SIMULATION OF RECTANGULAR, SINGLE-LAYER, COAX-FED PATCH ANTENNAS USING AGILENT HIGH FREQUENCY STRUCTURE SIMULATOR (HFSS)

Kunal Parikh

Thesis submitted to the Faculty of the Virginia Polytechnic Institute and State University in partial fulfillment of the requirement for the degree of

Master of Science
in
Electrical Engineering

Dr. Amir I. Zaghloul, Chair
Dr. William A. Davis
Dr. R Michael Buehrer

December 2003

SIMULATION OF RECTANGULAR, SINGLE-LAYER, COAX-FED PATCH ANTENNAS USING AGILENT HIGH FREQUENCY STRUCTURE SIMULATOR (HFSS)

Kunal Parikh

ABSTRACT

The Range Limited Antenna (RLA) is a device, which accurately estimates the range of incoming signals and rejects those that arrive from outside a certain, pre-determined range. This task is accomplished by using two multi-element arrays and applying direction finding (DF) algorithms on each of them. Rectangular, single-layer, coax-fed patch antennas are used as array elements for the specific purpose of tracking cell phones operating in the PCS band inside a given building. It is vital to ensure that the patch antenna is designed in such a manner that it resonates at the desired frequency.

This thesis introduces the Agilent High Frequency Structure Simulator (HFSS) as an effective tool for modeling electromagnetic structures. It presents a comprehensive and meticulous description of the process of modeling a rectangular coax-fed patch antenna in HFSS. Plots of S-parameter values are calculated and are compared with WIPL-D, which is another simulation software program, and with measurements performed at the George Washington University. Various important parameters of the HFSS simulation are varied and their effects are investigated to provide a deeper understanding of the program.

ACKNOWLEDGEMENTS

Table of Contents

List of Figures

Chapter 1

Introduction

1.1 Range Limited Antenna (RLA)

The work presented in this thesis is based on a project which aimed to design and analyze a device that would be able to estimate the range of all received signals and reject those that originate from sources outside a certain, pre-determined range. Such a device was given the name 'Range Limited Antenna'. Figure 1.1 illustrates the basic concept of the RLA.
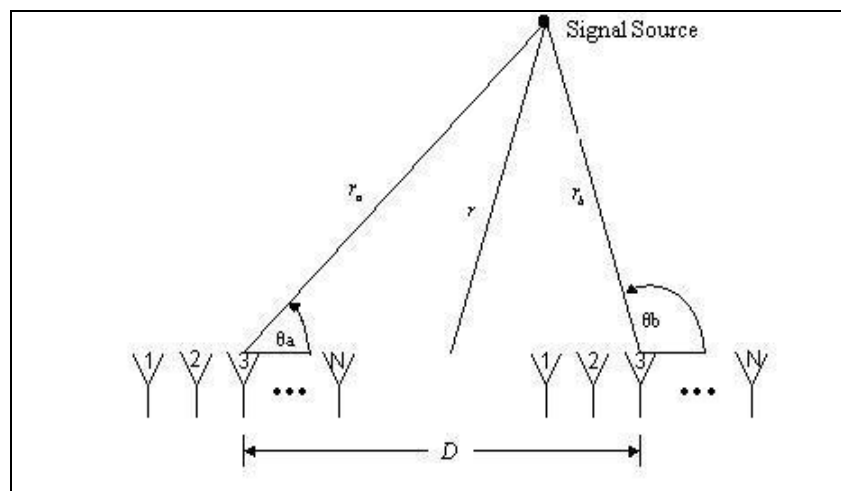
Fig 1.1: RLA Concept

As can be seen, the antenna is made up of two multi-element arrays. With the help of direction finding (DF) algorithms, each array estimates the angle of arrival (AoA) of the incoming signal at its center. These angles are shown as θ_a and θ_b. The distance between the centers of the two arrays D is known. Using the above 3 parameters and the laws of trigonometry, one can first estimate the distances r_a and r_b of the emitter from the centers of the left and the right arrays respectively. Next, the range r of the emitter from the center of the structure can easily be calculated. Once the range is known, signals coming from outside a desired range can be eliminated. Although not shown in the figure, the

1

signal propagates from the array elements to a PC via various hardware components such as phase shifters, mixers and power dividers. The corresponding DF algorithms such as Multiple Signal Classification (MUSIC) and Modified Root Pisarenko (MRP), which perform range estimation and emitter rejection, lie on the PC.

Although conceptually simple, the RLA design presented a number of challenges. To begin with, no apriori information about the source such as frequency band or modulation type would be available. The final estimated range is a trigonometric function of the difference in the estimated angles of arrival at the two arrays. Hence, any inaccuracies in AoA estimation would cause the range calculations to be significantly erroneous. The DF algorithms would also need to take into account the mutual coupling between the adjacent array elements and the effects of active element patterns. Ground reflections and multipath signals could cause inaccurate estimation of number of emitters, thereby considerably affecting the DF algorithm performance. Most algorithms assume simultaneous acquisition of data at all the array elements. Achieving this in practice would be extremely difficult. One of the major limitations of such algorithms is that they fail when the number of emitters is more than the number of array elements.

Having presented a brief introduction to the RLA project, the next section discusses the motivation for the work presented in this thesis.

1.2 Motivation

One specific application of the RLA that was looked at was for tracking and location of cell phones inside a given building. In other words, the RLA would identify all incoming signals from inside the building while rejecting others coming from different directions. Thus, the emitter would be in the PCS band. Specifically, the band considered was from 1.85-1.91 GHz. Microstrip patch antennas were chosen as the array elements because of their inherent robustness, high directional gain and narrow bandwidth. The next step then was to design the patch antenna with accurate dimensions so that it would resonate at the correct frequency in the band mentioned above. Therefore, it became necessary to use

simulation programs to test the performance of the patch before fabrication. As shall be discussed in the following chapters, the patch was designed to resonate at 1.904 GHz using the simulation program WIPL-D [1]. However, it was extremely important to establish some confidence level and to compare the WIPL-D results with some other program. That led to the modeling of the single layer coax-fed rectangular patch antenna in Agilent High Frequency Structure Simulator (HFSS), a detailed description of which follows in the next chapters.

1.3 The Finite Element Method and HFSS

In order to calculate the full three-dimensional electromagnetic field inside a structure and the corresponding S-parameters, HFSS employs the finite element method (FEM) [2]. FEM is a very powerful tool for solving complex engineering problems, the mathematical formulation of which is not only challenging but also tedious. The basic approach of this method is to divide a complex structure into smaller sections of finite dimensions known as *elements*. These elements are connected to each other via joints called *nodes*. Each unique element is then solved independently of the others thereby drastically reducing the solution complexity. The final solution is then computed by reconnecting all the elements and combining their solutions. These processes are named *assembly* and *solution* respectively in the FEM [3]. FEM finds applications not only in electromagnetics but also in other branches of engineering such as plane stress problems in mechanical engineering, vehicle aerodynamics and heat transfer.

FEM is the basis of simulation in HFSS. HFSS divides the geometric model into a large number of tetrahedral elements. Each tetrahedron is composed of four equilateral triangles and the collection of tetrahedra forms what is known as the finite element mesh. Figure 1.2 shows the finite element mesh for a sample horn antenna. This figure was taken from [2]. At each vertex of the tetrahedron, components of the field tangential to the three edges meeting at that vertex are stored. The other stored component is the vector field at the midpoint of selected edges, which is also tangential to a face and normal to the edge. Using these stored values, the vector field quantity such as the H-field or the E-

field inside each tetrahedron is estimated. A first-order tangential element basis function is used for performing the interpolation. Maxwell's equations are then formulated from the field quantities and are later transformed into matrix equations that can be solved using traditional numerical techniques.
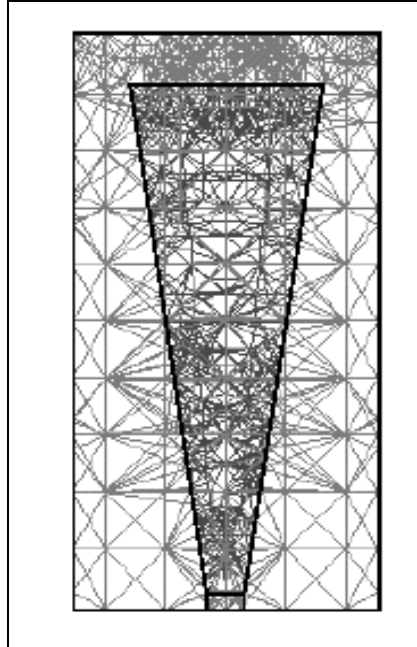
Fig 1.2: Finite element mesh for a horn antenna (printed by permission)

1.4 Organization of the Thesis

Following this introduction, Chapter 2 introduces HFSS and describes each step in the process of setting up a general simulation in HFSS. Chapter 3 illustrates the structure of the patch antenna that was built for the RLA project. A detailed description of the simulation of that antenna, which includes drawing the geometry, assigning vital boundaries and setting up the solution, is covered in that chapter. It also introduces the 3-element array of patch antennas that was simulated. Comparison of HFSS results with WIPL-D and with experimental data is presented at the end of Chapter 3. Chapter 4 aims to discuss the effect of modifying some key solution parameters on the results in HFSS. Ground plane size, patch thickness, accuracy level and mesh refinement frequency are the specific parameters considered. It also provides an introduction to the usage of fast

frequency sweep and the voltage source and shows how they compare against the use of discrete frequencies and port respectively. The final chapter presents some conclusions and recommendations for future work.

Chapter 2

An Understanding of HFSS

This chapter provides an insight into the various aspects involved in the process of setting up and running a simulation in HFSS. The version used is Agilent version 5.6. HFSS is a software package for electromagnetic modeling and analysis of passive, three-dimensional structures. It helps the user to observe and analyze various electromagnetic properties of the structure such as radiation patterns and scattering parameters. While it is not necessary to be an expert in numerical electromagnetics to use HFSS, it is important to understand each step of the modeling process in detail so as to obtain accurate and reliable results. This chapter aims to provide this understanding from a general point of view.

2.1 Process Overview

The first step is to draw the geometric model of the structure that is to be analyzed. The next step is to select the materials that the various drawn objects are made of. An accurate definition of boundaries for the structure, such as, perfect magnetic or electric conductor, follows next. In HFSS, a port or a voltage source needs to be defined to excite the structure. This is done as part of boundary definitions. Once the structure is completely modeled, the solution is set up. This includes definition of various parameters such as the frequency at which the adaptive mesh refinement takes place and the convergence criterion. Finally, after the completion of the simulation, the solution data is post-processed which may include display of far-field plots, Smith Chart graphs and tables of S-parameter data.

Figure 2.1 shows a snapshot of the first screen that the user encounters while building a new project. The *Project* menu deals with normal file management tasks such as saving and opening various projects. *Model*, *Materials*, *Boundaries*, *Solve* and *Post* take care of drawing, assigning materials, assigning boundaries, setting up a solution and post-

processing respectively. The remaining menus are used to define some miscellaneous parameters and they will be discussed later.
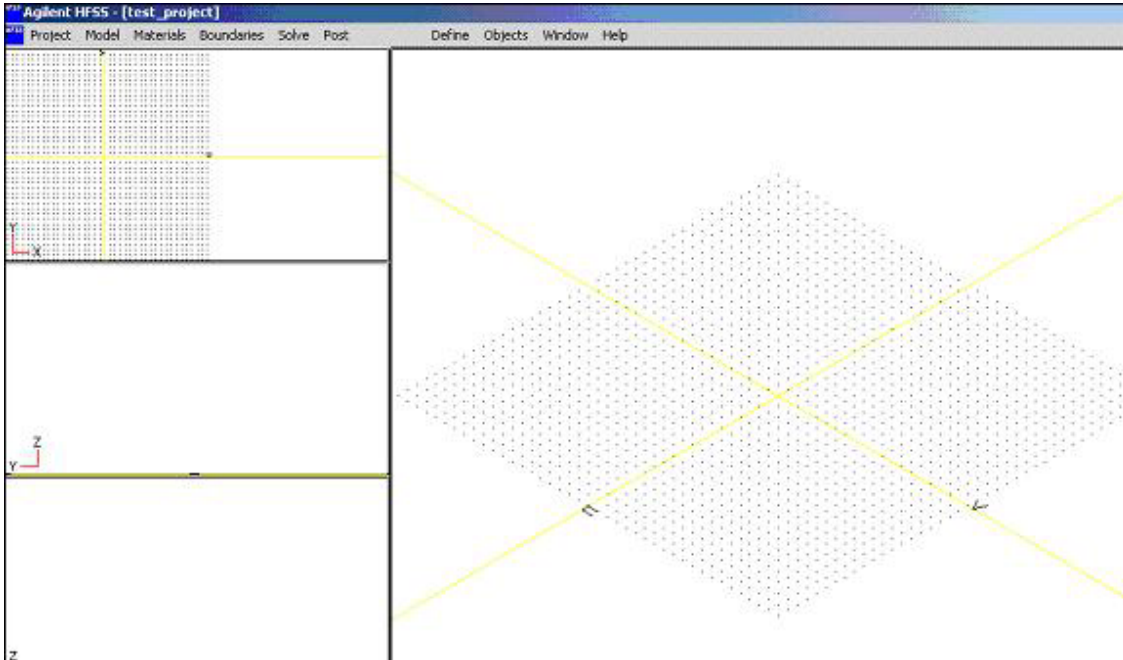
Fig 2.1: Snapshot of the HFSS environment

A detailed description of each of the above-mentioned steps follows in the sections below.

2.2 Drawing

The key to successful use of HFSS for solving any electromagnetic problem lies in the creation of the three-dimensional geometric model of the structure. The structure has to be visualized as a collection of two-dimensional (2D) and three-dimensional (3D) objects, each of which can later be assigned a specific material or a specific boundary as the need may. Also, it is extremely critical to ensure that the geometry is as simple as possible. This is so, because a more complex geometry would make the finite element mesh more complex, which in turn would require higher memory and processing power.

In order to facilitate easy and quick use, the drawing interface in HFSS is based on the industry-standard AutoCAD drawing tool. It also allows the user to import a 3D geometry that was created in any of a number of industry-standard formats such as layouts created in the Advanced Design System (ADS) version of Momentum. Selecting *Model >Draw* takes the user to the drawing interface. Here there are 4 menus namely *2D Objects*, *3D Objects*, *Object Library* and *Edit*.

A two-dimensional object is basically a polygon that lies in a single plane. Circles, ellipses and rectangles are examples of 2D objects which can be drawn directly in HFSS using menu commands. It is possible to create other 2D shapes by using polylines and arcs. 2D objects are important in HFSS because they can be used to model conductive surfaces such as a patch on a substrate, as shall be seen in the next chapter. They can be used as ports to identify where energy can enter or exit a structure. 2D objects can model openings in objects, such as a hole in a ground plane.

Consider the case of drawing a circle. After selecting *2D Objects > Circle*, one can position the pointer in the draw screen at the desired center of the circle and click the left mouse button. The pointer can then be moved to increase the circle to the desired diameter. At this point, clicking the left mouse button will open up the *Circle Template*, a snapshot of which is shown below.
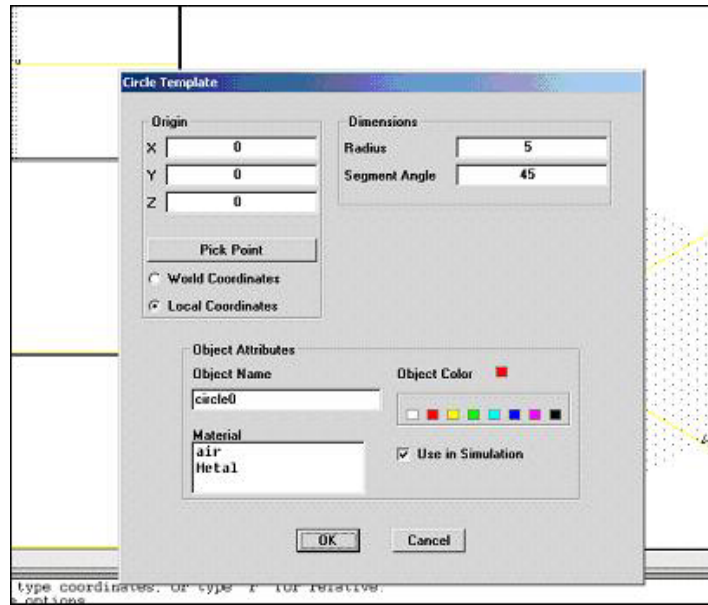
Fig 2.2: Snapshot of the Circle Template

In the template, the exact co-ordinates of the center of the circle can be specified with respect to a full, 3-dimensional x, y, and z space. Such co-ordinates are referred to as *World Coordinates*. Also, the exact value of the radius can be assigned. If it is preferred to work in a different orientation, *Local Coordinates* are used which are identified with a 2-dimensional u, v axis. Another important parameter to be specified here is the *Segment Angle*. Any curved object, whether it is 2D or 3D, is approximated by a certain number of line segments. *Segment Angle* is the number of degrees in each segment that constitute the curved object. For example, if it were specified as 60, it would mean that each segment occupies 60 degrees, and hence the circle is made up of 6 segments. Thus, the definition of the segment angle is a design trade-off that the user has to make because a very high value would mean that the curved object is approximated using a smaller number of segments leading to a coarse shape. On the other hand, a very small value of the segment angle might cause the curved object to be approximated by more segments than necessary resulting in added complexity to the structure.

As shown in Figure 2.2, the template also allows the user to specify some additional attributes to the object in question. Although HFSS provides a default name to each object, one can change it to a more relevant name using *Object Name*. The same holds

true for the color of the object. A material can be assigned to the object at this point; however, this can also be done using the *Materials* menu. The next section describes in detail the exercise of assigning materials. In some cases, it might be desired to exclude the object from the simulation and this can be accomplished by disabling *Use in Simulation*. The process of drawing 3D objects is identical to the one described above. It is possible to draw boxes, cylinders, cones and spheres using the *3D Objects* menu.

In most cases, it proves to be difficult to create a satisfactory three-dimensional model by just using the simple 2D and 3D objects mentioned in the above paragraphs. Inevitably, one feels the need to draw more complex objects. It is for this reason that the drawing interface in HFSS provides a few more options such as *Subtract*, *Unite* and *Intersect*. These commands can be used both for 2D as well as 3D objects. The *Subtract* command was used in removing the overlap between the inner and the outer conductors of a coaxial cable as shall be described in the next chapter. When this command is used, a core object and the object that is desired to be subtracted from the core object are selected. A copy of the latter is created, which is then subtracted from the former. In this way, the original object that was selected to be subtracted is retained. For all of the above 3 commands to work, the two objects in question must overlap, failing which, an error message appears. *Sweep*, *Revolve* and *Connect* are 3 additional commands that can be performed on 2D objects to create 3D objects. In addition to the commands, HFSS contains a library of commonly used parts which can be entered into complex structures with great ease. These parts are parameterized so that their dimensions can be changed to fit into the user's design. Rectangular and circular helixes, tapers, pyramids, spirals, waveguide twists, bridges and microstip components are the parts that can be found under the *Object Library* menu.

It is absolutely imperative for any user who hopes to carry out extensive analysis to be able to conveniently make changes to the once designed geometry. The *Edit* menu in HFSS provides this tool. Of utmost importance is the *Object Parameters* command, through which it becomes possible to view and change the dimensions of all the objects in the geometry. *Move*, *Copy* and *Delete* are some of the more commonly used commands under the *Edit* menu.

In the final geometric model of any structure, care should be taken that no two objects overlap or intersect with each other unless they are combined with the *Unite*, *Intersect* or *Subtract* commands. This is so because HFSS will not be able to determine the object that occupies the overlapping volume and hence will not be able to create the finite element mesh in the shared volume. It is valid for objects to share surfaces or be contained entirely within one another, as is the case when the *Subtract* command is used. If HFSS gives errors in the geometry, the user would have to come out of the drawing interface by the *File > Return to Main* command, and then use the *Model > Geometry Errors* command to locate the errors. In any case, when one exits from the drawing interface, one sees a template showing a list of all the 2D and 3D objects in the geometry. Also, a 'V' is assigned to each object that is visible, an 'S' is assigned to each object that is used in the simulation and a 'C' is assigned to each 2D object that is closed. Once the entire geometric model is ready, materials need to be assigned to each of the objects. This process is described in the next section.

2.3 Assigning Materials

While solving any structure for its electromagnetic properties, HFSS creates a finite element mesh for each object, which is based on the material that is assigned to the corresponding object. Thus, accurate assignment of materials is extremely essential. An object in HFSS can be assigned materials such as a lossless or lossy metal, an isotropic or anisotropic dielectric, a semiconductor or a resistor. A lossless metal is nothing but a perfect electrical conductor while an object can be defined a lossy metal by specifying a finite conductivity value. Permittivity, permeability, electric loss tangent and magnetic loss tangent are the properties associated with dielectrics. A dielectric can be specified as anisotropic by specifying how each of the above 4 properties varies with direction. An object can be defined as a semiconductor by choosing an appropriate value of either resistivity or conductivity along with values for permittivity and permeability. In order to model objects as resistors, values for resistivity would need to be entered in ohm-meters. HFSS organizes materials into a *Global Material List*, which is available to all projects, and a *Project Material List*, which is available only for the current project. It is advisable

to create any new material in the global database and then copy it over to the current project.

The global material database can be accessed by the *Materials > Global Data Base* command. By default, HFSS only creates the materials *air*, which is a dielectric with unity permeability and permittivity, and *metal*, which is a perfect lossless metal. Any other material would have to be manually created by the user. This can be done by typing the name of the material, selecting its type and then using the command *New Material*. Figure 2.3 shows a snapshot of the creation of a material named *test_material*, which is a dielectric. A new window opens where the properties of this dielectric can be defined. Existing materials can either be deleted or their properties can be changed using the commands *Delete Material* and *Edit Material* respectively. Once required materials have
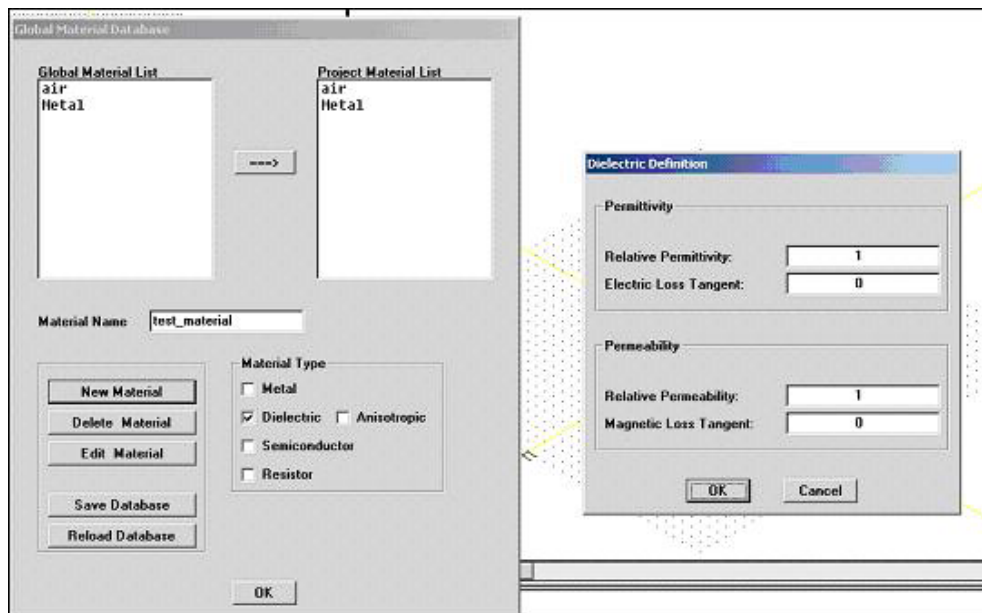
Fig 2.3: Snapshot of the Global Material Database and definition of a new material

been created, the global database should be saved. In order to use these materials in a specific project, they have to be moved to the *Project Material List* of that project before they can be assigned to individual objects. This can be done by accessing the global database from inside the project and using the ===> command. When the *Materials > Assignment* command is used from within the project, the window that opens shows a list

of all the objects in the geometry and a list of materials that have been moved from the global database to the project database. Each object can then be assigned the desired material by selecting the object and the corresponding material name.

