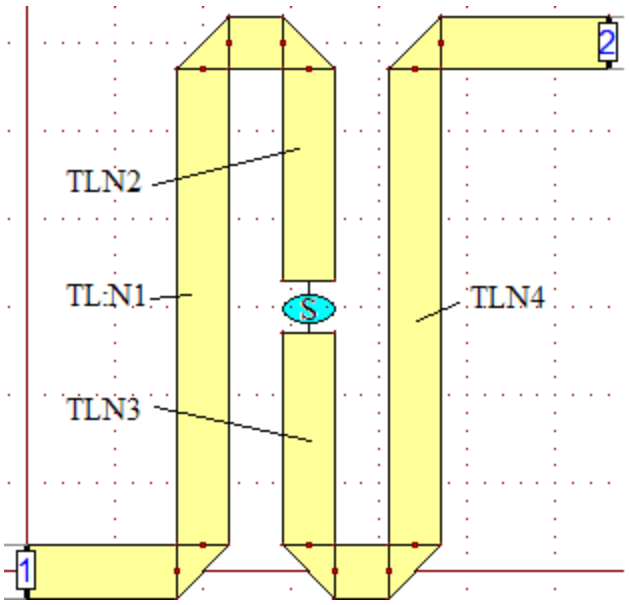


Figure AM.4 A planar structure with a lumped element.

The above discussion is on the origin of extension ports. Then, what is the origin of localized ports? The first localized port on IE3D, the Localized for MMIC, comes from the fact that we want to simulate a planar structure with a lumped element inside. A typical example is the the meander line structure discussed in Chapter 7. We want to connect a 2-port s-parameter file in between the gap of the meander line. Unlike traditional waveguide circuits, the TLNs are coupled strongly together because they are open TLNs. We can not break it into sub-circuit and simulate it with good accuracy. We have to simulate it in one structure with the s-parameter lumped element integrated into a mixed EM and nodal simulation. In order to do the mixed EM and nodal simulation, we have to define ports on both sides of the gap. As you can see, there is no room for the port extension. Also, we can not avoid the port extensions coupled with other TLN segments. For this reason, we have introduced the Localized for MMIC scheme which does not require an extension. For a Localized for MMIC port, we excite the structure at where the port is located and measure the port quantity

at the location. The source will create some higher order modes and the higher order modes will affect the accuracy. However, it is the only way out when we want to simulate structures with lumped elements.

The Localized for MMIC port is normally for microwave circuits with a ground plane. The ground plane automatically serves as the ground reference (or return path) for the port. In practical applications, we may encounter cases with probe feed or gap source like excitations (see Figure AM.5). For this kind of excitation, if the size of the feeding proximity is small compared to wavelength, we do not need to worry about the detail of the port. For example, a coaxial fed antenna. The probe radius can be important. However, the outer radius of the coaxial is not important. We do not need to worry about the small detail. Another common feature of such a feeding structure is that it is always a differential type having a positive terminal and a negative terminal. For this kind of structures, we introduce the Vertical Localized Port scheme for structures with the excitation in Z-direction and the Horizontal Localized port scheme for excitation in the XY plane.

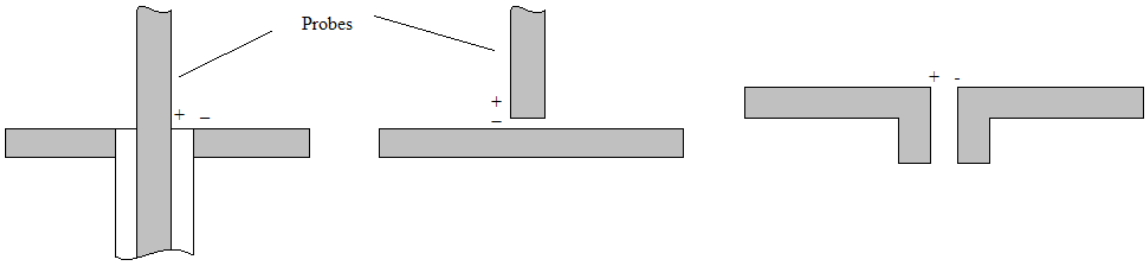


Figure AM.5 Different kinds of feeding structures may require localized ports.

Then, another question you may have is: what are those different kinds of extension ports? When we simulate a structure, we try to put some source at the end of the port extension (Figure AM.3b). It excites the standing waves on the TLN or the uniform transmission line. We measure the incident wave and reflected wave. Then, we will be able to solve the s-parameters based upon the incident wave and the reflected wave from the standard definition of s-parameters. In fact, it is what we do for the Extension for Waves scheme. We do not care about the characteristic impedance. We just try to find the incident wave and the reflected wave on the TLN and solve the s-parameters based upon them. This is the true definition of s-parameters and it is widely used in the academic area. We call the s-parameters extracted in this way the s-parameters normalized to port impedance.

However, in practical microwave circuit design, designers always want to use standard 50-ohm normalization. Where does 50-ohm normalization come is not known. We believe it is due to the fact that most measurement equipment are 50-ohm normalized systems. For measurement equipments, it is impossible to make it accepting the port impedance as the normalization impedance. When you do so, you would need the measurement equipment to have a probe matching the cross-section configuration of the TLN of each component you want to measure. If they do not match, the measured results will not be correct. Basically, there are numerous number of different cross-sectional configuration, we can not cover all of them. For this reason, most measurement equipments are normalized to the standard 50-ohms. We do not need to match the probe of the equipment to the port cross-section configuration. As long as their Zc is close to 50-ohms, the measured results should be reasonably accurate. There might be some higher order effects at the junction between the probe and the TLN of the device; some kinds of calibrations are performed to remove the effects. However, it is impossible to completely eliminate it. It is the best practical solution for it.

To match the measurement equipment, we have implemented the 50-ohms normalized ports: Advanced Extension, Extension for MMIC and 50-Ohms for Waves. Then, what are the differences between these different 50-ohm normalized extension port schemes. Their main differences are in how we calculate the characteristic impedance (Zc) of the port and how we find the port quantities.

For the 50-Ohms for Waves scheme, we solve the incident wave (Azc) and the reflected wave (Bzc) normalized to Zc. Then, we try to use some way to solve the Zc. We try to convert the Zc normalized incident wave and reflected wave into 50-ohm normalized incident wave (A50) and reflected wave (B50). Basically, we have:

$$V = (Azc + Bzc) \cdot \sqrt{Zc} = (A50 + B50) \cdot \sqrt{Z50} \quad \text{AM-1}$$
$$I = (Azc - Bzc) / \sqrt{Zc} = (A50 - B50) / \sqrt{Z50} \quad \text{AM-2}$$

Then, we solve the 50-ohm normalized s-parameters based upon the A50 and B50.

For the Advanced Extension scheme and the Extension for MMIC scheme, we are not solving the Azc and Bzc or even the A50 and B50 directly. We are solving the voltage V and current I on the TLN. The current I is a well defined quantity. It is the integration of current density over the cross-section of the TLN. What is the V?

For a TLN, we use V and I to characterize the energy flow along a TLN. The power it delivers is calculated as:

$$P = \tfrac{1}{2} V \cdot I^* \quad \text{AM-3}$$

Where I* is the conjugate of I. This formula comes from the Poynting vector formula. Electromagnetically, power flow is defined as:

$$P_z = \tfrac{1}{2} \int E \times H^* \, ds \quad \text{AM-4}$$

Where the integration is over the cross-section of the TLN. The *z* is the unit vector in the longitudinal direction.This formula is derived mathematically and it is precise. When the TLN is a pure TEM TLN, the field distribution in the cross-section is a conservative field. We can separate the integral over the cross-section into two parts:

$$P = \tfrac{1}{2} \int du \int dv \, E(u) \times H^*(v) \quad \text{AM-5}$$

Where the coordinate system (u, v) is some kind of orthogonal system with the u-coordinate matching the E-field's lines (E-field does not change with v on the lines) and the v-coordinate matching the H-field's lines (H-field does not change on the lines). The integration path for v should be a loop integral. We have,

$$V_{u'} = \int du \, E(u) \quad \text{AM-6}$$
$$I_{v'} = \int dv \, H^*(v) \quad \text{AM-7}$$
$$u' \times v' = z \quad \text{AM-8}$$

From AM-6 to AM-8, we can obtain AM-3. The above formulations are obtained under the condition of TEM mode.

For a non-TEM TLNs, the formulas AM-5 to AM-8 may not exist. In fact, the formulation AM-7 for current I is always correct even the loop integration is not along a path where H is independent of u, as long as the loop enclosing the positive conductor completely for a multiple conductor TLN system. Therefore, the quantity I is a certainly quantity, however the V in AM-3 is not a known quantity if we still represent P in terms of V and I in AM-3 for a non-TEM TLN.

If we enforce the P, V and I by equation AM-3, it means that we are finding a "suitable" integration path. The integral in AM-6 represents the voltage of the TLN along the "suitable" integration path. Then, the equation is: where is this "suitable" integration path? The answer is that it is some of the path from the

positive conductor to the negative conductor. However, no body knows where it is. In fact, the situation is similar to the integration theorem,

$$\int dx\, f(x) = f(x') \, (b - a) \qquad \text{AM-9}$$

Where the integration is over (a, b), f(x) is a continuous function from a to b and x' is a point in (a, b). We know x' must be some point between a and b. However, we do not know it unless we find the f(x) completely.

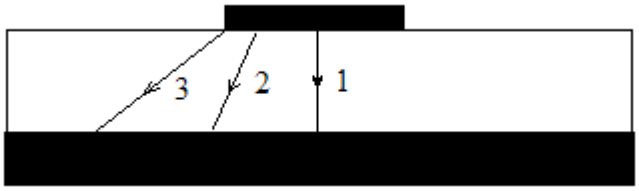


Figure AM.6 A microstrip cross-section

In some sense, in order to find the "suitable" integration path for the V, we have to find the E-field distribution in the cross-section completely. When we know the field distributions, we can solve the P directly from AM-4. Why should we need to worry about V?

For most of the cases, a typical non-TEM TLN such as a microstrip can still be considered as quasi-TEM mode. Basically, its field distribution on the cross-section is still quite close to conservative field at low frequency. The integrations of E along different paths may be different. However, the differences are quite small unless it is at very high frequency. The solved s-parameters are still consistent in some frequency range.

The fundamental difference between the Advanced Extension scheme and the Extension for MMIC scheme is in the integration path.

Extension for MMIC scheme was one of the first two extension ports (the other one is Extension for Waves) introduced in IE3D. It is a very versatile and robust scheme. It normally can go down to very low frequency and up to reasonably high frequency. However, we found that its accuracy may degrade very rapidly when it goes up to certainly frequency. A typical example is the 100-micron GaAs substrate microstrip line. Normally, the Extension for MMIC scheme can go up to 50-GHz. However, if you go beyond 50-GHz, you may see serious loss or even gain. It is not something wrong about the EM simulation part. It is more due to the fact that the power P calculated based upon the V and I in AM-3 is not correct. Basically, the V along the integration path for the Extension for MMIC scheme is getting farther and farther from the V for the "suitable" integration path when the circuit is getting more and more dispersive.

Some designers may push the frequency limit of 100-micron GaAs circuit up to 90 GHz before the 2^{nd} propagation mode happens somewhere between 90 and 100 GHz. They can not use the Extension for MMIC scheme if they go beyond 50-GHz.

To resolve the problem, we had introduced the 50-Ohms for Waves scheme. The 50-Ohms for Waves scheme can cover up to 90 GHz for a 100-micron GaAs circuit. However, 50-Ohms for Waves scheme can not go to very low frequency. At very low frequency, the current distribution does not vary much along the longitudinal direction. It is hard to extract the incident and reflected waves from the almost flat standing waves. Another issue with wave de-embedding schemes (Extension for Waves and 50-Ohms for Waves) is that it does require a long de-embedding arm. Even with the long de-embedding arm, it is easier to be disturbed by coupling. In some sense, the wave schemes are less stable than the ones using voltage and current concepts.

Another issue with wave de-embedding scheme is related to lossy TLNs. Wave schemes may not be complete precise with lossy TLNs. We will discuss about the topic later after we finish this topic.

With the Extension for MMIC, we can cover from very low frequency to not-very-high frequency. With the 50-Ohms for Waves scheme, we can cover from not-very-low frequency to very high frequency. Certainly, we can use localized schemes to cover from very low to very high frequency. However, localized schemes introduce higher order mode effects, these effects are getting more and more serious at high frequency when the positive terminal (or the strip) and negative terminal (or the ground) is getting electrical far. To resolve the above problem, we introduced the Advanced Extension port.

The Advanced Extension port is similar to Extension for MMIC port. It tries to find the V from the integration of E-field along a path. However, its integration path is different from the one for Extension for MMIC port. We found that the V calculated using the Advanced Extension port can yield very accurate results from very low to very high frequency. In fact, we found it can yield very good results with frequency more than 100 GHz for a 100-micron GaAs circuit even when the second propagation mode happens. Second propagation mode disturbs the standing waves. However, when we use V and I to solve the problem, the effect of the second propagation mode is much smaller.

The Advanced Extension port scheme is extremely versatile and stable even though we found some problem in IE3D 10.1 when we use it to model structures with multiple substrates. The problem is completely resolved in the IE3D 11. From now on, we always suggest users to use the Advanced Extension port scheme whenever 50-ohm normalized extension port is being used.

Now, we discuss lossy TLN issues. TLN theory is originally derived from waveguide theory. For a waveguide with non-lossy boundary (PEC or PMC etc), we found that the field pattern of a mode on different cross-sections are basically the same even though the magnitude can be different due to lossy material inside the waveguide. The field along the longitudinal direction can be expressed as:

$$E(x,y,z) = e(x,y) [A \cdot \exp(- \gamma z) + B \cdot \exp(\gamma z)] \quad \text{AM-10}$$

$$H(x,y,z) = h(x,y) [A \cdot \exp(- \gamma z) - B \cdot \exp(\gamma z)] / \eta \quad \text{AM-11}$$

Where the A and B are the coefficients for the incident wand reflected waves. The e(x,y) and h(x,y) are the field pattern distribution in the cross-section independent of the z-coordinate. The η is the so-called wave impedance and it should be a pure real number for a waveguide without loss on the cross-sectional boundaries. From AM-10 and AM-11, we can get the famous Telegraphist's equations:

$$V(z) = a \cdot \exp(- \gamma z) + b \cdot \exp(\gamma z) \quad \text{AM-12}$$

$$I(z) = [a \cdot \exp(- \gamma z) - b \cdot \exp(\gamma z)] / Zc \quad \text{AM-13}$$

All the equations are valid when the assumption of no lossy boundaries on waveguide walls. In practical applications, it is impossible to have non-lossy boundaries on waveguide walls. In such a case, the complete waveguide mode theory is no longer valid. When you have lossy walls, a rectangular waveguide will not have the cutting off frequency. Instead of an abrupt cutting off, it will become a gradually cutting off.

The basic point is that waveguide theory is valid for lossy TLNs if there is no power going out in the transverse direction or all the cross-sectional boundary is some kind of non-lossy boundary (PEC or PMC). You can have lossy material inside. It will create loss in the z-direction. You can not have loss in the cross-sectional directions or no power can be going out the cross-sectional boundary.

In planar transmission lines, we must have loss going into the transverse direction. It can be due to the lossy dielectrics or it can be due to the open boundary nature. For such a case, if we still use formula AM-12 and AM-13 to model it, we will have complex Zc. In fact, equations AM-12 and AM-13 or even equations AM-10 and AM-11 do not hold for such a situation. Basically, the field pattern at a cross-section will change with Z. If we still try to decompose the field into incident wave and reflected wave for TLN with lossy cross-sectional boundary, we are basically approximating it. From this sense, the Extension for Waves port is also an approximation for lossy TLNs. Anyway, all different de-embedding schemes have some kind of approximation in order to solve practical problems. The schemes using V and I are more stable, and it is normally better if the V calculated represents the power correctly. From our experiences, the Advanced Extension scheme is the best in all the extension ports and it can cover the whole frequency range.

Table AM.1 The application ranges of different ports.

Application	Advanced Extension	Extension for MMIC	Localized for MMIC	Extension for Waves	50-Ohms for Waves	Vertical Localized	Horizontal Localized
Low Freq	Excellent	Excellent	Excellent	Good	Good	Excellent	Excellent
Mid Freq	Excellent	Excellent	Excellent	Excellent	Excellent	Excellent	Excellent
High Freq	Excellent	Good (May fail at very high)	Good	Excellent	Excellent	Good	Good
Highly Packed	No room	No room	Excellent	No room	No room	Excellent	Excellent
Coupled Ports	Excellent	Excellent	Good	Not good	Not good	Excellent	Excellent
Single ended	Excellent	Excellent	Good	Excellent	Excellent	Excellent	Excellent
Differential	Excellent	Excellent	Good	Not good	Not good	Excellent	Excellent
Finite Ground	Excellent with + and -	Excellent with + and -	Not good	Not good	Not good	Excellent	Excellent

Please note the following: A Vertical Localized port (or a Horizontal Localized) port requires a user to define a positive terminal and a negative terminal. It is a differential by itself. For an Advanced Extension port (or an Extension for MMIC port or a Localized for MMIC port), we have to define "+" port and "-" port for differential applications.

Lately, circuits are packed more and more into a small estate. All the components are packed together. Also, differential feed is very popular. We should try to simulate such a structure in order to receive the highest accuracy. Breaking them into sub-circuits for simulation and combining the s-parameters from sub-circuits for the s-parameters of the complete circuit may introduce significant error. Certainly, simulating the whole circuit becomes computational intensive or even impossible. If we have to break a circuit into sub-circuits, we need to break them smartly in order to preserve the highest accuracy. For example, try to avoid breaking a differential structure without a ground reference. If you have a differential pair has a ground reference, try to define the "+" terminal of the original differential pair referenced to the ground as one differential pair, and the "-" terminal of the original differential pair referenced to the ground as another differential pair. If we use the same differential pair, we may miss something from the even mode in case the differential pair is not balanced.

Appendix AN. Different Files Involved in an IE3D Simulation

Multiple files are involved in an IE3D simulation. The file configuration is changed in the IE3D 11 for better management. The various files and the comparison between the IE3D 10 and IE3D 11 are documented in Table AN.1. Please note that the output files (.sp, .spt, .cur, .mpa, .pat, and .log) are used to be in the same directory as the input files (.geo and .sim). Starting from IE3D 11, they are saved into the OUTPUT directory beneath where the .GEO and .SIM files are. On IE3D 14, their locations can be configured through the Optional Parameters in the Basic Parameters.

Table 3.3 The various files involved in an IE3D simulation.

Extension	Description	File Type
.geo	It saves the geometry information. It is the file storing the geometry and simulation parameters for IE3D. The .geo file may also contains the simulated s-parameters for regular simulations and the FastEM Design Kit data for full-wave EM tuning.	Input/Output
.sim	It saves the simulation input data such as the frequency points and the optimization goals. It is the input file used by IE3D engine. An identical copy of the data is included inside the .geo file in IE3D 14 while the .sim file is still needed in a simulation.	Input
.wks	It is the workspace file for an IE3D project (.geo). It saves the status of the windows. It also links the .geo file with the simulation results files (the .cur file and the .mpa file, etc.). In case you delete it, the geometry will not be affected. However, you will not be able to access the .cur and .mpa files for current and pattern visualization on MGRID opening the .geo file. Certainly, you can still post-processing the .cur file and the .mpa file on MGRID and/or PATTERNVIEW even the .wks file is removed.	Input/Output
.log	It saves the intermediate data of a simulation. You can see the simulation time and predicted RAM usage in the file. It will save any error encountered in the simulation. In case IE3DOS stops for some reason, you should check the .log file. In case the .log file is not created, it may means that IE3DOS has difficulty in opening or reading t he .sim file.	Output
.sp	It is an output file. It saves the primary simulation results or the s-parameter data at all the frequency points. In case you enable the AIF, the .sp file will be saved at the end of the simulation. If AIF is not enabled, the .sp file will save all the finished s-parameters. The .sp file is in Agilent/EEsoft Touchstone compatible format. It can be used in other compatible software packages. It can also be used on MGRID, MODUA and MDSPICE. Starting from IE3D 12, a copy of the s-parameters are saved into the .geo file.	Output
.spt	It is an output file. It saves the s-parameter data at the actually simulated frequency points. It only appears when AIF is enabled. If AIF is disabled, the actual simulated s-parameter data will be saved in the .sp file.	Output
.cur	It is an output file. It saves the current distribution data for visualization in post-processing, if you choose to save the file in the Simulation Setup dialog. The .cur file can be opened by MGRID in the post-processing mode for current distribution visualization, near field calculation and visualization, and radiation pattern calculation. The .cur file can be automatically accessed on the MGRID opening the .geo file in IE3D 14.	Output
.pat	It is an output file. It saves the radiation pattern of specified excitation. The file can be added into the Pattern List on MGRID and PATTERNVIEW for display and post-processing. In case you did not enable saving the Radiation Pattern File in the Simulation Setup dialog on MGRID, you can still generate the .pat file based upon the .cur file or the	Output

	.mpa file on MGRID and PATTERNVIEW.	
.mpa	It is the general radiation pattern file. It stores complete pattern information with tunable excitations. The .mpa file allows you to find the radiation pattern for any specified excitation (.pat) on MGRID and PATTERNVIEW. Starting from IE3D 14, .mpa becomes the primary pattern file because you can use it as if an .pat file while its excitations can be changed anytime. The .mpa file can be accessed internally on MGRID opening .geo file on IE3D 14. In future versions, we will implement a feature allowing you to optimize the excitations of a multiple port structure without changing the geometry configuration based upon the .mpa file. It can be extremely useful for smart antenna design.	Output
.dsg	It is the MODUA file for circuit simulation.	Input/Output
.sts	It is a status file designed for using external matrix solver	Output
.ie3	The IE3DLibrary file for the layout	Input
.fld	It is the output file for near field calculation. It is normally in the same directory as the .cur file.	Output
.tra	It is the TLN parameters files created by MODUA in finding TLN parameters.	Output
.arr	It is the array factor file for PATTERNVIEW.	Input
.lib	It is the SPICE compatible file created by MODUA and MDSPICE.	Output
.ect	It is the excitation file created by MODUA or IE3D when we do mixed EM and circuit simulation.	Output
.ipa	It is the indexed parameter file created by IE3D to output specified parameters in tabular ASCII file format.	Output
.3dt	It is the 3D Text format for describing 3D geometry	Input/Output
.zeq	It is an ASCII file for the frequency dependent lumped element equivalent circuit.	Input/Output
.zlw	It is the workspace file for IE3D 12.x. It is replaced by the .wks file in IE3D 14 and it is obsolete.	Input/Output

Appendix AO S-Parameters Visualization and Post-Processing on MODUA

MODUA serves multiple purposes for the Mentor Graphics IE3D products. The 1st purpose is to display S, Y and Z-parameters etc. in different forms. The 2nd purpose is to post-processing S, Y and Z-parameters. For example, we can extract the RLC equivalent circuit from an s-parameter file on MODUA. The more advanced feature of MODUA is to perform a nodal simulation. In fact, MODUA allows the IE3D to perform a mixed EM and circuit co-simulation and optimization. It is a very powerful utility program for the IE3D and FIDELITY products.

Starting from IE3D 14, s-parameters visualization and many post-processing features have implemented into MGRID. However, the mixed EM and circuit co-simulation and optimization and the RLC extraction features are not transferred to MGRID. MODUA is still very useful before the 2 major features are integrated into MGRID.

In this appendix, we will concentrate on using MODUA for displaying various parameters. Assume you have already gone through the Chapter 3 of the IE3D user's manual and you have created a few s-parameter files (c_bend.sp and c_bend.spt in .\ie3d\practice\output directory). The c_bend.sp file contains the s-parameter of a microstrip bend with many frequency points. The c_bend.spt contains the s-parameters at selected frequency points. We will use the two s-parameter files to demonstrate to you how we display various parameters in different forms for you.

1. Displaying S-Parameters in Cartesian Graph.

Step 1 Run MODUA. Select File->Display Parameter Module. You will be prompted for the file name. Select .\ie3d\practice\output\c_bend.sp and MODUA will get the data and display the parameters based upon the default setting for the display. The default of default is to display dB[S(1,1)], dB[S(2,1)] and dB[S(2,2)] in a Cartesian display. It can be different depending upon whether it is changed to other default value. Changing the default is easy on the MODUA 11 and we will show it to you later.

Step 2 Select Control->Define Display Graph. You will be prompted for the e Display Parameters dialog (see Figure AO.1).

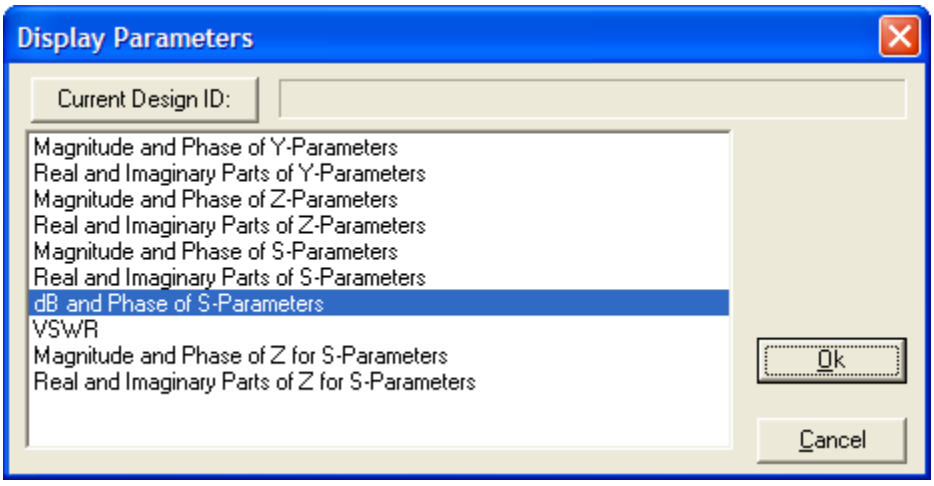


Figure AO.1, The Display Parameters dialog.

You can choose any of the listed type of display. You have the choices for S, Y and Z-parameters, and VSWR. You also have the choice for Z for S-Parameters which are not defined

in any textbook. This feature is requested by some IE3D users and we implemented it and we call it Z for S-Parameters. Basically, we define the Z for S-Parameters as:

$$Zsp(i,j) = \sqrt{(Zci \cdot Zcj)} \cdot [\,1 + S(i,j)] / [\,1 - S(i,j)] \qquad \text{(AO-1)}$$

Where Zci is the normalization (or terminating) impedance of port i. Some user may think that it is the definition of Z-parameters. In fact, it is only true for 1-port structure. For a multiple port structure, the relationship between S and Z parameters are much more complicated. For example, for 2-port 50-ohm normalized s-parameters, we have:

$$\Gamma = S(1,1) + S(1,2) \cdot S(2,1) / [\,1 - S(2,2)] \qquad \text{(AO-2)}$$

$$Z(1,1) = 50 \cdot (\,1 + \Gamma\,) / (\,1 - \Gamma\,) \qquad \text{(AO–3)}$$

On the dialog, there is an item called Current Design ID and it is empty by default. The Current Design ID is to identify the display items when we are comparing items from different files.

Step 3 Select "dB and Phase of S-Parameters". Select OK to continue. MODUA will prompt you for the Display Selection dialog.

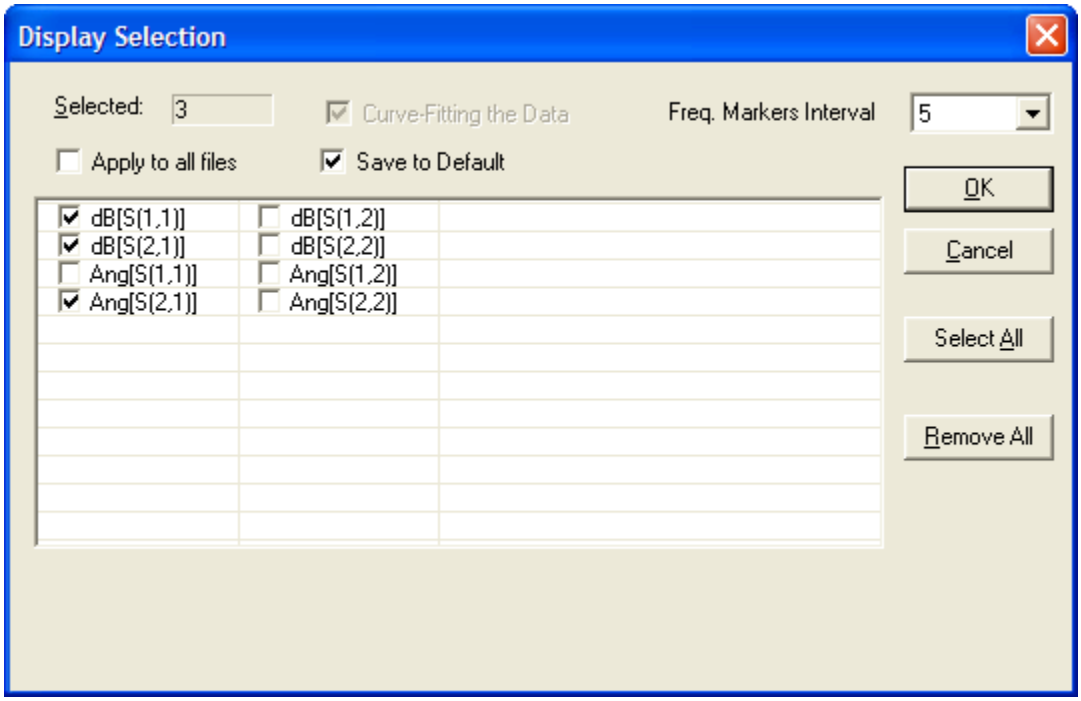


Figure AO.2 The Display Selection dialog for you to choose the items to be displayed.

Step 4 Select dB[S(1,1)], dB[S(2,1)] and Ang[S(2,1)]. Please OK to continue. MODUA will display the 3 items in a Cartesian graph (see Figure AO.3).

On the Display Selection dialog, there are two check boxes "Apply to all files" and "Save to Default". The "Apply to all files" allows you to select all the corresponding items of the S-parameter files in the Parameter File Queue when we compare different S-parameters. Comparing different S-parameters is discussed in Appendix AP. The "Save to Default" allows you to save the selected items into the default. Next time, when you display new S-parameters, the corresponding S-parameters will be displayed automatically.

You can do much on processing the data in the Cartesian display. You can define markers on the curves. You can find difference between two markers. For example, we see that the dB[S(1,1)] crosses the -30 dB line twice: one at about 10 GHz and the other at about 30 GHz. We want to know exactly where they are and the frequency difference between the two cross points. We can do it easily on MODUA.

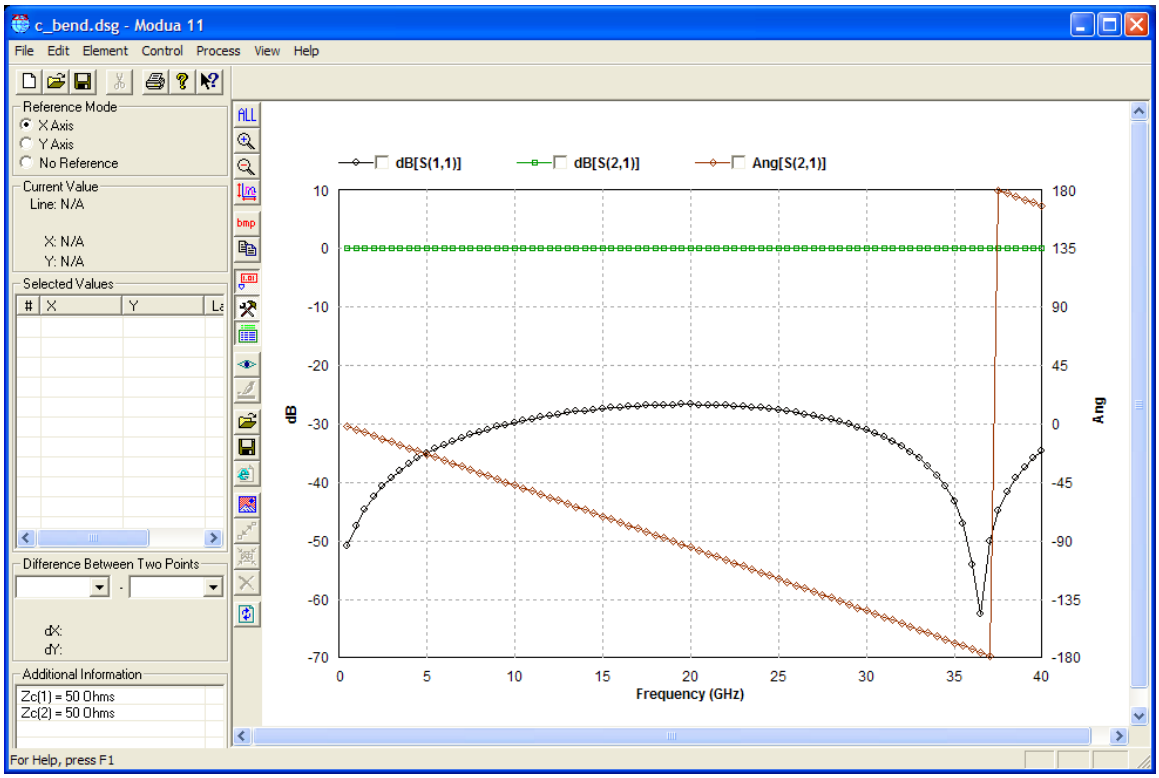


Figure AO.3 The Cartesian display for the 3 items of S-parameters.

2. Defining Markers on Curves:

Step 5 Check the checkbox on the legend for "dB[S(1,1)]". Make sure the Reference Mode (at top left) is X-Axis. In this mode, MODUA allows us to find the cross point between the dB[S(1,1)] curve and the X = constant (or Frequency = constant for this case).

Step 6 Move the cursor onto the graph and you will see a blue line representing the X = constant line. Move it to some where at about (X, Y) = (10, -30) and click. The marker "1" anchors at a point at the mouse location. The exact (X, Y) values are (9.8, -29.993), as shown in Figure AO.4.

MODUA is still in the mode. While you are moving the cursor to some where close to (X, Y) = (30, -30), you can see the blue line and it is indicating the exact cross point at (X, Y) = (28.8667, -30.0501) (see Figure AO.4). You may not be able to get to the exact -30 dB point using the mouse. However, it does not matter because we can fine tune it later. Please click the left mouse button again to anchor the 2nd marker.

Step 7 Please un-check the check box in the legend of "dB[S(2,1)]" to exit the mode. The two entered markers are still there while the blue line is gone.

The two markers are not exact at Y = -30 dB. We would like to find the exact Y = -30 dB points.

Step 8 Double click the "#1" point in the list box of the markers on the left pane of the MODUA window. The "No.1 Marker Property" dialog comes up (see Figure AO.5a).

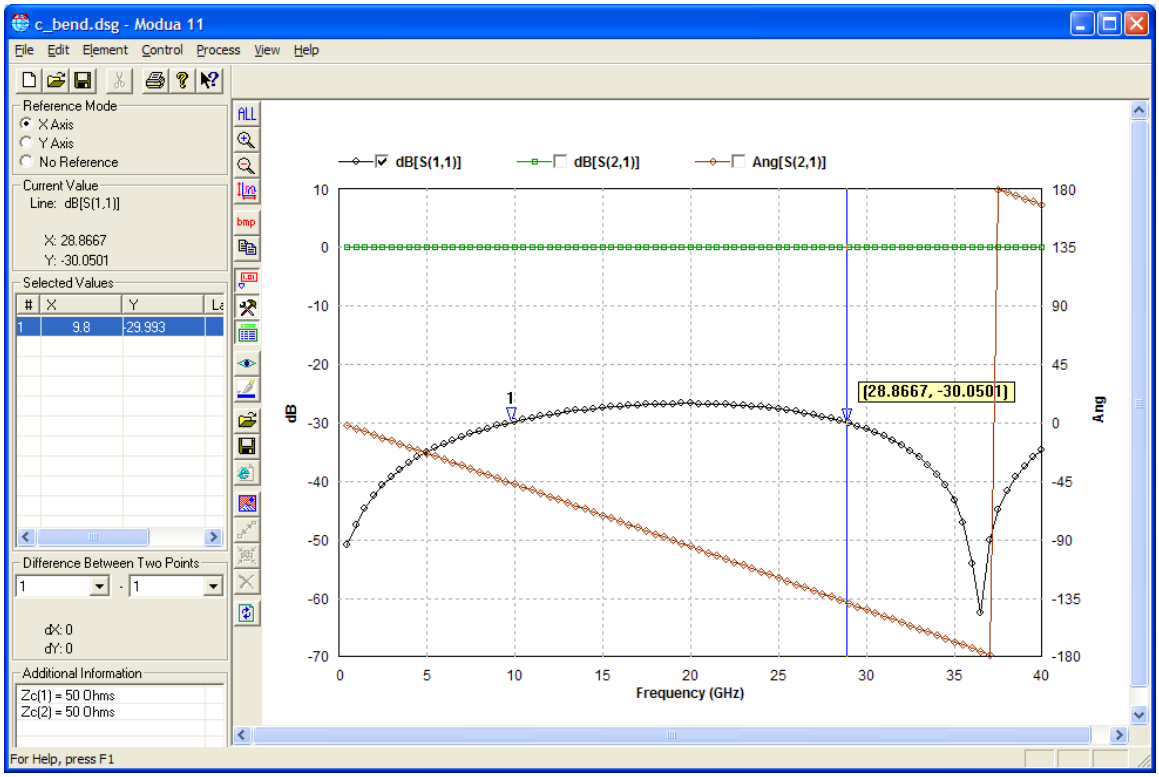


Figure AO.4 The marker "1" is defined while the marker "2" is being defined.

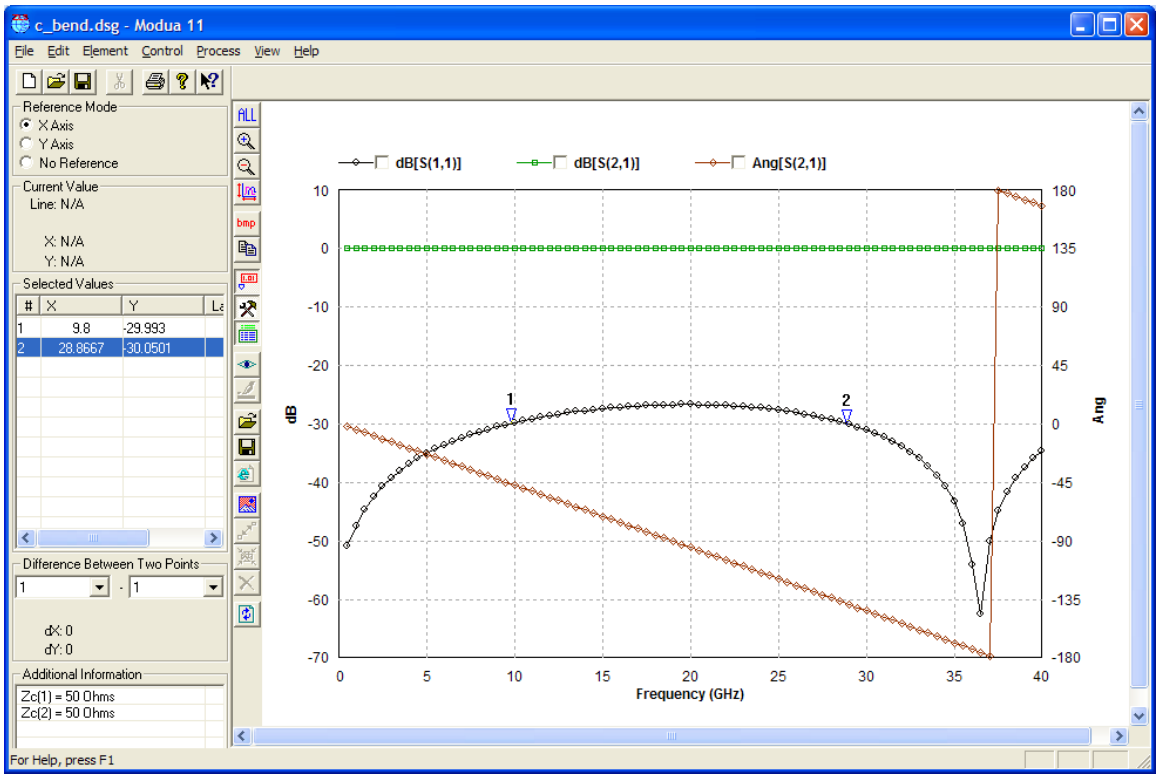
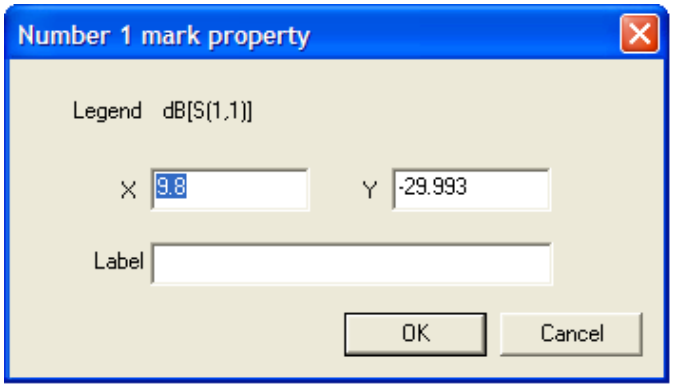
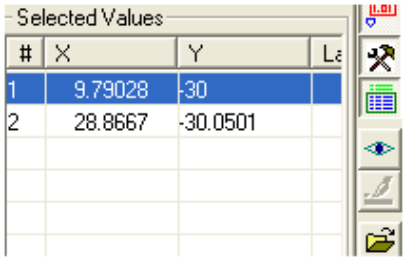


Figure AO.5 The MODUA display after two markers are defined and MODUA is out of the mode.

Step 9 Change the Y from -29.993 to -30. Enter OK. The No.1 marker is slight changed in the marker plane (or the Selected Values pane) to (X, Y) = (9.79028, -30). It is the exact cross point with dB[S(1,1)] = -30 dB.





(a) The No. 1 Marker Property dialog (b) The marker pane after the change.

Figure AO.6 The marker property editing dialog and the result after the change.

Step 10 Click the No.2 marker. Change the No.2 Marker Property to Y = -30. Select OK. The No.2 marker is updated moved slightly to Y = -30.

Step 11 Select the 1st combo box in Difference Between Two Points to "2". We get the difference between the No.2 marker and No.1 marker as dX = 19.0172 and dY = 0 (see Figure AO.7). Basically, it is what we were supposed to need: the frequency difference between the -30 dB points.

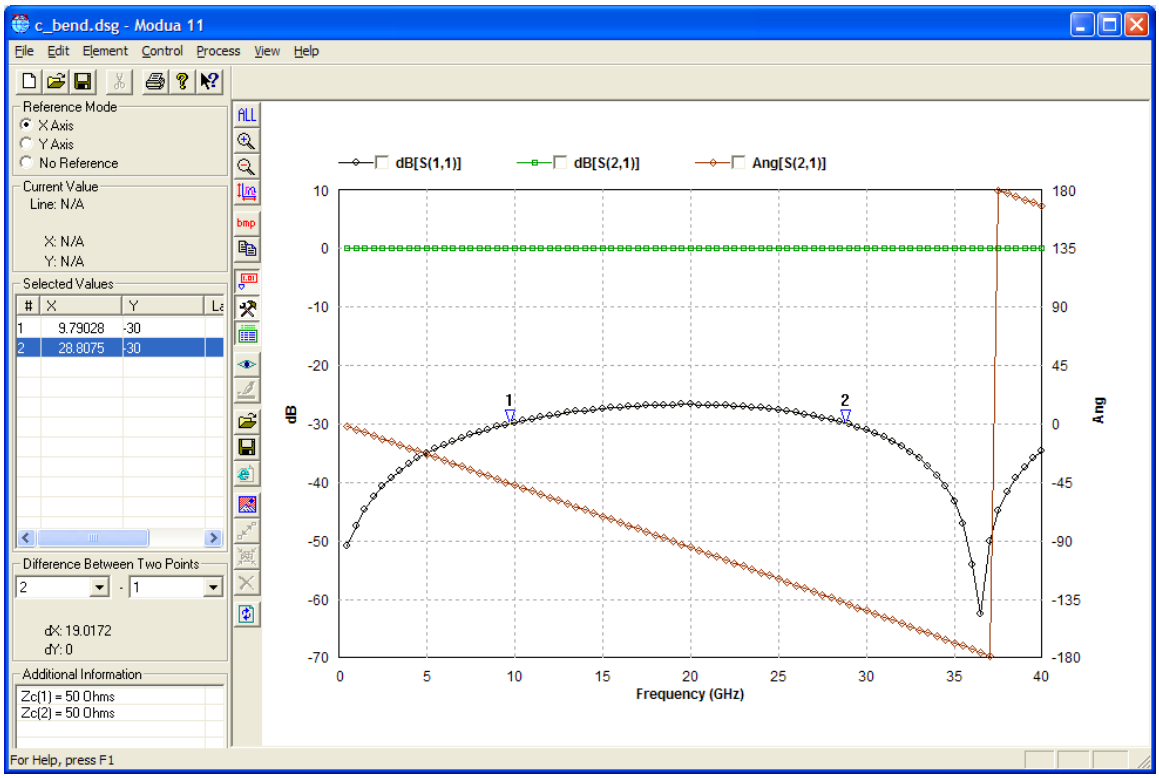


Figure AO.7 The fine tuned cross-points and their difference.

We have just demonstrated how we can define markers between curves and X = Constant lines. We can also define markers between curves and Y = Constant lines by selecting the Reference Mode = Y axis. We can also define markers at any point in the displayed region using the Reference Mode = No Reference.

There are some bitmaps on the tool bar between the left pane and the displayed area. These are the commands you can use for the display. For example, you can zoom in/out the picture. You can save the picture into a bitmap file or copy the picture into the clipboard. You can change the graph properties. You can save the pictures into an XML file and the file can be loaded for redisplay or the curves of the saved XML file can be imported into another display for comparison. You can also get a list of the data for the graph in a nicely tabulated XML format. In fact, the XML file for the picture contains the data in a nicely formatted ASCII format.

Also, you can access the commands by clicking the right mouse button to invoke the pop-up menu.

We just demonstrate how we display s-parameters in Cartesian plot. You can also display the S-parameters in Smith Chart.

3. Displaying S-Parameters in Smith Chart.

Step 11 Select Control->Define Display Smith Chart. Select S(1,1) and S(2,1) in the list. Select OK and the S(1,1) and S(2,1) are displayed in a Smith Chart. On Smith Chart display, we can also define markers for cross points between curves and Reflection Coefficient = Constant lines, curves and Impedance = Constant line.

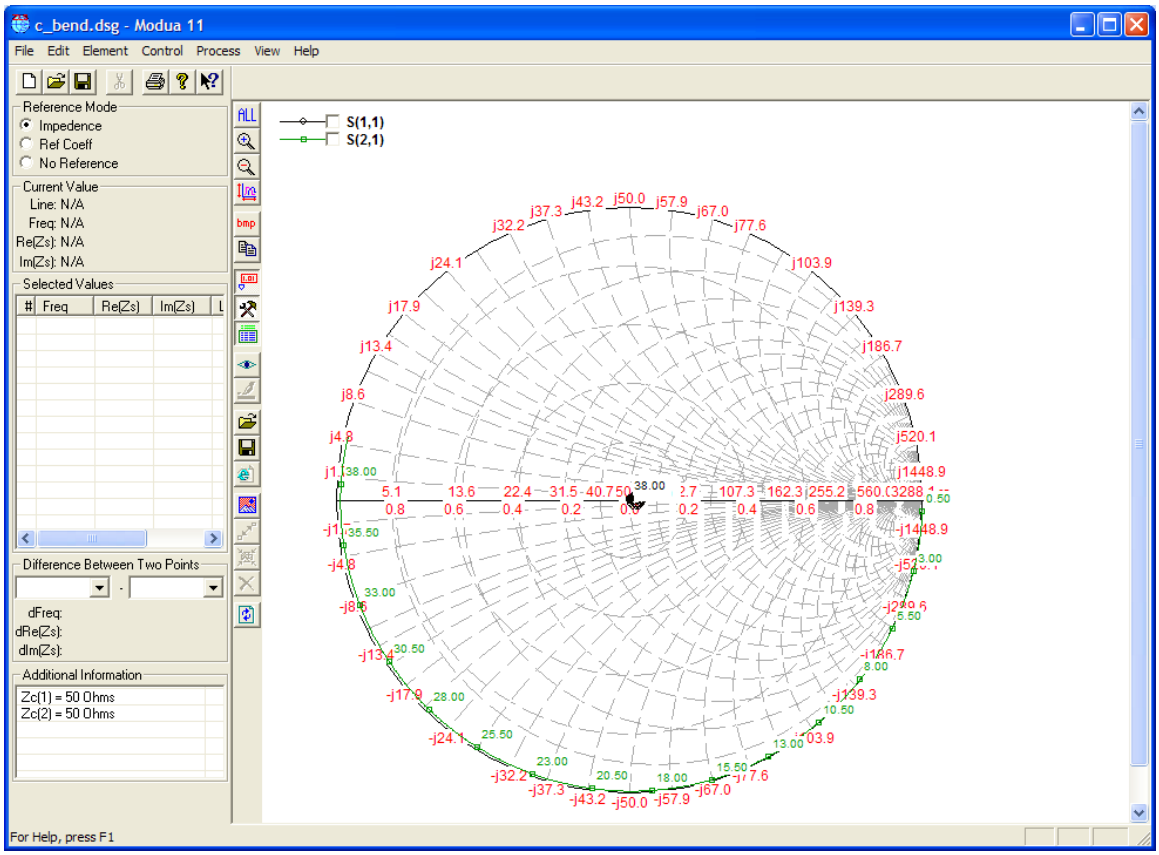


Figure AO.8 The Smith Chart display.

4. Displaying Y and Z-Parameters in Data List.

We have demonstrated how to display s-parameters in Cartesian graphs and Smith Chart. We certainly can also display Y and Z-parameters in Cartesian graphs. Here, we will demonstrate how we can list the data in ASCII format.

Step 12 Select Control->Define Display Data and select OK. Select "Real and Imaginary Parts of Y-Parameters" and select OK. Select Re[Y(1,1)], Re[Y(2,1)], Im[Y(1,1)] and Im[Y(2,1)], and select OK. The data is listed on the screen in text format.

Step 13 Select File->Save Displayed Data and enter the file name as: c_bend.txt. MODUA saves the data into the ASCII file and open it on Notepad (see Table AO.1).

Table AO.1 The displayed and saved data list

Freq[GHz]	Re[Y(1,1)]	Im[Y(1,1)]	Re[Y(2,1)]	Im[Y(2,1)]
0.5	4.505e-002	-0.457	-4.505e-002	0.4574
1.	1.166e-002	-0.2301	-1.166e-002	0.2309
1.5	5.313e-003	-0.1533	-5.313e-003	0.1545
2.	3.055e-003	-0.1146	-3.054e-003	0.1162
2.5	1.995e-003	-9.125e-002	-1.994e-003	9.323e-002
3.	1.413e-003	-7.557e-002	-1.412e-003	7.796e-002
3.5	1.058e-003	-6.43e-002	-1.056e-003	6.709e-002
4.	8.261e-004	-5.576e-002	-8.237e-004	5.896e-002
4.5	6.655e-004	-4.906e-002	-6.627e-004	5.267e-002

```
5.      5.498e-004   -4.365e-002  -5.465e-004  4.766e-002
5.5     4.636e-004   -3.916e-002  -4.599e-004  4.359e-002
6.      3.978e-004   -3.537e-002  -3.935e-004  4.023e-002
6.5     3.462e-004   -3.212e-002  -3.414e-004  3.74e-002
…
```

5. High Accuracy Interpolation.

When we simulate a structure at multiple frequency points, we always try to use the AIF. AIF allows IE3D to simulate the structure at selected frequency points adaptively and interpolate the s-parameters with guaranteed accuracy. The scheme is very robust. When we enable AIF, the .spt file contains the truly EM simulated results while the .sp file contains the interpolated results. In case we have simulated a structure in some coarse frequency points, and we want to find the frequency response with very fine frequency points without re-simulating the structure, we can still use interpolation on MODUA to do it.

Step 1 Run MODUA. Select File->Display Parameter Module. Select .\ie3d\practice\output\c_bend.spt. MODUA will display the frequency response saved in c_bend.spt. The curves are not smooth because there are few frequency points in the file (see Figure AO.9a).

Step 2 Select Process->Curve-Fitting and Interpolation. MODUA will prompt you to select a scheme. The default is General Intelli-Fit. There are other schemes. However, they are much less good as General Intelli-Fit. We should always use General Intelli-Fit. Select OK to continue. MODUA will prompt you to define frequency points.

Step 3 Please enter Start Frequency = 0.5, End Frequency = 40, Number of Frequency = 80. Select Enter to add the defined frequency range into the list. You can repeat the process to add more or remove some frequency points using the available buttons. Please understand that, similar to using the AIF, defining more frequency points normally do not slow down the process much. You can define as many as you like.

Step 4 Select OK to continue. MODUA will yield smooth results as shown in Figure AO.9b. The results should be either identical or very close to those in c_bend.sp.

You may wonder whether the General Intelli-Fit will yield accurate results. The answer is yes or no. For this particular example, it is yes because the frequency points in c_bend.spt are adaptively selected by the IE3D engine to yield accurate results over the frequency range for the structure. For a s-parameters with some arbitrarily selected frequency points, General Intelli-Fit may not guarantee the results are always reliable even though most of the times the results may be good. You should always keep in mind the above fact when you use General Intelli-Fit on MODUA. MODUA does not have a way to simulate the structure at additional frequency point and it can not detect whether the results are reliable.

6. Change Normalization or Terminating Impedances of Ports:

The s-parameters on IE3D are saved in Agilent/EEsof Touchstone compatible. Due to the limitation of the format, we have to normalize the s-parameters to 50-ohms. Users are allowed to change the port normalization impedance (Zc) in the Simulation Setup dialog. However, the user defined Zc is used in optimization only. The s-parameters saved are still normalized to 50-ohms in order to make it compatible to the Touchstone format. However, we can visualize or obtain the s-parameters normalized to other than 50-ohms on MODUA.

After you display any s-parameters in either Cartesian or Smith Chart display, you can select Control->Terminating Impedance to change the normalization impedance of s-parameters for each port.

Appendix AP. Comparing Different S-Parameters on MODUA

We have demonstrated how to use MODUA to display s-parameters in a file. We can also use MODUA to compare s-parameters from different files. In fact, we can even use MODUA to perform nodal simulation based upon s-parameters and other elements. Another very important feature of MODUA is to perform mixed EM and nodal simulation and optimizations with a design consisting of geometry modules, s-parameter modules and other lumped elements. In this appendix, we will concentrate our discussion on using MODUA to compare s-parameters from different files. We will demonstrate

Step 1 Run MODUA. Select File->Display Parameter Module. You will be prompted for the file name. Select .\ie3d\practice\output\\c_bend.sp and MODUA will display the default items from c_bend.sp on the window.

Step 2 Select Control->Define Display Graph. Select "dB and Phase of S-Parameters". Select OK to continue. Select dB[S(1,1)], dB[S(2,1)] and Ang[S(2,1)] in the list. Select OK to display the items on a Cartesian plot. You will see the same display in Figure AO.2.

Step 3 Select File->Parameter File Queue. MODUA will prompt you for the Parameter File Queue dialog.

Step 4 Click at the Current Design ID. MODUA will prompt you to enter an ID for it. Enter "AEC Layers=0" and select OK. The Current Design ID is changed. The label "AEC Layers=0" will be added into the legend of each item from the c_bend.sp file.

Step 5 Select Add File button. MODUA will prompt you for a new s-parameter file to be added into the list. Select .\ie3d\practice\output\c_bend3.sp. MODUA will prompt you for the ID of the newly added file. Enter the ID as "AEC Layers=1". Select OK to continue. The file c_bend3.sp is added into the queue list. You will see "No.1 (3 items)….". The "(3 items)" means that the default items of "dB[S(1,1)]", "dB[S(2,1)]" and "dB[S(2,2)]" are selected automatically since you have checked "Apply default display items to file".

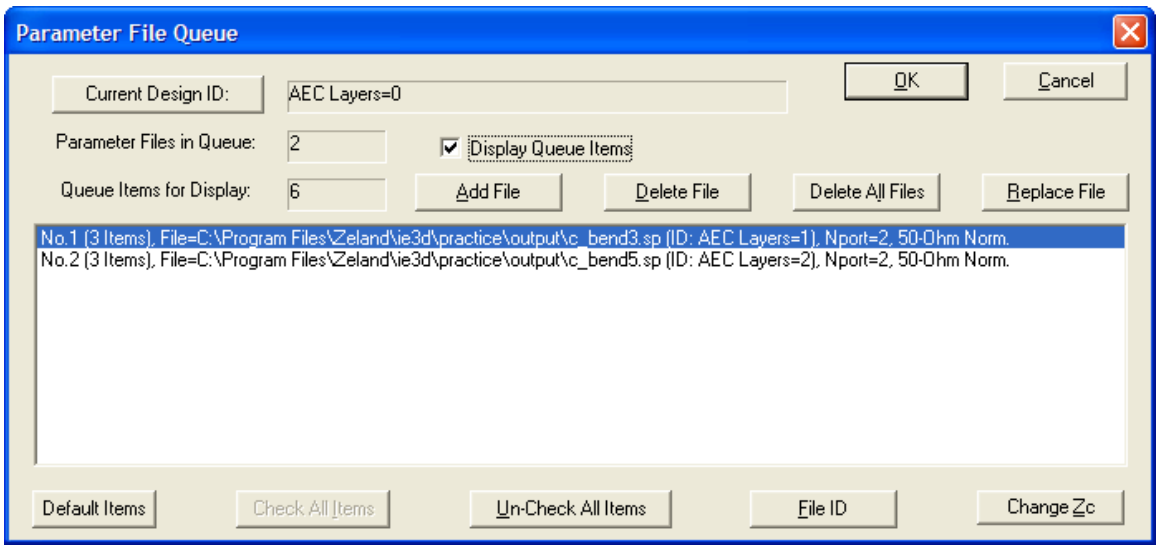


Figure AP.1 The Parameter File Queue dialog after two queue files are added into the list.

Step 6 Select Add File button again. MODUA will prompt you for a new s-parameter file to be added into the list. Select .\ie3d\practice\output\c_bend5.sp. MODUA will prompt you for the ID of the newly added file. Enter the ID as "AEC Layers=2". Make sure "Apply default display items to

the file" is checked. Select OK to continue. The file c_bend5.sp is added into the queue list. We will have the window shown in Figure AP.1.

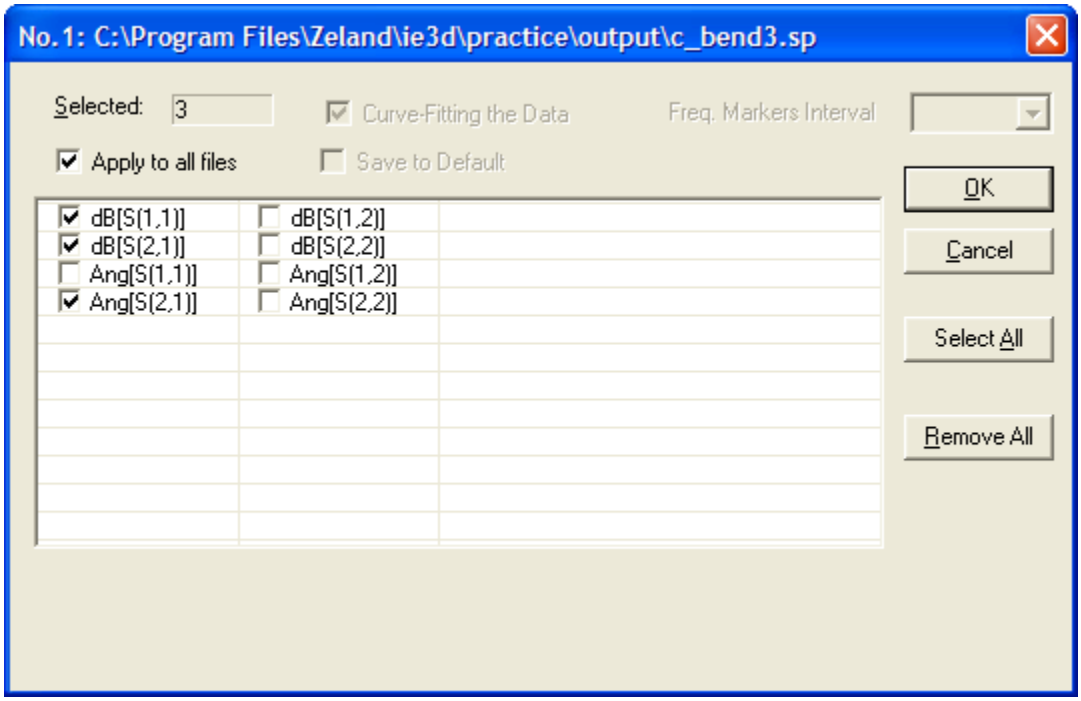


Figure AP.2 The dialog box to select the items to be displayed in the queue file.

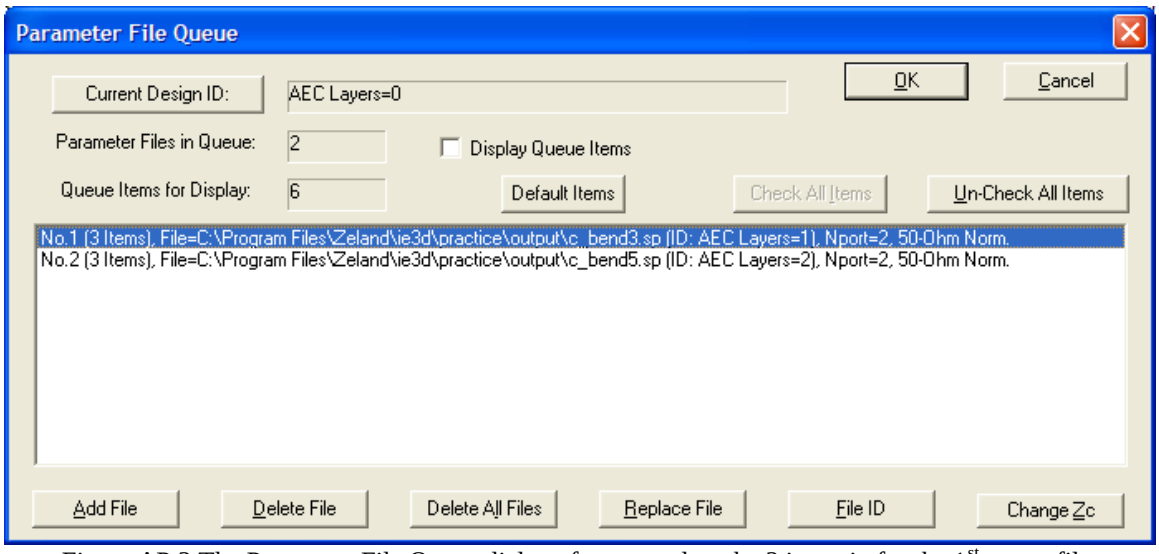


Figure AP.3 The Parameter File Queue dialog after you select the 3 items in for the 1st queue file.

Step 7 Double-click the "No.1 (0 items)…" in the listbox. MODUA will prompt you to select the items of the c_bend3.sp you want to display. The items dB[S(1,1)], dB[S(2,1)] and Ang[S(2,1)] are already selected and we don't need to change them. Please select OK to continue.

In case you need to change the display items for a queue file, you should double-click at it and change the selection.

Step 8 Please check the Display Queue Items in the Parameter File Queue dialog. Then, select OK. MODUA will display 9 items, 3 from each file, in the window (see Figure AP.4).

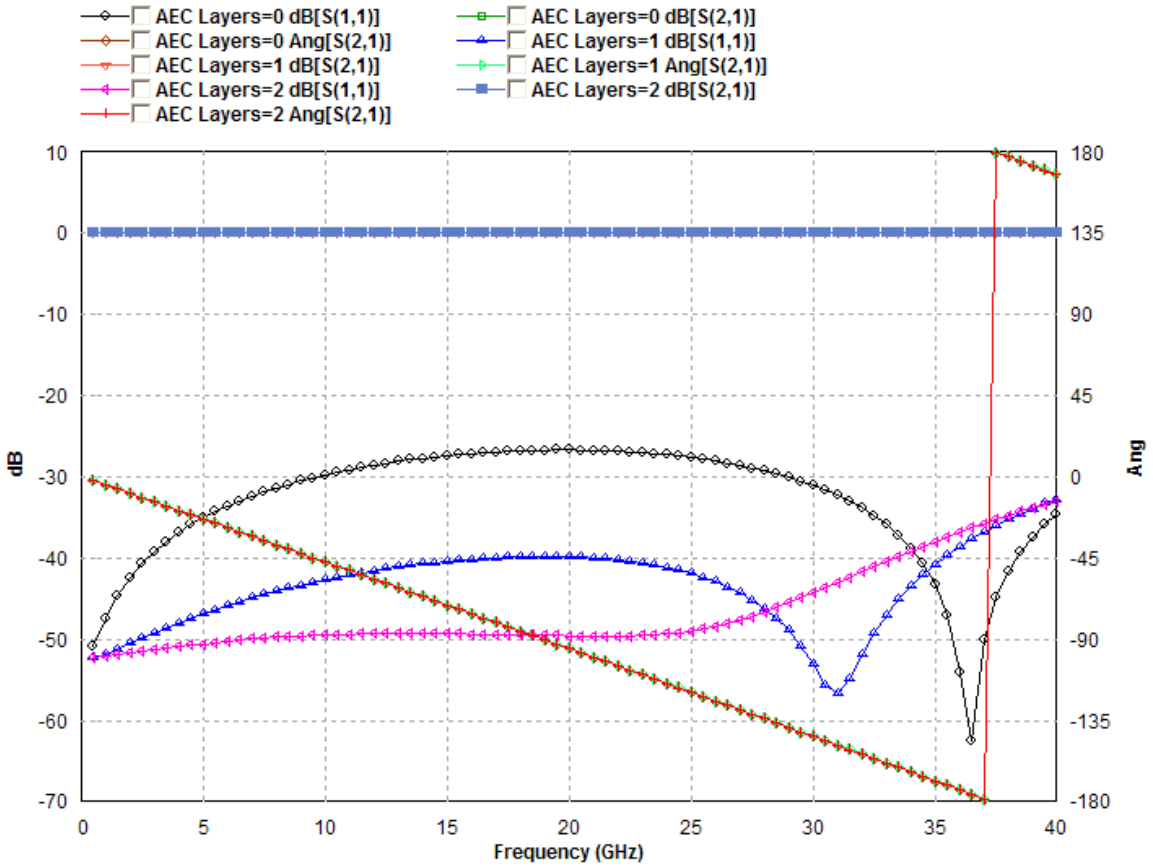


Figure AP.4 The 9 items from 3 files are displayed and compared.

Step 9 Select File->Parameter File Queue command. The Parameter File Queue dialog comes up. Uncheck Display Queue Items. Select OK. MODUA will display the 3 items from the current design only. It will not display the items from the queue files.

Appendix AQ. Finding the Zc of a Transmission Line

On the IE3D, we have an option either to normalize the s-parameters to the port impedance (using Extension for Waves scheme) or 50-ohms (other schemes). More discussion on normalization impedance can be found in Appendix AM. No matter which scheme we use, IE3D does not explicitly solve the characteristic impedance (Zc) of a port. However, in case you want to find the Zc of a port, you can do a simple simulation on IE3D to find it.

Step 1 Run MGRID. Open file: .\ie3d\samples\straight.geo. It is a uniform microstrip line with length of 0.5 mm, width of 0.075 mm, thickness of 0.002 mm. The substrate thickness is 0.1 mm and the permittivity is 12.9.

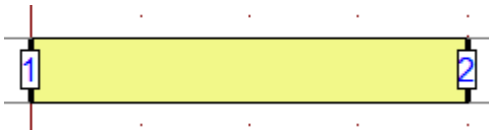


Figure AQ.1 A uniform microstrip line with length of 0.5 mm and width of 0.075 mm.

Step 2 Select Process->Simulate. Click at Enter button in the Frequency Parameters group. Enter Start Freq = 0.5 GHz, End Freq = 40 GHz, Number of Freq = 80. Select OK. The frequency points are added into the list box. Select OK on the Simulation Setup dialog. MGRID will invoke IE3D and run the simulation in seconds. After simulation, MODUA will be invoked to display the s-parameters saved in: .\ie3d\samples\straight.spt (not straight.sp). You can see markers at a few points on the curves, meaning that only those points are truly EM simulated. Other points are interpolated.

Step 3 Select Process->Find Transmission Line Parameters. The dialog comes up (Figure AQ.2). Please enter the Transmission Line Length = 0.5 mm.

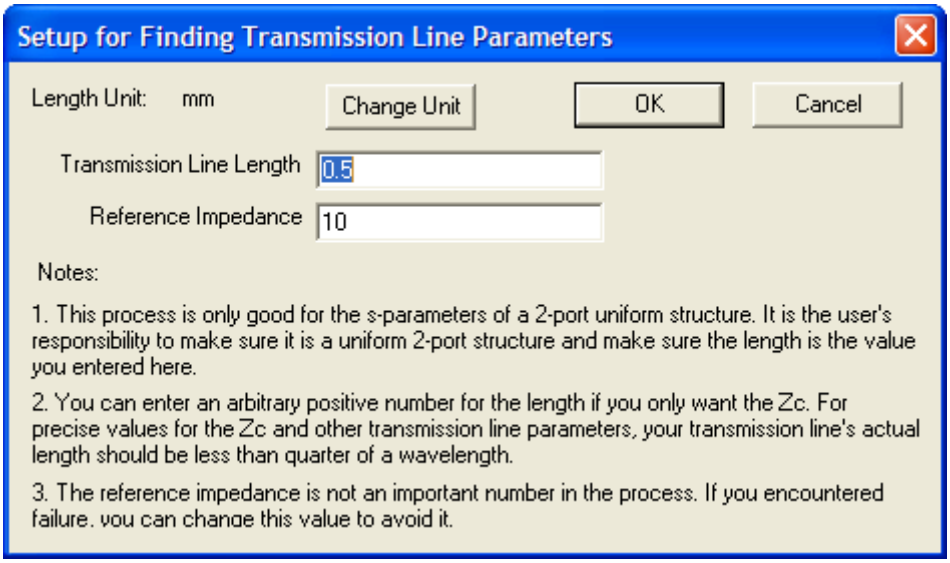


Figure AQ.2 The Setup for Finding Transmission Line Parameters dialog.

