

033288-1-T

PROGRESS ON THE DEVELOPMENT OF FEMATS

J.R. Natzke, M.D. Casciato and J.L. Volakis

**Naval Air Warfare Center Weapons Div.
China Lake, CA 93555-6001**

October 1995

33288-1-T = RL-2461

PROJECT INFORMATION

PROJECT TITLE: Further Development of FEMATS

REPORT TITLE: Progress on the Developments of FEMATS

REPORT No.: 033288-1-T

DATE: October 1995

SPONSOR: Dr. Helen Wang
Code 472210D, Bldg 01400
Naval Air Warfare Center Weapons Div.
China Lake, CA 93555-6001

SPONSOR P.O. or
CONTRACT No.: N68936-95-M-E248

U-M PRINCIPAL
INVESTIGATOR: John L. Volakis
EECS Dept.
University of Michigan
1301 Beal Ave
Ann Arbor, MI 48109-2122
Phone: (313) 764-0500 FAX: (313) 747-2106

CONTRIBUTORS
TO THIS REPORT: J. Natzke, M. Casciato, M. Nurnberger, J. Volakis

ABSTRACT

Progress on the Development of FEMATS

John R. Natzke, Mark D. Casciato, and John L. Volakis

This report presents a summary of the improvements to the FEMATS code since the last issue of the manual (U-M Report 031155-3-T, April 1994). The upgraded FEMATS code incorporates new scripts which simplify the porting, integration, and utility of the pre-processor. Most importantly, FEMATS is now integrated with its own prismatic mesher, which is simple to use, and incorporates its own interface. The option to use SDRC I-DEAS is still available. A most important upgrade to the FEMATS code is the incorporation of a new mesh termination scheme using artificial absorbers and this makes possible the use of conformal mesh truncation surfaces. This new absorber is tested by showing calculations for the almond EMCC benchmark.

Progress on the Development of FEMATS

John R. Natzke, Mark D. Casciato and John L. Volakis

Radiation Laboratory
Dept. of Electrical Engineering and Computer Science
University of Michigan
Ann Arbor, MI 48109-2122

October 23, 1995

1 Introduction

The numerous revisions which have been made to FEMATS include the porting of the code to an HP 9000 workstation, refinement of the preprocessor I/O, and the development of a single script file which initiates all of the preprocessor and main processor programs. Both a tutorial mesh generator and an automatic prismatic mesh generator have been added. Also an optional mesh truncation scheme, Finite Element Method - Artificial Absorber (FEM-AA), has been implemented.

2 Workstation version

Although FEMATS was originally designed to run on a variety of supercomputer architectures, the main processor program was written such that it could peacefully co-exist with the preprocessors, etc., on the workstation. Thus after a number of minor changes, the source code for the Cray was converted to run on an HP 9000 workstation. Necessary changes were made to the `Makefile`, which is used to compile the source code and install the executable program of the main processor. Each of the real and complex variables and arrays are declared as double precision during compilation, in an effort to maintain the inherent extended precision of the supercomputers. Also, the array dimensions were reduced to account for the reduced available disk space on the workstation. These dimensions can be readily changed in the parameter file `fem_data.h` to reflect the size of a particular geometry or the available space on a particular machine, and the procedure to do so is

described in the revised User's Manual. If any of the array dimensions are too small for a given geometry, an overflow error will be given.

The workstation version of FEMATS has been verified and for a system with 9712 unknowns the CPU per iteration was less than 0.2 seconds on an HP 9000/715. It was found that although the conversion to double precision slowed the program down, the convergence rate increased, and the resultant run-time was approximately the same as when using only single precision.