The outer surfaces of an object need not be of the same material that makes up the entire volume of the object. These surfaces can be assigned different characteristics by assigning them appropriate boundaries, the process of which is described in the next section.

2.4 Assigning Boundaries

Among the various phases involved in setting up a simulation in HFSS, assigning boundaries is the most critical. This also includes exciting the structure, and hence any error can result in inaccurate results. A boundary can be assigned to any two-dimensional area such as a plane, a face of an object or an interface between two objects. Most boundary conditions are used to define electromagnetic characteristics such as conductivity or resistivity. *Port* is the only boundary condition which is used to define a surface that permits energy flow into and out of a structure. Hence, it shall be discussed later along with voltage sources. Figure 2.4 shows a snapshot of the window that opens up when the *Boundaries > Add/Modify* command is used.
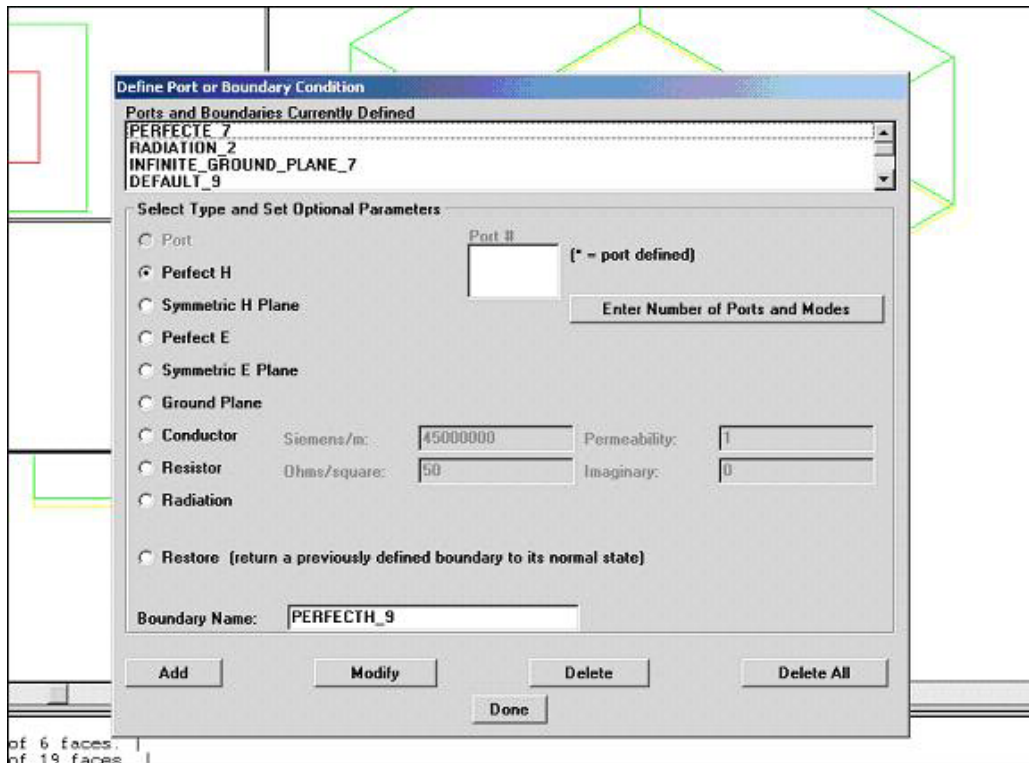
Fig 2.4: Snapshot of the Boundary Definition Template

As shown above, various boundary conditions can be declared in HFSS. *Perfect H* is a perfect magnetic boundary. It causes the magnetic field (H-field) to be in a direction normal to the surface that it is assigned to. Similarly, *Perfect E* is a perfect electric boundary or a perfect conductor and aligns the electric field (E-field) perpendicular to the defined surface. It should be kept in mind that the surface of any object, which is defined to be made up of a lossless metal, is automatically assigned the *Perfect E* boundary. It is possible to exploit the geometric and electromagnetic symmetry that frequently occurs in structures by using the *Symmetric H plane* or *Symmetric E plane* boundary. This boundary condition takes advantage of the fact that fields in one half of the structure are identical to those in the other half and thus helps simplify the simulation considerably. *Ground plane* is nothing but an infinite, Perfect E boundary condition. Properties of a surface assigned a *Conductor* boundary are similar to those objects which are made up of lossy metals. The *Resistor* boundary is similar to its counterpart described in the previous section. In order to analyze the far-field characteristics of a given structure, it is important that waves radiate out of the structure into space. From the point of view of a simulation,

these radiated waves would need to be absorbed, which is why, the *Radiation* boundary is used. In other words, the waves are absorbed by an object that is assigned a *Radiation* boundary, essentially, simulating radiation of waves into infinite space. *Restore* is used to revert a selected area on an existing boundary to its original material. This can be useful in modeling openings on structures such as a hole in a ground plane, as shall be seen in the next chapter.

Once any of the various boundary conditions described above is selected and the *Add* command is used, there are 4 different ways of identifying its location. *3-point Plane* enables the user to select 3 distinct points that form an entire plane. If one wishes to select a finite area within a plane and assign a separate boundary to that region, *3-point Bounded Plane* should be used. For instances like the Radiation boundary explained above, the entire outer surface of an object needs to be assigned a single boundary and this can be conveniently accomplished by specifying the object using *Object Name*. Sometimes, it is desired to assign a certain boundary to the area of overlap between two surfaces. It is for this purpose that the *Surface Intersection* command is provided.

Just as it is important to be able to view the dimensions of objects in the geometry, it is extremely essential to be able to view already defined boundaries. Using the *Display* command in the Boundaries menu opens a window that shows a list of all the defined boundaries. One can then select any particular boundary and the corresponding pattern definition by choosing a color, design and scale. The *Draw* command then will fill the surface to which the chosen boundary is assigned with the selected pattern. Figure 2.5 on the next page shows a snapshot displaying all the surfaces, which were assigned the Radiation boundary, in slant lines. It is also possible to select a visible boundary from the geometry and identify the name of the corresponding boundary condition. This is done using *Boundaries > Query*. For this command to work, the boundaries should be displayed as explained above.
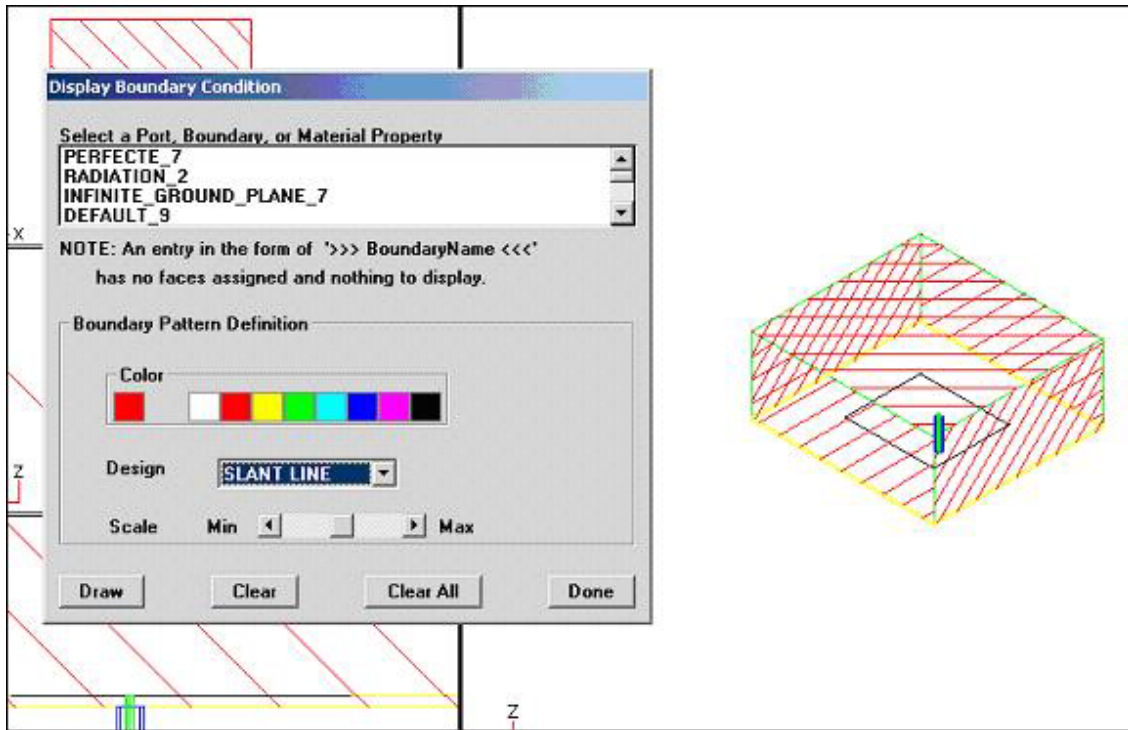
Fig 2.5: Snapshot showing the display of all surfaces assigned the *Radiation* Boundary

For any simulation to run, it has to be excited. In other words, energy should be able to enter the structure in some way. Ports and Voltage sources are tools in HFSS for this purpose. As can be seen in Figure 2.4, *Port* is also one among the various boundary conditions. In order to add ports, firstly the window shown in Figure 2.4 needs to be invoked. The *Enter Number of Ports and Modes* command will open a window shown in the snapshot of Figure 2.6. Based on the number of ports defined, a list will appear in the *Select a Port and Allocate its Modes* list. Each individual port can be selected from this list and the corresponding number of modes for that particular port can be defined. The default value of *Impedance Multiplier* is 1.This needs to be changed only if symmetry planes are used in the model because the computed impedances then will not be for the entire structure. Once this is done, the corresponding number of ports will show up in the *Port #* list. At this point, each port can be separately assigned to its corresponding surface in the same manner as was described for the other boundary conditions. If the port has been applied correctly, an asterisk appears next to its name in the *Port #* list and the port appears in the *Ports and Boundaries Currently Defined List*.

16
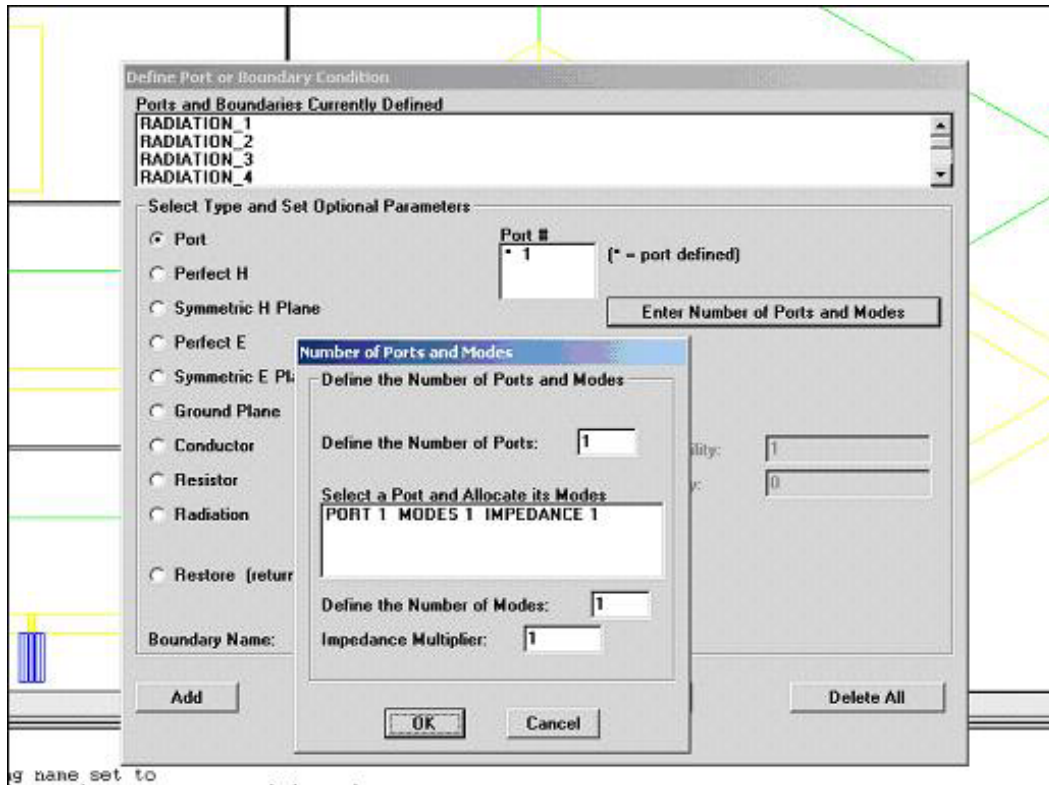
Fig 2.6: Snapshot showing the definition of 1 port with 1 mode

All the ports that are added to the structure need to be calibrated. Calibration basically determines the direction and polarity of fields and is required to carry out voltage calculations. This process of calibration can be carried out by using *Boundaries > Port Calibration*. For each mode of each port, different types of lines can be defined such as calibration lines, polarization lines and impedance lines. These lines have to be defined for all modes on all ports unless the structure has only one port and one mode. Also, using the same command, the user can run a ports-only solution, without running the entire simulation. This can help to identify the number of modes propagating on a port and sometimes it is also advisable to run a ports-only solution to identify the kinds of lines that are applied during calibration and their direction.

There are few important aspects to be taken care of while defining ports. Firstly, only the surfaces of the structure which are exposed to the background can be defined as ports. Secondly, a port must lie in a single plane. Finally, each port requires a length of uniform

cross section added to it because HFSS assumes that each defined port is connected to a semi-infinitely long waveguide.

In addition to ports, voltage sources can also be used to excite structures in HFSS. In some specific cases, it turns out to be more advantageous to use voltage sources instead of ports. As already mentioned, one of the limitations of ports is that they cannot be used in the interior of structures. This is where voltage sources can be applied. Also, there are instances, where the device responsible for injecting energy is not physically part of the structure being simulated. In such cases, a voltage source can be connected so that it not only excites the structure but also effectively models the coupling effects.

A voltage source can be added by the *Boundaries > Voltage Sources* command. Figure 2.7 shows the snapshot of the voltage source template. Once the number of sources and the source name are specified, the source position has to be identified. This can be done in 3 different ways. The most common method is to enter the co-ordinates of the start and
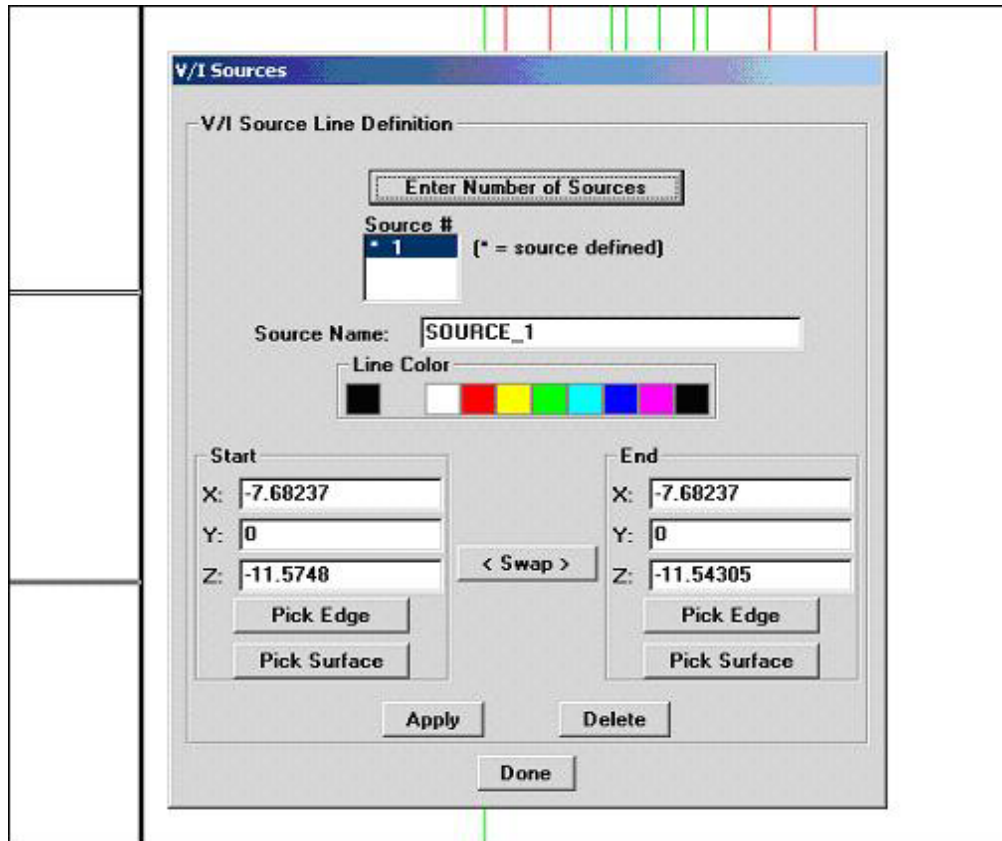
Fig 2.7: Snapshot for voltage source definition

end points. The *Apply* command should be used not only after entering the co-ordinates each time, but also one more time finally so that the voltage source will be defined correctly and an asterisk sign will show up next to the source name. The current flow from the source is such that the base is negative and the tip is positive. If need arises, this can be changed by using the *Swap* command. Instead of defining individual points, it is also possible to pick surfaces or edges from the geometry.

The most important consideration while defining voltage sources is the fact that a voltage source can only be defined between two 2D areas. Thus, a source can be assigned to either a 2D object, a surface of a 3D object or an existing boundary. Also, it should be made absolutely sure that both the start and the end points of the source lie on 2D areas. Once every object in the geometry is assigned a material, at least one port or voltage source is applied to the structure and other boundary conditions are identified, the user is ready to set up the solution.

19

2.5 Setting up the Solution

As mentioned in the previous chapter, HFSS uses the finite element method for performing calculations. This method is implemented by creating a mesh that breaks down a structure into small cells. Choosing the mesh parameters and selecting the frequencies at which the structure will be solved are the two most important parameters of setting up the solution.

Using *Solve > Setup* opens a window that has 4 tabs namely, *Refinement*, *Frequencies*, *Save Solutions* and *Advanced*. This window might take some time to open, and if a lot of programs are open, it might open behind all of them, that is, on the desktop. Figure 2.8 shows a snapshot of the window when the *Refinement* tab is selected. Now, HFSS begins its calculations with a certain mesh. However, the first mesh that is created is a very coarse one and the solution based on it might not be very accurate. Hence, using mesh refinement is highly advisable. The next parameter that needs to be selected is the starting mesh. Now, if the simulation is being run for the first time, the only option available is to use the *initial* mesh. Such a mesh is created by using the vertices of the objects as the vertices of the mesh cells. In cases where the simulation has been run earlier for a few iterations, one could use either the *previous* mesh or the *last* mesh. Selecting *previous* one
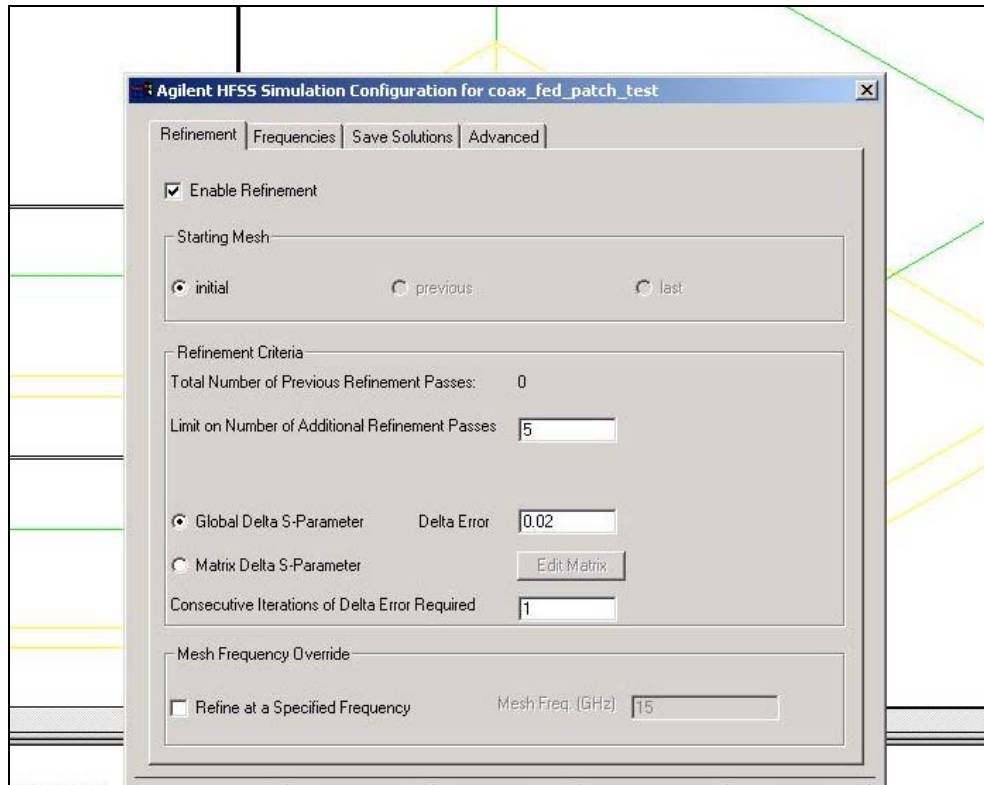
Fig 2.8: Snapshot of the *Refinement* Tab in the solution setup

would cause the simulation to start with a mesh that was created prior to the last one. This is important because when the simulation finally converges, it performs one more iteration and thus refines the mesh one more time to confirm that convergence has been achieved. In such a scenario, the penultimate mesh may not only be adequate to continue the simulation but also significantly less complex than the last one. It is possible to view the S-parameter values after each iteration and if one finds that the difference in their values between the last two refinement iterations is negligible, there is all the more reason to resume the simulation with the *previous* mesh. Sometimes a simulation may not converge and the user may want to perform subsequent refinements based on the last mesh that was created. This can be accomplished by selecting the *last* command.