The length information is not saved in the s-parameter file which is processed by MODUA. We need to enter the length value in order to let MODUA know how long the TLN is. If we are just interested in the Zc of the TLN, we can enter any positive value for it. However, the calculated

waveguide wavelength and effective permittivity will become scaled values. We have to enter the correct physical length in order get accurate parameters related to physical length (waveguide wavelength, effective permittivity).

You may want to know how to find the physical length of a structure on MGRID. It is documented in Appendix AR.

There is another value called Reference Impedance (Zr). It is used in the extraction. The default value of Zr is 10 and we do not need to change it. We found out that the solved Zc value is kind of independent of the Zr value even it is used in the extraction. In fact, there are 2 reference impedances. The other one is chosen to be Zr2 = 10 Zr (=100 for default case). Unless the Zc value is extremely close to the Zr and Zr2 values, it is not affected by the chosen value of Zr.

Step 4 Select OK to continue. MODUA will find the Zc of the TLN from the 2-port s-parameters and display them on the window.

Step 5 Select File->Save TLN Parameters command. MODUA will prompt you to change the default file name: straight.tra. Select OK. MODUA will save the information in the ASCII file and open it on Notepad (see Table AQ.1).

Table AQ.1 The solved Zc values at different frequency points.

1 0 5
Design: C:\MentorGraphics\<latest_release>IE3D\SDD_HOME\IE3D\ie3d\practice\output\straight.dsg

Freq (GHz)	Re(Zc)	Im(Zc)	Lambda(mm)	Alpha(dB/mm)	Re(Ereff)	Im(Ereff)
0.50000	50.756286	-2.765090	204.183616	-0.014690	8.596977	-0.947863
10.50000	50.324567	-0.199064	9.807058	-0.025974	8.475705	-0.079121
20.50000	50.483800	-0.143885	5.007620	-0.037568	8.528366	-0.058797
31.50000	50.765361	-0.144774	3.238578	-0.051306	8.635898	-0.052586
40.00000	50.994457	-0.165675	2.535205	-0.071263	8.739580	-0.057864

As you can see, the Zc at selected frequency points are solved only. Those are the frequency points available in the .\ie3d\samples\straight.spt file. Other frequency points displayed on MODUA in Step 2 are interpolated only. If you want to find the Zc at all the frequency points, you should select File->Display Parameter Module and select the file: .\ie3d\samples\straight.sp which contains all the frequency points. Then, select Process->Find Transmission Line Parameters command for it.

Please understand that the Find Transmission Line Parameters command can only be used in certainly conditions: (1) The 2-port s-parameter must be for uniform structure. In case, it is not a uniform structure, it may cause degrading of accuracy, even though you may still be able to use it for non-uniform structure to get some reasonable results. (2) The electrical length of the TLN should be shorter than half of a waveguide wavelength. In case the TLN is close to half a wavelength at some frequency, the solved Zc will be affected. Basically, it is impossible to solve the Zc of a lossless TLN from the 2-port s-parameters. You always have |S(1,1)|=0 and |S(2,1)|=1 at half of a wavelength no matter what Zc your TLN is. (3) The s-parameters should not be normalized to the Zc of the TLN.

Appendix AR. Measuring Dimensions and Using Saved Information

Occasionally, we may want to measure some distance on MGRID. For example, we have saved a uniform transmission line in .\ie3d\samples\straight.geo. We want to measure how long and how wide the TLN is. MGRID allows you to find out such information easily and it also allows you to do much more.

1. Measuring Dimensions on MGRID:

Step 1 Run MGRID. Open file: .\ie3d\samples\straight.geo. The straight microstrip is shown in Figure AR.1 (the label for the vertices are not in the display).

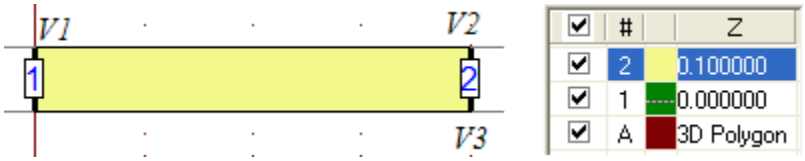


Figure AR.1 A uniform microstrip and the current layer window.

Basically, we want to know the distance between V1 and V2 and between V2 and V3.

Step 2 Click at the No.2 Z = 0.1 layer on the Layer Window to shift the input focus to the layer.

Step 3 Select Input->Set to Closest Vertex if it is not checked. If it is checked, please leave it as it is. In this mode, it allows mouse input to snap to vertex.

Step 4 Click at around V1 to snap a vertex at V1 (see Figure AQ.1). Click at V2 to snap the 2nd vertex at it. Click at around V3 to snap the 3rd vertex at it. We basically entered a line connecting the vertices V1, V2 and V3.

Step 5 Select Input->Info On Last Entry command. MGRID shows you the information about the V1 and V3 and about the entered path (see Figure AQ.2).

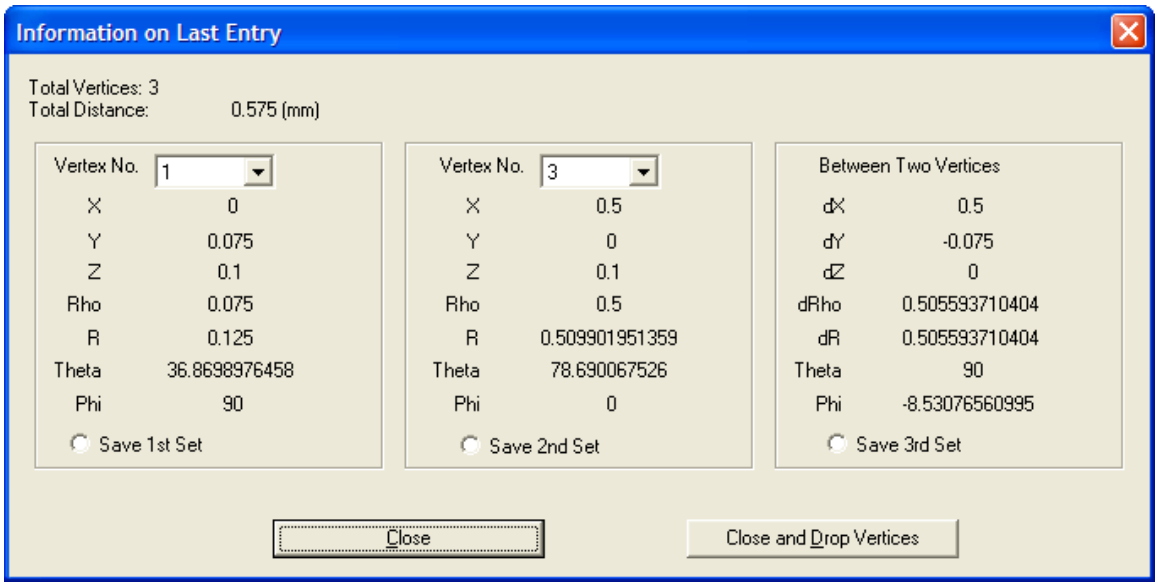


Figure AR.2 The Info on Last Entry dialog.

Basically, it tell you the entered path has 3 vertices. The total distance along the path from V1 to V2 and to V3 is 0.575 mm. The V1 is at (x, y, z) = (0, 0.075, 0.1) and the V3 is at (x, y, z)=(0.5, 0.0, 0.1). It also shows the coordinates in the cylindrical and spherical systems. Then, it shows you the information between V1 and V3: (dx, dy, dz)=(0.5, -0.075, 0.0). Basically, the strip length is 0.5 mm and the width is 0.075 mm.

In case you want to know the information on V1 and V2, you can select Vertex No. 2 in the 2nd combo box (currently Vertex No.3 selected). Then, MGRID will show the information about V1 and V2 and relation between them. You can select any 2 vertices and find their relative location.

2. Save and Use Dimension Information.

Step 1 While MGRID is still in Input->Info on Last Entry mode, please check the "Save 1st Set" button in Figure AR.2. Then, select Close and Drop Vertices button. MGRID will get out of the dialog. It also drops the entered matching vertices V1, V2 and V3. It is still in drawing mode.

In case you select the Close button instead of the Close and Drop Vertices button, MGRID will exit the dialog while it will keep all the entered vertices before it got into the dialog.

Step 2 We are going to enter a vertex by keyboard input. Type Alt+A (or select Input->Key In Absolute Location). MGRID will prompt you to enter the x and y-coordinates of the vertex. You can see the saved (x, y) coordinate of the V1 (No.1 Set Data in Info on Last Entry dialog) in the Saved Information group. You can use the Get Saved X button to retrieve the saved x for the x-coordinate and use the Get Saved Y button to retrieve the saved y for the y-coordinate of the vertex we are going to create.

Saved information from Info on Last Entry can be accessible in different dialogs. You can save some information and use it at your convenience. We just want to demonstrate this feature. You can stop here without going further on keyboard input. You can select Cancel and close the MGRID window now.

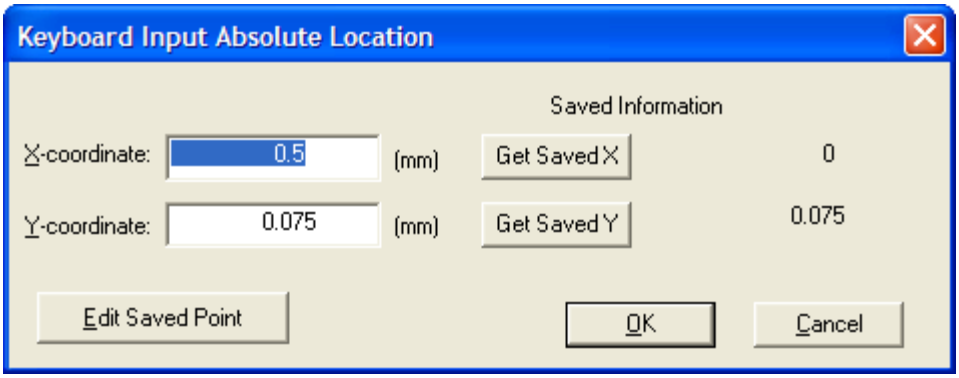


Figure AR.3 The saved data is accessible from Keyboard Input Absolute Location dialog.

Appendix AS. 3D View and Handling

We have improved the window system of MGRID significantly in IE3D 14. Before IE3D 14, MGRID is a single-document interface (SDI). The 3D view is an add-on and it is covering the polygon editor. It makes it inconvenient to manage multiple windows. On IE3D 14, we have changed MGRID to multiple-document interface (MDI). It allows us to open multiple windows to render different data to you. The multiple windows make MGRID much more convenient to you.

The 3D view window is significantly improved in IE3D 14. Its handlings are different from before. The tool bar on the 3D view window is still the same as before (see Figure AS.1). You can zoom, change the light source, the background color etc. You can also change the zoom factors for any of the X, Y and Z directions. The main differences are in the mouse-related operations.

On IE3D 14, you press down the left button and move the mouse to rotate the structure. To zoom in/out a structure, you press down Ctrl and move the mouse to define the zoomed region. The above procedures are still the same as before. You can use the arrow keys, the Home key and End key to rotate the structure in 6-directions. This is also the same as before. Another simple way to zoom the structure is to use the mouse-wheel, which were used for panning in Y-direction. To pan the structure on IE3D 14, you press down SHIFT and move the mouse. Right mouse button is used to bring up the pop-up menu on IE3D 14. The new configuration is more convenient to the users.

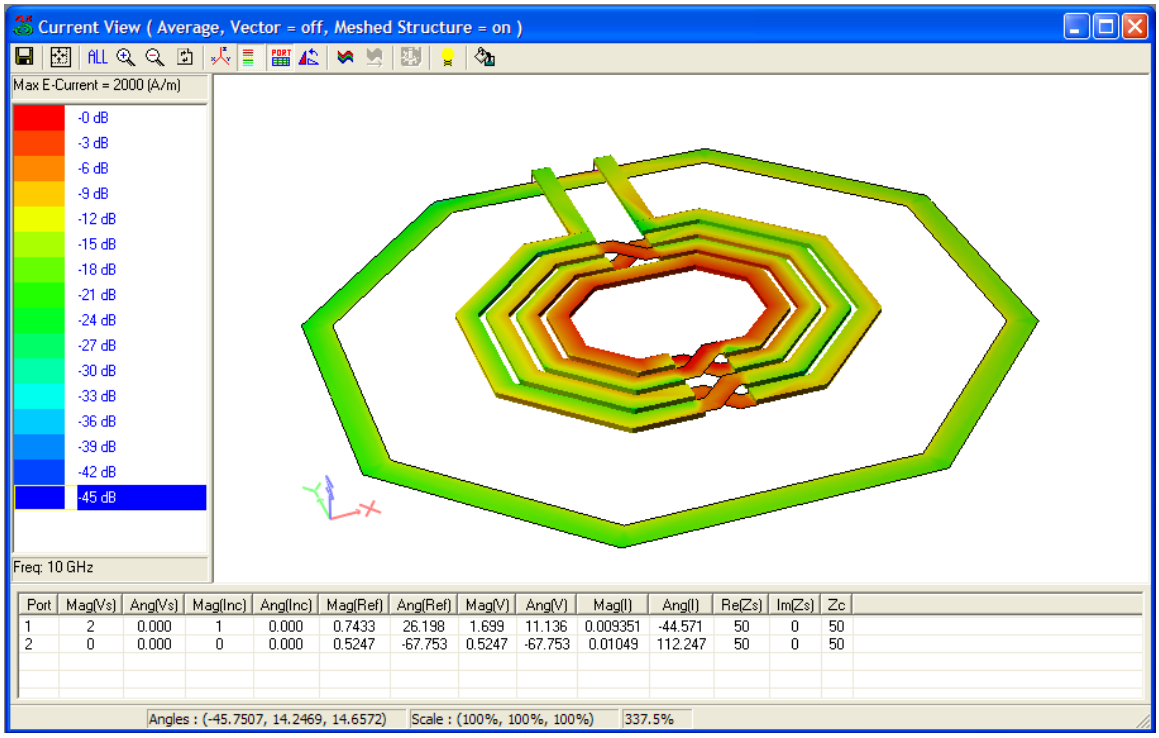


Figure AS.1 The 3D View on Current Distribution.

Appendix AT. The Layer Window and Selection of 3D Objects

Most polygons involved in IE3D are 2D or horizontal polygons on some layers with constant Z-coordinates. 3D polygons or polygons with not all vertices on a constant Z-coordinate are common in an IE3D structure. The layer window and its control help us to visualize and control the polygons in an IE3D structure. This appendix concentrates on the layer window and its control and selection of 3D objects.

Take the file in .\ie3d\sample\bridge.geo discussed in Chapter 4 as an example. Please open it on MGRID and open the 3D view for it.

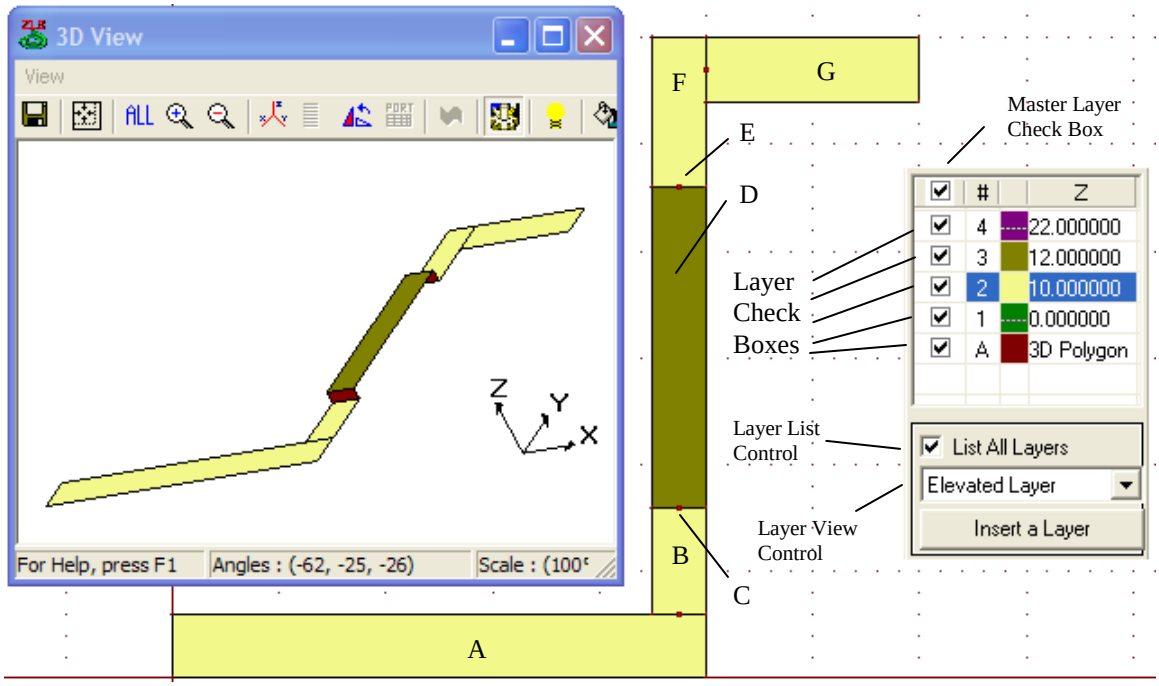


Figure AT.1 The .\ie3d\practice\bridge.geo and its Layer Window.

The structure contains of 7 polygons: polygons A, B, F and G on No.2 Layer at Z = 10; polygons D on No.3 Layer at Z = 12; polygons C and E are 3D polygons with vertices on No.2 layer and No.3 layer.

There is a layer check box on each of the layer in the Layer Window. The check box is used to control whether a selection has a focus on the layer. There is a Master Layer Check Box. It is used to check and uncheck all the Layer Check Boxes. There are 5 layers in this example. The layer A is always the 3D Polygon layer. It is actually not a layer. However, it is more convenient to call it as 3D Polygon layer and consider it as a special layer.

One of the layers is the input focused layer. For the picture in Figure AT.1, the No.2 Layer at Z = 10 is the focused layer. Any mouse input on the top main window is on the focused layer. In case the 3D Polygon layer is the input focused layer, MGRID will not respond to mouse input. You will need to use keyboard input (Input->Key In Absolute Location and Input->Key In Relative Location) to entry vertices or use other editing commands to enter polygon(s).

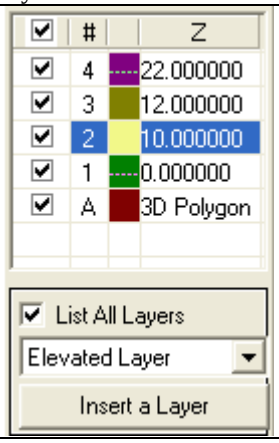
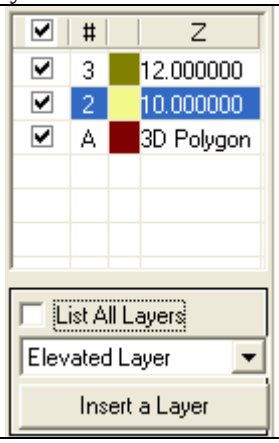
Please understand that there are two layer systems in IE3D: One layer system is the substrate layer system defined in the Basic Parameters dialog. The other layer system is the polygon (or metallic) layer system shown in the Layer Window. The substrate layer system defines how the substrate stickups and the layers and their properties are critical to the EM properties of the structure. The polygon layer system just

try to display where some 2D polygons, vertices and substrate interfaces are. The polygon layer system is managed automatically by MGRID. There might be some redundant polygon layers and you may not see any polygons on them. It is ok and you do not need to worry about it. On IE3D, polygon layers can be inserted inside a substrate layer. If you want to enter some polygon on a specific Z-coordinate, you can just select the Input a Layer button in the Layer Window (see Figure AT.1) and provide the Z-coordinate. MGRID will create a new polygon layer in the Layer Window for you. You should not try to define a substrate with the top surface at the Z-coordinate in the Basic Parameters even though it is not wrong to do so.

As it is mentioned in the above paragraph, polygon layers are those Z-coordinates where you may have some 2D polygons, some vertices of 3D polygons or substrate interfaces. For the bridge.geo example, the No.2 Layer at Z = 10 and No.3 Layer at Z = 12 are layers where 2D polygons are on. The No.1 Layer at Z = 0 and No.4 Layer at Z = 22 are where substrate interfaces are at. You may notice that there are some horizontal lines across the color bitmaps on the No.1 Layer at Z = 0 and No.4 Layer at Z = 22 in the Layer Window. The lines mean that that those two layers are the interfaces between infinite ground planes and regular substrates. Any 2D polygon on such a layer does not represent a piece of metal. Instead, it represents a slot on the infinite ground plane. It will be modeled using the so-called magnetic current model for the slot structure.

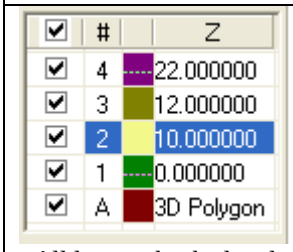
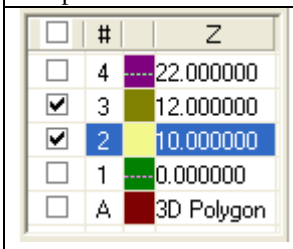
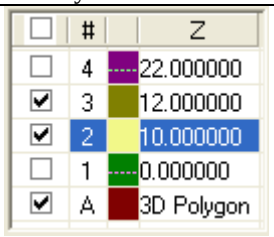
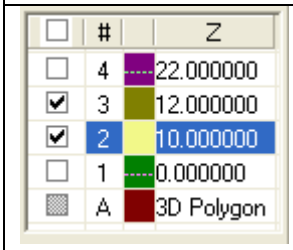
When we are building strong 3D structures such as helix antennas and wire bonds, we may create many 3D polygons with vertices at different Z-coordinates. Every distinct Z-coordinate will correspond to a layer in the Layer Window. We may end up with tens of hundreds of layers in the Layer Window while 2D polygons are available on a few of the layers. To make the Layer Window easier for the users, we have the Layer List Control check box. You can un-check it so that MGRID will only list those layers where 2D polygons are on. Table AT.1 shows the two states of the Layer List Control check box.

Table AT.1 The two states of the Layer List Control check box

Case	Layer List Control checked	Layer List Control unchecked
Layer Window		
Explanation	When the Layer List Control check box is checked, all the polygon layers are listed in the Layer Window.	When the Layer List Control check box is unchecked, only those layers where 2D Polygons are on are listed. There are 2D polygons on Z = 10 and 12 and the layers are listed.

The Layer Check Boxes are used to control how we can select objects in the selection modes (Edit->Select Polygon, Edit->Select Polygon Group and Edit->Select Vertices) and Port->Port for Edge Group mode. Table AT.2 documents most of the different situations. Please note that the Port->Port for Edge Group mode has the same behavior as the Edit->Select Vertices mode except it require two consecutive vertices (or an edge) of a polygon to be selected.

Table AT.2 The states of Layer Check Boxes and how they affect the scope of selection using mouse.

State	Selection Mode	Description
 All layers checked and Input Focus at Z = 10	Select Polygon	You can select any polygon at all layers. However, the polygon at the focused layer (Z = 10) has priority when you click at more than 1.
	Select Polygon Group	You can select any polygons at all layers by windowing them.
	Select Vertices	You can select any vertices at all layers by windowing them.
 Only No.2 and No.3 layers checked.	Select Polygon	You can select any polygon at the checked layers. The polygon at the focused layer has priority when you click at more than 1.
	Select Polygon Group	You can select any 2D polygon on the checked layers (No.2 and No.3).
	Select Vertices	You can select the vertices at the checked layers (No.2 and No.3) of any polygons.
 No.2, No.3 and 3D layers checked.	Select Polygon	You can select any polygon on the checked layers (No.2, No.3 and 3D). The polygon at the focused layer has priority.
	Select Polygon Group	You can select any 2D polygon on the checked layers, and any 3D polygon with all its vertices on the checked 2D layers (No.2 and No.3).
	Select Vertices	You can select the vertices at the checked 2D layers (No.2 and No.3) of any polygons.
 No.2 and No.3 layers checked. 3D layer grayed.	Select Polygon	You can select any polygon on the checked layers (No.2, No.3 and 3D). The polygon at the focused layer has priority.
	Select Polygon Group	You can select any 3D polygon with all its vertices on the checked 2D layers (No.2 and No.3). However, no 2D polygons can be selected.
	Select Vertices	You can select the vertices at the checked 2D layers (No.2 and No.3) of any polygons.

The Layer Display Control combo box is used to control how we display the polygons in the top main window. We use the structure in .\ie3d\samples\bridge2.geo as our example. A triangle is located at No.2 Layer at Z = 10. A rectangle is located at No.3 Layer at Z = 12. A 3D polygon spans from Z = 0 to Z = 22. You may see two thick orange lines on two edges of the 3D polygon if you have checked the View->Show Ground Connection. The thick orange lines indicate the edges are connected to the infinite grounds. How can we identify a thick orange line is on the infinite ground at Z = 0 or Z = 22? How can we identify the layers where the vertices of a 3D polygon are at? Basically, we can not display so much information visually. We have to select the polygon or parts of its vertices and use the Edit->Object Properties to know what it is.

For the discussion on how the Layer Display Control combo box affects the display, we will only concentrate on the additional triangle and rectangles. Table AT.3 document how the states of the Layer

Display Control combo box and the Input Focus on the layers are affecting the appearance of the polygons on the top view window.

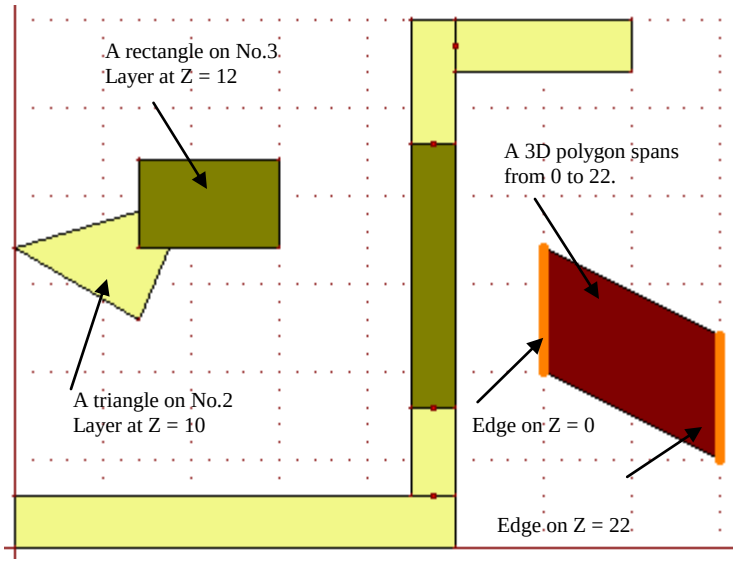
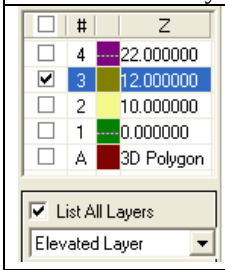
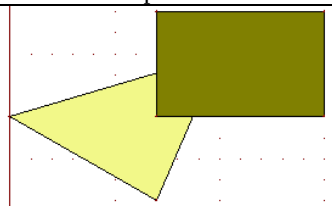
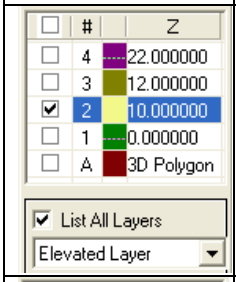
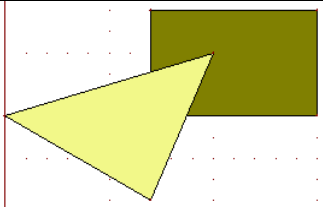
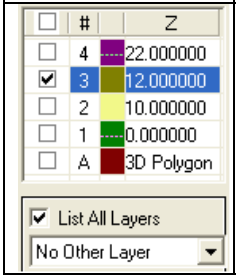
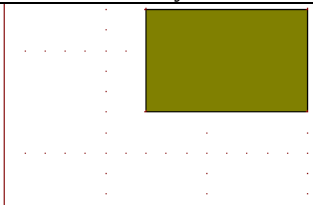
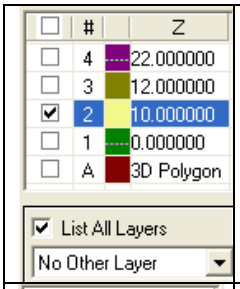
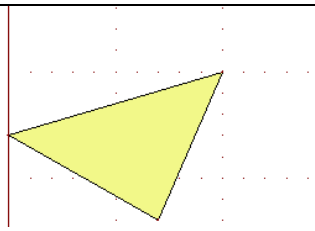
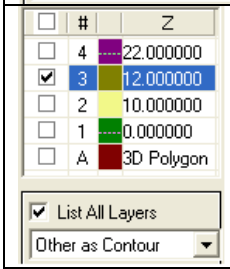
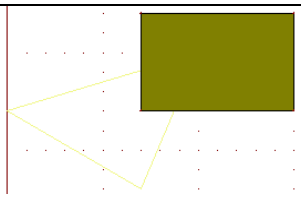
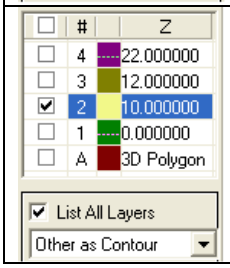
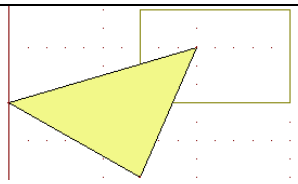


Figure AT.2 The polygons in .\ie3d\samples\bridge2.geo.

Table AT.3 The states of the Layer Window and their corresponding display on the top view.

Layer Window		Top View
	Input Focus at Z = 12 and Layer Display Control is Elevated Layer.	 The rectangle on Z = 12 is covering the triangle on Z = 10 because Z = 12 is the focused layer.
	Input Focus at Z = 10 and Layer Display Control is Elevated Layer.	 The triangle on Z = 10 is covering the rectangle on Z = 12 because Z = 10 is the focused layer.
	Input Focus at Z = 12 and Layer Display Control is No Other Layer.	 Only the rectangle on the focused layer at Z = 12 is displayed.

	Input Focus at Z = 10 and Layer Display Control is Elevated Layer	 Only the triangle on the focused layer at Z = 10 is displayed.
	Input Focus at Z = 12 and Layer Display Control is Other as Contour.	 The rectangle on Z = 12 is displayed as a solid while the triangle on Z = 10 is displayed as wire frame.
	Input Focus at Z = 10 and Layer Display Control is Other as Contour.	 The triangle on Z = 10 is displayed as a solid while the rectangle on Z = 12 is displayed as wire frame.

Appendix AU. EM Optimization and Different Schemes

IE3D allows automated EM optimization. The topic is discussed in different chapters and Appendices and we will not discuss how we perform EM optimization here. We will focus on the different optimization schemes implemented into IE3D. There are two groups of optimization schemes: (1) Local optimizer; (2) Global optimizes. They have their own advantages. Table AU.1 documents the different features of the two groups of optimizers.

Table AU.1 Local optimizers and Global optimizers

Feature	Local Optimizer	Global Optimizer
Typical Schemes	Powell, Simplex, Conjugate Gradient	Random, Genetic
Speed & Robustness	They are normally fast when the error function has one single local minimum exists. They may fail when the error function has more than one local minimum. We can say they are more efficient if we can find a solution. However, the chance for failure can be much higher.	They are relatively relative slow with many calls to the error functions. However, they normally can always approach the right solution with increasing number of calls to the error functions. In the other word, they are more robust than local optimizers.
Requirements on Error Function	Normally, they require the error function has the 1st order derivative. Powell does not require the user of derivatives. However, smooth error function certainly helps the convergence.	They normally do not have special requirement on the error function.
Suitability	They are normally more suitable to those error functions with good behaviors (smooth and having one local minimum).	They are more suitable to general cases especially when the error function is very complicated and it has more than one local minimum.

On the IE3D, we have implemented Powell Optimizer, Random Optimizer and Genetic Optimizer. Besides, we also implement a novel optimizer called AdaptiveEM optimizer. The AdaptiveEM optimizer combines the advantages of local optimizers and global optimizers. It can allow us to achieve optimization goals with much less calls to the error function.

An optimization basically involves many calls to the error function and we try to detect when we can achieve minimum error function. For an EM optimization, a call to an error function is basically an EM simulation. An EM simulation is a relative slow process. The key to efficiency is to reduce the number of calls to the error function while we can find the point where we can achieve the minimum value of the error function. Local optimizers normally do not need many calls and they can find a local minimum much easier than global optimizers. The problem is that the found local minimum may not be the global minimum we are interested in. It is normally hard for a local optimizer to get out of the "trap" from the local minimum. Global optimizers normally require large number of calls to the error function so that it can sort out where the global minimum is. Normally, the more runs you let a global optimizer to do; it will get closer to the global minimum. The more runs you run, the higher chance a global optimizer will find the global minimum for you.

The AdaptiveEM optimizer basically combines the good things from local optimizers and global optimizers. In some sense, AdaptiveEM is very similar in concept to the Space Mapping scheme. Space Mapping scheme uses a coarse model and a precise model. It tries to find the global minimum of the precise model while it performs the optimization on the coarse model. It has many calls to the error function of the coarse model while it only needs limited number of calls to the precise model. For an EM optimization, the precise model is an EM simulation while the coarse model is a circuit model. It can significantly reduce the number of EM simulations to find the global minimum of the error function which is an EM model.

As Space Mapping is a good scheme, it always requires a coarse model for a given precise model (or an EM model). In fact, we may not be able to find a good circuit model for every general EM problem. Such a fact limits the use of Space Mapping for EM optimization to some limited number of problems.

On the AdaptiveEM optimizer, we also build a “coarse” model and the precise model. The precise model is basically the EM model. The “coarse” model is basically an automatically created model based upon the finished EM simulations. When we are optimizing a structure, we are gradually increasing the number of EM simulations. For each additional EM simulator, we will automatically create a better coarse model for it. We can consider our coarse model is an adaptive coarse model and this coarse model does not require any knowledge to the EM structure. It only requires us to provide the EM simulation results to it. By the time the adaptive coarse model is approaching the EM model, the optimization is finished with the best results you can achieve. The scheme combines local optimizers and global optimizers in the search of the global minimum. Normally, it can beat a local optimizer for simple error functions and global optimizers for complicated error functions in both efficiency and accuracy.

Appendix AV. Radiation Pattern File and General Radiation Pattern File

Two types of pattern files are involved in IE3D and they serve different purposes. On IE3D, we can setup a simulation to save the current distribution (.cur file) and find the radiation pattern (.pat file). The .pat file is the radiation pattern with specified excitation and termination. We can also generate the .pat file from the .cur file on MGRID in the post-processing mode. When we change an excitation, we have to re-generate the .pat file. The pattern calculation process (from .cur file) can be time consuming. For a small or even medium size antenna, the pattern calculation may take longer time to perform the EM simulation. For this reason, we have introduced the General Radiation Pattern File (.mpa). When you find the .pat file, we automatically save the .mpa file. The .mpa file contains the pattern with all excitation possibilities. You can derive the .pat file of specified excitation from the .mpa file without repeated pattern calculation. Such a feature is implemented into the PATTERNVIEW in the Edit->Process General Pattern command. It takes no time to find a .pat file from the .mpa file no matter how big the pattern file is. The comparison between .pat file and .mpa file is shown in Table AV.1.

On the IE3D 11, we also introduce the concept of Conjugate Match Gain and Conjugate Match Efficiency. The definitions of the original Gain and Antenna Efficiency are for constant wave sources while the Conjugate Mach Gain and Conjugate Match Efficiency are for constant voltage source. A constant wave source generates constant incident wave. The reflected wave from the antenna is completely absorbed and it will not contribute to the incident wave. The original Antenna Efficiency measures how much of the power from the incident wave is radiated out. For some applications, you may have a constant voltage source. With the antenna impedance and the constant voltage source given, we can adjust the source impedance to achieve maximum input power for the antenna. The maximum input power can be achieved by conjugate matching the source impedance and the antenna impedance no matter which one you want to change. In such a case, the antenna Input Power is 50% of the power delivered by the constant voltage source. The other 50% is consumed by the source impedance. For this reason, the Conjugate Match Efficiency defined in IE3D is always less than 50% because the radiated power can never exceed the Input Power and the Input Power can never exceed 50% of the constant voltage source power.

Besides the above mentioned usage, we also use the .mpa file to find the transfer function between transmitting antenna (TX) and receiving antenna (RX). It is discussed in Appendix AJ.

Table AV.1 The comparison between the two pattern file types.

Feature	Radiation Pattern File (.pat)	General Pattern File (.mpa)
Data Stored	It stores the radiation pattern data with the excitation and termination defined. It provides the pattern information such as gain, directivity and efficiency etc to the users.	It stores the radiation pattern data with all possible excitation and termination. However, the pattern data for a specific excitation and termination state is not readily available. You need to use the Edit->Process General Pattern command to create the radiation pattern with specified excitation and termination (.pat) file for the specific pattern data.
What is it good for?	The data describes the state of an antenna and the data is readily available.	Pattern calculation is a time consuming process. We had to perform a pattern calculation when we change the excitation and termination before we introduced the .mpa file. Finding a .pat file for a specific excitation and termination from an .mpa file takes no time. The introduction of .mpa file makes it possible to optimize the excitation and termination for specific performance.

Table AV.2 The definition of different pattern parameters in a radiation pattern file.

Parameter	Description
Incident Power	It is the net power of incident waves from all the ports. There are reflected waves. The reflected waves may get reflected again from the source. The incident wave is the considered as the net incident wave in the steady state.
Reflected Power	It is the power from the reflected waves on the ports.
Input Power	It is the net input power into the antenna. It is defined as: Input Power = Incident Power – Reflected Power.
Radiated Power	Radiated Power = Input Power – Loss on the Antenna
Loss on the Antenna	The loss on the antenna includes surface wave loss, metallic loss and dielectric loss. Many users may wonder whether we can separate the different losses. They are inseparable because they are related. For example, surface wave loss decays as $1/\sqrt{\rho}$ in lossless cases only. In a practical antenna, surface wave is eventually lost as metallic loss or dielectric loss or converted back to radiation wave for finite substrates and ground plane.
Radiation Efficiency	It is defined as: Radiation Efficiency = Radiated Power / Input Power It is to measure how much energy is radiated and how much is lost inside the antenna from the net input power.
Antenna Efficiency	It is defined as: Antenna Efficiency = Radiated Power / Incident Power It considers the reflection as loss to the antenna. The Antenna Efficiency definition is good for constant wave source which generates a constant incident wave independent of the reflected wave.
Conjugate Match Efficiency	It assumes the source impedance is the conjugate of the antenna impedance. It is good for constant voltage source. The power of the constant voltage source is always twice as large as the Input Power in conjugate matching. Conjugate Match Efficiency is always 50% of the Radiation Efficiency.
Directivity	It is a measure on how much an antenna concentrates on the radiation at specific angles. It is defined as: $\text{Directivity} = 4\pi \lvert E(\theta, \varphi)\rvert^2 / \int d\theta' \int d\varphi' \sin(\theta') \lvert E(\theta', \varphi')\rvert^2$ Where $\lvert E(\theta, \varphi)\rvert$ is the relative E-field density at specific angles or the radiation pattern distribution of the antenna. Please note that the Directivity of an antenna is only dependent upon the shape of the pattern or the $E(\theta, \varphi)$ at all the angles. It is independent of the matching and losses of the antenna. Its unit is dBi means the dB value compared to an ideally isotropic pattern or a pattern with constant $\lvert E(\theta, \varphi)\rvert$.
Gain	It is defined as: Gain (dBi) = Directivity (dBi) – Loss on the Antenna (dB) – Mismatch Loss (dB)
Conjugate Match Gain	It is defined as: Conjugate Match Gain (dBi) = Directivity (dBi) – Loss on the Antenna (dB) – 3 dB The 3 dB loss is due to the conjugate match. The conjugate source impedance absorbs the same amount of energy delivered to the antenna.
Total Field Properties	It is the properties of the antenna with the entire field considered.
Theta Field Properties	It is the properties of the antenna with E-theta field considered.
Phi Field Properties	It is the properties of the antenna with E-phi field considered.
Left Hand Circular Field	It is the properties of the antenna with LH circular polarized field considered.
Right Hand Circular Field	It is the properties of the antenna with RH circular polarized filed considered.

Table AV.3 The radiation pattern (.pat) file in .\ie3d\samples\pfed_ant.pat.

Content	Section
10 5.0 C:\MentorGraphics\<latest_release>IE3D\SDD_HOME\IE3D\ie3d\practice\output\pfed_ant.pat	1
	2
2 2 1 0 1 C:\MentorGraphics\<latest_release>IE3D\SDD_HOME\IE3D\ie3d\practice\pfed_ant.ect	3
0 0.00e+0 0.00e+0	4
0 0.00e+0 0.00e+0 0.00e+0	5
1 0.00e+0 0.00e+0 0.00e+0	6
0 1.00e+0 1.00e+0 0.00e+0 0.00e+0	7
37	8
1 0.00e+0	9b
2 5.00e+0	
3 1.00e+1	
...	
36 1.75e+2	
37 1.80e+2	9e
37	10
1 0.00e+0	11b
2 1.00e+1	
3 2.00e+1	
...	
35 3.40e+2	
36 3.50e+2	
37 3.60e+2	11e
9	12
2.66e+0 1.58508527816755e+0 0.00e+0	13
5.00e+1 0.0 8.77e-1 -4.96e+1 1.43e-2 -1.14e+2 6.73e-1 -7.85e+1 4.34e-1 -1.07e+0 6.73e-1 -7.854e+1 5.00e+1 0.0	14
5.00e+1 0.0 1.36e+0 -9.19e+1 2.12e-2 -1.54e+2 1.03e+0 -1.18e+2 6.42e-1 -4.48e+1 1.03e+0 -1.189e+2 5.00e+1 0.0	15
0.00e+0 0.0 0.0 0.0	16
8.184e-3 1.530e-2 9.284e-3 8.814e-1 5.348e-1 9.28e-3 0.00e+0 0.00e+0	17
7.089e+0 6.108e+0 -9.815e-1 4.37e+0 3.390	18
0.00e+0 2.50e+2 1 8.45438584945049e+1 1 9.48944380583036e+1	19
1 0.00e+0 1.90e+2 1 8.56218721974612e+1 1 9.24226191372571e+1	20
0.00e+0 0.00e+0 5.67312190938750e-1 3.78314349277164e+1 8.23502809960153e-1 -1.75400411031064e+0	21b
5.00e+0 0.00e+0 5.67800675252225e-1 3.73170934863606e+1 8.19897847553759e-1 -1.81970215486891e+0	
1.00e+1 0.00e+0 5.62817868393506e-1 3.67388685705140e+1 8.09132326171367e-1 -2.01484914986040e+0	
...	
1.65e+2 0.00e+0 7.83106636905141e-2 -8.89599579124063e+1 9.27730154811359e-2 4.34371377921999e+1	
1.70e+2 0.00e+0 7.64730888525261e-2 -9.03040193191381e+1 9.64902177898383e-2 4.66182054505988e+1	
1.75e+2 0.00e+0 7.32524040995489e-2 -9.10839264830872e+1 9.89442939911981e-2 4.84380610649934e+1	
1.80e+2 0.00e+0 6.86497371693534e-2 -9.13142270382115e+1 9.97911076737005e-2 4.90393045148696e+1	
0.00e+0 1.00e+1 6.75078107584125e-1 3.00738489240731e+1 7.37746263061109e-1 -6.63522417234878e+0	
5.00e+0 1.00e+1 6.75531906034220e-1 2.96814973632528e+1 7.34464359409077e-1 -6.70016508665055e+0	
1.00e+1 1.00e+1 6.69214110184913e-1 2.92319586593806e+1 7.24704674984902e-1 -6.90486746516837e+0	
1.50e+1 1.00e+1 6.56371016113287e-1 2.87202668708165e+1 7.08798609128685e-1 -7.25158316600397e+0	
...	
1.70e+2 3.60e+2 7.64730888525262e-2 -9.03040193191381e+1 9.64902177898384e-2 4.66182054505989e+1	
1.75e+2 3.60e+2 7.32524040995489e-2 -9.10839264830872e+1 9.89442939911982e-2 4.84380610649935e+1	
1.80e+2 3.60e+2 6.86497371693537e-2 -9.13142270382115e+1 9.97911076737004e-2 4.90393045148697e+1	21e
2.69e+0 1.63670191197304e+0 0.00e+0	22b
...	…

Table AV.4 The explanation of the file .\ie3d\practice\pfed_ant.pat.

Section	Explanation
1	The "10" and the "5.0" are the version numbers. Please leave them as they are. The following string is the file name of this file.
2	This section is an empty string. It is for the file ID when we compare patterns.
3	"2 2 1 0 1 C:\MentorGraphics\<latest_release>IE3D\SDD_HOME\IE3D\ie3d\practicepfed_ant.ect" means total 2 global ports, total 2 net ports, source type = 1 or constant wave source, 0 array element and using the feed network from pfed_ant.ect file.
4	"0 0 0" are no plane wave, plane wave theta = 0 and plane wave phi = 0.
5	"0 0 0 0" are for 2nd ground plane and any rotation. Please leave them as they are.
6	"1 0 0 0" are for the length unit and shifting in X, Y and Z. "1" means mm. The "0 0 0" means no shifting.
7	"0 1 1 0 0" are for the 2nd ground. "0" means no 2nd ground. The "1 1 0 0" are for the Er, Mur, Sigma and Height.
8	"37" means 37 theta values.
9	The 37 theta values from 0 to 180.
10	"37" means 37 phi values.
11	The 37 phi values from 0 to 360.
12	The "9" means 9 frequency points.
13	The 1st frequency at 2.66 GHz. The "1.58508…" is a magnitude value for the pattern referred as Emax later. The 3rd value is not critical.
14	For the No.1 port, Zc = (50, 0), Mag[V] = 0.877 Ang[V] = -49.6. Mag[I] = 1.43e-2, Ang[I] = -114, Mag[a]=0.673, Ang[a]=-78.5, Mag[b]=0.434, Ang[b]=-1.07, wave source Mag[a0]=0.673, Ang[a0]=-78.5. Zi=(50, 0) for the source.
15	It is for the No.2 port.
16	Plane wave Mag[E-theta], Ang[E-theta], Mag[E-phi] and Ang[E-phi] if there is any.
17	Radiated Power = 8.184e-3 w, Incident Power = 1.530e-2 w, Input Power = 9.284e-3 w, dummy, dummy, Power at the Feed = 9.284e-3 W, Plane Wave Incident Density = 0 w/(m*m), Max RCS = 0 (for plane wave excitation only).
18	Total Field Directivty = 7.089 dBi, Dominant Circular Field Directivity = 6.108 dBi, Dummy, Total Field Gain = 4.37 dBi, Dominant Circular Field Gain = 3.390 dBi. They will be re-calculated in PATTERNVIEW automatically.
19	Maximum at Theta = 0 and Phi = 250. The next 4 numbers are for the 3 dB beam width. They will be re-calculated automatically in PATTERNVIEW automatically.
20	"1" means it is RHS dominant. The other parameters are similar to the ones in Section 19. However, they are for the RHS circular field. They are re-calculated in PATTERNVIEW.
21	The section describes the normalized E-field at each theta and phi angle. Theta Phi Mag[E-theta] Ang[E-theta] Mag[E-phi] Ang[E-phi] …
22	It is the 2nd frequency similar to the Section 13. You will have the similar Section 14 to Section 21 for the 2nd frequency. Totally, we have 9 blocks from Section 13 to Section 21 in the file.

The file format of the radiation pattern file (.pat) is documented in Tables AV.3 and AV.4. Many items in Section 17 to Section 20 are extra information. They can be derived from mainly the three sets of data: (1) The Emax in Section 13; (2) The normalized E-field in Section 21; (3) The excitation information in Section 14 to Section 16.

The normalized E-field and the Emax are defined from (AV-1) to (AV-4).

$$\text{Maximum} = \text{Max}[\sqrt{(|\text{E-theta}|^2 + |\text{E-phi}|^2)}] \text{ for all the angles} \quad \text{(AV-1)}$$

$$\text{Normalized E-theta} = \text{E-theta} / \text{Maximum} \quad \text{(AV-2)}$$

$$\text{Normalized E-phi} = \text{E-phi} / \text{Maximum} \quad \text{(AV-3)}$$

$$\text{Total Radiated Power} = 1/(2\eta) \cdot \text{Emax}^2 \cdot \int d\theta \cdot \sin(\theta) \int d\varphi \left(|\text{E-theta}|^2 + |\text{E-phi}|^2 \right) \quad \text{(AV-4)}$$

The integration over θ is from 0 to π and the integration over φ is from 0 to 2π. The E-theta and E-phi in (AV-4) are the normalized values. The η is the wave impedance in free space and it is 120 π. In the other word, we can use Emax and the normalized E-theta and E-phi to predict the far field as:

$$E(\theta, \varphi) = \text{Emax} \cdot [(\theta \cdot \text{E-theta} + \varphi \cdot \text{E-phi}) / r] \cdot \exp[j (\omega t - k r)] \quad \text{(AV-5)}$$

$$H(\theta, \varphi) = r \times \mathbf{E}(\theta, \varphi) / \eta \quad \text{(AV-6)}$$

Where $E(\theta, \varphi)$ and $H(\theta, \varphi)$ are the vector E-field in v/m and H-field in A/m. The (θ, φ, r) are the unit vectors in the spherical system. The r is the distance from the origin to the field point in meter. All the other parameters about the pattern are derived from the three groups of basic parameters.

Appendix AW Finding Equivalent Circuit of Coupled TLNs and Interconnects

Coupled TLN and interconnects are used in modern IC and PCB much. At low frequency, the coupled interconnects can be modeled as some RLC equivalent circuit in Figure 9.3. The SPICE equivalent circuit can be extracted using the schemes discussed in Chapter 9.

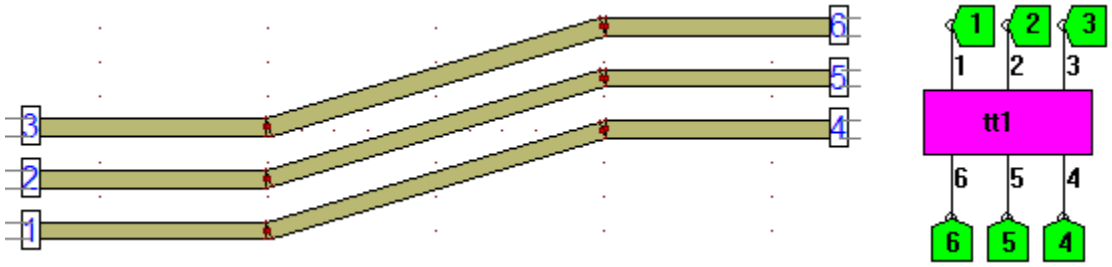
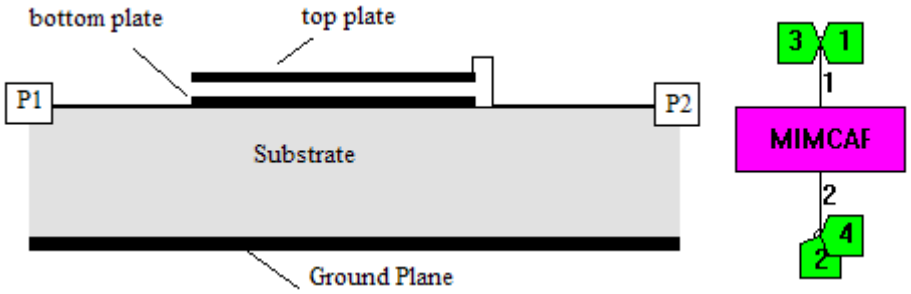


Figure AW.1 The structure consisting of three coupled traces.

For example, we have a structure consisting of 3 coupled traces shown in Figure AW.1. We can simulate it on IE3D to get the 6-port s-parameters. Then, on MODUA, we define the ports as shown in Figure AW.1. We can select Process->LC Equivalent command on MODUA to find the RLC equivalent circuit for it. If we choose one single frequency (normally at low frequency), we will be able to find the RLC equivalent circuit at the frequency. We can select File->Save SPICE File to save the RLC equivalent circuit in SPICE format. If we choose multiple frequency points, the extracted RLC equivalent circuit is frequency dependent and we can not fit it into SPICE. However, we can still save it into some ASCII file format.

In case we are modeling a capacitor shown in Figure AW.2a, its equivalent circuit is basically a PI-network similar to the one in Figure 6.10 except there should be a shunt resistor for each capacitor. How can we extract the RC-equivalent circuit from it? We have not implemented a special scheme for it. However, we can use the Process->LC Equivalent Circuit command or the C-Equivalent Circuit command to extract the equivalent circuit. What we should do is to define 4 ports on the 2-terminals of the 2-port s-parameters. Then, we can apply either LC-Equivalent command or C-Equivalent command on it. In case it is LC-Equivalent command, the solved series L and R will be virtually 0. The solved shunt RC will be precise.



(a) an MIM Capacitor (b) MODUA setup for RC Extraction.

Figure AW.2 An MIM Capacitor and the MODUA setup for RC-Equivalent Circuit.

Appendix AX. Back Simulation, Separating and Removing S-Parameters

Shift of reference plane used to be discussed in this Appendix. Starting from IE3D 14, we have devoted Appendix BJ to the topic. We will discuss other involved topics here.

Shift of reference plane allows us to shift the reference plane of a port or the reference planes of multiple coupled ports to other locations. They are for convenience of actual design of microwave circuits. However, shift of reference planes only allow us to shift the reference planes on some uniform port structures. We are not able to shift the reference plane of a structure with non-uniform line. For the structure in Figure AX.1. A Y-junction has an additional bended line on each other. We are not able to shift the reference plane from the end of the bended line to the start point of the bended line using the shift of reference plane scheme discussed in Appendix BJ. However, we can achieve such a goal using the Back Simulation scheme on MODUA. In this appendix, we will discuss Back Simulation, Separating S-Parameters and Removing S-Parameters. The three features are related.

1. Back Simulation:

Assuming we have a 3-port structure W in Figure AX.1. We can break W into 4 sub-sections A, B, C and D in Figure AX.1.

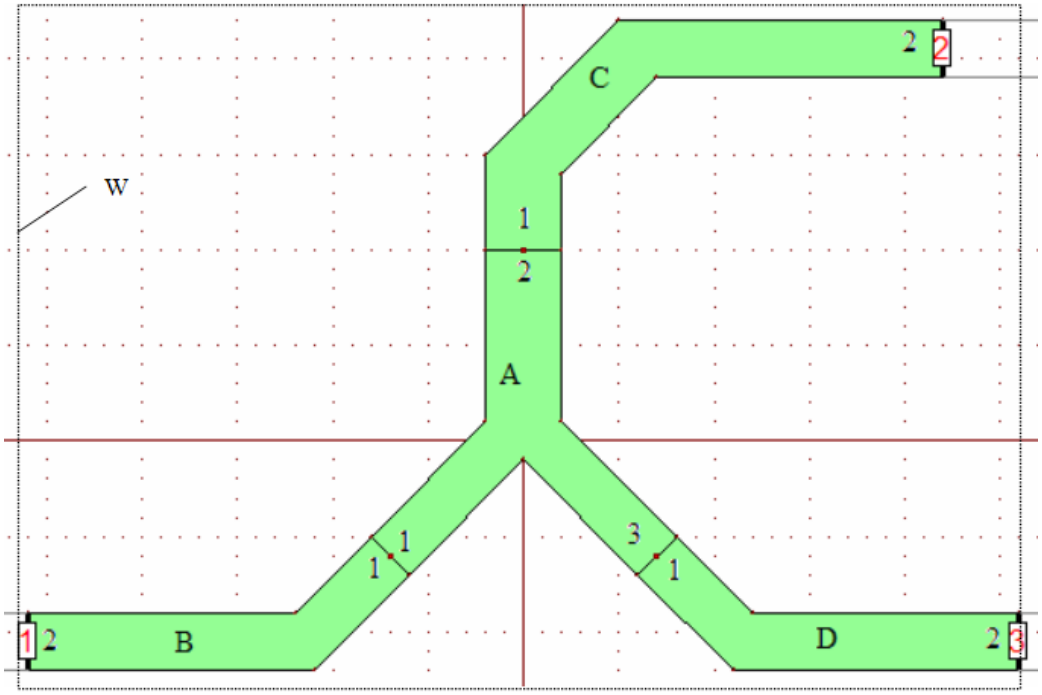


Figure AX.1 A whole circuit W is broken down into A, B, C and D.

From microwave network theory, we know the following procedure: When there is little coupling among the A, B, C and D, we can obtain accurate s-parameters of W from those of A, B, C and D.

The Back Simulation is basically the reverse process. If we know the s-parameters of W, B, C and D, we would like to find the s-parameters of A.

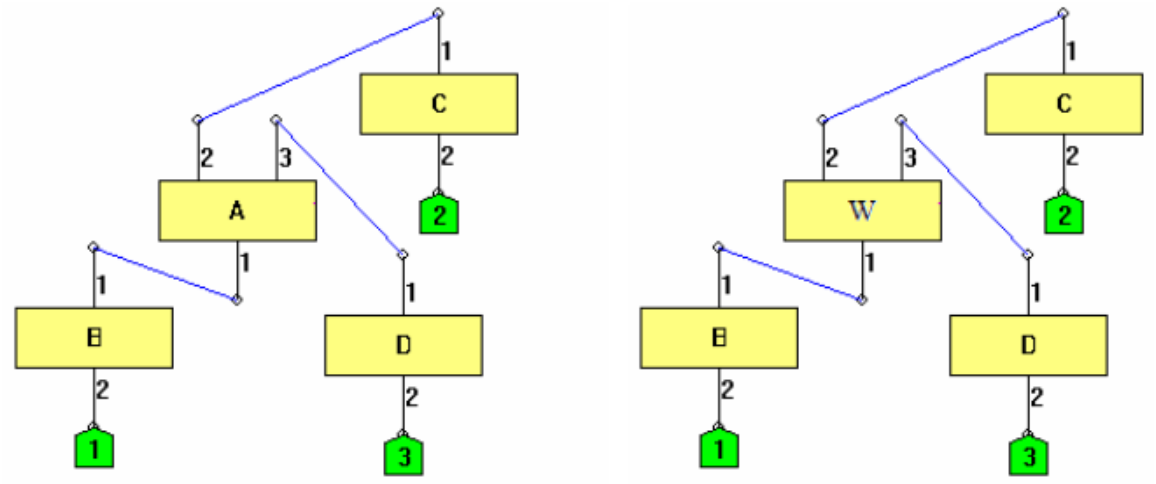
In practical applications, we may encounter a case that we cannot get the s-parameters of A accurately (The reason can be that we do not have enough space for the extension of port 1, or the port is

too close to discontinuities inside A). However, we can get the s-parameters of W, B, C and D. We can use the **Back Simulation** to find the s-parameters of A from those of W, B, C and D.

On MODUA, we can setup a design shown in Figure AX.2a to find the s-parameters of W from those of A, B, C and D. Please pay attention to the terminals of each sub-circuit. They are labeled by the numbers in Figure AX.1 and AX.2. The connection of the terminals between the modules is very critical. You cannot switch the terminals of B, C and D if they are not symmetrical for both Forward Simulation and Back Simulation.

Similarly, we can connect the s-parameters of W, B, C and D with W replacing the A in Figure AX.2a for the Figure AX.2b. We get the setup for finding the s-parameters of A from those of W, B, C and D. The ports or terminals in the Back Simulation can be quite confusing. However, from the comparison between the Forward Simulation and Back Simulation, you can find the common points.

Another comment we would like to make is that the port numbering for the resulting s-parameter file for the **Back Simulation** follows that of module W. The actual ports defined on the terminals of B, C and D do not mean anything. When you perform a **Back Simulation** on MODUA, it is required that each terminal of W should be connected with a 2-port s-parameter module. In case you do not want to remove something out of a specific port, you just connect the s-parameter file of a perfect through line (S11=S22=0 and S12=S21=1) on it. You can create the s-parameter for a perfect through line using the **Process->Create S-Parameter File** command on MODUA.



(a) Setup for (Forward) Simulation for W (b) Setup for Back Simulation for A.

Figure AX.2. The schematic representations for (Forward) Simulation and Back Simulation.

Again, the **Back Simulation** on MODUA is very useful for shift of reference plane and de-embedding of complicated circuits (including isolated port structures and coupled port structures). For shift of reference plane, the B, C and D sub-circuits are basically uniform transmission line sections. The Back Simulation feature actually allows them to be arbitrary structures as long as they are not coupled to each other and A.

2. Separate S-Parameters:

The Back Simulation is a very useful feature. In fact, there are two similar and even more powerful commands on MODUA worth discussing: (1) Separate S-Parameters; (2) Remove S-Parameters.

The **Separate S-Parameters** is for the following purpose. There are 3 s-parameter files involved: A, B and C. Each of them has the same even number of ports. A typical structure example is shown in Figure AX.3. The schematic diagram is shown in Figure AX..4. Assuming the ports 4, 5, 6 of the A-module are connected to the ports 1, 2 and 3 of the B-module. The results will be the 6-port s-parameters for the C-module. Algebraically, we can describe it as: A + B = C. Assuming we know the s-parameters of B and the C, we would like to get the s-parameters of A. We can algebraically describe it as A = C – B. Such a feature is implemented into the MODUA as the **Process->Separate S-Parameter**s command. To use the command, we put the s-parameters of C in MODUA and define the 6-ports as shown in Figure AX.4b. Then, we add the 6-port s-parameter file of the B-module into **File->Parameter File Queue** as if we are doing comparison of s-parameters. Finally, we select the **Process->Separate S-Parameters** command on MODUA. The s-parameters for the A-module will be extracted and saved into a file with provided name.

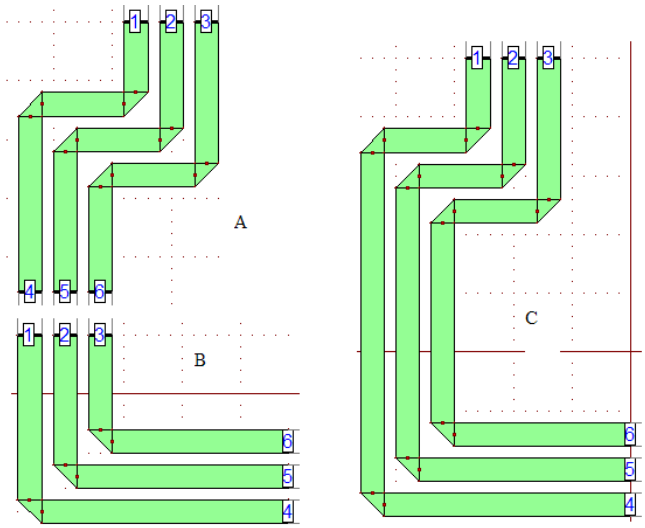
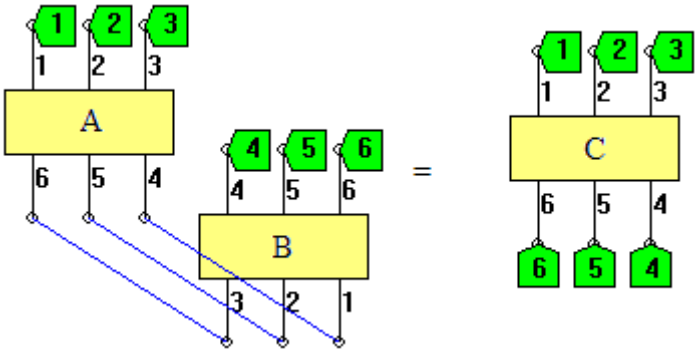


Figure AX.3 Three structures involved in a typical Separate S-Parameters command.



(a) A and B connected (b) C sub-circuit

Figure AX.4 The schematic representation.

The **Process->Separate S-Parameters** command on MODUA is a very general implementation. The s-parameter files of A or B are not limited to coupled transmission lines only. Each of them can be assumed as a black box. There are only two requirements: (1) The A, B and C have the same number of terminals and the number of terminals is an even number; (2) The A and B are not coupled.

3. Remove S-Parameters:

The scenario for the **Process->Remove S-Parameters** command on MODUA is quite similar to the **Process->Back Simulation** command. In fact, the **Remove S-Parameters** command can do whatever the

Back Simulation command can do and more. We also use a 3-port structure C separable into structures A with 3-ports and structure B with 6-ports as an example (see Figure AX.5). The schematic is shown in Figure AX.6. We can also represent the case as: A + B = C and A = C – B. Basically, we know B and C and we would like to get the A. Similar to the **Separate S-Parameters** command, we put the s-parameters of C onto MODUA and connect the 3 ports to its terminals. Then, we put the 6-port s-parameter file of B into the **File->Parameter File Queue**. Select the **Process->Remove S-Parameters** command and we will find the s-parameters of A saved in a specified file. The **Remove S-Parameters** command can do whatever the **Back Simulation** command can do. Basically, we can consider the B, C and D in Figure AX.1 as the B in Figure AX.5. For the **Back Simulation**, we assume there is no coupling between B, C and D. For the **Remove S-Parameters** command, it does not matter whether there is any coupling between the B, C and D. The two basic requirements for the **Remove S-Parameters** command are: (1) The structures A and B are not coupled. (2) The structure B has twice as many ports as the structure A.

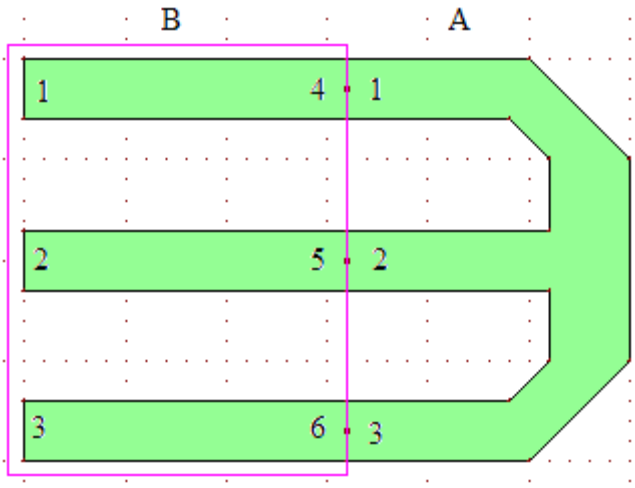


Figure AX.5 A 3-port structure C-separable into a 3-port structure A and a 6-port structure B.

We will finish the discussion on the **Back Simulation**, **Separate S-Parameters** and **Remove S-Parameters** commands on MODUA now. The 3 commands are very powerful for s-parameter de-embedding and extraction.

Appendix AY Equation-Based Geometry Modeling on IE3DLibrary

The default IE3D layout editor is MGRID. The basic geometry elements of MGRID are the vertices and polygons. MGRID allows you to edit geometry in the level of vertices and polygons. You can manage small details of a structure by changing the vertices and polygons. You can also define the locations of vertices as optimization variables so that you can perform EM optimizations on your structure. However, vertices and polygons are low level elements. Sometime, it is too tedious to edit low level elements for some high level geometry editing. For example, we can build a circle by some polygons on MGRID. However, it will be tedious to change the radius after the circle is built because the circle is no longer an object with its parameters accessible after it is built as polygons and vertices on MGRID.

To access high level geometry parameters, we have implemented the IE3DLibrary as the 2nd IE3D layout editor. On the IE3DLibrary, we implement a list of basic objects. Each object is parameterized. For example, a circle can be parameterized as its center location, its radius and number of segments for a polygon to approximate the circle. Each of the parameters can be changed. Each dimensional parameter can also be associated with some tuning variables. When tuning variables are changing, the dimensions associated can be updated simultaneously. In this way, we can control the shape of a structure consisting of a number of objects easily by controlling the values of the tuning variables.

We have included some examples in the context of this manual without detail steps of creating and some sophisticated structures on IE3DLibrary. Detailed documentation on IE3DLibrary can be found in the ie3dlibrary_manual.pdf in this installation.

Appendix AZ. IE3D Seamlessly Integrated into Microwave Office from AWR

An EM Socket is implemented to link the IE3D simulation engine and MGRID to the Microwave Office of Applied Wave Research. You can transfer layout geometry between IE3D and Microwave Office. You can also perform IE3D EM simulation directly from Microwave Office transparently.

For this initial release of the IE3D EM Socket for MWO, some limitations on the EMSight are still in the socket. For example, ports must be defined in the X, -X, Y and –Y directions. True thickness can not be built automatically. IE3D is using a non-uniform meshing. Meshing is critical to a successful IE3D simulation. On MGRID, we can display the meshing result easily. However, the display meshing command is not implemented in Microwave Office. The EMSight from Microwave Office is using the layout grids for the meshing and its meshing is normally more than enough. However, for IE3D, the meshing is controlled by the Highest Application Frequency (Fmax) and Cells per Wavelength (Ncell). We need to display meshing from time to time in order to optimize the meshing so that we can use the cells to yield high accuracy results efficiently. You are suggested to use MGRID to check the meshing of the geometry created by Microwave Office from time to time to make sure the meshing is fine enough. You are able to invoke MGRID as native layout editor on the Microwave Office.

Appendix B. IE3D Geometry File Formats

In this appendix, we will illustrate the geometry file formats for the IE3D. The file formats are changed in the last few versions. The three recent file formats are listed in Table B.1. We will concentrate on the IE3D 9.0 and IE3D XML 2.0 formats. The file format was slightly changed in the IE3D 9.3 release due to the fact that we introduced a new formulation to model thick traces. We will indicate the minor difference in the IE3D 9.3 format when we discuss the IE3D 9.0 format.

Table B.1 The different file formats in IE3D.

Format	Description
IE3D 9.0	It is used in IE3D 9.x and IE3D 10.0. It is in ASCII format.
IE3D XML	It is used in IE3D 10.1 and 10.2. It is the initial release of the IE3D file format in XML format. XML format uses tags to identify and organize different sections of data. It is very versatile and flexible. However, the file size is significantly larger than the ASCII file format. It is also much slower in parsing the file when it is loaded. The IE3D XML file format may be too small when the structure.
IE3D XML 2.0	It is used in IE3D 11. We try to re-organize the data so that the file size is smaller than the IE3D XML format. Also, file loading does not require multiple runs of parsing the XML string so that loading large file is much faster than the IE3D XML format.