3 Preprocessor refinement

Considerable refinements have been made to the preprocessor programs of FEMATS to increase the efficiency of the user interface. The input data requirements have been minimized, reduced by about 50%. This was accomplished by extracting all of the pertinent geometry related information from the universal file which is unchanged from run to run. The input parameters entered during each run-time are then only the desired tolerance, maximum number of iterations, aspect parameters (type of pattern, etc.), and material properties which the user may wish to vary for each run. The I/O of the preprocessors has been consolidated, and a number of the prompts have been made more self-explanatory. In addition, overflow checks for the array dimensions have been added at necessary points to prevent an erroneous run from taking place.

4 Script file

The user interface for FEMATS has been changed extensively. A single script file is now used to initiate all of the preprocessing and main processing programs. The only exception is if the final processing is to be done on a different machine. In most cases this would be a supercomputer, and the preprocessing would be done on a workstation. This requires that a second script be started on the supercomputer once the geometry and run-time data files have been transferred. This second script can be easily modified to account for machine specific dependencies, to automatically submit the job to a batch queue, etc.

Using the script files in this way increases the user friendliness of the interface since running FEMATS requires a series of steps for its completion. For example, the script file checks whether or not the preprocessing has been done already and skips that step if it is not necessary to repeat it. Also within the script some file management is done, such as deleting unnecessary files or compressing others when appropriate. This was of particular concern for the workstation version, since the geometry mesh files can consume considerable amounts of disk space, especially if numerous geometries are being processed at the same time. If the final processing is to be done on a different machine,

then the script will `tar` and `compress` all of the required data files, producing a single file for ease of transfer.

Another distinguishing feature is that the prompting for the input parameters is now done by a separate file initiated after the completion of the preprocessing and not from the main processor program. In this manner, the final processing is completely noninteractive. This change was made so that all of the user input could be done on the workstation for the cases when the final processing is to be done on a supercomputer. The primary motivation for this was that the eventual implementation of the GUI will require that all of its interfacing be done within the workstation environment. Thus the time and effort was invested for this as well as a portion of the aforementioned modifications to prepare FEMATS for the development of the GUI.

5 Tutorial Mesh Generator

The development of a mesh for a given geometry typically requires a large portion of the time devoted to completing the entire FEM process. Therefore, to free the first time user from the time consuming burden of generating meshes and to enable them to focus on the operation of FEMATS itself, a tutorial mesh generator was developed for a select number of the benchmark test cases.

This mesh generator is based on a structured mesh approach (vs. a free mesh approach) which utilizes prismatic elements. Volume meshes are generated for rectangular, cylindrical, or spherical regions containing simple geometries which readily adhere to these coordinate systems. The geometries which can currently be meshed automatically are as follows:

1. PEC rectangular plate: flat
2. Dielectric rectangular plate: finite thickness
3. PEC rectangular inlet: dielectric filled
4. Dielectric cylinder
5. PEC circular inlet: dielectric filled
6. PEC sphere

Upon selecting the geometry, the required input is simply the dimensions of the geometry and mesh termination and the average element dimension (typically $0.075\text{--}0.085\lambda$). The first step of the code is to triangularize the surface mesh—a planar surface for the rectangular and cylindrical regions or a closed surface for the spherical region—which contains an appropriate layer or cross section of the structure. Once generated, prisms are grown from this surface, and the appropriate element faces are selected to identify the given geometry. Examples of the three types of structured meshes are