The next step is to specify the refinement criteria. The idea is to calculate the change in the magnitude of the vector difference of S-parameters between two consecutive iterations and this value is called the *Delta Error*. There are two options available in specifying this error metric. The first one is to apply the same value to all the S-

parameters calculated by selecting *Global Delta S-Parameter*. The second choice is to select *Matrix Delta S-Parameter* which allows one to assign a separate value to each S-parameter. The *Edit Matrix* command opens up a window where one can specify the delta magnitude and phase for each S-parameter. It is also necessary to specify the number of consecutive iterations for which the error criterion should be met. In order to achieve a higher degree of convergence, one must specify *Consecutive Iterations of Delta Error Required* to be more than 1, which is the default. Another way of stopping the simulation is to specify the maximum number of iterations. Thus, if the number chosen in *Limit on Number of Additional Refinement Passes* is met, the simulation will stop even if the specified accuracy has not been achieved.

A significant aspect of setting up the solution is that mesh refinement is performed only at one frequency. Thus, even though the user may want the solution at multiple frequencies, HFSS will try to attain convergence for only a single frequency. By default, this is the highest frequency specified in the simulation. This can be changed by selecting *Refine at a Specified Frequency* and then entering the desired frequency. For example, it is best to choose the resonant frequency as the mesh refinement frequency while calculating the return loss of a patch antenna.

The next step is to select the frequencies using the *Frequencies* tab. Here again, one can choose to set up a Fast Frequency Sweep (FFS) or select Discrete Frequencies or both. FFS is a quick method of solving for a frequency sweep. This is so, because it takes a minimal number of frequency samples and then compares them to a rational fitting model. In other words, it performs interpolation on the data based on a small number of frequency values and wherever data variations are higher, it takes more samples. When the data at the sampled points fits the model, the frequency sweep is complete. It can thus be seen that when FFS is used, the user has no control over the number of frequencies as well as the specific frequencies for which the solution will be generated. Both these parameters will be governed by the performance of the data at the sampled points against the fitting model. Figure 2.9 on the next page shows a snapshot of the *Frequencies* tab in the solution setup.
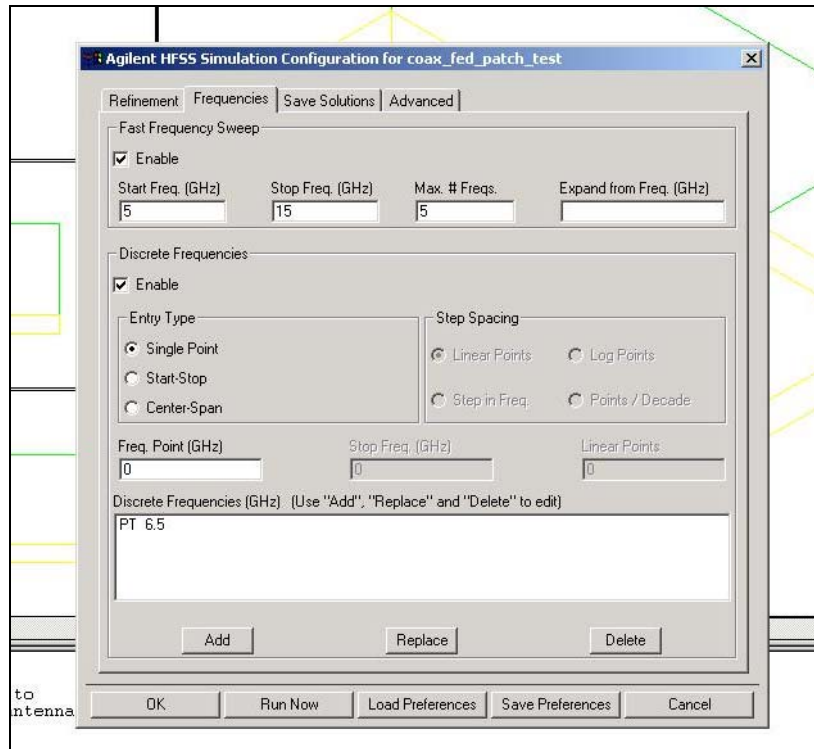
Fig 2.9: Snapshot showing the *Frequencies* Tab in the solution setup

It is possible to limit the number of frequencies used in an FFS to the value entered in the *Max # Freqs* field. Also, if it is desired to generate the solution at a certain frequency, that can be entered in the *Expand from Freq. (GHz)* field. The FFS will then begin at this frequency. If no frequency is specified, the FFS will begin at the midpoint of the sweep.

As opposed to the concept of FFS, if it is necessary to collect solution data at fixed and specific frequency points, discrete frequencies should be used. One can specify a single frequency by selecting *Single Point*. A group of frequencies can be simulated by selecting either *Start-Stop* or *Center-Span*. In any case, the spacing between frequencies will have to be specified. For example, with *Start-Stop* and *Linear Points* selected, specifying *Freq.Point (GHz)* as 2, *Stop Freq (GHz)* as 3 and *Linear Points* as 5, the solution will be generated at 2 GHz, 2.25 GHz, 2.5 GHz, 2.75 GHz and 3.0 GHz. It should be kept in mind that only the field data and S-parameters are calculated at each frequency and that mesh refinement is still done only at one frequency.

After setting up mesh refinement and selecting the frequencies, there are a few minor details that the user should be aware of. In the *Save Solutions* Tab, one can choose to save only the field solutions associated with the dominant mode as opposed to saving all modal solutions. Similarly, an option is available to save the field solution only for the last frequency point in the simulation as compared to saving field solutions for all the discrete frequency points. One should not forget to select *Compute Far Field Solution* in the same tab otherwise far-field calculations will not be performed. In the *Advanced* Tab, there are various mesh seeding parameters, which are usually set to default values. Finally, there is flexibility to perform aggressive mesh refinement. This causes more points to be added to the mesh after each refinement, thereby, taking smaller steps between meshes as compared to standard mesh refinement. It is believed to be advantageous for solving large structures.

And last but not the least, the *Run Now* command in Figure 2.9 will start the simulation. Another way is to come out of the Setup template and select *Solve > Run*. As soon as the simulation is started, a new window opens which shows the progress of the simulation and this is called the *Job Control* window. This window shows the status of the simulation including the current iteration, data of past iterations such as resulting error, and some general parameters such as frequency of refinement. It should be monitored closely to keep track of the simulation.

2.6 Post Processing the Solution Data

The status in the Job Control window will change to *DONE* when the simulation is over. At this point, the user is ready to view and analyze the solution data. Post > Start Post Processor is the command which invokes the post processor window. Figure 2.10 shows a snapshot of the default post processor window. There are various menus in this window that facilitate one to look at various types of data such as S-parameters, E and H fields, far fields, antenna parameters and transmission line data. As can be seen in Figure 2.10, on the bottom left of the screen is a list of different parameters which can be plotted in the post processor window. When the window is first invoked, the 3D model of the structure is plotted and the plot type is *Objects*. Based on the type of the plot, there are

different sets of controls available. In case of the geometry, it can be seen that commands such as *Views* and *Rotation* are present, which provide sufficient flexibility in viewing the structure. It is possible to display 4 different types of plots simultaneously on the screen using the *Window > Tile* command. When this is done, one particular plot has to be selected to apply the corresponding controls on it.
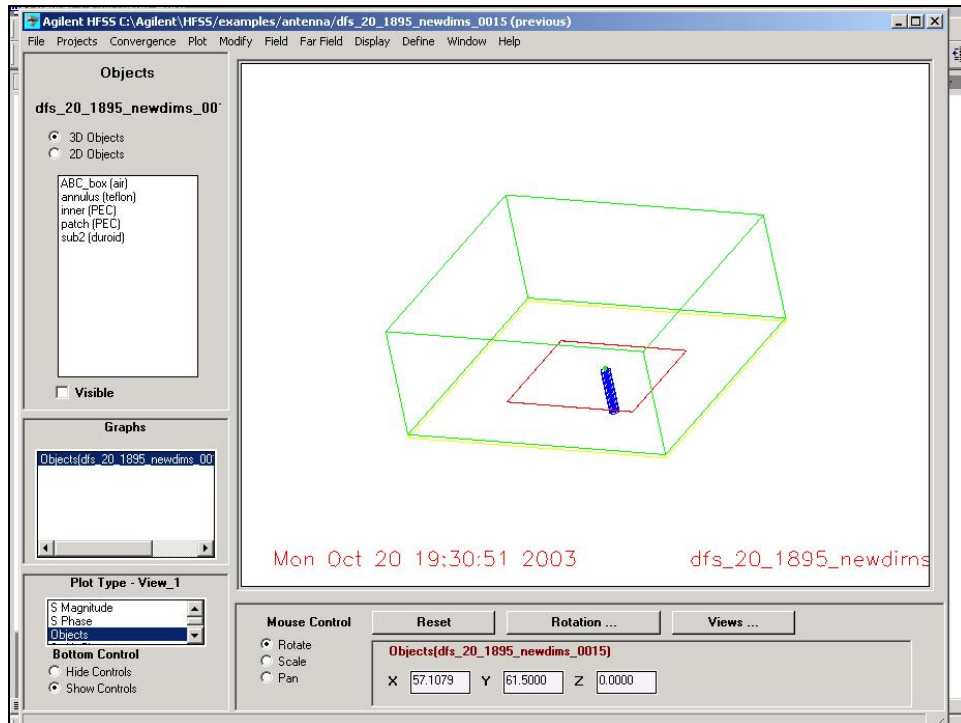
Fig 2.10: Snapshot of the default Post Processor Window

One of the most important and commonly used menus is the *Plot* menu. Many important characteristics such as magnitude and phase of S-parameters, Smith's Chart, propagation constant gamma and characteristic impedances of ports can all be plotted versus frequency. Data of the above parameters in tabular format can also be viewed. For example, *Plot > S Mag Plot* will plot the magnitude of the S-parameters for the whole sweep of frequencies specified. On the other hand, *Plot > Matrix* will provide the values of the whole S-Matrix for each frequency in a tabular format. It is important to note that when mesh refinement is enabled, S-parameters will be calculated based on the last mesh and the one prior to the last one. While plotting, the user will have to select from two sets, one of which has *last* in parenthesis and the other has *previous* in parenthesis. Also,

when a view showing a plot is selected, commands are available to change the properties of the axes, to edit the legend and to format the color and type of lines used in the plot. Figure 2.11 shows a snapshot in which the magnitude of S-parameters has been plotted with respect to frequency.
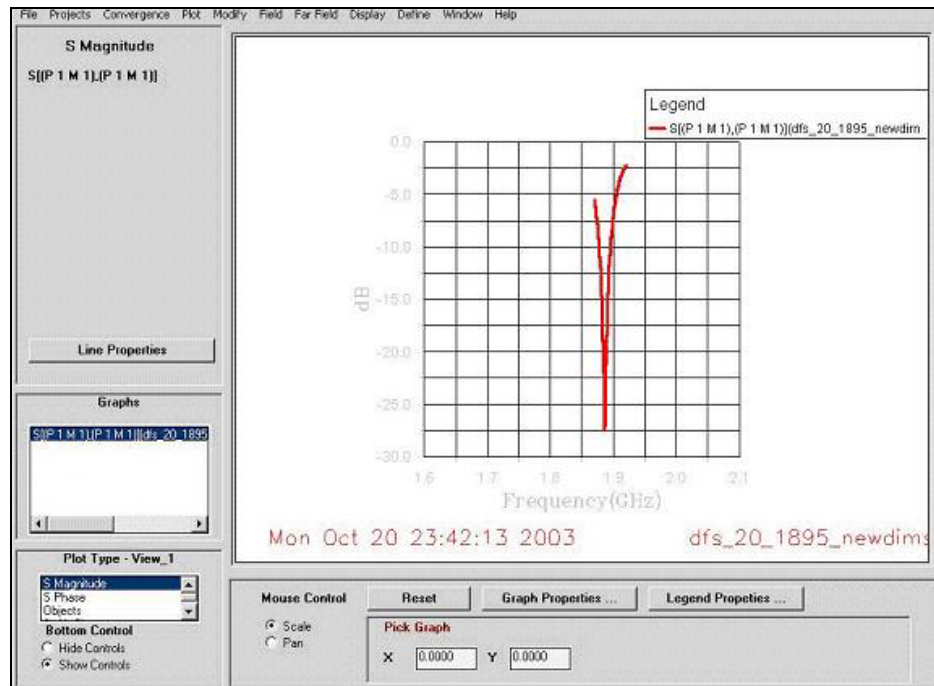
Fig 2.11: Snapshot showing a plot of the magnitude of S-parameters versus frequency

In most cases, it is extremely interesting to analyze the radiation characteristics of electromagnetic structures. The *Far Field* menu is used for this purpose. Selecting *Far Field > Far Field Plot* will create a 3D plot based on the maximum E-field. Such a plot is extremely useful in many cases. For example, the 3D far-field plot of an ideal dipole can be seen to have the shape of a doughnut. In more complicated cases, it is usually desired to view 2D polar plots of the far-field with respect to theta (θ) or phi (ϕ). This is a two-step process. Firstly, *Cut 3D Far Field* is selected which opens a template. In this template, one has to select either *Theta Cut* or *Phi Cut*. If the goal is to plot a far-field pattern versus ϕ for a specific value of θ, *Theta Cut* should be selected and the corresponding constant value of θ should be specified. The next step is to select the *Plot Far Field Cut* option under the *Far Field* menu. This will open up a window where the user has to choose the specific E-field parameter whose plot is desired. In addition, one

26

has to decide whether the plot should be in Cartesian or polar format. It is also possible to normalize the plot, use a logarithmic scale and specify the minimum value to be used in the plot. Figure 2.12 shows a snapshot of the E field plot versus θ when φ was specified to be 0.
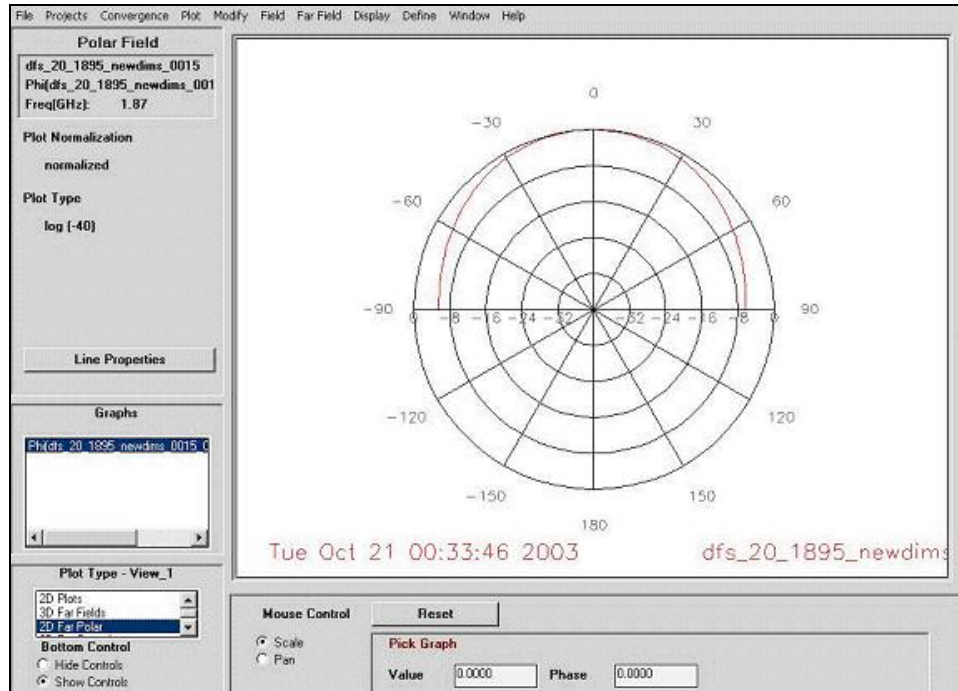
Fig 2.12: Snapshot of a 2D far-field plot

Comparing Figures 2.10, 2.11 and 2.12, it should be noted how the data on the left hand side of the plot and the corresponding controls change for each plot type. Finally, *Far Field > Antenna Parameters* shows the values of some important quantities such as radiated power, directivity, gain and effective angle.

Some of the remaining menus have some useful commands. The *File* menu has commands which make it possible to export data from HFSS to files so that it can be saved. Every now and then, it is required to compare plots of various projects created in HFSS. The *Projects* menu has commands to read other projects, select each of them and then plot corresponding data simultaneously on one window. It may be of interest to study the convergence behavior of the simulation. In other words, the user may want to view the S-parameter values after each iteration. This can be accomplished by using

commands from the *Convergence* menu. The *Job Control* window, which was described in the previous section, can also be viewed from this menu. In cases involving multiple ports or voltage sources, it may be desired to set different weights on different excitations. An example of this is the calculation of active element patterns in an array environment where only the active element should be excited. The *Set Excitations* command in the *Fields* menu is used for this purpose. While working with multiple types of data, there will always be a need to erase plots from a given view and plot them again. This can be done using the *Window > Erase Plot* command. Also, the *Window > Preferences* command should be used in order to choose colors of the background and the axes along with their labels.

There are certain miscellaneous commands in HFSS, which can be used at any point in the entire process of creating a simulation. These are very useful commands and need not be necessarily used as a part of any particular step like the ones described above. From the main window environment shown in Figure 2.1, *Project > Project Manager* is used to manage all existing projects in HFSS. It is from here that one can create new projects, open existing ones, copy or delete projects, create new folders and rename projects or folders. It is also possible to open and analyze some projects which are created in HFSS as examples. The *Object* menu in the main window has commands which can be used to change the properties of objects such as their name, color, whether the object should be visible and whether it should be used in the simulation. Again, this can be done at any time during the process and does not necessarily have to be done while drawing. Similarly, the *Window* menu has commands to change some general features of the environment such as the number and layout of the views, directions of viewing and zoom levels. Commands are also available to specify project preferences such as units of measurement while drawing, the extent of axes in the drawing window and the spacing between consecutive grids. Some system level preferences like colors and font used in the working environment and some file settings can also be modified using this menu.

2.7 Conclusion

In this chapter, an attempt was made to provide an understanding of HFSS to someone who might be using it for the first time. Stress was laid on explaining the design methodology used in approaching any simulation in HFSS. Sufficient information on each step is provided so that a new user can get started and is exposed to all the basic commands. However, this chapter does not claim to be a detailed manual of HFSS. The interested reader is encouraged to study [2] and [4] for a complete understanding of all the features in HFSS.

Chapter 3

Simulation Details and Comparison of HFSS with WIPL-D and Measurements

This chapter begins with a brief introduction to microstrip antennas in general. It then describes the geometry and dimensions of the rectangular, single-layer coax-fed patch antenna that was built and simulated as part of the Range Limited Antenna (RLA) project. The major part of the chapter deals with the exact details of simulating the above-mentioned structure in HFSS. The return loss calculated from the simulation in HFSS is then compared with that computed using WIPL-D and with experiments.

3.1 Overview of Microstrip Antennas

Modern printed circuit fabrication techniques have made it possible to build low profile antennas that are extremely useful. Such antennas are referred to as *microstrip* or *printed circuit* antennas. A microstrip antenna is made up of two parallel conductors that are separated by a dielectric substrate. The lower conductor usually acts as a ground plane and the upper conductor is a patch, which is why such antennas are also called *patch* antennas. The patch can be of various shapes such as rectangular, circular, square, elliptical, dipole and triangular among others [5]. Patch antennas are inherently resonant antennas characterized by extremely low bandwidths. In addition, they are usually light in weight and easy to install because of which they are highly used in aircraft, satellite and missile applications [5]. Simplicity, low manufacturing cost and the flexibility to configure to specialized geometries are some of the other advantages of patch antennas [6].

Substrate properties such as dielectric constant and thickness are important considerations in the design of microstrip antennas. Various substrates are available with values of dielectric constant between 2.2 and 12. Thick substrates with lower dielectric constant

result in better efficiency, larger bandwidth and a larger antenna size [5]. On the other hand, thin substrates with higher dielectric constant cause reduced efficiency, smaller bandwidths and smaller element sizes [5]. Of equal importance is the method which is used to feed the antenna. Microstrip line, coaxial probe, aperture coupling and proximity are the four commonly used methods [5].

3.2 Patch Geometry and Feed Details

As mentioned previously, the work presented in this thesis was mainly carried out as part of the Range Limited Antenna (RLA) project. The final goal of the project was to track and locate cell phones inside a building. In order to accomplish this, it was decided to use an array of rectangular, single-layer, coax-fed patch antennas. After repeated applications of WIPL-D for the single element case, the final patch dimensions and the substrate thickness were chosen such that the resonant frequency would be 1904 MHz [7]. The WIPL-D simulation was carried out by the George Washington University (GWU).
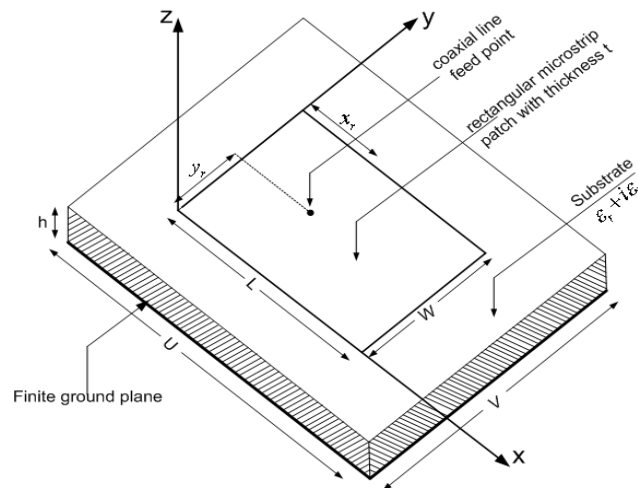
Fig 3.1 Geometry of the Rectangular patch antenna

As shown in Figure 3.1, L indicates the length of the patch and W the width. The thickness of the patch is denoted by t and the height of the substrate is h. u and v in the figure show the length and width of the ground plane respectively. The dimensions that were chosen by GWU are as follows [7]:

- $L = 51.22$ mm
- $W = 60$ mm
- $h = 1.5748$ mm
- $u = L + 40h$
- $v = W + 40h$
- $t = 35\mu m$

The material of the substrate was selected to be RT/duroid 5880 the relative permittivity of which is 2.2 and electric loss angle is 0.0004.