The most difficult portions of a geometry are the ports. We try to document a geometry file with different extension ports and vertical localized ports. The structure we want to illustrate is shown in Figure B.1. It is saved in .\ie3d\samples\u_3ports.geo. It is a U-shaped structure with 2-arrms. A differential extension port is defined on the two arms (port +1 and -1). A vertical localized port (port 2) is defined between the 2 levels of one arm and another vertical localized port (port 3) is defined between the 2 levels of the other arm. This example involves polygons on different layers (Z = 150 microns for the lower level and Z = 250 microns for the upper level) and differential ports. It should involve most of the data types we would like the users to know.

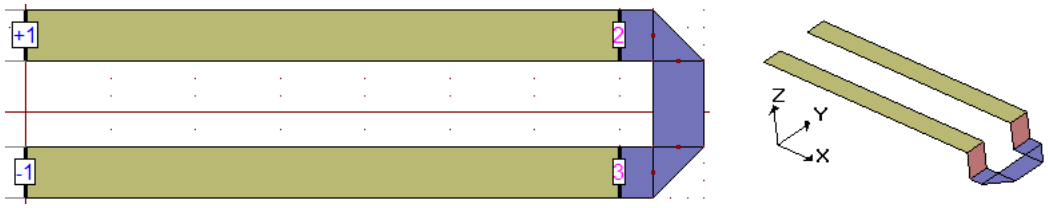


Figure B.1 The U-shaped structure with 3 differential ports on it.

1. IE3D XML 2.0 File Format:

Table B.2 and B.3 document the IE3D XML 2.0 file format. Except the Section 1 which is required for XML, every section has a beginning tag (denoted as “b” in the tables), and end tag (denoted as “e” in the tables). Each parameter inside a section normally has its own tag. The order of the tags inside a section is not critical. For those parameters are not useful for a particular section, you can omit it. IE3D will automatically create the default.

On IE3D, the default frequency unit is GHz and the default conductivity unit is in s/m. You can configure which length unit you want. The length unit is defined in the header sections. After it is defined, all length quantities are using the same unit, even though each section in the IE3D XML file formats allows you to define a length unit for it.

You are notified with the following fact: You are able to open the IE3D XML 2.0 file for IE3D 11 even on IE3D 10.2x. However, many of the objects will be dropped.

Table B.2 The IE3D XML 2.0 file for .\ie3d\samples\u_3ports.geo.

Content	Section
<?xml version="1.0" ?>	1
<ZlsDoc type="Ie3dSim xml file" version="2.0" copyright="Mentor graphics Corp." >	2b
<Ie3dSim unit="micron" version_str="11.0 " final_ports="0" license_version="0" >	3b
<!-- Ie3dSim base type begin -->	Comment
<Ie3dGeom format_version="20.1" comments="" eps_length="1e-006" netports="3" syn_layers="1" nbr="0" scheme_deembed="1" nhext_mmic="5" nhext_wave="5" Bmetal="0" Bgroundconnect="1" Bshowextent="1" Bsavecurrent="0" Bkeepdiscretize="1" Bshowvertex="0" dc_mode_index="0" >	4b
<!-- Ie3dGeom base type begin -->	Comment
<MeshingParametersBase ereff="5.450002474999382" fmax="20" ncells="20" warning_limit="4000" aec="0" aec_level="0" aec_ratio="0.1" multi_aec_ratio="0.4" meshing_optim="1" detect_overlapping="1" meshing_scheme="0" align_meshing="4" max_layer_distance="0.0005" cmax_regular="0.3210423766983152" refined_ratio="0.2" rectanglizations="3" merge_polygons="0" option_2d="2" option_3d="2" >	5b
</MeshingParametersBase>	5e
<!-- Ie3dGeom base type end -->	Comment
<Ie3dSubstrateArray id="subs" knmin="1000000" maxType="100" ztop_max="100000000" substrate_disp_margin="0.2" ermur_max="500" >	6b
<ObjList id="this">	
<Ie3dSubstrate comment="" enclosure_index="0" ground="0" type="0" ztop="0" transparency="0" xmax_disp="0" xmin_disp="0" ymax_disp="0" ymin_disp="0" zmax_disp="0" zmin_disp="0" >	6.1b
<!-- Ie3dSubstrate base type begin -->	
<Dielectrics im_epsr="0" im_mur="0" im_sigma="0" re_epsr="1" re_mur="1" re_sigma="49000000" >	6.1.1b
</Dielectrics>	6.1.1e
<!-- Ie3dSubstrate base type end -->	
<DielectricsFreqArray id="dfa" >	6.1.2b
<ObjList id="this">	
</ObjList>	
</DielectricsFreqArray>	6.1.2e
</Ie3dSubstrate>	6.1e
<Ie3dSubstrate comment="" enclosure_index="0" ground="1" type="0" ztop="250" transparency="0.5" xmax_disp="0" xmin_disp="0" ymax_disp="0" ymin_disp="0" zmax_disp="0" zmin_disp="0" >	6.2b
<!-- Ie3dSubstrate base type begin -->	
<Dielectrics im_epsr="-0.009900000000000001" im_mur="0" im_sigma="0" re_epsr="9.9" re_mur="1" re_sigma="0" >	6.2.1b
</Dielectrics>	6.2.1e
<!-- Ie3dSubstrate base type end -->	
<DielectricsFreqArray id="dfa" >	
<ObjList id="this">	
</ObjList>	
</DielectricsFreqArray>	
</Ie3dSubstrate>	6.2e
<Ie3dSubstrate comment="" enclosure_index="0" ground="1" type="0" ztop="1e+018" transparency="0" xmax_disp="0" xmin_disp="0" ymax_disp="0" ymin_disp="0" zmax_disp="0" zmin_disp="0" >	6.3b
<!-- Ie3dSubstrate base type begin -->	
<Dielectrics im_epsr="0" im_mur="0" im_sigma="0" re_epsr="1" re_mur="1" re_sigma="0" >	
</Dielectrics>	
<!-- Ie3dSubstrate base type end -->	
<DielectricsFreqArray id="dfa" >	6.3.1b
<ObjList id="this">	
</ObjList>	
</DielectricsFreqArray>	6.3.1e

</Ie3dSubstrate>	6.3e
</ObjList>	
</Ie3dSubstrateArray>	6e
<Ie3dMetallicsArray id="mets" maxType="100" >	7b
<ObjList id="this">	
<Ie3dMetallics comment="" thickness="5" type="0" factor="1" >	7.1b
<!-- Ie3dMetallics base type begin -->	
<Dielectrics im_epsr="0" im_mur="0" im_sigma="0" re_epsr="1" re_mur="1" re_sigma="49000000" >	
</Dielectrics>	
<!-- Ie3dMetallics base type end -->	
<DielectricsFreqArray id="dfa" >	
<ObjList id="this">	
</ObjList>	
</DielectricsFreqArray>	
</Ie3dMetallics>	7.1e
</ObjList>	
</Ie3dMetallicsArray>	7e
<LayoutEntryVector id="layouts" current_entry="1" >	8b
<ObjList id="this">	
<LayoutEntry gridsize="10" xlayt0="-100" xlaytw="1200" xratio="2.5" xZoom="180" ylayt0="-146.8000000000001" ylaytw="346.7999999999999" >	8.1b
</LayoutEntry>	8.1e
</ObjList>	
</LayoutEntryVector>	8e
<BoxArray id="boxes" >	9b
<ObjList id="this">	
<Box box_length="0" box_width="0" box_xi="0" box_yi="0" boxed="0" called="1" f0="1" ikx0="0" iky0="0" ix1="0" ix2="0" iy1="0" iy2="0" kx0="0" ky0="0" layer_end="2" layer_start="-1" max_m="30" max_mp="4" max_mpat="0" max_n="30" max_np="4" max_npat="0" min_m="10" min_n="10" ni="0" periodical="0" u_extracted="1" zmax="1e+050" zmin="-1e+050" >	9.1b
<ObjList id="this">	
</ObjList>	
<ObjList id="ximages">	
</ObjList>	
<ObjList id="yimages">	
</ObjList>	
</Box>	9.1e
</ObjList>	
</BoxArray>	9e
<Ie3dLayerArray id="layers" layerAccessedLast="3" >	10b
<ObjList id="this">	
<Ie3dLayer brush_index="0" comment="" selected="0" type="0" z="0" >	10.1b
</Ie3dLayer>	10.1e
<Ie3dLayer brush_index="0" comment="" selected="0" type="1" z="0" >	10.2b
</Ie3dLayer>	10.2e
<Ie3dLayer brush_index="0" comment="" selected="0" type="0" z="150" >	10.3b
</Ie3dLayer>	10.3e
<Ie3dLayer brush_index="0" comment="" selected="1" type="0" z="250" >	10.4b
</Ie3dLayer>	10.4e
</ObjList>	
</Ie3dLayerArray>	10e
<PlaneWaveArray id="pws" >	11b
<ObjList id="this">	

</ObjList>	
</PlaneWaveArray>	11e
<Ie3dPortArray id="ports" netports="3" wave="0" shift_exist="0" auto_adjust="1" min_length_ratio="3" min_extension="5" max_extension="80" good_reference="10" >	12b
<ObjList id="this">	
<Ie3dPort scheme="3" emtype="-1" shift="0" property="1" dir="3.141592653589793" extension="5" csize="321.0423766983152" x1="0" y1="0.07500000000000001" z1="0.25" x2="0" y2="0.15" z2="0.25" index="1241008" re_zc="2.633369892333844e-321" im_zc="2.633812687762936e-308" re_gamma="9.603995534815466e-315" im_gamma="1.088588245325833e-311" area="2.800082785199893e-301" >	12.1b
<ObjList id="this">	
</ObjList>	
<Ie3dPortCellRows id="ipcrs" >	12.1.1
<ObjList id="this">	
</ObjList>	
</Ie3dPortCellRows>	
<Ie3dPortSideRows id="ipsrs" >	12.1.2
<ObjList id="this">	
</ObjList>	
</Ie3dPortSideRows>	
</Ie3dPort>	12.1e
<Ie3dPort scheme="3" emtype="-1" shift="0" property="-1" dir="3.141592653589793" extension="5" csize="321.0423766983152" x1="0" y1="-0.05" z1="0.25" x2="0" y2="-0.125" z2="0.25" index="1241008" re_zc="2.633369892333844e-321" im_zc="2.633812687762936e-308" re_gamma="9.603995534815466e-315" im_gamma="1.088588245325833e-311" area="7.467061350839557e-299" >	12.2b
<ObjList id="this">	
</ObjList>	
<Ie3dPortCellRows id="ipcrs" >	
<ObjList id="this">	
</ObjList>	
</Ie3dPortCellRows>	
<Ie3dPortSideRows id="ipsrs" >	
<ObjList id="this">	
</ObjList>	
</Ie3dPortSideRows>	
</Ie3dPort>	12.2e
<Ie3dPort scheme="5" emtype="1" shift="150" property="2" dir="1.710329278597495e-311" extension="0" csize="250" x1="0.875" y1="0.07500000000000001" z1="0.15" x2="0.875" y2="0.15" z2="0.25" index="1241008" re_zc="2.633369892333844e-321" im_zc="2.633812687762936e-308" re_gamma="9.603995534815466e-315" im_gamma="1.088588245325833e-311" area="1.073657590703224e-300" >	12.3b
<ObjList id="this">	
</ObjList>	
<Ie3dPortCellRows id="ipcrs" >	
<ObjList id="this">	
</ObjList>	
</Ie3dPortCellRows>	
<Ie3dPortSideRows id="ipsrs" >	
<ObjList id="this">	
</ObjList>	
</Ie3dPortSideRows>	
</Ie3dPort>	12.3e
<Ie3dPort scheme="5" emtype="2012577136" shift="150" property="3" dir="1.797331106027071e-311"	12.4b

extension="0" csize="250" x1="0.875" y1="-0.125" z1="0.15" x2="0.875" y2="-0.05" z2="0.25" index="1241008" re_zc="2.633369892333844e-321" im_zc="2.633812687762936e-308" re_gamma="9.603995534815466e-315" im_gamma="1.088588245325833e-311" area="5.079583445535814e-299" >	
<ObjList id="this">	
</ObjList>	
<Ie3dPortCellRows id="ipcrs" >	
<ObjList id="this">	
</ObjList>	
</Ie3dPortCellRows>	
<Ie3dPortSideRows id="ipsrs" >	
<ObjList id="this">	
</ObjList>	
</Ie3dPortSideRows>	
</Ie3dPort>	12.4e
</ObjList>	
</Ie3dPortArray>	12e
<Ie3dPolygonArray id="polygons" err_msg="" eps="1.499e-007" eps_relative="1e-006" cmax="1" >	13b
<ObjList id="this">	
<Ie3dPolygon domain="0" emtype="0" ncells="20" csize="4.374152446414581e-005" start_index="0" end_index="0" attributes="00000000" enclosure_index="0" geom_type="0" state1="0" layer="3" mets="1" comment="" >	13.1b
<!-- Ie3dPolygon base type begin -->	
<NodeAndSideArray >	13.1.1b
0,-50,250,2	
0,-125,250,0	
875,-125,250,4	
875,-50,250,0	13.1.1e
</NodeAndSideArray>	
<!-- Ie3dPolygon base type end -->	
</Ie3dPolygon>	13.1e
<Ie3dPolygon domain="0" emtype="0" ncells="20" csize="4.374152446414581e-005" start_index="0" end_index="0" attributes="00000000" enclosure_index="0" geom_type="0" state1="0" layer="3" mets="1" comment="" >	13.2b
<!-- Ie3dPolygon base type begin -->	
<NodeAndSideArray >	13.2.1b
0,75.00000000000001,250,0	
875,75.00000000000001,250,3	
875,150,250,0	
0,150,250,1	
</NodeAndSideArray>	13.2.1e
<!-- Ie3dPolygon base type end -->	
</Ie3dPolygon>	13.2e
<Ie3dPolygon domain="0" emtype="0" ncells="20" csize="4.374152446414581e-005" start_index="0" end_index="0" attributes="00000000" enclosure_index="0" geom_type="0" state1="0" layer="2" mets="1" comment="" >	13.3b
<!-- Ie3dPolygon base type begin -->	
<NodeAndSideArray >	13.3.1b
925,-50,150,0	
875,-50,150,4	
875,-125,150,0	
925,-125,150,-1	
</NodeAndSideArray>	13.3.1e

<!-- Ie3dPolygon base type end -->	
</Ie3dPolygon>	13.1e
<Ie3dPolygon domain="0" emtype="0" ncells="20" csize="4.374152446414581e-005" start_index="0" end_index="0" attributes="00000000" enclosure_index="0" geom_type="0" state1="0" layer="2" mets="1" comment="" >	13.4b
<!-- Ie3dPolygon base type begin -->	
<NodeAndSideArray >	13.4.1b
925,-125,150,0	
1000,-50,150,-2	
925,-50,150,-1	
</NodeAndSideArray>	13.4.1e
<!-- Ie3dPolygon base type end -->	
</Ie3dPolygon>	13.4e
<Ie3dPolygon domain="0" emtype="0" ncells="20" csize="4.374152446414581e-005" start_index="0" end_index="0" attributes="00000000" enclosure_index="0" geom_type="0" state1="0" layer="2" mets="1" comment="" >	13.5b
<!-- Ie3dPolygon base type begin -->	
<NodeAndSideArray >	13.5.1b
925,75.000000000000001,150,-3	
1000,75.000000000000001,150,0	
925,150,150,-4	
</NodeAndSideArray>	13.5.1e
<!-- Ie3dPolygon base type end -->	
</Ie3dPolygon>	13.5e
<Ie3dPolygon domain="0" emtype="0" ncells="20" csize="4.374152446414581e-005" start_index="0" end_index="0" attributes="00000000" enclosure_index="0" geom_type="0" state1="0" layer="2" mets="1" comment="" >	13.6b
<!-- Ie3dPolygon base type begin -->	
<NodeAndSideArray >	13.6.1b
925,150,150,0	
875,150,150,3	
875,75.000000000000001,150,0	
925,75.000000000000001,150,-4	
</NodeAndSideArray>	13.6.1e
<!-- Ie3dPolygon base type end -->	
</Ie3dPolygon>	13.6e
<Ie3dPolygon domain="0" emtype="0" ncells="20" csize="4.374152446414581e-005" start_index="0" end_index="0" attributes="00000000" enclosure_index="0" geom_type="0" state1="0" layer="2" mets="1" comment="" >	13.7b
<!-- Ie3dPolygon base type begin -->	
<NodeAndSideArray >	13.7.1b
1000,75.000000000000001,150,0	
1000,-50,150,-2	
925,-50,150,0	
925,75.000000000000001,150,-3	
</NodeAndSideArray>	13.7.1e
<!-- Ie3dPolygon base type end -->	
</Ie3dPolygon>	13.7e
<Ie3dPolygon domain="0" emtype="0" ncells="20" csize="4.374152446414581e-005" start_index="0" end_index="0" attributes="00000000" enclosure_index="0" geom_type="5" state1="0" layer="0" mets="1" comment="" >	13.8b
<!-- Ie3dPolygon base type begin -->	
<NodeAndSideArray >	13.8.1b

875,75.00000000000001,250,3	
875,150,250,0	
875,150,150,3	
875,75.00000000000001,150,0	
</NodeAndSideArray>	13.8.1e
<!-- Ie3dPolygon base type end -->	
</Ie3dPolygon>	13.8e
<Ie3dPolygon domain="0" emtype="0" ncells="20" csize="4.374152446414581e-005" start_index="0" end_index="0" attributes="00000000" enclosure_index="0" geom_type="5" state1="0" layer="0" mets="1" comment="" >	13.9b
<!-- Ie3dPolygon base type begin -->	
<NodeAndSideArray >	13.9.1b
875,-125,250,4	
875,-50,250,0	
875,-50,150,4	
875,-125,150,0	
</NodeAndSideArray>	13.9.1e
<!-- Ie3dPolygon base type end -->	
</Ie3dPolygon>	13.9e
</ObjList>	
</Ie3dPolygonArray>	13e
<Ie3dOptimVariables id="variables" >	14b
<ObjList id="this">	
</ObjList>	
</Ie3dOptimVariables>	14e
<Ie3dOptimCalls id="optim_calls" >	15b
<ObjList id="this">	
</ObjList>	
</Ie3dOptimCalls>	15e
<Ie3dDielectricsCallArray id="dielectrics_calls" >	16b
<ObjList id="this">	16.1b
</ObjList>	16.1e
<Ie3dPolygonArray id="polygons" err_msg="" eps="1e-007" eps_relative="1e-006" cmax="1" >	16.2b
<ObjList id="this">	
</ObjList>	
</Ie3dPolygonArray>	16.2e
<Ie3dDielectricsArray id="dielectrics" >	16.3b
<ObjList id="this">	
</ObjList>	
</Ie3dDielectricsArray>	16.3e
</Ie3dDielectricsCallArray>	16e
<VertexIndex id="movingRef_pv" pindex="-1" vindex="-1" >	17b
</VertexIndex>	17e
</Ie3dGeom>	4e
<!-- Ie3dSim base type end -->	
<Ie3dMeshed id="meshed" bUnknowns_OnEdgesOfSurfs="0" bUnknowns_OnSurfsOfVols="0" VieLengthUnit="0" accuracy="0" accuracy3d="0" unknowns="0" >	18b
<NodeArray id="nodes" >	
</NodeArray>	
<Ie3dCellNSArray id="cells" >	
<ObjList id="this">	
</ObjList>	
</Ie3dCellNSArray>	

<Ie3dSideArray id="sides" >	
<ObjList id="this">	
</ObjList>	
</Ie3dSideArray>	
<Ie3dSimMetalLayers id="metalayers" nlayers="0" >	
<ObjList id="this">	
</ObjList>	
</Ie3dSimMetalLayers>	
<Ie3dPortArray id="ports" netports="0" wave="0" shift_exist="0" auto_adjust="0" min_length_ratio="3" min_extension="5" max_extension="80" good_reference="10" >	
<ObjList id="this">	
</ObjList>	
</Ie3dPortArray>	
<PlaneWaveArray id="pws" >	
<ObjList id="this">	
</ObjList>	
</PlaneWaveArray>	
<Division3DArray id="divs" >	
<ObjList id="this">	
</ObjList>	
</Division3DArray>	
<NodeArray id="nodes_fd" >	
</NodeArray>	
<Ie3dCellNSArray id="cells_fd" >	
<ObjList id="this">	
</ObjList>	
</Ie3dCellNSArray>	
<Ie3dSideArray id="sides_fd" >	
<ObjList id="this">	
</ObjList>	
</Ie3dSideArray>	
<Ie3dVolumeCellArray id="vca_fd" >	
<ObjList id="this">	
</ObjList>	
</Ie3dVolumeCellArray>	
<Ie3dVolumeSurfaceArray id="vsa_fd" >	
<ObjList id="this">	
</ObjList>	
</Ie3dVolumeSurfaceArray>	
<MeshCurrentArrayFreqSweep id="mcafs" >	
<ObjList id="this">	
</ObjList>	
</MeshCurrentArrayFreqSweep>	
</Ie3dMeshed>	18e
</Ie3dSim>	3e
</ZlsDoc>	2e

Table B.3 Explanation for the IE3D XML 2.0 file for .\ie3d\samples\u_3ports.geo.

Section	Explanation
1	The symbol <?xml version=”1.0” ?> indicates it is an XML file.
2	This the beginning section for Mentor Graphics IE3D Document XML file. The version=”2.0” indicates it is the IE3D XML 2.0 file format.
3	It is the beginning of Ie3dSim object. Ie3dSim object includes the geometry and the

	simulation result; unit="micron" indicates the length is in "micron". You can choose other length units. Please save a file in other unit to see what symbols are used for other units. The version_str="11.0" indicates it is IE3D 11. Other items are not critical.
Comment	Any thing like "<!....> " is a comment.
4	It is the Ie3d geometry portion or the data for the layout editor. The format_version="20.1" is in coincident with the version="2.0" in Section 2b. The eps_length="1.0e-6" defines the minimum length used in the geometry. It is used for IE3D to judge when two vertices are close enough to be considered as one vertex. The netports="3" indicates there are total 3 net ports in the structure. This parameter is used internally and it can be any value in the file. For this structure, we actually define 4 global ports: +1, -1, 2 and 3. The +1 and -1 ports form the port 1. Therefore, we have 3 net ports only. Other parameters are not used for controlling how we display the geometry in the layout editor or used internally. They are not critical.
5	This section defines the meshing parameters. There is the same section in the corresponding Simulation Input File (.sim). When we setup a simulation, MGRID will try to get necessary information from this section as default for the .sim file. If the simulation is invoked by MGRID, the data in this section will not be used. The corresponding section in .sim file is used. If we perform a simulation on MODUA involving one or multiple .geo file in the .dsg file, each .geo file will use its own data in the MeshingParametersBase section. It will not use the corresponding section in the .sim file. You can find most items in this section from the Meshing Parameters dialog. The ereff and cmax_regular are not critical. They are automatically calculated from other parameters.
6	This section defines the substrates. Knmin="1000000" means the Conductor Assumption Limit (CAL) = 1.0e+6. The ztop_max="100000000" and ermur_max="500" indicate that we will use those layers with Ztop below 1.0e+8 and Er*Mur<500 to estimate the waveguide wavelength for meshing. This estimate can be very rough when the substrates are complicated. The maxType="100" is used for old version to identify frequency dependent substrates. They are no longer meaningful in IE3D XML files. The substrate_disp_margin="0.2" means that we optionally display the substrate as finite size with a margin of 0.2 wavelength at Fmax.
6.1	This section defines the No.0 substrate. The top surface z-coordinate is ztop="0" and the layer belongs to enclosure_index="0". For open structure, we should always use enclosure_index=0 for all the layers. Other parameters are not critical.
6.1.1	This section defined substrate properties of the No.0 substrate. The permittivity Er= (1, 0), permeability Mur= (1, 0) and conductivity Sigma= (4.9e+7, 0). It is basically the ground plane.
6.1.2	This section is for frequency dependent substrate parameter for the No.0 substrate. It is empty because it is not frequency dependent.
6.2	This section defines the No.1 substrate. The top surface z-coordinate is ztop="250" microns.
6.2.1	This section defines the No.1 substrate with Er=(9.9, -0.0099) (tanδ=0.001), Mur=(1, 0), Sigma=(0,0). Again, the definition of tanδ and Sigma are not in the standard definitions in order to resolve the consistency problem between them. Explanation can be found in Section 2 of Chapter 3).
6.3	This section define the No.2 substrate with Ztop="1e+18", Er=(1, 0), Mur=(1, 0), Sigma=(0, 0). It is the open air above the No.1 substrate.
7	This section defines the metallic strip types. The maxTypes="100" is used to identify the frequency dependent substrates in old versions. It is not useful for IE3D XML files anymore.
7.1	It defines the No.1 metallic strip type with Er=(1, 0), Mur=(1, 0) and Sigma=(4.9e+7, 0) and Thickness="5" microns. The type="0" means it is normal conductor type. The factor="1" is for the cross-sectional factor. It should be 1 for normal thin strip. It should

	be a value close to “0.5” if it is for the thick strip model.
8	It is for the types of layouts and grids. You can define multiple different grid systems for your layout. IE3D’s grids are not used for meshing. They are used for drawing exclusively.
8.1	This section defines, when the zoom is 100% (xZoom=”100”), the gridsize=10 pixels with each pixel representing a physical size of 2.5 microns (xratio=”2.5”) in your physical layout. The size of each grid is gridsize*xratio=10*2.5=25 microns. Other parameters are not useful.
9	This section defines the enclosures for the structure. On IE3D, you can define multiple infinite ground planes (large Sigma values) separated by some normal substrates. Each section between two infinite substrates can be defined with an enclosure by the Enclosure Index.
9.1	This section defines an enclosure without any side walls. It is basically an open structure. All the substrates are referencing this Enclosure Index and all the substrates do not have boundaries in the X and Y-directions. It is basically an infinite substrate and ground plane configuration.
10	This section defines the polygon layers or metallic layers in the structure. They are automatically sorted and managed by MGRID and IE3D. They are not critical in the file. When the file is read, MGRID or IE3D will automatically find them.
11	This section defines whether there is plane wave excitation. At this time, IE3D can only accept one plane wave excitation only. You can define no or one excitation for it. For this structure, there is no plane wave excitation.
12	This section defines the ports on the structure. The netports=”3” indicates it has total 3 ports for the structure (The +1 and -1 considered as one net port). However, there are total 4 global ports (+1, -1, 2 and 3). The parameters: auto_adjust, min_length_ratio, min_extension, max_extension and good_reference are for the automatic adjustment of the port extension. You can use the default values here. Other parameters are created automatically for internal use.
12.1	This section defines the port +1 (property=”1”). It is using Advanced Extension scheme (scheme=”3”). The default port extension is 5 (extension=”5”). The reference plane shift is 0 (shift=”0”). Please note that Shift is used to define the Negative Level for a Vertical Localized port. The direction (dir) is automatically calculated for extension ports and Localized for MMIC port. However, it is defined by the user if it is for Horizontal Localized port. The “Csize” defines the cell size on the port extension for extension ports. It is automatically set by MGRID or IE3D. However, it is used to define the positive level for a Vertical Localized port. Other parameters define the location of the port which is calculated by MGRID or IE3D. You may be curious where how the port location is calculated by MGRID if no other data is used to define it. The reality is that we have to define the port location by the polygons. Basically, this section only defines the basic properties of the port “+1”. The location information is stored in the polygons. A port on IE3D involves a port object plus some other data from polygons to define it. It makes it tricky to define a port. However, there is no simpler way for it when flexibility, accuracy and robustness are all taken into consideration.
12.1.1	This section and Section 12.1.2 are used internally. You can omit them.
12.2	This section defines the port -1 (property=”-1”). It is also an Advanced Extension port (scheme=”3”). Other parameters are similar to the port +1. Please note there can be multiple “+” terminals and/or multiple “–“ terminals for a net port.
12.3	This section defines the port 2 (property=”2”). It is a Vertical Localized port (scheme=”5”) with Positive Level at Z = 250 microns (CSize=”250”) and Negative Level at Z = 150 microns (Shift=”150”). Other parameters are not useful.
12.4	This section defines the port 3 (property = “3”). It is a Vertical Localized port (scheme=”5”) with Positive Level at Z = 250 microns (CSize=”250”) and Negative Level at Z = 150 microns (Shift=”150”).

	Please note that other extension ports and the Localized for MMIC port are very similar to the Advanced Extension port except the Scheme value is different. The Horizontal Localized port is not documented here. Its scheme value is: Scheme="7". Other useful parameters for a Horizontal Localized port are: Property defining its index and Dir defining the polarity. Other parameters are not useful.
13	This section defines all the polygons. The eps="1.499e-7" is used to judge whether two vertices are close enough to be considered as one. It is supposed to be consistent with the eps_length value in Section 4b. It is possible they may not be the same even both of them are small values. The eps_relative="1.0e-6" is default and the cmax="1" is automatically set and calculated internally.
13.1	It describes the basic properties of the No.1 polygon. It is defined with No.1 metallic type (mets="1"). Other parameters are set internally and they can be omitted. Please note that layer="3" means that this polygon is a 2D polygon with constant Z and it is No. layer No.3. As it is mentioned, layers are managed by MGRID/IE3D and you can omit this information even though it does tell you the properties of the polygon if the file is created by MGRID.
13.1.1	This section defines the vertices and sides of the No.1 polygon. The 1st line defines the No.0 vertex at (X, Y, Z, S) = (0, -50, 250, 2). S = 2 means that the edge from No.0 vertex and No.1 vertex is an edge with global port 2 defined on. A global port can be defined on a group of edges. Also, for our case, global port 2 is basically the Port -1 instead of the port 2 defined above. The No.1 vertex is at (X, Y, Z, S) = (0, -125, 250, 0). The S = 0 indicates that the edge formed by No.1 vertex and No.2 vertex of this polygon is not common to other polygon, or this edge is an open edge. Please note that the S value is automatically set by MGRID if it is not positive. If it is of positive value, it is defined by the user to indicate the edge is an edge for a port. If it is 0 or negative value, it is automatically defined by MGRID. If the .geo file is created externally without saved by MGRID, the non-positive value for S may not have the exact meaning. In some sense, S = 0 may not mean it is an open edge because it is not checked by MGRID. The No.2 vertex is at (X, Y, Z, S)=(875, -125, 250, 4). The S = 4 indicates that the edge formed by No.2 vertex and No.3 vertex is an edge with global port 4 defined on. The global port 4 is basically the port 3 defined above (port +1 is the global port 1, port -1 is the global port 2 and port 2 is the global port 3). The No.3 vertex is at (X, Y, Z, S)=(875, -50, 250, 0). The S = 0 indicates the edge by No.3 vertex and No.0 vertex is an open edge. This polygon has total 4 vertices.
13.2	This section defines the No.2 polygon. It is of metallic type 1 (mets="1").
13.2.1	This section defines the vertices and sides of the No.2 polygon. No.0 vertex is at (X, Y, Z, S)=(0, 75, 250, 0). No.1 vertex is at (X, Y, Z, S)=(875, 75, 250, 3) and the edge belongs to global port 3 (or port 2). The No.2 vertex is at (X, Y, Z, S)=(875, 150, 250, 0). The No.3 vertex is at (X, Y, Z, S)=(0, 150, 250, 1) and the edge belongs to global port 1 (or port +1). The No.2 polygon has 4 vertices.
13.3	This section defines the No.3 polygon. It is of metallic type 1 (mets="1").
13.3.1	This section defines the vertices and sides of the No.3 polygon. No.0 vertex is at (X, Y, Z, S)=(925, -50, 150, 0). No.1 vertex is at (X, Y, Z, S)=(875, -50, 150, 4) and the edge belongs to global port 4 (or port 3). The No.2 vertex is at (X, Y, Z, S)=(875, -125, 150, 0). The No.3 vertex is at (X, Y, Z, S)=(925, -125, 150, -1). The S=-1 indicates that the edge formed by the No.3 vertex and the No.0 vertex is connected to other polygons on the edges with the same S value (S=-1). Again, this value is automatically set by MGRID. You can not just specify the same negative S value on two different edges of different polygons and expect IE3D to accept it. You need to make sure the edges do match and MGRID/IE3D will automatically confirm it. It may assign a different negative S value for it. The No.3 polygon has 4 vertices.
13.4	This section defines the No.4 polygon. It is of metallic type 1 (mets="1").

13.4.1	This section defines the vertices and sides of the No.4 polygon. No.0 vertex is at (X, Y, Z, S)=(925, -125, 150, 0). No.1 vertex is at (X, Y, Z, S)=(100, -50, 150, -2). The edge between No.1 and No.2 vertices is connected to other edges of other polygons with connection index = -2 (S=-2). The No.2 vertex is at (X, Y, Z, S)=(925, -50, 150, -1) and the edge between No.2 vertex and No.0 vertex is connected to other polygons with connection index = -1 (S=-1). This polygon has 3 vertices.
13.5	This section defines the No.5 polygon with metallic type=1.
13.5.1	This section defines the 3 vertices of the No.5 polygon.
13.6 – 13.7	The sections define the No.6 and No.7 polygons of the structure.
13.8	This section defines a special polygon (Polygon 8) for the global port 3.
13.8.1	For the No.8 polygon, the No.0 vertex is at (X, Y, Z, S)=(875, 75, 250, 3); the No.1 vertex is at (X, Y, Z, S)=(875, 150, 250, 0); The No.2 vertex is at (X, Y, Z, S)=(875, 150, 3) and the No.3 vertex is at (X, Y, Z, S)=(875, 75, 150, 0). What are special for it? This polygon has 4 vertices. The polygon is a rectangle. It is absolutely vertical with 2 vertices having the same (X, Y)-coordinates and the other two vertices having the same (X, Y)-coordinates. The edge on Z = 150 (both vertices are at Z = 150) belongs to global port 3 (port 2). The edge on Z = 250 belongs to global port 3 (port 2). The above properties are used to identify this polygon is polygon belonging to the global port 3 (between the Positive and Negative Levels). For this particular example, the edges with non-positive S-values indicate the polygon is not connected to other polygons on the vertical edges (spanning from Z = 150 to 250). If there is a negative S-value edge in this polygon, it means that multiple such polygons connected together to form the global port. Those connected polygons must be the polygons of the same global port. It can not be other polygons. Otherwise, MGRID/IE3D will detect it and remove this port automatically.
13.9	This section defines the special polygon for global port 4.
14	This section defines the optimization variables (Ie3dOptimVariables). For this structure, there is no optimization variable.
15	This section defines the calls to optimization variables (or Ie3dOptimCalls). An Ie3dOptimCall associates a group of vertices in the polygons to an Ie3dOptimVariable. It defines when a variable is changing, how the associated vertices are changing. No variables and calls are defined for this structure.
16	This section defines the finite dielectric blocks defined in the structure. For this example, no finite dielectric blocks are defined.
16.1	This section defines the possible Ie3dDielectricsCallArray for a number of finite dielectric blocks. Each block will reference the finite dielectric polygons and dielectric types defined below. No call is available for this structure.
16.2	This section defines the polygons may be used to define the finite dielectric blocks. No finite dielectric polygon is defined in this structure.
16.3	This section defines the finite dielectric types. No such type is defined in this geometry.
17	This section is used by geometry modeling. It can be omitted.
18	Section 8 is for the simulation results. It can be omitted.

2. IE3D 9.0 File Format:

For compatibility, we document the IE3D 9.0 format here. This is the last ASCII file format for the IE3D. This file format and the document are not updated in IE3D 11. We keep it here for those users want to save the file in IE3D 9.0 format. When you use IE3D 9.0 format as default, MGRID may automatically save the file in IE3D 9.3 format if it detects there are new thick trace models in the data.

The file .\ie3d\samples\u_3ports_9.geo is in IE3D 9.0 (9.3) format is documented in the Table B.4 and Table B.5. It describes the same structure as t he one in u_3ports.geo shown in Figure B.1. The IE3D 9.0 file is normally much smaller than an IE3D XML 2.0 file. However, everything is in ASCII file format. The order of the data is extremely critical. Slight mis-alignment will cause a problem. On the IE3D XML

2.0 file format, every parameter has its own tag. It can be anywhere in the same section. Different sections can be located in different locations inside the bigger section. Also, we may add more parameters into one section in future version. When we add new parameters to it, the file format will not change. Older versions of MGRID should still be able to read later version IE3D XML files while those added parameters will be omitted. In the long run, we should adopt the IE3D XML file format even though its file size is much bigger, and accessing the file may be much slower. XML file format is an industrial standard anyway.

Table B.4 The IE3D 9.0 geometry file in .\ie3d\samples\u_3ports_9.geo.

Content	Section
9	1
1.499e-7 0 3.2104237670e+1 1 1	2
	3
2.0e+1 3 2 1 1.0e+6 5.0e+2	4
0.0 1.0 0.0 1.0 0.0 4.9e+7 0.0 0 0	5
2.5e+2 9.9 -9.9e-3 1.0 0.0 0.0 0.0 0 0	6
1.0e+018 1.0 0.0 1.0 0.0 0.0 0.0 0 0	7
5.0 1.0 0.0 1.0 0.0 4.9e+7 0.0 0	8
1 1	9
-1.0e+2 -1.468e+2 1.2e+3 3.468e+2 2.5 10	10
1	11
0 0 0 0 30 30 0 0 0.0 0 0.0 0	12
0.0 0.0 0.0 0.0	13
9 4 4 0	14
0.0 1	15
1.5e+2 0	16
2.5e+2 0	17
3 5 3.2105045329e+2 3.1415926536 0.0 1	18
3 5 3.2105045329e+2 3.1415926536 0.0 -1	19
5 0 2.5e+2 1.7103292786e-311 1.5e+2 2	20
5 0 2.5e+2 1.7973311060e-311 1.5e+2 3	21
4 0 20 3 1	22
0.0 -5.0e+1 2.5e+2 2	23
0.0 -1.25e+2 2.5e+2 0	
8.75e+2 -1.25e+2 2.5e+2 4	
8.75e+2 -5.0e+1 2.5e+2 0	
4 0 20 3 1	24
0.0 7.5e+1 2.5e+2 0	25
8.75e+2 7.5e+1 2.5e+2 3	
8.75e+2 1.5e+2 2.5e+2 0	
0.0 1.5e+2 2.5e+2 1	
4 0 20 2 1	26
9.25e+2 -5.0e+1 1.5e+2 0	27
8.75e+2 -5.0e+1 1.5e+2 4	
8.75e+2 -1.25e+2 1.5e+2 0	
9.25e+2 -1.25e+2 1.5e+2 -1	
3 0 20 2 1	28
9.25e+2 -1.25e+2 1.5e+2 0	29
1.0e+3 -5.0e+1 1.5e+2 -2	
9.25e+2 -5.0e+1 1.5e+2 -1	
3 0 20 2 1	30
9.25e+2 7.5e+1 1.5e+2 -3	31
1.0e+3 7.5e+1 1.5e+2 0	
9.25e+2 1.5e+2 1.5e+2 -4	

4 0 20 2 1	32
9.25e+2 1.5e+2 1.5e+2 0	33
8.75e+2 1.5e+2 1.5e+2 3	
8.75e+2 7.5e+1 1.5e+2 0	
9.25e+2 7.5e+1 1.5e+2 -4	
4 0 20 2 1	34
1.0e+3 7.5e+1 1.5e+2 0	35
1.0e+3 -5.0e+1 1.5e+2 -2	
9.25e+2 -5.0e+1 1.5e+2 0	
9.25e+2 7.5e+1 1.5e+2 -3	
4 0 20 0 1	36
8.75e+2 7.5e+1 2.5e+2 3	37
8.75e+2 1.5e+2 2.5e+2 0	
8.75e+2 1.5e+2 1.5e+2 3	
8.75e+2 7.5e+1 1.5e+2 0	
4 0 20 0 1	38
8.75e+2 -1.25e+2 2.5e+2 4	39
8.75e+2 -5.0e+1 2.5e+2 0	
8.75e+2 -5.0e+1 1.5e+2 4	
8.75e+2 -1.25e+2 1.5e+2 0	
	40
0 0	41

Table B.5 The descriptions on the sections in .\ie3d\samples\u_3ports_9.geo file.

Section No	Description
1	The version number is "9" meaning version 9.0 format
2	The 1st number "1.499e-7" is the smallest distance value used in the geometry. Any two vertices with distance smaller than this value should be merged into 1. The 2nd number "0" means that AEC is disabled. The 3rd number "3.2104237670e+1" is the width of the edge cell in the length unit defined later. The 4th number "1" means that the Meshing Optimization is enabled. The 5th number "1" means that there is one comment line.
3	The empty line here is the No.1 comment line of the geometry file. The last number in the last line defines there is one comment line.
4	The 1st number "2.0e+1" is the Highest Frequency for the meshing. The 2nd number "3" means that the geometry is using "micron" for the Length Unit. (1 for "mm", 2 for "mils" etc). The 3rd number "2" is the number of substrates (not including the No.0 substrate). The 4th number "1" is the number of the metallic strip types. The 5th number "1.0e+6" is the CAL discussed in Appendix A. The 6th number is optional. It is the Max DK value in the Substrate dialog of the Basic Parameters. It controls the limit of the Er used when we try to estimate the waveguide wavelength of the TLN for a multiple layer structures. When the Er is larger than the Max DK value, the substrate will not be used in the estimation of the waveguide wavelength for meshing. Again, the user can leave omit the 6th number and the IE3D will assume a default value for it.
5	The parameters are for the No.0 substrate. The 1st number "0.0" is the Ztop. The 2nd and 3rd numbers "1.0 0.0" are Re(Epsr) and Im(Epsr). The 4th and 5th numbers "1.0 0.0" are the Re(Mur) and Im(Mur). The 6th and 7th numbers "4.9e+7 0.0" are the Re(sigma(and Im(sigma) The 8th number is for the substrate type. Normally, it should be 0 for regular substrate. If it is 2, it becomes HTS ground. The 9th number "0" means that the substrate layer is bounded by the enclosure No.0.
6	The parameters are for the No.1 substrate. Ztop at 250 microns. Re(Epsr) = 9.9 and Im(Epsr) = -9.9e-3. Re(Mur) = 1 and Im(Mur) = 0. Re(Sigma) = 0 and Im(Sigma) = 0.
7	The parameters for the No.3 substrate. It is the half free space.

8	The No.1 metallic type. The 1st number "5" is the strip thickness in microns. The 2nd and 3rd numbers are Re(Epsr) and Im(Epsr). The 4th and 5th numbers are Re(Mur) and Im(Mur). The 6th and 7th numbers are the Re(Sigma) and Im(Sigma). For the 9.0 format, the 8th number is the metallic type. Normally it should be 0. However, for the 9.3 format when the version number in the 1st line is 9.3, there should be 9 numbers on each line defining a metallic type. The 8th number is a floating point number for the trace. Its value should be from (0 to 1). The 9th number is the metallic type and it should be 0 normally.
9	The 1st number "1" is the number of Layout and Grids set. The 2nd number "1" means that the No.1 set is used.
10	It is the No.1 set of Layout and Grids parameters. The 1st four numbers are not used anymore. The 5th number means that each pixel on the screen is 0.1 mm when the zoom is 100%. The 6th number is always 10 and 10 pixels is equivalent to 1 grid size. The numbers in this line are not important.
11	The number of enclosures used in the structure. It is 1 for our case.
12	The parameters for the No.0 enclosure. Please leave the numbers as they are.
13	The parameters for the locations of the enclosure walls. Just put "0 0 0 0" for no enclosure walls.
14	The 1st number "9" is the number of polygons. The 2nd number "4" is the number of layers in the geometry. The 3rd number "4" is the number of global ports. For this structure, we have 4 global ports (+1, -1, 2 and 3). The last number "0" means there is no plane wave excitation.
15	The parameters for the No.1 layer. The 1st number "0" means that the Z-coordinate is 0. The 2nd number "1" means that the polygons on this layer are magnetic current polygons. If it is "0", the polygons on the layers are electric current polygons. This value is automatically set by MGRID and it is not critical in the .geo file.
16	The parameters for the No.2 layer. They are automatically set by MGRID and they are not critical.
17	The parameters for the No.3 layer.
18	The parameters of the No.1 global port. The 1st number 3 means that it is using the Advanced Extension port scheme. The 2nd number "5" means there are 5 cells on the de-embedding arm. The 3rd number "3.21…" is the length of each cell on the arm. The 4th number "3.14159…" is the direction of the port in rad. The 5th number "0.0" is the distance of shift reference plane. The 6th number "1" means it is the net port +1.
19	The parameters of the No.2 global port. The 1st number 3 means that it is using the Advanced Extension port scheme. The 2nd number "5" means there are 5 cells on the de-embedding arm. The 3rd number "3.2105…" is the length of each cell on the arm. The 4th number "3.14159…" is the direction of the port in rad. The 5th number "0.0" is the distance of shift reference plane. The 6th number "-1" means it is the net port "-1".
20	The parameters of the No.3 global port. The 1st number 5 means that it is a Vertical Localized port. The 2nd number "0" is a dummy. The 3rd number "2.5e+2" is the Positive Level of the port. The 4th number "1.71…" is a dummy. The 5th number "1.5e+2" is the Negative Level. The 6th number "2" means it is the net port 2.
21	The parameters of the No.4 global port. The 1st number 5 means that it is a Vertical Localized port. The 2nd number "0" is a dummy. The 3rd number "2.5e+2" is the Positive Level of the port. The 4th number "1.797…" is a dummy. The 5th number "1.5e+2" is the Negative Level. The 6th number "3" means it is the net port 3.
22	The parameters of the 1st polygon. The 1st number "4" means there are 4 vertices. The 2nd number "0" is reserved. The 3rd number "20" means that the polygon is meshed as 20 cells per wavelength. It is automatically set by IE3D. The 4th number "3" means that the polygon is on layer 3. It is also automatically set. The 5th number means that the polygon is using the No.1 metallic type. This section is similar to the one in the XML file format.

23	The coordinates (X, Y, Z, S) of 1st vertex of the 4 vertex polygon defined in Section No.22. The meaning of S is discussed in Table B.3
24-39	The sections describe the other 9 polygons.
40	It is an empty line after all the polygons
41	The "0 0" means there is no optimization variable. It is the end of the file.

Appendix BA. Generally Efficient Matrix Solver (GEMS) and Adaptive Matrix Solver

In the IE3D 11.06, we have released the GEMS matrix solver. Like AIMS II and AIMS III, GEMS is an iterative matrix solver. Due to their iterative nature, they can be used to solve large structures with much less RAM or much faster. The GEMS should be much more robust than the AIMS solvers. AIMS solvers can normally be used for loosely coupled structures with infinite ground planes. When infinite ground planes do not present, the AIMS solvers are easy to fail in the iteration convergence. On the other hand, the GEMS solver can solve large structures without infinite ground planes. Its chance of failure is much lower. Both AIMS and GEMS do not neglect any coupling in the structure. For the initial release of the GEMS, it is only available for structures with infinite substrates. When finite substrates are used, it will be automatically disabled.

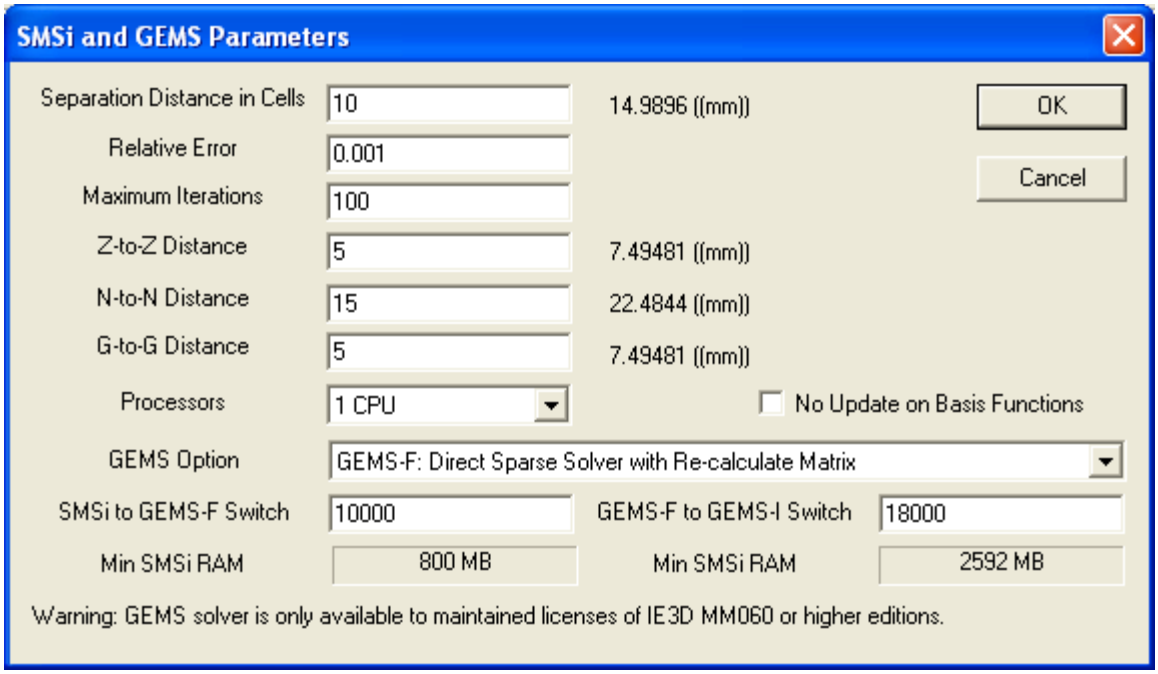


Table BA.1 GEMS Option

Case	Matrix Solver Used
GEMS-F: Direct Sparse Solver with Re-Calculated Matrix	GEMS will use direct sparse solver in the process. It will be faster while it may use more RAM than the GEMS-I options. For each iteration, GEMS will re-calculate the matrix. It is good for more planar structures because re-filling the matrix is fast.
GEMS-F: Direct Sparse Solver with Saved Matrix	GEMS will use direct sparse solver in the process and it will save the matrix in the hard-drive for re-use. It may take large amount of HD space in the temporary directory.
GEMS-I: Iterative Sparse Solver with Re-Calculated Matrix	GEMS will use iterative sparse solver in the process. It may be slower than GEMS-F. For each iteration, GEMS will re-calculate the matrix. It is good for more planar structures because re-filling the matrix is fast.
GEMS-I: Iterative Sparse Solver with Saved Matrix	GEMS will use iterative sparse solver in the process and it will save the matrix in the hard-drive for re-use. It may take large amount of HD space in the temporary directory.

There are different options for the GEMS solver. They are documented in Figure BA.1 and Table BA.1. The most important parameters are the Separation Distance in Cells. Normally, we set it to about 5-10. For the Relative Error, we can set it from 0.025 to 0.001. It seems that Relative Error = 0.025 may yield acceptable results in most the cases. However, to be safer, you can choose it to be 0.001. The Z-to-Z Distance, N-to-Z Distance, N-to-N Distances are not used temporarily. The G-to-G distance is used. You can set it to some value around 5. You can choose more than 1 CPU for it if you have multiple CPUs on your computer. It will only be valid for the GEMS-F choices.

Due to the fact that GEMS is much more robust than the AIMS solvers, we are able to combine GEMS with SMSi matrix solver to make it adaptive. When a user chooses SMSi or GEMS solver, the SMSi and GEMS Matrix Parameters dialog will come up. You need to setup the parameters correctly in order to use the adaptive matrix solver selection. Figure BA.1 shows the selection dialog. The different situations are documented in Table BA.2.

When you change the SMSi to GEMS-F Switch, the Min SMSI RAM value below it will change. It indicates how much RAM is required for the SMSi solver at the value. The same is true for the GEMS-F to GEMS-I Switch. For example, if you have 1 GB RAM, we will suggest you to choose the SMSi to GEM-F Switch no more than 10,000 with 800 MB RAM for SMSi at the switch. Normally, SMSi matrix solver speed is not easy to beat when the unknown number is below 5,000. Even it is possible the GEMS solver can beat the speed of SMSi with unknown = 5,000 with some special cases, you normally should not choose the SMSi to GEMS-F Switch too small because SMSi is certainly the most robust solver.

Please note the following fact. If you change the settings in a simulation, the settings will be available for the current instance only. If you want to change it for all the instances, you should change the settings in the PARAM->OPTIONAL PARAMETERS dialog of MGRID.

Table BA.2 Adaptive Matrix Solver selection based upon the SMSi and GEMS Parameters

Case	Matrix Solver Used
SMSi selected and the unknown number below the SMSi to GEMS-F Switch	IE3D will use SMSi solver.
SMSi selected and the unknown number above SMSi to GEMS-F Switch and below GEMS-F to GEMS-I Switch	IE3D will use GEMS-F with re-calculated matrix option except the structure is too highly packed. For very highly packed structure, SMSi will be used.
SMSi selected and the unknown number above GEMS-F to GEMS-I Switch	IE3D will use GEMS-I with re-calculated matrix option except the structure is too highly packed. For very highly packed structure, SMSi will be used.
SMSi selected and the unknown number is too small.	IE3D will use SMSi solver.
GEMS selected and the unknown number is below GEMS-F to GEMS-I Switch	IE3D will use GEMS solver with the selected GEMS option except the structure is too highly packed.
GEMS selected and the unknown number is above GEMS-F to GEMS-I Switch	IE3D will use GEMS-I with re-calculated matrix option except the structure is too highly packed.

Appendix BB. The Inductance at Very Low Frequency

IE3D is a high frequency full-wave EM simulator. The word "full-wave" means that we are solving the Maxwell's equations numerically with little approximation. In contract, "static" or "quasi static" solvers are solving the Maxwell's equations numerically by assuming the frequency is approaching 0 (DC) or some other assumptions which are only valid when frequency is approaching DC.

Full-wave EM simulators are supposed to be accurate at high frequency because they do not assume the frequency is approaching DC. How about the accuracy of full-wave EM simulators at low frequency approaching DC? All full-wave algorithms have a singularity at DC. All of them will fail when frequency is approaching DC. Depending upon the nature of the algorithms and the implementations of the algorithms, the low frequency limit will be different. In fact, this limit is structure dependent too. Usually, the limit is lower for larger structures and higher for smaller structures. From our tests, IE3D's low frequency limit is about 10 KHz for structures in the size of cm. Again, this is just a general rule only. The reality is that the limit is dependent upon the real structure. Also, the limit can be further pushed down by using quadruple precision in the matrix. It is expected that this limit can be pushed down to below 1 Hz in case we use quadruple precision. Certainly, it will double the RAM usage and possibly slow down the simulation for the same structure using quadruple precision.

At low frequency, it is more convenient to use lumped elements (RLC) to represent a circuit. Some people are curious about how the RLC equivalent circuit may behave at very low frequency and how accurate full-wave EM simulators are in finding the RLC equivalent circuit. Accurate modeling of loss effect due to finite thickness and finite conductivity is discussed in Appendix S. Loss effect corresponds to the R and we will not discuss R here. For most situations, C is normally less frequency dependent and we will not discuss it here. We will focus on the discussion of inductance at low frequency in this appendix.

Let's take a simple microstrip line as an example (Figure BB.1). A real microstrip line is of some width and thickness with finite conductivity. It should also have substrate and ground plane. To simplify the discussion, we assume the strip is of width W. The distance from the strip to the ground plane is H. We also assume it is air substrate. We assume the strip is PEC and the thickness is 0 and we will concentrate on how the ground plane is affecting the inductance at low frequency. The ground plane can be PEC or non-PEC with finite conductivity. The ground can be assumed to be infinitely extended or finite sized. We want to see how the inductance value of the microstrip structure may change when the frequency is approaching DC.

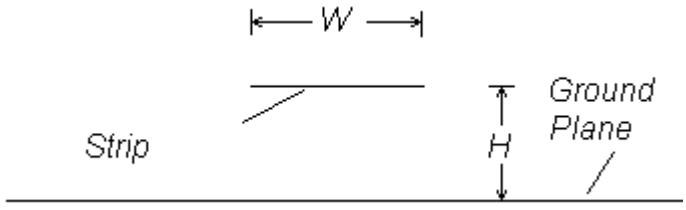


Figure BB.1 A microstrip with air substrate above ground plane.

If the ground plane is an infinitely extended PEC structure, we know we can solve the problem analytically using the mirror image to replace the ground plane (see Figure BB.2). For the parallel strip structure, we can use some conservative mapping in the complex domain to solve the structure analytically. The solution can be expressed as some special functions that can be solved numerically to high degree of accuracy. The inductance per unit length L will be frequency independent.

If the ground plane is a finite PEC structure, the equation governing the solution is the Laplace equation and it can be solved numerically. The inductance per unit length is also a frequency independent value.

Let's take a look at the case with infinitely extended ground plane and it is not PEC. At a specific frequency f1, we are able to solve the structure numerically. Assume the solved inductance per unit length is L1 at f1. Let's see how the L may change with frequency when the frequency is approaching DC.

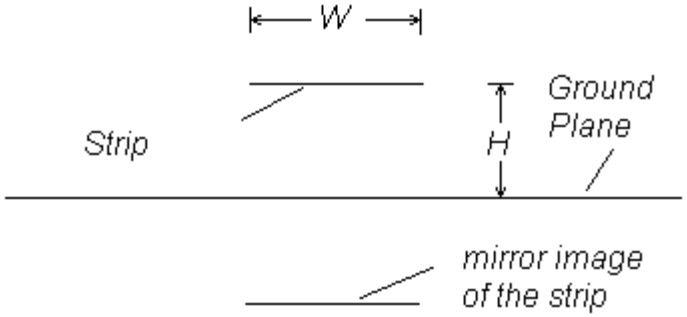


Figure BB.2 The solution is the same as the parallel strips with the original and the mirror image.

When the ground plane and the strip are PEC, we know the current is concentrated at the portion of the ground plane underneath the strip no matter what frequency it is. The current distribution does not change with frequency and it is why the L, which is dependent upon how the current flows, is frequency independent. When the ground plane is not a PEC and the frequency is approaching the DC, you will see that the current on the ground plane is becoming uniform assuming the conductivity of the ground plane is the same everywhere. Basically, the current takes the least impedance path to flow. If the ground plane is infinitely extended, you will see the current density on the ground plane is approaching 0 because the current is getting uniform in the cross-section of the ground plane. Eventually, we will expect the inductance per unit length at DC to reach a value L0 and L0 is usually smaller than L1 for a typical microstrip structure. It can be larger than L1 for other types of transmission lines such as coaxial lines. We will see a curve to be like what is shown in Figure BB.3.

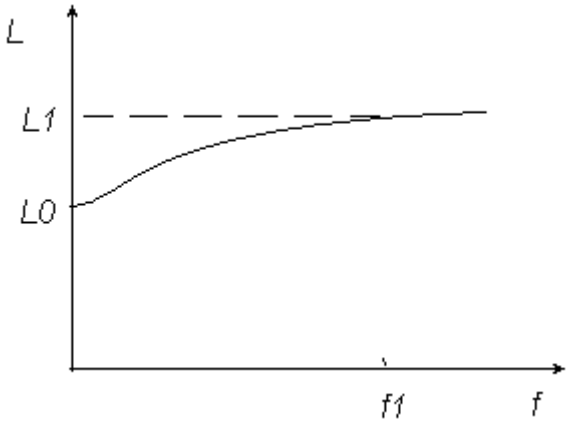


Figure BB.3 The possible frequency dependency of L for a non-PEC structure.

How big is L0? It is apparently determined by the cross-section and the conductivity of the strip and the ground plane. If the ground plane is infinitely extended, the current will be distributed in the infinite range and its contribution to the L0 is approaching 0. If the ground plane is finite, the current distribution will be approaching a non-zero number and it will contribute some thing to L0.

If the microstrip has a shielding box of rectangular shape (see Figure BB.4), you will see that the L0 is related to the size of the rectangular shape, the thickness and the conductivity of the walls. Even if the inner shape of the box is fixed, the L0 is not fixed because it is affected by the thickness of the walls. If the

bottom wall and the top wall of the box are not PEC while the side walls are PEC, can you imagine what will happen at DC? You will see that current flows only in the side walls and there will not be any current in the bottom and top walls because current takes the path with least impedance. In such a situation, the impedance on the PEC side walls is 0 and it is a non-zero value on the bottom and top walls. Therefore, the current will flow only in the side walls.

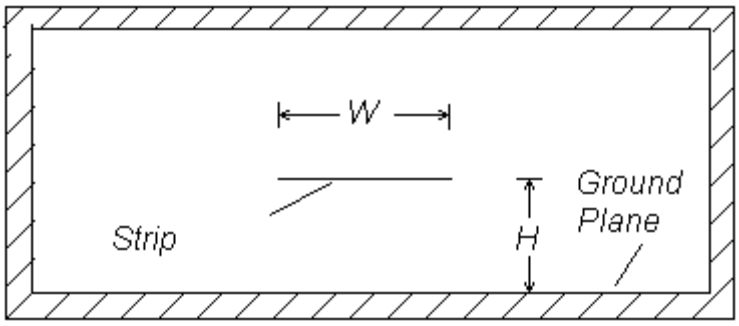


Figure BB.4 A microstrip inside a rectangular box.

The key point for this discussion is that the L of a microstrip structure is not only dependent upon the strip. It is also dependent upon the size, the shape, the thickness and the conductivity of the ground plane and the enclosure if it is boxed. Without knowing all the information about the strip and the ground plane or the box, the L value at low frequency can't be determined exactly. If you are after the convergence of the L value at very low frequency, you will not get it unless you have all the information on the strip, the ground or the shielding box.

In practical applications, a circuit is very complicated. We have to divide a big circuit into small pieces or sub-circuits. We model each sub-circuit and cascade them together to find the network parameters of the larger structure. As you can see, when the frequency is approaching DC, the L value is changing with the metallic patterns. When we divide a larger circuit into smaller blocks, the metallic pattern changes. It means that we are never able to find the L value at DC using the divided sub-circuits because the L value is dependent upon the metallic patterns of the complete structure. When we divide it into smaller sub-circuits, we are changing the metallic patterns of the complete structure and we are changing the value of the L.

Is such a situation miserable to a circuit designer? Yes, if the circuit designer is looking for the very accurate L value at DC or at some very low frequency. You can never get the precise value unless you build your model with complete information on the traces, the size and thickness of the ground plane. If the structure has an enclosure, you will need to find the size and the thickness and the conductivity of the enclosure. Any difference in the structure parameters will lead the L to converge to a different value at DC.

Fortunately, what a circuit designer really needs, in most situations, is the impedance. Please take a close look at the impedance of an inductor:

$$Z = j\,\omega\,L \qquad \text{(BB-1)}$$

When the frequency is very low, the |Z| is very small. The incomplete information about the complete geometry may not lead to the precise value of L (see Figure BB.5). However, the Z value usually is very small in this range and we may not need to worry about the small difference because of the small value.

In case you do need to worry about the small difference in the L value in very low frequency, you should make sure the following: (1) You should not divide your complete structure into sub-circuits. When you divide it into sub-circuits, you already change the L value of the structure with frequency approaching DC. If you solve the sub-circuits and cascade them together, the resulting L is in fact not the true value

anyway. (2) You should do a refined meshing. Ideally, you can use refined volume meshing in order to capture the field and current change in the metallic structure. You can also use multiple sheets of metal in both the horizontal and vertical directions (see Figure BB.6). Don't expect that refined meshing in the traces only can yield the precisely inductance value. The refined meshing should also be applied to the ground plane and the box if there is any. The refined meshing should be in the metallic structure for MOM while it should be everywhere for FEM and FDTD types of field solvers. This kind of refined meshing may use up the computation resources very rapid for even a very simple structure. It may defeat the purpose of engineering applications. For engineering applications, the tube like model for thickness discussed in Appendix S should normally be accurate enough because making finer meshing on the strip does not mean the solved inductance value at very low frequency is closer to the reality.

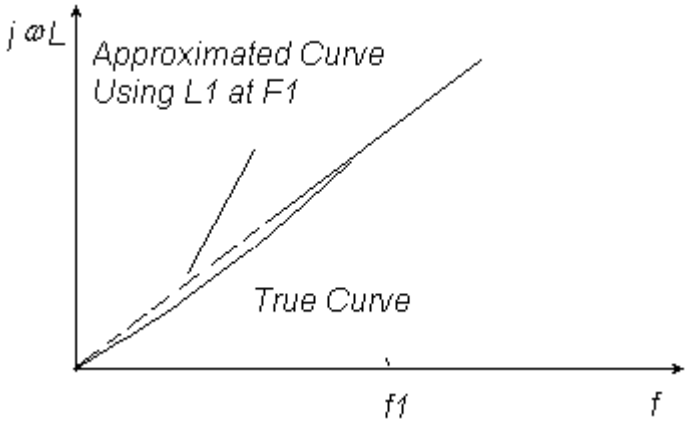


Figure BB.5 The impedance curves of a typical inductor at low frequency.

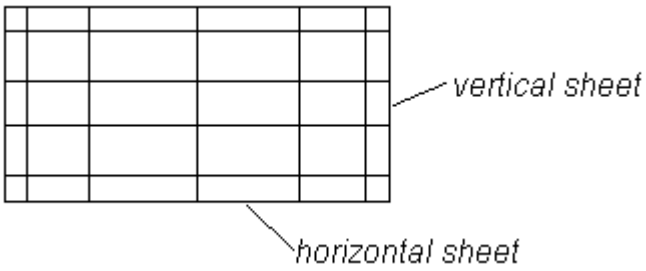


Figure BB.6 The multi horizontal and vertical model in the cross-sectional of a metallic structure for capturing the low frequency inductance change.

This is a very interesting topic. Highly technical persons have given suggestions on it. Their efforts are greatly appreciated.

Appendix BC. Antenna Correlation Calculation

It is expected that multiple antennas working in the same bandwidth may increase the capacity of the communication system in wireless applications. However, most wireless devices are small in sizes. The multiple antennas have to be closed placed. Closely placed antennas working in the same bandwidth will coupled to each other. The coupling may eventually reduce the capacity of the communication system. Antenna correlation is a measure on how good two antennas may work together to enhance the capacity of the system. The lower the antenna correlation is the better performance the coupled antennas can achieve. Antenna correlation calculation is implemented into the IE3D 11.5. This appendix discusses how we can find antenna correlation from a structure.

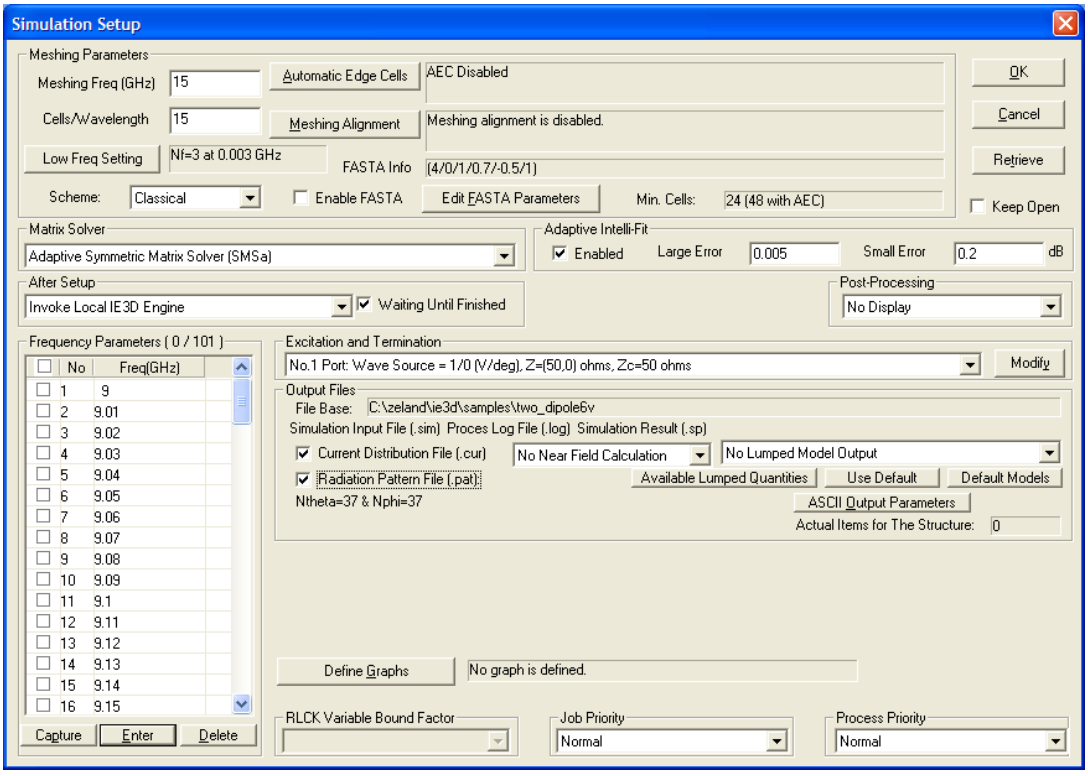


Figure BC.1 The Simulation Setup dialog for two_dipole6v.geo

To find the antenna correlation of a structure, we start from an antenna structure with 2-ports. Saved in .ie3d\samples\two_dipole6v.geo is an example. It is a 2 dipole structure separated by 3 mm. Each dipole is 15-mm long. Each dipole is fed by a v-localized port. We will setup a simulation from 9 to 10 GHz for 101 frequency points (see Figure BC.1). Please select Radiation Pattern File for pattern calculation with 37 points in both elevation and azimuth directions. MGRID will also prompt you to enter the Excitation and Termination of the ports. The excitation and termination information will be used to find the radiation pattern with the specified excitation and termination condition. The radiation pattern will be saved into the .pat file. For antenna correlation calculation, we don't need to worry about the excitation and termination at this time. We will use the General Radiation Pattern saved in the .mpa file for the calculation. Please select OK in the Simulation Setup dialog to simulate the structure. It takes seconds to finish the simulation. You can use either MGRID or PATTERNVIEW to find the antenna correlation of a multi-port antenna from the created with the general pattern.

On PATTERNVIEW, you can select Edit->Process General Pattern. On MGRID, you can just access the General Pattern's properties dialog (Figure BC.2).