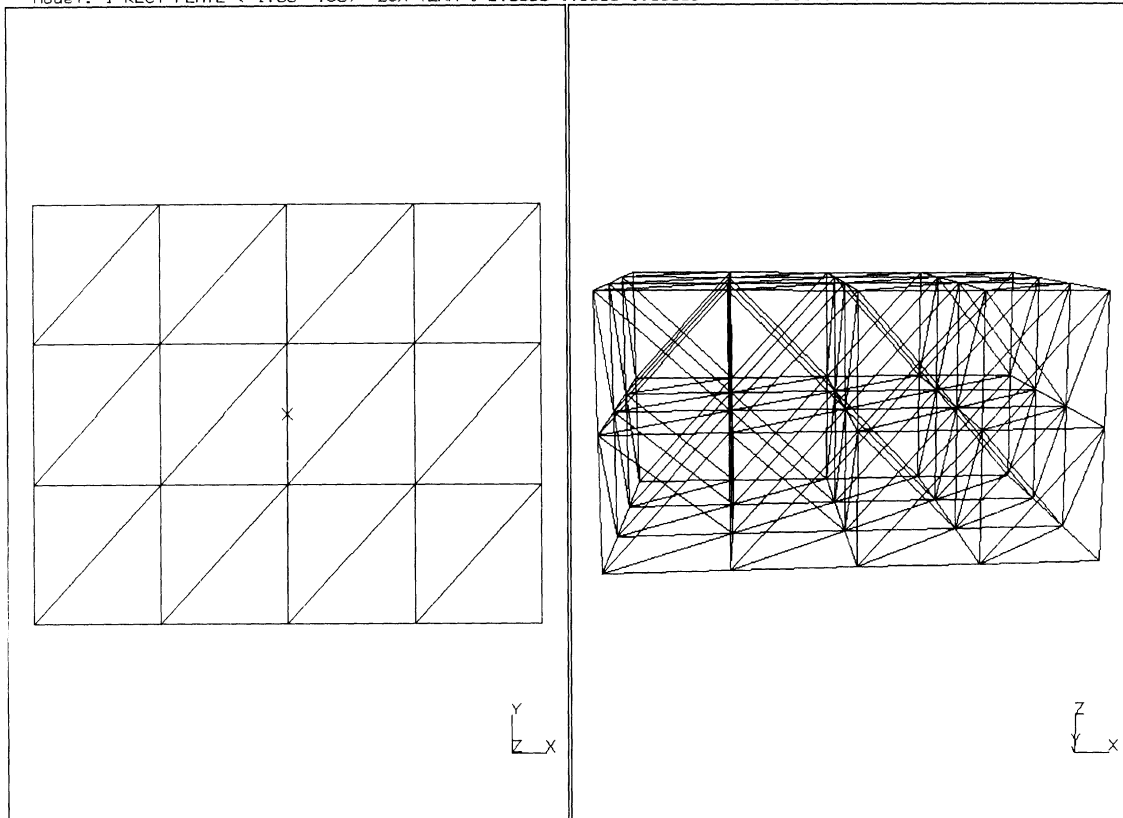


Figure 1: Structured mesh for a rectangular region.

shown in Figs. 1 through 3, which include both the surface triangularization and the grown volume mesh. Since at this time FEMATS can only handle tetrahedral elements, the prisms are further divided into tetrahedrals, each prism giving three of them. This growth and subsequent division of the prisms is an extremely efficient process due to the structured nature of the mesh.

The tutorial mesh generator automatically outputs the FEM mesh in the particular Universal file format required by FEMATS. The generated mesh can be viewed by any graphics software which accepts the Universal file format, e.g. I-DEAS. The user then simply needs to continue with the preprocessing, input parameters, and so forth to obtain scattering data of the chosen geometry.

As a validation of this mesh generator, scattering patterns were obtained and compared with the FEMATS benchmark results, which were each based on free meshes generated by I-DEAS. The structured meshes were generated with the same termination dimensions used for the benchmark test cases wherever possible, and an effort was made to match the total number of unknowns (edges) as well. Figure 4 shows the comparison between the two meshing techniques for the $3.5\lambda \times 2\lambda$ rectangular PEC plate, and excellent