As mentioned in the previous section, various types of feed methods are possible. In this case, the coaxial probe method was used. In this method, the inner conductor of the coax is connected to the patch through the substrate while the outer conductor is attached to the ground plane [5]. With respect to Figure 3.1, x_r and y_r indicate the co-ordinates of the feed point, with $x_r = 0.35L$ and $y_r = W/2$. The cross-section of the patch antenna along with the coax cable is shown in Figure 3.2.
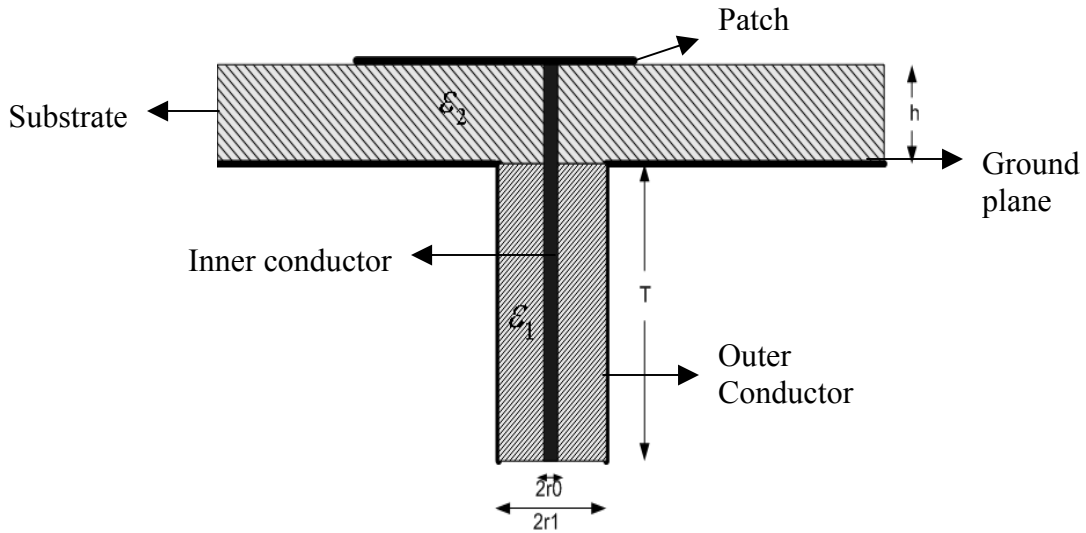
Fig 3.2: Cross-section of the patch antenna along with the coax cable

As shown in the figure, T is the length of the coaxial cable including the coaxial connector, $r0$ is the radius of the inner conductor and $r1$ that of the outer conductor. ε_1 is the dielectric constant of the coaxial line. The coaxial cable was modeled as a standard SMA connector with a characteristic impedance of 50 Ωs. For this purpose, the values of the above parameters were chosen as follows:

- $r0 = 0.635$mm
- $r1 = 2.0574$mm
- $\varepsilon_1 = 2.07$
- $T = 20$mm

With the structure and dimensions shown above, calculations were carried out in HFSS. The next section describes the various steps of the simulation in detail.

3.2 Simulation Details

This section provides explanation of each task that was involved in creating a simulation for the structure described above.

3.2.1 Drawing

Figure 3.3 shows the list of all the objects that were created. As mentioned in the previous chapter, this list can be accessed by using the *Edit > Object Parameters* command in the drawing interface.
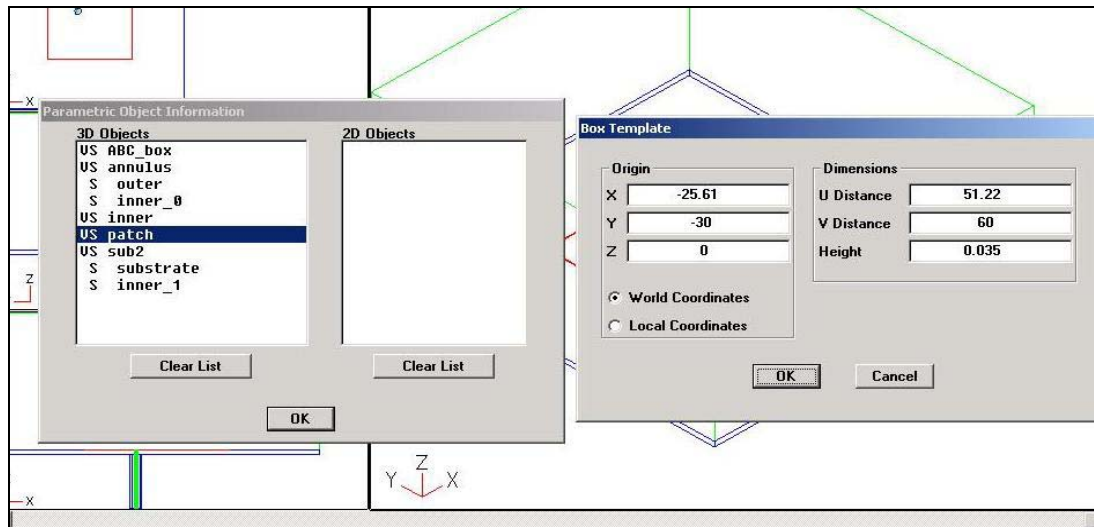
Fig 3.3: List of all objects and dimensions of the patch

As can be seen, all the objects are 3D objects. Also shown in the figure is the template that was used to create the patch. This was done using *3D Objects > Box*. The units of measurement were chosen to be millimeters. It should be noted that *World Coordinates* were used throughout the project and that the patch was centered at the origin. The dimensions of the patch are in accordance with those mentioned in the previous section. The patch was drawn in such a way that it extends from $z = 0$ to $z = t$. A similar box template was also used to draw the substrate. The length and width of the substrate is same as that of the ground plane and the substrate extends from $z = 0$ to $z = -h$. No separate object was used for the ground plane because it was modeled using a boundary condition as shall be described later.

The inner and the outer conductors of the coaxial cable were drawn using *3D Objects > Cylinder*. Figure 3.4 shows the template for the inner conductor.

Fig 3.4: Snapshot showing the dimensions of the inner conductor

The origin of the cylinder corresponds to the feed point location that was shown by x_r and y_r in Figure 3.1. The segment angle was specified as 45°, which means that the curved surface of the cylinder was approximated by an octagon. It can be seen that the inner conductor extends from $z = 0$ to $z = - (T + h)$. Thus, the inner conductor 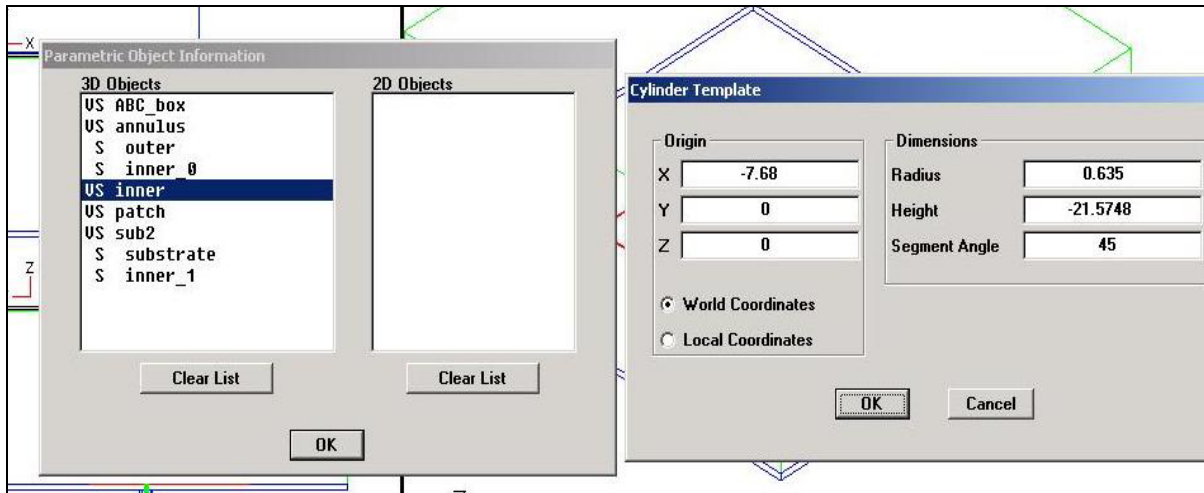is connected to the patch through the substrate according to the coaxial line method of feeding. The outer conductor was also drawn using a similar template. However, it extends from $z = -h$ to $z = - (T + h)$, thereby staying attached to the ground plane.

As mentioned in the previous chapter, HFSS does not allow any overlapping volume in the model. It is obvious from the above description that the inner conductor not only overlaps with the outer conductor but also the substrate. Hence, it became necessary to use subtraction of objects. Thus, using *3D Objects > Subtract*, the inner conductor was subtracted from the outer conductor and a new object was created by the name of *annulus*. Similarly, the inner conductor was subtracted from the substrate and the resulting object was named *sub2*. In the above subtraction process, 2 copies of the inner conductor are created automatically, namely, *inner_0* and *inner_1*. It is these copies that are actually subtracted while the original object *inner* is not only untouched but also used in the simulation. This is clearly visible in Figures 3.3 and 3.4.

As explained in the previous chapter, HFSS needs an object that is assigned the *Radiation* boundary in order to perform far-field calculations. This object should be exposed to the background and should be convex with respect to the radiating source. Also, it should be located at least a quarter of a wavelength away from the structure. In order to simulate a patch radiating into the half-space above the ground plane, a box that extended from $z = 0$ was used. The length and width of the box were same as that of the ground plane while the height was chosen to be 50 mm, which is slightly more than one-quarter of the wavelength associated with the lowest frequency specified in the simulation. This object is named *ABC_box*.

Figures 3.5 and 3.6 show the side view and top view of the entire structure respectively. It can be seen in Figure 3.5 that the inner conductor passes through the substrate while the outer conductor ends where the substrate begins. Similarly, the box begins where the substrate ends. In other words, the top surface of the substrate coincides with the bottom surface of the box. Although not seen in Figure 3.5, the bottom surface of the patch is same as that of the box.
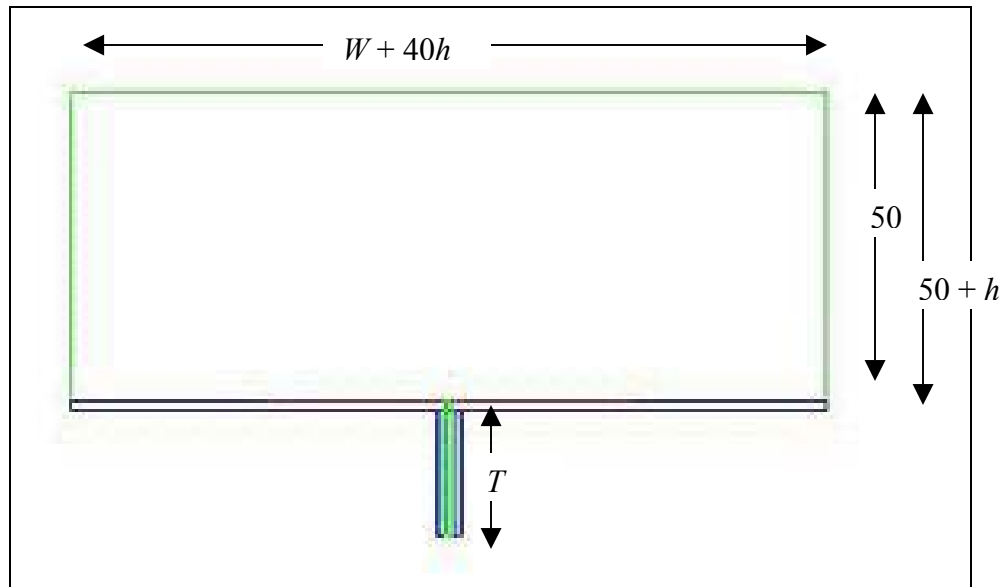
Fig 3.5: Side View of the Patch antenna with the coaxial cable
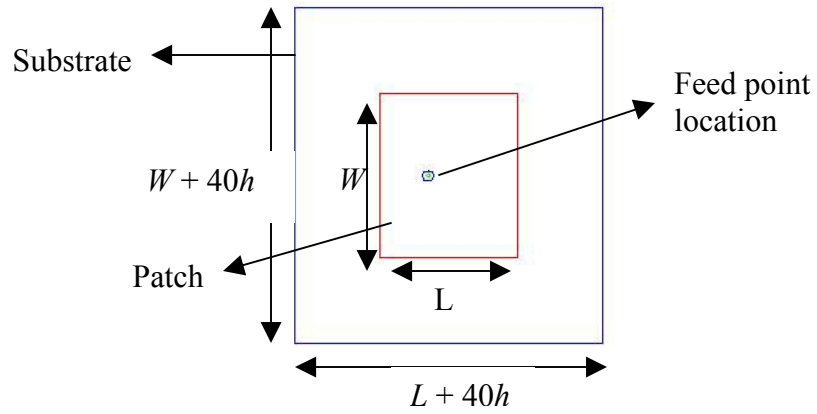
Fig 3.6:Top View of the Patch Antenna with the coaxial cable

Having described the geometry in detail, the next section highlights the materials that were assigned to each of the above objects.

3.2.2 Assigning Materials
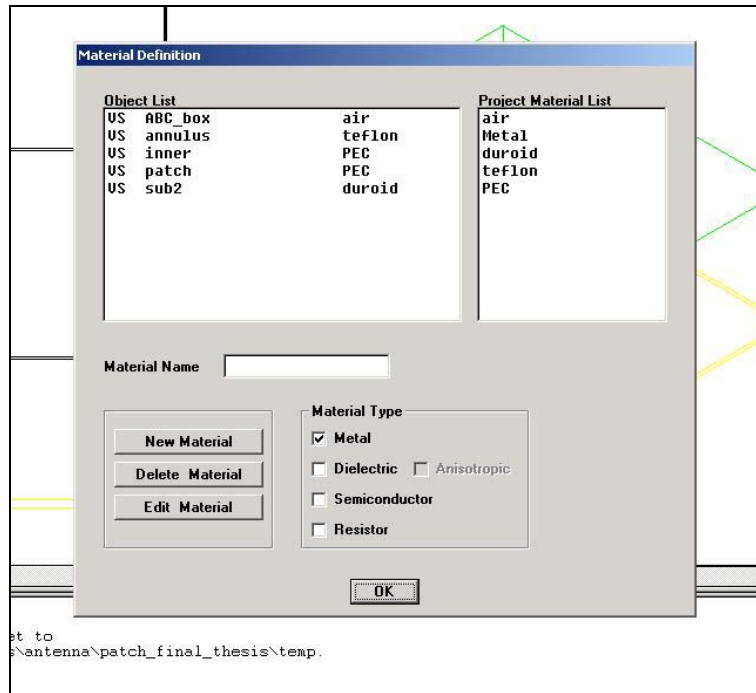
Fig 3.7:Snapshot showing the materials assigned to each object

Figure 3.7 shows a snapshot of the materials assigned to each object created in the drawing. This can be invoked by using the *Assignment* command in the *Materials* menu. *ABC_box* is the dummy object created for far-field calculations. Hence, it was assigned the material *air*, which is a dielectric with values of relative permittivity and permeability assigned to be 1. As explained earlier, *annulus* is the name of the object that remains after subtracting the inner conductor from the outer conductor. With respect to Figure 3.2, it should be assigned a material that has a dielectric constant of ε_1 (= 2.07). Hence a dielectric material by the name of *teflon* was created. Figure 3.8 shows its properties.
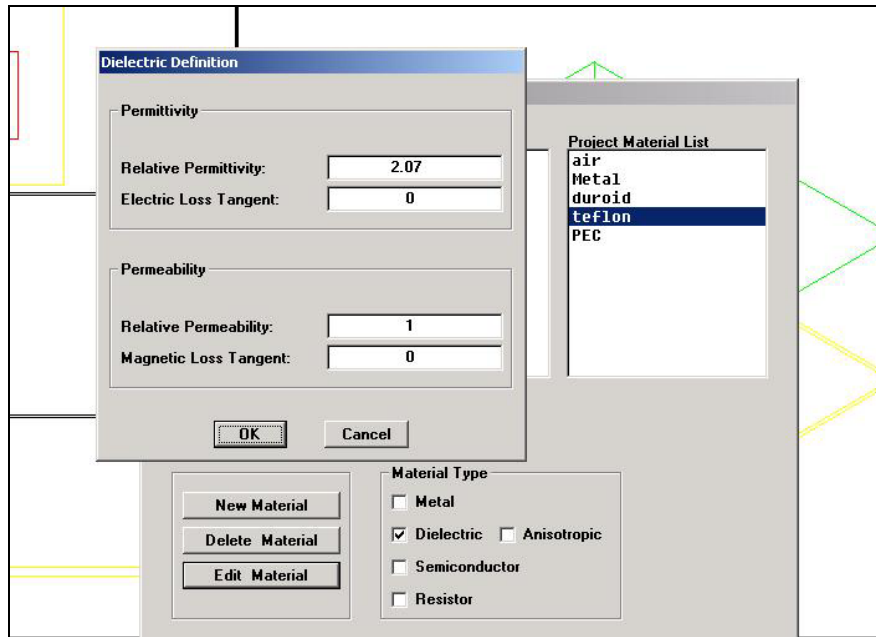
Fig 3.8:Snapshot showing the properties of material *teflon*

Both the inner conductor and the patch itself are perfect electric conductors (PECs). As a result, they were assigned a material by the name of *PEC*, which is basically a lossless metal. *sub2* denotes the substrate that was chosen to be made up of RT/duroid 5880. Hence, similar to teflon, a dielectric material called duroid was created having a relative permittivity of 2.2 and an electric loss tangent of 0.0004.

The next step in the process of setting up the simulation was to assign accurate boundary conditions to surfaces in the structure.

3.2.3 Assigning Boundaries

Although the object *annulus* is assigned a dielectric material, its outer surfaces need to be perfect conductors so as to simulate the coaxial cable accurately. It is for this purpose that boundary conditions are used in HFSS. Hence, the *Perfect E* boundary condition was assigned to *annulus* using *Object Name*. By doing so, its entire outer surface was modeled to be a perfect electric conductor. Slant lines show the above boundary condition in a zoomed view of the coaxial cable in Figure 3.9. It should be noted that the

39

inner conductor, which extends all the way up to the patch has already been chosen to be made up of metal. All snapshots in this section are created using *Boundaries > Display*.
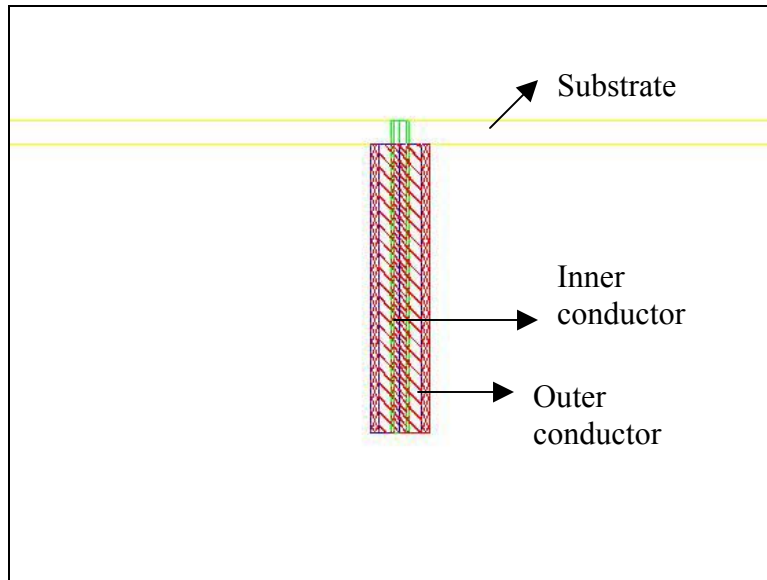
Fig 3.9:*Perfect E* boundary applied to *annulus*

As mentioned in the previous section, no object was created in the geometry that would represent the ground plane. Instead, the *Ground Plane* boundary condition was used. Comparing Figures 3.2 and 3.9, it can be seen that the bottom surface of the object *sub2* in the geometry corresponds with the location of the ground plane in the original structure. Hence, this particular surface was selected using the *3-point Plane* option and the *Ground Plane* boundary condition was applied. Figure 3.10, when viewed carefully, shows the slant lines starting and ending at the bottom surface of the substrate, thereby identifying the boundary location.
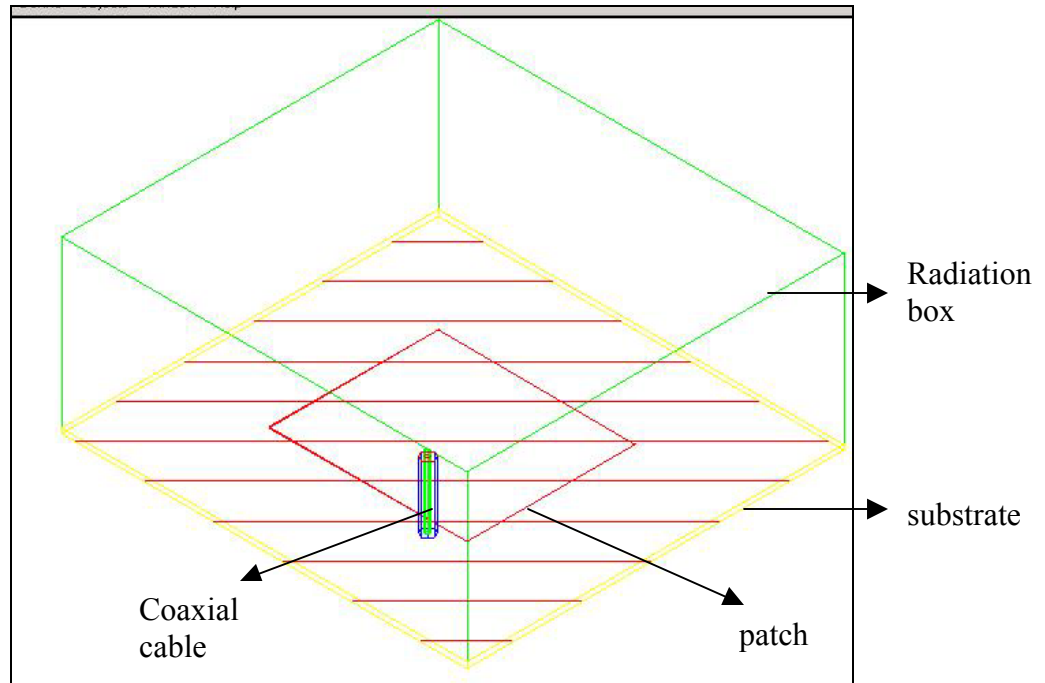
Fig 3.10:Boundary to model the ground plane
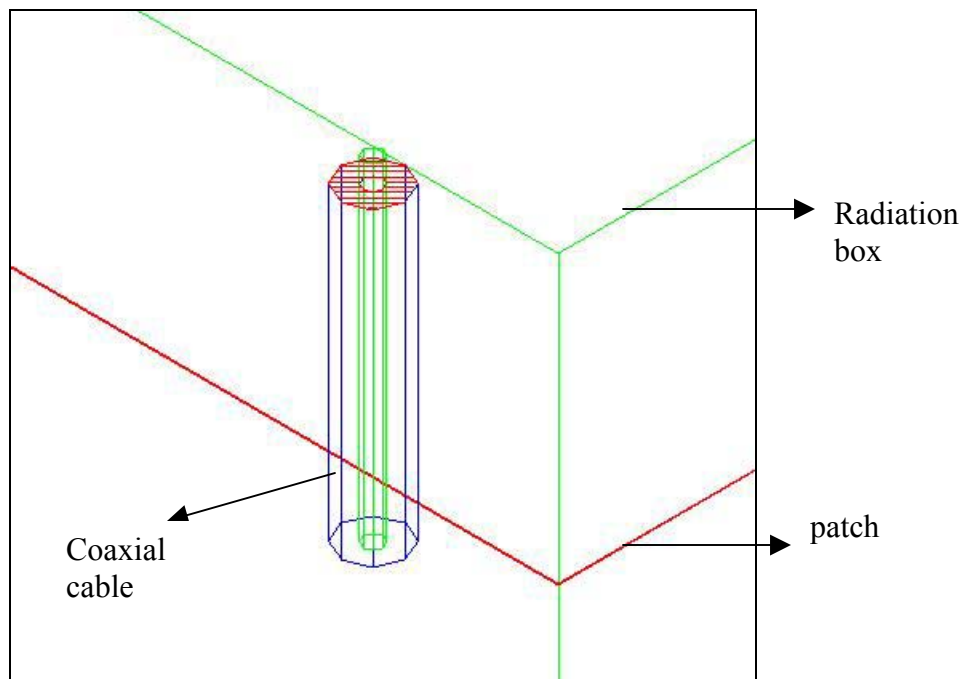
Fig 3.11:Boundary to model the hole in the ground plane