Antenna correlation of a 2-port antenna system involves two antenna patterns with different excitations and terminations. For the 1st pattern, the port 1 of the antenna is excited while the port 2 is terminated with appropriate impedance (or the Zc for port 2). For the 2nd pattern, the port 2 is excited while the port 1 is terminated with appropriate impedance (or the Zc for port 1). In the Process General Pattern or General Pattern properties dialog, you are allowed to define the magnitude and phase of the excitation, and the source impedance and the Zc of each port. You are allowed to define different values for excitations and terminations at each frequency. Starting from IE3D 14, the excitations are terminations are saved into the General Pattern. You can select Save to save the changed excitations and termination into the general pattern file. You can also select "Find Pattern of Structure with Specified Excitation" and select the "Go!" button beside it to save the radiation pattern into a specific pattern (.pat) file.

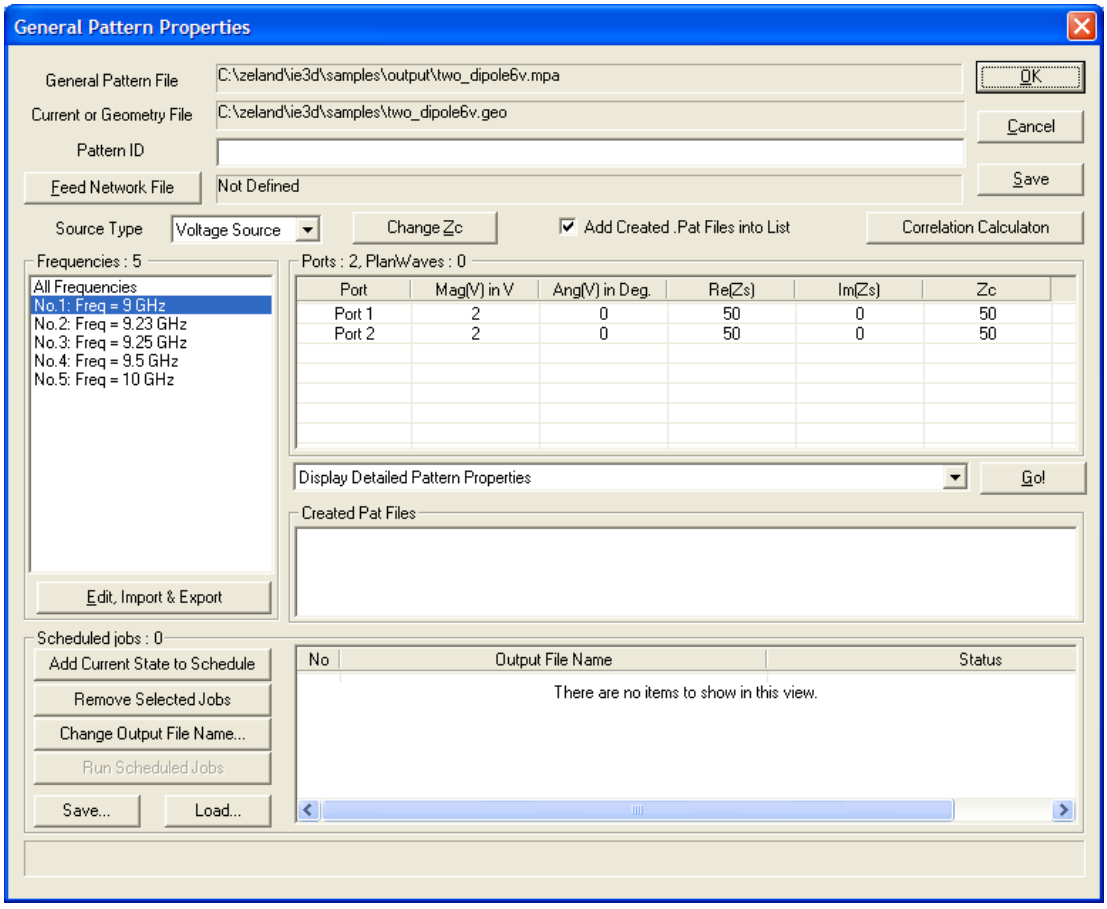


Figure BC.2 The Process General Pattern dialog on PATTERNVIEW.

For antenna correlation calculation, only the Zc at each frequency for each port defined in the list box is useful. It will be the terminating impedance of the port at the frequency when the other port is excited. By default, the Zc of each port is 50-ohms. Please understand this is not the Zc of the transmission line. It is the Zc you want the s-parameters to be normalized to. In antenna correlation calculation, it is the terminating impedance of the port when the other port is excited. Please define the appropriate Zc for each port (at each frequency if it is frequency dependent).

Select Correlation Calculation button in the dialog. It will show you the Antenna Correlation Calculation dialog (Figure BC.3). The antenna correlation is associated with the Propagation Model used. Six different propagation models are implemented. Each model is defined with different parameters. You can double click at the propagation model in the list to edit the parameters.

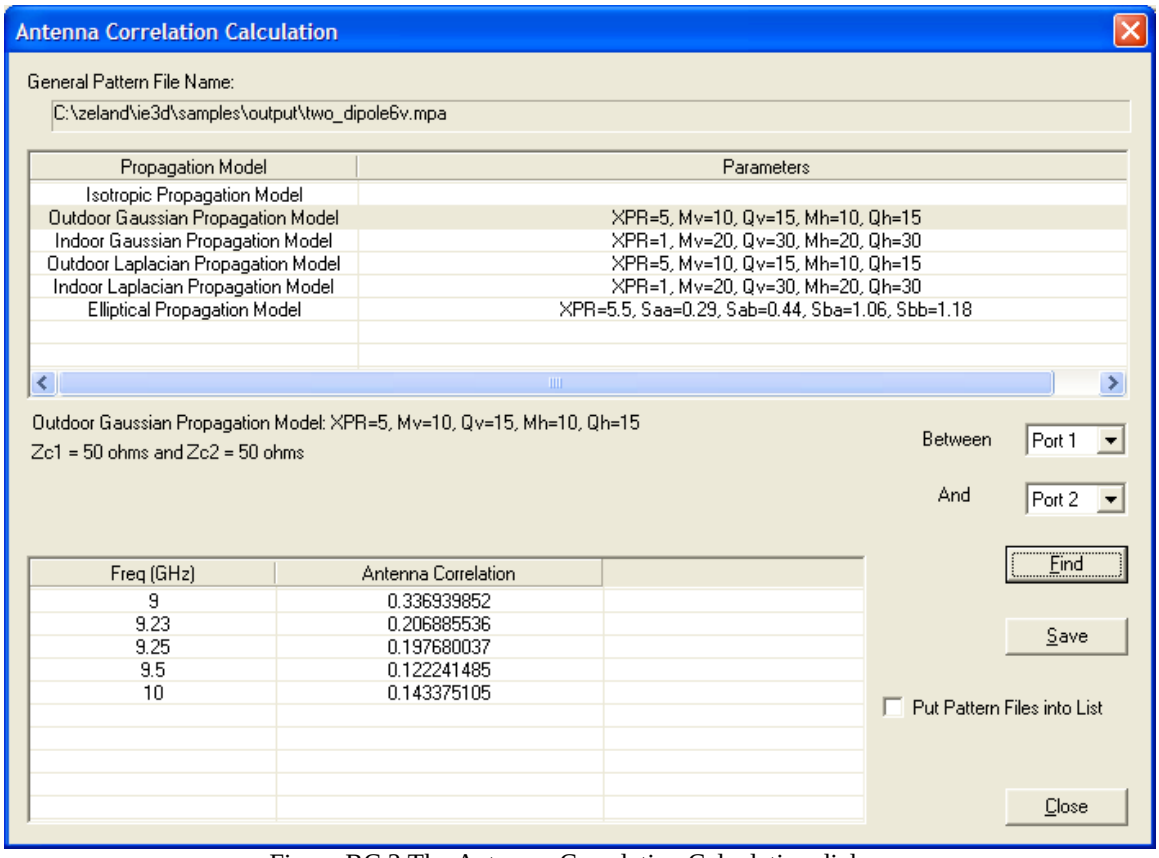


Figure BC.3 The Ante nna Correlation Calculation dialog.

Please select the Outdoor Gaussian Propagation Model in the list. Select Find button. It will find the Antenna Correlation coefficient of the structure at each frequency point. The results are listed in the list box in the lower portion of the dialog (see Figure BC.3). To save the result into an ASCII file for document purposes, please select save button to save the data into .\ie3d\samples\output\two_dipole6v.txt file. The two antenna patterns with different excitations are saved into: two_dipole6v_e1.pat and two_dipole6v_e2.pat in the directory. \ie3d\samples\output.

On IE3D V14, we have implemented Antenna Correlation Calculation for more than 2-port antennas.

Appendix BD. IE3D FASTA for Efficient Simulation of Large Structures

One of the main features of IE3D 11.5 is the implementation of IE3D FASTA (Full-wave Accelerated Simulation Technology Algorithm). IE3D is a full-wave EM simulator with an advanced integration equation MOM implementation. It offers the best accuracy for planar and 3D structures in MMIC, RFIC, microwave circuits, signal integrity and antennas. It is much more flexible for other planar EM simulators due to the general non-uniform and full-3D implementation in IE3D. IE3D is also extremely efficient for structures below 5,000 unknowns.

When structures getting bigger and bigger, the standard procedure of IE3D will require more and more RAM and the simulation will take longer and longer. The standard procedure of IE3D uses an advanced symmetric matrix solver (SMSi). For SMSi, its RAM requirement is proportional to N^2 while the simulation time is proportional to N^3 when the number of unknowns is more than 3,000. Table BD.1 lists the RAM and time requirements for a typical planar structure.

Table BD.1 Typical RAM and time requirements for patch antenna structures on a P4-3.0 GHz PC.

Unknowns	1,000	2,000	4,000	8,000	16,000
RAM	8 MB	32 MB	128 MB	512 MB	2 GB
Time/Freq	3 seconds	15 seconds	110 seconds	850 seconds	5,600 seconds

Please note that relationship between the number of unknowns and the simulation time is not fixed. It is structure dependent. Normally, single substrate structure with mainly horizontal traces is the fastest case. Multi-substrate layers with strong 3D polygons (vias, wire bonds, and thickness polygons) may slow down the simulation significantly. For example, a 1,000 unknown patch antenna structure may take 3 seconds per frequency point. A BGA structure with wire bonds and vias of 1,000 unknowns may take 30 seconds per frequency to solve.

Some users may have encountered problem in solving structures more than 12,000-unknown structure on a Win32 PC with 2 GB or more RAM installed. The matrix for 12,000-unknown structure is about 1.15 GB. They may see the IE3D engine failing to allocate the matrix. They may wonder why IE3D engine fails to allocate the memory while it is enough. The reason is due to the fact that IE3D engine requires the memory for the whole matrix to be in a flat memory with continuous space. A typical Win32 OS can accept maximum 2+2 GB or 4 GB RAM. In fact, we would like to suggest users not to install more than 2 GB of RAM because it can be a waste. Win32 uses maximum 2 GB RAM for data and 2 GB RAM for the program. Most programs are small and they don't occupy much RAM. However, a large matrix may use all the 2 GB RAM. It is possible that a Win32 OS may be configured for 3+1 GB with 3 GB data space. However, we have tried it and we have never been able to allocate more than 2 GB data for IE3D. For the 2 GB data in a Win32 OS, it will allocate the memories for system use and all program use. It may allocate some memory here or there in the flat 2 GB space. When IE3D engine tries to allocate 1.8 GB space for a 15,000 unknown structure using SMSi, it sends a request to the OS. The OS may not be able to find a continuous memory space of 1.8 GB in the 2 GB space because it may have allocated some space here or there to other programs. It will reject the request from IE3D engine and the IE3D engine will fail to continue. That is the reason why IE3D for Win32 may fail to solve some structures with more than 12,000 unknowns even enough RAM is installed on the computer. To solve bigger structures, you certainly can go for the IE3D for Win64 on Intel EM64T or AMD 64. You should be able to use much more than 2 GB RAM on the IE3D for Win64. Certainly, you can rely on the special features of IE3D for solving large structures using limited RAM and the IE3D FASTA is for such a purpose.

As it may be pointed out earlier, the simulation time for IE3D can mainly divided into 2-portions: (1) Filling Matrix Time (FMT); (3) Solving Matrix Time SMT). The FMT is proportional to N^2 while the SMT using the default SMSi solver is proportional to N^3. In some sense, the total simulation time (TST) can be represented as:

$$TST = FMT + SMT \quad \text{BD-1}$$

For the standard SMSi solver, the total simulation time denoted as: TST_smsi can be represented as:

$$TST\_smsi = FMT\_smsi + SMT\_smsi = C_f N^2 + C_s N^3 \quad \text{BD-2}$$

Where C_f and C_s are kinds of constants when the problem type is specified.

In fact, C_s is structure independent while C_f is structure dependent. The C_f for a patch antenna structure is very small while it is much bigger for a full-3D structure containing vias, wirebonds and thickness polygons.

No matter what type a structure is, when it is getting bigger and bigger, the dominant part of TST_smsi is the SMT_smsi or the Solving Matrix Time. As it is shown in Table BD.1, the SMT_smsi can increase quite fast with increasing number of unknowns. For a 16,000-unknown problems, it may take about 15,000 seconds per frequency point or the TST_smsi = 15,000. In fact, the FMT_smsi may be much less than 1,000 while the rest is the SMT_smsi.

To reduce RAM requirement and the simulation time, we have implemented different kinds of iterative matrix solvers. The latest iterative matrix solvers are the AIMS II, AIMS III (advanced iterative matrix solvers) and GEMS-F and GEMS-I (generally efficient matrix solvers). AIMS and GEMS matrix solvers are able to solver large loosely coupled structures using much less RAM. We are able to use AIMS II and GEMS-F solvers to solve a patch antenna array structure with more than 60,000 unknowns using less than 600 MB RAM in about 2 hours per frequency point. In comparison, a 60,000-unknown structure may require about 32-GB RAM and it may take tens of or even more than one hundred hours per frequency point. We may be able to further reduce the RAM requirement by using the AIMS III and GEMS-I matrix solvers while AIMS III and GEMS-F may be as fast as AIMS II and GEMS-F and they may be even much slower. In the extreme, they may be even slower than SMSi while they always require less RAM.

What are the similarity and differences between AIMS II and GEMS-F? They use different iterative algorithms. They both require the user to define a Separation Distance (SD). The SD is used to sparsify the matrix so that we can use iterative procedure to solve the problem. Both of them can yield numerically exact if the iterative procedures converge. When we say “numerically exact”, it means that it does not apply any approximation unless it may suffer from numerical round-off error which can never be avoided. The difference between AIMS II and GEMS-F is that SD for AIMS II normally needs to be around 10 and you can reduce it much, while the SD for GEMS-F can be reduced down to 6 or 5 or 4. In the other word, GEMS-F may be able to solve more strongly coupled structures than AIMS II. However, AIMS II can solve multi-port structures without slowing down the iterative procedure. For the GEMS-F, the simulation time is proportional to the number of ports.

In summary, we have found that iterative matrix solvers are good for loosely coupled structures such as antenna arrays. However, they are not good for strongly coupled structures. Typical strongly coupled structures in modern designs are multi-layered spiral inductors and transformers with shields, PCB structures with wire bonds, antenna structures with finite ground planes. The chances of successful convergence for iterative matrix solvers for strongly coupled structures are very low or even impossible for strongly coupled structures. The RAM and simulation time requirements for iterative matrix solvers for solving strongly coupled structures may even be more than those for SMSi in many cases. Also, iterative matrix solvers are only good for structures with few ports. For large number of ports, the simulation time can be very long. For example, to simulate a 400-element antenna array with 400-ports using GEMS-F will be 400 times slower than simulating such a structure with 1-port. The best you can do might be using the AIMS II solver for multi-port loosely coupled structures. Iterative matrix solvers by nature cannot always guarantee convergence. They always have stability problem especially for not-so-loosely-coupled structures.

Can we solve a general structure (not limited to loosely coupled structure) with the same accuracy as the SMSi matrix solver with significantly reduced time? At this time, we have not seen such a possibility. Such a goal may not be achievable. However, we can revise our goal to the following: We would like to solve a general structure with much less RAM in much shorter time with slightly reduced simulation accuracy.

It is always possible to simulate a general structure with less RAM and shorter time with reduced accuracy on the standard SMSi solver. For a general microwave structure, we recommend users to use 20-cells per wavelength (or Ncells = 20) and appropriate AEC for meshing. We normally can get very accurate results. In fact, we can reduce Ncells from 20 to 15 with little accuracy degrading. To further reduce RAM, we can push Ncells down to 10. We should normally not push it below 10. Normally, it will yield bad results with Ncells = 10 or below. Reducing Ncells may significantly reduce the RAM and simulation time. For example, reducing Ncells from 20 to 15 may save 30-70% of the RAM or 30-90% of the simulation time.

Certainly, disable AEC (Automatic Edge Cells) is a good way to reduce RAM and simulation time. It may reduce the RAM and simulation time by an order. However, disable AEC may risk losing accuracy significantly especially for structures containing strongly coupled traces.

In summary, it is always possible to optimize the meshing of a general structure and reduce the meshing density appropriately to get faster results with much less RAM with little reduction in accuracy. Experienced users should try to control the meshing for more rectangles and fewer triangles in the meshing. One rectangle is equivalent to at least two triangles of the same size. If we can use more rectangles and fewer triangles, we are able to reduce the number of unknowns, and then the simulation time and RAM. While reduce the meshing density and optimize polygons for better meshing smartly are good ways to reduce RAM and simulation time with little loss in accuracy, there are always limits for it. You can't do much beyond the limit. Also, it requires the users to be experienced enough in order to optimize the meshing appropriately. Optimize the polygons for meshing normally will need manual dividing and merging of polygons. It can be a tedious work.

Can we further reduce the RAM and simulation time without reducing the meshing density with little loss of accuracy? Researchers have developed different types of fast algorithm to solve large MOM problems. For example, people have developed Fast Multi-pole Method (FMM) and Multi-Level Fast Multi-pole Method (MLFMM). MLFMM is good for structures in uniform dielectrics. It is able to solve huge antenna and scattering structure in uniform dielectric environment. However, MLFMM can't apply for structures with different dielectrics. In reality, how many large structures can have uniform dielectrics? There are few. In some sense, the application range of MLFMM is greatly reduced by the requirement of uniform dielectrics. Also, MLFMM has made some assumptions. The numerical errors caused by the assumptions are still not known. People just take them as granted. MLFMM reduces RAM requirement and simulation time by grouping cells from meshed structure. Normally, they require the structures to be solved to be much larger than wavelength in order to save RAM and time. This is the other severe requirement. This requirement makes it difficult to solve low frequency problems. Low frequency MLFMM methods are also developed to solve electrically small structures. However, the methods are not good to electrically large structures. In some sense, MLFMM can't apply to general structures in a wide frequency bandwidth. In modern electronic designs (wireless and semiconductor), wide frequency bandwidth are common place. Digital circuits and UWB communication systems are all wide bandwidth.

Implementing an efficient method for wide frequency bandwidth and generally layered or 3D structures is strongly demanded and it is an extremely tough task. To achieve this goal, we have implemented the IE3D FASTA or Full-wave Accelerated Simulation Technology Algorithm. The detail algorithm and implementation for FASTA are confidential and we will not discuss them here. One thing similar to MLFMM is that IE3D FASTA also uses grouping. In some sense, we try to group the cells together in order to reduce the RAM and simulation time. There are many advanced technologies involved in handling the grouped polygons in IE3D FASTA. By appropriate grouping, we are able to solve general

structures with significantly reduced RAM and simulation time while we do not reduce the meshing density with little loss in accuracy in IE3D FASTA.

The IE3D FASTA is released in IE3D 11.5. IE3D FASTA is a very sophisticated technique. We may not be able to optimize the algorithm for the best accuracy and efficiency in this initial release. There is still much room for further development. We may be able to further improve the algorithm significantly in future versions.

Enabling IE3D FASTA is very simple. You just need to check FASTA in the meshing parameters in Basic Parameters dialog or Simulation Setup dialog. There are a few parameters you may change to control IE3D FASTA for efficiency and accuracy (see Figure BD.1). You may change them for different types of structures with guidelines documented in Table BD.2.

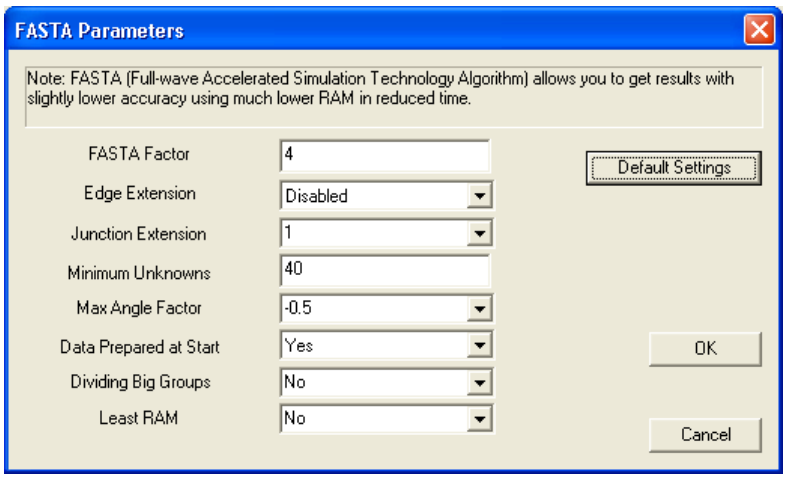


Figure BD.1 The dialog for FASTA parameters.

Table BD.2 The FASTA parameters

Parameter	Description
FASTA Factor	It is an important parameter. It is used to control how big a group is. Its value should be from 2 to 10. The default is 4.
Edge Extension	It is not a very critical parameter. Just use the default value of "Disabled".
Junction Extension	It is not a very critical parameter. Just use the default value of 1.
Minimum Unknowns	It is an important parameter. It is also used to control how big a group is. For more transmission line oriented structures, you can use a smaller value such as 40. For patch antenna arrays, you may want to choose a bigger value. Its value ranges from about 20 to 300 or even bigger.
Max Angle Factor	It is not a very critical parameter. Just use the default value of –0.5.
Data Prepared at Start	It is very critical for the efficiency. Please use the default value of "Yes".
Dividing Big Groups	It is an important parameter. It is also used to control the divided groups. The default value is "No". Normally, we should not select "Yes" even though it may reduce the RAM requirements. However, selecting "Yes" for finite ground structures may slow down the simulation significantly while it normally can save RAM. For some structures, it may be even slower than using SMSi matrix solver while it always uses less RAM.
Least RAM	It is an option not implemented in IE3D 11.5 yet. It will be used in future version.

Tables BD.3, BD.4 and BD.5 document the comparisons between IE3D FASTA, SMSi matrix solver and GEMS-F matrix solver on solving 8by2, 8by4 and 8by8 patch antenna array (see Figure BD.2). For

IE3D FASTA, we choose the FASTA Factor = 4, Minimum Unknowns = 100 and Dividing Big Groups = No. For the GEMS-F solver, we choose the Separation Distance = 7 cells and Relative Error = 0.001. It is a loosely coupled structure. Both IE3D FASTA and GEMS-F can save much RAM and simulation time for the 3 structures. IE3D FASTA is even better than GEMS-F. The results of GEMS-F should almost identical to the SMSi results while there is some slight difference in the IE3D FASTA results (see Figure BD.3). IE3D FASTA is a generally stable and faster scheme while GEMS-F is only good for loosely coupled structure. The tradeoff of IE3D FASTA is that it may reduce the accuracy and this degrading is normally very small. The level of difference in results can be seen even with different settings for meshing and you normally may not need to worry about it.

Table BD.3 The comparison on the simulations between different solvers for the 8x2 antenna array saved in C:\MentorGraphics\<latest_release>IE3D\SDD_HOME\IE3D\ie3d\samples\8by2.geo.

Solver	SMSi	FASTA	GEMS-F
Unknowns	3928	3928	3928
RAM Required (MB)	125	14	30
Seconds / Freq	135	34	100

Table BD.4 The comparison on the simulations between different solvers for the 8x4 antenna array saved in C:\MentorGraphics\<latest_release>IE3D\SDD_HOME\IE3D\ie3d\samples\8by4.geo.

Solver	SMSi	FASTA	GEMS-F
Unknowns	7868	7868	7868
RAM Required (MB)	500	28	59
Seconds / Freq	825	161	402

Table BD.8 The comparison on the simulations between different solvers for the 8x8 antenna array saved in C:\MentorGraphics\<latest_release>IE3D\SDD_HOME\IE3D\ie3d\samples\8by8.geo.

Solver	SMSi	FASTA	GEMS-F
Unknowns	15831	15831	15831
RAM Required (MB)	2,086	72	120
Seconds / Freq	5338	617	2100

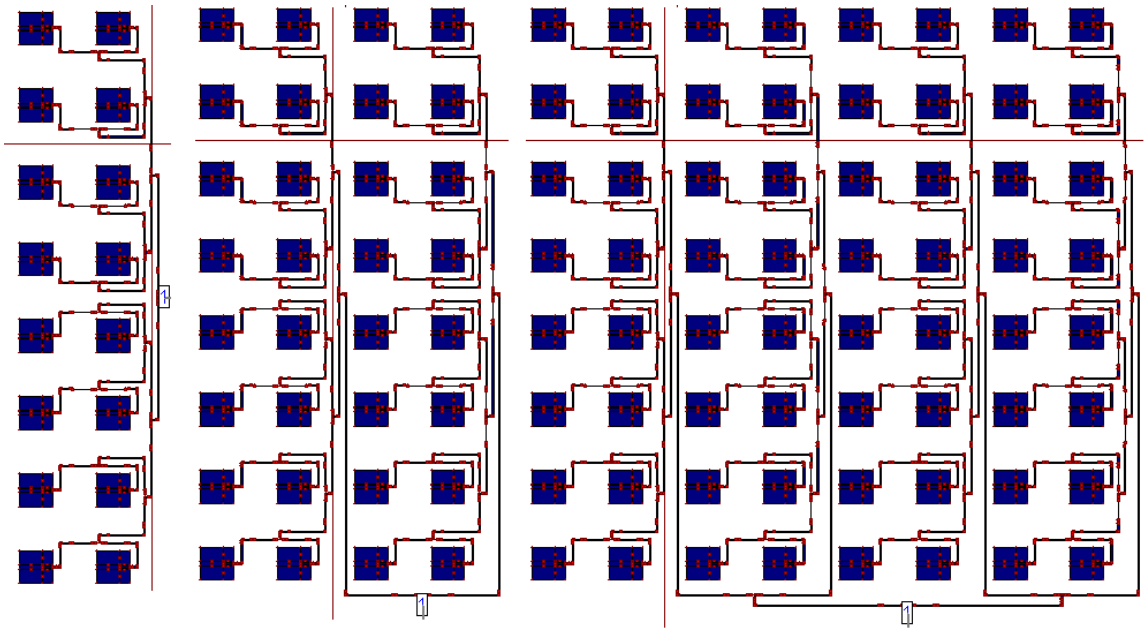


Figure BD.2 The 8by2, 8by4 and 8by4 antenna arrays for comparison for IE3D FASATA.

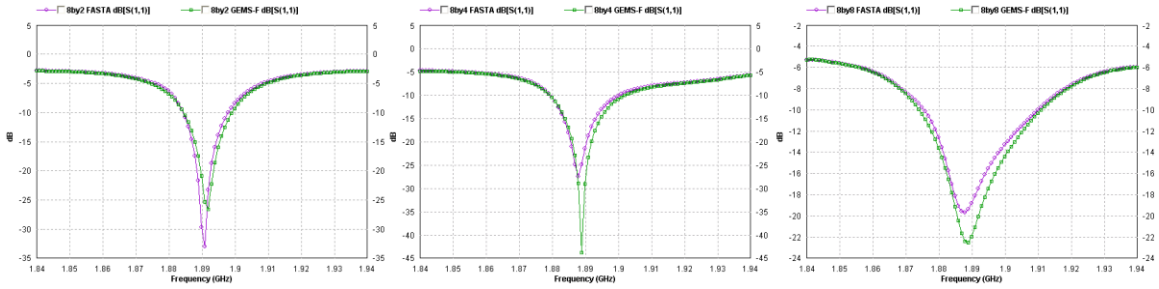


Figure BD.3 Comparison between FASTA and GEMS-F on the 8x2, 8x4 and 8x8 patch arrays.

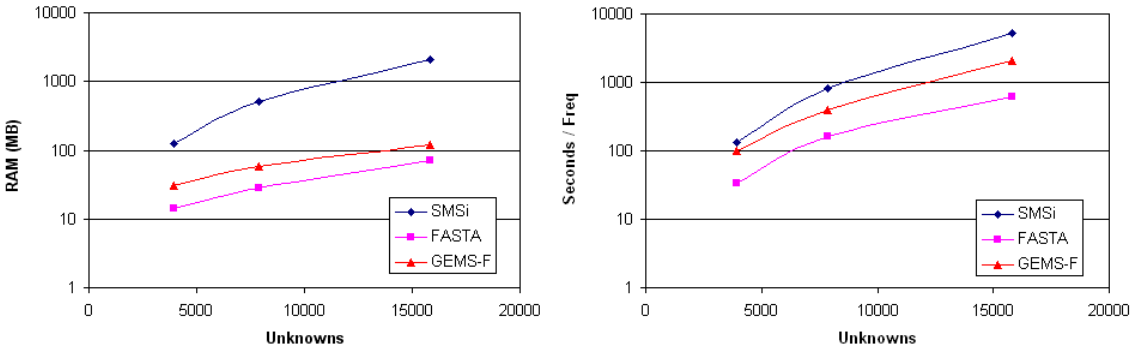


Figure BD.4 Comparison between FASTA and GEMS-F on the patch arrays.

The patch antenna arrays examples are for loosely coupled structures. For loosely coupled structures, both FASTA and GEMS-F are working very well. How about strongly coupled structures? We will use some spiral inductors examples to check it. A typical spiral inductor may have many turns. There are strong coupling between the turns. It is no longer a loosely coupled structure. We are going to use three different structures: oct_spiral_layers.geo, two_oct_spirals_layers.geo and four_oct_spirals_layers.geo in C:\MentorGraphics\<latest_release>IE3D\SDD_HOME\IE3D\ie3d\agif directory. The structures are shown in Figure BD.5. The structures are strongly coupled structures. If we choose GEMS-F solver, IE3D will detect it is not good for GEMS-F and it will switch back to SMSi. We can change some settings to force it to run on GESM-F solver. However, the simulation time and RAM requirements will be larger than the SMSi and it loses the meaning. We will just compare the SMSi and FASTA results. For the FASTA, we choose FASTA Factor = 4 and Minimum Unknowns = 40. The comparisons are shown in Tables BD.9, BD.10 and BD.11, and Figures BD.6, BD.7 and BD.8.

Table BD.9 The comparison on the simulations between different solvers for one octagonal spiral saved in C:\MentorGraphics\<latest_release>IE3D\SDD_HOME\IE3D\ie3d\agif\oct_spirals_layers.geo.

Solver	SMSi	FASTA	GEMS-F
Unknowns	1845	1845	N/A
RAM Required (MB)	43	19	N/A
Seconds / Freq	37	30	N/A

Table BD.10 The comparison on the simulations between different solvers for two octagonal spirals saved in C:\MentorGraphics\<latest_release>IE3D\SDD_HOME\IE3D\ie3d\agif\two_oct_spirals_layers.geo.

Solver	SMSi	FASTA	GEMS-F
Unknowns	3690	3690	N/A
RAM Required (MB)	125	38	N/A
Seconds / Freq	153	73	N/A

Table BD.11 The comparison on the simulations between different solvers for four octagonal spirals saved in C:\MentorGraphics\<latest_release>IE3D\SDD_HOME\IE3D\ie3d\agif\four_oct_spirals_layers.geo.

Solver	SMSi	FASTA	GEMS-F
Unknowns	7383	7383	N/A
RAM Required (MB)	470	68	N/A
Seconds / Freq	760	254	N/A

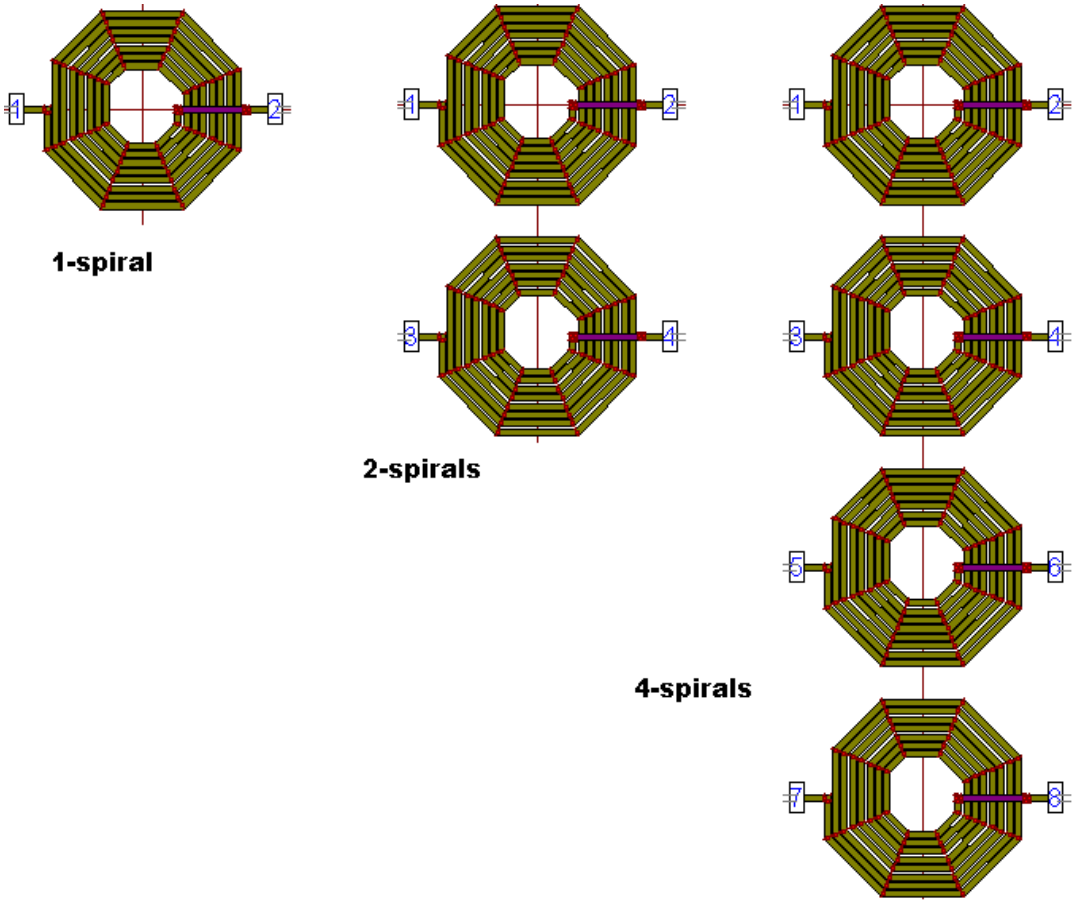


Figure BD.5 The structures with 1, 2 and 4-spirals.

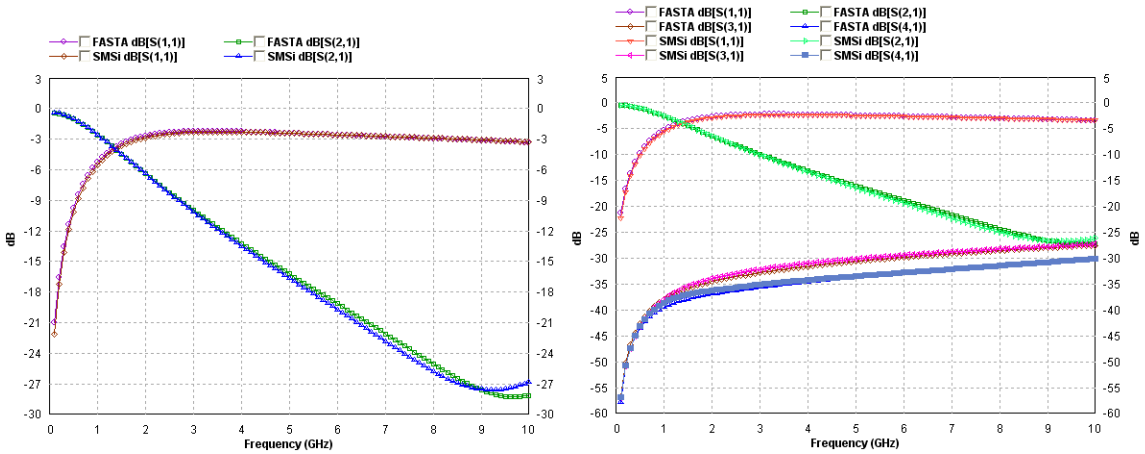


Figure BD.6 The comparison on the two and four spirals between FASTA and SMSi.

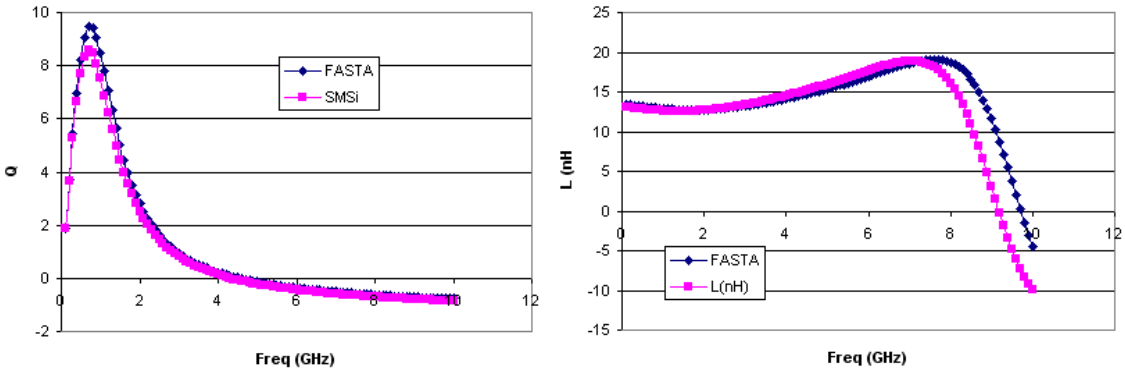


Figure BD.7 The comparison on L and Q of the single spiral between FASTA and SMSi.

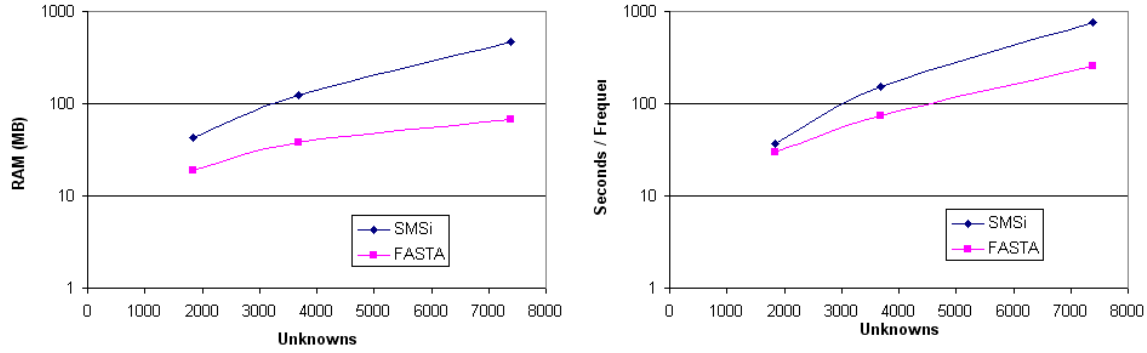


Figure BD.8 The comparison on computer resources between FASTA and GEMS-F on spirals..

As you can see, there is little difference between the SMSi and FASTA results. Normally, the differences are more obvious for L and Q values because L and Q values are more sensitive than s-parameters. Again, you can see FASTA is much more efficient than SMSi even though SMSi is also doing a good job. These are not loosely coupled structures. FASTA is still doing well even though the advantage is smaller than the cases for loosely coupled structures.

FASTA can always save the RAM significantly. It may be able to save simulation time significantly for large structures. For small structures, the saving in simulation time may be small or it may even be slower than the SMSi matrix solver. Anyway, SMSi matrix solver is extremely efficient and accuracy for small and medium size structures and we really don't need FASTA when we are solving small and medium size structures.

Appendix BE. Using AGIF for Automated IE3D Simulations from GDSII and Other Layouts

We have implemented Automatic Geometry to IE3D Flow (AGIF) in IE3D 11.1 for automated IE3D EM simulations from GDSII files and other layouts. Users are able to build IE3D models and perform automated IE3D simulations directly from GDSII files, Cadence Virtuoso layouts and Cadence Allegro layouts. AGIF simplifies IE3D full-wave EM simulations significantly for MMIC, RFIC, PCB and other designs because users can run IE3D simulations directly from other layout tools by just one click of the mouse.

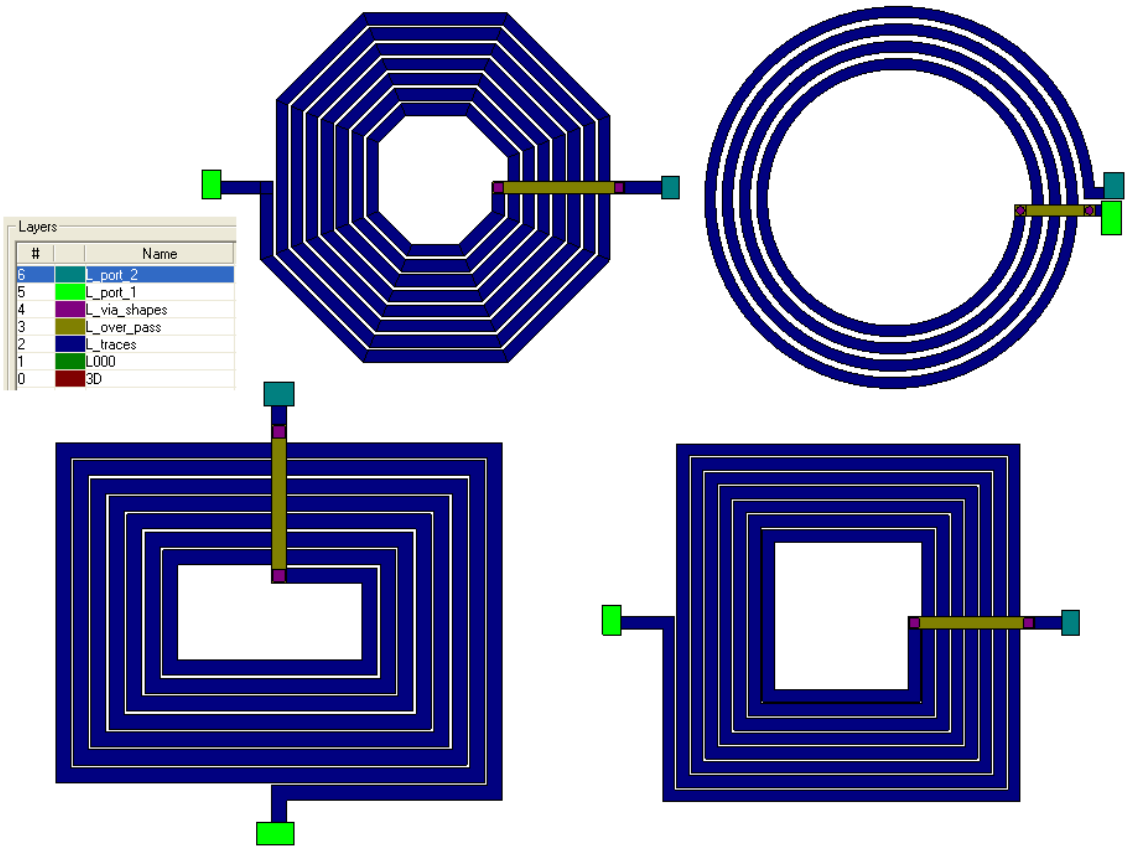


Figure BE.1 Four different spiral inductor layouts with layered polygons in GDSII files.

Complete and detailed documentation of AGIF is in a separate document "AGIF User's Guide" in C:\MentorGraphics\<latest_release>IE3DIE3D\SDD_HOME\IE3D\ie3d\agif. In this appendix, we will just go through some simple examples so that you can know what AGIF can do. Please note that AGIF's capability has expanded much in AGIF 11.5 while the AGIF User's Guide was written based upon AGIF 11.1. The basic concepts are well-documented in the AGIF User's Guide while the interfaces discussed might be different. Please note that AGIF is compiled using cross-platform compilers for both Windows and Linux. It is using character "/" (Linux convention) instead of "\" (Windows convention) for anything related to path while both characters are accepted.

Figure BE.1 shows 4 different spiral inductors in 4 different GDSII files: oct_spiral_layers.gds, circ_spiral_layers.gds, rect_spiral_layers.gds and square_spiral_layers.gds in the directory: C:\MentorGraphics\<latest_release>IE3DIE3D\SDD_HOME\IE3D\ie3d\agif. How can we model and simulate them on IE3D? In a standard IE3D importing procedure, we need to do the following for each of the GDSII files: (1) Import the GDSII file into MGRID. (2) Define the Substrate Layers; (3) Perform Simulation Setup to define the frequency parameters; (4) Define the metallic strip types; (5) Map each layer

to the appropriate z-coordinate; (6) Assign a metallic strip to a layer. (7) Use the shapes of some polygons to build the vias between other layers; (8) Define ports on the structure; (9) Run Procedure to create the IE3D model or select Simulate to simulate the

The basic procedure is the above steps with minor revision. For example, if we want to remove some vertices on curved structure on the circular spiral inductor in order to simplify the model without losing much accuracy, we should use the command Adv Edit->Remove Redundant Vertices on Curvature to achieve it. In case we want to model the structure with true thickness, we should use command Edit->Layers->Grow Thickness on Layer to build the thickness on the traces. IE3D has implemented many advanced geometry modeling features to allow you to import a structure and clean it to fit an IE3D simulation. For each GDSII file, we have to go through a similar procedure to fit it into an IE3D simulation. It can be very tedious when we have many structures to simulate. In case we make a mistake in a step and we did not notice it, we will have to re-do the procedure and it can be a waste of time.

In a typical MMIC or RFIC design, we may have to simulate many different shaped structures with similar layers and metallic type configurations. If we can automate the procedure to create an IE3D EM model and perform the simulation, we can save much effort from the users. AGIF is implemented for such a purpose. We will go through the procedure for modeling and simulating the 4 GDSII files in Figure BE.1. We may just give some brief explanations on the steps we take if you don't understand the reasons for a specific step, please read the AGIF User's Guide for more information.

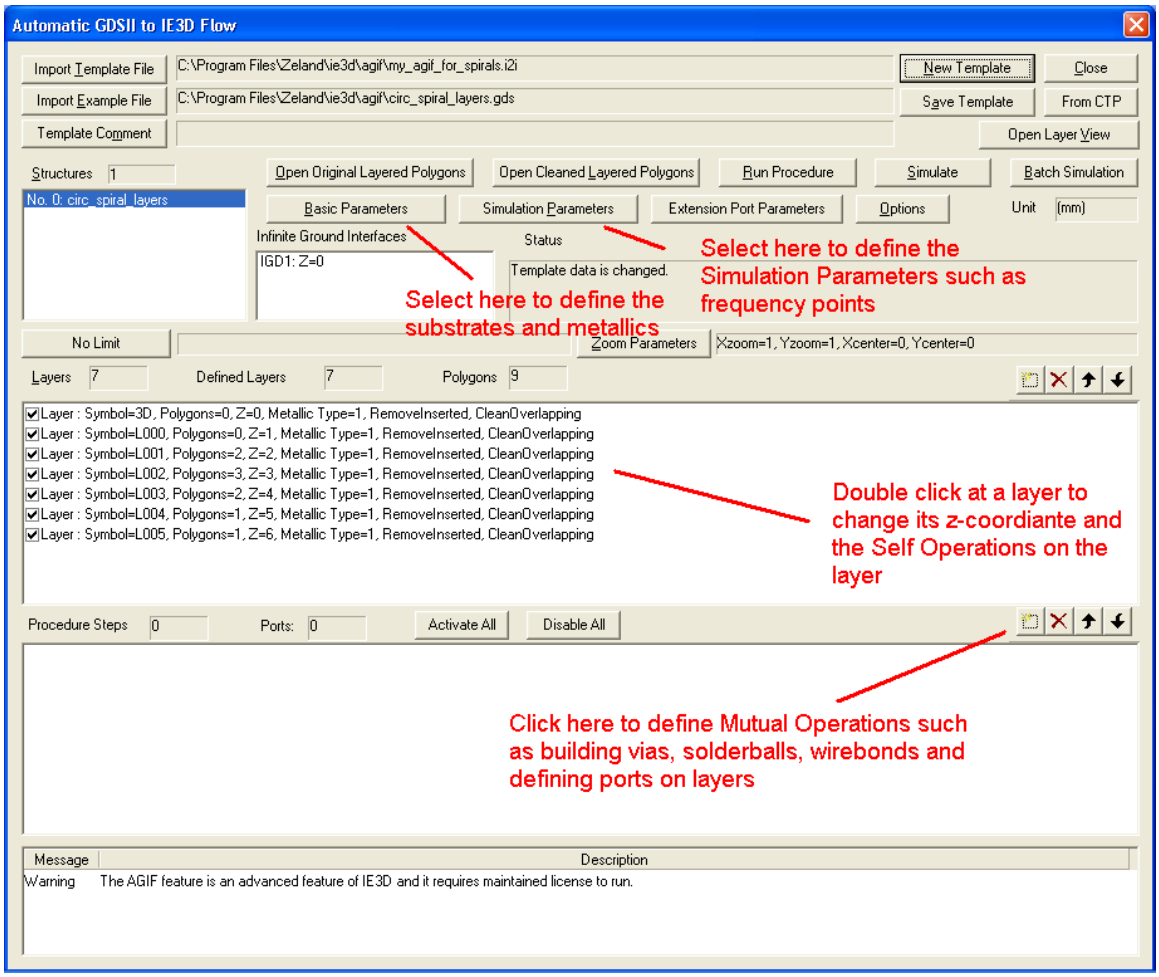


Figure BE.2 The AGIF dialog after the GDSII file is imported.

1. Build AGIF Template for the Four Spirals:

Please start AGIF by selecting File->Automatic Geometry to IE3D Flow in MGRID (or IE3D->AGIF in IE3D Program Manager). Please select Create and Edit AGIF Template for GDSII to IE3D Flow in AGIF 11.5 (or Cadence Virtuoso to IE3D Flow in earlier edition). Select OK. The Automatic GDSII to IE3D Flow dialog comes up. Our 1st step is to create an AGIF template for a class of GDSII files with the same layer and port configuration. If we want to simulate some GDSII files using some pre-defined template files, we can open the AGIF template file by selecting Import Template File button in the Automatic GDSII to IE3D Flow dialog to open an AGIF template file (.i2i). Then, please select Batch Simulations button to batch process your GDSII files using the template. For our example, we want to show you starting from building the template.

Select New Template button. AGIF will prompt you the New AGIF Template File dialog. Please enter a name "my_agif_for_spirals.i2i" in C:\MentorGraphics\<latest_release>IE3DIE3D\SDD_HOME\IE3D\ie3d\agif directory. AGIF will prompt you for the Import Example File. An import example file should be a .GDS file to test the AGIF procedure. For our 4-spirals, we can choose the circular spiral for example. Please select the file: C:\MentorGraphics\<latest_release>IE3D\SDD_HOME\IE3D\ie3d\agif\circ_spiral_layers.gds. The layers and polygons in the GDSII file are imported and the layers are listed in Figure BE.2.

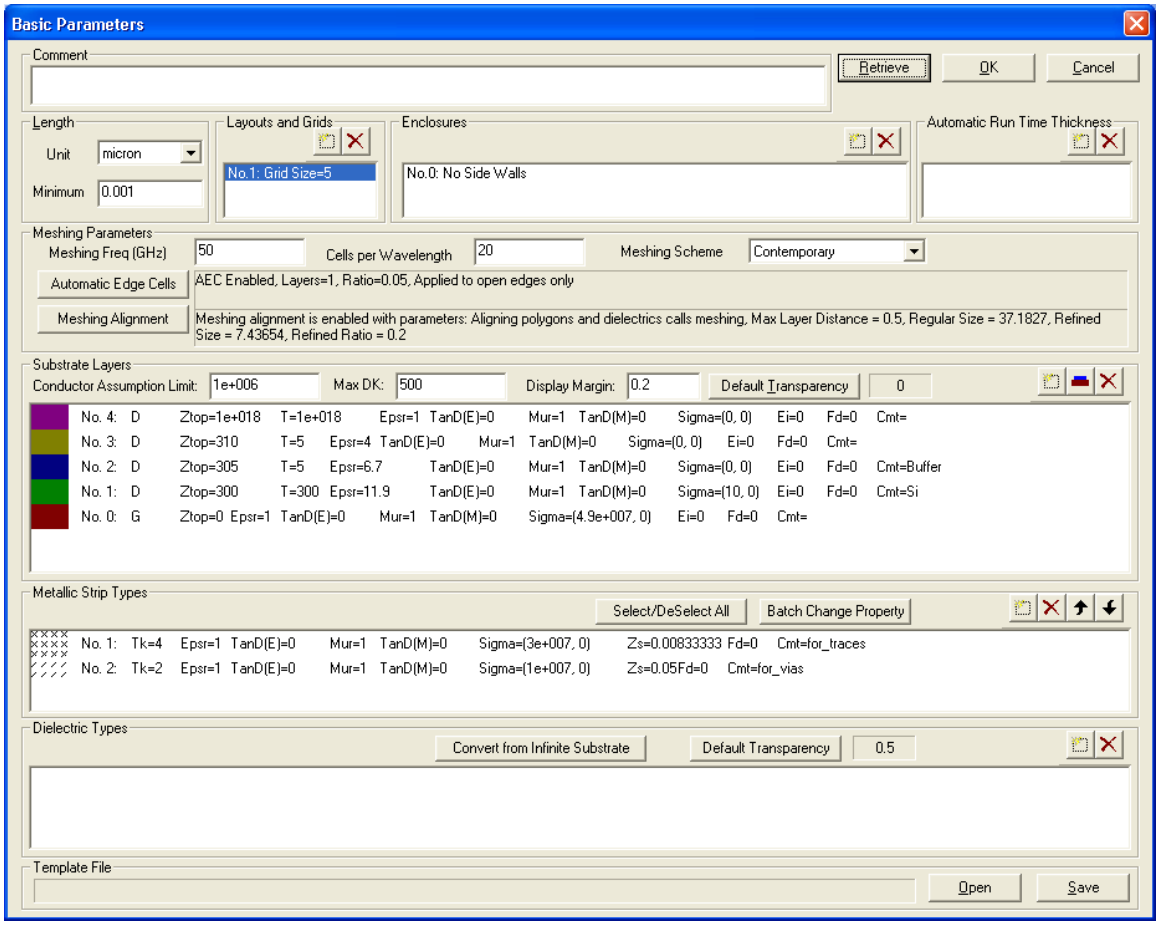


Figure BE.3 The retrieved substrate and metallic information.

Our next step is to define the substrates and metallic strip types. Select Basic Parameters in the dialog. It is the same Basic Parameters dialog as MGRID. We can define the substrates and metallic types here. In fact, we can just retrieve the information from an existing .geo file. Please select Retrieve button in Basic Parameters. Select the file: C:\MentorGraphics\<latest_release>IE3D\SDD_HOME\IE3D\ie3d\agif \circ_spirals_layers.geo file. The substrate and metallic information from the file are retrieved. From -∞ to 0 is the ground plane. From 0 to 300 microns is the semiconductor layers. From 300 to 305 is the layer with permittivity of 6.7. From 305 to 310 is a layer with permittivity of 4. There are two metallic types and the 1st one is for the traces and the 2nd type is for the vias. Please select OK to finish defining the Basic Parameters. AGIF may warn you on multiple thin substrate layers and ask you whether you want to merge the thin layers. Please select No so that AGIF will not merge the thin layers automatically for you.

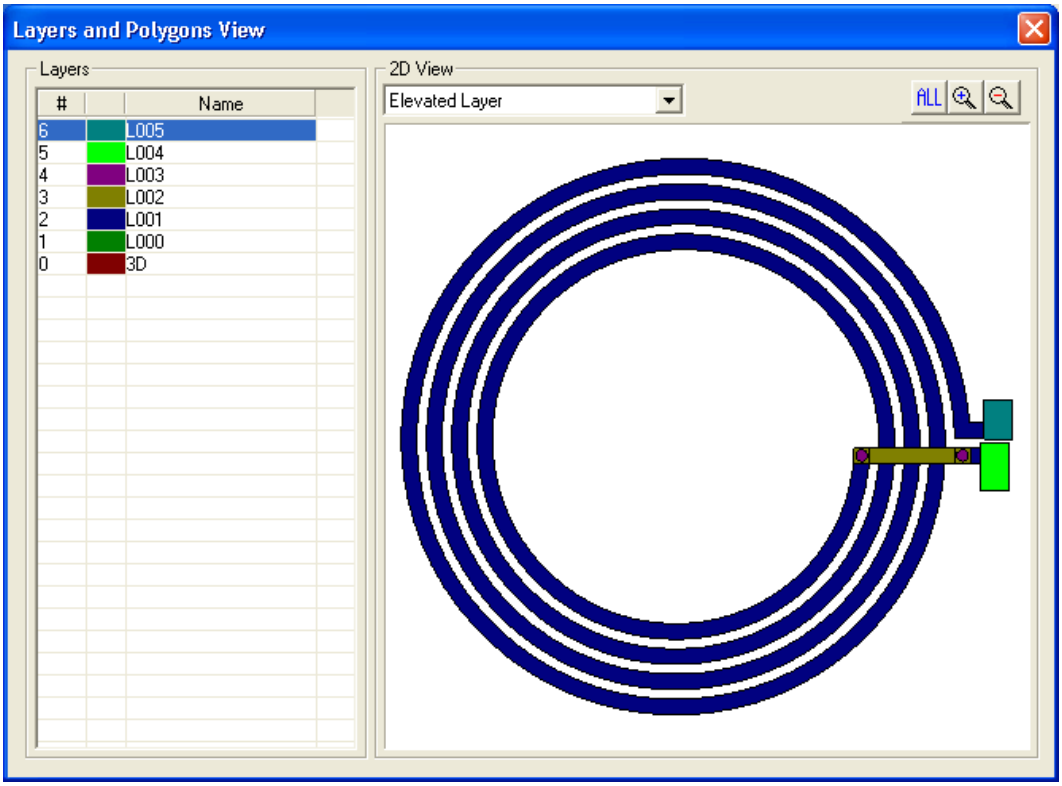


Figure BE.4 The imported layers and polygons from the GDSII file.

Our next step is to map the layered polygons to the right z-coordinates. Please select Open Layer View in the AGIF dialog (see Figure BE.4). We have 7 layers in the list box (3D, L000, L001, L002, L003, L004, L005, L006). The "3D" is not useful. There are not polygons in the L000. The polygons on each of the other layers are for some special purposes. We will map them to the appropriate layers and apply appropriate Self Operations on them. Double click the "L001" layer on the layer list. Enter its Z-Bottom as 305 or you can select click at the No.2 layer and AGIF will capture the "305" from it (see Figure BE.5). Please enable "Remove Redundant Vertices from Curvature" and change the Min Distance to 20 microns. This will allow IE3D/AGIF to remove some redundant vertices on the circular spiral to save computational resources in the simulations without losing much accuracy. Choose the No.1 metallic type for it. Please change the name of the layer from "L001" to "L_traces". Select OK to finish it. Please do the same thing to the other layers.

Edit "L002" and change it to "L_bridge" because the polygons will be used for the bridge. Change its Z-bottom to "310". Choose the No.1 metallic type for it.

Edit “L003” and change it to “L_for_vias” because the polygons on the layer will be used for the shapes of the vias. Please choose the No.2 metallic type. IE3D will be able to use derive a suitable metallic type from it for the vias for precise modeling of the vias. Define the Z-Bottom to 315. This z-coordinate is not critical because we will use the polygons to build vias and the polygons will not be actually put on z = 315. Please select “Square” for Convert in Shape. We want to convert the shapes into squares. Vias are normally small. We can just use square shapes for them to reduce computational resources without losing much accuracy.

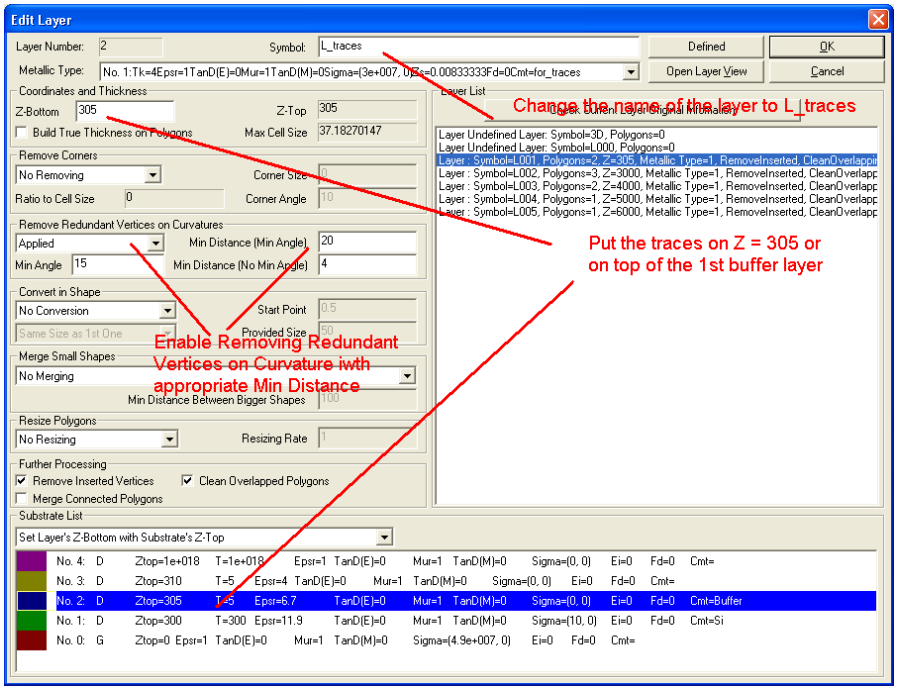


Figure BE.5 The Edit Layer dialog for L001 or L_traces.

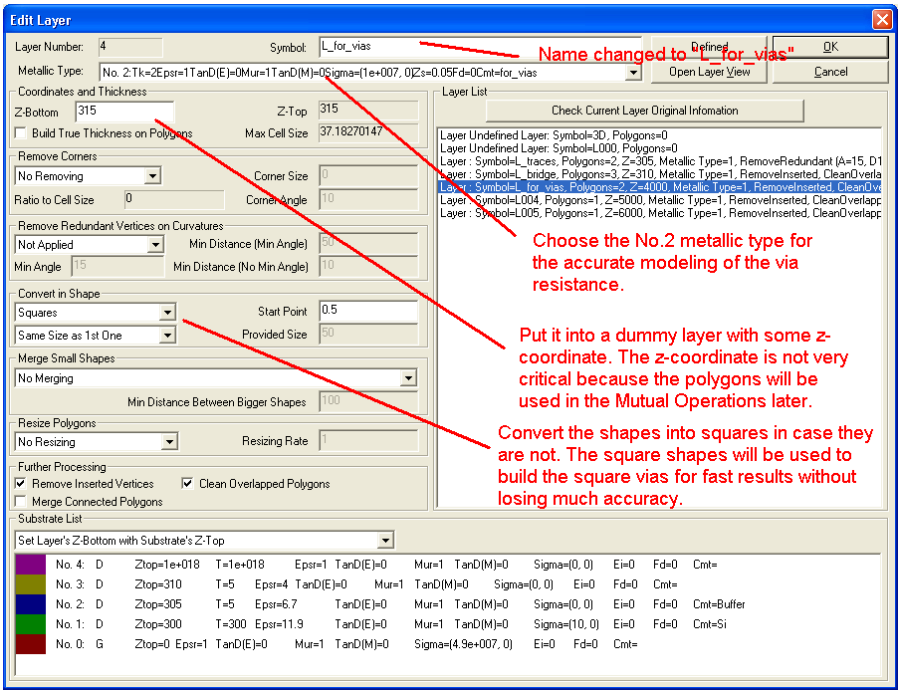


Figure BE.6 The Edit Layer dialog for L003 or L_for_vias.

Edit the layer "L004" and change it to "L_port_1". Basically, the layer should contain one polygon. It will be used for defining the port 1 later. Please put it at Z = 320. Its z-coordinate is not critical because the polygon will be used to define the port 1. It will not appear in the final IE3D layout.

Edit the layer "L005" and change it to "L_port_2". We will use the polygon on it to define the port 2. Set its z-coordinate to Z = 325.

After we define all the layers, we will have the layers shown in Figure BE.7. Our next step is to define the Layer Mutual Operations. Basically, we will define the vias and ports.

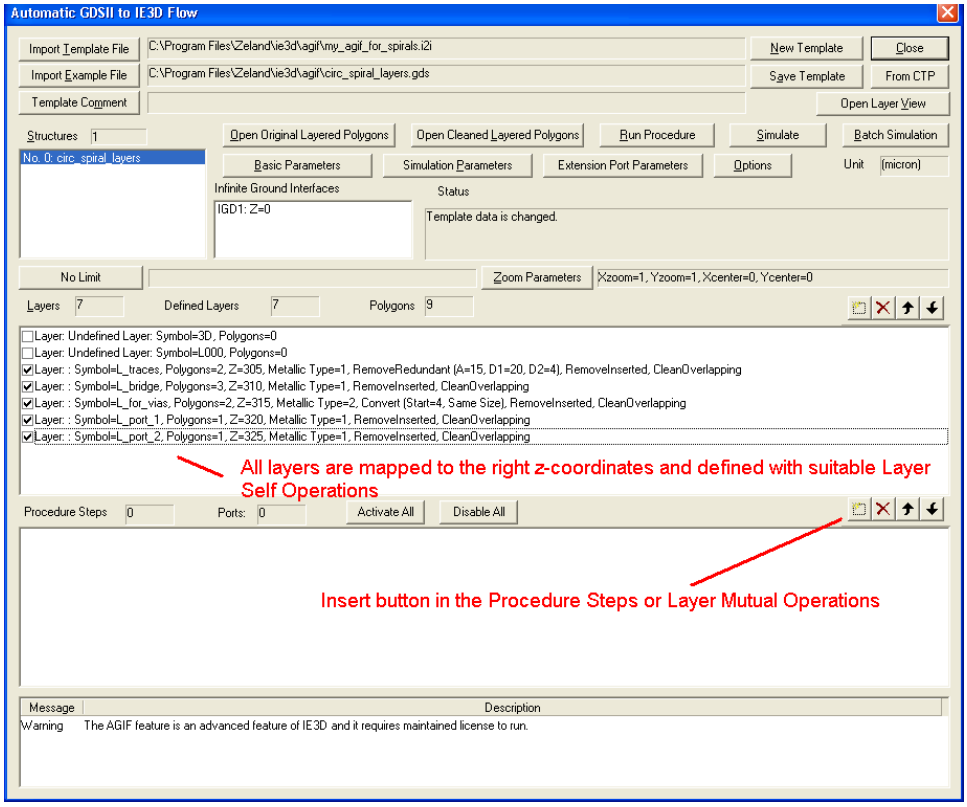


Figure BE.7 The AGIF dialog after defining the Layer Self Operations.

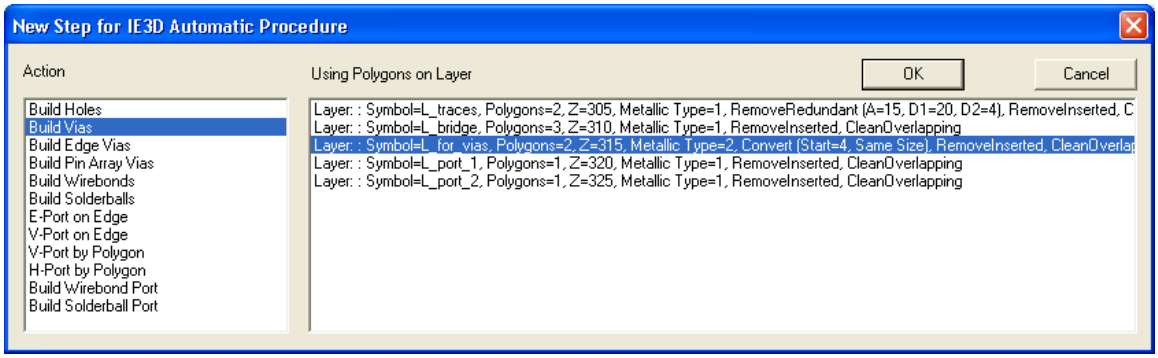


Figure BE.8 The dialog for selecting the shape to build vias.

Select the Insert button in the Procedure Steps (or Layer Mutual Operations). Please select "Build Vias" for Action and "Layer: Symbol=L_for_vias…" for Using Polygons on Layer (see Figure BE.8). Select OK. AGIF will prompt you to select the Passive Layers for building vias. Please select "Layer: Symbol=L_traces…" and "Layer: Symbol=L_bridge…" in the list (see Figure BE.9). Basically, we want to build use the shapes in L_for_vias to build vias between L_traces and L_bridge. Select OK to finish defining the Layer Mutual Operation.

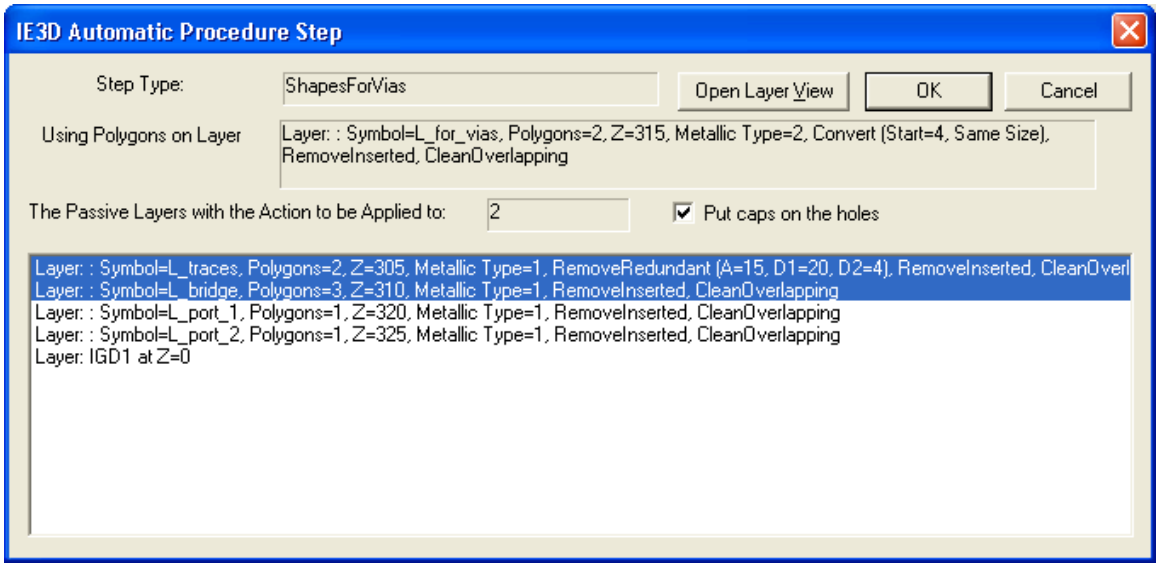


Figure BE.9. The dialog for defining the passive layers for vias.

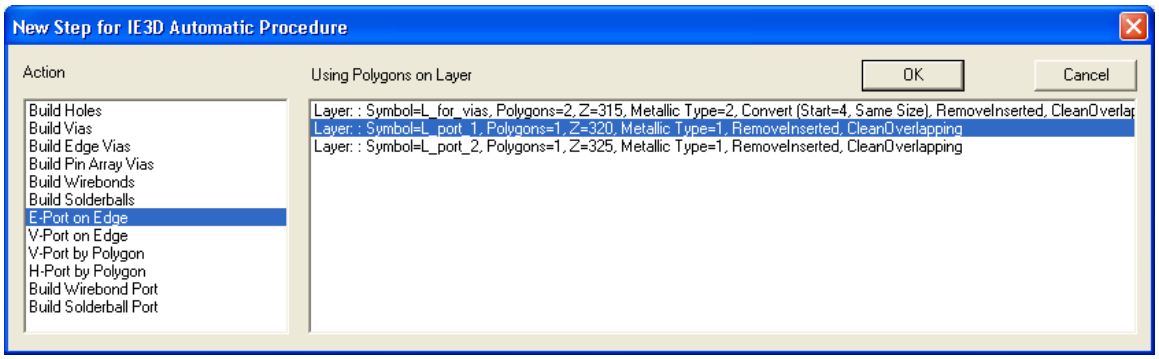


Figure BE.10. The dialog for defining a port.

Select Insert button again. Select "E-Port for Edges" for Action. Select "Layer: Symbol=L_port_1…" for Using Polygon on Layer (see Figure BE.11). We are going to use the only polygon on "L_port_1" to enclose some edge(s) on another layer to define an extension port on it. Select OK. AGIF will prompt you to choose the layer where you want the port to be on. Please select "Layer: Symbol=L_traces…" because we want to define port 1 on the layer. Please choose Advanced Extension port. For the Port Index, we should choose New Port. In case it is the negative port of a differential port, we can choose other option. Please select OK to finish defining the port 1.

Please select Insert again. Select "E-Port for Edges" for Action. Select "Layer: Symbol=L_port_2…" for Using Polygon on Layer. Select OK for the next dialog. Please select "Layer: Symbol=L_traces…" because we want to define port 2 on the layer. We should have a similar dialog

shown in Figure BE.11. Select OK to continue. We will get the AGIF dialog shown in Figure BE.12. You can select Save Template button to save the results. You may be interested in how the IE3D model looks like after the template is applied to the GDSII file. Please select Run Procedure. AGIF will apply the template to the GDSII file. It will finish converting the GDSII file into an IE3D geometry and prompt you. Please select Continue and an MGRID is invoked to show you the converted geometry.

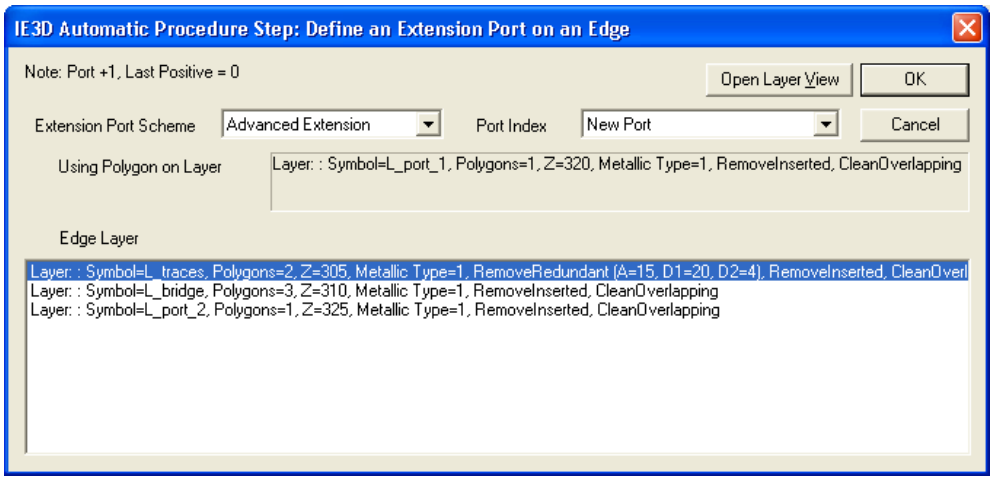


Figure BE.11 The 2nd dialog in defining an extension port.

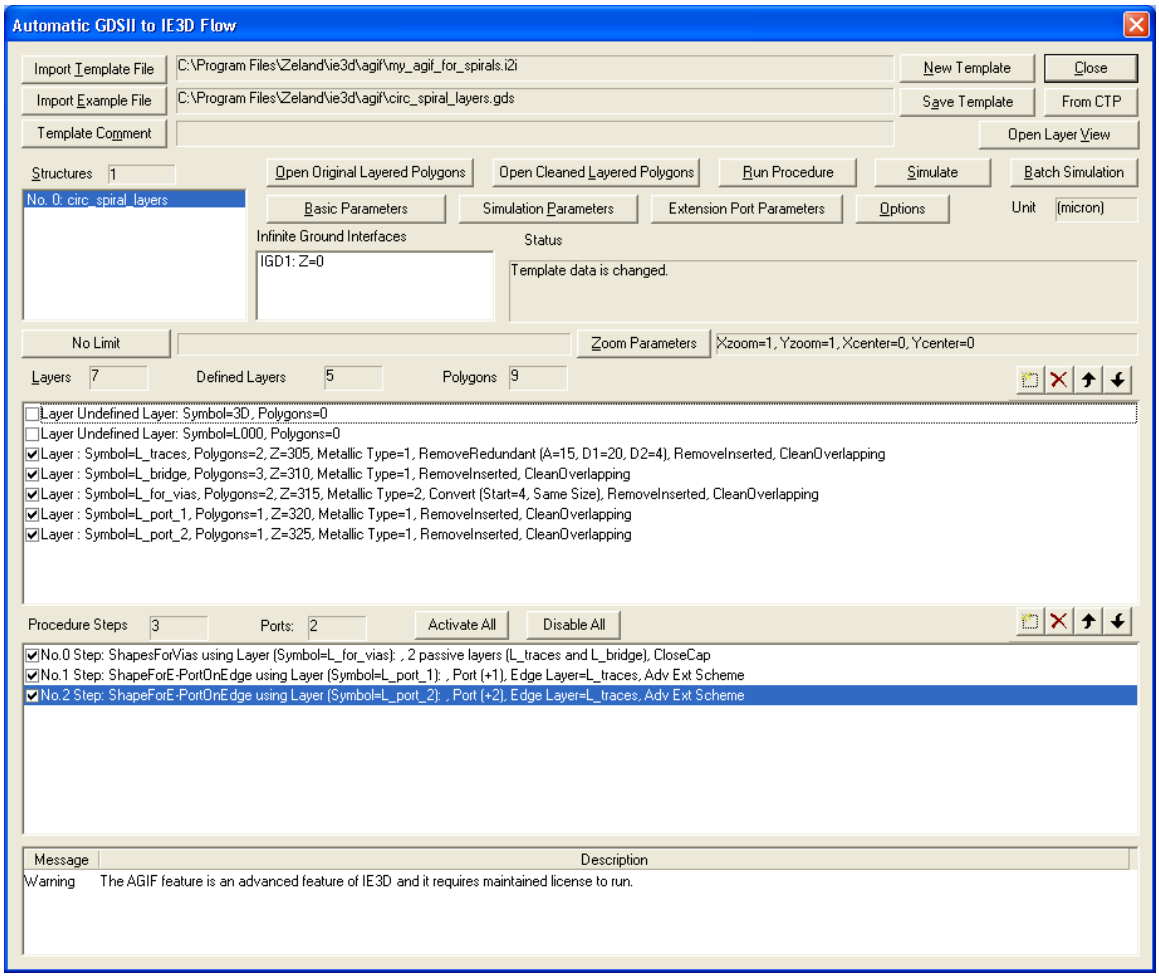


Figure BE.12 The AGIF dialog with all Layer Self Operations and Layer Mutual Operations defined.

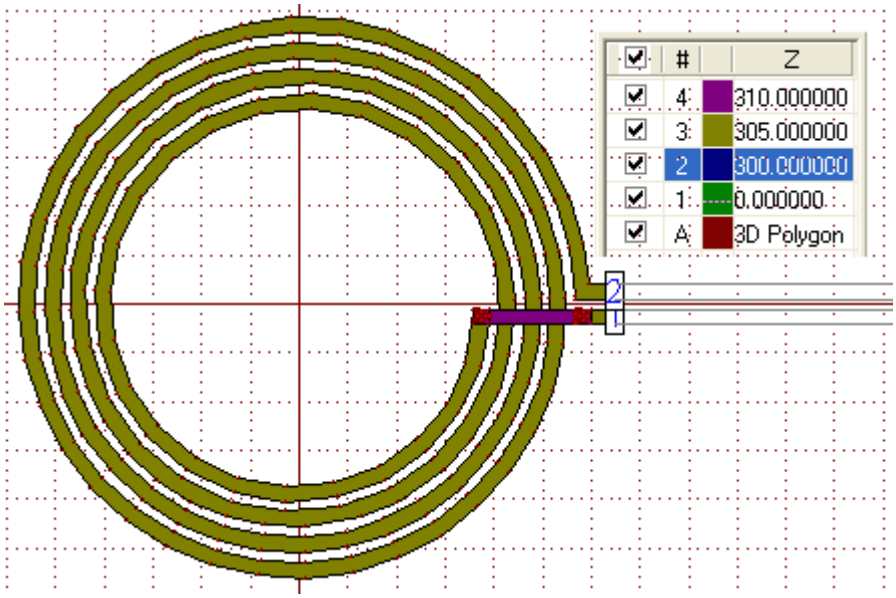


Figure BE.13 The converted circular spiral on IE3D/MGRID.

The created IE3D geometry file is ready for simulation. We can select Process->Simulate in MGRID to simulate the structure. In fact, we don't need MGRID to do the simulation setup. We can't do it directly from AGIF by selecting Simulate button. Before we do so, we should select Simulation Parameters button in AGIF dialog. The Simulation Setup dialog comes up. Please define 100 frequency points from 0.1 to 10 GHz. Select OK to close the dialog. Please select Simulate and select OK button. AGIF will first convert the GDSII file into IE3D geometry and then it will go directly into simulation.

You may consider that the Simulate button in AGIF just does the same thing as the one in MGRID. It still requires you to define the frequency points etc. It does not save you time. The advantage of AGIF is in handling a batch simulation. Please select the Batch Simulation button in AGIF. The AGIF Batch Simulation dialog comes up. Please select the Insert button and select the four GDSII files: circ_spiral_layers.gds, oct_spiral_layers.gds, rect_spiral_layers.gds and square_spiral_layers.gds in the C:\MentorGraphics\<latest_release>IE3D\SDD_HOME\IE3D\ie3d\agif directory. You can select them all by holding them the "shift" key when you are clicking at a file. After you select OK in the selecting file dialog, AGIF will insert all the files into the list in the AGIF Batch Simulation dialog (see Figure BE.14). We can do much in this dialog. You can select Run All. AGIF will batch convert the GDSII files into IE3D geometry files and run all the structures one by one. It may take 15 to 30 minutes to finish the 4 simulations. After the simulations, you can select all the files in the list and select the Display Geometry and Display Results button to see how the created IE3D models look like and how the results look like. Basically, they are the 4 spiral inductors shown in Figure BE.1. Please select OK to get back to AGIF dialog. Please select Save Template to save all the data including the batch simulation list into the template. You can use the template for other GDSII files with the same process as long as their layer configurations are the same.

We have demonstrated a simple example in using AGIF to simplify the process of building model. In case you make a mistake, AGIF normally can detect it for you and tell you which step the mistake happens. You can try to revise the step. Then, you can re-run the procedure without wasting the previous effort. In case you want to change your steps, you can also modify it from the saved AGIF template file.

For example, if you want to add thickness model for the trace, you just need to change the settings in L_traces layer and/or the L_bridge layer and save the template into a new file.

In summary, you can do much on IE3D/AGIF and it really helps you to reduce the modeling time and avoid mistakes. It is a great feature of IE3D. AGIF is available on the IE3D MM08X or the Special Flow Edition. You will need updated IE3D MM08X licenses to use it.

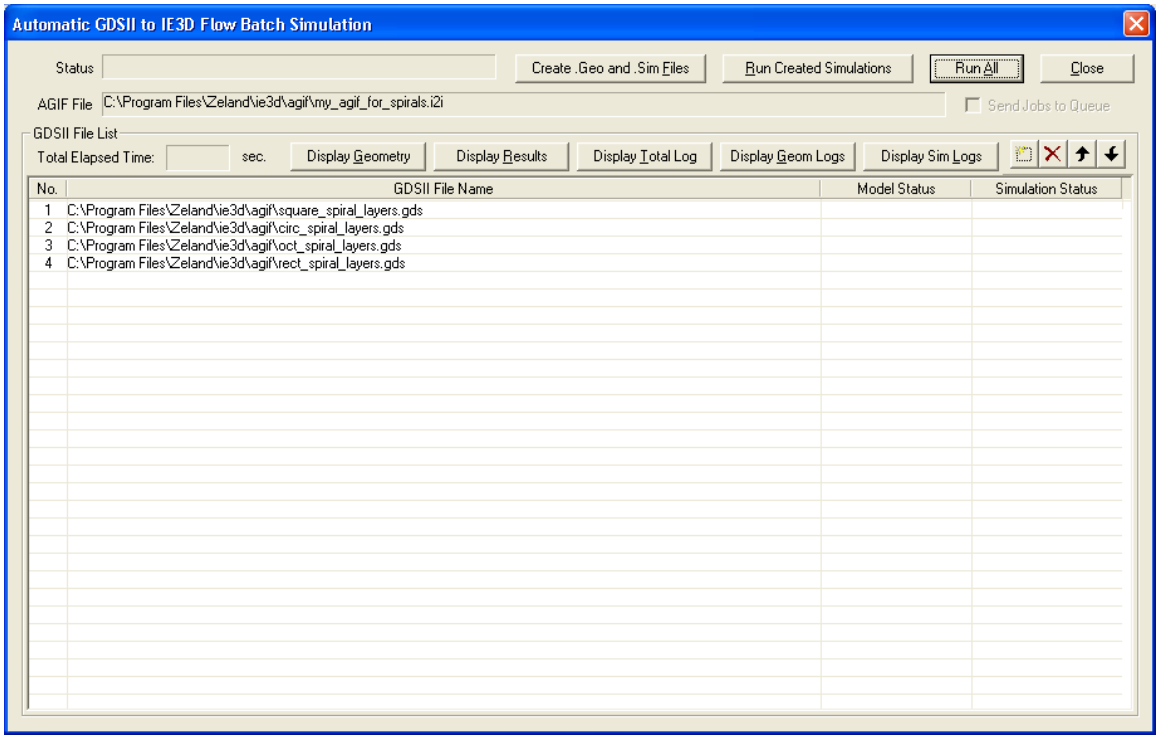


Figure BE.14. The AGIF Batch Simulation dialog after we inserted the 4 GDSII files.

Appendix BF Windows 32/64, Linux 32/64 and Multi-CPU Supports

The full-package of IE3D is certainly running on Windows 32. The addressing capability of 32-bit computing is limited. A 32-bit OS can only access maximum 4 GB of RAM. Among the 4 GB, 2 GB addressing is for the program and the other 2 GB addressing is for the data on Windows 32. The maximum amount of memory allocated for the users for all running programs including the OS is 2 GB. When you are running a large IE3D simulation, you are suggested to close as many applications on your computer as possible to free some available data to IE3D. Even you may have 2 GB or more RAM installed, IE3D may not be able to grant so much memory when its requirement is getting close to 2 GB. It may fail at about 1.5 GB, 1.2 GB or even lower. Whether the memory is available is not controlled by IE3D but by the OS. When such thing happen, it may issue an error message "dynamic memory allocation failed!" or similar or even crash IE3D.

Normally, the programs or executables are not big. It uses far less than 2 GB memory. For this reason, if your computer is Win32 or Linux 32, you really should not install more than 2 GB because the additional RAM does not improve the capability of your computer anyway.

To expand the capability of IE3D, we have implemented the IE3D engine onto Windows 64 and Linux 64. Theoretically, 64-bit computing is supposed to be able to access 4 G by 4 G RAM. It is huge! Actual implementation may have some limitation. However, its accessing capability should be much higher than 4 GB.

All IE3D source codes are compatible with both Windows 32 and Windows 64. However, for consideration of easy management and installations, we only release the IE3D engine on Windows 64. Actually, graphic interfaces need much less than 2 GB RAM. It normally doesn't need to be 64-bit. The Windows 64 GUI will be released later after we separate the installations between Windows 32 and Windows 64. At this time, we have created the Windows installation into one single package for Windows 32 and Windows 64. The installation automatically detects your OS type and installs the IE3D engine for either Win32 or Win64 for you. The name of the IE3D engine is: ie3dos.exe. The IE3D engine is located at: .\SDD_HOME\IE3D\exe.

IE3D has three ways to create layouts for simulations: (1) MGRID: Polygon based geometry editing; (2) IE3DLibrary: Object-oriented geometry editing with parameterization. (3) AGIF: Automatic layout generation from GDSII, Cadence Virtuoso, Allegro and APD. MGRID and IE3DLibrary are on Windows only. AGIF is available on both Windows and Linux. When AGIF is running on Linux, it can automatically create IE3D layout and run the IE3D engine for Linux transparently. You are able to visualize s-parameters and current distribution on AGIF. AGIF on Linux is available to IE3D MM08X users only.

Two more important pieces of software on both Windows and Linux are the ZDM (IE3D Distributed Agent) and ZDS (IE3D Distributed Server). The process of distributed computing is described in the following:

ZDS is running as a server and it is managing the distributed jobs. A user running MGRID (or IE3DLibrary, AGIF or Cadence Virtuoso accessing AGIF) is able to submit a distributed job to ZDS. When MGRID is sending out a distributed job, it is sent to ZDS and managed locally by JobsManager. JobsManger is the client sending jobs to ZDS. Then, ZDS dispatches the jobs to different ZDM agents to perform IE3D simulations.

Basically, there are three parts involved in a distributed EM simulation: (1) Client (or MGRID); (2) Server (ZDS); (3) Agents (ZDM and IE3D engine).

If you have implemented the ZDS/ZDM distributed computing, you can run IE3D simulations on Linux while the GUI can be on Windows. Certainly, you can have everything on Linux if you are doing your simulations from AGIF..

You may be curious about the speed comparison on IE3D between different OSs. From our experience, IE3D engines on Linux are faster than the engines on Windows. The 64-bit engines are faster than 32-bit engines.

Another interesting topic is multi-CPU and multi-core support. IE3D engine is supporting multiple CPUs on major procedures.

Appendix BG S-Parameter Visualization and Processing on MGRID

S-parameter visualization and processing used to be performed on MODUA or MDSPICE. Starting from IE3D 14, we have integrated the s-parameters processing into the IE3D EM Design System or MGRID. MGRID is able to do layout, import, simulation and optimization, visualization and processing of s-parameters, current distribution, radiation pattern and near field. This appendix will focus on the s-parameters processing capabilities. For visualization, you just need to add plots for different parameters.

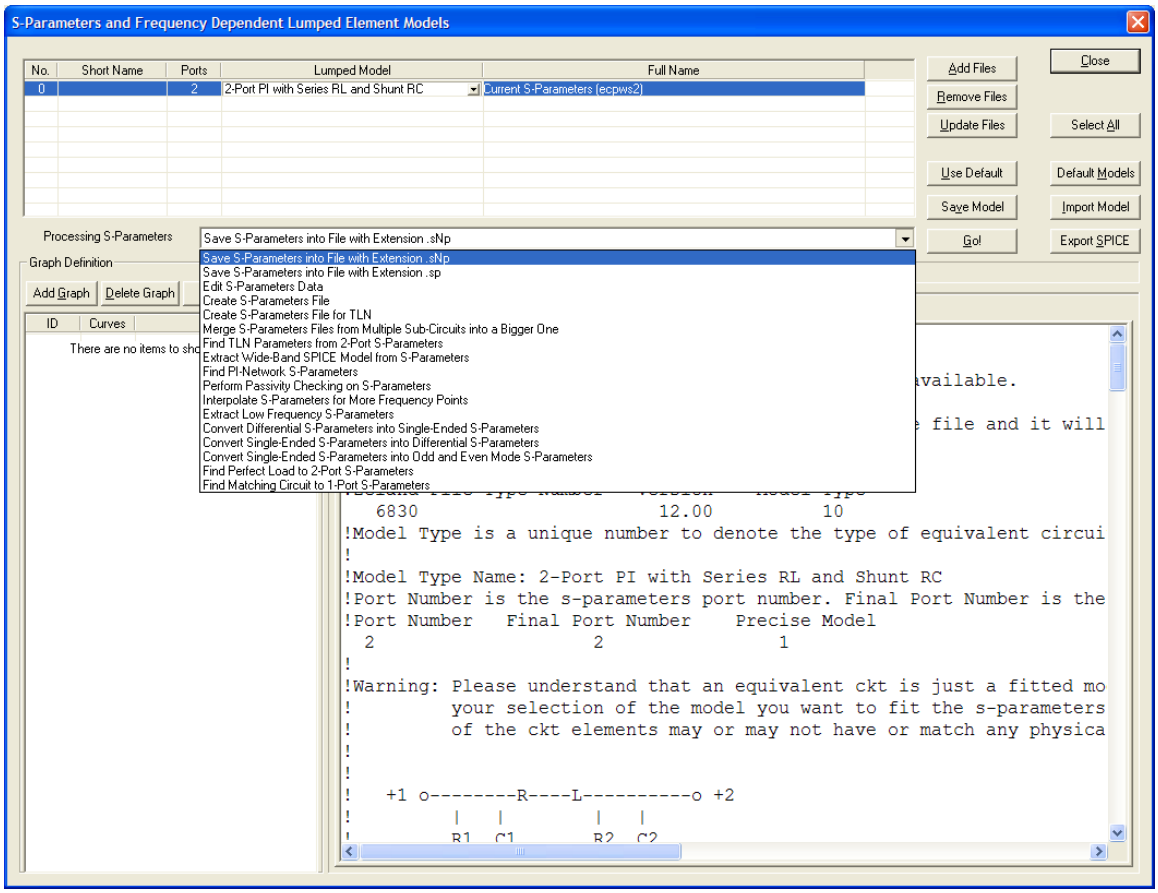


Figure BG.1 The s-parameters visualization and processing dialog.

Take the file: .\ie3d\samples\ecpws2.geo as an example. We can simulate this 2-port CPW line structure. It takes seconds to simulate it from 0.1 to 20 GHz for 200 frequency points. After simulation, MGRID should bring up the S-Parameters and Frequency Dependent Lumped Element Models dialog (see Figure BG.1). It corresponds to Process->S-Parameters and Lumped Equivalent Circuit command or the Window->S-Parameters Display->Define Plots command. There is a list of Processing S-Parameters commands there (see Figure BG.1). You can select a command from the list and select Go to process the s-parameters. Table BG.1 give a brief explanation on what the commands are. We will not be able to provide one example for each functionality here.

Select "Find TLN Parameters from 2-Port S-Parameters" and select "Go". MGRID will prompt you the Transmission Line Information dialog. You are required to define the Length Unit you want and specify the physical length. If the selected s-parameter file is internal s-parameters belonging to the current geometry file, MGRID will automatically detect the length unit and the length for you. For external s-parameters data, you will need to enter the length because the information is not carried in the s-parameters file. Even for the internal s-parameters, you should also double check the detected values.

Select OK. MGRID will show you the extracted TLN parameters. You have some options to show different parameters. You can select OK and MGRID will save the TLN data into an ASCII file. We will not discuss more s-parameter processing features. Please try to explore them.

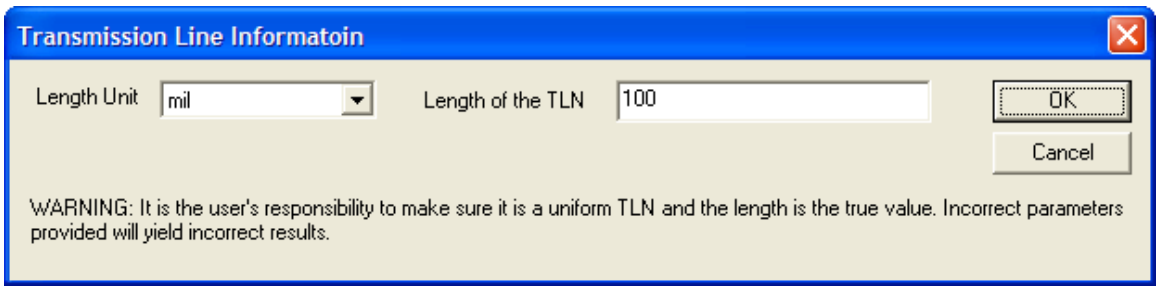
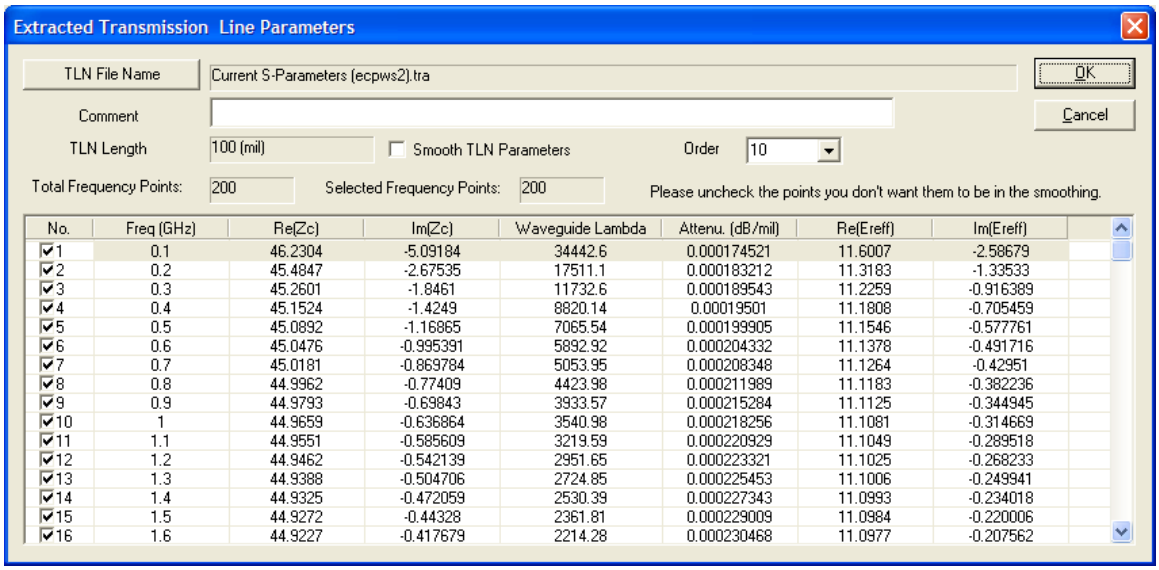


Figure 2 The Transmission Line Information dialog.



No.	Freq (GHz)	Re(Zc)	Im(Zc)	Waveguide Lambda	Attenu. (dB/mil)	Re(Ereff)	Im(Ereff)
1	0.1	46.2304	-5.09184	34442.6	0.000174521	11.6007	-2.58679
2	0.2	45.4847	-2.67535	17511.1	0.000183212	11.3183	-1.33533
3	0.3	45.2601	-1.8461	11732.6	0.000189543	11.2259	-0.916389
4	0.4	45.1524	-1.4249	8820.14	0.00019501	11.1808	-0.705459
5	0.5	45.0892	-1.16865	7065.54	0.000199905	11.1546	-0.577761
6	0.6	45.0476	-0.995391	5892.92	0.000204332	11.1378	-0.491716
7	0.7	45.0181	-0.869784	5053.95	0.000208348	11.1264	-0.42951
8	0.8	44.9962	-0.77409	4423.98	0.000211989	11.1183	-0.382236
9	0.9	44.9793	-0.69843	3933.57	0.000215284	11.1125	-0.344945
10	1	44.9659	-0.636864	3540.98	0.000218256	11.1081	-0.314669
11	1.1	44.9551	-0.585609	3219.59	0.000220929	11.1049	-0.289518
12	1.2	44.9462	-0.542139	2951.65	0.000223321	11.1025	-0.268233
13	1.3	44.9388	-0.504706	2724.85	0.000225453	11.1006	-0.249941
14	1.4	44.9325	-0.472059	2530.39	0.000227343	11.0993	-0.234018
15	1.5	44.9272	-0.44328	2361.81	0.000229009	11.0984	-0.220006
16	1.6	44.9227	-0.417679	2214.28	0.000230468	11.0977	-0.207562

Figure 3 The extracted TLN parameters.

Table BG.1 The implemented s-parameters processing commands:

Command	Description
Save S-Parameters into File with Extension .sNp	By default, the Agilent/EEsof Touchstone format is using file extension .sNp. For 1-port, it is .s1p. For 2-ports, it is .s2p, etc. No port information is stored inside the file. On IE3D, we save the s-parameters into Touchstone compatible format. The file extension is .sp (or .spt, .spm for some other purposes) for all port numbers. The port number is saved into the .sp file as a comment. Starting from IE3D 12, we have saved the s-parameters as an object in the geometry file. You can export the data into a .sNp file with the port number embedded into the file extension.
Save S-Parameters into File With Extension .sp	You can also export the s-parameters data in the geometry file into a file with extension .sp. The file can be used on MODUA or other application programs accepting Touchstone compatible files.
Edit S-Parameters Data	You can edit the s-parameters data as you want.
Create S-Parameters Data	You can manually create an s-parameters data.
Create S-Parameters File for	You can find the s-parameters of a section of transmission line with

TLN	specified transmission line parameters.
Merge S-Parameters Files from Multiple Sub-Circuits into a Bigger One	The feature allows you to merge the s-parameters of multiple sub-circuits into one s-parameters object of a bigger circuit.
Find TLN Parameters from 2-Port S-Parameters	The feature allows you to find the TLN parameters from 2-port s-parameters. You are required to provide the physical length of the TLN even though it might be able to detect it. Also, you may want to make sure the length of the line should not exceed half a wavelength. Otherwise, the accuracy of the solved TLN parameters may not be guaranteed. The reason is that there is no unique solution at the frequency points with the length as integral of half wavelength.
Extract Wide-Band SPICE Model from S-Parameters	It allows you to find the RLC equivalent circuit matching the frequency response of the s-parameters over a wide frequency range. However, the RLC elements are not guaranteed to be passive.
Find Pi-Network S-Parameters	This feature allows you to find the 1-port s-parameters associated with the PI-network when you break an N-port s-parameters into the 1-port s-parameters forming a PI-shape. For example, for a 2-port s-parameters data, you can cast it into the Pi-network as: o----------[sps]----------o | | [spp1] [spp2] | | [gnd] [gnd] where sps, spp1 and spp2 are 1-port s-parameters data blocks.
Perform Passivity Checking on S-Parameters	It allows you to check whether the s-parameters are passive. You have the option to enforce passivity.
Interpolate S-Parameters for More Frequency Points	You can perform interpolation on the existing s-parameters for dense frequency points. The accuracy may not be guaranteed at those interpolated points.
Extract Low Frequency S-Parameters	It allows you to perform extrapolation on the existing frequency points for the low frequency response. The accuracy may not be guaranteed in the extrapolated frequency points.
Convert Differential S-Parameters into Single-Ended S-Parameters	It allows you to divide one differential port into two single ended ports. You can divide multiple differential ports into multiple pairs of single ended ports simultaneously. However, you can't use the two single ended ports freely. You need to make sure you connect the two terminals of a differential port of an element across the pair of two single ended ports derived from one single differential port when you are simulating a larger circuit by connecting the sub-circuits together.
Convert Single-Ended S-Parameters into Differential S-Parameters	It allows you to find differential s-parameters from single ended s-parameters. One differential port will correspond to two single ended ports. The results have fewer ports.
Convert Single-Ended S-Parameters into Odd and Even Mode S-Parameters	It allows you to find the s-parameters based upon odd and even modes from the single ended s-parameters. The results have the same number of ports because two single ended ports correspond to a pair of odd mode and even mode ports.
Find Perfect Load to 2-Port S-Parameters	Assume you want to connect a 1-port load to the port 2 of a 2-port s-parameters data. This feature allows you to find the 1-port s-parameters of the load making no return loss at the port 1 of the 2-port s-parameters.
Find Matching Circuit to 1-Port S-Parameters	This feature allows you to find a simple LC circuit with perfect match at specified frequency.

Appendix BH Pattern Visualization and Processing on MGRID

Traditionally, pattern visualization and process are on PATTERNVIEW. Starting from IE3D 14, we have integrated all functionalities of PATTERNVIEW into MGRID. You are able to visualize and post-processing pattern data on MGRID completely. This appendix explains the functionalities and basic procedures.

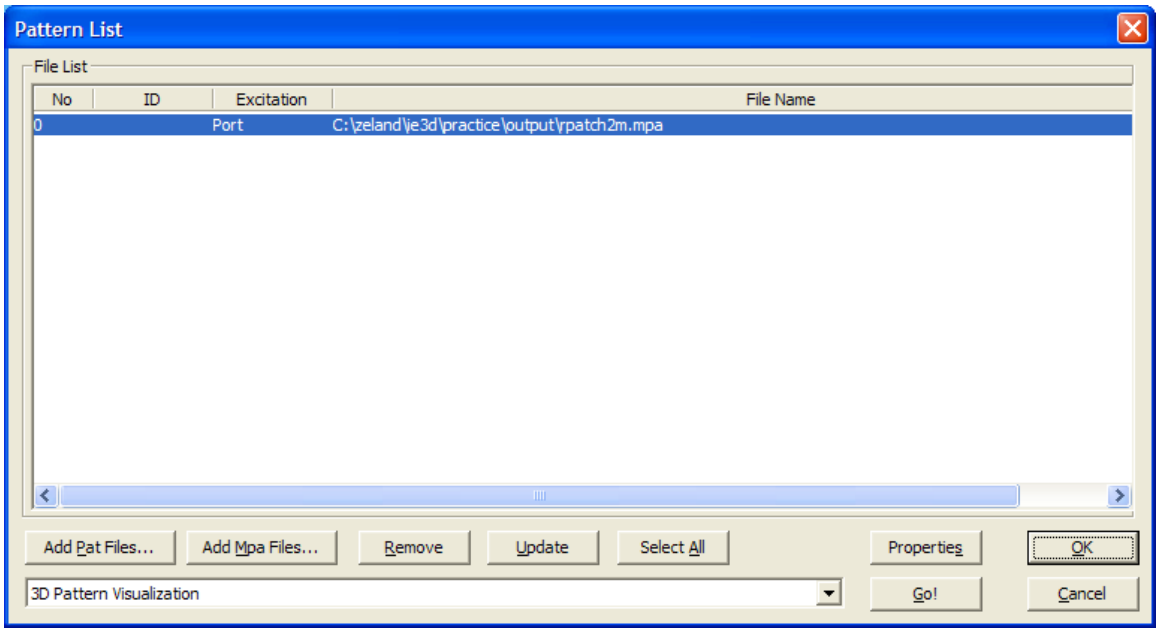


Figure BH.1 The Pattern List dialog for processing of pattern data.

Select Window->Radiation Pattern Properties, MGRID will show you the Pattern List dialog (see Figure BH.1). There are many things you can do.

If you have simulated the current structure with saved pattern data, the general pattern file will be added into the list by default. You can select "Add Pat Files" button to add more specific excitation pattern files into the list or select "Add Mpa Files" button to add more general pattern files into the list. A general radiation pattern file (.mpa) contains all the possibility of the excitations with the default specific excitations. It can be used to generate the specific excitation pattern file (.pat) for any excitations and terminations. Certainly, it is not necessary to generate a specific excitation pattern file for the excitations and terminations; you can always select the Properties button in the Pattern List dialog or double click at the .mpa file in the list to edit it for the desired excitations and terminations.

You can select Remove button to remove the files from the list. You can select Update to update the pattern data from files.

Select a specific pattern file (.pat) in the list and select Properties button will show you the pattern properties of the .pat file. If you select an .mpa file and select Properties button, it will allow you to edit the excitations and terminations of the general pattern (see Figure BH.2). For example, you can change the source impedance Zs, the normalization impedance Zc, the excitation magnitude and phase of a port at different frequency points. You can also perform Antenna Correlation calculation by selecting the Correlation Calculation button. More information can be found from Appendix BC.

There are some more processing functionalities at the bottom of the dialog. You can select an item and select "Go" button for it. The functionalities are listed in Table BH.1.

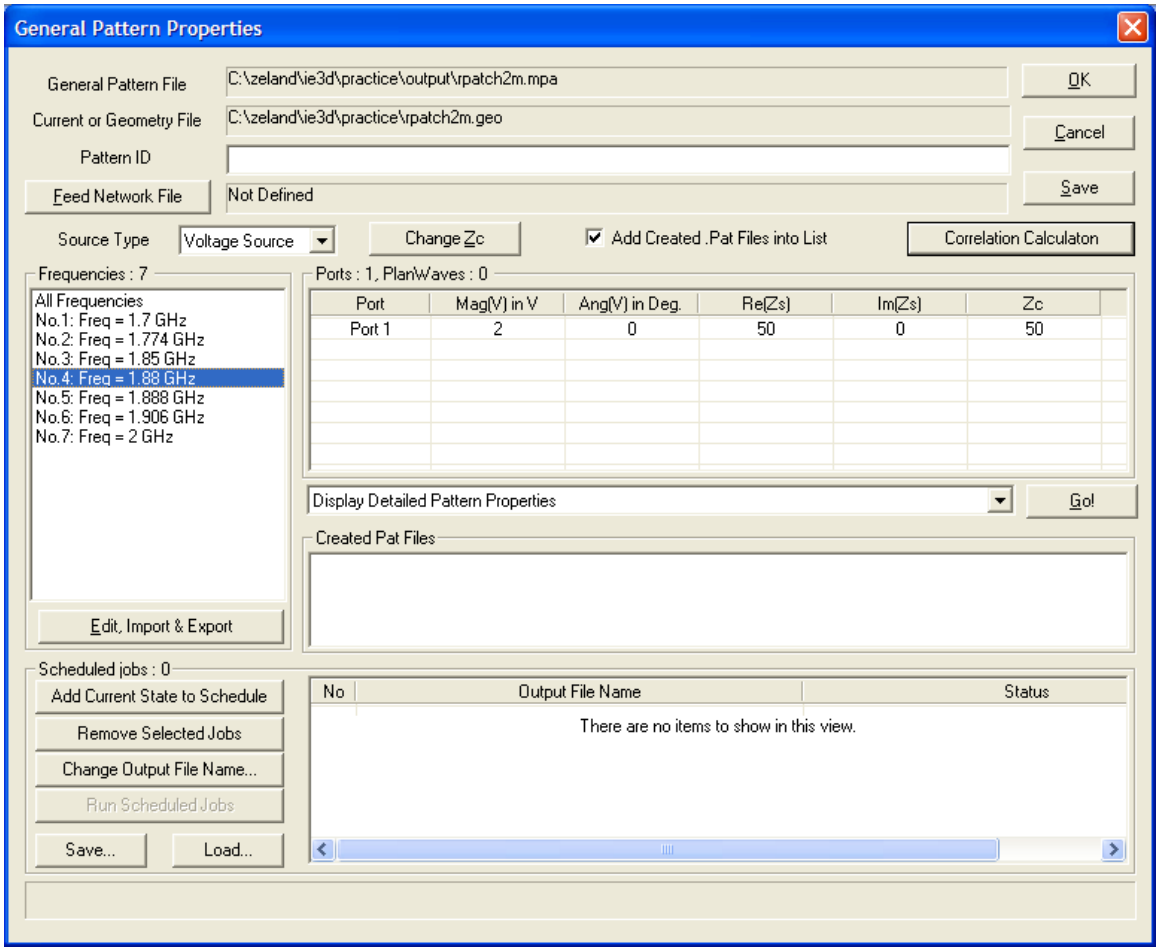


Figure BH.2 The Properties dialog of an .mpa file.

Table BH.1

Function	Description
3D Pattern Visualization	It allows you to display the "True 3D" radiation pattern and the "Mapped 3D" radiation pattern of a pattern file in the list at a selected frequency. The difference between "True 3D" and "Mapped 3D" patterns is shown in Figure BH.3.
2D Pattern Visualization	It allows you to display the patterns as curves in Cartesian coordinates. The selections of the curves can be from different files at different frequency points so that you can compare the patterns.
Find Tx and Rx Transfer Function	It allows you to find the transfer function between a Tx antenna and an Rx antenna based upon the radiation patterns of the 2 antennas. The Tx antenna is required to have one-port. The Rx antenna is required to have one-port and plane-wave excitations at the angles pointing to the Tx antenna. More explanation can be found from Appendix AJ.
Merge Pattern Files in the List into One	It allows you to merge the pattern of multiple files assuming they are located at where they are in the geometry files and assuming the coupling between them do not affect the current distribution and the pattern.
Perform Near Field to Far Field Transformation	It allows you to find the radiation pattern from some near field distribution. More explanation can be found from Appendix AK.

Perform Pattern Manipulation and Wave Propagation Calculation	It allows you to rotate the pattern by some angles and/or shift the pattern by some distance. It also allows you to find the field distribution at some location based upon the radiation pattern.
Find Array Radiation Pattern	It allows you to find the radiation pattern of an array. The array elements can be any one in the Pattern List.

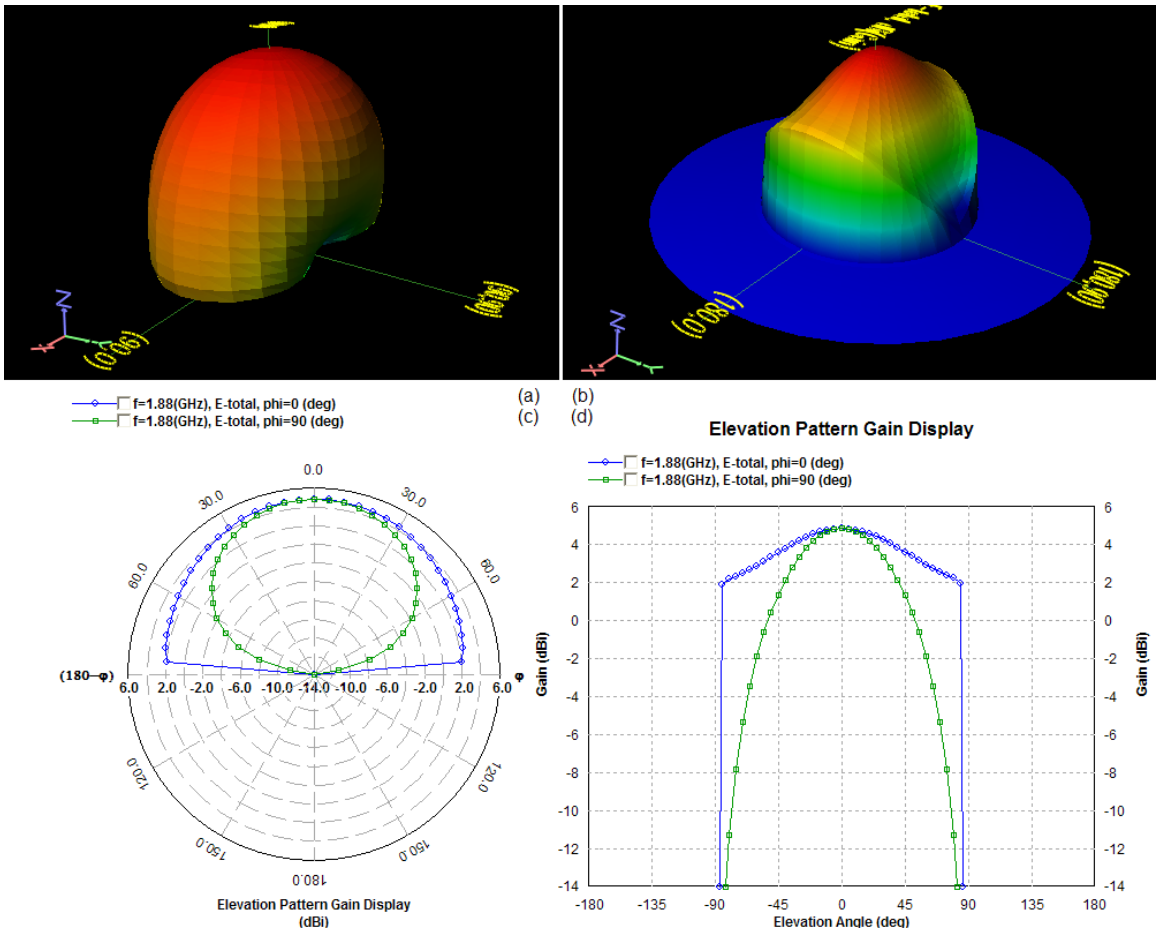


Figure BH.3 (a) True 3D Radiation Pattern; (b) Mapped 3D Radiation Pattern; (c) 2D Radiation Pattern (Polar Plot); (d) 2D Radiation Pattern (Cartesian Plot).

You can display the 3D pattern as "True 3D" pattern and "Mapped 3D" pattern. You can also display the 2D pattern in "Polar Plot" and "Cartesian Plot". The "Polar Plot" is a cut on the "True 3D" pattern while the "Cartesian Plot" is a cut on the "Mapped 3D" pattern.

The "theta" and "phi" in the "True 3D" pattern correspond to the same "theta" and "phi" in the antenna coordinate system. For the "Mapped 3D" pattern, the "phi" corresponds to the same "phi" in the antenna coordinate system while the "rho" in the "Mapped 3D" pattern corresponds to the "theta" in the antenna coordinate system.

Appendix BI RFID Design and Conjugate Match Factor

RFID are widely used nowadays. IE3D is excellent for RFID designs. RFID designs may involve RFID tag antennas, reader antennas and the ICs. IE3D is good for all the aspects. Specifically for RFID tag antennas, we should be more careful because the requirements are different from regular microwave antennas.

A regular microwave antenna is fed by a wave source. We would like the incident wave to be completely consumed by the antenna without reflection. For this purpose, we define the antenna gain (AG) as:

AG (dB) = Directivity (dB) + Radiation_Efficiency (dB) + Return_Loss (dB) (BI-1)

The Return_Loss is the loss due to reflection. The reflected wave back into the source will be absorbed by the source and it is considered as loss.

For an RFID tag antenna (Za), the source can be considered as a constant voltage source with complex source impedance (Zs). The system diagram is shown in Figure BI-1.

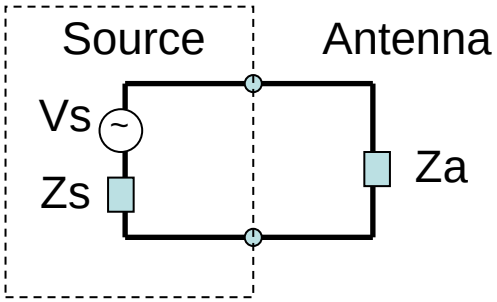


Figure BI-1. The system diagram of an RFID tag.

No matter what Zc is used, the reflected wave should not be considered as loss. The Antenna Gain and Antenna Efficiency listed in the Pattern Properties on IE3D are not useful to RFID tag antenna designers. Instead, we should use the Conjugate Match Efficiency, Voltage Source Efficiency and Conjugate Match Factor to judge how good the RFID antenna is. Conjugate Match Efficiency is defined on IE3D as the efficiency of the constant voltage source when the Za = Zs* (conjugate of Zs). In such a case, Za consumes 50% of the power and Zs consumes 50% of the power. Of the 50% of the consumed power by Za, the power radiated out is represented as Radiation Efficiency. Therefore, Conjugate Match Efficiency is basically 50% of the Radiation Efficiency. Similarly, we define Voltage Source Efficiency in IE3D as the ratio between the Radiated Power and the Total Power of the constant voltage source with any Zs and Za values, while the Conjugate Match Efficiency is the Voltage Source Efficiency when Za = Zs*.

Voltage Source Efficiency and Conjugate Match Efficiency are accurate measures of how efficient the constant voltage source Vs is or can be. However, for RFID designs, we should not look at the maximum efficiency. We can easily achieve higher Voltage Source Efficiency than Conjugate Match Efficiency with Za not equal to Zs. To get the maximum efficiency, we in fact can increase Za indefinitely and we will be able to achieve the maximum efficiency while the power delivered to Za is 0.

What we really should look for in RFID tag antenna design is the maximum power delivered to Za. To achieve maximum delivered power, we should guarantee Za = Zs*. We will be able to achieve maximum delivered power to Za. In this condition, we will have the Voltage Source Efficiency equals to Conjugate Match Efficiency. The question is how far we are from the optimum situation when Za is not the same as

Zs*. Certainly, we can't use Voltage Source Efficiency to judge it because comparison between Voltage Source Efficiency and Conjugate Match Efficiency does not mean anything.

Some designers have been considering normalizing s-parameters to the complex source impedance to see how good an RFID tag antenna is. Normalizing s-parameter to complex Zc is not theoretically solid. Serious complex Zc, with imaginary part comparable to the real part, may introduce many conflicting results and conclusions. Traditionally, the reflection coefficient Γ, defined on real value Zc, is

$$\Gamma = (Za - Zc) / (Za + Zc) \qquad \text{(AG-2)}$$

It is a measure about how much reflection is created by Za. The$|\Gamma|^2$ is an indication how much of the incident power is reflected back to the source by the antenna. If we use complex Zc in the above formula, the Γ will lose the meaning of reflected power. In fact, you may even get $|\Gamma| > 1$ for passive Z in some special cases. For example, assume we have Zc = 20 – j 100 and Za = 15 + j 80. Substituting them into the formal Γ = (Za-Zc)/(Za+Zc) = (5–j180)/(35+j20) = 4.467 /-118.15°. Apparently, $|\Gamma| > 1$ and it is impossible for a passive circuit to yield more reflected power than incident power. The Γ definition is certainly not appropriate for a measure on how good a passive system is when Zc is a complex value. Some people use the definition Γ = (Z-Zc*)/(Z+Zc) and some other definitions. The definitions may avoid the serious problem of $|\Gamma| > 1$ for passive elements. However, they will lose the meaning of $|\Gamma|^2$ represents the reflected power of a single port system. What is the problem? The fundamental problem is that complex Zc can't be used precisely to describe the performance of a circuit. Please read the Appendix D for more detailed explanation.

The question is, "what should we use to judge how good an RFID antenna is in the reality?" To answer the question, we have defined Conjugate Match Factor (CMF) on IE3D for this purpose. We denote the power delivered to Za as Pa. When we achieve Za = Zs*, Pa achieves its maximum Pa(max). We define Conjugate Match Factor as:

$$CMF = Pa / Pa(max) \qquad \text{(AG-3)}$$

Assume Zs = Rs + j Xs and Za = Ra + j Xa, we will have

$$CMF = 4 Ra Rs / | Zs + Za |^2 \qquad \text{(AG-4)}$$

$$CMF(dB) = 20 \log10(CMF) \qquad \text{(AG-5)}$$

When Za = Zs*, CMF = 1 and we achieve the maximum power delivered to Za with given Vs and Zs. When Za is not the same as Zs*, CMF < 1 and CMF^2 is the measure how much less power you achieve compared to the optimum case. In the other word, given the Zs and Za, CMF(dB) is an indication of how much you lose in the delivered power compared to the optimum case. Apparently, CMF serves the same purpose as the reflection coefficient Γ in a microwave transmission line system with real Zc or slightly complex Zc. When the Zc is slightly complex with the imaginary part much smaller than the real part, the definition of Γ is still quite accurate even though it is not exact.

CMF can be visualized and processed on MGRID, MODUA and PATTERNVIEW. You should use CMF, instead of complex Zc, to judge how good matching your RFID tag antenna is. To judge how good it radiates, you should use Radiation Efficiency. If you can achieve CMF = 1 and the maximum Radiation Efficiency, you should achieve the best design.

Appendix BJ Shift of Reference Plane and Coupled Ports.

For an extension port, we can shift the reference plane of the port to some other location. The location can be fictitious (see Figure BJ.1) or the location of the shifted reference can be beyond the start point of the uniform transmission line (TLN). When you use the extracted s-parameters as a sub-circuit of a larger circuit, we still need to have a long enough section of uniform TLN in your larger circuit in order to match the reality better. If that is the case, why should we shift the reference plane? We can just simulate the complete sub-circuit and use it as a building block for a larger circuit. It is true for most of cases. However, there are cases we still want to shift the reference plane.

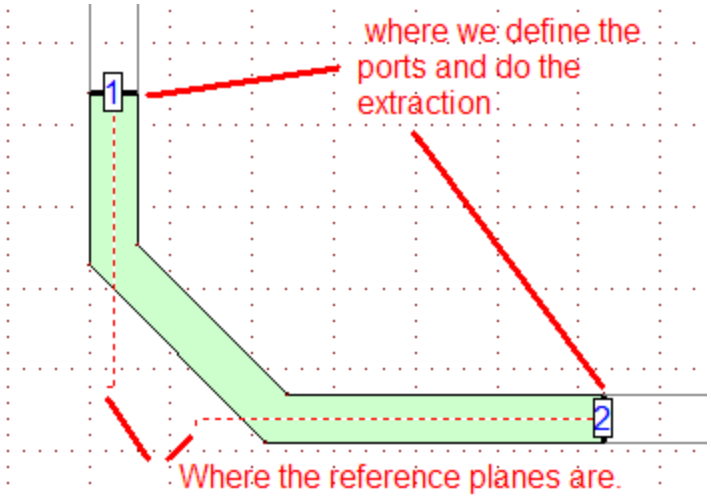


Figure BJ.1 The illustration of shifting of reference plane (.\ie3d\samples\for_shift_reference_plane.geo).

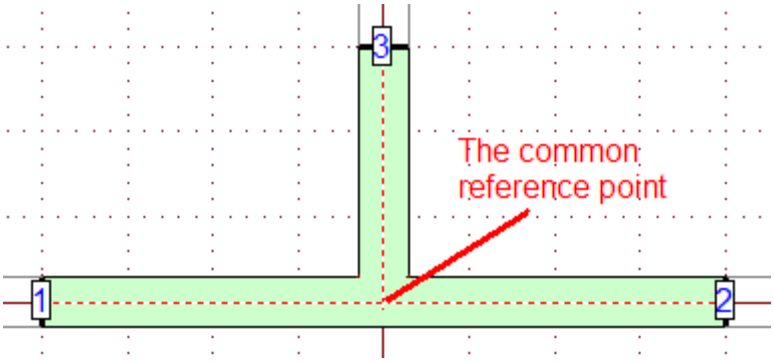


Figure BJ.2 A T-junction with TLN segments on the arms.

Shown in Figure BJ.2 is a T-junction with an additional TLN on each port. We certainly can simulate it as a complete structure and use its s-parameters as a building block for a larger circuit. For each different setting of the TLN lengths, we can build a different T-junction with different TLN lengths. It may not be as convenient to break the structure into smaller building blocks. There is little coupling between the three TLN segments. We can consider that the structure is composed of four parts: The T-junction without length and a TLN segment on each of the port of the T-junction. The length of each TLN segment is measured from the reference to the port location (see Figure BJ.2). By cascading the four parts together, we are able to get almost identical results. In case we change the lengths of a TLN, we can just change the s-parameters of the TLN easily. We may be able to get accurate s-parameters of a TLN with any given length value easily from a simple TLN calculator. Using this approach, we may be able to simplify the procedure of circuit synthesis.

It is easy to define a shift of reference plane on an extension port on IE3D/MGRID. To define the shift of reference plane on a port, immediately after you select the edges for a port, you can select Port->Define Reference Plane and use the mouse to click at wherever you want to be the reference plane. You will be prompted for the value of the final shift. Then, you select OK after you correct the value and the reference plane is defined. To change the reference plane of a port, you can select Port->Ports Properties and you will be able to modify the shift of reference plane and other parameters of a port.

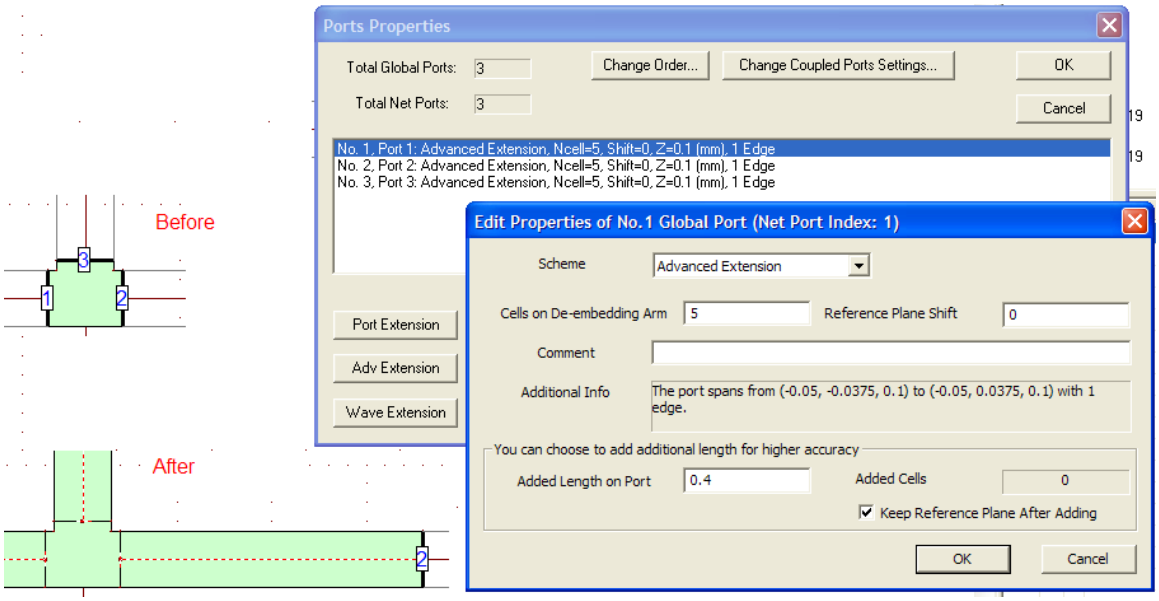


Figure BJ.3 Adding lengths to ports without shifting original reference planes.

It is suggested that you should try to use some additional length of TLN on an extension port in order to get the highest accuracy. Shown in Figure BJ.3 is the situation. The TLNs in structure in "Before" (saved in .\ie3d\samples\short_arms_tee.geo) are too long. The extracted s-parameters might be affected by the discontinuities of the junction even though it may not be very significant. For best accuracy, you are suggested to add additional length for it. You can go into Port->Ports Properties and double click at each port to change the Added Length on Port. For this particular example, we enter the Added Length on Port as 0.4 mm. In reality, adding even 0.1 mm (more than the width 0.075 mm of the port) will make the results more accurate. Please make sure you check "Keep Reference Plane After Adding". MGRID will automatically redefine the reference planes of the ports so that they are not changed after the adding additional lengths. When you use this feature to add additional length to a port, you are suggested not to use too short or too long an additional length. Either case may decrease the accuracy. If it is too short, the results are affected by the discontinuities. If it is too long, the shift of reference plane will introduce accumulated error. Normally, we suggest you to use a length of about 1 to 5 times of the port width.

Shift of reference plane was implemented in early version of IE3D. It has been working very well for an isolated port or a port without other coupled ports around. However, it was not suitable for shift of reference planes for coupled ports. If you define shift of reference plane on some coupled ports, it might yield incorrect results.

One of the major implementations we have done on IE3D 14 is the feature of automatic shift of reference plane of coupled ports. We are able to shift the reference planes of coupled ports. Saved in .\ie3d\samples\coupled_ports.geo is an example (see Figure BJ.4). It contains three ports (or 4-ports if we count port 1 as + and – ports). Port 1 is a differential port with +1 and -1 ports coupled together. Port 2 and port 3 are single-ended ports while they are coupled. The reference planes of all the ports are shifted to the center. Theoretically, we should get dB[S(1,1)] = dB[S(2,2)] = dB[S(3,3)] = about -9.5 dB and dB[S(2,1)]

= dB[S(3,1)] = dB[S(3,2)] = about -3.5 dB. If you have IE3D V12 or other earlier versions, you can try to simulate it. You will get the expected results at low frequency while the results will be different at high frequency. This is due to the fact we shift the reference plane of individual port assuming other ports are not coupled together. It neglects the coupling between the arms.

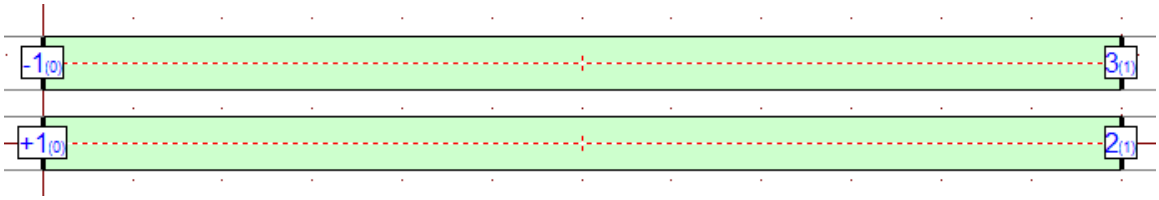


Figure BJ.4 Coupled ports with shift of reference planes in .\ie3d\samples\coupled_ports.geo.

In IE3D Version 14, we have implemented an automatic shift of reference planes for coupled ports. IE3D 14 automatically detects that port +1 and port -1 are coupled ports and port 2 and port 3 are another group of coupled ports. The group indices are shown as subscript in the port indices (see Figure BJ.4). As it is shown, port +1 and port -1 belong to the coupled port group 0 while port 2 and port 3 belong to the coupled port group 1. Simulate the structure in .\ie3d\sample\coupled_ports.geo from 0.1 to 20 GHz. You will see its results are close to the expected values over the whole frequency range.

The question comes up is: What are coupled ports? For an open system, all the ports are coupled together. Should we consider them as coupled ports? If we do, the simulation process will be greatly complicated. It may make EM simulations impractical if we have to consider the coupling between all ports when there are very slight coupling between them.

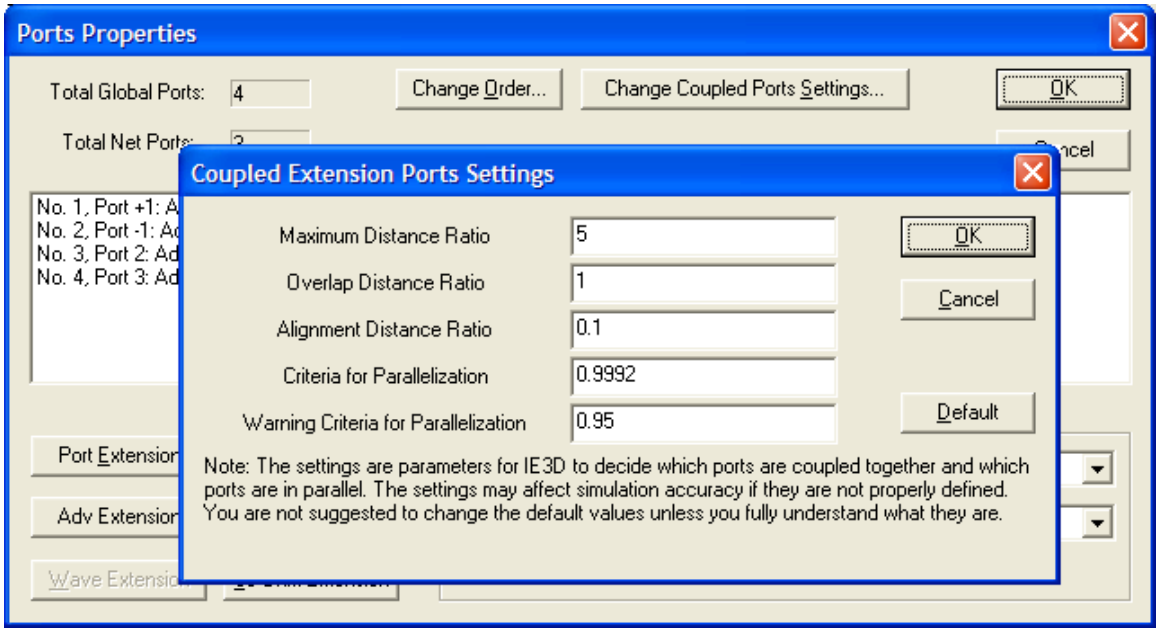


Figure BJ.5 The criteria on coupled ports.

Normally, the coupling between two ports is very small when they are perpendicular to each other. A typical example is the port 1 and port 3 in the T-junction in Figure BJ.2. The coupling between two ports is strong when they are parallel to each other and they are side by side. A typical example is the port 2 and port 3 in Figure BJ.4. Another example is the port +1 and port -1 in the same Figure. If two ports are not side by side or aligned, the coupling between them will be small even they are parallel to each other. A typical example is the port 1 and port 2 in the T-junction in Figure BJ.2. Based upon the above facts, we

have defined some criteria for judging whether the ports are coupled or not coupled. They are accessible from Port->Ports Properties->Change Coupled Ports Settings (see Figure BJ.5). Five parameters are used to decide whether two ports are coupled together. They are documented in Table BJ.1. The default values are arbitrarily chosen. You can change the default values. Again, there are limitations on coupled ports. They should be parallel and aligned so that they can be modeled with high accuracy. If some ports are not very close to each other while they are not far. We may not apply the limitations of parallelization and alignment to them. However, we may introduce, big or small, some numerical error if we don't assume them as coupled ports. As a designer, you will need to make a decision which approach you want to take. The related issue is that how we should divide a larger circuit into sub-circuits. It is more accurate to simulate a complete larger circuit as a whole. However, it may take more RAM and longer time or it even can't simulate it when it is too large. Dividing a larger circuit into sub-circuit will make the simulation much fast while you will lose some accuracy. You have to make a decision on it.

Table BJ.1 The criteria for coupled ports.

Parameters	Description
Maximum Distance Ratio (Short Name: MDR) (Default: 5)	If the ratio between distance between two ports and the meshing cell size is within MDR, we may consider them as coupled ports if they meet other criteria.
Overlap Distance Ratio (Short Name: ODR) (Default: 1)	If the ratio between the overlapped portion of two parallel ports and the meshing cell size is larger than ODR, we may consider them as coupled ports if they meet other criteria.
Alignment Distance Ratio (Short Name: ADR) (Default: 0.1).	If the two coupled ports are mis-aligned by a ratio of ADR compared to the meshing cell size, we will issue error message to you. You will need to correct the port configuration. Mis-aligned coupled ports may cause accuracy problem.
Criteria for Parallelization (Short Name: CFP) (Default: 0.9992)	It is the limit for cos (phi) where phi is the angle formed by two extension ports. If cos (phi) = 1, the two ports are completely parallel. If cos (phi) < CFP, we will consider the ports are coupled ports when they meet the other conditions. If cos (phi) > CFP, we will issue error message to you to change the ports so that they will become parallel.
Warning Criteria for Parallelization (Short Name: WCFP) (Default: 0.95)	When the cos (phi) > 0.95 for two coupled ports, we will issue warning to you. You should try to correct them so that they will not affect the accuracy.

Appendix BK Wide Band Simulations from 1 Hz to Very High Frequency

Full-wave EM tools, if implemented properly, are excellent for high frequency applications. They normally can include the radiation effects, the loss effects and high frequency other effects. Depending upon the algorithms and implementations, they can yield results with good accuracy.

The question is can we use it for very low frequency such as DC or 1 Hz? For some applications, we need to cover frequency range from DC to very high frequency.

Full-wave EM theory certainly is not limited to high frequency only. It is also applicable to 1 Hz or even lower as long as it is not DC. The problem is in the limited word length in digital computers. Due to limited word length, real numbers are represented as floating point numbers on a digital computer. Real number is continuous from -∞ to +∞. There is no gap between them. When we use floating point numbers on a digital computer to represent real number, we can only cover from about -1.0e+308 to about -1.0e-308 and from about 1.0e-308 to about 1.0e+308 using double precision. Also, the floating point numbers are isolated or countable number in the ranges. Apparently, using floating point numbers to represent real numbers is an approximation. For most applications, they are good enough. However, there are situations they are not good. One such situation is the case in any EM simulation algorithm with frequency going down to approach DC. When the frequency is so low, some involved positive real numbers may be much smaller than -1.0e-308 and some may be much bigger than 1.0e+308. When we use digital computers to do calculations for those values, we will get incorrect results ore lose accuracy. The fundamental of digital computer makes it impossible for us to go from very low frequency to very high frequency. To make this failure simple, we can't even represent frequency = 1.0e-500 Hz using a double precision number on a digital computer.

Some full-wave algorithms are developed to tackle low frequency simulation accuracy issues. They are claimed to be able to approach DC. However, it is strongly believed that it is possible such approaches may lower the low frequency limit while they can't completely remove the limit.

Some designers also suggest us to implement some special static EM algorithm to solve the DC solution and combine the DC solution with the high frequency EM solutions to cover the whole range. Such a scheme sounds reasonable. However, it is not practical due to the fact that different algorithms will yield different numerical results even with the same meshing. Assume we use IE3D to simulate a structure and its dB[S (1, 1)] is smoothly approaching -30 dB. If we use a static solver to solve the same problem, we may get the solution as -30.5 dB. If we combine the results together to have a single representation, there is a big jump between DC and the next frequency point. This big jump will cause serious accuracy problem in further circuit simulation such as HB simulation. The jump should not be a special case but a common case if we combine full-wave algorithms and static EM results together.