The next step was to open up a hole in the ground plane for the inner conductor of the coaxial cable to pass through and extend up to the patch. In order to accomplish this, the *Restore* boundary condition was used. The exact location was identified by specifying a common surface between two objects, namely, *annulus* and *sub2*. Figure 3.9 shows the reason for specifying the above two objects and the intersecting surface is shown in Figure 3.11 via slant lines.
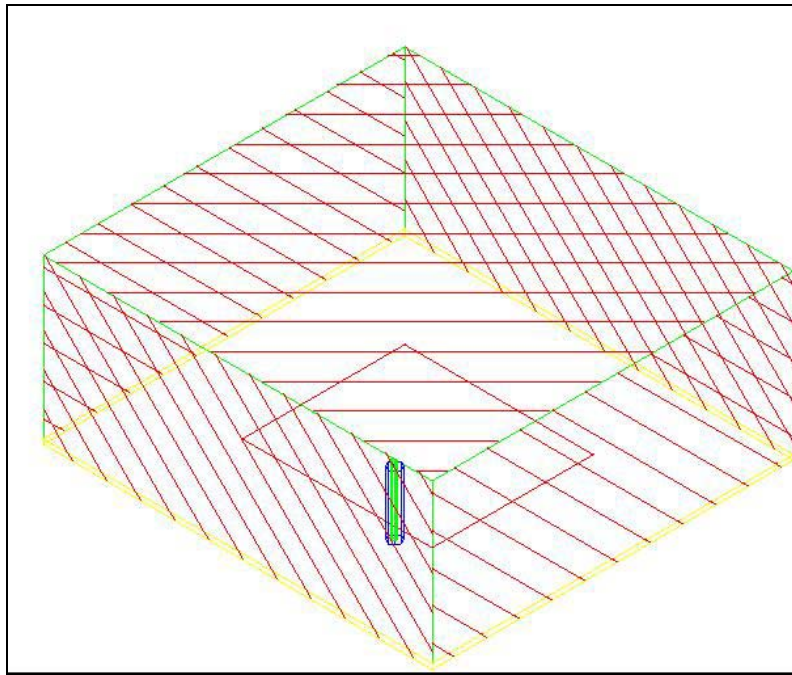
Fig 3.12:Surfaces assigned the *Radiation* boundary

As mentioned earlier, the object *ABC_box* was specifically created so that it can be assigned the *Radiation* boundary. However, the patch lies on its bottom surface and hence the boundary condition should not be applied to that surface. That is why each of the remaining surfaces needed to be individually selected and declared as *Radiation* boundaries as opposed to selecting the whole object. These surfaces are highlighted with slant lines in Figure 3.12. It should be pointed out that each side surface of the substrate lies right below and touches the corresponding side surface of the box. While assigning boundaries, both the touching faces are treated as one surface and the *radiation* boundary is assigned accordingly. This is visible in Figure 3.12.

The final step was to devise the means to inject energy into the structure. This was done by setting up a port. Only one port with one mode was required for this structure. As a result, there was no need to specify impedance, polarization or calibration lines. The logical location of the port was at the bottom surface of the coaxial cable. This is shown using slant lines in Figure 3.13.
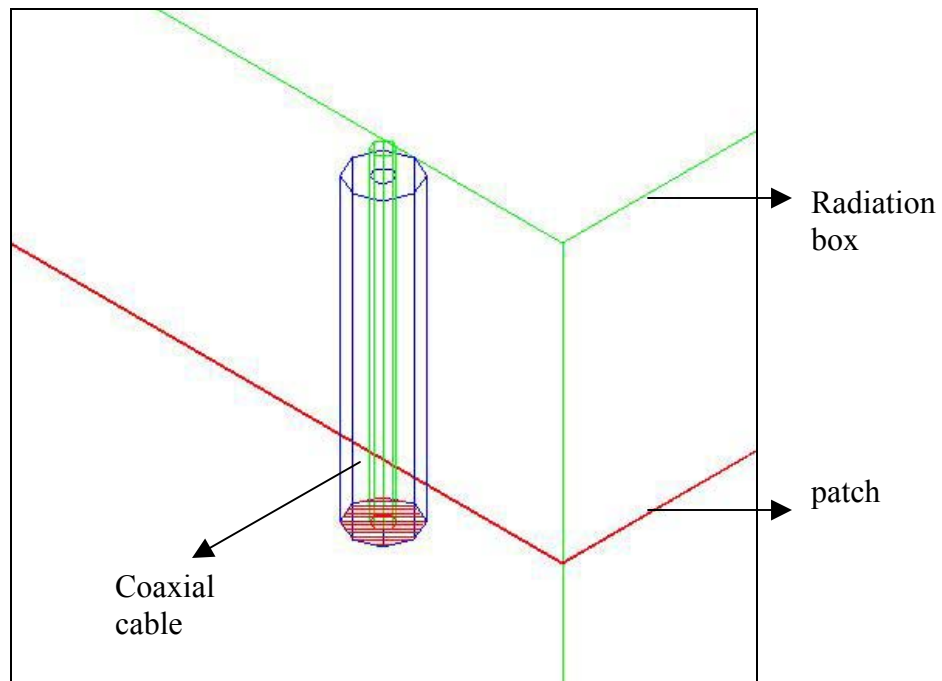
Fig 3.13:Location of the port

The location shown in the figure was applied by choosing 3 points, two of which, were on the bottom surface of the outer cylinder and one was on the bottom surface of the inner cylinder.

This ended the process of defining ports and boundary conditions. All successfully defined boundaries appear in the *Ports and Boundaries Currently Defined* list, which can be seen by selecting *Add/Modify* in the *Boundaries* menu. With this, the design of the simulation was completed. The next step was to set up the solution by specifying various parameters such as mesh refinement frequency and acceptable level of accuracy. The following section provides information about the parameters specified for one particular

solution. However, these parameters were changed to investigate their effect in HFSS, details of which are presented in the next chapter.

3.2.4 Setting up the Simulation

As discussed in section 2.5, there are two important tabs in the set-up window, namely, *Refinement* and *Frequencies*. Figure 3.14 shows a snapshot of the way the *Refinement* Tab was set up.
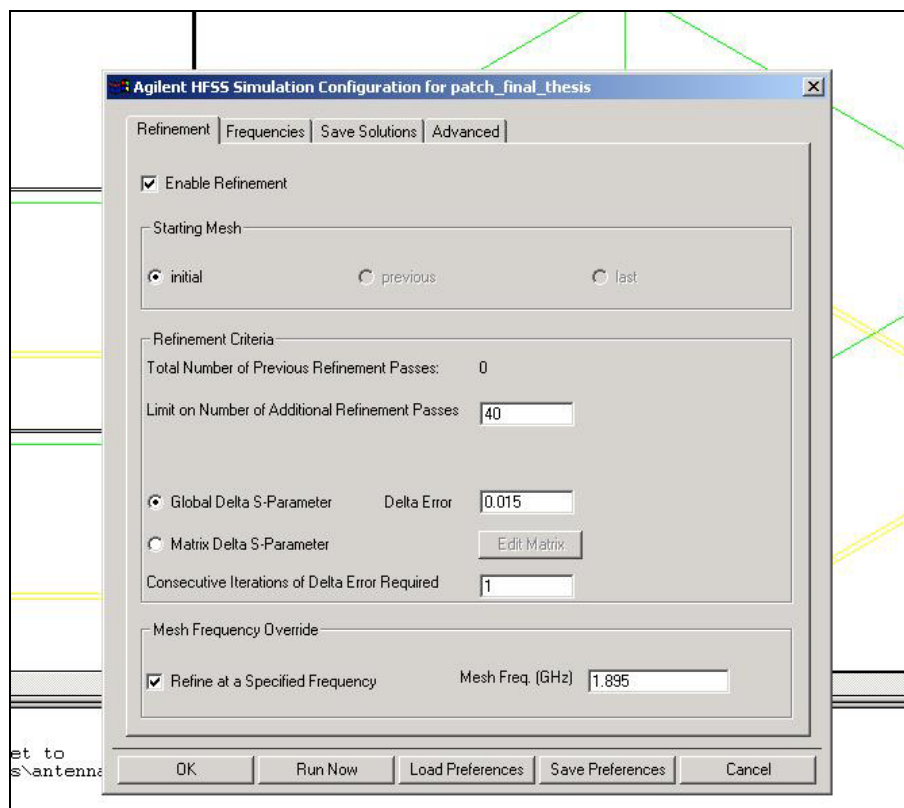
Fig 3.14: Snapshot of the *Refinement* Tab for the coax-fed patch antenna

As can be seen, mesh refinement was enabled and the initial mesh was specified as the starting mesh because the structure was being solved for the first time. The limit on number of iterations was specified to be 40, which is a large number and the simulation was expected to converge before that number could be reached.

S-parameters or scattering parameters describe the scattering and reflection of traveling waves when a network is inserted into a transmission line. They are measured as a function of frequency. For each port, the incident and the reflected wave are measured. With the presence of only one port in the patch antenna, s_{11}, which represented the input reflection, was the only S-parameter that could be calculated. Hence, it did not quite matter whether the error was specified in *Global Delta S-parameter* or *Matrix Delta S-parameter*. The actual value of error was specified to be 0.015. Also, the simulation would converge as soon as the *Delta Error* between two consecutive iterations became less than 0.015 for the first time. Finally, the mesh refinement frequency was specified as 1.895 GHz because the return loss at this frequency was expected to be 10 dB or better. The Frequencies tab is shown in Figure 3.15.
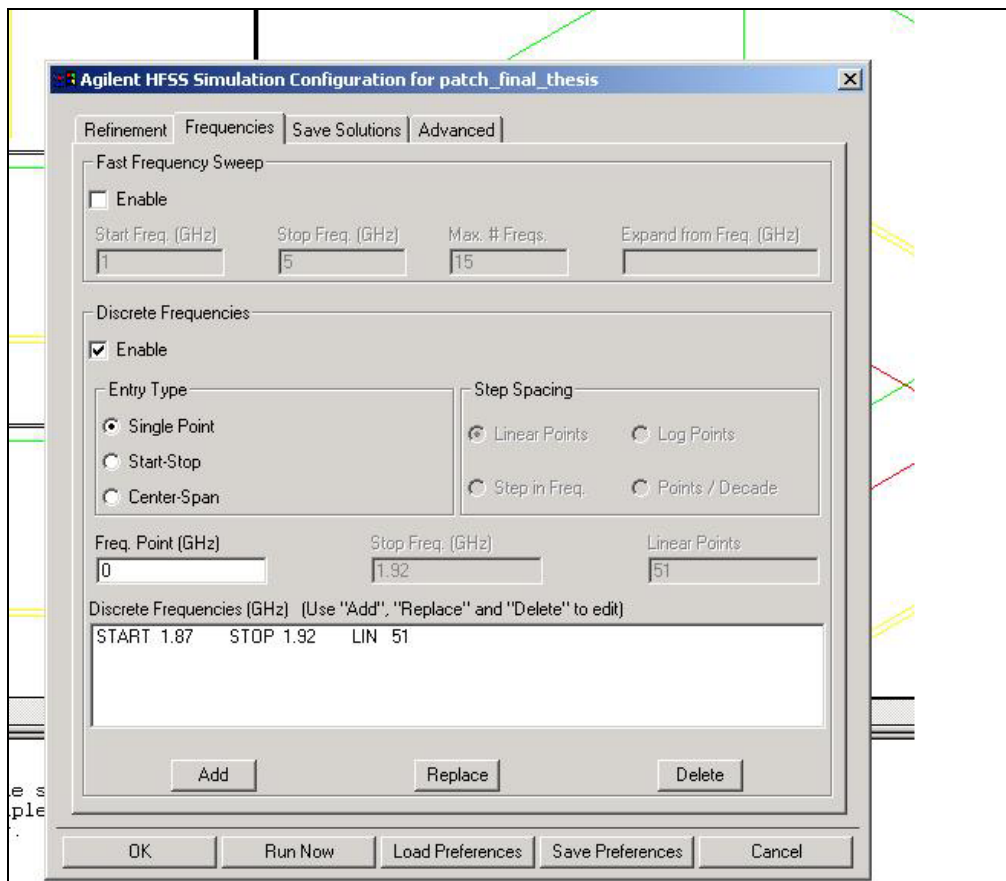
Fig 3.15:Snapshot of the *Frequencies* tab for the coax-fed patch antenna

As explained earlier, enabling the Discrete Frequencies solution gives the user control over the exact specific frequencies for which the structure will be solved. Hence, this option was selected and 51 frequencies were specified between 1.87 GHz and 1.92 GHz. This basically meant that a resolution of 1 MHz was used for solving the coax-fed patch antenna. This was believed to be the right choice considering the fact that a lower resolution would significantly increase the simulation duration without any added advantage in the S-parameter plot.

The mesh seeding parameters under the *Advanced* tab were set to defaults and aggressive mesh refinement was not used. Also, all the solutions were saved.

Finally, the simulation was started using Solve > Run. The process of mesh refinement continued for 18 iterations after which the change in magnitude of s_{11} was 0.014959. This caused the simulation to converge. At this point, the number of unknowns in the simulation was approximately 199,000. Figure 3.16 shows a snapshot of the convergence, which was created using the *Convergence > Log File* command from the post-processor.
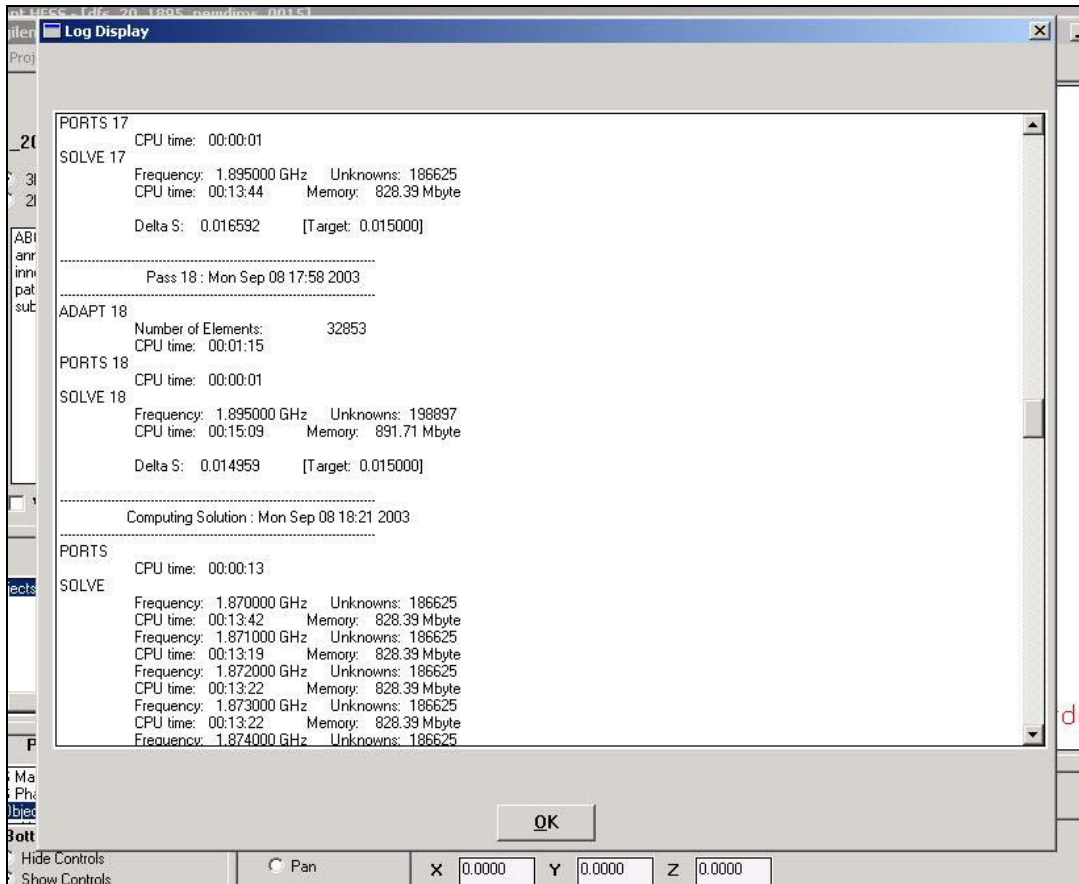
Fig 3.16:Snapshot of the simulation convergence

As shown in the figure, the solution of the structure for each of the 51 individual frequencies began after the mesh refinement was completed at 1.895 GHz. The total duration of the simulation, which was run on a PC with an Intel Pentium 1 GHz processor and 512MB RAM, was 21 hours, 21 minutes and 21 seconds.

3.3 Details of the 3-element array

Having modeled a single element antenna, the next step was to simulate the performance of a linear array of coax-fed patch antennas. As has been mentioned, the final aim of the RLA project was to use an array for the purpose of direction finding. As a result, a 3-element linear array was modeled. The layout of the array is shown in figure 3.17 where all dimensions are in mm.

47
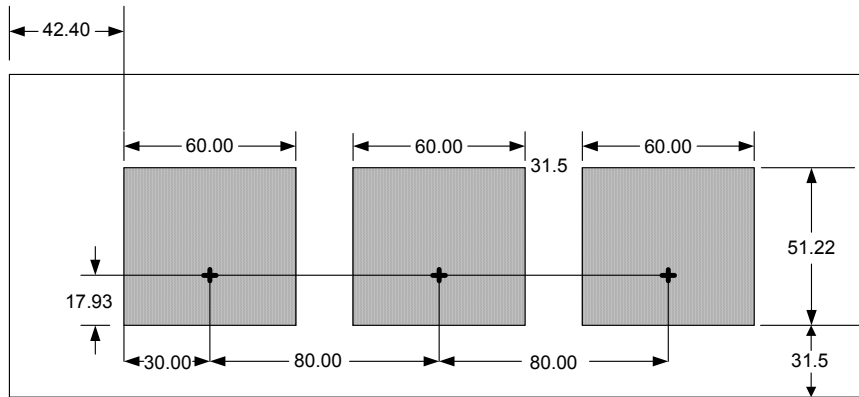
Fig 3.17: Layout of the 3-element array

It can be seen that the dimensions of the patch itself remain the same as they were for the single element case. The location of the feed point also remains at the same relative location with respect to the length and width of the patch. The length of the ground plane u remains same while its width v increases to 304.8 mm. The spacing between the patches is approximately $\lambda/2$ at a frequency of 1.875 GHz.

3.3.1 HFSS Simulation Description

Although a major part of the simulation design was the same as that for the single element, there were a few significant aspects that were different for the array.

Having changed the dimensions of the ground plane, the next step was to draw the remaining two elements. Two copies each of *patch*, *inner* and *annulus* were created using the *Edit > Copy* command. These were shifted from each other in accordance with the inter-element spacing. Creating a copy of *annulus*, which was originally formed by a subtraction operation, also generated copies of the parent objects, namely, *outer* and *inner*. It may be recalled that the object *sub2* in case of a single patch was formed when the inner conductor was subtracted from the substrate. Now, in case of the array, this process of subtraction had to be performed thrice. In other words, after creating *sub2*, the inner conductor of the second element was subtracted from it to created *sub3*. Subsequently, the inner conductor of the third and final element was subtracted from

sub3 to create *substrate_final*, that is, the substrate remaining after inner conductors of all elements had been removed. Fig 3.18 shows this hierarchical process of subtraction.

```
┌─────────────────────────────┐
│  Original object substrate  │
└─────────────────────────────┘
              │   Subtract inner conductor of
              │   patch 1
              ▼
        ┌────────────────┐
        │  Create sub2   │
        └────────────────┘
              │   Subtract inner conductor of
              │   patch 2
              ▼
        ┌────────────────┐
        │  Create sub3   │
        └────────────────┘
              │   Subtract inner conductor of
              │   patch 3
              ▼
     ┌──────────────────────────┐
     │ Create substrate_final   │
     └──────────────────────────┘
```
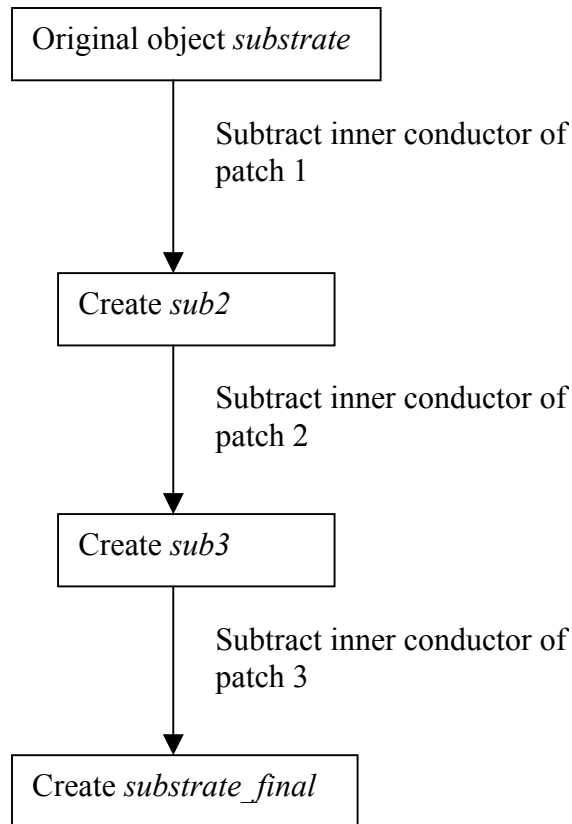
Fig 3.18: The subtraction process to create the final substrate

Finally, the width of the radiation box was also changed so that it would coincide with the width of the ground plane, as was the case earlier.

When objects are copied in HFSS, the new objects are assigned the same material as that of the original objects. Hence, all the new objects created for the array had the right material assigned to them automatically.

Similar to the objects, the corresponding boundaries also needed to be duplicated in order to accurately simulate the array. Three of the boundaries described in section 3.2.3 were specific to the element, namely, the *Perfect E* boundary applied to the *annulus*, the

Restore boundary applied to open up a hole in the ground plane and the *port* boundary. Thus, it was necessary to apply all these boundaries to the two newly created elements. Assigning the *Perfect E* and the *Restore* boundaries was straightforward and was done in a manner identical to that described in section 3.2.3. However, defining ports to all of them was slightly complicated. As shown in Figure 3.13, the bottom surface of the coaxial cable was used to set up the port. Due to this, if the lengths of the coaxial cable of all the 3 elements were equal, the bottom surface of each of them would lie on the same plane. Now, HFSS allows only a unique port in a single plane and hence it became impossible to define 3 separate ports for the 3 elements with equal cable lengths. In order to avoid this problem, the lengths of the cables had to be made unequal by altering them slightly. Hence, the cable length of the middle element was specified as 19.9 mm while that of the rightmost element was specified to be 20.1 mm. Locations of all the three ports are shown in Figure 3.19 in dark lines.
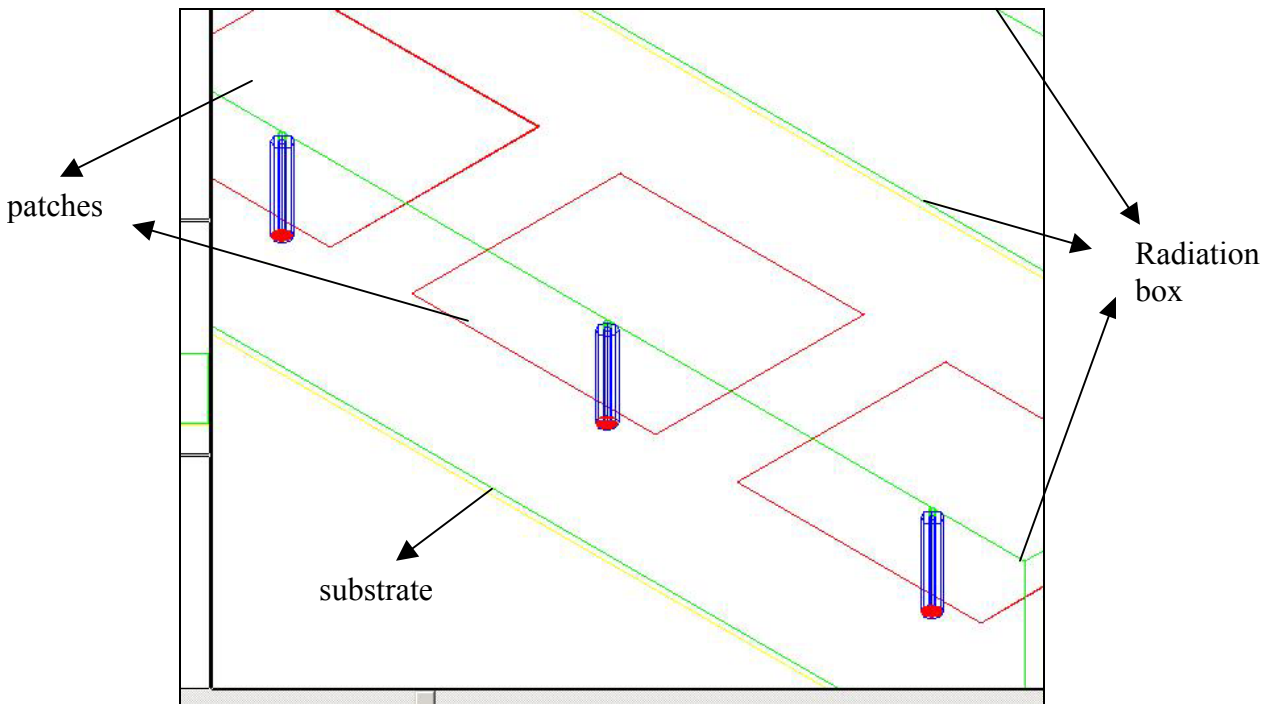
Fig 3.19:Location of the ports for the 3 elements in the array

The solution set-up process was more or less identical to that for the single element. It should be pointed out that the presence of 3 ports in the structure led to the calculation of

a 3 x 3 S-Matrix. A snapshot that was created using the *Plot > S Mag Plot* command is shown in Figure 3.20.
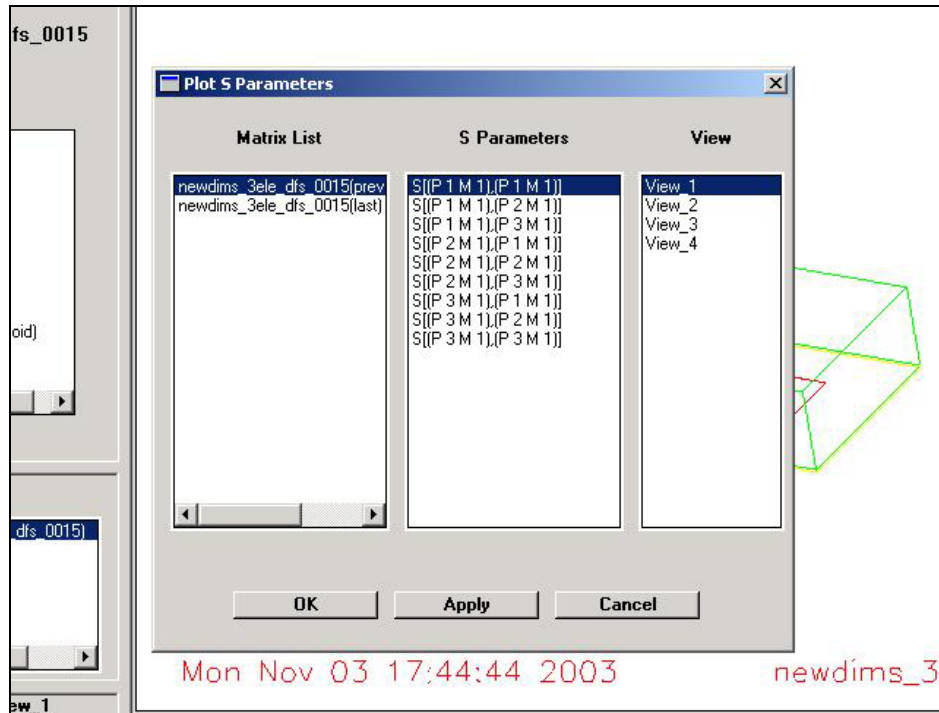
Fig 3.20: List of S-Parameters calculated after the array simulation

In the figure, 'P' stands for port and 'M' stands for mode. It can be clearly seen that each port has only one mode and any of the S-parameters can be selected and further processed.

In this way, sections 3.2 and 3.3 provided an elaborate explanation of the various issues involved in the process of simulating a coax-fed patch antenna in HFSS. As expressed in the beginning of the chapter, the S-parameter plots generated by HFSS were then compared with those generated at GWU by WIPL-D and measurements. The next section gives a brief introduction to the simulation software WIPL-D and to the experimental set-up.

3.4 Introduction to WIPL-D and Experimental Setup

WIPL-D is an efficient program that enables the user to model metallic and/or dielectric/magnetic structures such as antennas, scatterers, passive microwave circuits, etc [6]. The code uses Method of Moments (MoM) as its basis to solve 3D electromagnetic structures and performs calculations in the frequency domain. The geometry of any structure can be defined in an interactive way through a combination of wires, plates and material objects. This is also the reason for the abbreviation WIPL-D where WI stands for wires, PL stands for plates and D stands for dielectrics [1].

A program based on MoM could have either surface currents or volume fields as its unknown quantities. WIPL-D uses surface currents as unknown quantities and hence it is a MoM/SIE code where SIE abbreviates Surface Integral Equation. This is in stark contrast to programs like HFSS, which are based on the Finite Element Method (FEM), and in which volume fields and currents are the unknown quantities. Because of this, the number of unknowns in WIPL-D is usually much lower as compared to those in HFSS resulting in shorter analysis durations in WIPL-D. As is the case with HFSS, there is no error function in WIPL-D. On the other hand, the accuracy levels for current expansion and integral accuracy can be specified based on the number of grid points and parameters [7]. With WIPL-D and HFSS using two completely different approaches, it was interesting to compare their results. Simulations of structures identical to those described in the previous sections were also carried out in WIPL-D at GWU and a comparison is presented in the next section.

In order to evaluate the performance of both WIPL-D and HFSS, it was important to compare their results with actual measurements. Various patch antennas with dimensions within 10 microns of the values used in the computations were manufactured. Each panel consisted of copper on both sides with thickness of 35 microns. A substrate made up of RT/5880 duroid and having a thickness of 62 mils was interspersed between the two layers of copper. The upper side of the panel was etched so as to produce patch radiators. On the bottom side, SMA flanged connectors with 50 Ω impedances were affixed. S-Parameter measurements were then performed on them at GWU using an accurately

calibrated HP 8722D network analyzer. Two identical sets of panels were fabricated and measured both for the single element and for the array. The resulting measurement data was exactly same for both the cases. In this way, it was made sure that the measurements were repeatable and also that there were no significant dimensional inaccuracies in the panels.

Fig 3.21:Top and bottom views of fabricated patches

Fig 3.22:Array connected to the network analyzer

Figure 3.21 shows pictures of top and bottom views of the single element patch and the 3-element array. Fig 3.22 shows how the array was connected to the network analyzer while performing measurements. These pictures were taken at GWU.

The next section of this chapter presents results of the simulation described. These are then compared with WIPL-D and experimental measurements, both for the single element and for the array.

3.5 HFSS Results and Comparison

3.5.1 Single element

Figure 3.23 shows the plot of the S-parameter magnitude in dB versus frequency in GHz for the single patch. All plots in this section have been generated by exporting data from the respective simulation program into MATLAB.
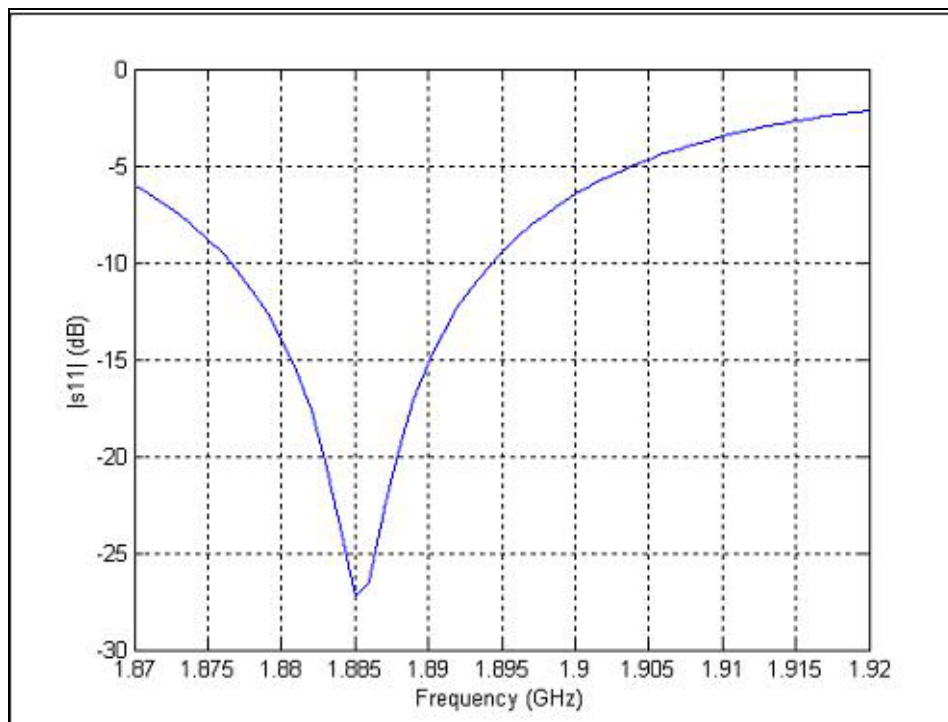

Fig 3.23: Return loss for a single element patch in HFSS

As can be seen, the resonant frequency of the patch is 1.885 GHz. Also, the bandwidth of the patch at −10 dB points is approximately 20 MHz. Figure 3.24 shows the comparison of the return loss values as predicted by HFSS and WIPL-D with those that were measured. Firstly, there is significant difference in the results of HFSS and WIPL-D. Also, none of them match with the experimental values. In comparison to the resonant frequency of 1.885 GHz obtained through HFSS, WIPL-D estimates a resonant frequency of 1.904 GHz. However, both these values differ from the measured resonant frequency of 1.91 GHz. It may be recalled that the patch was designed to resonate at 1.904 GHz in WIPL-D. The −10 dB bandwidth as calculated by WIPL-D is 20 MHz, which is similar to that in HFSS but different from the measured bandwidth which is 30 MHz. It can thus be seen that, while both HFSS and WIPL-D deviate significantly from experiment, WIPL-D is marginally closer to measurements than HFSS.
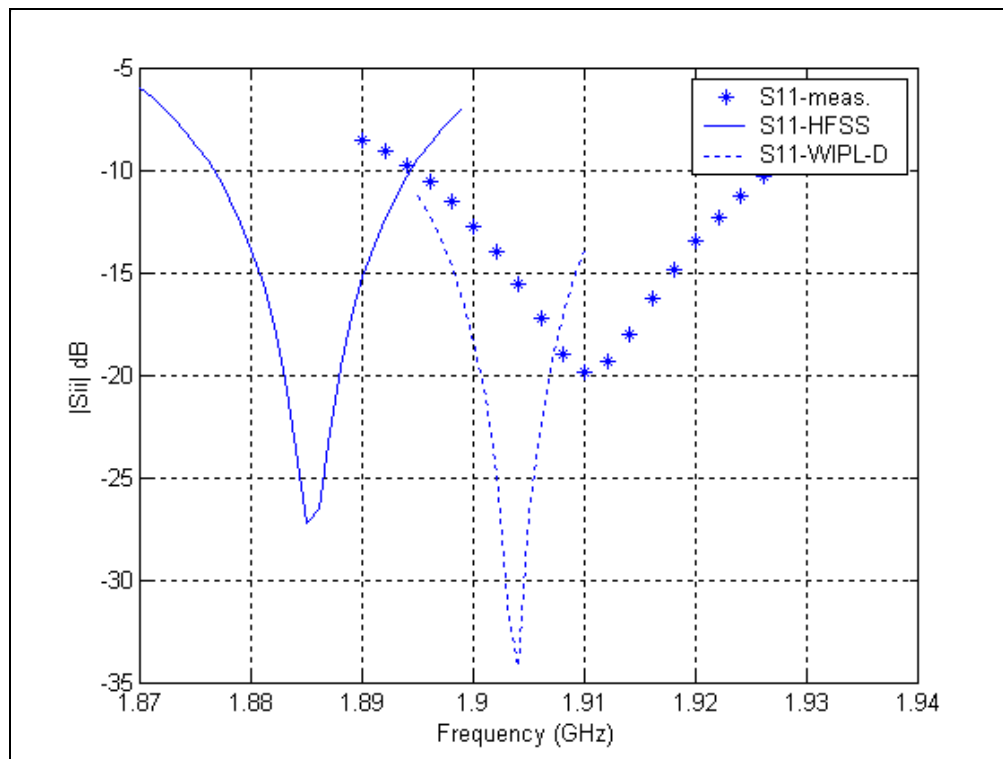

Fig 3.24:Comparison of S-parameter plots between HFSS, WIPL-D and measurements

3.5.2 3-Element Array

As shown in Figure 3.20, a 3 x 3 S-Matrix was calculated and measured for the 3-element array. Figure 3.25 shows HFSS plots of the diagonal members of the S-matrix. As expected, the middle element resonates at a slightly higher frequency as compared to the edge elements. It can be observed that the edge elements resonate at 1.873 GHz and the middle element resonates at 1.877 GHz. Figure 3.26 shows HFSS plots of the distinct off-diagonal members of the S-matrix, namely, s_{12}, s_{13}, and s_{23}. Although not shown here, symmetry in the S-matrix was verified. In other words, the plots of s_{12} and s_{21} were exactly identical to each other. This held true for s_{13} and s_{31} and also for s_{23} and s_{32}. It should be noted from the two figures that the slight differences in the lengths of the coaxial cables of the 3 elements did not have any effect on the final results.
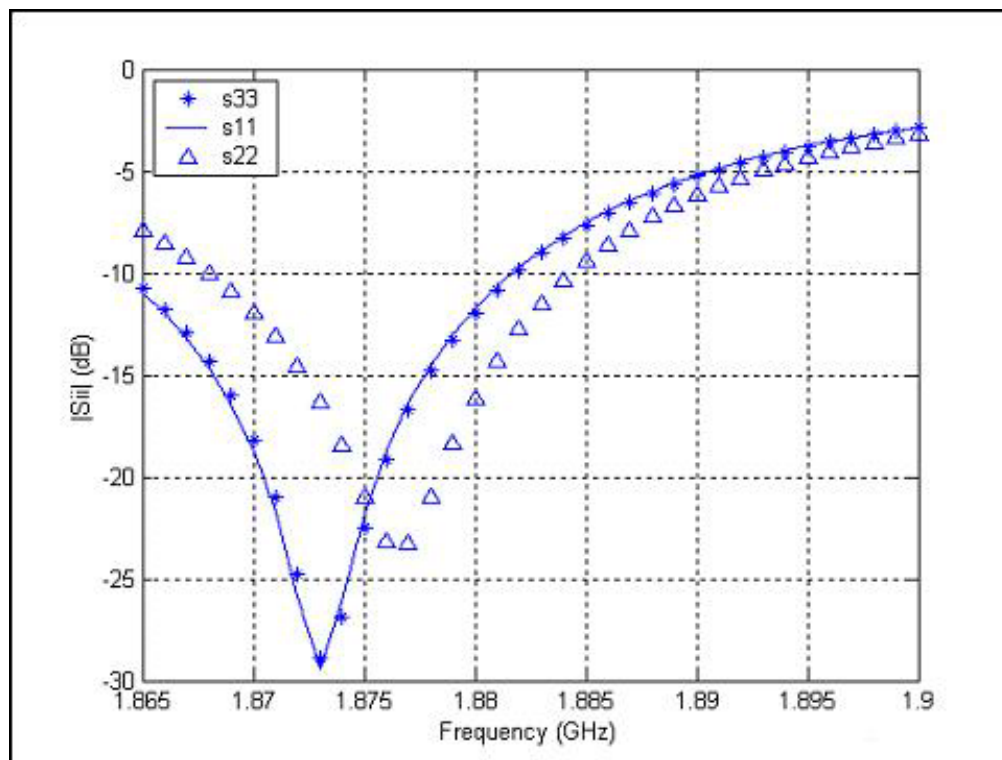
Fig 3.25:Plots of s_{11}, s_{22} and s_{33} in HFSS

56
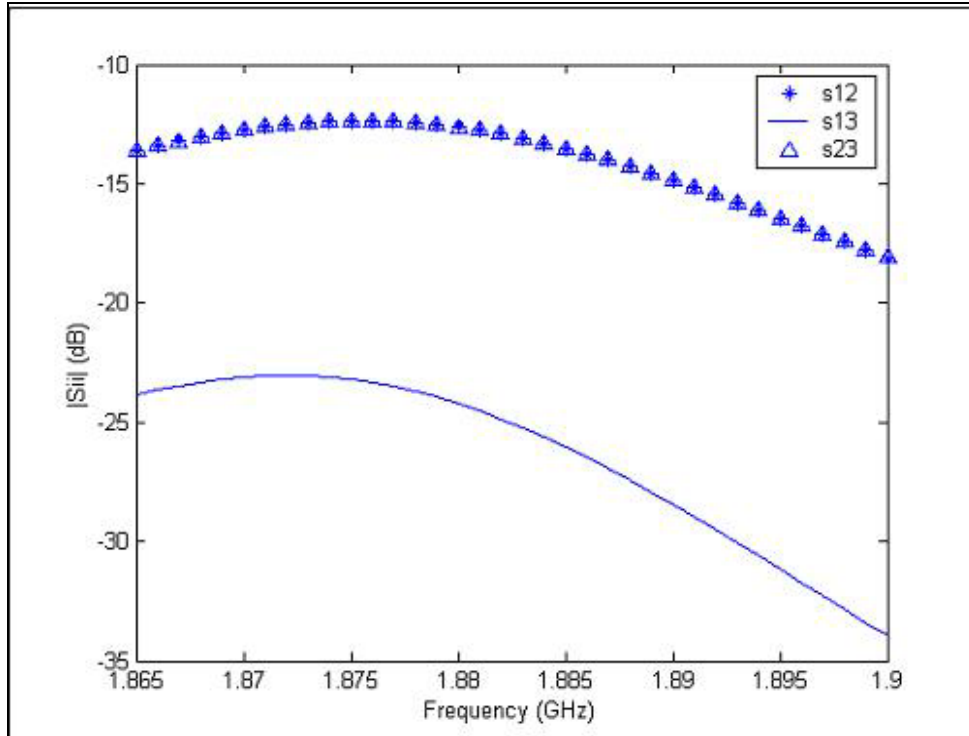
Fig 3.26:Plots of s_{12}, s_{13} and s_{23} in HFSS

Finally, the results shown in Figure 3.25 are compared with WIPL-D and measurements in Figure 3.27. As was the case with HFSS, the measured resonant frequency of the middle element also shows an upward shift. However, such a trend is not seen in the WIPL-D calculations. The resonant frequency of all the elements as calculated by WIPL-D is seen to be the same as that of the single element (1.904 GHz). This is also true in case of experimental data in which the edge elements resonate at 1.91 GHz as earlier. Thus, it can be concluded that although both programs get close to successfully modeling the array, each of them has substantial differences with the actual measured data.
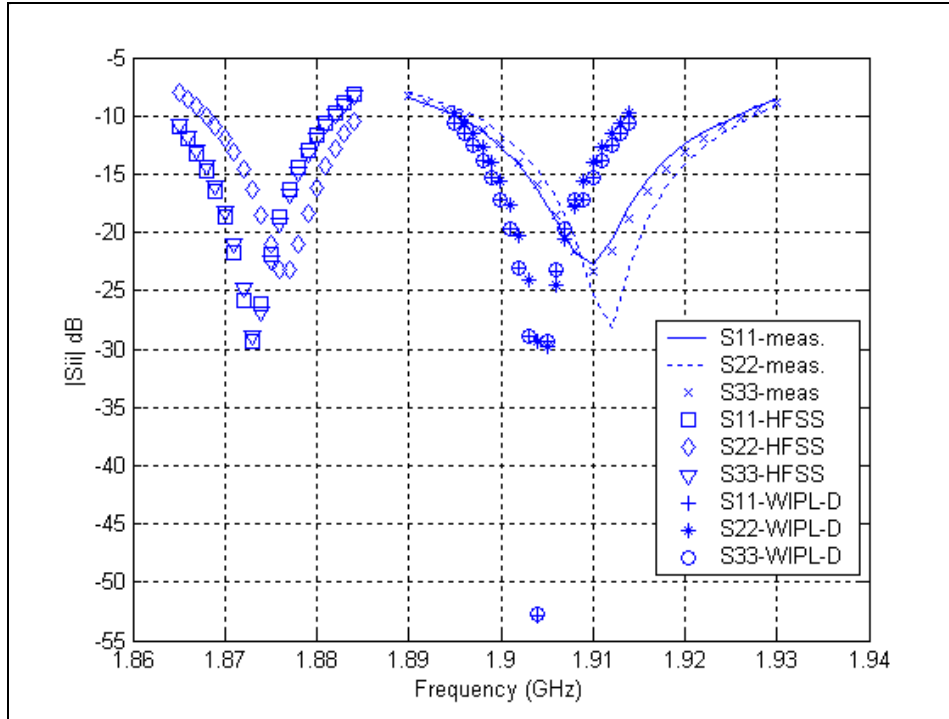
Fig 3.27: Comparison of S-parameter plots between HFSS, WIPL-D and measurements for the array

3.6 Conclusion

Having presented a comprehensive description of HFSS in the previous chapter, the main aim of this chapter was to throw light on the coax-fed patch antenna structures that were used in the RLA project. With the help of snapshots, an effort was made to provide a meticulous description of the simulation of those structures in HFSS. On comparing the results of HFSS and WIPL-D with measurements, noticeable differences were found which led to the conclusion that neither of the programs could be successfully used as completely reliable tools for coax-fed patch antenna design [5].