What are the solutions? Apparently, if we can extract the asymptotic behavior of the s-parameters approaching DC, we should be able to get smooth and accurate results covering from DC to very high frequency. The question is how we can extract the asymptotic behavior of the s-parameters approaching DC reliable and accurate. In IE3D 14, we have implemented some physics based algorithm allowing IE3D to extract the asymptotic behavior of the s-parameters accurately and robustly.

Saved in .\ie3d\samples\low_freq_to_1Hz.geo is an example. It is a spiral inductor structure. We have simulated it from 1 Hz to 10 GHz. The layout and s-parameters are shown in Figure BK.1 with log frequency scale. It is interesting to see the smooth change from 1 Hz to 10 GHz on both dB[S (1, 1)] and dB[S (2, 1)].

You may be curious why the dB[S (1, 1)] does not approach -∞ when the frequency is approaching DC. Instead, it approaches about -32 dB. It is believed that the dB[S (1, 1)] of an ideal TLN should approach -∞ when the frequency is approaching DC in many text books. A spiral is not a TLN while its low frequency behavior should be close to a TLN. The answer is that the dB[S (1, 1)] should approach -∞ when

the frequency is approaching DC, if the structure is a completely lossless structure. If there is no metallic loss, no dielectric loss and other losses, it will be true. For this particular example, those losses are there and it will approach to some finite value. Even we define no metallic and dielectric loss to the structure, the dB[S (1, 1)] value will not approach -∞. Instead, it will approach some value much lower than -32 dB.

The low frequency extrapolation is controlled by the Low Frequency Settings in Basic Parameters. The low frequency for simulation is set to be 0.02% of the Meshing Frequency. We perform extrapolation based upon the results around this frequency for the results below the frequency points. You just need to make sure IE3D results are accurate above this frequency point. IE3D is able to yield accurate results below the frequency based upon some physics-based advanced extrapolation scheme. In practical applications, you may adjust the values for high accuracy results. If you adjust the values (not widely), you may get slightly different results. For example, you may see the dB[S (1, 1)] may approach -33 dB when frequency is approaching DC while the results are always smooth. It is impossible to avoid some small differences due to the fact the simulation algorithm is a numerical algorithm. There are always numerical errors involved. The most important thing is that we are able to obtain reliable and smooth high accuracy results from 1 Hz to very high frequency. It is enough for further ultra-wide band circuit simulations.

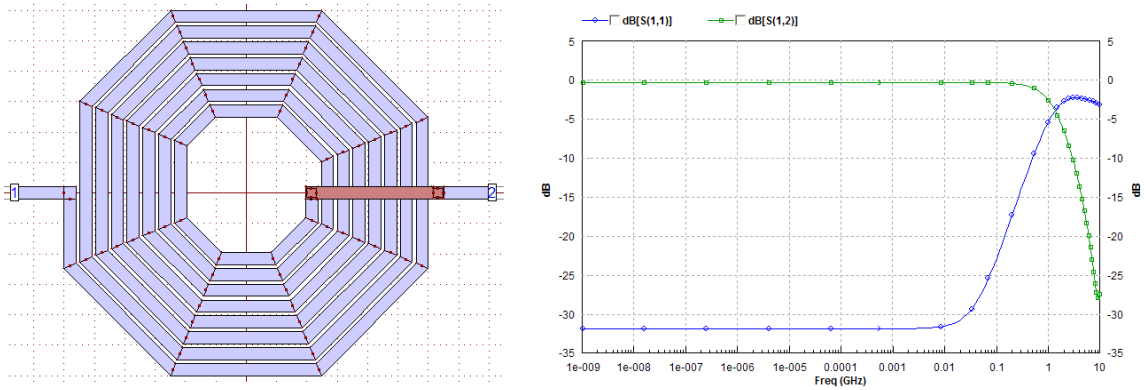


Figure BK.1 The simulated s-parameters from 1 Hz to 10 GHz.

Appendix BL Avoid Time-Consuming Geometry Editing for Large Structures.

Modern computers are much more powerful than those years ago. IE3D is also much more powerful than earlier versions. The structures we are simulating are also much bigger. When the number of polygons involved in a structure is getting bigger and bigger, the geometry editing process is also becoming tougher and tougher.

The geometry editing on MGRID is based upon polygons. MGRID allows users to create and manipulate the polygons to create the model for IE3D full-wave EM simulation. The polygons created should meet some kinds of specifications so that IE3D can yield accurate results top you. For every command you do, MGRID will do many internal geometry checking to make sure the polygons and ports are correct. For example, it will try to connect the common edges of the polygons. If two vertices are very close to each other, it will try to merge them into one. It detects whether the polygons are overlapping to each other. It will also to check whether the ports are correct after a geometry editing command. The checking may take much computational power. When the number of polygons N is getting big and big, the geometry checking may take longer and longer time. Apparently, the checking is mainly between polygons. The time required is kind of proportional to N^2. It is increasing very fast with large N. It may take many seconds or some minutes of waiting time after one geometry editing command when the N is large. For this reason, we have implemented a scheme to allow MGRID to skip some geometry validations when the N is too big. Shown in Figure BL.1 is the Optional Parameters dialog (accessible from Basic Parameters or Parameters->Optional Parameters). We have implemented Polygon Limit for Checking (PLC) and Geometry Checking Skipping (GCS). PLC=2000 means that when the polygon count N exceeds 2000, we will skit some important geometry checking after each command. GCS=0 means that MGRID will warn you about the fact every time an avoided geometry checking happens. If you choose GCS = 9, MGRID will warn you about the fact every 10-times it skips the checking. You can change both values for the current structure when MGRID is issuing the warning. To change the settings for the whole application, you should do it on Parameters->Optional Parameters and apply the changes to the whole application when you exit.

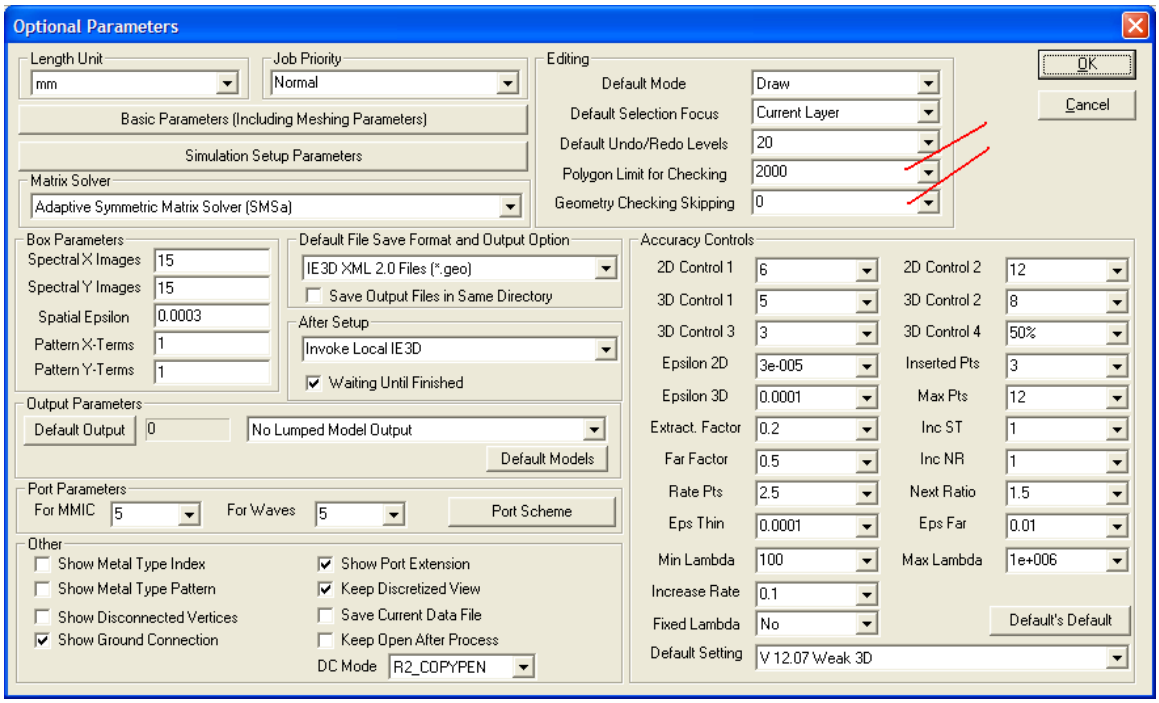


Figure BL.1 The optional parameters dialog.

Appendix BM Major Features of IE3D 14.1

IE3D Release 14 was released in February 2008. In less than three months, we have further improved IE3D significantly and we released the IE3D 14.1 in May 2008. This appendix documents the major features in IE3D 14.1.

Major Features of IE3D 14.1:

(1) Full implementation of OpenMP multi-CPU support for the major processes of IE3D full-wave simulation engine. For most structures, the V14.1 engine can be more than twice as fast as the V14 engine on a quad-core computer.
(2) Further improved GUI to simplify the layout editing and visualization. Some memory leakage problem on 3D view in IE3D 14.0x is fixed in IE3D 14.1.
(3) Implementation of non-isotropic metallic types for more accurate and efficient modeling of pin via arrays in modeling RFIC structures.
(4) Implementation of advanced simulation algorithm for accurate modeling of thick metal with multi-fold simulation speed improvement.
(5) Implementation of automatic model simplification by removing small dummy polygons and slots required in 45-nm CMOS processes in AGIF. It allows significant improvement in simulation speed in modeling RFIC without losing much accuracy.

Major Features of IE3D 14.0:

(1) Complete EM design environment with integrated layout editing, full-wave EM simulation, real-time EM tuning and optimization, s-parameters, current distribution, near field and radiation pattern visualization and post-processing.
(2) Implementation of text objects for better documentations of designs.
(3) Much improved layout editing with easy access and control of geometry objects, and options to speed up geometry editing of large structures.
(4) Implementation of Boolean operations for connecting arbitrary 3D objects for high accuracy and easy geometry modeling.
(5) Modeling of surface roughness.
(6) Wide frequency s-parameter extraction covering from 1 Hz to 10 THz.
(7) Coupled ports and differential ports de-embedding in highly packed circuits.
(8) Improved multi-CPU support and significantly improved simulation speed on Windows 64.
(9) Further improved layout editing on IE3DLibrary for Boolean driven, equation-based geometry modeling with easy parameterization, access and control of objects.
(10) Automatic creation of IE3D/AGIF templates from TSMC unified technology files for automated layout creation and batch IE3D EM simulations from GDSII files.

1. Full Implementation of OpenMP Multi-CPU Support:

OpenMP allows compiled code to fully utilizing multiple CPUs in modern computers. Implementation of OpenMP into C++ codes requires significant testing and validation in order to avoid problems caused by out of sync threads. We have released multi-CPU support for matrix solving in IE3D 11.5. The bottleneck for speeding up an IE3D simulation was on the matrix solving. We have spent much effort to implement OpenMP into the matrix filling part of IE3D. Major numerical processes for surface meshing on IE3D 14.1 are now supported by multi-CPUs. The implementations for volume meshing and some other less important features on IE3D 14.1 are still going on.

For a medium size problem, IE3D 14.1 is about 50% faster than IE3D 14.0x on a Core 2 Duo computer. It is more than twice as fast on a Core 2 Quad computer.

2. Further Improved GUI to Simplify Layout Editing and Visualization:

We have significantly improved the layout editing and visualization by introducing the MDI (multiple document interface) on IE3D 14. In IE3D 14.1, we have further polished it. On IE3D 14.0X, we allow users to zoom structures by scrolling the mouse wheel. However, the direction was in opposite to Microsoft Office. We have reversed the direction to make it compatible with Microsoft Office. Panning on IE3D 14.0 requires users to hold down SHIFT button and draw left mouse button. On IE3D 14.1, users can draw press down the mouse wheel and draw to do panning, in the same was as it is on some standard Windows applications.

IE3D 12.x does not clear the Undo queue when users perform simulations. On IE3D 14.0x, we have cleared the Undo queue for simulations in order to release as much memory as possible for the IE3D simulation engine. Upon customers' requests, we have re-implemented saving undo queue for simulations. Users can decide how many undo items in the queue for preserving previous changes.

We implemented a new 3D view on IE3D 14.0. It greatly improves the 3D visualization quality. However, there is some memory leakage in the 3D view which may accumulate to large amount of wasted memory in using IE3D EM Design System. We have fixed the memory leakage problem in IE3D 14.1.

3. Implementation of Non-Isotropic Metallic Type for More Accurate Modeling of Pin Vias in RFIC:

Pin array vias are used extensive in CMOS semiconductor process. In full-wave modeling, IE3D allows designers to merge the pin vias automatically into big vias to simplify the simulation without losing much accuracy. The big via model is a good model for pin array vias. One aspect of pin array vias was not addressed in IE3D 14 and earlier version. The pin array vias do not allow current to flow in the horizontal direction while the big vias allow such flow. They may introduce some differences at high frequency. To further improve the accuracy in modeling pin array vias in RFIC, we have implemented non-isotropic metallic types on IE3D. AGIF users can choose "Omit XY-Current" in the Layer Mutual Operations for merging pin vias into bigger ones. For pin array vias modeling, "Omit XY-Current" should yield more accurate and faster results.

Non-isotropic metallic type is also implemented in some (not all) commands for building vias on IE3D EM Design System (MGRID). Users can choose "Omit XY-Current" option. MGRID will create new non-isotropic metallic strip types for the polygons for the vias. Normally, the model may be slightly less accurate if X and/or Y-directed current do exist in the real structure. The simulation may be faster with non-isotropic current modeling.

4. Implementation of Advanced Simulation Algorithm for Accurate Modeling of Thick Metal with Multi-Fold Speed Improvement:

The thickness modeling in IE3D has been proved to offer high accuracy simulation results for structures where the strip thickness is no longer relatively small. However, thickness modeling does slow down a simulation by a big factor. It may be more than 10 times slower than the thin model due the fact the number of cells and the number of unknowns involved are increased substantially.

For a thick strip, the current is mainly flowing horizontally on the top, bottom and the side surfaces. There is a big waste of computational resource when we do a full-3D model on it. Using the non-isotropic metallic strip type, we are able to limit the current flowing in X and/or Y-direction for thickness traces. In this way, we can significantly reduce the number of unknowns involved in thickness models. The simulation speed is increased substantially while the model can still capture the major effects of thickness trace.

When users try to build thickness on layers or polygons on IE3D/AGIF, users have an option of "Omit Z-Current". IE3D/AGIF will automatically create some non-isotropic metallic types for the polygons for the thickness traces.

During editing, some non-isotropic metallic polygons may be changed. For example, a 3D polygon with metallic type with option "Omit XY-Current" may become a horizontal polygon. IE3D will automatically detects it and ignore the "Omit XY-Current" option.

In case you have created your model with "Omit XY-Current" for vias and "Omit Z-Current" for thickness polygons, and you still want to obtain the results without non-isotropic metallic types, you can select the option "Convert Non-Isotropic to Isotropic Metallic Types at Run-Time" in the Metallic Strip Types tab of Basic Parameters for the project. It is a general switch to overwrite any non-isotropic metallic types defined at run-time. Anytime, you don't want to overwrite the individual non-isotropic metallic strip types, you can select the option "Non-Isotropic Metallic Types as Defined".

For Automatic Run-Time Thickness (ARTT), you can choose the option "Use Non-Isotropic Types in Automatic Run-Time Thickness" if you want to improve the simulation speed by using non-isotropic metallic strip types for ARTT.

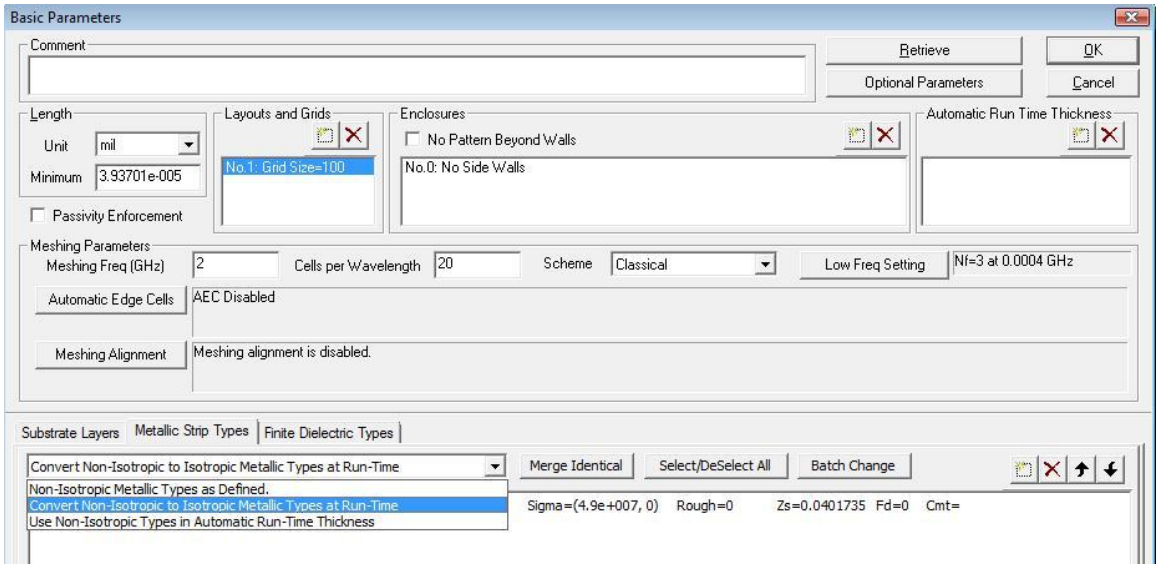


Figure BM.1 The Metallic Strip Types tab in Basic Parameters.

5. Implementation of Automatic Model Simplification by Removing Small Dummy Polygons and Slots Required in 45-nm CMOS processes in AGIF:

IE3D and AGIF make it simply to create models and perform accurate and efficient full-wave simulations on RFIC structures. There are some special requirements in 45-nm CMOS processes. A 45-nm layout in GDSII is actually shrinked to 40-nm in manufacturing. On AGIF, you can define the Zoom parameters to achieve such a shrinking. In 45-nm CMOS processes, it is required to create some dummy metallic objects in the layout and insert some slots in wide traces. The dummy metallic objects or slots create much detail and it will slow down any numerical simulation substantially if they are not "cleaned". On IE3D/AGIF, users can choose to remove the dummy polygons and slots in the Layer Self Operations. Omitting the details from the dummy polygons and slots may sacrifice a little in accuracy while it makes an EM simulation much faster. With the small detail, it may be even impossible to simulate the structure. IE3D/AGIF can perform the "cleaning" automatically for you with criteria controlled by the users.

There are many other improvements and bug fixing based upon the feed back from customers. We appreciate the great supports to us from our customers.

Appendix C. Verification on Lossy Dielectric Modeling

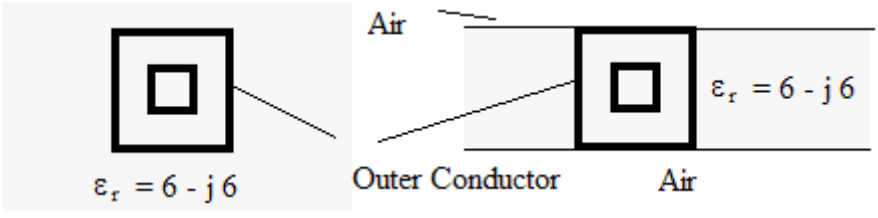
IE3D is implemented with the capability to model lossy dielectrics. We will use some well-defined examples to verify the robustness and accuracy of the lossy dielectric modeling.

As you may know, the transmission properties of coaxial structures are independent of the environment surrounding it, regardless the shape of the cross-section. IE3D's basic formulation is for open structures. We will demonstrate the accuracy of the IE3D in modeling coaxial structures and verify the modeling of lossy dielectrics, by using some rectangular coaxial structures.

Saved in **.\ie3d\samples\rcoax.geo** is a rectangular coaxial structure. The size of the outer conductor is 0.1 mm. The size of the inner conductor is 0.05 mm. The coaxial length is 0.5 mm. The coaxial structure is embedded into an infinite dielectrics of $\varepsilon_r = 6 - j\ 6$. Building the structure is quite simple. The most important issue is the definition of the balanced port. As modeling of coaxial structures will be discussed in Appendix J, we will not provide any detail on building the rectangular coaxial structure.

We simulated the structure and saved the result in **.\ie3d\samples\rcoax.sp**. We expect the ε_{reff} should be the same as $\varepsilon_r = 6 - j\ 6$. We can find out the transmission parameters of the **rcoax.sp** using MODUA: (1) Run MODUA. (2) Select **Display Parameter Module** to display **rcoax.sp**. (3) **Select Display Toggle** in **Control** for the design display. (4). Select **Find Transmission Line Parameters** in **Process** menu and enter the **Transmission Line Length** as 0.5 mm. MODUA will save the transmission parameters into **.\ie3d\samples\rcoax.tra**. Open the **rcoax.tra** file on a text editor. You will see the IE3D calculated $\varepsilon_{reff} = 6.0152 - j\ 6.0108$ at 20 GHz which is very close to supposed value of $(6 - j\ 6)$.

For **.\ie3d\samples\rcoax.geo**, we built the whole space as a single dielectrics with $\varepsilon_r = 6 - j\ 6$. Certainly, we can get very accurate modeling of lossy dielectrics. The question is how accurate the lossy dielectric modeling will be for multi-layer lossy dielectrics. Saved in **.\ie3d\samples\rcoax1.geo** is an example of the rectangular coaxial with multilayer lossy dielectrics. Theoretically, **rcoax1.geo**'s result should be identical to that of **rcoax1.geo** because the dielectrics enclosed by the coax are still the same. Only the outer environment is changed. Practically, **rcoax1.geo** is more challenging check on the lossy dielectric modeling. The simulation result of **rcoax1.geo** is saved in **.\ie3d\samples\rcoax1.sp**. The transmission parameter file is saved in **.\ie3d\samples\rcoax1.tra**. The IE3D predicted $\varepsilon_{reff} = 5.92009 - j\ 5.82249$ at 20 GHz. The error is about 2%. We can claim that the result is quite accurate, considering the ε_{reff} is the most sensitive parameters to numerical error (2% error in ε_{reff} corresponds to 1% error in frequency). For **rcoax1.geo**, we also have a coarse meshing. Each side of the rectangular coaxial is only meshed into 1 to 2 cells. The outer conductor will have difficulty to completely stop field from penetrating into the air region that is just outside the outer conductor. When field is slightly penetrating into the air, we naturally will see some small drop in the ε_{reff}.



RCOAX.GEO RCOAX1.GEO

Figure C.1 The cross-sections and dielectric setup of the RCOAX.GEO and RCOAX1.GEO. In the IE3D simulation, only 1-2 cells are used on each side of the rectangular coaxials.

Appendix D. Some Comments on Complex Zc

In reality, a TLN structure always involves loss. When loss is involved, we will encounter complex Zc. Naturally, we may expect the voltage and current to behave like:

$$V = Vi \cdot \exp(-\gamma z) + Vr \cdot \exp(\gamma z) \quad \text{(D-1)}$$
$$I = Vi / Zc \cdot \exp(-\gamma z) - Vr \cdot \exp(\gamma z) \quad \text{(D-2)}$$

(D-1) and (D-2) imply the following formula still hold,

$$V = a + b \quad \text{(D-3)}$$
$$I = (a - b) / Zc \quad \text{(D-4)}$$

Where a and b are the incident wave and reflected wave defined as,

$$a = Vi \cdot \exp(-\gamma z) \quad \text{(D-5)}$$
$$b = Vr \cdot \exp(-\gamma z) \quad \text{(D-6)}$$

The question is "Are (D-1) to (D-6) still valid?" If they are invalid, what are the correct formulas for them? The answer is the following.

Precisely, waveguide theory is based upon the assumption of some uniform structure in z-direction bounded by some PEC, PMC or some wave-stopping boundaries in the side boundaries. For traditional rectangular waveguide and circular waveguides, they are PEC. We can also find the corresponding cases for PMC boundaries. For optical waveguide situation, we may not be able to have PEC or PMC boundaries. However, we may be able to have some kind of wave-stopping boundary condition. For example, a typical optical waveguide is a dielectric cylinder of permittivity Er1 inside a a uniform dielectric space of permittivity Er2. When Er1 and Er2 are both real and Er1 > Er2, the waves can be bounced back and forth inside the cylinder with total reflection. In such a case, even the boundaries are not PEC and PMC, they do stop the wave from propagating in the transverse directions. We will refer PEC, PMC and other boundaries stopping the waves from propagating in the transverse directions as wave-stopping boundaries in the following discussion.

When no energy is dissipated in the cross-sectional directions, we are able to find the modes of the uniform structure. From the mode, we can find the incident wave and reflected wave along the z-axis. For each mode, the field pattern (either E-field or H-field) at a specific cross-section is the same as the one at other cross-sections except the magnitude is different. This is a very important concept. When the field patterns do not change, we can define an impedance as the ratio of the E-field magnitude and the H-field magnitude. We define this ratio as the wave impedance η. When the cross section's boundary is a wave-stopping boundary, we will have the wave impedance η as a pure real number even though the material inside the waveguide can be lossy. The loss inside the waveguide will cause the magnitude of a field pattern to change along the z-direction while the relative shape of the field pattern will not change.

Assume we have a lossy transmission line (or waveguide). The loss should be from the non PEC, non PMC or other non-wave-stopping boundary for the cross-section and it does not matter whether the filling dielectrics has loss. You may intuitively expect the non-wave-stopping boundary in the cross-section will yield complex Zc. In fact, it is not correct precisely. When such a situation happens, waveguide theory is no longer precise. Such a claim is from the following reasoning: When you have non PEC or non PMC boundary for the cross-section, you will expect complex Zc. Complex Zc means that E and H field are not in phase. When the E and H are not in phase, there will be energy flow in the X or Y-direction. The energy flow in X or Y-direction comes from the non-wave-stopping boundary in the cross-section anyway. The problem is that this energy flow will cause the field pattern changes along the z-axis. When the field pattern changes along the z-axis, the foundation of waveguide theory is no longer there. The conclusion is that

when complex Zc (or non pure real Zc) is involved, waveguide theory and transmission line theory are no longer precise. They are also approximations. When waveguide theory and transmission line theory are approximations, microwave network theory based upon Zc is also an approximation. We can no longer write the waves or voltage as some waves propagating along the z-axis. We have to expect them to propagate in all the directions. We have different formulas to describe energy flow:

$$P = \tfrac{1}{2} \int E \times H^* \, ds, \text{ over an area} \quad \text{(D-7)}$$

$$P = \tfrac{1}{2} (|a|^2 - |b|^2) \quad \text{(D-8)}$$

$$P = \tfrac{1}{2} \, V \cdot I^* \quad \text{(D-9)}$$

The Poynting vector in (D-7) is always valid for any EM field. When a TLN is a TEM TLN, E and H are conservative field at a cross-section, (D-7) is reduced to (D-9). For a non-TEM mode, (D-9) is no longer precise. When a waveguide does not have a lossy boundary in the cross-section, the E and H field can be separated into incident wave and reflected wave in z-direction and the Zc is complex. (D-7) is reduced to (D-8) and we can represent energy flow using (D-8), while (D-9) is an approximation. When there is a lossy boundary in the cross-section, the E and H field can not be separated into incident wave and reflected wave along z-direction. (D-8) and (D-9) are approximation. (D-1) to (D-6) are also approximations. In fact, if we use complex Zc in (D-1) to (D-6), we may end up with S(i,j) > 0 dB in some special situations. Some users may be very serious about S(i,i) > 0 dB because it means non-passive. In fact, please keep in mind that S-parameters are already approximations when lossy transmission lines are encountered. S-parameters are defined based upon the incident wave and reflected wave along the transmission line direction. When the boundary in the cross-section is lossy, incident wave and reflected wave are already approximations. The s-parameters based upon the incident waves and reflected wave are certainly approximations.

In reality, all waveguides will have non-wave-stopping boundaries. When we use waveguide theory to design them, we should keep in mind that the results we obtain from waveguide theory are approximated ones. When the boundaries are very low loss, we can obtain highly accurate results. When the boundaries become more and more lossy, the results obtained using waveguide theory will degrade. What are the criteria we may obtain inaccurate results using waveguide theory? When we provide the structure and material properties into a waveguide cross-sectional solver, it always can find some modes. That is because we have already assumed something in the foundation of the waveguide cross-sectional solver. The modes may not be necessarily there in reality. You may want to monitor the transverse E and H fields and the Kc value of the outer most dielectrics. ($Kc^2 = K^2 + \gamma^2$ for each dielectric type in the cross-section). If the E and H field in the outer most dielectrics are significant, the phase difference between the transverse E and H fields becomes big or the imaginary part of Kc value has significant imaginary part, it is an indication that much power is dissipated in the transverse direction. Using waveguide theory on such a uniform structure may cause accuracy problem. Just the imaginary part of the Kc value of the outer most dielectrics does not form complete criteria for such a judgment. For example, a rectangular waveguide with very good conducting walls. You can consider the conducting wall is the outer most layer. We may expect the imaginary part of Kc can be quite significant compared to the real part of Kc. We can still get very accurate results using waveguide theory on it. The reason is that the field getting into the wall is so small that very little energy is going in the transverse directions. The waves are still dominantly in the +z and –z directions.

You may ask how we can avoid non-passive s-parameters. From our experience, trying to normalize the s-parameters to a real value (such as 50-ohms) is a good way. Even though it is an approximation based upon the voltage and current, it is normally quite accurate. In most of the situations with non-TEM waves, (D-9) holds quite well if we chooses the right integration path for V. As you know, V is not a unique value when non-TEM waves are concerned.

Some people use the following formulas for V, I, a and b.

$$V = Zc^* \cdot a + Zc \cdot b \quad \text{(D-10)}$$

$$I = (a - b) / \sqrt{Re(Zc)} \quad \text{(D-11)}$$

Precisely, it should also be an approximation. However, we do notice some software packages are using the definitions. On MODUA 11, we also use the (D-10) and (D-11) while earlier version of MODUA uses (D-3) and (D-4). On MGRID 11 and PATTERNVIEW 11 for current visualization and pattern calculation, we still use (D-3) and (D-4). It is hard to say which one is better. We do encounter cases of passive 50-ohm normalized s-parameters becomes non-passive when we use (D-3) and (D-4) for re-normalization. We have not done an extensive investigation on (D-10) and (D-11). We implemented the formulas just for consistency with other packages. From what we see, we should also encounter cases with non-passive s-parameters if we use (D-10) and (D-11) for re-normalization with complex Zc.

When Zc is pure real, (D-3) and (D-4) are consistent with (D-10) and (D-11). It is safe to use pure real Zc for re-normalization.

Appendix E. Building Vias, Probe-Feeds and Solder Balls

Vias, probe-fed structures and solder balls are frequently encountered in multiple circuits. IE3D has implemented easy ways to model conical vias, probe-feed proximity and solder balls in one shot. For all 3 different procedures, we need you to enter a series of vertices with each vertex specifying the (X, Y) location of a via (a probe or a solder ball).

We take the file .\ie3d\samples\for_vias.geo as our example. We have a strip at Z = 300. We would like to build 3 vias at (X, Y) = (100, 100), (500, 100) and (900, 100) from Z = 0 (the ground) to Z = 300. We can enter the 3 vertices as series vertices as if we were entering the vertices for a polygon. We can also import them from an ASCII file using Input->Create and Edit Vertices command. We will use the saved locations in file: .\ie3d\samples\for_vias.txt for it.

1. Building Conical Vias:

Run MGRID and open .\ie3d\samples\for_vias.geo. Select Input->Create and Edit Vertices. Select Import for the vertices saved in: .\ie3d\samples\for_vias.txt file. Keep selecting OK until you get back to the main window of MGRID. You will have the 3 vertices entered along a straight line.

Select Entity->Conical via command. Enter the parameters as shown in Figure E.1. Basically, we want to create 3 conical vias with 6 sides. The radius at Z = 0 is 100 and the radius at Z = 300 is 50. Select OK and MGRID will built the conical vias as shown in Figure E.1. There are different options for it. You can choose Open Cap for via holes instead of Close Cap for vias only. You can enter Number of Segments for Circle = 0 for a rectangular via.

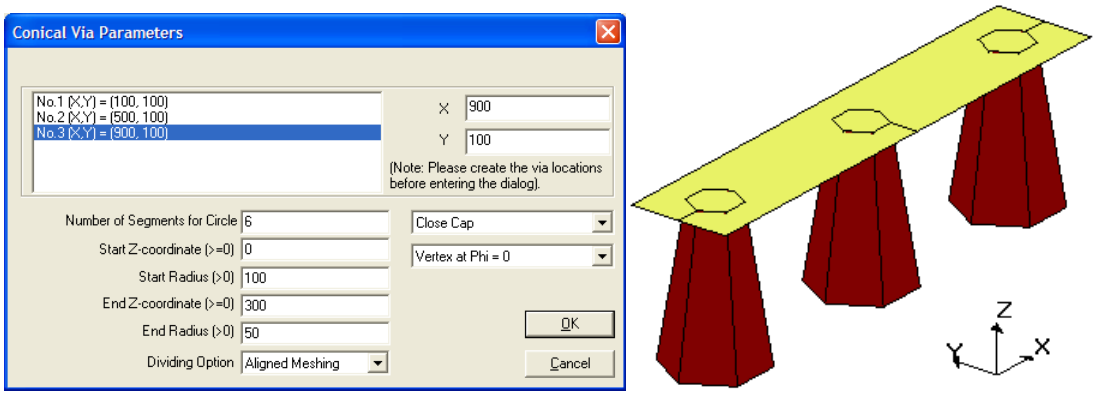


Figure E.1 The Conical via Parameters dialog and the conical via built.

2. Building Probe-Feed Structures:

Assume we want to build a patch fed by 3-probes. We use the file .\ie3d\samples\for_vias.geo as an example. Open it. Select Input->Create and Edit Vertices. Select Import for the vertices saved in: .\ie3d\samples\for_vias.txt file. Keep selecting OK until you get back to the main window of MGRID. You will have the 3 vertices entered along a straight line.

Select Entity->Probe Feed to Patch command. Enter the parameters as shown in Figure E.2. Basically, we want to build 3 square probes from Z = 0 to Z = 300. We want the excitation between Z = 0 (negative terminal) and Z = 3 (positive terminal). The probe's radius is 20 microns. After you select OK, MGRID will build 3 probe feeds for you automatically.

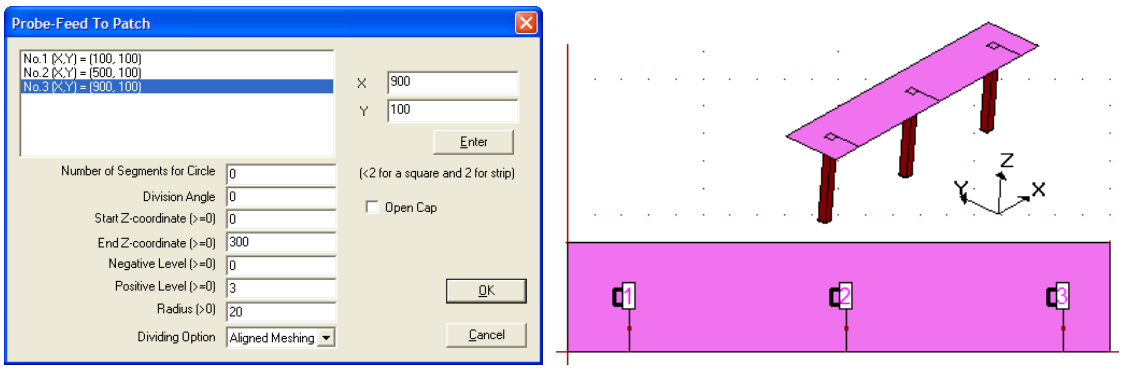
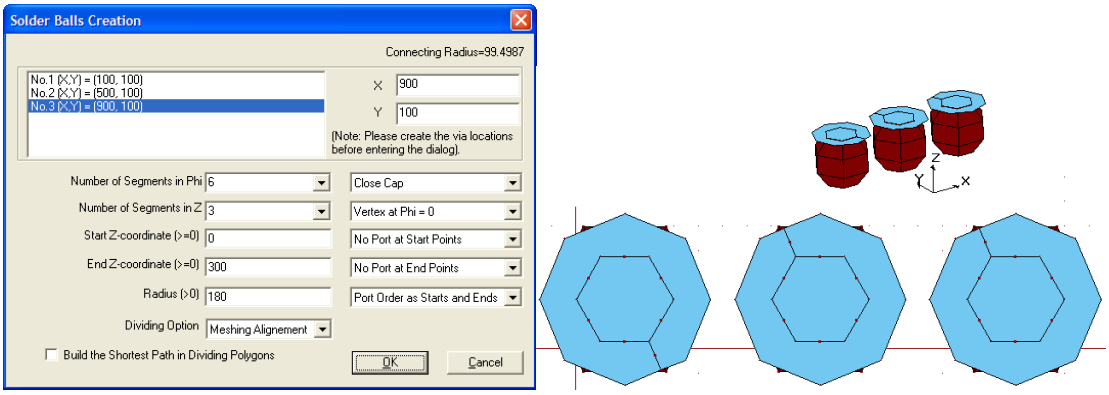


Figure E.2 The Probe-Feed to Patch dialog and the structure built.

3. Building Solder Balls:

Building solder balls is very similar. Normally, we should have the solder ball pads. We use the file .\ie3d\samples\for_solder_balls.geo as an example. Open it. Select Input->Create and Edit Vertices. We can enter a series of vertices across each solder ball pad. Then, we can use Input->Find Centers of Vertex Pairs to find the center locations of the solder ball pads as a series of vertices. We can also Import for the vertices saved in: .\ie3d\samples\for_vias.txt file (discussed in Appendix P for wire bonding). You should have the 3 vertices entered along a straight line.

Select Entity->Solder Balls command. Enter the parameters as shown in Figure E.3. Basically, we want to build 3 solder balls from Z = 0 to Z = 300. The radius of a solder ball is 180. We want to crate the solder ball as 3-segments in Z-direction and 6-segments in Phi direction. We have the option to define ports at the bottom of the solder balls and the top of the solder balls. For this example, we did not specify ports.



Appendix F. Plane-Wave Excitation and Radar Cross-Section

Plane wave incident problems can be solved by the IE3D. You can define a plane wave incident from any angle at the upper half space. To define or change an incident plane wave, you **Port->Plane Wave Excitation** and define the incident angles (theta and phi). They are the elevation angle and azimuth angle for the spherical coordinate system. After you define the plane wave excitation, MGRID will indicate the incident angles in the status bar. Simulation of plane wave excited structures is the same as simulating a circuit structure. IE3D will only create the current distribution data and optionally the radiation pattern. If you define some ports on the structure, IE3D will solve both s-parameters and the current distribution for you.

There are two polarization schemes for the plane wave incident scattering. The two cases are shown in Figure F.1. You are not required to specify which polarization the incident wave when you are in layout mode on MGRID. You can change it the polarization any time in the Simulation Setup or when you are using MGRID as post processor to visualize current distribution and find radiation patterns.

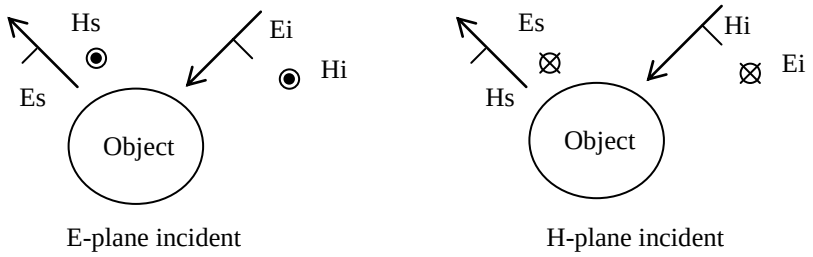


Figure F.1 The E-plane incident and H-plane incident illustration

If you define ports on the structure with plane wave excitation, you can change the port excitation and termination as well as the incident wave polarization and magnitude. You can also change the port excitation and termination such as the parameters of a diode tie to the port in order to get the maximum radiation or the radar cross-section.

Radiation pattern in the full 3D space instead of the upper half space is available to those structures with the lower half space defined as air. Radar cross section is available only when the full 3D radiation pattern is available.

Another important application is to find the Tx/Rx transfer function. For such an application, we require the receiving antenna to have plane-wave excitation and 1-port for the Rx antenna. More discussion can be found in Appendix AJ.

Appendix G. Comparative, Symbolic and Mixed Parameter Optimization

We have demonstrated how to optimize the shapes of the structures for some specified goals. You may have noticed that the goals we have discussed are direct goals. What direct goal means is that the goal is a known constant? There are cases for in-direct goals. Table G.1 shows the differences between direct goals and in-direct goals. Implementing in-direct goals is relatively difficult. For simplicity in implementation, we implement the symbolic goals instead of in-direct goals. Symbolic goals can achieve the same as in-direct goals. Comparative goals can also be used to replace in-direct goals. Symbolic goals can do the same and more. As the implementation in IE3D 7.0, a comparative goal can be defined on maximum 4 parameters. However, a symbolic goal can be defined on as many as you like.

Table G.1 Direct goals, in-direct goals, symbolic goals and comparative goals.

Direct Goal Examples	In-Direct Goal Examples	Symbolic Goals	Comparative Goal
dB[S(1,1)] = -20 dB	N/A	N/A	N/A
dB[S(1,1)] < - 20 dB	N/A	N/A	N/A
N/A	Ang[S(2,1)] = Ang[S(3,1)]	Ang[S(2,1)] = S0 Ang[S(3,1)] = S0	Ang[S(2,1)] / Ang[S(3,1)] = 1 or Ang[S(2,1)]–Ang[S(3,1)]= 0
N/A	dB[S(2,1)] < dB[S(3,1)]	dB[S(2,1)] = S0 dB[S(3,1)] > S0	dB[S(2,1) – dB[S(3,1)] < 0
N/A	dB[S(2,1)] =dB[S(3,1)]+10	DB[S(3,1)] = S0 dB[S(2,1)] = S0 + 10	dB[S(2,1)-dB[S(3,1)] = 10

Let's take the Y-junction in **c:\ie3d\samples\y.geo** as an example. We want to optimize the Ang[S (3, 1)] = Ang[S (2, 1)] by adjusting the length of the port 3 arm (or arm 3). In fact, if we can make the length of the arm 3 to be the same as the arm 2 (the port 2 arm), we should be able to get the Ang[S (2, 1)] = Ang[S (3, 1)]. We just use this simple example to illustrate how we define comparative goals and symbolic goals. We have defined the length of the arm 3 as optimization variable. We are going to define either the comparative goal of Ang[S (2, 1)] - Ang[S (3, 1)] = 0 or the symbolic goal for Ang[S (2, 1)] = Ang[S (3, 1)].

1. Comparative Optimization:

Step 1 Run MGRID and open **c:\ie3d\samples\y.geo**. Select **Set Optimization** in **Process** menu. Enter **Start Freq = 10**, **End Freq = 20**, **Number of Freq = 2**. Select **Enter** to define the two frequencies. Disable the **Adaptive Intelli-Fit**. Select Add button to define the goals. MGRID will prompt you for the goal.

Step 2 Select Ang(S) for the Parameter Type. Select (2, 1) for the 1st Parameter. Select Minus for the Operator. Select (3, 1) for the 2nd Parameter. Select Optimization Quantity = Objective1 for Objective Type. Enter 0 for Objective 1. Select OK to define the goal. Select OK to perform the optimization. The IE3D will be invoked to perform the optimization. It will be finished in tens of seconds and the optimized geometry will be saved in **c:\ie3d\samples\ym.geo**.

2. Symbolic Optimization:

Step 1 Open **c:\ie3d\samples\y.geo**. Save it as **c:\ie3d\samples\y1.geo**. Select Set Optimization in Process menu. All the parameters are still there.

Step 2 Select Remove All to remove the comparative goal we defined earlier. Select Add button and we are going to define the symbolic goals for it. MGRID will prompt for the goal.

Step 3 Select Ang(S) for the Parameter Type. Select (2, 1) for the 1st parameter. Select By Itself for the Operator. Select Optimization Quantity = Symbol + Constant for Objective Type. Enter 0 for the

Constant. Select OK to continue. MGRID will add the symbolic goal into the list. Then, it continues to prompt you for the next goal for the same symbol.

Explanation:

MGRID will continuously prompt you for the next goal for the same symbol until you select Cancel. If you cancel it after you define the 1st symbolic goal, the symbolic goal will have no effect. At least two goals need to be defined for one single symbol if you want the goals to be effective.

Step 4 Change the 1st parameter to (3, 1). Select OK to accept the other parameters. The 2nd goal will be included in the list. MGRID will prompt you for the 3rd goal definition for the same symbol.

Step 5 Select Cancel to stop the symbol. MGRID will list the two goals as shown in Figure G.1. They are on the same symbol S0.

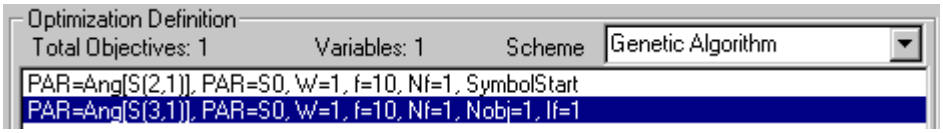


Figure G. 1 The symbolic goals.

The 1st line in the list (or the 1st goal) defines that Ang[S (2, 1)] = S0 + 0. The 2nd line defines that Ang[S (3, 1)] = S0 + 0. Then, we will have Ang[S (2, 1)] = Ang[S (3, 1)].

Step 6 Select OK to start the symbolic optimization. The IE3D will be invoked to perform the optimization. It will also take tens of seconds only.

As it is demonstrated, the comparative optimization and symbolic optimization expand the optimization capability of the IE3D very much.

3. Mixed Parameter Optimization and Symbols Used in Defining Optimization Goals.

The improvement on the IE3D 7.0 optimization capability is not only limited to comparative and symbolic optimization. On the IE3D 7.0, we can also optimize different parameters simultaneously. For examples, we can optimize the s-parameters, z-parameters and y-parameters simultaneously. We can also optimize the directivity, the efficiency and axial ratio of an antenna simultaneously. We cannot illustrate many examples here. Interested users can explore the flexible and powerful optimization capability of the IE3D by themselves. Table G.2 lists the symbols used in optimization.

Table G.2 Symbols used in defining the optimization goals.

Symbol	Description
dB(S)	S-parameter in dB.
Re(S)	Real part of S-parameter
Im(S)	Imaginary part of S-parameter
Mag(S)	Magnitude of S-parameter
Ang(S)	Phase of S-parameter
Re(Zsp)	Real part of Z for S-parameter defined in Appendix A
Im(Zsp)	Imaginary part of Z for S-parameter
Mag(Zsp)	Magnitude of Z for S-parameter
Ang(Zsp)	Phase of Z for S-parameter
Re(Z)	Real part of Z-parameter
Im(Z)	Imaginary part of Z-parameter
Re(Y)	Real part of Y-parameter
Im(Y)	Imaginary part of Y-parameter
VSWR	VSWR of a port.
GAIN (LP)	The antenna gain for the total field in dB
DIR(LP)	The directivity for the total field in dB
Min3dbBW(LP)	The min 3dB beam width for total field
Max3dbBW(LP)	The max 3dB beam width for total field
GAIN (TP)	The antenna gain for the theta field
DIR(TP)	The directivity for the theta field
Min3dbBW(TP)	The min 3dB beam width for theta field
Max3dbBW(TP)	The max 3dB beam width for theta field
GAIN (PP)	The antenna gain for the phi field
DIR(PP)	The directivity for the phi field
Min3dbBW(PP)	The min 3dB beam width for phi field
Max3dbBW(PP)	The max 3dB beam width for phi field
GAIN (LCP)	The antenna gain for the LHCP field
DIR(LCP)	The directivity for the LHCP field
Min3dbBW(LCP)	The min 3dB beam width for LHCP field
Max3dbBW(LCP)	The max 3dB beam width for LHCP field
GAIN (RCP)	The antenna gain for the RHCP field
DIR(P\RCP)	The directivity for the RHCP field
Min3dbBW(RCP)	The min 3dB beam width for RHCP field
Max3dbBW(RCP)	The max 3dB beam width for RHCP field
AxialRatio	The axial ratio of the antenna in dB
AntenEff	The antenna efficiency
RadiatEff	The radiation efficiency
RCS (LP)	The RCS for total field
Norm RCS (LP)	The normalized RCS for total field.
RCS (TP)	The RCS for theta field
Norm RCS (TP)	The normalized RCS for theta field.
RCS (PP)	The RCS for phi field
Norm RCS (PP)	The normalized RCS for phi field.
RCS (LCP)	The RCS for LHCP field
Norm RCS (LCP)	The normalized RCS for LHCP field.
RCS (RCP)	The RCS for RHCP field
Norm RCS (RCP)	The normalized RCS for RHCP field.

Appendix H. Electric, Magnetic and Periodic Walls and Multiple Enclosures

Electric, magnetic and periodic walls are implemented to take advantage of symmetry. Basically, we use images to represent the walls. When there is one wall in the x-direction, we will have one image. The image will give the exact solution to the wall. The same statement applies to the case with the wall in the y-direction. When there are two walls in either the x-direction or the y-direction, there will be infinite number of images. Numerically, we cannot take infinite number of images. You need to provide how many images you want. Usually, we can define the number of images in x-direction or y-direction as 4. It should be able to capture the coupling effect of the walls. Using the image theory, it will be very difficult to capture the resonant effect of the enclosure.

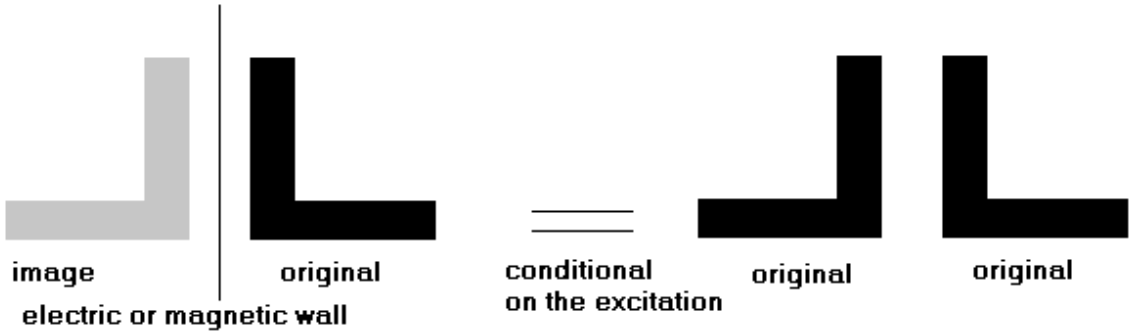


Figure H.1 Absolute equivalent solution with one wall in x-direction.

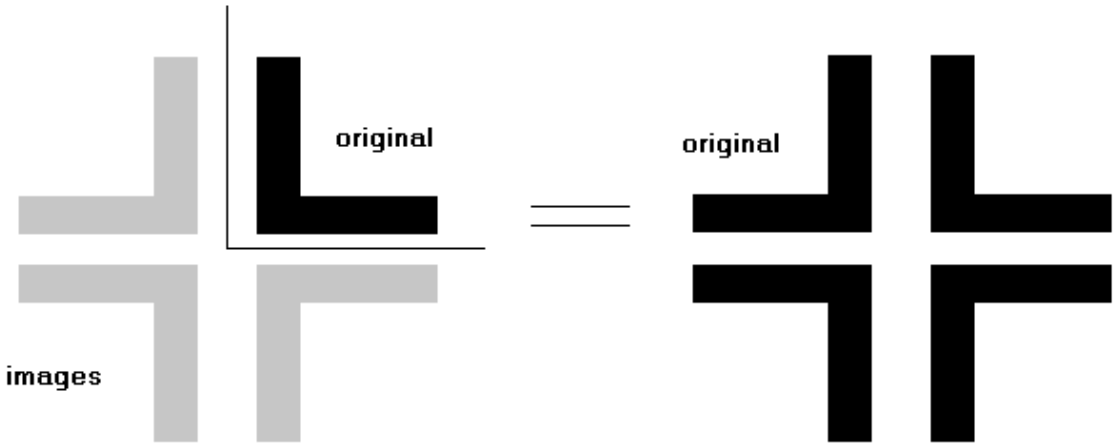


Figure H.2 Absolute equivalent solution with one wall in both x- and y-directions.

To define a wall is very simple. You just select the menu item Enclosure Walls in **Entity** menu. You can define left wall, right wall, top wall, or bottom wall as either electric or magnetic wall. There are some limitations on the wall definition. If you do not define the wall correctly, the IE3D will not simulate the structure.

It is interesting to note that we can simulate a symmetrical structure as half of the structure. It will reduce the size of the problem to half of the original structure as shown in Figure H.5. One comment on the equivalence in Figure H.5 is that the current at the port in Figure H.5A is twice as much as that in Figure H.5B. We need to do a conversion on the s-parameter files in order to get the correct s-parameters. In order to get the result of A from B, we can connect two module B's together on MODUA (see Figure H.5). Another comment on the equivalence is that MGRID will yield the same radiation pattern for both A and B.

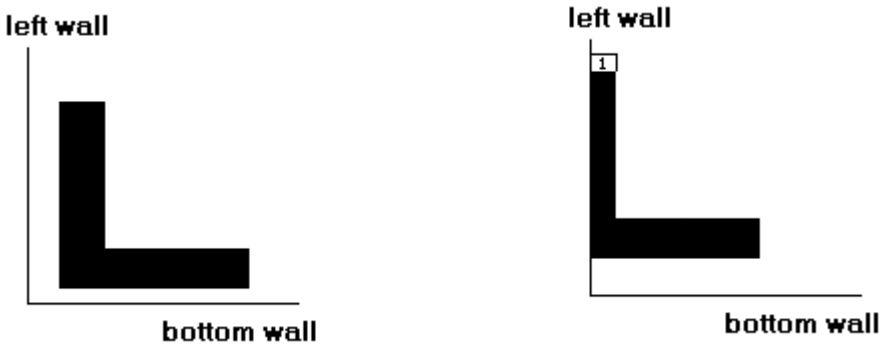


Figure H.3 Structures allowed on IE3D 2.15.

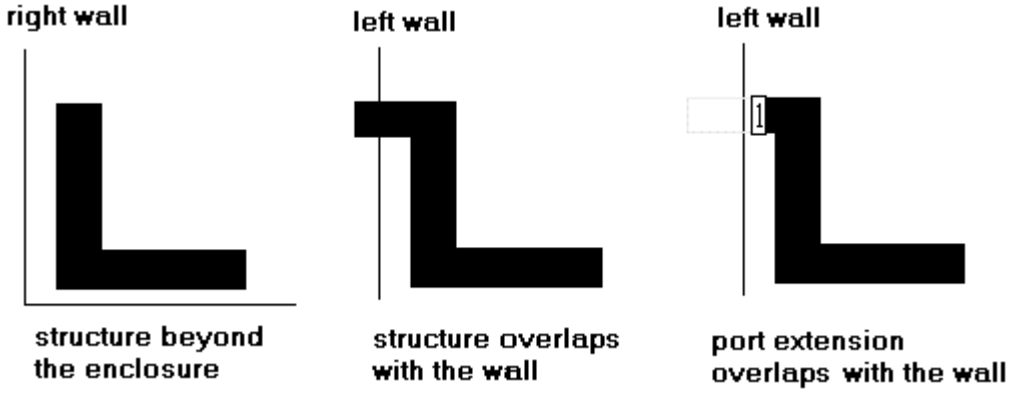


Figure H.4 Structures not allowed.

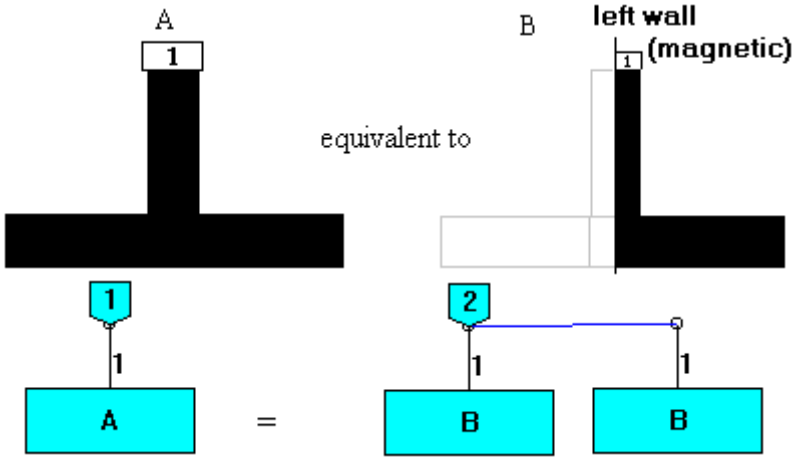


Figure H.5 Symmetrical structure equivalent in solution.

Periodical walls are specially designed for the simulation of large uniform infinite arrays. Periodical walls must come with pair: left and right, or top and bottom. Figure H.6 shows an antenna with left and right periodical walls.

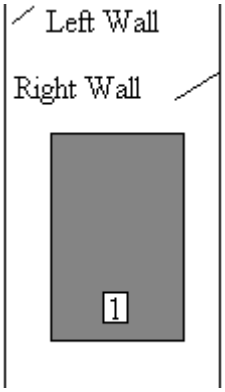


Figure H. 6 A probe-fed patch antenna with left and right periodical walls.

There are 4 additional parameters for the periodical walls: 2 for the periodical walls parallel to the y-axis, and 2 for the periodical walls parallel to the x-axis. For the left and right walls, which are parallel to the y-axis, the 2 additional parameters are the **X-Phase Factor** in degree and the **X-Image Index**.

Assuming we define Number of Images along X = 4, then the structure in Figure H.6 is equivalent to the one in Figure H.7. The **X-Phase Factor** is the excitation phase difference between the Patch 1 and Patch 0. For example, if the phase difference is - 30° and the Patch 0 is of phase of 0, then, the Patch 1 is of phase -30°, Patch 2 is of phase - 60°, Patch 3 is of phase -90° and Patch 4 is of phase -120°.

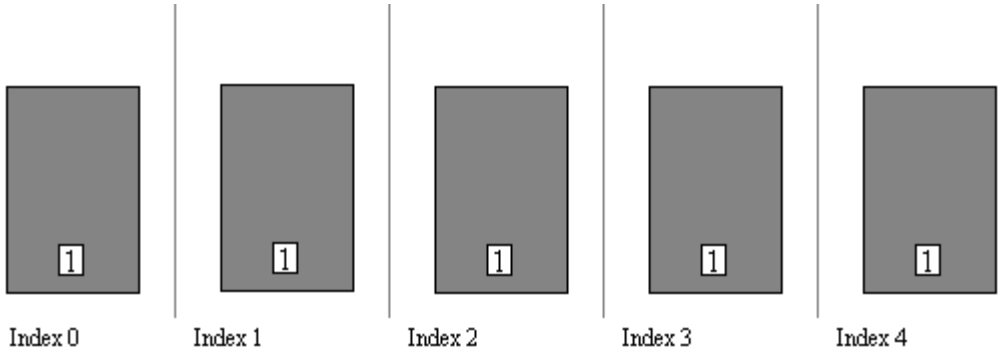


Figure H.7 The equivalent structure to Figure H.6 with 4 images.

In treating the images of the periodical walls, we assume the current distribution is the same except there is a phase difference. Then, the voltage will be different at the port of each patch. So, we introduce the **X-Images Index**. If we specify the **X-Image Index = 0**, we are looking at the input impedance of the Patch 0. If we specify the **X-Image Index** = 2, we are looking at the input impedance of the Patch 2. In this way, we can calculate the input impedance of a patch in a large uniform phase array, no matter whether the patch is inside the array or on the boundary of the array.

Starting from the IE3D 8.0, we have the precise modeling of periodic and electric walls implemented when the 4 walls are periodic or electric. In the simulation, infinite number of images will be included. However, for pattern calculation, a user still needs to enter the **X-Pattern Images** and **Y-Pattern Images**. They control how many images are used in the pattern calculation.

Appendix I. One-Step Construction of Complicated Structures

Some pre-defined structures are built into the MGRID. All these pre-defined structures are created by selecting the menu items in **Entity** menu on MGRID. The pre-defined structures are discussed in this appendix. Shown in the figures are the meanings of the parameters. Please compare the figures to the dialog for better understanding of the parameters. We welcome suggestions on building the Entity library.

1. Rectangle:

   The last entered vertex, if there is any, will be set as the reference point. The reference point can be chosen as the center, lower left corner, lower right corner, upper left corner or upper right corner. Please be careful that the actual reference point location, the length and the width are related to the rotation angle. If the rotation angle is 90 degrees, the Length will become Width, and the lower left corner of the 90 degree rotation is the lower right corner of the 0 degree rotation.

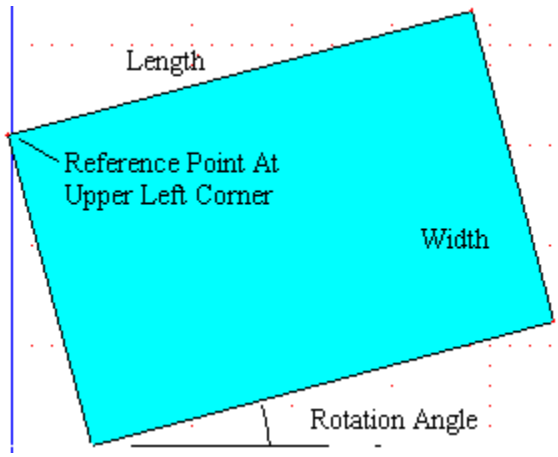


Figure I.1 The Rectangle entity and its parameters.

2. Symmetrical T-Junction:

   The last entered vertex, if there is any, will be set as the center. A user can change the Vertical Arm's Length and Width, Horizontal Arm's Length and Width. The Chamfer Size takes the value between [0, 1).

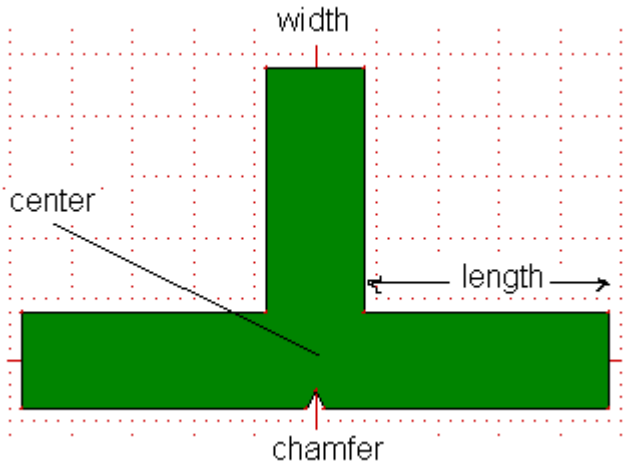


Figure I.2 The Symmetrical T-Junction entity and its parameters.

3. Symmetrical Y-Junction:
   The last entered vertex, if there is any, will be set as the center. A user can choose the Arm Length from Center and Arm Width. The Start Angle defines the angle of the 1st arm.

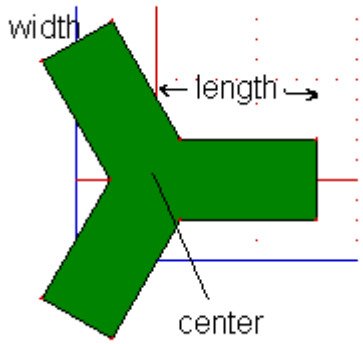


Figure I.3 The Symmetrical Y-Junction entity and its parameters.

4. Symmetrical Step:

   The last entered vertex, if there is any, will be set as the center.

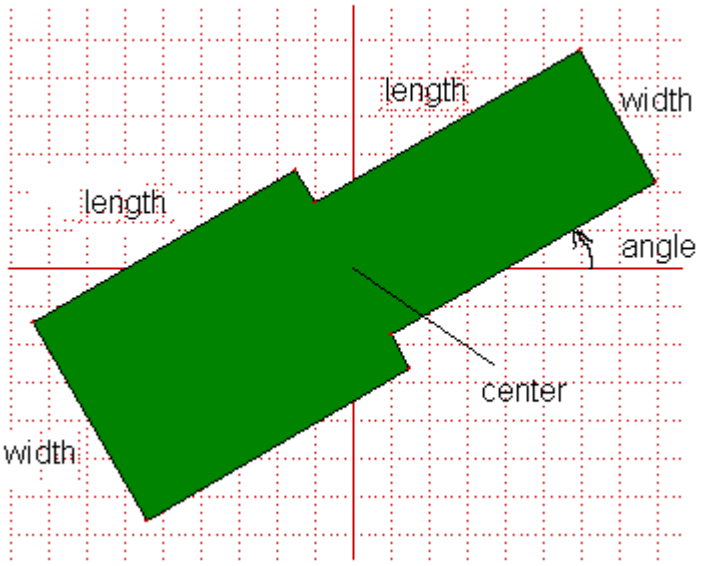


Figure I.4 The Symmetrical Step entity and its parameters.

5. Chamfered Bend:

   The last entered vertex, if there is any, will be set as the center, or the inner corner of the bend. A user can define the length and width of the arms. The Rotation Angle defines the 1st arm's angle. The Chamfer Size takes value from 0 to 1. When it is 0, there is not chamfer. When it is 1, the smallest width for the bend is 0.

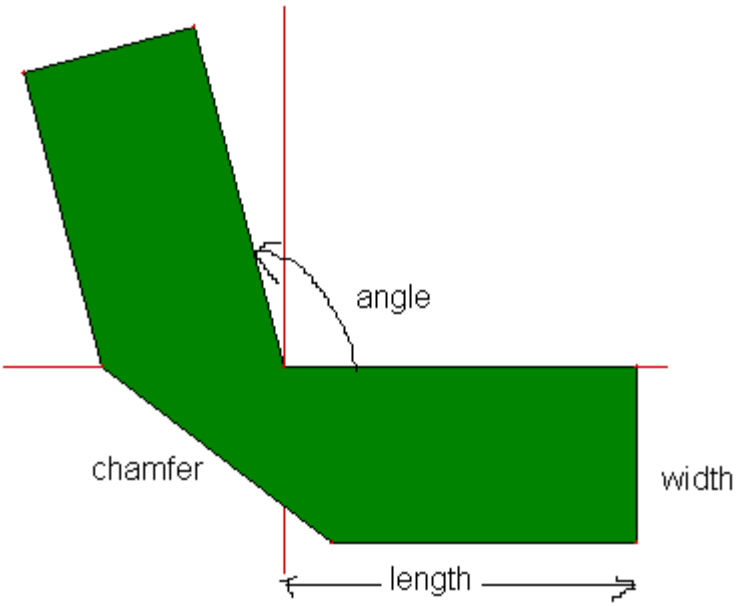


Figure I.5 The Chamfered Bend entity and its parameters.

6. Circle:

The last entered vertex, if there is any, will be set as the center. The starting point is not measured in degrees. It is measured in segments. For example, if we choose the Number of Segments for Circle = 16, each segment corresponds to 90/4 = 22.5 degrees. If we specify the starting point as 1.5, then the actual angle is 22.5 by 3 / 2 = 33.75 degrees.

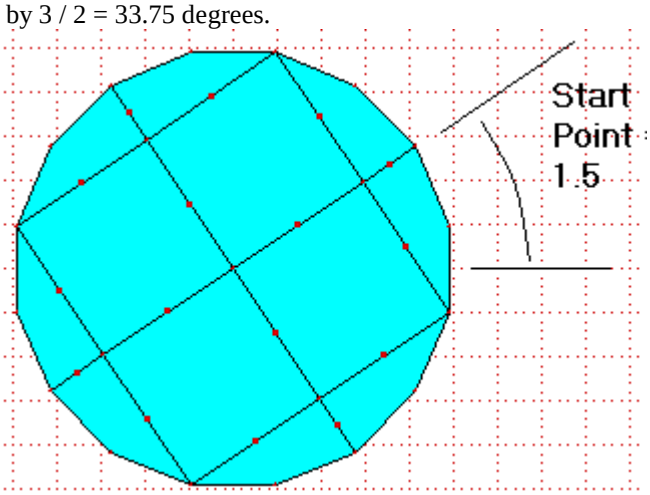


Figure I.6 the Circle entity and its parameters.

7. Circular Sector:

The last entered vertex, if there is any, will be set as the center of a corresponding circle.

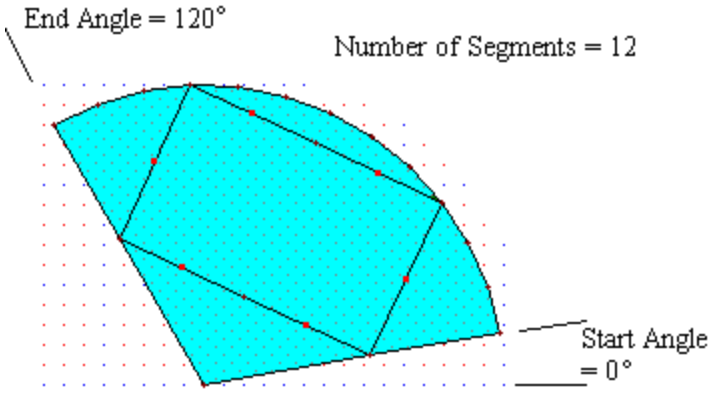


Figure I.7 The Circular Sector entity and its parameters

8. Ring:

   The last entered vertex, if there is any, will be set as the center of a corresponding circle. Also there are options for slightly different shapes.

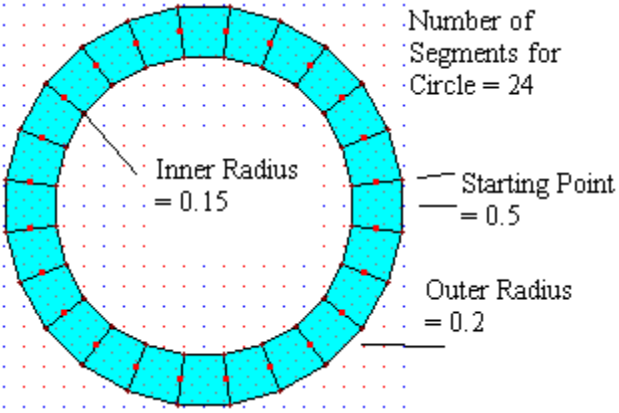


Figure I.8 The Ring entity and its parameters.

9. Annular Sector:

   The last entered vertex, if there is any, will be set as the center of a corresponding circle.