Chapter 4

Effects of Various Parameters in HFSS

Having simulated the patch antenna that was built for the RLA project, effects of varying some major parameters of the simulation were investigated. From the point of view of the geometry of the structure, ground plane size and patch thickness were the parameters studied. Results of changing two important solution set-up parameters, namely, mesh refinement frequency and the allowable error in S-parameter magnitude, are also presented in this chapter. Merits and demerits of using a fast frequency sweep as opposed to discrete frequencies in the simulation are shown. The chapter illustrates the method of using a voltage source in order to excite the structure and shows a comparison of results with those generated using a port. Finally, effects of varying the size of the radiation box are also presented.

4.1 Effect of Ground Plane Size

As mentioned in the previous chapter, the length u and the width v of the ground plane of the patch antenna used in the RLA project are $L + 40h$ and $W + 40h$ respectively, where L and W are the length and width of the patch respectively and h is the height of the substrate. To understand the effect of ground plane size, a patch antenna with a smaller ground plane was simulated in HFSS. The length and width of the ground plane were reduced to $L + 20h$ and $W + 20h$ respectively. All other parameters such as the patch length and width, substrate height, patch thickness and the relative feed point location were same as earlier. Figure 4.1 shows a comparison of the return loss plots versus frequency for the two structures with different ground plane sizes. In both the cases, the solution was set up for a mesh refinement frequency of 1.895 GHz and an error in S-parameter magnitude of 0.015. As can be seen, there is a significant difference in the S-parameters plots. While the structure with the smaller ground plane has a higher resonant frequency, the return loss in dB at the resonant frequency is higher for the structure with

the larger ground plane. The difference in resonant frequencies is about 5 MHz and the difference in return loss values at the resonant frequencies is also approximately 5 dB.
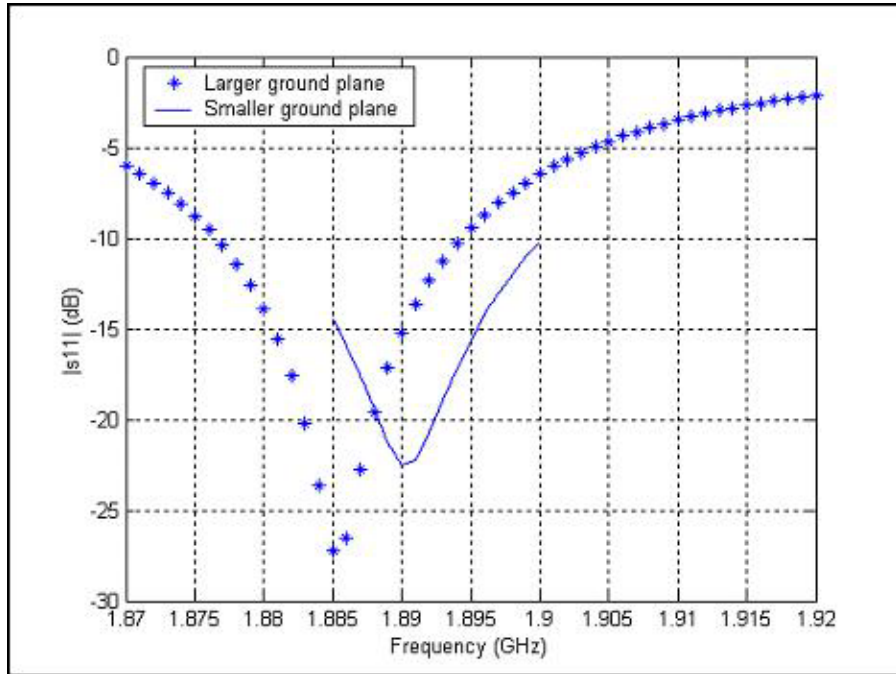
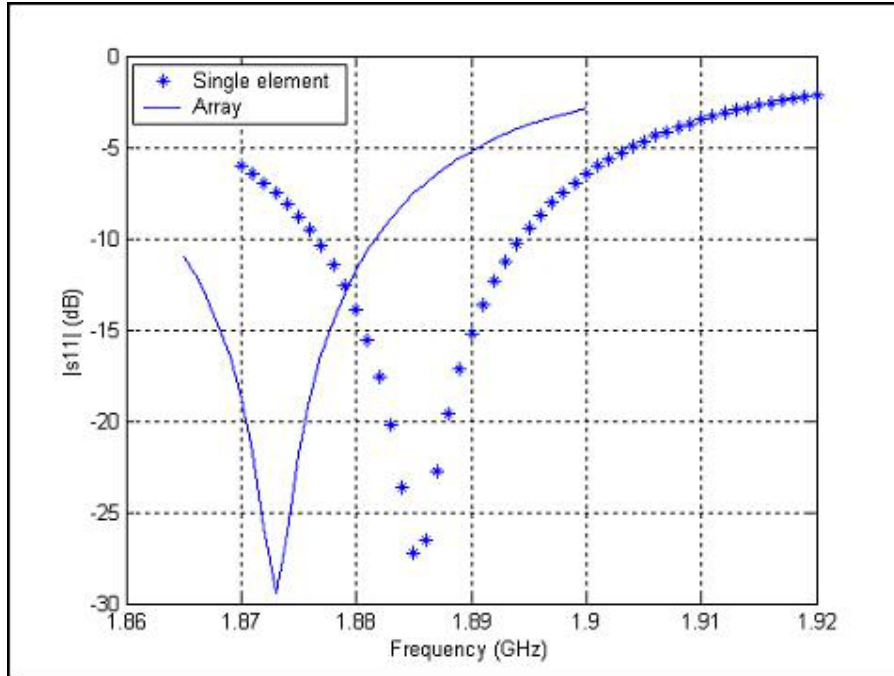Fig 4.1:Comparison of S-parameter plots for different ground plane sizes

Fig 4.2: Comparison of s_{11} plots for the single element and array

A comparison of s_{11} plots in the case of the single element and the array, which were built for the RLA project, is depicted in figure 4.2. The ground plane for the array is naturally larger than that for the single element. This figure further confirms the trend seen earlier, that is, the s_{11} plot for the array has a lower resonant frequency and a higher return loss at the resonant frequency. It should be noted that this is in stark contrast to WIPL-D, which estimates identical resonant frequencies for the array and for the single element. This can be confirmed by comparing the resonant frequencies of WIPL-D calculations from figures 3.24 and 3.27.

It should be pointed out that the results presented in the following sections were all based on computations of the patch antenna with the smaller ground plane. This is so, because the larger the structure, the more complicated is the finite element mesh created by HFSS resulting in more stringent memory requirements. For example, the simulation of the patch antenna with a larger ground plane did not converge when the error was specified to be 0.01 instead of 0.015 because it ran out of memory. On the other hand, the smaller ground plane case converged successfully for an identical error. Issues regarding memory

and accuracy shall be further addressed in Section 4.3, where it is shown that higher accuracy requires more memory for the simulation to converge.

4.2 Effect of Patch thickness

The next parameter examined was patch thickness. In order to accomplish this, patch antennas with 3 different values of thickness were simulated. Each of them was set to be refined at a frequency of 1.895 GHz. Also, the *Delta Error* was specified to be 0.015 in each case. Apart from the standard thickness of 0.035 mm, which was used in the RLA project, an antenna with a thickness of 0.05 mm was simulated. Also, a structure having zero patch thickness was also created. This was done by modeling the object *patch* as a 2D object and by drawing it using the *Rectangle* command in the *2D Objects* menu. Figure 4.3 shows the S-parameter plots for all the 3 cases. It can be seen that there is no drastic change in the return loss calculations by varying the patch thickness. However, there is an increase of about 1 MHz in the resonant frequency when the thickness is increased from 0 to 0.035 mm and similarly, from 0.035 mm to 0.05 mm. The return loss values at the resonant frequency are almost identical for the 3 cases considered.
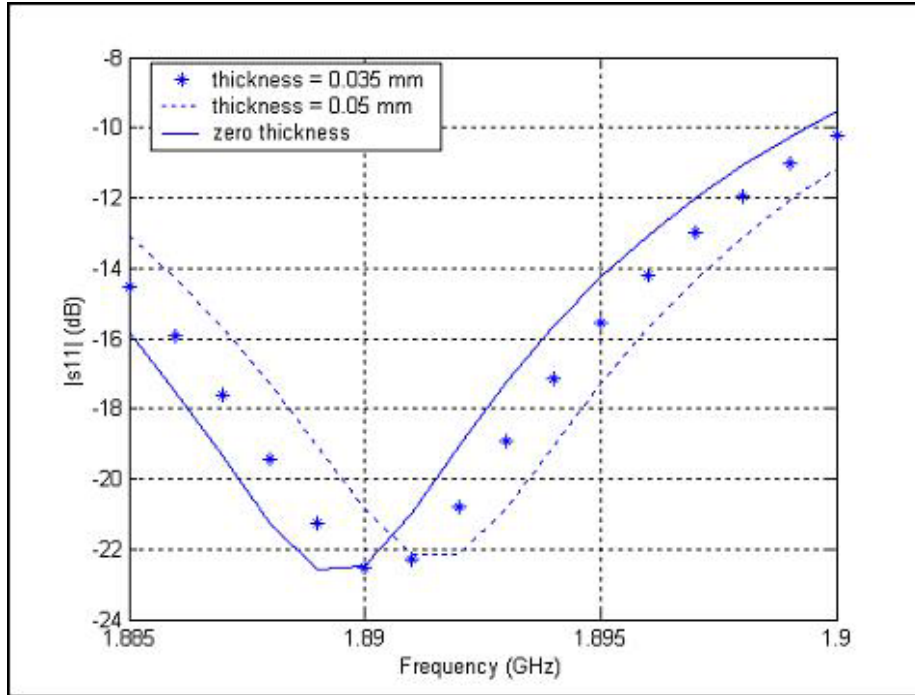
Fig 4.3:Effect of patch thickness

4.3 Effect of Accuracy

Mesh refinement frequency and the acceptable change in S-parameter magnitude are the two most important parameters that need to be specified while setting up the solution of any structure in HFSS. As was discussed in chapter 2, after performing each refinement, HFSS checks if the change in magnitude of S-parameters between the last two refinements is less than the specified value. If that is not the case, it performs another refinement by creating a mesh with a larger number of points thereby requiring more memory to solve it. Thus, it is imperative to understand the effect of this parameter.

Three different cases were simulated with specified *Delta S* values of 0.01, 0.015 and 0.03. They converged after 21, 16 and 13 iterations respectively. Clearly, the lesser is the specified error, the higher is the number of iterations required to reach it. Each of them had a mesh refinement frequency of 1.895 GHz. A comparison of the results is shown in Figure 4.4.
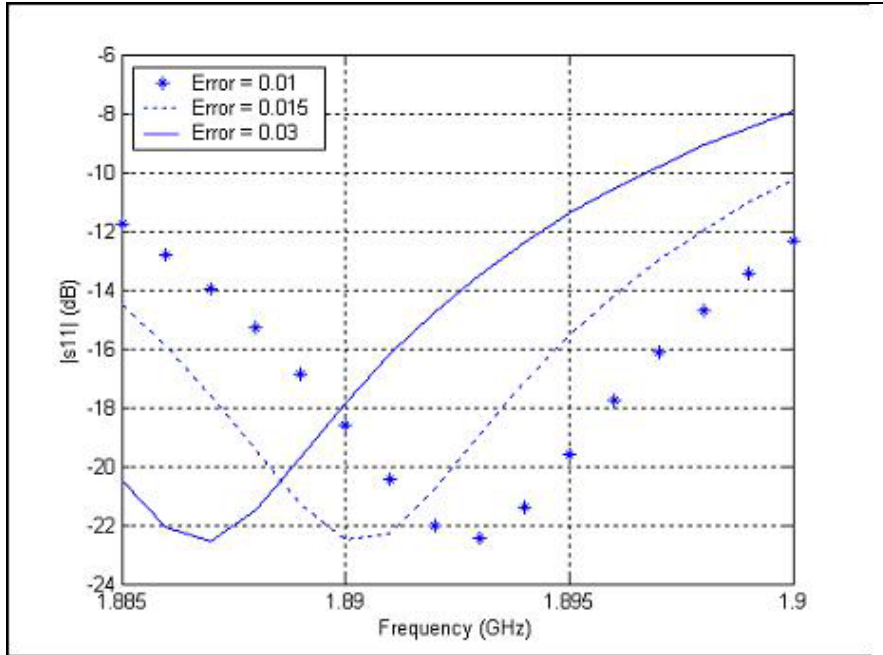
Fig 4.4: Comparison of S-parameter plots for different accuracy levels

As can be seen from the figure, the results of the 3 cases are significantly different from each other. The entire plot of the case with a *Delta S* of 0.03 is shifted by approximately 6 MHz from that generated when the error was specified to be 0.01. For narrowband patch design that usually has a 10 dB bandwidth of about 20 MHz, this shift can be considered to be quite large. It can thus be concluded that the values of *Delta S* should be specified as small as possible in order to obtain the most accurate results. However, in doing so, the user would force HFSS to build extremely fine meshes, which may not be solved using the available memory causing the simulation to end without converging. It would be best to use the lowest possible error level such that the simulation still converges. A *Delta S* value of 0.01 is usually considered as a standard value. However, as mentioned earlier, the simulation of the patch antenna with the larger ground plane failed to converge at this value and that is why all the results presented in section 3.5 were generated at a *Delta S* value of 0.015.

4.4 Effect of mesh refinement frequency

In order to obtain the most accurate solution, the use of mesh refinement is indispensable. Consequentially, the frequency at which this refinement is carried out is another crucial parameter that needs to be specified during the solution set-up phase. As was shown in the previous section, the refinement criterion is the change in magnitude of S-parameters. It is therefore important to refine at a frequency for which the return loss is high and hence the mesh refinement can produce accurate results. Various cases were simulated in order to investigate how this parameter affects the results in HFSS. The value of *Delta S*, which was common to all of them, was specified as 0.01. Before presenting the comparison, the result in which is the mesh refinement frequency is closest to the resonant frequency is re-plotted in Figure 4.5. This was obtained at a mesh refinement frequency of 1.895 GHz.
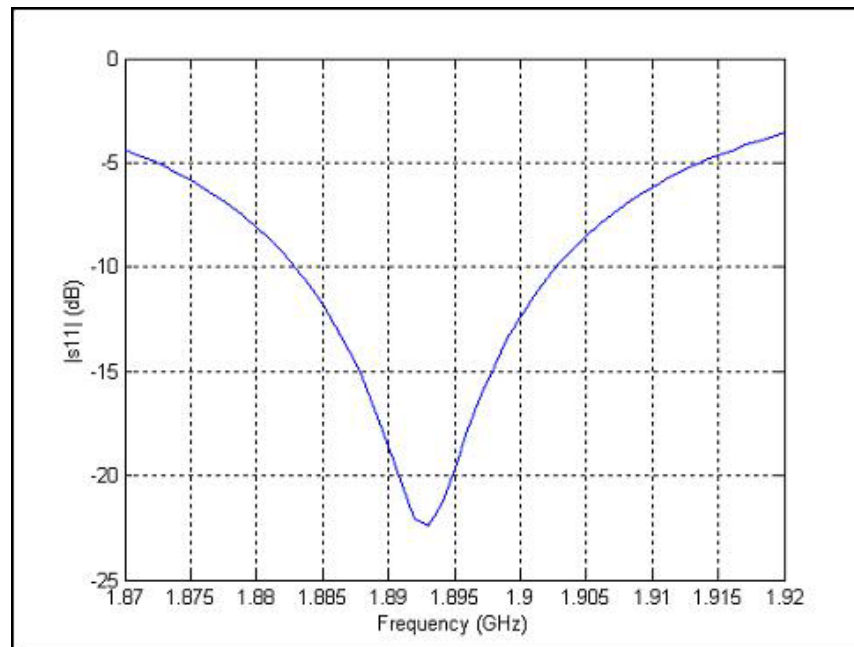

Fig 4.5: S-parameter plot for a mesh refinement frequency of 1.895 GHz

It is clearly seen from the figure that the return loss at the 1.895 GHz is better than 20 dB. The different frequencies for comparison were chosen based on the corresponding values

of return loss as evident from the above figure. One of the frequencies chosen was 1.8 GHz, which was completely out of band for the patch antenna in question. The second frequency picked was 1.85 GHz at which, although not seen in the figure, the return loss would be close to 0 dB. The remaining two frequencies used for refinement were 1.87 GHz and 1.902 GHz, at which the values of return loss were approximately 5 dB and 10 dB respectively. Figure 4.6 shows a comparison of the plots generated in each of the 5 cases mentioned above for a range of frequencies between 1.885 and 1.9 GHz.
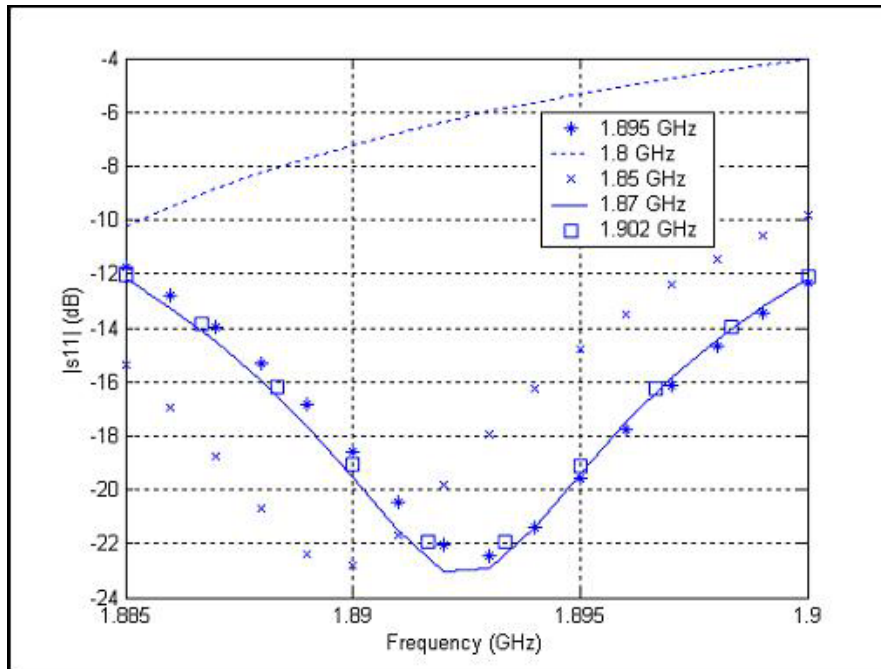
Fig 4.6: Comparison of S-parameter plots for different mesh refinement frequencies

A few inferences can be made from the above figure. Firstly, a mesh refinement frequency of 1.8 GHz produces completely inaccurate results. Secondly, a frequency like 1.85 GHz at which the return loss is close to 0 dB generates a plot that is shifted from the most accurate one by about 3 MHz. In the context of the patch antenna, such a shift is not acceptable. Finally, plots obtained at frequencies of 1.87, 1.902 and 1.895 GHz are almost on top of each other. Thus, it can be concluded that, while it is not compulsory for the mesh refinement frequency to be exactly equal to the resonant frequency, utmost care should be taken to ensure that it does not significantly differ from the resonant frequency.

Although Figure 4.6 shows a 5 dB return loss frequency as acceptable, a good rule of thumb would be to choose a frequency at which the return loss is expected to be 10 dB or better.