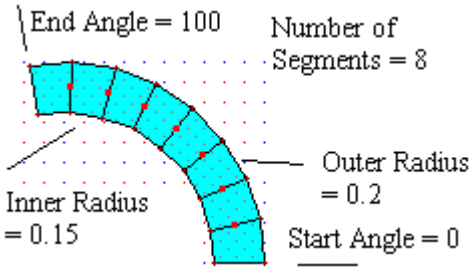


Figure I.9 The Annular Sector entity and its parameters.

10. Conical Tube:

    The center parameters in the dialog are the parameters of the center of the initial point. The cross-section of the initial point is a circle and the starting point is of the same definition of the circle entity.

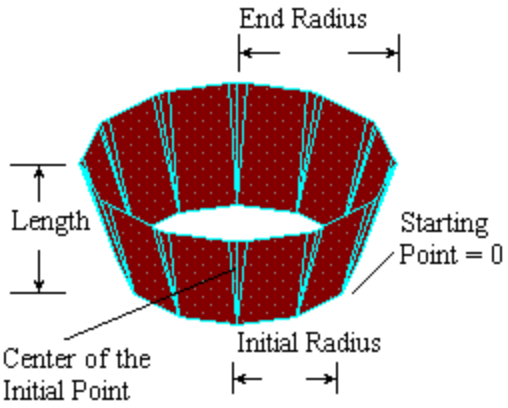


Figure I.10 Conical Tube entity and its parameters.

11. Tube Bend:

Following are the definitions of the parameters:

Segments for Tube Circle:	The number of segments for the tube only. It is not for the bend.
Tube Radius:	The radius of the tube.
Starting Point:	The starting point of the tube.
Center Coordinates:	That is the center of the bend.
Section Index:	The location of the 90-degree bend in the quadrant. Section Index = 0 means it is in the first quadrant, and so on.
Segments for Bend:	The number of segments for the 90-degree bend.
Bend Radius:	The radius of the tube bend.

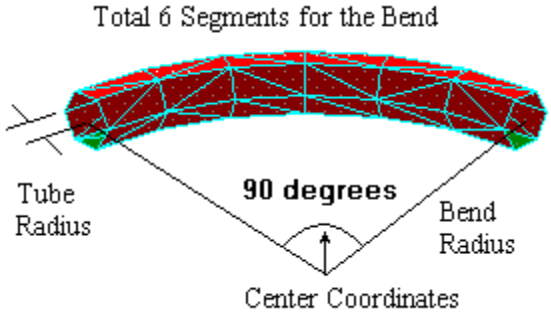


Figure I.11 The Tube Bend and its parameters.

12. Sphere:

The Starting Point is for the horizontal segments only. For the other direction, the segment always starts from the axis direction.

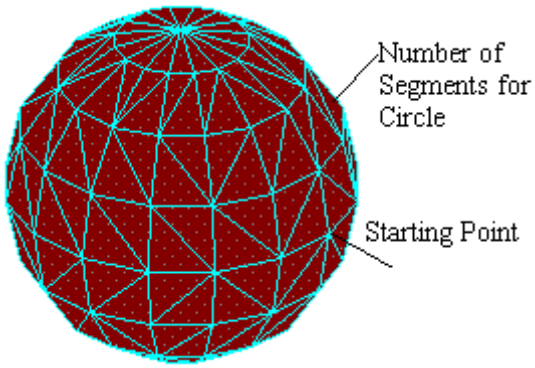


Figure I.12 Sphere entity and its parameters.

13. Cylindrical Helix:

    There are 2 dialogs for it. The first dialog is for the wire parameters: the segments for the wire circle and the radius of the wire. The center coordinates are for the starting point.

    The second dialog is for the helix parameters: the segments for each turn, the total segments, helix length, helix radius, start vertical wire length, end vertical wire length, and start angle. The start angle is for the connection between the end of the start vertical wire and the start of the first segment of the helix. The angle of the connection is the start angle.

    It should be pointed out that the total segments could take negative number. If it is a positive integer, the helix turns in the counter clockwise direction. If it is a negative integer, the helix turns in the clockwise direction.

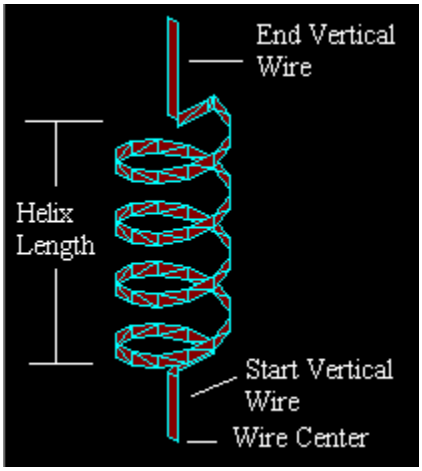


Figure I.13 Cylindrical Helix entity and its parameters.

14. Conical Helix:

    There are 2 dialogs for it. The first dialog is for the wire parameters: the segments for the wire circle and the radius of the wire. The center coordinates are for the starting point.

    The second dialog is for the helix parameters: the segments for each turn, the total segments, exponential constant, initial radius, conical angle, feed line length and start angle.

The cylindrical radius at each angle is R = R0 EXP[A (φ - φ_0)], where R0 is the initial radius, A is the exponential constant, φ is the angle in gradient, φ_0 is the start angle in gradient.

The start angle is for the connection between the end of the feed line and the start of the first segment of the helix. The angle of the connection is the start angle.

The total segments can take negative number. If it is a positive integer, the helix turns in the counter clockwise direction. If it is a negative integer, the helix turns in the clockwise direction.

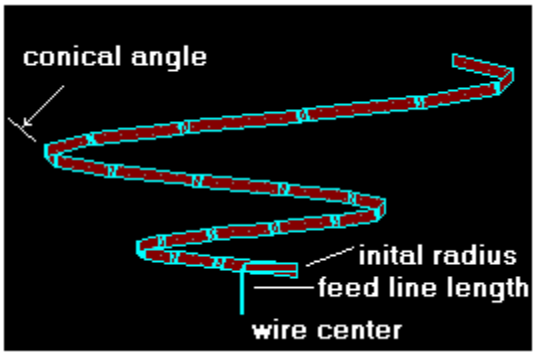


Figure I.14 Conical helix and its parameters.

15. Circular Hole in Square:

The Circular Hole in Square does not dig a circular hole on a polygon. It just builds the shape of a circular hole in a square. To build a circular or arbitrary hole on an existing polygon, you need to use the **Dig Hole from Selected Polygons** in **Adv Edit** menu.

The last entered vertex, if there is any, will be set as the center of the circle.

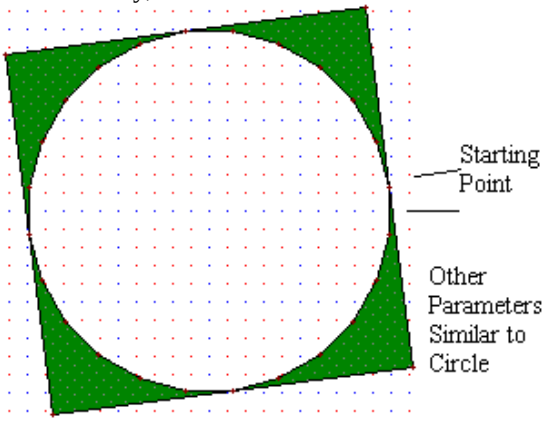


Figure I.15 Circular Hole in Square entity and its parameters.

16. Circular Hole in Rectangle:

The Circular Hole in Rectangle also does not dig a circular hole on a polygon. It just builds the shape of a circular hole in a rectangle. To build a circular or arbitrary hole on an existing polygon, you need to use the **Dig Hole from Selected Polygons** in **Adv Edit** menu.

The last entered vertex, if there is any, will be set as the center of the circle. The parameters are similar to the circle. The circle and the rectangle are always co-centered. The additional parameters are the rectangle length and width.

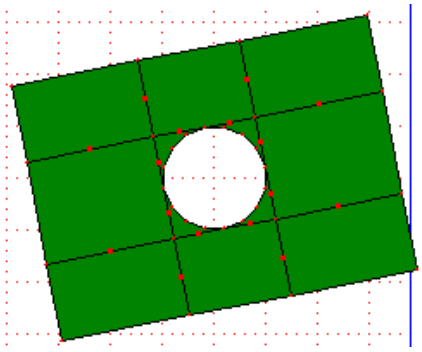


Figure I.16 The Circular Hole in Rectangle entity and its parameters.

17. Rectangular Spiral:

    Each quarter is a rectangular corner. 4 quarters form a turn. Starting point index is to identify where the starting point is. When the starting point index is 0, the starting angle is at φ = 0 degree. When the starting point index is 1, the starting angle is at φ = 90 degrees. The Number of Quarters can take negative value for opposite turning direction.

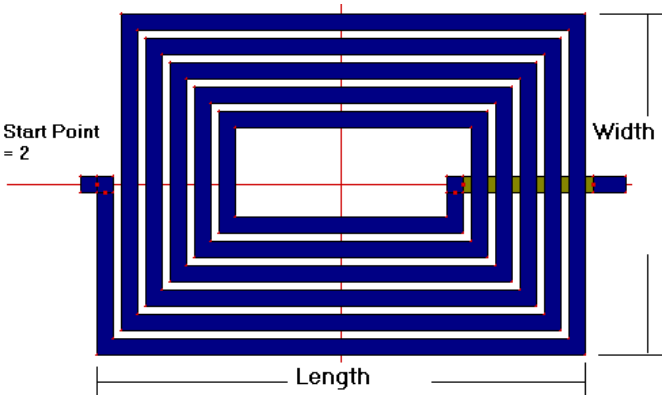


Figure I.17 Rectangular Spiral entity with starting point index of 1, and 18 quarters.

18. Octagonal Spiral:

    The last entered vertex, if there is any, will be set as the center. The size of the spiral is decided by the number of quarters. The starting point is similarly defined as the rectangular spiral.

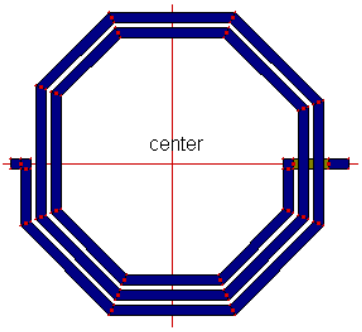
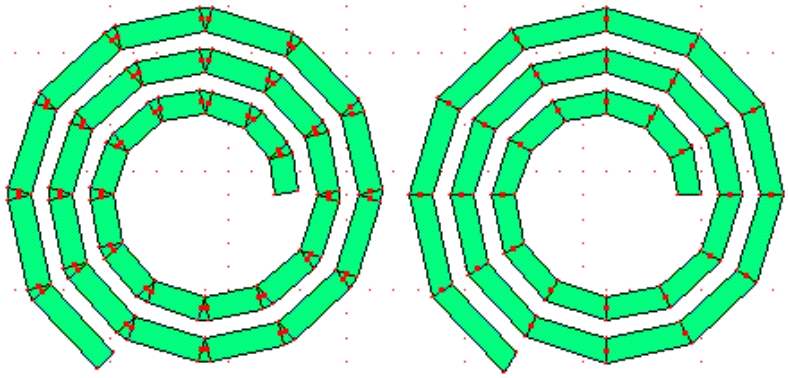


Figure I.18 The Octagonal Spiral entity and its parameters.

19. Circular Spiral:

There are a few options for the circular spiral. The starting point can be From Inner Radius or From Outer Radius. The corner style can be smooth change or abrupt change. The last option is whether you want to guarantee the trace width or the vertex location. Because we approximate the circular spiral with segments, we can guarantee either the trace and gap widths or that the locations of the vertices are on the spiral.



Smooth Change Corner Style Abrupt Change Corner Style

Figure I.19 Circular Spiral entity and its parameters.

20. Interdigital Capacitor:

The last entered vertex, if there is any, will be set as the center.

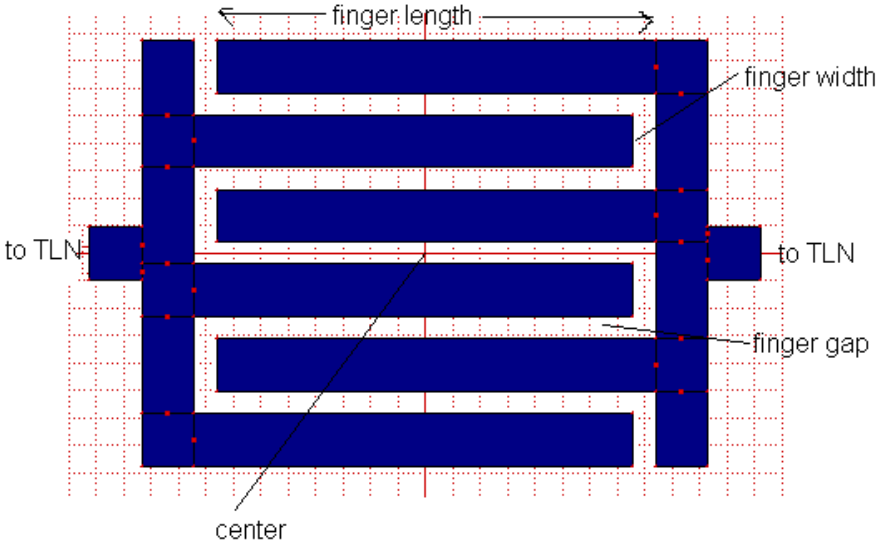


Figure I.20 The Interdigital Capacitor entity and its parameters.

21. MIM Capacitor:

The last entered vertex, if there is any, will be set as the center. The meshing between the top and bottom plates is aligned. Also, the user has the option to create a circular or rectangular via on the bottom plate.

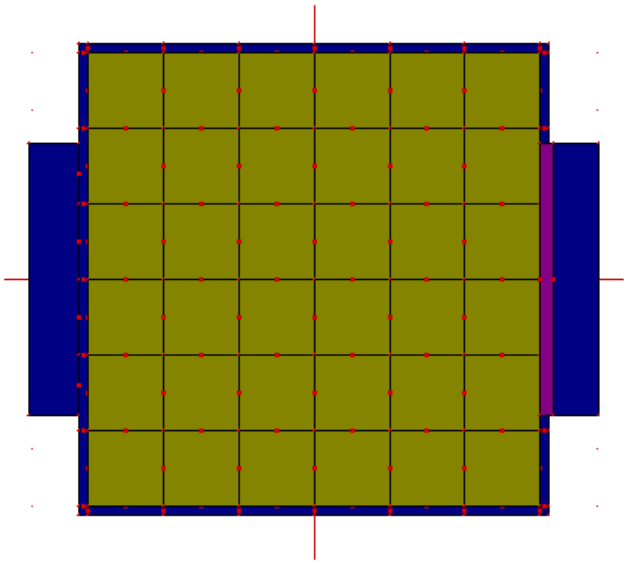


Figure I.21 The MIM Capacitor entity and its parameters.

22. Probe-Feed to Patch:

    It is an automatic way to build a **Vertical Localized** port. It is very useful for constructing probe-fed patch antennas. However, its application is more than probe-fed patch antennas. The probe will be modeled at least as a rectangular tube (with Number of Segments for Quarter = 1). You should make sure there is metal on the start z-coordinate and the end z-coordinate. Metal means that it must be some polygons representing metallic structures or a ground plane defined as substrate. The Positive Level and the Negative Level can be at the start z-coordinate or end z-coordinate. The Positive Level should be different from the Negative Level. However, the difference should be very small compared to the waveguide wavelength. For probe-fed patch antenna, it is suggested that the Negative Level should be located at the Start Z-coordinate and the difference between the Positive Level and Negative Level should be about 1% of the probe length. The Positive Level can also be smaller than the Negative Level.

    The last entered vertex, if there is any, will be set as the location of the probe.

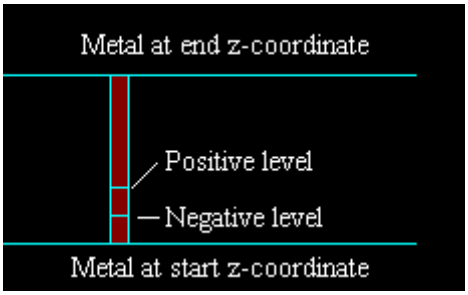


Figure I. 22 Probe-Feed to Patch entity and its parameters.

23. Conical Via:

    The Conical Via is similar to the Conical Tube entity. The difference is that Conical Via will make the electrical connection between the conical tube and the metal at the Start Z-Coordinate and End Z-Coordinate, if there is any, as it does for the Probe-Feed to Patch entity.

If you enter 1 vertex before you select the Conical Via menu item, the location of the vertex will be used for the center of the conical via end. If you entered more than 1 vertex before you select the Conical Via menu item, the location of the second last vertex will be used for the default center of the conical via end. The distance between the second last vertex and the first last vertex will be used for the default value of the radius.

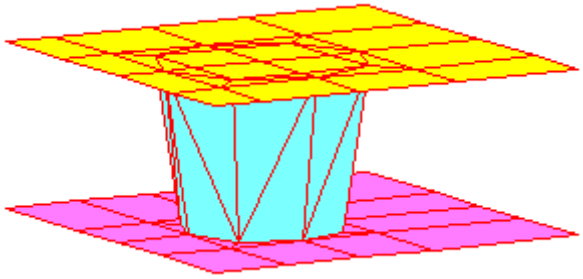


Figure I. 23 The Conical via entity and its parameters.

Many of the entities can be constructed in either of the x, y and z-direction. It should be pointed out that the there may not be an overlapping checking for the pre-defined structures. Please perform an overlapping checking or cutting overlapping polygons.

Documented in this appendix may not a complete list of the entity objects in the software. If you have any suggestion on any entity you would like to build into MGRID, please feel free to report to us. We will try our best to construct it for your convenience.

Appendix J. Simulation of Coaxial Structures

In the following, we will demonstrate the simulation of a coaxial step structure as shown in Figure J.1. Basically, we need to create 4 entities: one cylindrical tube for the inner conductor, one cylindrical tube for the smaller outer conductor, one cylindrical tube for the larger outer conductor and one ring for joining the two outer conductors.

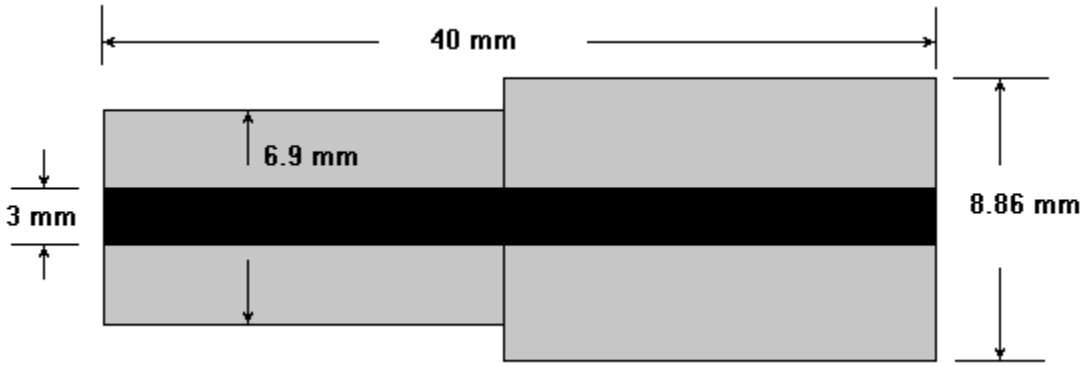


Figure J.1. The longitudinal cross-sectional view of the coaxial step structure.

First we define the basic parameters:

unit length = mm.
grid size = 1 mm
substrate parameters (free space):
No.0 layer, z=0, permittivity = 1, permeability = 1, conductivity = 0
No.1 layer, z=1.0e+10, permittivity = 1, permeability = 1, conductivity = 0
metallic strip parameters:
No.1 type, thickness = 0.01 mm, conductivity = 4.9e+7 (s/m)
discretization parameters:
highest frequency = 1 GHz, Number of cells per wavelength = 20

The second step is to create the inner conductor:

Select Conical Tube in Entity menu and define the following parameters:

x-direction for the tube
number of segments = 12
starting point = 0.5
center x-coordinate = 0 mm
center y-coordinate = 20 mm
center z-coordinate = 10 mm
length = 50 mm
initial radius = 3 mm
end radius = 3 mm

We define the inner conductor to be longer so we can define the port on it without mixing the edges with those on the outer conductor. We select x-direction because we want the coaxial to be on the x-axis. We define the starting point = 0.5 because we want to have at least one polygon to be horizontal. The **Extension for MMIC** de-embedding scheme requires at least one edge to be from a horizontal polygon. Figure J.2 shows the top view of the cylindrical tube on MGRID. The tube is from x = 0 mm to x = 50 mm. If you do not get similar view, you may have entered something wrong. For example, you may see a circle if you forgot to change the default z-direction to the x-

direction. If you define the start point = 0 instead of 0.5, there will be no horizontal polygon. Whenever something is wrong, you can select **Undo** in **Edit** menu to recover.

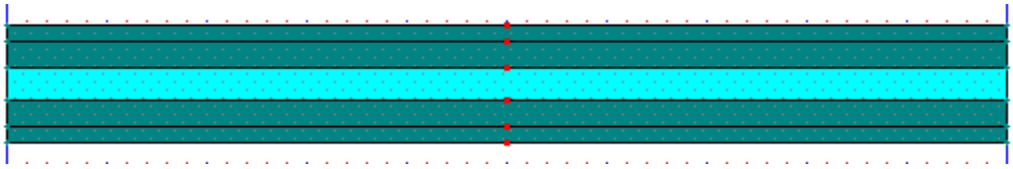


Figure J.2 The inner conductor (the center strip with different color is the horizontal polygon).

The third step is to create the smaller outer conductor:

Select Conical Tube in Entity menu and define the following parameters:

x-direction for the tube
number of segments = 12
starting point = 0.5
center x-coordinate = 5 mm
center y-coordinate = 20 mm
center z-coordinate = 10 mm
length = 20 mm
initial radius = 6.9 mm
end radius = 6.9 mm

The result is shown in Figure J.3.

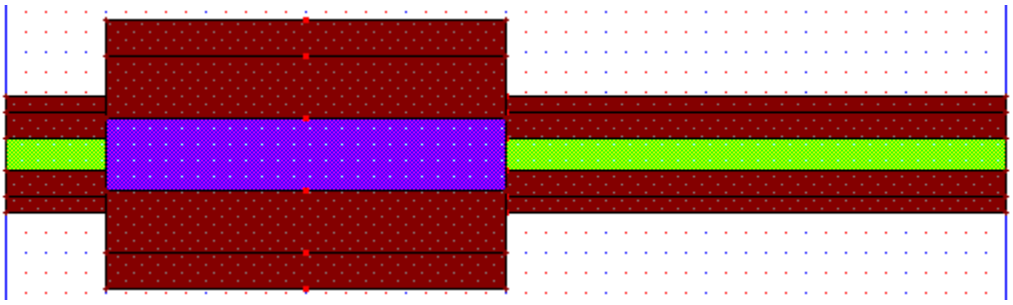


Figure J.3. Top view of the inner conductor and the smaller outer conductor.

The fourth step is to create the larger outer conductor:

Select Conical Tube in Entity menu and define the following parameters:

x-direction for the tube
number of segments = 12
starting point = 0.5
center x-coordinate = 25 mm
center y-coordinate = 20 mm
center z-coordinate = 10 mm
length = 20 mm
initial radius = 8.86 mm
end radius = 8.86 mm

The result is shown in Figure J.4.

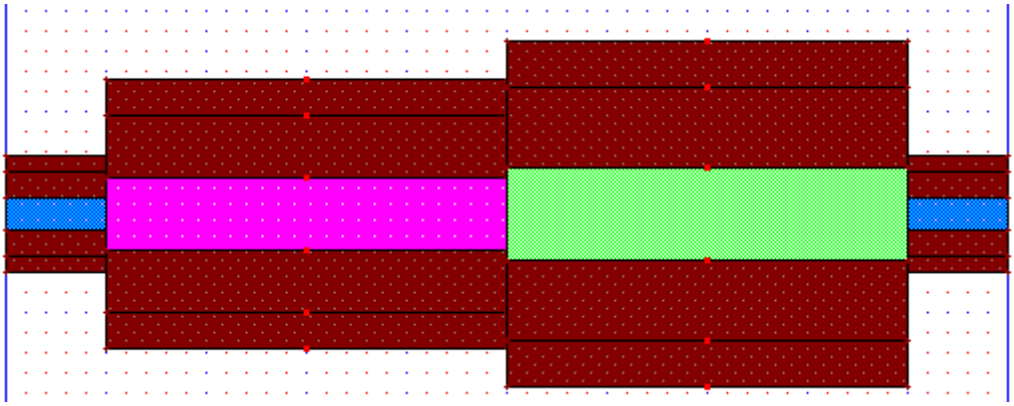


Figure J.4 Top view of the inner and outer conductors.

The fifth step is to create the ring joining the outer conductors:

Select Ring in Entity menu and define the following parameters:

x-direction for the tube
number of segments = 12
starting point = 0.5
center x-coordinate = 25 mm
center y-coordinate = 20 mm
center z-coordinate = 10 mm
inner radius = 6.9 mm
outer radius = 8.86 mm

The result will be similar to Figure J.4 except there are some red dots on the junctions of the outer conductors.

The sixth step is to define the ports:

We must use the Extension for MMIC scheme for coaxial structures. We also need to define the positive and negative ports using the **Port for Edge Group** in **Port** menu. Figure J.5 shows the structure with the positive and negative ports defined.

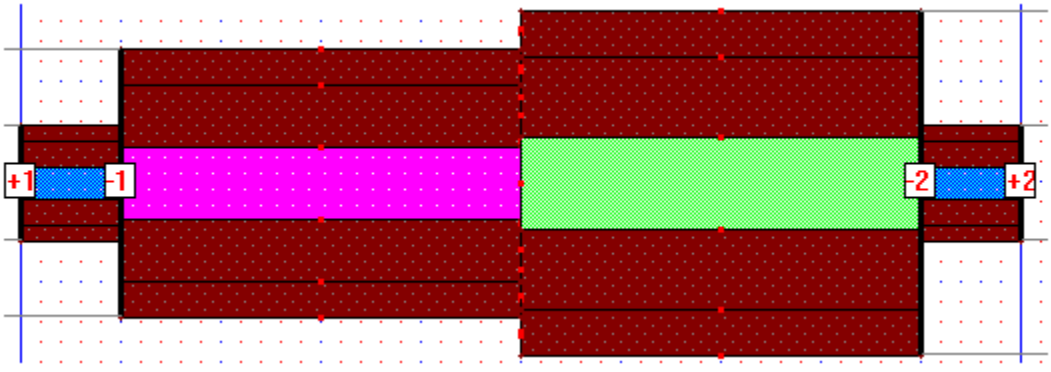


Figure J.5 The structure with the ports defined.

The seventh step is to shorten the inner conductor.

We use the Select Vertices and Move Object commands in Edit menu to move the vertices on both ends of the inner coaxial. The offsets for the left end are dx = 5 mm and dy = 0. The offsets for the right end are dx = -5 mm and dy = 0 mm. The final result is shown in Figure J.6.

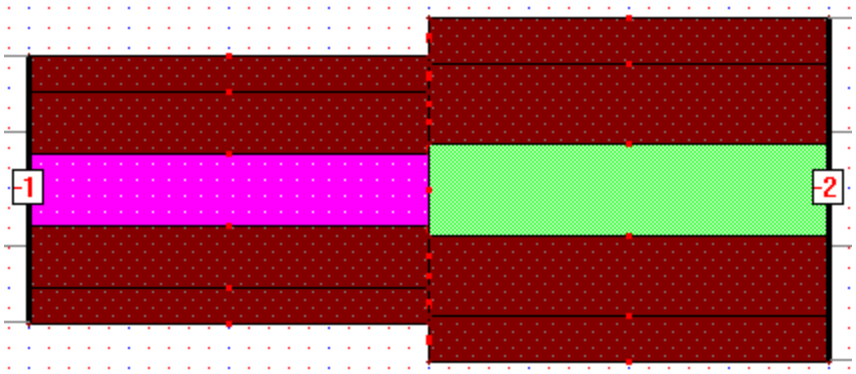


Figure J.6 The structure with the inner conductors aligned with the outer conductors.

The final structure is saved in **c:\ie3d\samples\coaxstep.geo**.

Appendix K. Strategy in Modeling Large Lead Frame and Wire Bond Structures

High-speed digital circuit or RF circuit packages are the most complicated structures to model. They take many polygons. The most computational expensive portions of the structure are the wire bonds. When we use 4-side tube approximation to the wire bonds, a structure with only 10-wire bonds uses more than 3000 unknowns (see .\ie3d\samples\frame.geo discussed in Chapter 9). If we have tens of wire bonds, it will be extremely tough to model. We need to find a good way to simulate such structures.

We can try to use ribbons to model the wire bonds. It is less accurate; however, it will save about 70% of the unknowns on the wire bonds. Normally, ribbon model will predict larger inductance effect. You can consider it as the worst case. For some packages, ribbon bonds are actually used.

The 2nd approach is to simulate parts of the leads first. As you can see from Table 9.7, the coupling coefficient between the 1st lead and the 4th lead is below 0.1 (L14=0.110 vs. L11=1.468). It means that the coupled power from port 1 to port 4 is normally below 1% of the input power. We may be able to neglect them. We can simulate a few adjacent leads once at a time. For example, we can perform 10-reduced size simulations listed in Table K.1. We can find the equivalent circuit for each simulation. Then, we will use the **File->Save Matrix File** command on MODUA to get the parameter values for each sub-circuit. The question is how we can construct the 10 by 10 equivalent circuit based upon the ten equivalent circuits of the 4 by 4 sub-circuit. Such a feature is implemented as the **File->Merge Matrix Files** command on MODUA. You just provide the matrix files for the 10-sub equivalent circuits. MODUA will combine them together to yield the 10-by-10 equivalent circuit for you optimally. You can save the final result as matrix file or as SPICE file. For the 10-lead structure, we may not need to do it because it does take much effort to build and simulate the sub-structures, even though the simulation of each sub-structure may be much faster.

Table K.1 Reduced size simulations.

Simulation	From	To
1	Lead 1	Lead 4
2	Lead 2	Lead 5
3	Lead 3	Lead 6
4	Lead 4	Lead 7
5	Lead 5	Lead 8
6	Lead 6	Lead 9
7	Lead 7	Lead 10
8	Lead 8	Lead 1
9	Lead 9	Lead 2
10	Lead 10	Lead 3

Appendix L. Different Types of Current Distribution Display

On the MGRID, we can display 6-types of current distribution. What are the meanings of the different types? This appendix explains the differences.

At any location, the time-harmonic current density can be described as a complex vector:

$$\boldsymbol{J}(x,y,z) = Jx\ \boldsymbol{x} + Jy\ \boldsymbol{y} + Jz\ \boldsymbol{z} \quad \text{(L-1)}$$

where $\boldsymbol{x}$, $\boldsymbol{y}$ and $\boldsymbol{z}$ are the unit vectors.

$$Jx = Jxr + j\ Jxi \quad \text{(L-2)}$$
$$Jy = Jyr + j\ Jyi \quad \text{(L-3)}$$
$$Jz = Jzr + j\ Jzi \quad \text{(L-4)}$$

Then, we can get,

$$\boldsymbol{J}(x,y,z) = (Jxr\ \boldsymbol{x} + Jyr\ \boldsymbol{y} + Jzr\ \boldsymbol{z}) + j\ (Jxi\ \boldsymbol{x} + Jyi\ \boldsymbol{y} + Jzi\ \boldsymbol{z}) \quad \text{(L-5)}$$
$$\boldsymbol{J}(x,y,z) = Jmr\ \boldsymbol{r} + j\ Jmi\ \boldsymbol{i} \quad \text{(L-6)}$$

where $\boldsymbol{r}$ and $\boldsymbol{i}$ are the unit vectors for the real and imaginary parts, respectively

$$Jmr = \sqrt{ (Jxr^2 + Jyr^2 + Jzr^2) } \quad \text{(L-7)}$$
$$Jmi = \sqrt{ (Jxi^2 + Jyi^2 + Jzi^2) } \quad \text{(L-8)}$$
$$\boldsymbol{r} = (Jxr\ \boldsymbol{x} + Jyr\ \boldsymbol{y} + Jzr\ \boldsymbol{z}) / Jmr \quad \text{(L-9)}$$
$$\boldsymbol{I} = (Jxi\ \boldsymbol{x} + Jyi\ \boldsymbol{y} + Jzi\ \boldsymbol{z}) / Jmr \quad \text{(L-10)}$$

At a specific time, the time-harmonic current density is,

$$\boldsymbol{J}(x,y,z,t) = Re[\ \boldsymbol{J}(x,y,z)\ \exp(j\omega t)\ ] = \boldsymbol{r}\ Jmr\ \cos(\ \omega t\ ) - \boldsymbol{i}\ Jmi\ \sin(\ \omega t) \quad \text{(L-11)}$$

Equation (L-11) is the actual current density at a specific location at a specific time. Clearly, both the value and direction of $\boldsymbol{J}(x,y,z,t)$ are changing with time. Table L-.1 shows the correspondence between the current display functions and the quantities.

Table L-1 Correspondence between the current display functions and the quantities

Display Type	Quantity	Display Features
Average Current Distribution	$\sqrt{ (Jmr^2 + Jmi^2) }$	Shows the average intensity at each location
Vector Current Distribution	$\boldsymbol{J}(x,y,z,t)$	Shows the direction and intensity at specific location and time as vectors on arrows.
Average and Vector Current Distribution	$\sqrt{ (Jmr^2 + Jmi^2) }$ and $\boldsymbol{J}(x,y,z,t)$	Shows the average intensity as color on polygons and direction of current density at a specific time with vectors on arrows.
Scalar Current Distribution Animation	$\lvert\boldsymbol{J}(x,y,z,t)\rvert$	Shows the current density at different locations at different time.
Vector Current Distribution Animation	$\boldsymbol{J}(x,y,z,t)$	Shows the direction and intensity at different locations at different time.
Scalar and Vector Current Distribution Animation	$\lvert\boldsymbol{J}(x,y,z,t)\rvert$ and $\boldsymbol{J}(x,y,z,t)$	Shows the direction and intensity at different locations at different time.

Appendix M. Rectangular Mesh versus Triangular Mesh

IE3D uses a non-uniform, mixed rectangular and triangular meshing scheme. Usually, rectangular cells are efficient for regular shaped portion of a structure. Each rectangular cell is equivalent to 2 triangular cells. Triangular cells are flexible on modeling irregular shaped portion of a structure. It can fit the irregular boundary easily (see Figure M.1). Some people claimed that triangular cells could not yield accurate results because of the zigzag in the meshing. Such a claim is certainly not true, at least for IE3D. In this appendix, we will show that modeling using triangular cells is as accurate as modeling using rectangular cells. We will also demonstrate the efficiency of rectangular cells.

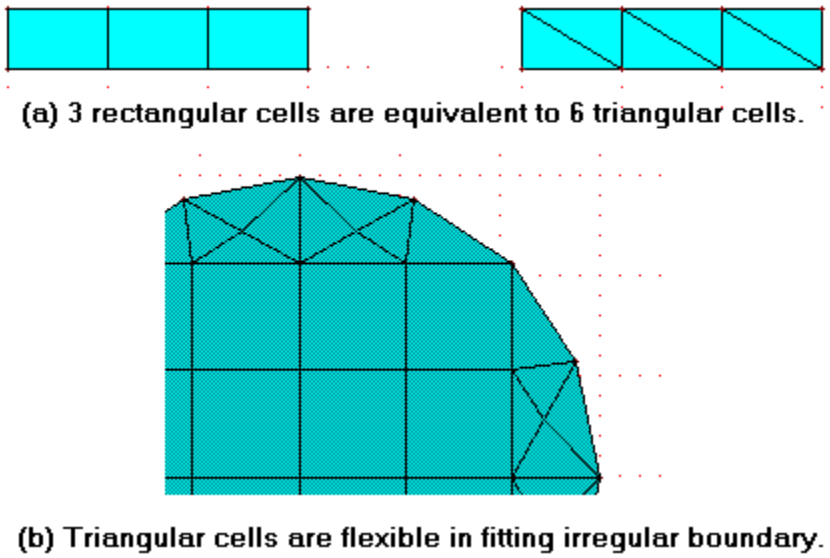


Figure M.1 Comparison between triangular and rectangular cells.

A simple rectangular patch antenna is used as our example. The structure using rectangular mesh is saved in **.\ie3d\samples\rcell.geo** and the structure using triangular mesh is saved in **.\ie3d\samples\tcell.geo**. They are identical except the difference in meshing. The meshed structure is compared in Figure M.2.

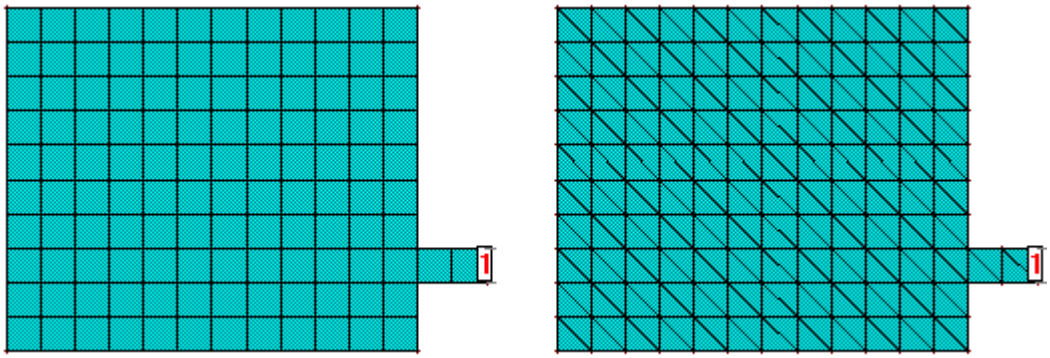


Figure M.2 Meshed structure using rectangular cells (rcell.geo) and triangular cells (tcell.geo).

Table M.1 and Figure M.3 show the comparison between the simulation results using rectangular cells (**.\ie3d\samples\rcell.geo**) and triangular cells (**.\ie3d\samples\tcell.geo**). They are almost identical on the Smith Chart. In fact, there is a slightly frequency shift due to the difference in the meshing. Certainly, the triangular meshing scheme takes more memory and time to simulate the structure.

Table M.1 The comparison between rectangular cells and triangular cells.

	Number of Cells	Number of Unknowns	Memory Required	Simulation Time Per Frequency
Rectangular Mesh (.\ie3d\samples\rcell.geo)	126	225	2 M	0.6 seconds
Triangular Mesh (.\ie3d\samples\tcell.geo)	248	347	4 M	2 seconds

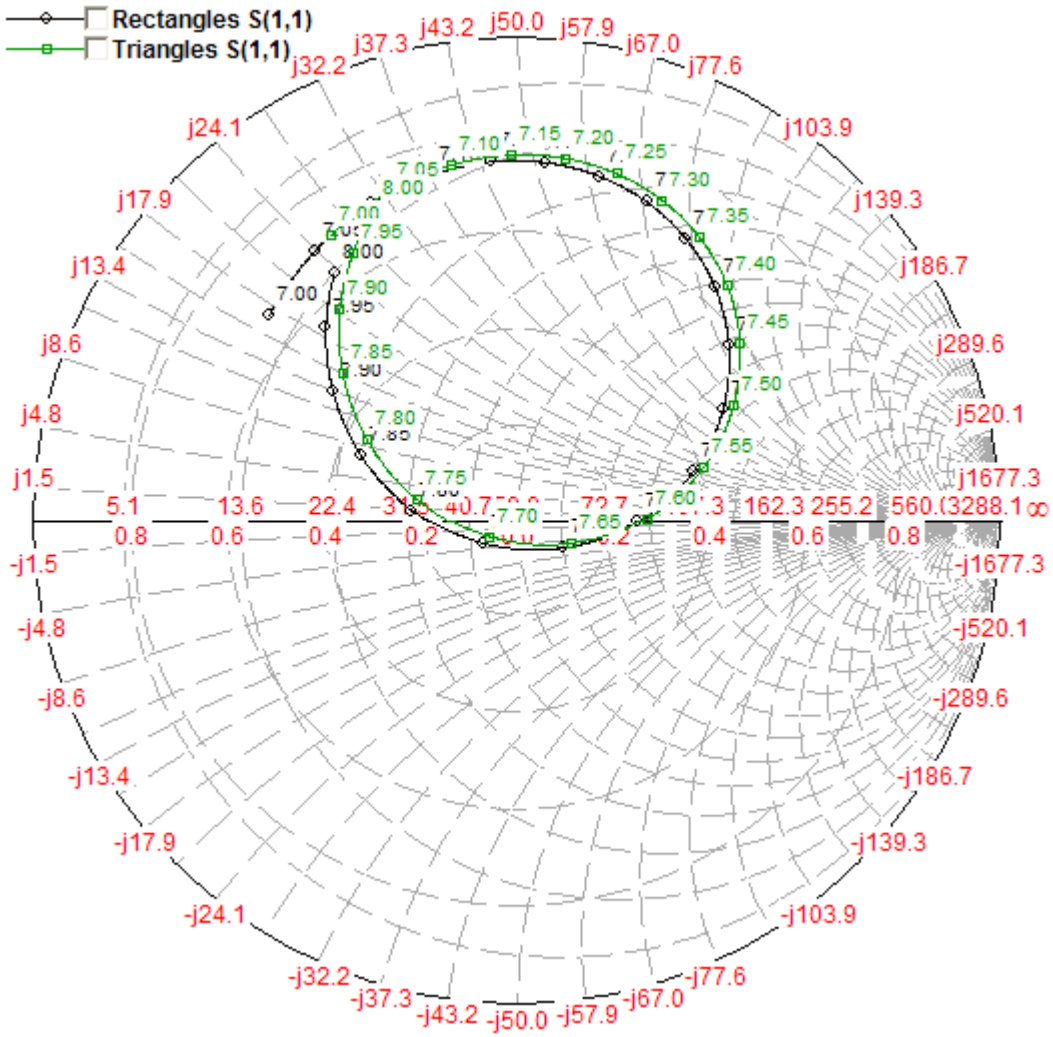


Figure M.3 The comparison between rectangular meshing and triangular meshing.

Appendix N. Uniform Grid versus Non-Uniform Meshing

We have discussed the theoretical comparison between uniform grid and non-uniform grid in Section 2 of Chapter 1. We will provide an actual structure for comparison between using the uniform grid and non-uniform grid. The structure is derived from the geometry saved in **.\ie3d\samples\lpass.geo**. The structure in **lpass.geo** can be best fitted into a uniform grid with grid size of 1 mil. The problem we encountered is that we cannot simulate the structure **lpass.geo** using a uniform grid of size 1 mil on the IE3D with even 256 M RAM without swapping the memory. What we can do is to first shorten the feed-line and then adjust the geometry to fit it into a uniform grid of 2 mils. In such a case, we can solve the problem using about 32 M RAM without swapping. We simulated the structure using the following 3 schemes:

1. Non-Uniform Grid without Edge Cells:
   The structure is saved into **.\ie3d\samples\lpass1.geo** and shown in Figure N.2a.
2. Non-Uniform Grid with Edge Cells for accuracy enhancement:
   The structure is saved into **.\ie3d\samples\lpass2.geo** and shown in Figure N.2b.
3. Uniform Grid Structure:
   The structure is saved into **.\ie3d\samples\lpass3.geo** and shown in Figure N.2c.

Comparison between the three schemes is shown in Table N.1. The simulation result is shown in Figure N.1. It is interesting to note that the results between Non-Uniform Grid with Edge Cells and Uniform Grid Structure compare very well. However, the Non-Uniform Grid with Edge Cells uses much less memory and simulation time to solve the problem. We still get close result without adding the edge cells. Certainly, adding the edge vertices to create small edge cells will improve the simulation accuracy.

It is also interesting to note that the adjusted structure (**lpass1.geo, lpass2.geo** and **lpass3.geo**) yield quite different results from the original structure in **.\ie3d\samples\lpass.geo** especially when the return loss is concern. It is due to the fact that we try to snap the structure into some uniform grids in order to make it easier for the uniform grid based meshing. Our conclusions on the uniform and non-uniform grids are

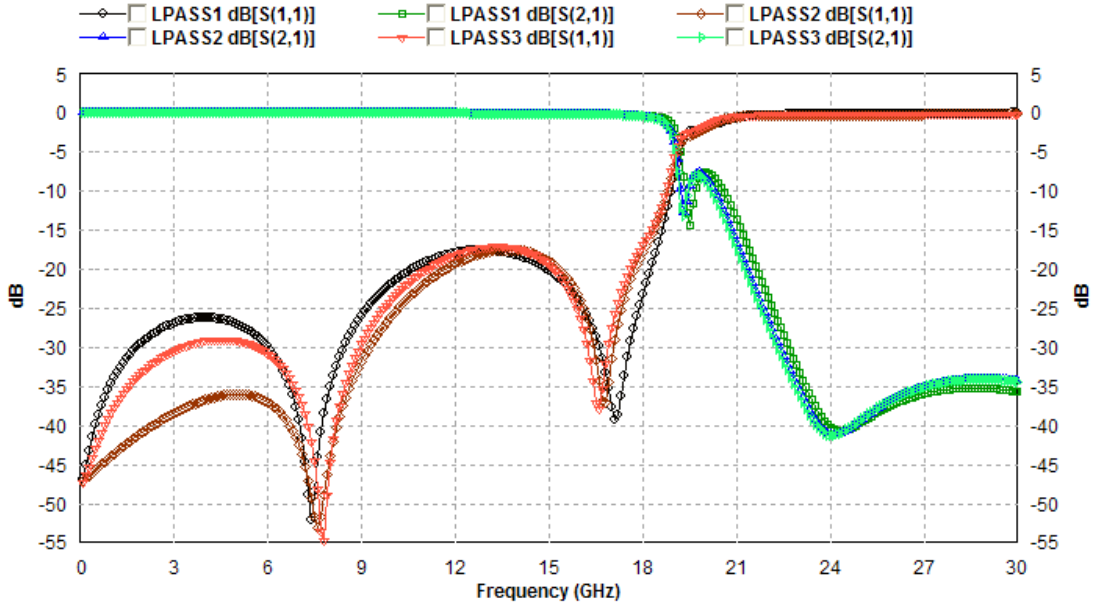


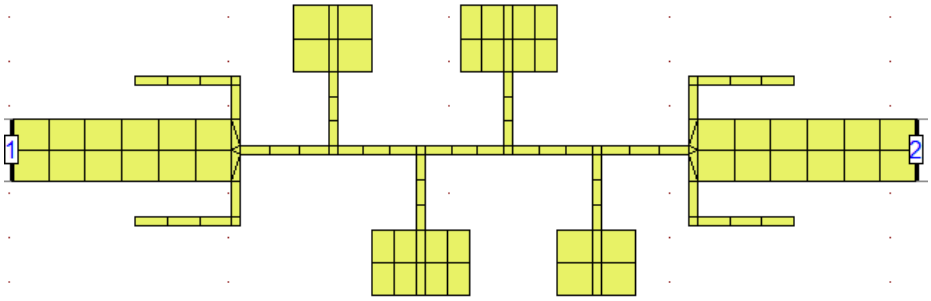
Figure N.1 The comparison of the results from uniform grid, non-uniform grid with or without edge cells.

1. Non-uniform grid scheme is much more efficient than uniform grid scheme.

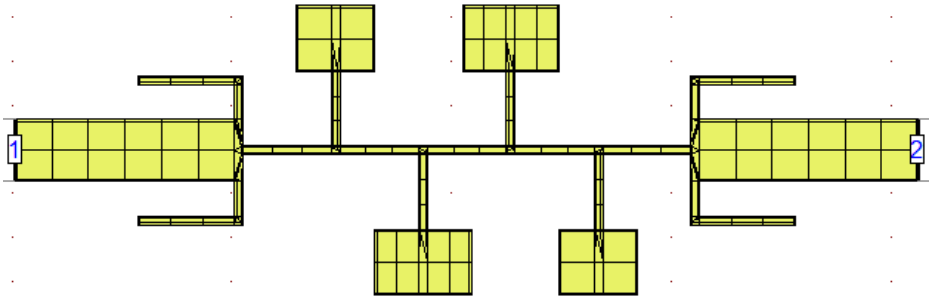
2. Non-uniform grid with edge cells enhancement is at of the same accuracy of the uniform grid scheme when the edge cell size of the non-uniform grid is the same as that for the uniform grid.
3. Non-uniform grid without edge cells enhancement yields reasonably accurate result with extremely high efficiency.
4. Fitting structure into a uniform grid may create significant error.
5. Uniform grid is extremely low efficiency and it is not suitable for large circuit analysis.

Table N.1 Comparison between Uniform Grid and Non-Uniform

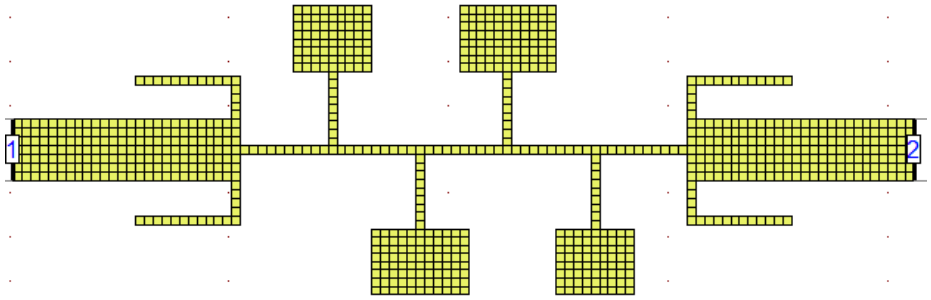
	Number of Cells	Number of Unknowns	Memory Required	Simulation Time Per Frequency
Non-Uniform without AEC (**lpass1.geo**)	141	180	1 M	0.5 seconds
Non-uniform with AEC (**lpass2.geo**)	475	776	5 M	5 seconds
Uniform Grid (**lpass3.geo**)	933	1582	20 M	27 seconds



(a) Non-uniform meshing without AEC.



(b) Non-uniform meshing with AEC.



(c) Uniform meshing without AEC.

Figure N.2 The meshed structure using uniform and non-uniform schemes.

Appendix O. Import and Export of 3D Structures in Text Format

The import and export of IE3D⇔GDSII⇔DXF⇔GERBER⇔ACIS are implemented into ADIX and integrated into MGRID. The GDSII, CIF and GERBER are all 2D formats. DXF is normally 2D format. However, it is possible it can be used to describe 3D objects. Import and export of 3D DXF are supported on ADIX/MGRID. ACIS format from Spatial Technology, Inc is a 3D format. Both 3D DXF and ACIS formats are complicated. On the IE3D, we implemented the simple 3D Text format for easy construction of 3D structure.

The 3D Text format is quite straightforward and it is like a script language. We use the **.\ie3d\samples\wires.3dt** as an example. The file is self-explanatory. To import it, you select **File->Import** command of MGRID. Then, choose the **3D Text** format. You will be prompted for the **Unit** to be used. This will be the unit you will use in the geometry file. The structure defined in the **3D Text** format will have its own unit. No matter what unit you choose on MGRID, it does not affect the size of the structure described in the **3D Text** file. After you go through the process, MGRID will convert the imported structures into polygons.

```
-------------------------start of the file: .\ie3d\samples\wires.3dt----------------------------------------
(3D text import file for IE3D. This is a comment)
(Following are the commands available now: UNIT, POLY[GON])
(TRAC[E], WIRE, BOND[WIRE], ENDO[FTEXT]. The commands are)
(Case insensitive. Any suggestion on additional commands)
(Are welcomed)
UNIT 1
(The above lines define the length unit as: mm)
(1 means mm, 2 means mil and 3 means micron)
(The next line define a 3D polygon with 4 vertices)
(The (x,y,z) of the vertices are listed)
(No comment can be inside the definition block for the polygon)
(There should be at least one space between the POLY and 4).
POLYGON 4
0.00 0.00 0.50
1.00 0.00 0.50
1.00 1.00 0.50
0.00 1.00 0.50
(The next line defines 2D trace. The trace has 6 vertices)
(The trace is at z = 0.5 mm and the width is 0.4 mm)
TRACE 6 0.5 0.4
1.00 0.50
3.00 0.50
4.00 1.50
4.00 4.00
6.00 4.00
6.00 2.00
(The next line defines a 3D wire path. The wire path has)
(4 vertices. The wire path will be built with 2 segments)
(On the wire, or the wire is constructed as a flat strip)
(The radius of the wire path is 0.2 mm. For this case, the)
(Width of the strip is 0.4 mm)
WIRE 4 2 0.2
6.00 2.00 0.50
6.00 0.00 0.50
6.00 -1.0 1.50
```

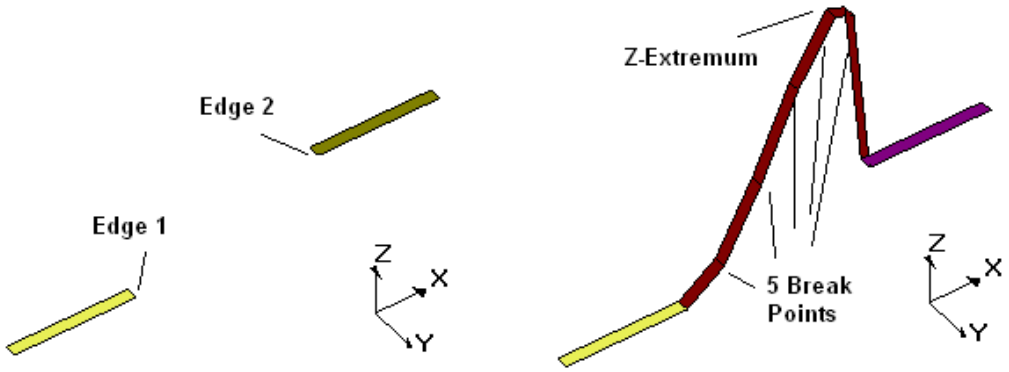
```
6.00 -2.0 1.50
(The next line defines a 3D polygon with 4 vertices)
POLYGON 4
10.0 -2.0 3.00
11.0 -2.0 3.00
11.0 -1.0 3.00
10.0 -1.0 3.00
(The next line defines a 3D wire bond with 6 break points)
(The wire bond will be approximated with 4 sides. The radius)
(Of the wire bond is 0.1 mm. It is of style 1. The style 1 is)
(Non-symmetrical. The other style is style 2 that is symmetrical.)
(The z-maximum is 5.00 mm. The 2 pads will be built at the ends of)
(The wire bond with electrical connection if there is any 2D)
(Polygon at the location)
BONDWIRE 6 4 0.1 1 5.00
6.00 -1.7 1.50
10.5 -1.5 3.00
(The next line defines a 3D polygon with 4 vertices)
POLYGON 4
0.00 2.00 5.00
1.00 2.00 5.00
1.00 3.00 5.00
0.00 3.00 5.00
(The next line defines a 3D wire path. This wire path is)
(Different from the one discussed above. This wire path)
(Will include 2 pads on each end, specified by the 1 and 1)
(You can use any combination of 0 and 1 for the pads)
(If any 2D polygon is defined at the location of the pads)
(MGRID will make the electrical connection. For this case)
(It is a wire path with 4 vertices. The wire is built as a)
(Rectangular tube. The radius of the tube is 0.2 mm)
WIRE 4 4 0.2 1 1
0.50 0.50 0.50
0.50 0.50 3.00
0.50 2.50 3.00
0.50 2.50 5.00
(The next line is end of the text. You can terminate the)
(Text without the ENDOFTEXT)
ENDOFTEXT
----------------------------------------------------end of file: .\ie3d\samples\wires.3dt-------------------------
```

Appendix P. Building Wire Bonds and Solder Balls

We have demonstrated how to build wire bonds in Chapter 9. This appendix documents more detail on wire bonding. Constructing a bond wire using the mouse input is difficult. What we have achieved is to do it automatically. There are multiple ways to build it.

1. Building Ribbon Wire Bonds.

As it is shown in Figure P.1, we want to add a wire bond between Edge 1 and Edge 2 in Figure P.1. What you need to do is to select Edge 1 and Edge 2. Then, you select the **Edit->Wire bonding** command. It will prompt you for the parameters: the **Number of Break Points**, the **Z-Extremum** and **Wire Bond Style** (Symmetric or Non-Symmetric). There is another option **3-Segment (JEDEC)**. It is not available for wire bond built in this way. You can enter a value larger than the maximum Z of Edge 1 and Edge 2 or smaller than the minimum Z of Edge 1 and Edge 2 for the **Z-Extremum**. MGRID will calculate how long the wire bond is after you enter the parameters.



(a) Before adding the wire bond (b) After adding the wire bond

Figure P.1 Wire bonding feature in the MGRID

2. Building Multiple Wire Bonds in one Shot.

The ribbon wire bond feature allows you to build a wire bond between two selected edges. In fact, IE3D can allow you much more flexibility in building multiple wire bonds in one shot. The multiple wire bond feature allows you to build either ribbon wire bonds or solid wire bonds for a number of vertex pairs. Each vertex pair defines the start and end points of a wire bond.

Table P.1 The 6 vertices (3 pairs) defining 3 wire bonds.

Wire Bond	Start Vertex	Start (X, Y, Z)	End Vertex	End (X, Y, Z)
0	1	(160, -50, 100)	2	(1020, 0, 200)
1	3	(160, 100, 100)	4	(1020, 100, 200)
2	5	(160, 250, 100)	6	(1020, 200, 200)

We will demonstrate a 3-wire bond example. The initial structure is shown in Figure P.2 and it is saved in .\ie3d\samples\for_wire_bonds1.geo. There are 3 traces on Z = 100 microns and 3 traces on Z = 200 microns. We would like to build 3 wire bonds between the 3 traces on Z = 100 and the 3 traces on Z = 200. We need to enter the 6 vertices defining the locations of the wire bonds. The vertices are documented in Table P.1. We can use mouse entry, keyboard entry or even the Input->Create and Edit Vertices command to enter the vertices. Actually, the vertices are stored in an ASCII file .\ie3d\samples\for_wire_bonds1.txt. We can go to Input->Create and Edit Vertices dialog and import the vertices from the file.

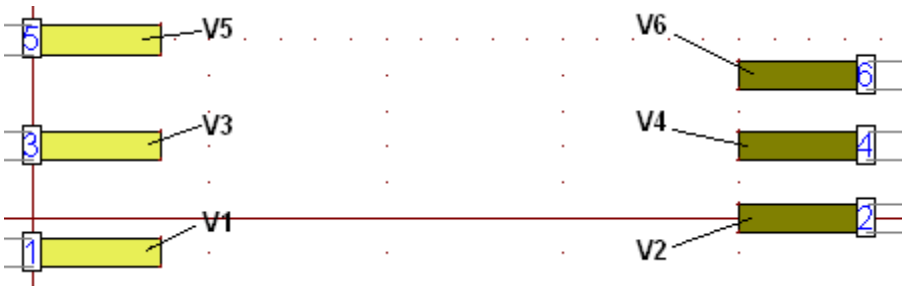


Figure P.2 The 6 traces we want to build 3 wire bonds on.

After we import the vertices into Create and Edit Vertices dialog and exit from the dialog, we will be in the mouse input mode with the 6 vertices entered as if they were entered by mouse or keyboard (see Figure P.3). Select Edit->Wire Bonding command. MGRID will prompt you for the parameters. You can use the parameters to define how the wire bonds will be built. The different parameters are discussed in Table P.2.

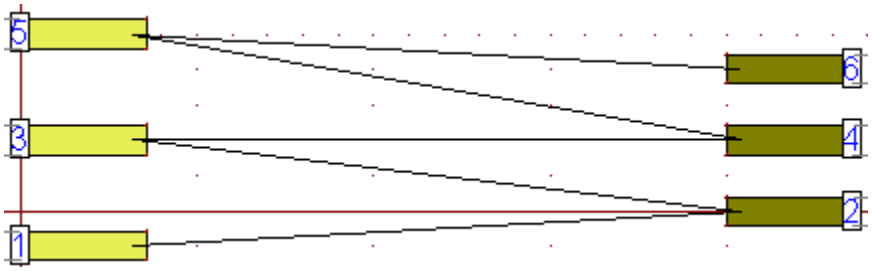


Figure P.3 The structure with the 6 vertices entered.

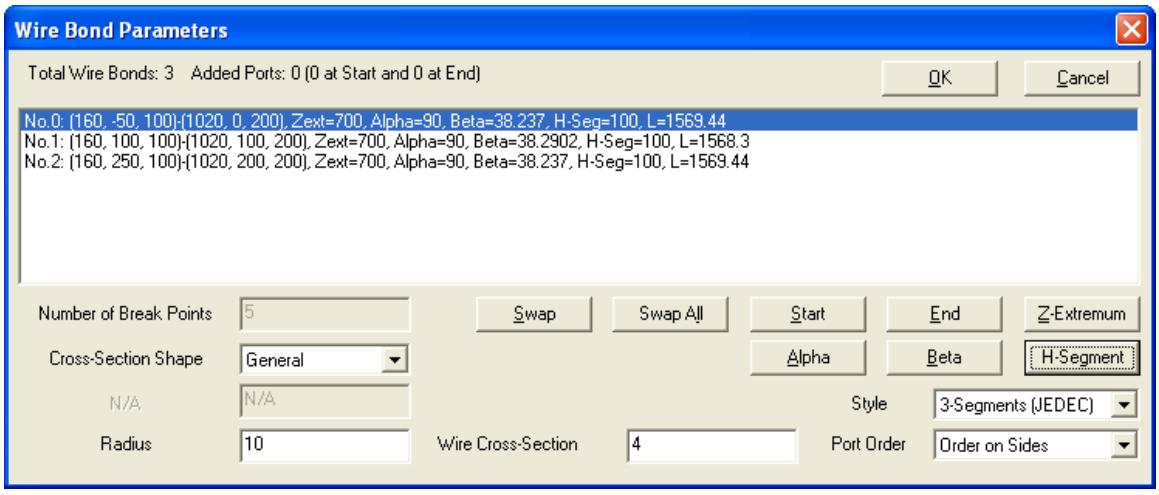


Figure P.4 The Wire Bond Parameters dialog.

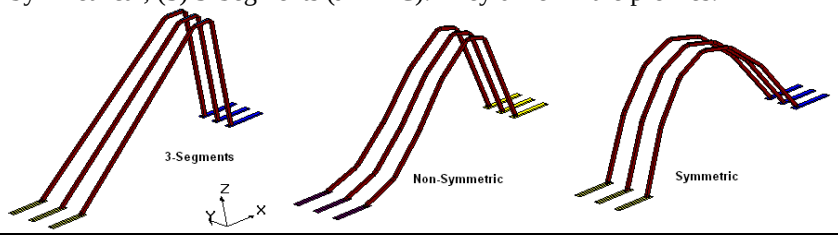
Saved in .\ie3d\samples\for_wire_bonds1_jedec_forward.geo is the wire bond structure with Forward-Directed ports defined on both the Start points and End points. The Forward-Directed port at the Start point of a wire bond means that the positive terminal is on the wire bond and the negative terminal is on the bond pad. For the End point, the positive terminal is on the bond pad and the negative terminal is on the wire bond (see Figure P.5).

Saved in .\ie3d\samples\for_wire_bonds1_jedec_backward.geo is the wire bond structure with Backward-Directed ports defined on both the Start points and End points. The Backward-Directed port at the Start point of a wire bond means that the negative terminal is on the wire bond and the positive terminal

is on the bond pad. For the End point, the negative terminal is on the bond pad and the positive terminal is on the wire bond (see Figure P.6). From the look in the top view, the structures for different port polarizations are the same. You need to go to Port->Port Properties to check the direction of each vertical localized port created by MGRID based upon your specification in the Wire Bond Parameters dialog.

Also, the two structures for the wire bonds are built using the Port Order of "Order on Sides". In case you choose the Port Order of "Order on Wire Bonds", you will get ports 5 and 6 for the No.0 wire bond, ports 7 and 8 for the No.1 wire bond and ports 9 and 10 for the No.2 wire bond.

Table P.2 The Wire Bond Parameters controlling how the wire bonds are built.

Parameters	Description
Style	We can build the wire bonds in different styles: (1) Non-Symmetrical. (2) Symmetrical; (3) 3-Segments (JEDEC). They differ in the profiles. 
Number of Break Points (NBP)	We use straight segments to approximate the curved wire bonds in the Non-Symmetric and Symmetric styles. The NBP defines how many break points are created in the wire bonds.
Cross-Section Shape (CSS)	You have the choice of General and Rectangular shape for the CSS. If it is Rectangular, the wire bond is fixed as rectangular shape. If it is General, you have the option for number of segments for the cross-section of the wire bonds defined by Wire Cross Section (WCS). WCS = 2 means it is a ribbon. WCS = 3 means it is triangular. WCS = 4 means it is square.
Swap	Swap the 2 vertices of a wire bond.
Swap All	Swap the 2 vertices of each wire bond.
Start	Allowing you to edit the Start point's properties. You can define a port at the start point of a wire bond. You can define ports at the end points of all the wire bonds.
End	Allowing you to edit the End point's properties. You can define a port at the end point of a wire bond. You can define ports at the end points of all wire bonds.
Z-Extremum	Allowing you to define the z-max or z-min of a wire bond or all wire bonds.
Alpha	It is the angle of slanted segment at the end point for the 3-Segments (JEDEC) style. You can change the Alpha for one wire bond or all wire bonds in one shot.
Beta	It is the angle of slanted segment at the start point for the 3-Segments (JEDEC) style. You can change the Beta for one wire bond or all wire bonds in one shot.
H-Segment	It is for the horizontal segment of the 3-Segment (JEDEC style) only. You can change the H-Segment value for one or all wire bonds in one shot.
Port Order	It defines how the ordering of the ports are if you define ports at the Start points and End points simultaneously. If it is Order on Sides, the ports will start from all the start points first. Then, it will be for all the end ports.

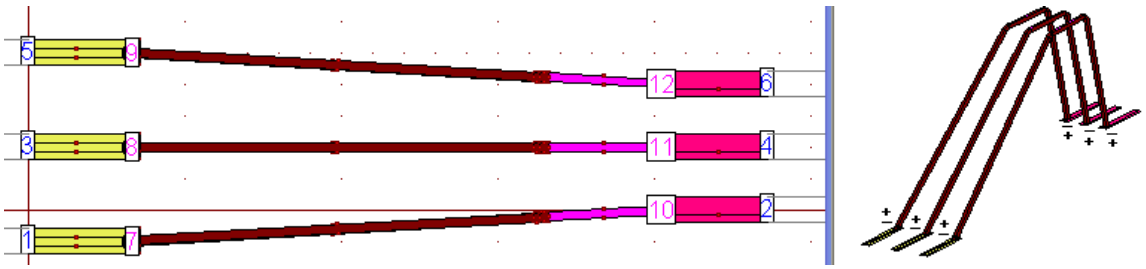


Figure P.5 The wire bonds with Forward-Directed ports on both the Start and End points.

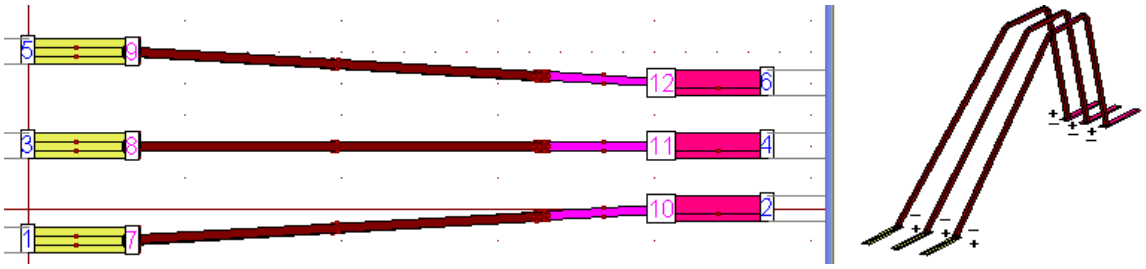
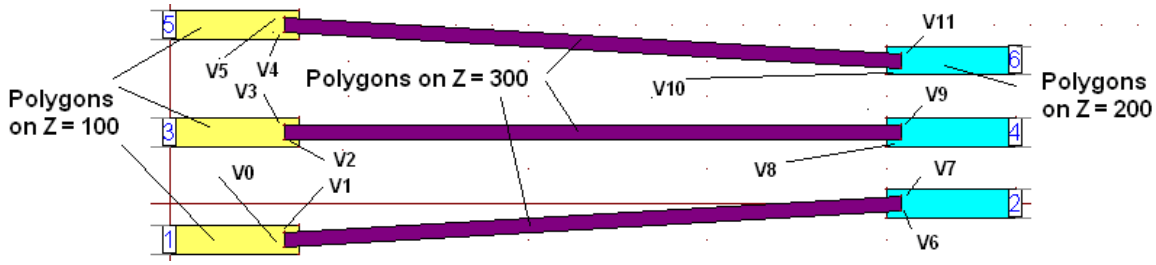


Figure P.6 The wire bonds with Backward-Directed ports defined on the wire bonds.

3. Finding Wire Bond Start Points and End Points from an Imported GDSII File.

When we import a structure from a GDSII file, a wire bond is normally defined by a trace on a specific layer. A typical example is the file in .\ie3d\samples\bga1.geo. The 6 traces on Z = 1900 represent 6 wire bonds. The start point of each wire bond is at Z = 1000 and the end point of each wire bond is at Z = 1250.



The ribbon wire bond feature allows you to build a wire bond between two selected edges. In fact, IE3D can allow you much more flexibility in building multiple wire bonds in one shot. The multiple wire bond feature allows you to build both ribbon wire bonds and solid wire bonds for a number of vertex pairs. Each vertex pair defines the start and end points of a wire bond.

The more general implementation resolves the problem. It is the same command. However, it does not require the users to select the edges. Instead, it requires the users to enter two vertices as if they are entering the vertices for a polygon. If the user selects **Wire Bonding** in **Edit** menu after entering two vertices, MGRID will prompt the user for the wire bond parameters. Besides the **Number of Break Points**, the **Z-Extremum** and **Wire Bond Style**, the users are also required to enter the **Radius** and **Circle Segments**. The **Circle Segments** is the number of segments for approximating the wire.

If a user wants to build multiple wire bonds in one step, he can enter 2N vertices before he selects the **Wire Bonding** command in **Edit** menu. MGRID will assume the user wants to build N-wire bonds: vertices 1 and 2 for the 1st wire bond, vertices 3 and 4 for the 2nd wire bond … etc. The user can have the option to have all the wire bonds to be of the same height or different height by changing the status of the **Same Z-Extremum** option (see Figure P.2).

The most important issue is the polygon connection between the wire bonds and the bond pads. The Wire Bonding feature is so advanced that the MGRID will automatically build the connection between the wire bonds and bond pads for you. You do not need to worry about the connection. However, careful planning on the wire bonds and bond pads will save you some cells. You need to practice it before you can realize how careful planning on constructing the wire bonds will save the cells. The Section 7 of Chapter 9 in fact is a good example.

Also, you can always import and export vertices for wire bonds from files using the Create and Edit Vertices command in Input menu. For example, if you want to build the wire bonds for many different packages. You can save the wire bond vertices into a file. You can import them whenever you need. It will save you much time for re-entering the vertices.

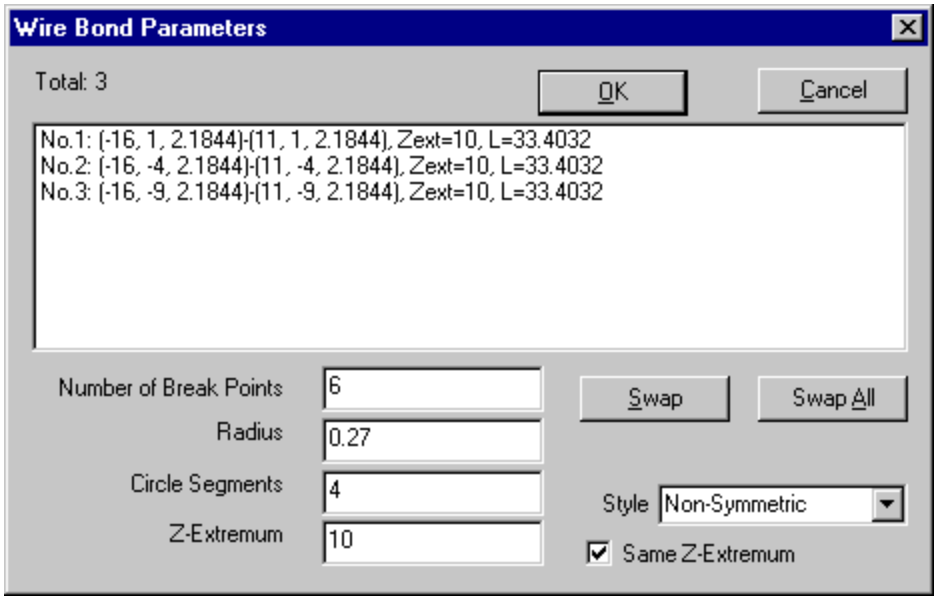


Figure P.2 The dialog for the wire bond parameters.

(3) The most flexible way to build a wire bond:

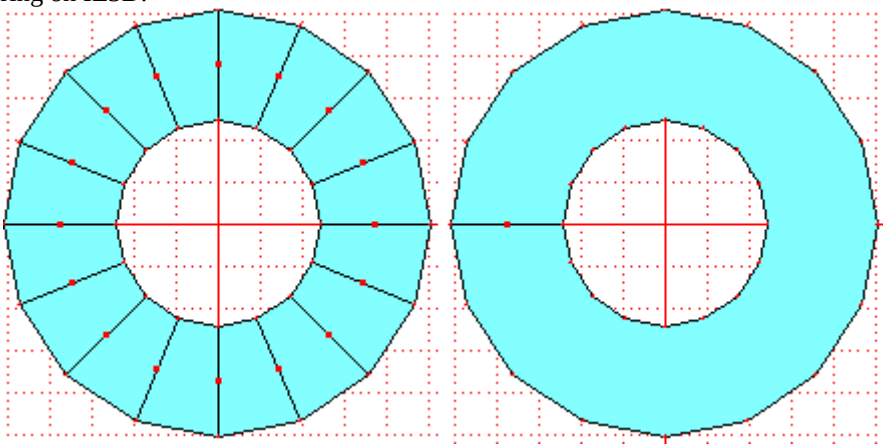
As you can see, the wire bond shapes are limited to two types for the Wire Bonding command in Edit menu: symmetrical and non-symmetrical. If you have an arbitrary shaped wire bond, you will not be able to build it using the command. In fact, we even have a more flexible way to build a wire bond. Basically, you can enter the shape of the wire bond as a series of vertices (you can create it in a text file and import it from Create and Edit Vertices command in Input menu). Then, you select the **Build Wire Path** in **Adv Edit** menu. It will allow you to build a wire bond with the shape defined by the vertices. You can specify whether you want the electrical connections at the both ends. For example, if you check the check box with **Build Connection at the Start**, MGRID will try to build electrical connection at the start point.

Compared to the **Wire Bonding** command, the **Build Wire Path** command is more flexible on the shape. However, it cannot allow you to build a number of wire bonds in one step. If you have a few wire bonds to build and each wire bond has a special shape, is there any fast way to build it. The answer is yes.

You can use the 3D Text format to build the wire bonds. Please check the Appendix O for more information on the 3D Text format.

Appendix Q. Merging Polygons, Removing Inserted Vertices and Checking Twisted 3D Polygons

In Chapter 5, we discussed how we could use the **Mesh Selected Polygons** in **Adv Edit** menu to discretize the selected polygons into small rectangles and triangles for better accuracy in modeling MIM capacitors. In the other way, we may want to merge some polygons together. For this purpose, we implemented the **Merge Selected Polygons** feature in **Adv Edit** menu. The use of the Merge Selected Polygons is quite straightforward. You just select the polygons you want to merge, and select **Merge Selected Polygons** in **Adv Edit** menu. The only comment we would like to make is about the single connected polygon. As you may know, IE3D only accepts single connected polygons. For the ring structure shown in Figure Q.1A, when you select all the polygons and perform the **Merge Selected Polygons**, MGRID 4.02 will merge them into 1 single polygon as shown in Figure Q.1B. However, you should understand that there is always an edge cutting the ring. It is impossible to build the ring without the cutting edge for the ring on IE3D.

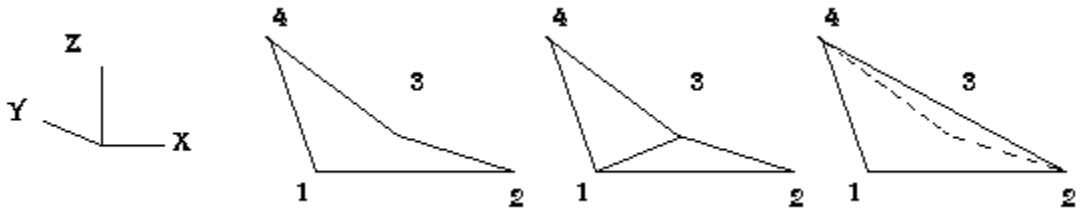


(A) Before Merged (B) After Merged

Figure Q.1 The result of merging polygons.

Another cleaning command implemented into MGRID 4.02 is the **Remove Inserted Vertices** in **Adv Edit** menu. As you know, we can use **Add Edge Vertex** in **Edit** menu to insert vertices on edges to control the meshing. However, there is no easy way in MGRID 4.0 to remove the inserted vertices. On MGRID 4.02, you can use the **Remove Inserted Vertices** to remove the inserted vertices.

Twisted 3D polygons create ambiguity. Figure Q.2 shows an example. The 3D polygon shown in Figure Q.2A has 4 vertices. Such a 3D polygon does not have a unique definition when the 4 vertices are not on the same plane. The 4 vertex 3D polygon in Figure Q.2A can be sub-divided into 2 triangles in 2 schemes: (1) triangle 1 with vertices 1, 2, 3 and the triangle 2 with vertices 1, 3, 4 (2) triangle 1 with vertices 1, 2, 4 and triangle 2 with 3 vertices 2, 3, 4. The scheme (1) is shown in Figure Q.2B and the scheme (2) is shown in Figure Q.2C. They represent 2 completely different structures. The **Check Twisted 3D Polygons** in **Adv Edit** is implemented to detect the twisted 3D polygons and divide them automatically. Certainly, you can manually divide them if you like.



(A) (B) (C)

Figure Q.2 The ambiguity involved in 3D twisted polygons.

Appendix R. Excitation Sources

On MGRID and MODUA, we are frequently prompted to enter the excitation and termination for the ports. There are 3 different excitation sources available. They are inter-changeable except special cases. We would like to explain more on them.

1. Wave Source:

Wave source is more a microwave concept. For microwave applications, voltage and current may not be clearly defined for some structures. However, waves are defined clearly. The wave source is defined as:

$$a = a0 + \Gamma\, b \qquad \text{(R-1)}$$
$$\Gamma = (Z0 - Zc) / (Z0 + Zc) \qquad \text{(R-2)}$$

Where a is the incident wave; b is the reflected (or outgoing) wave; a0 is the wave source; Γ is the reflection from the source; Z0 is the source impedance; Zc is the characteristic impedance of the transmission line.

For the implementation on IE3D, we always assume the Zc = 50-ohms. The limitation on the Wave Source is that we cannot have a source impedance to be –Zc.

There are two different waves: voltage wave and current wave. The difference is that there is a 180-degree phase difference in the reflected wave (or outgoing wave). Also, there is a difference of a factor of Zc in the magnitude. On the IE3D, we always use the voltage wave concept.

We define antenna gain and efficiency based upon the wave sources.

2. Voltage Source:

Voltage source and current source are the concepts of any electrical circuits. The voltage source is defined as:

$$V = V0 - I\, Z0 \qquad \text{(R-3)}$$
$$I = V0 / (Z0 + Z) \qquad \text{(R-4)}$$

Where V is voltage at the port; I is the current at the port (looking into the circuit); V0 is the voltage source; Z0 is the source impedance; Z is the input impedance at the port.

The limitation of voltage source is that we cannot assume the Z and Z0 to be equal to 0.

3. Current Source:

The current source is defined as

$$I = I0 - V / Z0 \qquad \text{(R-5)}$$
$$V= I0\, Z0\, Z / (Z0 + Z) \qquad \text{(R-6)}$$

Where V is voltage at the port; I is the current at the port (looking into the circuit); I0 is the current source; Z0 is the source impedance; Z is the input impedance at the port.

The limitation of current source is that we cannot assume the Z and Z0 to be approaching infinity.

You may want to know what source you should use. It really depends upon the situation you are encountering. If we want the voltages at the ports to be fixed, we should use voltage sources with the Z0 =

0. If we want the currents at the ports to be fixed, we should use current sources with the Z0 = ∞. If we want the incident powers at the ports to be fixed, you can use wave source with Z0 = Zc (50-ohms or non 50-ohms). Can we specify the net input power into a port? The answer is no. Theoretically, we can adjust the different sources to make the net input power at each port to be fixed. However, it is a non-linear problem and we do not implement it. In fact, for a single port structure, fixing the power at the port is not difficult at all because it degenerates to a linear problem again. Fixing the power at each port for a multiple port structure is non-trivial. It requires non-linear optimization.

Different source types are inter-changeable. It can be easily proved there exist the following equations.

$$V0 = I0\ Z0 \quad \text{(R-7)}$$

$$a0 = V0\ Zc\ /\ (\ Z0 + Zc\ ) \quad \text{(R-8)}$$

4. Definitions of Gain and Efficiency:

The different source types may cause confusion in antenna gain and efficiency. If you have a wave source, the incident wave is a constant. It does not care whether there is any reflection coming back to the source. For an ideal wave source, the reflected power will be absorbed. We will consider the efficiency of the circuit as the net input power over the incident power, where net input power is the incident power minus the reflected power. For antenna applications, only part of the net input power is radiated out. On IE3D, we define Antenna Efficiency as the radiated power over the incident power.

In case you have a voltage source, you may consider the efficiency is the ratio between the radiated power and the output power of the voltage source. For this reason, we have defined Conjugate Match Antenna Efficiency in the IE3D 11. It is defined as the ratio between the radiated power and the output power of the voltage source when the source impedance and the input impedance are conjugate match. At conjugate match, the net input power will be always the same as the power consumed in the source impedance. Even the net input power can be all radiated out (radiation efficiency = 100%), the Conjugate Match Antenna Efficiency is only 50%. On the IE3D, Antenna Gain is defined based upon Antenna Efficiency. We also defined Conjugate Match Antenna Gain based upon Conjugate Match Antenna Efficiency. The definitions are documented in Appendix AV.

Please understand that the value of Conjugate Match Antenna Efficiency is unique only for 1-port structure. For a 2 or multiple port structure, the voltage over current ratio (V/I) at a specific port is dependent upon the network s-parameters and the excitations and terminations at all the ports. What we do is the following: (1) Given the excitations and terminations at all the ports, we find the V and I at each port. (2) From the V and I at each port, we find the Zin = V / I at a port. Then, we assume a constant voltage source V0 with the source impedance as Zs = Conjugate(Zin). The V0 is chosen to maintain the original V and I values. Then, we solve the Conjugate Match Efficiency based upon the V0 value and other values at each ports. Again, the solved Conjugate Match Efficiency is for the specific excitations and terminations of the antenna.

Appendix S. Accurate Modeling of Metallic Thickness and Surface Roughness

As we have discussed before, we have 2 ways to model metallic thickness. The default way is to use infinitely thin strip to model the strip. Then, we use some analytical formula for the surface impedance (Zs) to model the loss effect of the strip. The skin effect is included precisely in the analytical formula for the Zs. We call this model as Thin Strip Model. The infinitely thin strip model is good when the strip width is much bigger than the strip thickness. Normally, it can model the loss and the reactive effects quite well when the thickness of the strip is small compared to the width.

The other way is an accurate model for thick strips. Instead of approximating the strip as infinitely thin strip, we model the current on the 4 sides of the strip as it is shown in Figure S.1 (b). We call this model the Thick Strip Model.

The IE3D 14 also implement the surface roughness modeling.

(a) (b)

Figure S.1 (a)The cross-section of the infinitely thin strip model. The current is assumed on the one single strip only. (b) The cross-section of the finite thick strip model. The current is assumed on the 4 sides of the strips.

The Thick Strip Model is a much more sophisticated scheme. It models a thick strip as some current flowing on the 4 surfaces of the strip. It can yield high accuracy results even for thick strips. As you know, the current is mainly flowing on the surfaces at high frequency. The Thick Strip Model certainly can match the physics very well. In fact, it turns out that the Thick Strip Model is also extremely accurate at low frequency. At low frequency, the current is kind of uniform on the cross-section. The surface impedance can capture the uniform current's effect at low frequency automatically. The Thick Strip model is also very precise for low frequency. Basically, the Thick Strip model can cover wide frequency range from very low to very high frequency.

However, one issue had been bothering the Thick Strip Model is the loss effect. As we mentioned above, the loss effect is modeled as the surface impedance and it is quite accurate when we apply it to the Thin Strip Model. However, the surface impedance is a function of thickness and conductivity. When we build thickness on a thick strip, we basically create 4 polygons to represent the thick strip. Each polygon is only part of the strip. If we use the original polygon properties (thickness and conductivity), the simulation will predict less loss than the reality. In order to preserve the accuracy of loss calculation, we need to have a way to correct the loss when we model a thick trace as 4 polygons.

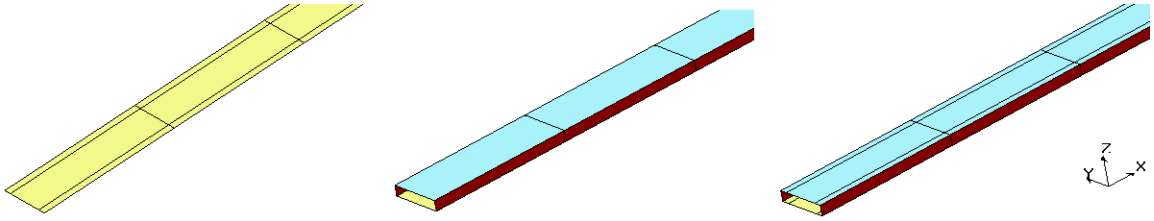
Before the IE3D 9.3, we have a way to correct the loss when build a trace as 4 polygons using the Grow Metallic Thickness commands. However, we realized that the correction is not good enough. Normally, it will still under estimate the loss at high frequency.

On the IE3D 9.3, we have implemented a new scheme for it. On the IE3D 9.3, when you build the thickness on a trace, we will automatically use the Thick Strip Model to model the strip as 4 polygons. Also, we will create some additional metallic type for the 4 newly created polygons. The new metallic type will have the same metallic properties as the original strip. However, we will introduce a factor called Cross Section Factor for the new metallic type for the 4 polygons. The Cross Section Factor is defined as:

Cross Section Factor = W / (2 W + 2 T) (S-1)

Where W is the width of the trace and T is the thickness of the trace. It is normally slightly below 0.5. The Cross Section Factor is automatically calculated precisely when you use the **Edit->Layer->Grow Metal Thickness on Layer** or the **Edit->Grow Metal Thickness on Selected Polygons** commands. Internally, we will use the Cross Section Factor as well as the other parameters to predict the surface impedance Zs. It turns out that the new model creates highly accurate results.

The standard IE3D procedure is using Thin Strip model. However, from the Thin Strip model to Thick Strip model is quite simple. If we want to make the Thick Strip model for all the traces on a specific layer. We can focus the input on the layer. Then, we select Edit->Layer->Grow Metal Thickness on Layer. You have the option to build the thickness on the polygons on this layer in the +z or –z direction.



(a) Thin Strip with AEC (b) Thick Strip without 3D AEC (c) Thick Strip with 3D AEC

Figure S.2 The different models.

We will discuss an example here to demonstrate the importance of Thick Strip model and AEC. Saved in .\ie3d\samples\thin_strip.geo is a Thin Strip model of a TLN. Its meshing with AEC enabled is shown in Figure S.2a. The Thick Strip model of the same structure is saved in .\ie3d\samples\thick_strip.geo. AEC is enabled for it. However, no edge cells are created in the meshing (see Figure S.2b). Edge cells are used to capture the effect of the edge singularity. Thick Strip model has 2 vertical strips along the side edges. The vertical strips can capture the edge effects precisely normally. It is the reason why regular AEC will not create edge cells on thickness strips. However, some users want even high accuracy. They want to be able to apply edge cells along the edges of thickness strip. To meet their requirements, we have implemented the 3D AEC scheme. Saved in .\ie3d\samples\thick_strip_3d_aec.geo is such an example. In fact, it is the same structure as the one in thick_strip.geo except one of its meshing parameter is different. It is the AEC Level in the Automatic Edge Cells parameters in the Meshing Parameters. For the thick_strip.geo, we define AEC Level = Applied to open edges only. For the thick_strip_3d_aec.geo, we define AEC Level = Applied to thickness edges too. Using 3D AEC, IE3D will mesh the structure into more cells. In return, it yields even more accurate results.

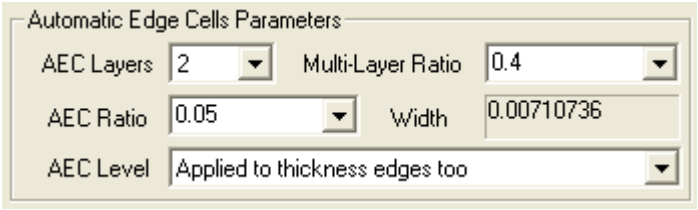


Figure S.3 The AEC parameters section for the thick_strip_3d_aec.geo.

Figure S.4 shows the comparison between them. The two Thick Strip models yield quite close results while the result from Thin Strip AEC model is obvious different in dB[S (1, 1)]. This is a 50-micron wide and 10-micron thick strip. The Width over Thickness ratio is only 5. We certainly can not expect the Thin Strip model can yield very accurate results even with AEC. The comparison in the TLN parameters between the three models is shown in Figure S.5. Assuming the result from Thick Strip 3D AEC is the standard value, we can see the error for the Thick Strip model (without 3D AEC) is about 1% in the Re[Zc] while the

error for the Thin Strip model is about 7%. Please note that 7% in Re [Zc] may yield obvious difference in the dB[S (1, 1)]. However, for most applications, you may not see significant difference in S (2, 1) and coupling effects. It may still be ok. However, to get more accurate results, you have to go for the Thick Strip model.

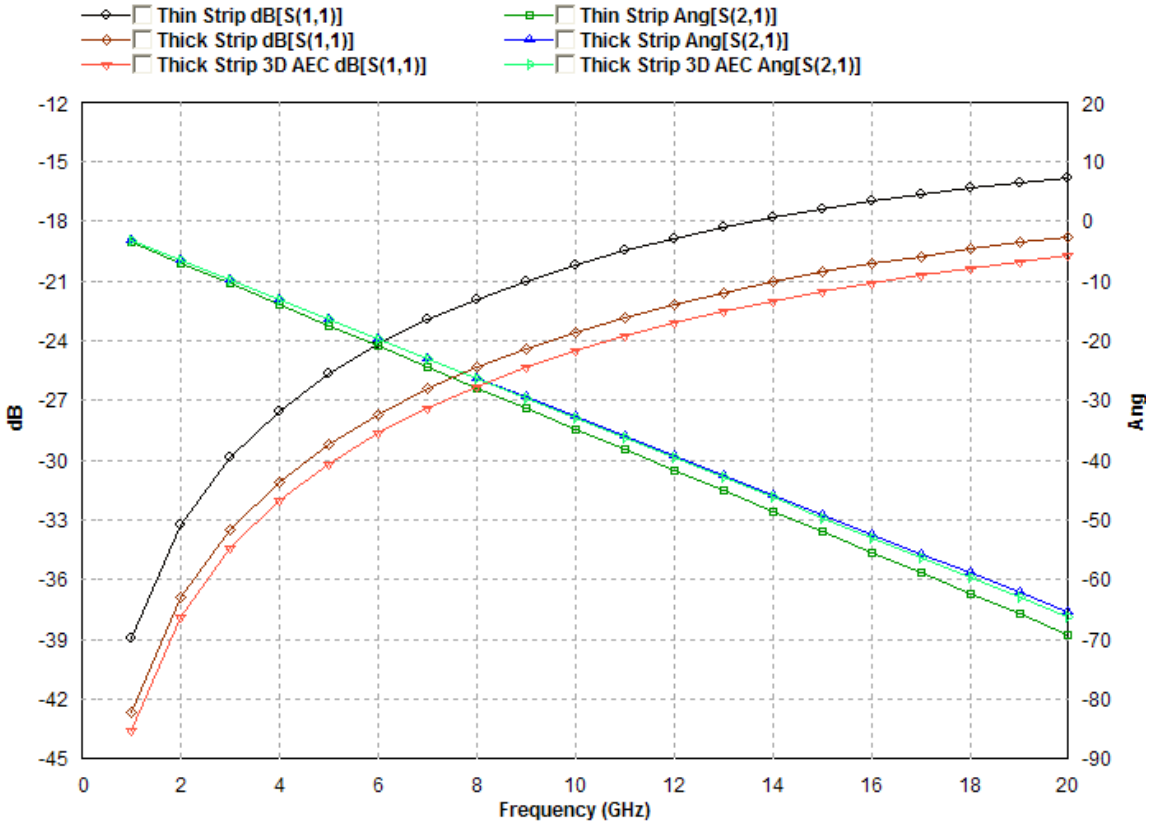
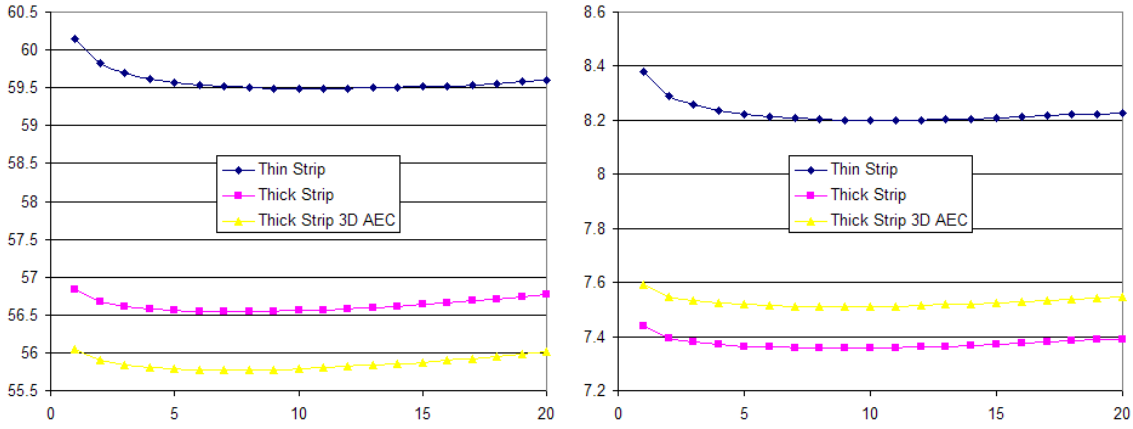


Figure S.4 The comparison in dB[S (1, 1)] and Ang[S (2, 1)] between the three models.



(a) Comparison on Re [Zc] (b) Comparison on Re (Ereff)

Figure S.5 The comparison in the TLN parameters between different models.

Surface roughness is defined as the height from the valley to the peak. It can be defined in the Metallic Strip parameters in the Basic Parameters dialog.

Appendix T Definition of Dielectric Layers and Metallic Layers

We notice some users have difficulty to understand the dielectric layer and metallic layers in early versions of the IE3D. Basically, there are 2 layer systems on the IE3D: (1) the dielectric layer system; (2) the metallic layer system.

The dielectric layer system is defined in Parameters->**Basic Parameters**. By default, the No.0 dielectrics is always at z = 0. The default No.0 dielectrics is always the ground plane. What it means is that the ground plane is from z=-∞ to 0. In practice, the ground plane is only a thin metallic plate. Why is there such a difference? Mathematically, a finite thick or infinite thick ground plane yields the same result when they are large in the horizontal directions. Therefore, we just define that the ground plane is occupying the whole lower half space for a structure with a large ground plane.

The metallic layer system is shown on the lower right window of the MGRID. We always assume a dielectric interface defined in the dielectric layer system as a metallic layer, for the convenience of geometry entry. You can define as many metallic layers as you like. It is not necessary that you create some polygons or vertices of polygons on the metallic layers defined. Assume you want to create a metallic layer that is not a dielectric interface. An example is the strip line structure in the 3rd example in Table 2.1. MGRID will not create a metallic layer at z = 5 (where the metallic trace will be) because it is not a dielectric interface (dielectric interface is at z = 10 only). To create a new metallic layer, you should go to 2D Input in Edit menu. Table 2.1 shows examples of dielectric.

Table 2.1 The definition of dielectric layers on IE3D.

Cases	Picture	Definition
1-substrate with thickness= 3, ε_r = 2. Upper space is air	Air Substrate ε_r = 2 Ground Plane	No.0, Ztop=0, Epsr=1,…, Sigma=(4.9e7,0) s/m No.1, Ztop =3, Epsr=1,…, Sigma=(0,0) s/m Note: There is no need to define the air on the top layer.
1-substrate with thickness=3, ε_r = 2. Upper space is filled with dielectrics ε_r = 1.5. Below 0 is semi-conductor with ε_r = 10, σ=5 s/m.	Dielectric ε_r =1.5 Substrate ε_r = 2 Semi-conductor	No.0, Ztop=0, Epsr=10, TanD(Epsr)=0,.., Sigma=(5,0) s/m No.1, Ztop=3, Epsr=2,…, Sigma=(0,0) s/m No.2, Ztop=1.0e+15, Epsr=1.5,…, Sigma=(0,0) s/m Note: You can define dielectric loss as either loss tangent or conductivity. If you define both, they will be summed up.
Strip-line structure with ground-to-ground distance of 10. Metallic trace is at the middle. Substrate ε_r = 2 and loss tangent = 0.01.	Top Ground Substrate ε_r = 2 Substrate ε_r = 2 Bottom Ground	No.0, Ztop=0, …, Sigma=(4.9e7,0) s/m No.1, Ztop=10, Epsr=2, TanD(Epsr)=0.01,…, Sigma=(0,0) No.2, Ztop=1.0e+15,…, Sigma=(4.9e7,0) s/m Note: Even though we will build metallic traces at z = 5 for the strip line, we do not need to define a substrate there. We can define the substrate in the whole range. Some people may separate the No.1 layer as two layers (No.1, Ztop=5, No.2, Ztop=10, No.3, Ztop=1.0e+15) with identical dielectric properties. In fact, it is not necessary.
Two dielectric layers with the 1st dielectric ε_r = 12.9, thickness = 4, 2nd dielectrics ε_r = 6.7 with thickness = 0.04 on top of the 1st dielectrics.	Air Substrate ε_r=6.7 Substrate ε_r=12.9 Bottom Ground	No.0, Ztop=0, …, Sigma=(4.9e7,0) s/m No.1, Ztop=4, Epsr=12.9,… Sigma=(0,0) s/m No.2, Ztop=4.04, Epsr=6.7,…, Sigma=(0,0) s/m Note: Because the 2nd dielectric layer is on top of the 1st one, the No.2 Ztop = No.1 Ztop + Thickness of the 2nd layer.

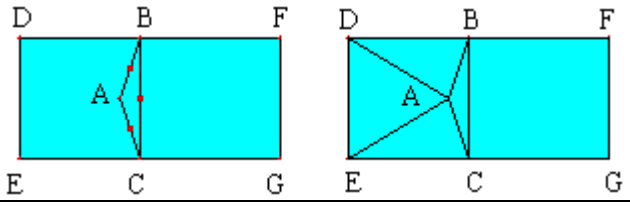
The whole space is filled with air	Air on Upper Half Space Air on Lower Half Space	No.0, Ztop=0, Epsr=1,…, Sigma=(0,0) s/m No.1, Ztop=1e+15, Epsr=1,… Sigma=(0,0) s/m
A double sided microstrip structures or (microstrip structure) with a common ground plane.	Air Substrate ε_r=6.7 Substrate ε_r=6.7 Air	No.0, Ztop=0, Epsr=1,…, Sigma=(0,0) s/m No.1, Ztop=4, Epsr=6.7,… Sigma=(0,0) s/m No.2, Ztop=4, Epsr=1,…, Sigma=(4.9e7,0) s/m No.3, Ztop=7, Epsr=6.7,…, Sigma=(0,0) s/m Note: It is assumed that the ground plane thickness is 0. The ground plane is located at z=4. The substrate thickness for the bottom microstrip is 4. The substrate thickness for the top microstrip is 3.The microstrip trace for the bottom structure should be defined on z=0. The microstrip trace for the top structure should be defined on z=7. You can define polygons on z = 4 (or the common ground plane level). However, the polygons on z = 4 will represent holes on the ground plane instead of metallic traces.

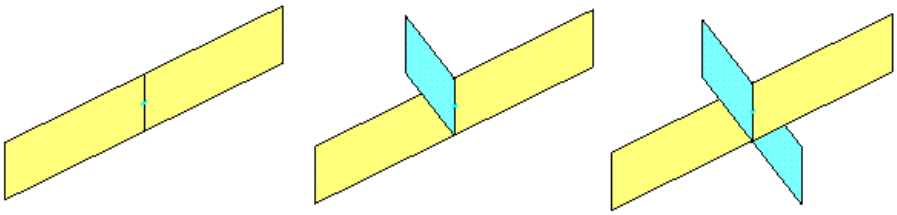
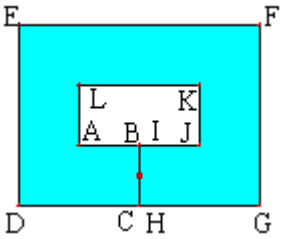
Starting from Version 4.0, the IE3D accepts HTS ground plane for HTS II. More information about HTS modeling is available in the definition of metallic strips. If you want to define a HTS ground plane, you can select the CHANGE button and define the appropriate parameters.

Appendix U. Warning and Error Messages on IE3D

When you setup IE3D simulation, you may occasionally encounter the warning and error messages. In this appendix, we will document those frequently encountered warning and error messages.

Table U.1 Warning and error messages encountered frequently in the IE3D.

Message	Explanation
WARNING: Minimum forced meshing size happens between (0, 0, 20) – (0.08, 0, 20) with size 0.08 (<0.2). Are you sure you want to continue?	IE3D uses the entered polygons and vertices to mesh the geometry. If a user wants to increase the meshing density, he can purposely insert some vertices on the edges to create some smaller cells at specific region. On the IE3D, user can enable the Automatic Edge Cells (AEC) option to let IE3D create small cells on the edges. In case a user accidentally inserted some close vertices or specified small edge cell width for AEC, the IE3D will create some tiny long cells, which may not be good for numerical stability, the IE3D will issue warning whenever it feels the possible irregular meshing cell size. For the example warning message, it says that an edge of width 0.08 will be created between (x, y, z) = (0, 0, 20) and (0.08, 0, 20). The width 0.08 is smaller than the proposed limit of 0.2. In our experience, the user does not need to worry about the warning message unless the width value is larger than 1% of the limit. Quite often, the users can still get very good results on the s-parameters even with the width value as small as 0.01% of the limit. However, tiny small cells may have big impact on the calculated radiated pattern. We do encounter some cases that the IE3D predicts accurate s-parameters, however, omni-directional radiation pattern for some structures. The omni-directional radiation pattern is in fact created by the irregular tiny long cells. Whenever a user encounters omni-directional radiation pattern or warning message with the width value smaller than 1% of the limit, the user should check the current distribution. If the user sees some "hot" spot with very high current density at the location of the small tiny cell, it is like the "hot" spot and the omni-directional radiation pattern is created by the irregular meshing. A typical example is shown in the following figure. A strip is created with 3 polygons: DBACE, ABC and BFCG. If we do not turn on the Meshing Optimization, the meshing normally will create 4 triangles (DBA, DAE, AEC and ABC) and one rectangle (BFCG). However, when the vertex A is approaching the edge BC, the triangle ABC becomes a tiny and long triangle. The tiny and long triangle ABC will not affect the current continuities on the trace. Therefore, it will not affect the s-parameters. However, the tiny cell ABC will create very high current density on it in order to maintain the current continuity. The high current density on ABC will create omni-directional radiation pattern. 
ERROR: Dynamic memory allocation failure.	This message is normally created when no enough RAM is available for simulation of a large structure. Please try to reduce the number of cells or unknowns, or try to increase the RAM on the computer.
ERROR: Edge common to more than 4 cells.	The limitation causing this error message is completely removed on the IE3D 11. On the IE3D 10 and earlier editions, when two polygons have a common edge, we consider the two polygons are electrically connected on the edge. The common edge is common to the two polygons. Same situations can be defined for the cases of 3 polygons having a common edge or 4 polygons having a common edge (shown below). Special schemes are implemented into the IE3D 10 to handle the "edge common to 2 cells", "edge common to 3 cells" and "edge common to 4 cells" cases. However, we have not implemented schemes to handle the case for edge common to more than 4 cells. A user may be curious about when such a situation happens. For the case with edge common to 4 cells, if we duplicate one cell

	in the exact location of one of the 4 cells, it will become the edge common to 5 cells. IE3D 10 will issue the error message. How can we create identical or similar cells in the same location? We found that users may make such a mistake when they are using the "Add Via on Edges" command. For example, if a user tries to select an edge to make vertical polygons. Accidentally, he may have selected more than 1 edge at the same xy location (different z-location), if he does not focus the selection to a specific layer. He may create some overlapped vertical polygons. The overlapped vertical polygons may not be able to be detected by the MGRID, and they may create the case with edge common to more than 4 cells. Anyway, the limitation is removed on the IE3D 11 and you should not see this error message on IE3D 11.  Edge common to 2 cells Edge common to 3 cells Edge common to 4 cells
ERROR: Automatic discretization failure.	There might be many reasons for the meshing program to fail to mesh the structure. However, we know one case is frequently encountered in the old IE3D versions if a user does not enter the polygons correctly. The following rectangle ring is an example. Certainly, it is good idea to create it as multiple polygons. However, if we want to create it using one polygon, we have to be very careful on the orders of the vertices. We should create the polygon with the order of the vertices as: ABCDEFGHIJKLA. In this way, we are creating a singly linked polygon. However, if we create the polygon with the order of the vertices as: ABHGFEDCIJKL, the polygon will not be singly linked polygon. They will look exactly the same. However, the 2nd polygon will cause Automatic discretization failure on the IE3D 10. A special scheme has implemented into IE3D 11 to handle the special case. 

Appendix V Scheduled Simulations and IE3D Simulations on IE3D

You can use batch simulation mode for IE3D to schedule your IE3D simulation. To submit a batch simulation, you run through the Process->Simulate process on MGRID. You select "Create .SIM File Only" in the After Setup combo in the Simulation Setup dialog. The simulation input file (*.sim) is created. You can run the simulation in the command-prompt (or DOS mode) as:

"\my_exe_dir\ie3dos" "\my_geo_dir\my_geo_file.sim" (V-1)

Where "\my_exe_dir\" is where the executables of IE3D are located. It is normally ".\ie3d\exe". The double quotations are needed while the path name contains special characters such as SPACE and TAB, etc.

If we have a number of simulations, we can go thru the Simulation Setup process with "Create .SIM File Only" option. Then, we create a batch file (.bat) with each line of the file in the format of (V-1). We can run the batch file for a batch IE3D simulation.

The above way is simple and it is still being supported. It just needs some minor manual work to create the batch file.

On the latest IE3D, we have implemented JobsManager. JobsManager was originally developed for network distributed EM simulations using ZDS/ZDM. Please read Appendix Z for more information on ZDS and ZDM. JobsManager also allow users to schedule simulation jobs.

When you setup an IE3D simulation, you can select "Invoke Local IE3D Engine". If you check the "Wait until Finished" check-box next to it, MGRID will invoke the IE3D engine to perform the simulation in the background. If you un-check "Wait until Finished", MGRID will send the job to the JobsManager. In case the JobsManager is not started, it will be automatically started and perform the simulation for you. You can perform multiple Simulation Setups with "Invoke Local IE3D Engine" and "Wait until Finished" unchecked. The jobs will be sent to JobsManager for scheduled IE3D EM simulations.

Appendix W Setup for Priority for Simulation Jobs on IE3D

Starting from the IE3D 8.0, we can setup Process Priority on the IE3D. On the Simulation Setup dialog or Optimization Setup Dialog, you can choose different options in Process Priority to control how important your job is. If you choose Highest Priority, the OS will devote as much computing power to the job as possible. If you choose Above Idle, the OS may run your simulation job when no other processes are running on the computer. You should be notified that the selection other than "Normal" might not be guaranteed due to the limitation in the OS.

Another priority is the Job Priority in the Simulation Setup dialog. It is for the scheduled IE3D simulations in the JobsManager. It only affects the order of the job in the job list in the JobsManager. After it is submitted to ZDS/ZDM or the local IE3D engine for simulation, the process priority for the OS is by the Process Priority selection discussed above.

Appendix X. More Comments on Pattern Handling and Optimization

The PatternView application is much improved on the IE3D 7.0. There are a few powerful features implemented on the PatternView 7.0.

1. Pattern Optimization:

Starting from the IE3D 7.0, you are allowed to define all different the pattern parameters as optimization objectives. In order to define pattern parameters as optimization objectives, you need to first enable saving the Radiation Pattern File before you select Add button in the optimization setup dialog in both MGRID and MODUA. Future editions planned will allow users to define the excitations as optimization variables for radiation pattern of special shape.

2. Merging Pattern from Different Files:

We have demonstrated the capability of simulation and pattern calculation of a design consisting of multiple geometry files in Chapter 13. The key to such a feature is mixed-electromagnetic simulation and pattern merging. Mixed-electromagnetic simulation has been available since the IE3D 2.0. Pattern merging is implemented in the IE3D 7.0. A user can define a design consisting geometry modules on MODUA. Then, he can enable pattern calculation for the whole design. The IE3D will simulate the network s-parameters and find the combined radiation pattern of the geometry modules in the design. This process is completely automated. In case, a user wants to merge two existing pattern files, he can still do it using the **Edit->Merge Patterns** in PatternView. It will merge all the pattern files added into the list.

3. Adding Array Factor to a Set of Radiation Patterns:

The array factor feature has been demonstrated in Chapter 13. We will not repeat it here.

4. Far Field Distribution Calculated from Radiation Pattern File:

The PatternView allows a user to predict the far field distribution from an antenna using the plane wave propagation and reflection concept. It allows a user to define the final ground plane for the antenna. Such a feature should be very useful for wave propagation and wireless base station design.

Antenna pattern merging and adding array factor are based upon approximation. The approximation assumes no coupling between the individual radiators involved. You need to understand the limitation of such an assumption. When there is no coupling or the coupling is insignificant, the results of the pattern handling should be precise. When the coupling is weak, the shape of the pattern may still be quite accurate. However, some parameters may be affected. When the coupling is strong, we have to simulate each strongly coupled portion in one single electromagnetic simulation. Otherwise, we should see significant difference in the pattern and its parameters.

Total radiated power and radiation efficiency are two of the parameters that may be affected much by neglecting the coupling among elements. If we fix the net input power, normally the total radiated power and the radiation efficiency will be over estimated. We will use a 2-element array to illustrate the reason.

Assume we have an antenna element. The net input power is Pi. The total radiated power is Pr. The radiation efficiency is $\eta = Pr / Pi$. If we use 2-elements to form an array, the radiation pattern will be different. Assume the electrical distance between the two elements is kd. By neglect the coupling, we will have the following:

$$Pi2 = 2\ Pi. \tag{X-1}$$

$$|E2(\theta, \varphi)| = |E(\theta, \varphi)\ [\ 1 + \exp(jkd\ \sin\theta\cos\varphi)]| \tag{X-2}$$

$$Pr = \int d\theta \int d\varphi \, |E(\theta, \varphi)|^2 \sin\theta \quad \text{(X-3)}$$

$$Pr2 = \int d\theta \int d\varphi \, |E2(\theta, \varphi)|^2 \sin\theta \quad \text{(X-4)}$$

Where Pi2 is the total input power at the two elements; $E(\theta, \varphi)$ is the far field pattern of the single element; $Ei2(\theta, \varphi)$ is the far field pattern of the two-element array; Pr2 is the radiated power of the two-element array.

We cannot solve the array radiated power Pr2 analytically based upon single element radiated power Pr and the electrical distance between the two element (kd). The normal trend is that the efficiency of the array $\eta 2 = Pr2 / Pi2$ normally is getting higher.

Assume the electrical distance between the element kd = 0. We will end up with Pr2 = 4 Pr. Then, the array efficiency $\eta 2 = 2 \eta$. If the element efficiency η exceeds 50%, we will end up with $\eta 2$ exceeding 100%. It certainly is not true. The problem is in equation (Z-2). When the two elements are too close, we can no longer use the equation (Z-2) to describe the far field distribution. In fact, it is much more complicated.

Appendix Y ADIX Format Converter and LEF/DEF Conversion On the IE3DLibrary

To make IE3D more compatible with other packages, we have implemented the ADIX Format Converter for bi-directional conversion between IE3D⇔IE3D 3dt⇔GDSII⇔DXF⇔ACIS⇔GERBER. The functionality of ADIX is integrated into MGRID. When ADIX license is enabled, you are able to use all the functionality of ADIX on MGRID.

GDSII is a standard 2D format for IC industry. It is well implemented and its shapes can always be imported into IE3D/ADIX.

DXF is the AutoCAD format. IE3D/ADIX accepts both 2D DXF and 3D DXF formats. When you use DXF format conversion, you may want to pay attention to multiple things: (1) DXF is a mechanical drawing format. It accepts wide range of objects. Not all the objects are accepted by IE3D. IE3D only accept those objects with closed boundaries, such as closed polylines. If you have a rectangle, you should try to build it as a closed polyline object. You should not try to build it as four line segments on DXF. Otherwise, IE3D may not be able to recognize the shape. (2) Older versions of DXF do not have length unit. You may need to try to assign the right length unit when you import a DXF file. In case you can't find a matching one, you may need to scale the structure after importing. You can use the command Input->Info on Last Entry or other way to find the length of a specific dimension. Then, you calculate the scaling factor between the expected dimension on DXF and the dimension on MGRID. You use Adv Edit->Change Dimension Scales command to scale the imported polygons.

The ACIS (.sat) is a full 3D format. It accept bodies (or lumped) described as surfaces. ADIX allows conversion between IE3D and ACIS. We only support selected SAT versions.

GERBER format is for PCB manufacturing. It has many parameters additional to the polygons. You may also need some settings in order to import/export GERBER layouts. Please read the ADIX User's Manual, adix_manual.pdf, for more information.

Besides, the IE3DLIbrary also offers bi-directional conversion through the LEF/DEF format from Cadence. It converts those higher level objects than polygons from Cadence format in LEF/DEF into the IE3Dlibrary format.

As it is mentioned in the manual, the AGIF has automatic and streamlined geometry creations and IE3D EM simulations. It is much more powerful than the ADIX. Interested users can also read the AGIF User's Manual, agif.pdf.

Appendix Z ZDS/ZDM and Distributed EM Simulation and Optimization on Windows and Linux

The ZDS/ZDM consists of a set of programs allowing you to distribute one or multiple IE3D simulation into a group of networked computers. Each distributed simulation instance is taking care of one frequency point of one job. The ZDS/ZDM manages the distributed simulations and optimizations, and combines the results to each user submitting multi-frequency jobs into the ZDS.

When you have multiple ZDM licenses (part number: ZDS070), you are able to configure ZDS/ZDM to run network distributed EM simulations for you to speed up the simulation by large factor. In case you have not purchased ZDM licenses while you have multiple IE3D full licenses and/or multiple IE3D GUI licenses, you can also configure ZDS/ZDM to distribute the jobs between the multiple IE3D full licenses. You are able to run a ZDM license the computer you are running an IE3D full license.

The diagrams for the ZDS/ZDM are shown in Chapter 2. The detailed installation and working mechanism are documented in: .\SDD_HOME\IE3D\docs\pdfdocs\installation_zdszdm.pdf.

Third-Party Information

This section provides information on open source and third-party software that may be included in the IE3D product.

- This software application may include Boost version 1.42.0 third-party software. Boost version 1.42.0 is distributed under the terms of the Boost Software License version 1.0 and is distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. See the license for the specific language governing rights and limitations under the license. You can view a copy of the license at: <your_Mentor_Graphics_documentation_directory>/legal/ boost_1.0.pdf. Portions of this software may be subject to the Boost Artistic License. You can view a copy of the Boost Artistic License at: <your_Mentor_Graphics_documentation_directory>/legal/boost_artistic_2000.pdf. Portions of this software may be subject to Apache version 2.0 license. You can view a copy of the Apache version 2.0 License at: <your_Mentor_Graphics_documentation_directory>/legal/apache_2.0.pdf. Portions of this software may be subject to jQuery which is dual licensed under the MIT license or the GNU General Public License. We elect to distribute jQuery under the MIT license. Boost version 1.42.0 may be subject to the following copyrights:

  © 2001-2003 William E. Kempf

  © 2001-2003 David Moore, William E. Kempf

  Permission to use, copy, modify, distribute and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation. William E. Kempf makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

  © 1994, 2002 Hewlett-Packard Company

  Permission to use, copy, modify, distribute and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation. Hewlett-Packard Company makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

  © 1996, 1997, 1998, 1999 Silicon Graphics Computer Systems, Inc.

  Permission to use, copy, modify, distribute and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation. Silicon Graphics makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

  © 2001 Ronald Garcia

  Permission to use, copy, modify, distribute and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appears in all copies and that both that copyright notice and this permission notice appear in supporting documentation. Ronald Garcia makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

  © 2001 Jeremy Siek

  Permission to use, copy, modify, distribute and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appears in all copies and that both that copyright notice and this permission notice appear in supporting documentation. Silicon Graphics makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

  © 2000 Jeremy Siek, Lie-Quan Lee, and Andrew Lumsdaine

  Permission to use, copy, modify, distribute and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appears in all copies and that both that copyright notice and this permission notice appear in supporting documentation. We make no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

  © 2002 CrystalClear Software, Inc.

End-User License Agreement

The latest version of the End-User License Agreement is available on-line at:
www.mentor.com/eula

IMPORTANT INFORMATION

USE OF ALL SOFTWARE IS SUBJECT TO LICENSE RESTRICTIONS. CAREFULLY READ THIS LICENSE AGREEMENT BEFORE USING THE PRODUCTS. USE OF SOFTWARE INDICATES CUSTOMER'S COMPLETE AND UNCONDITIONAL ACCEPTANCE OF THE TERMS AND CONDITIONS SET FORTH IN THIS AGREEMENT. ANY ADDITIONAL OR DIFFERENT PURCHASE ORDER TERMS AND CONDITIONS SHALL NOT APPLY.

END-USER LICENSE AGREEMENT ("Agreement")

This is a legal agreement concerning the use of Software (as defined in Section 2) and hardware (collectively "Products") between the company acquiring the Products ("Customer"), and the Mentor Graphics entity that issued the corresponding quotation or, if no quotation was issued, the applicable local Mentor Graphics entity ("Mentor Graphics"). Except for license agreements related to the subject matter of this license agreement which are physically signed by Customer and an authorized representative of Mentor Graphics, this Agreement and the applicable quotation contain the parties' entire understanding relating to the subject matter and supersede all prior or contemporaneous agreements. If Customer does not agree to these terms and conditions, promptly return or, in the case of Software received electronically, certify destruction of Software and all accompanying items within five days after receipt of Software and receive a full refund of any license fee paid.

1. **ORDERS, FEES AND PAYMENT.**

1.1. To the extent Customer (or if agreed by Mentor Graphics, Customer's appointed third party buying agent) places and Mentor Graphics accepts purchase orders pursuant to this Agreement ("Order(s)"), each Order will constitute a contract between Customer and Mentor Graphics, which shall be governed solely and exclusively by the terms and conditions of this Agreement, any applicable addenda and the applicable quotation, whether or not these documents are referenced on the Order. Any additional or conflicting terms and conditions appearing on an Order will not be effective unless agreed in writing by an authorized representative of Customer and Mentor Graphics.

1.2. Amounts invoiced will be paid, in the currency specified on the applicable invoice, within 30 days from the date of such invoice. Any past due invoices will be subject to the imposition of interest charges in the amount of one and one-half percent per month or the applicable legal rate currently in effect, whichever is lower. Prices do not include freight, insurance, customs duties, taxes or other similar charges, which Mentor Graphics will state separately in the applicable invoice(s). Unless timely provided with a valid certificate of exemption or other evidence that items are not taxable, Mentor Graphics will invoice Customer for all applicable taxes including, but not limited to, VAT, GST, sales tax and service tax. Customer will make all payments free and clear of, and without reduction for, any withholding or other taxes; any such taxes imposed on payments by Customer hereunder will be Customer's sole responsibility. If Customer appoints a third party to place purchase orders and/or make payments on Customer's behalf, Customer shall be liable for payment under Orders placed by such third party in the event of default.

1.3. All Products are delivered FCA factory (Incoterms 2000), freight prepaid and invoiced to Customer, except Software delivered electronically, which shall be deemed delivered when made available to Customer for download. Mentor Graphics retains a security interest in all Products delivered under this Agreement, to secure payment of the purchase price of such Products, and Customer agrees to sign any documents that Mentor Graphics determines to be necessary or convenient for use in filing or perfecting such security interest. Mentor Graphics' delivery of Software by electronic means is subject to Customer's provision of both a primary and an alternate e-mail address.

2. **GRANT OF LICENSE.** The software installed, downloaded, or otherwise acquired by Customer under this Agreement, including any updates, modifications, revisions, copies, documentation and design data ("Software") are copyrighted, trade secret and confidential information of Mentor Graphics or its licensors, who maintain exclusive title to all Software and retain all rights not expressly granted by this Agreement. Mentor Graphics grants to Customer, subject to payment of applicable license fees, a nontransferable, nonexclusive license to use Software solely: (a) in machine-readable, object-code form (except as provided in Subsection 5.2); (b) for Customer's internal business purposes; (c) for the term of the license; and (d) on the computer hardware and at the site authorized by Mentor Graphics. A site is restricted to a one-half mile (800 meter) radius. Customer may have Software temporarily used by an employee for telecommuting purposes from locations other than a

Customer office, such as the employee's residence, an airport or hotel, provided that such employee's primary place of employment is the site where the Software is authorized for use. Mentor Graphics' standard policies and programs, which vary depending on Software, license fees paid or services purchased, apply to the following: (a) relocation of Software; (b) use of Software, which may be limited, for example, to execution of a single session by a single user on the authorized hardware or for a restricted period of time (such limitations may be technically implemented through the use of authorization codes or similar devices); and (c) support services provided, including eligibility to receive telephone support, updates, modifications, and revisions. For the avoidance of doubt, if Customer requests any change or enhancement to Software, whether in the course of receiving support or consulting services, evaluating Software, performing beta testing or otherwise, any inventions, product improvements, modifications or developments made by Mentor Graphics (at Mentor Graphics' sole discretion) will be the exclusive property of Mentor Graphics.

3. **ESC SOFTWARE.** If Customer purchases a license to use development or prototyping tools of Mentor Graphics' Embedded Software Channel ("ESC"), Mentor Graphics grants to Customer a nontransferable, nonexclusive license to reproduce and distribute executable files created using ESC compilers, including the ESC run-time libraries distributed with ESC C and C++ compiler Software that are linked into a composite program as an integral part of Customer's compiled computer program, provided that Customer distributes these files only in conjunction with Customer's compiled computer program. Mentor Graphics does NOT grant Customer any right to duplicate, incorporate or embed copies of Mentor Graphics' real-time operating systems or other embedded software products into Customer's products or applications without first signing or otherwise agreeing to a separate agreement with Mentor Graphics for such purpose.

4. **BETA CODE.**

4.1. Portions or all of certain Software may contain code for experimental testing and evaluation ("Beta Code"), which may not be used without Mentor Graphics' explicit authorization. Upon Mentor Graphics' authorization, Mentor Graphics grants to Customer a temporary, nontransferable, nonexclusive license for experimental use to test and evaluate the Beta Code without charge for a limited period of time specified by Mentor Graphics. This grant and Customer's use of the Beta Code shall not be construed as marketing or offering to sell a license to the Beta Code, which Mentor Graphics may choose not to release commercially in any form.

4.2. If Mentor Graphics authorizes Customer to use the Beta Code, Customer agrees to evaluate and test the Beta Code under normal conditions as directed by Mentor Graphics. Customer will contact Mentor Graphics periodically during Customer's use of the Beta Code to discuss any malfunctions or suggested improvements. Upon completion of Customer's evaluation and testing, Customer will send to Mentor Graphics a written evaluation of the Beta Code, including its strengths, weaknesses and recommended improvements.

4.3. Customer agrees to maintain Beta Code in confidence and shall restrict access to the Beta Code, including the methods and concepts utilized therein, solely to those employees and Customer location(s) authorized by Mentor Graphics to perform beta testing. Customer agrees that any written evaluations and all inventions, product improvements, modifications or developments that Mentor Graphics conceived or made during or subsequent to this Agreement, including those based partly or wholly on Customer's feedback, will be the exclusive property of Mentor Graphics. Mentor Graphics will have exclusive rights, title and interest in all such property. The provisions of this Subsection 4.3 shall survive termination of this Agreement.

5. **RESTRICTIONS ON USE.**

5.1. Customer may copy Software only as reasonably necessary to support the authorized use. Each copy must include all notices and legends embedded in Software and affixed to its medium and container as received from Mentor Graphics. All copies shall remain the property of Mentor Graphics or its licensors. Customer shall maintain a record of the number and primary location of all copies of Software, including copies merged with other software, and shall make those records available to Mentor Graphics upon request. Customer shall not make Products available in any form to any person other than Customer's employees and on-site contractors, excluding Mentor Graphics competitors, whose job performance requires access and who are under obligations of confidentiality. Customer shall take appropriate action to protect the confidentiality of Products and ensure that any person permitted access does not disclose or use it except as permitted by this Agreement. Customer shall give Mentor Graphics written notice of any unauthorized disclosure or use of the Products as soon as Customer learns or becomes aware of such unauthorized disclosure or use. Except as otherwise permitted for purposes of interoperability as specified by applicable and mandatory local law, Customer shall not reverse-assemble, reverse-compile, reverse-engineer or in any way derive any source code from Software. Log files, data files, rule files and script files generated by or for the Software (collectively "Files"), including without limitation files containing Standard Verification Rule Format ("SVRF") and Tcl Verification Format ("TVF") which are Mentor Graphics' proprietary syntaxes for expressing process rules, constitute or include confidential information of Mentor Graphics. Customer may share Files with third parties, excluding Mentor Graphics competitors, provided that the confidentiality of such Files is protected by written agreement at least as well as Customer protects other information of a similar nature or importance, but in any case with at least reasonable care. Customer may use Files containing SVRF or TVF only with Mentor Graphics products. Under no circumstances shall Customer use Software or Files or allow their use for the purpose of developing,

enhancing or marketing any product that is in any way competitive with Software, or disclose to any third party the results of, or information pertaining to, any benchmark.

5.2. If any Software or portions thereof are provided in source code form, Customer will use the source code only to correct software errors and enhance or modify the Software for the authorized use. Customer shall not disclose or permit disclosure of source code, in whole or in part, including any of its methods or concepts, to anyone except Customer's employees or contractors, excluding Mentor Graphics competitors, with a need to know. Customer shall not copy or compile source code in any manner except to support this authorized use.

5.3. Customer may not assign this Agreement or the rights and duties under it, or relocate, sublicense or otherwise transfer the Products, whether by operation of law or otherwise ("Attempted Transfer"), without Mentor Graphics' prior written consent and payment of Mentor Graphics' then-current applicable relocation and/or transfer fees. Any Attempted Transfer without Mentor Graphics' prior written consent shall be a material breach of this Agreement and may, at Mentor Graphics' option, result in the immediate termination of the Agreement and/or the licenses granted under this Agreement. The terms of this Agreement, including without limitation the licensing and assignment provisions, shall be binding upon Customer's permitted successors in interest and assigns.

5.4. The provisions of this Section 5 shall survive the termination of this Agreement.

6. **SUPPORT SERVICES.** To the extent Customer purchases support services, Mentor Graphics will provide Customer updates and technical support for the Products, at the Customer site(s) for which support is purchased, in accordance with Mentor Graphics' then current End-User Support Terms located at http://supportnet.mentor.com/about/legal/.

7. **AUTOMATIC CHECK FOR UPDATES; PRIVACY.** Technological measures in Software may communicate with servers of Mentor Graphics or its contractors for the purpose of checking for and notifying the user of updates and to ensure that the Software in use is licensed in compliance with this Agreement. Mentor Graphics will not collect any personally identifiable data in this process and will not disclose any data collected to any third party without the prior written consent of Customer, except to Mentor Graphics' outside attorneys or as may be required by a court of competent jurisdiction.

8. **LIMITED WARRANTY.**

8.1. Mentor Graphics warrants that during the warranty period its standard, generally supported Products, when properly installed, will substantially conform to the functional specifications set forth in the applicable user manual. Mentor Graphics does not warrant that Products will meet Customer's requirements or that operation of Products will be uninterrupted or error free. The warranty period is 90 days starting on the 15th day after delivery or upon installation, whichever first occurs. Customer must notify Mentor Graphics in writing of any nonconformity within the warranty period. For the avoidance of doubt, this warranty applies only to the initial shipment of Software under an Order and does not renew or reset, for example, with the delivery of (a) Software updates or (b) authorization codes or alternate Software under a transaction involving Software re-mix. This warranty shall not be valid if Products have been subject to misuse, unauthorized modification or improper installation. MENTOR GRAPHICS' ENTIRE LIABILITY AND CUSTOMER'S EXCLUSIVE REMEDY SHALL BE, AT MENTOR GRAPHICS' OPTION, EITHER (A) REFUND OF THE PRICE PAID UPON RETURN OF THE PRODUCTS TO MENTOR GRAPHICS OR (B) MODIFICATION OR REPLACEMENT OF THE PRODUCTS THAT DO NOT MEET THIS LIMITED WARRANTY, PROVIDED CUSTOMER HAS OTHERWISE COMPLIED WITH THIS AGREEMENT. MENTOR GRAPHICS MAKES NO WARRANTIES WITH RESPECT TO: (A) SERVICES; (B) PRODUCTS PROVIDED AT NO CHARGE; OR (C) BETA CODE; ALL OF WHICH ARE PROVIDED "AS IS."

8.2. THE WARRANTIES SET FORTH IN THIS SECTION 8 ARE EXCLUSIVE. NEITHER MENTOR GRAPHICS NOR ITS LICENSORS MAKE ANY OTHER WARRANTIES EXPRESS, IMPLIED OR STATUTORY, WITH RESPECT TO PRODUCTS PROVIDED UNDER THIS AGREEMENT. MENTOR GRAPHICS AND ITS LICENSORS SPECIFICALLY DISCLAIM ALL IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT OF INTELLECTUAL PROPERTY.

9. **LIMITATION OF LIABILITY.** EXCEPT WHERE THIS EXCLUSION OR RESTRICTION OF LIABILITY WOULD BE VOID OR INEFFECTIVE UNDER APPLICABLE LAW, IN NO EVENT SHALL MENTOR GRAPHICS OR ITS LICENSORS BE LIABLE FOR INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES (INCLUDING LOST PROFITS OR SAVINGS) WHETHER BASED ON CONTRACT, TORT OR ANY OTHER LEGAL THEORY, EVEN IF MENTOR GRAPHICS OR ITS LICENSORS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. IN NO EVENT SHALL MENTOR GRAPHICS' OR ITS LICENSORS' LIABILITY UNDER THIS AGREEMENT EXCEED THE AMOUNT RECEIVED FROM CUSTOMER FOR THE HARDWARE, SOFTWARE LICENSE OR SERVICE GIVING RISE TO THE CLAIM. IN THE CASE WHERE NO AMOUNT WAS PAID, MENTOR GRAPHICS AND ITS LICENSORS SHALL HAVE NO LIABILITY FOR ANY DAMAGES WHATSOEVER. THE PROVISIONS OF THIS SECTION 9 SHALL SURVIVE THE TERMINATION OF THIS AGREEMENT.

10. **HAZARDOUS APPLICATIONS.** CUSTOMER ACKNOWLEDGES IT IS SOLELY RESPONSIBLE FOR TESTING ITS PRODUCTS USED IN APPLICATIONS WHERE THE FAILURE OR INACCURACY OF ITS PRODUCTS MIGHT RESULT IN DEATH OR PERSONAL INJURY ("HAZARDOUS APPLICATIONS"). NEITHER MENTOR GRAPHICS NOR ITS LICENSORS SHALL BE LIABLE FOR ANY DAMAGES RESULTING FROM OR IN CONNECTION WITH THE USE OF MENTOR GRAPHICS PRODUCTS IN OR FOR HAZARDOUS APPLICATIONS. THE PROVISIONS OF THIS SECTION 10 SHALL SURVIVE THE TERMINATION OF THIS AGREEMENT.

11. **INDEMNIFICATION.** CUSTOMER AGREES TO INDEMNIFY AND HOLD HARMLESS MENTOR GRAPHICS AND ITS LICENSORS FROM ANY CLAIMS, LOSS, COST, DAMAGE, EXPENSE OR LIABILITY, INCLUDING ATTORNEYS' FEES, ARISING OUT OF OR IN CONNECTION WITH THE USE OF PRODUCTS AS DESCRIBED IN SECTION 10. THE PROVISIONS OF THIS SECTION 11 SHALL SURVIVE THE TERMINATION OF THIS AGREEMENT.

12. **INFRINGEMENT.**

12.1. Mentor Graphics will defend or settle, at its option and expense, any action brought against Customer in the United States, Canada, Japan, or member state of the European Union which alleges that any standard, generally supported Product acquired by Customer hereunder infringes a patent or copyright or misappropriates a trade secret in such jurisdiction. Mentor Graphics will pay costs and damages finally awarded against Customer that are attributable to the action. Customer understands and agrees that as conditions to Mentor Graphics' obligations under this section Customer must: (a) notify Mentor Graphics promptly in writing of the action; (b) provide Mentor Graphics all reasonable information and assistance to settle or defend the action; and (c) grant Mentor Graphics sole authority and control of the defense or settlement of the action.

12.2. If a claim is made under Subsection 12.1 Mentor Graphics may, at its option and expense, (a) replace or modify the Product so that it becomes noninfringing; (b) procure for Customer the right to continue using the Product; or (c) require the return of the Product and refund to Customer any purchase price or license fee paid, less a reasonable allowance for use.

12.3. Mentor Graphics has no liability to Customer if the action is based upon: (a) the combination of Software or hardware with any product not furnished by Mentor Graphics; (b) the modification of the Product other than by Mentor Graphics; (c) the use of other than a current unaltered release of Software; (d) the use of the Product as part of an infringing process; (e) a product that Customer makes, uses, or sells; (f) any Beta Code or Product provided at no charge; (g) any software provided by Mentor Graphics' licensors who do not provide such indemnification to Mentor Graphics' customers; or (h) infringement by Customer that is deemed willful. In the case of (h), Customer shall reimburse Mentor Graphics for its reasonable attorney fees and other costs related to the action.

12.4. THIS SECTION 12 IS SUBJECT TO SECTION 9 ABOVE AND STATES THE ENTIRE LIABILITY OF MENTOR GRAPHICS AND ITS LICENSORS FOR DEFENSE, SETTLEMENT AND DAMAGES, AND CUSTOMER'S SOLE AND EXCLUSIVE REMEDY, WITH RESPECT TO ANY ALLEGED PATENT OR COPYRIGHT INFRINGEMENT OR TRADE SECRET MISAPPROPRIATION BY ANY PRODUCT PROVIDED UNDER THIS AGREEMENT.

13. **TERMINATION AND EFFECT OF TERMINATION.** If a Software license was provided for limited term use, such license will automatically terminate at the end of the authorized term.

13.1. Mentor Graphics may terminate this Agreement and/or any license granted under this Agreement immediately upon written notice if Customer: (a) exceeds the scope of the license or otherwise fails to comply with the licensing or confidentiality provisions of this Agreement, or (b) becomes insolvent, files a bankruptcy petition, institutes proceedings for liquidation or winding up or enters into an agreement to assign its assets for the benefit of creditors. For any other material breach of any provision of this Agreement, Mentor Graphics may terminate this Agreement and/or any license granted under this Agreement upon 30 days written notice if Customer fails to cure the breach within the 30 day notice period. Termination of this Agreement or any license granted hereunder will not affect Customer's obligation to pay for Products shipped or licenses granted prior to the termination, which amounts shall be payable immediately upon the date of termination.

13.2. Upon termination of this Agreement, the rights and obligations of the parties shall cease except as expressly set forth in this Agreement. Upon termination, Customer shall ensure that all use of the affected Products ceases, and shall return hardware and either return to Mentor Graphics or destroy Software in Customer's possession, including all copies and documentation, and certify in writing to Mentor Graphics within ten business days of the termination date that Customer no longer possesses any of the affected Products or copies of Software in any form.

14. **EXPORT.** The Products provided hereunder are subject to regulation by local laws and United States government agencies, which prohibit export or diversion of certain products and information about the products to certain countries and certain persons. Customer agrees that it will not export Products in any manner without first obtaining all necessary approval from appropriate local and United States government agencies.

15. **U.S. GOVERNMENT LICENSE RIGHTS.** Software was developed entirely at private expense. All Software is commercial computer software within the meaning of the applicable acquisition regulations. Accordingly, pursuant to US FAR 48 CFR 12.212 and DFAR 48 CFR 227.7202, use, duplication and disclosure of the Software by or for the U.S. Government or a U.S. Government subcontractor is subject solely to the terms and conditions set forth in this Agreement, except for provisions which are contrary to applicable mandatory federal laws.

16. **THIRD PARTY BENEFICIARY.** Mentor Graphics Corporation, Mentor Graphics (Ireland) Limited, Microsoft Corporation and other licensors may be third party beneficiaries of this Agreement with the right to enforce the obligations set forth herein.

17. **REVIEW OF LICENSE USAGE.** Customer will monitor the access to and use of Software. With prior written notice and during Customer's normal business hours, Mentor Graphics may engage an internationally recognized accounting firm to review Customer's software monitoring system and records deemed relevant by the internationally recognized accounting firm to confirm Customer's compliance with the terms of this Agreement or U.S. or other local export laws. Such review may include FLEXlm or FLEXnet (or successor product) report log files that Customer shall capture and provide at Mentor Graphics' request. Customer shall make records available in electronic format and shall fully cooperate with data gathering to support the license review. Mentor Graphics shall bear the expense of any such review unless a material non-compliance is revealed. Mentor Graphics shall treat as confidential information all information gained as a result of any request or review and shall only use or disclose such information as required by law or to enforce its rights under this Agreement. The provisions of this Section 17 shall survive the termination of this Agreement.

18. **CONTROLLING LAW, JURISDICTION AND DISPUTE RESOLUTION.** The owners of certain Mentor Graphics intellectual property licensed under this Agreement are located in Ireland and the United States. To promote consistency around the world, disputes shall be resolved as follows: excluding conflict of laws rules, this Agreement shall be governed by and construed under the laws of the State of Oregon, USA, if Customer is located in North or South America, and the laws of Ireland if Customer is located outside of North or South America. All disputes arising out of or in relation to this Agreement shall be submitted to the exclusive jurisdiction of the courts of Portland, Oregon when the laws of Oregon apply, or Dublin, Ireland when the laws of Ireland apply. Notwithstanding the foregoing, all disputes in Asia arising out of or in relation to this Agreement shall be resolved by arbitration in Singapore before a single arbitrator to be appointed by the chairman of the Singapore International Arbitration Centre ("SIAC") to be conducted in the English language, in accordance with the Arbitration Rules of the SIAC in effect at the time of the dispute, which rules are deemed to be incorporated by reference in this section. This section shall not restrict Mentor Graphics' right to bring an action against Customer in the jurisdiction where Customer's place of business is located. The United Nations Convention on Contracts for the International Sale of Goods does not apply to this Agreement.

19. **SEVERABILITY.** If any provision of this Agreement is held by a court of competent jurisdiction to be void, invalid, unenforceable or illegal, such provision shall be severed from this Agreement and the remaining provisions will remain in full force and effect.

20. **MISCELLANEOUS.** This Agreement contains the parties' entire understanding relating to its subject matter and supersedes all prior or contemporaneous agreements, including but not limited to any purchase order terms and conditions. Some Software may contain code distributed under a third party license agreement that may provide additional rights to Customer. Please see the applicable Software documentation for details. This Agreement may only be modified in writing by authorized representatives of the parties. Waiver of terms or excuse of breach must be in writing and shall not constitute subsequent consent, waiver or excuse.

Rev. 100615, Part No. 246066