4.5 Discrete Frequencies and Fast Frequency Sweep (FFS)

All the S-parameter plots presented until this point were produced by solving the structure in question by using a discrete frequencies simulation (DFS). As a result, the number of frequencies and the specific frequencies at which the structure would be solved was always known before hand. As discussed in section 2.5, another way of solving for a number of frequencies is by setting up a fast frequency sweep (FFS). It would thus be interesting to compare the results generated by these two different methods. Two different cases, each having a *Delta S* of 0.015 and a mesh refinement frequency of 1.895 GHz, were simulated. One of them was solved using 15 discrete frequencies between 1.885 and 1.9 GHz and the other was solved using a fast frequency sweep in the same range. The maximum number of frequencies to be solved for was limited to 15 and no frequency was specified in the *Expand from Freq. (GHz.)* field. Figure 4.7 shows the comparison.
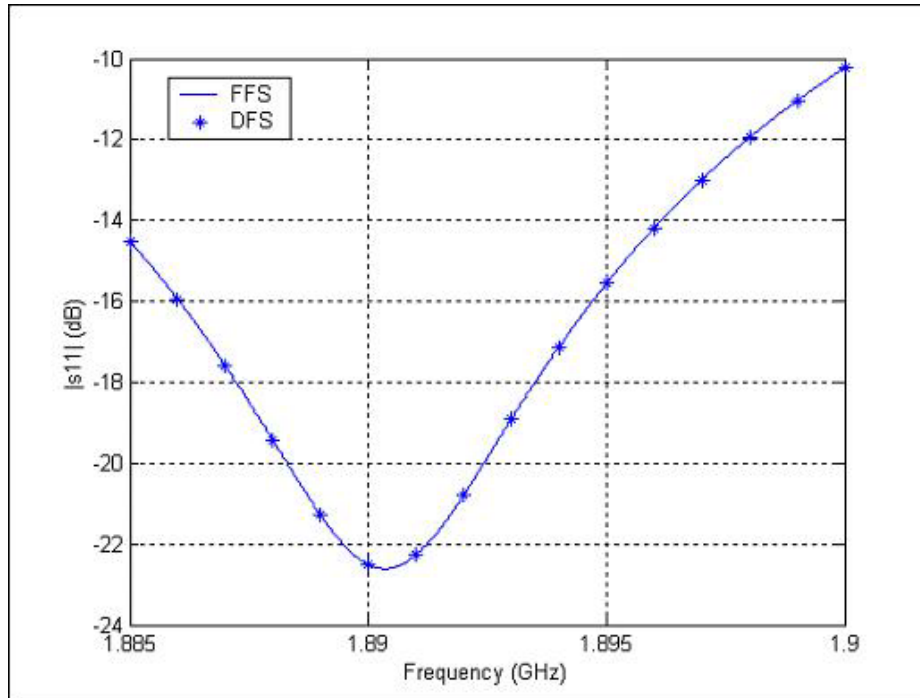
Fig 4.7: Comparison of fast frequency sweep with discrete frequencies

It is obvious that the two plots are exactly identical to each other. Also, the FFS simulation took approximately 1 hour and 45 minutes as compared to about 4 hours and 25 minutes for the discrete frequencies simulation. As expected, the FFS simulation was significantly faster because it does not solve the entire structure for each frequency as explained in section 2.5. Based on Figure 4.7, one might be tempted to conclude that FFS should always be used rather than using discrete frequencies. However, there are two important aspects of FFS that need to be kept in mind. Firstly, as mentioned earlier, the user would have no control over the frequencies at which the structure would be solved in an FFS simulation. Also, FFS requires more memory to perform the interpolation. In solving large structures, there might not be enough memory available to carry out a fast frequency sweep. An example of this was the case of the patch antenna with the larger ground plane. It can thus be concluded that FFS should be used if there is enough memory available to perform the sweep and if the solution is not desired at some discrete and specific frequencies.

4.6 Voltage Source Simulation

Ports have been used so far in all the simulations to infuse energy into the structures. However, as mentioned in section 2.4, voltage sources can also be as means of excitation. With respect to the patch antenna, a voltage source can be used to simulate an object at the base of the coaxial cable, which would energize the antenna. It was also pointed out in section 2.4 that a voltage source can only be defined between two 2D areas that are in different planes. With respect to the geometry of the coaxial cable as described in section 3.2.1, it would not be possible to set up a voltage source at the base of the coaxial cable because both the inner and outer conductors coincide and their bottom surfaces lie on the same plane. Hence, a slight modification needed to be done in the geometry. The length of the outer conductor was made 2 mm more than that of the inner conductor so that its bottom surface would be slightly below that of the inner conductor. In other words, the outer conductor now extended from $z = -h$ to $-(T + h + 2)$ instead of $-(T + h)$. The inner conductor extended from $z = 0$ to $z = -(T + h)$ as earlier. In this way, it became possible to define a voltage source between the bottom surfaces of the two conductors, which were now separated by 2 mm. It may be recalled from section 3.1 that the length of the coaxial cable T is 20 mm and the height of the substrate h is 1.5748 mm.

An attempt to show the voltage source definition is made in Figure 4.8. There exists a line between the two small circles located at the centers of the bottom surfaces of the two conductors. Although difficult to visualize in the figure because it coincides with the surfaces of the cylinder, it is this line, which signifies the current flow between the negative base and the positive tip.

69
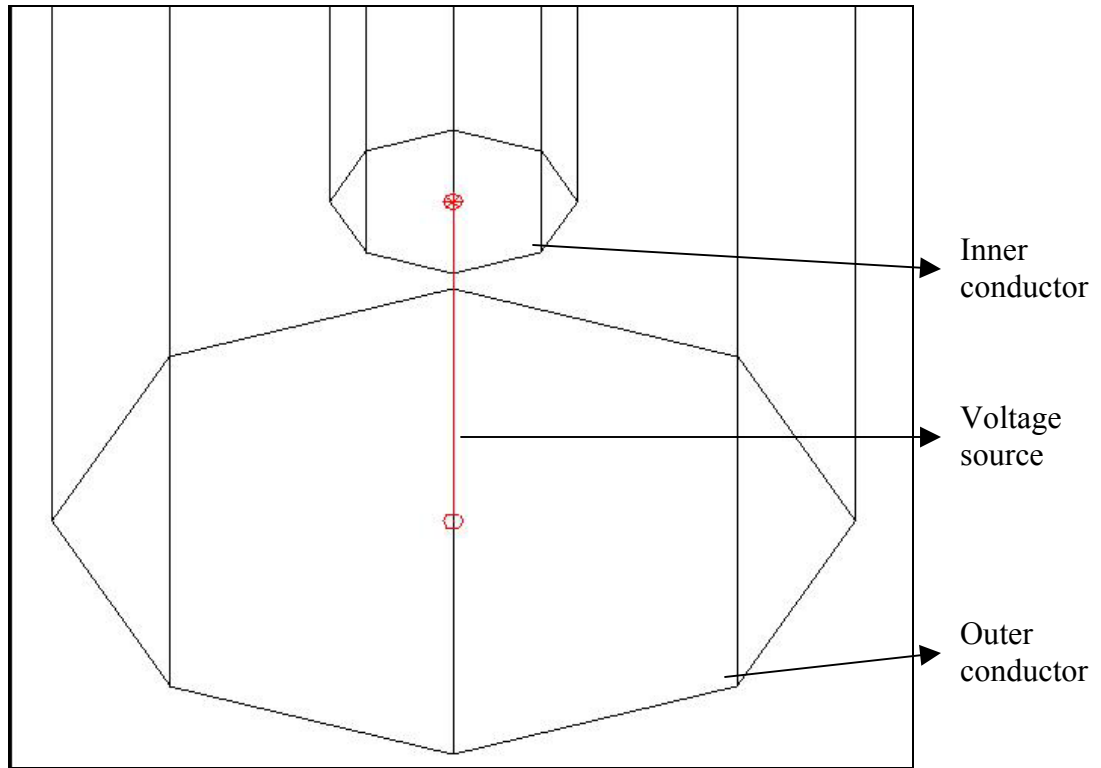
Fig 4.8:Voltage source definition between the outer and inner conductor

The actual definition of the voltage source was accomplished by using the Voltage Sources command in the Boundaries menu. The exact location was specified by entering the co-ordinates of the centers of the two bottom surfaces. Figure 4.9 shows a snapshot of the voltage source definition template. As can be seen in the figure, there is an asterisk sign next to the listing of the voltage source. This indicates that the corresponding voltage source has been defined successfully.

Fig 4.9: Snapshot showing the voltage source definition template
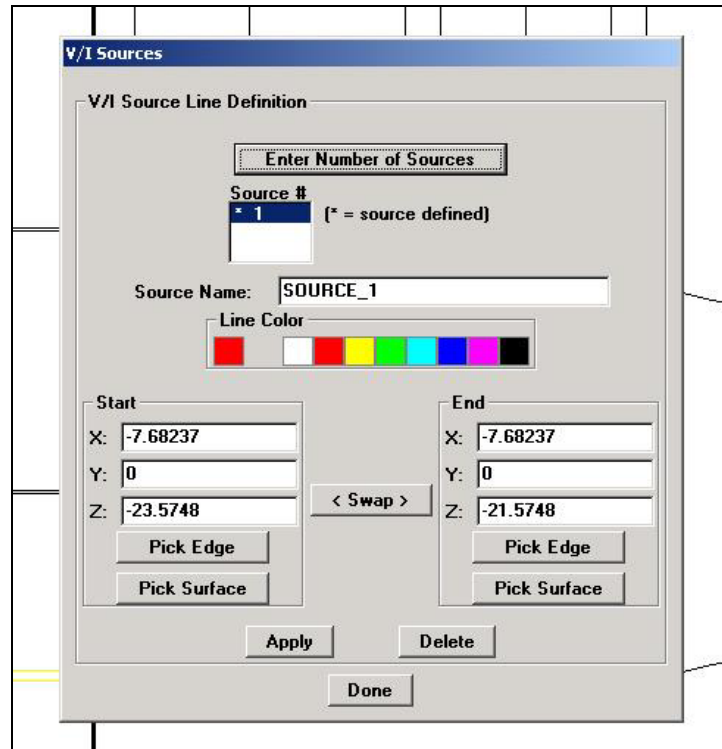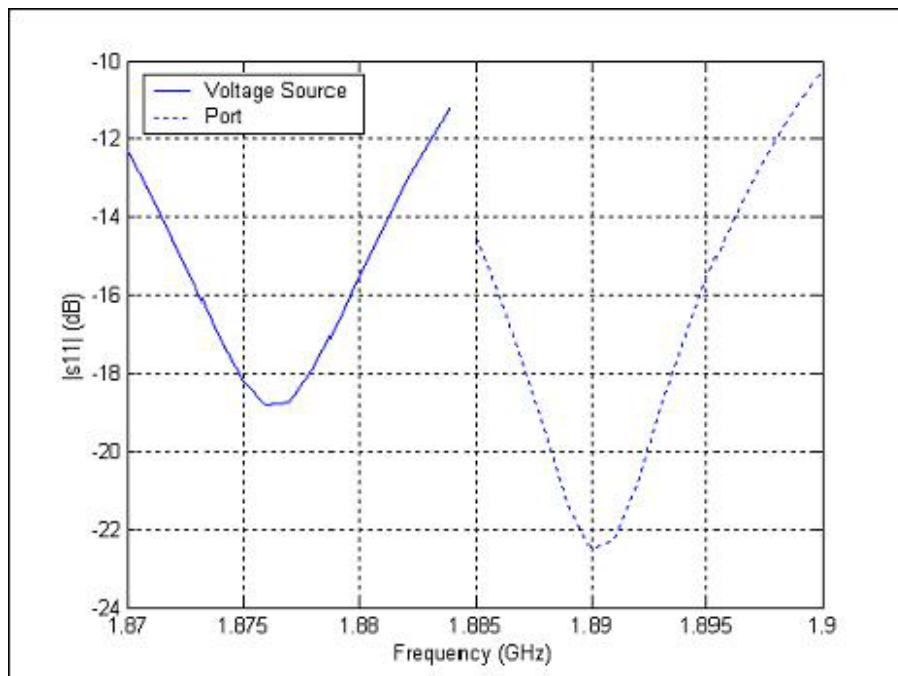

Fig 4.10: Comparison of results for port and voltage source excitations

For comparison of the voltage source simulation with the port simulation, two cases were simulated. Each of them had a mesh refinement frequency of 1.895 GHz and a *Delta S* value of 0.015. Discrete frequencies were used in both cases. Figure 4.10 shows the comparison of the S-parameter plots generated for the above two cases. As compared to the resonant frequency of about 1.89 GHz for the port case, the voltage source simulation produces a resonant frequency of approximately 1.877 GHz. In other words, the two plots are shifted by 13 MHz. Also, the return loss at the resonant frequency is lower in the voltage source case by about 3.5 dB. These differences can be attributed to the fact that the voltage source does not model the coaxial feed accurately. The gap of 2 mm that was created to facilitate the voltage source definition does affect the results significantly. Thus, it is quite evident that using a port is a better option for simulating the coax-fed patch antenna in HFSS. However, it should be pointed out that the voltage source simulation took significantly less time than the port simulation. Hence, in applications where a voltage source can accurately model the feed, it might be advisable to use it instead of the port.

4.6.1 Effect of Voltage Source Gap

As was described, the gap between the outer conductor and the inner conductor in the definition of the voltage source was specified to be 2 mm. It was interesting to investigate the effect of this parameter in HFSS. For this purpose, two different simulations were carried out at 1 mm and 3 mm. All the remaining parameters in the simulation were identical to those used above. Figure 4.11 shows the comparison of the three cases. It was difficult to come to a clear conclusion because no fixed trend was seen as far as the resonant frequency is concerned. However, it can be seen that the return loss at the resonant frequency reduces as the gap is widened. In other words, the Quality factor (Q) decreases as the gap is increased from 1 to 3 mm.
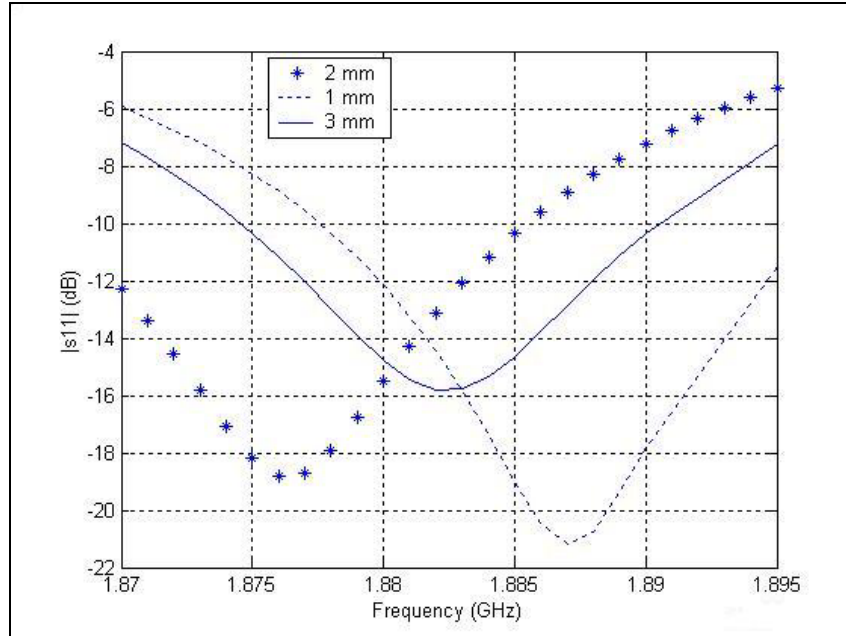
Fig 4.11: Comparison of results for different voltage source gaps

4.7 Effect of Radiation Box Size

As shown in Chapter 3, *ABC_box* was created specifically so that it can be assigned the *Radiation* boundary to facilitate far-field calculations. In all the simulations discussed till this point, the length and width of the box were identical to that of the substrate. In other words, the box did not extend beyond the ground plane.

In order to investigate how the size of this box affects the results, two simulations were performed. Each of them used voltage sources as means of excitation and had values of mesh refinement frequency and *Delta S* specified as 1.895 GHz and 0.015 respectively. The gap was specified as 2 mm. In both cases the box was extended beyond the ground plane, that is, the length and width of the box were increased as compared to the previous cases. While the original box size for the smaller ground plane case was 82.7118 x 91.496, the two new box sizes simulated were 140 x 160 and 180 x 200 where all dimensions are in mm. Also, the bottom surface of the substrate, which was earlier assigned the *Ground plane* boundary condition, was now changed to *Perfect E*. This was done to simulate the effects of a finite ground plane. Figure 4.12 shows the comparison.
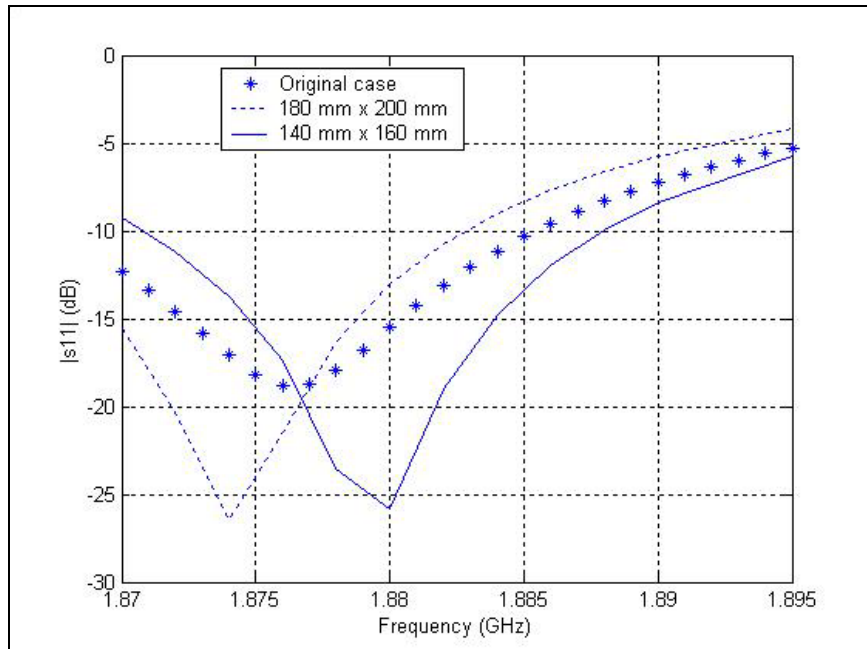
73

Fig 4.12: Effect of Radiation box size

As can be seen, there are no significant differences from the point of view of the resonant frequency. The resonant frequencies for all the 3 cases are within 5 MHz of each other. However, the return loss at the resonant frequency increases by approximately 10 dB when the box is made larger as compared to the ground plane. It should also be pointed out that a similar simulation was performed for the patch antenna that was built but the simulation failed to converge for a *Delta S* of 0.015.

4.8 Conclusion

The main aim of this chapter was to estimate how some of the major parameters affect the simulation results. It was shown that a larger ground plane produces a lower resonant frequency. Patch thickness does not significantly change the S-parameter plots. By varying the Delta S parameter, it was shown that specifying a higher error could lead to significantly inaccurate results. Similarly, various cases were simulated to examine the effect of mesh refinement frequency in HFSS. Advantages and disadvantages of using fast frequency sweep against discrete frequencies were discussed. The details of a voltage

source simulation were explained and results were compared with the port case. The voltage source gap and the radiation box size were also varied and their effects were shown.

Chapter 5

Conclusions and Future Work

This thesis detailed the various aspects associated with the modeling of a single layer, rectangular coax-fed patch antenna. One of the goals was the introduction of HFSS as an effective tool for electromagnetic analysis. An effort was made to impart understanding of the design process in HFSS, which would aid the reader in building any simulation in HFSS. Patch antenna structures that were built for the RLA project were introduced. A comprehensive and graphic description of each step taken in creating the simulation of the patch antenna was presented. The return loss values as estimated by HFSS were compared with those calculated by WIPL-D and with those that were measured. It was seen that both HFSS and WIPL-D estimated resonant frequencies within 25 MHz of the actual measured resonant frequency. However, neither was good enough to be used as a stand-alone tool in coax-fed patch antenna design. As far as HFSS is concerned, it was difficult to pinpoint specific factors responsible for the mismatch. However, the inherent use of FEM is believed to be the major factor. It was observed that the resonant frequency estimated by WIPL-D was not only higher than that of HFSS but also closer to experimental data. In order to provide a better understanding of the modeling process in HFSS, effects of varying some vital parameters were delved into. The allowable change in S-parameter magnitude and mesh refinement frequency were shown to be extremely crucial for accurate results. Although not used in the patch antenna simulation, voltage sources and fast frequency sweep were discussed in brief.

Being a program based on the finite element method (FEM), HFSS is significantly memory-limited. As mentioned in chapter 4, there were various instances, especially in the simulation of larger structures, when convergence could not be achieved because of a lack of adequate memory. These simulations were carried out on a CPU with 512 MB RAM. There is enough evidence to suggest that better and more accurate results could be achieved if HFSS were to be used with higher memory chips. In addition to accuracy, one

can also hope to significantly speed up some of the simulations by using methods such as fast frequency sweep if more memory were to be available.

A continuation of this work could be the use of HFSS to model other different types of microstrip antennas such as those with more than one layer of substrate. It is also possible to simulate structures with different patch sizes as mentioned in Chapter 3. Apart from the coaxial probe method, several other feeding methods were also listed in Chapter 3; each of which could be simulated and their performances could then be compared. Even for the rectangular patch antenna, only one specific band of frequencies was considered. Different patch antennas resonating at different frequencies and having different bandwidths can be designed and their performance can be compared with the help of simulations. In addition to the return loss values, one can also evaluate other parameters such as near-field values, far-field patterns and impedance values.

There is further scope of work in understanding the comparison of the results between HFSS and WIPL-D. As mentioned earlier, these programs use two fundamentally different approaches to solve electromagnetic problems. It would be interesting to check whether the differences seen between the results are inherent to the characteristics of the method used or are program-specific. In other words, one can repeat the simulation described in this thesis in other programs that use FEM and MoM to investigate if a similar trend is seen in the comparison.

HFSS can be effectively used to examine the performance of other antennas such as horn and helical. Structures pertaining to other fields such as low-frequency electromagnetics, planar electromagnetics, circuit design and waveguide design can be easily modeled using HFSS. Although this thesis focused on the Agilent version of HFSS, most of the principles introduced are equally valid for the newer Ansoft version. The latest version of HFSS released by Ansoft is version 9.0. This version is highly sophisticated and has applications in areas such as automobiles, aerospace and packaging.

References

[1] http://www.wipl-d.com/ie/index.html

[2] Agilent High-Frequency Structure Simulator User's Guide, 2000

[3] *Felippa, C.A.,* "Introduction to Finite Element Method," available online at http://caswww.colorado.edu/courses.d/IFEM.d/IFEM.Ch01.d/IFEM.Ch01.pdf

[4] Agilent High-Frequency Structure Simulator 'Getting Started', 2000

[5] *Balanis, C.A.,* "Antenna Theory: Analysis and Design," pp. 724-725, John Wiley & Sons, 1997

[6] *Stutzman, W.L.; Thiele, G.A.*, "Antenna Theory and Design," pp. 210-212, John Wiley & Sons, 1998

[7] *Abdallah, H; Wasylkiwskyj, W.; Parikh, K; Zaghloul, A.I.,* "Comparison of Return Loss of Microstrip Patch Antenna Calculations using WIPL-D and HFSS with Measurements", submitted for publication, 2003

Vitae

Kunal Parikh was born on June 14, 1980 in Ahmedabad, India. He earned his Bachelors of Engineering (B.E.) degree in Electronics and Communications from Nirma Institute of Technology affiliated with the Gujarat University in 2001. He then joined the Blacksburg campus of Virginia Tech as a Master's student in the Bradley Department of Electrical and Computer Engineering. In the summer of 2002, he moved to the Alexandria Research Institute of the Northern VA campus, where he worked as a Graduate Research Assistant for the rest of his program.