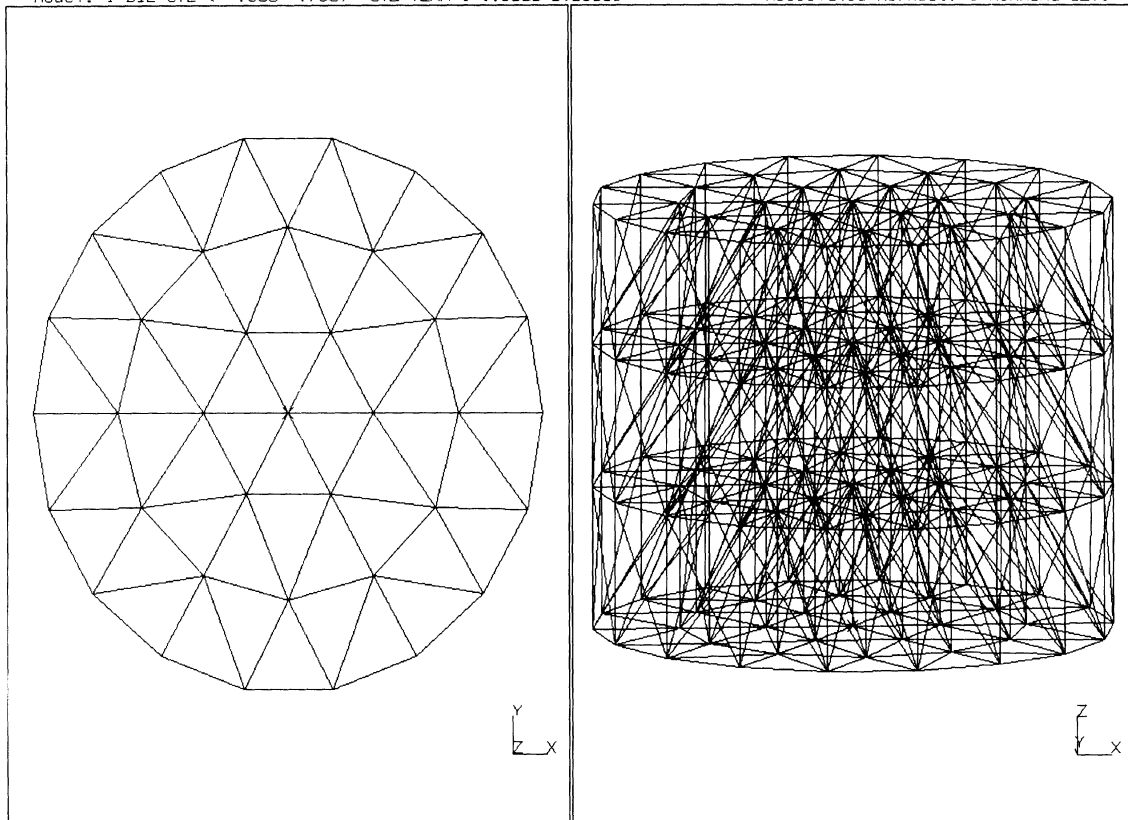


Figure 2: Structured mesh for a cylindrical region.

agreement is obtained. In Fig. 5, backscatter patterns in two different cut planes are shown for the $1.75\lambda \times 1\lambda \times 0.125\lambda$ glass plate. The discrepancy between the results of the two techniques is primarily due to the homogeneous element size of the structured mesh, compared to the varying element size in the free mesh, which enables the decrease in element size within the glass plate to account for the much shorter wavelength. Thus, for the same dimensions and approximately the same number of total unknowns (~ 150000), the free mesh models the plate itself with 28584 elements, while the structured mesh would model it with only 5916 elements. The structured mesh used to generate the results in Fig. 5 had a total of 231666 unknowns and 8160 elements within the glass plate. Another difference between the two models and thus a possible contributor to the discrepancies is that the structured mesh used a conformal box termination, whereas the free mesh model used the mix termination scheme containing rounded edges and corners.

Results for the rectangular and circular inlets are presented in Figs. 6 and 7, and conformal mesh terminations were used in both cases. As shown, excellent agreement between the two meshing techniques was obtained for the rectangular inlet in the $\phi = 0$ plane, and the same agreement was observed for a $\phi = 45^\circ$ cut. For the circular inlet, the agreement is not as good in the

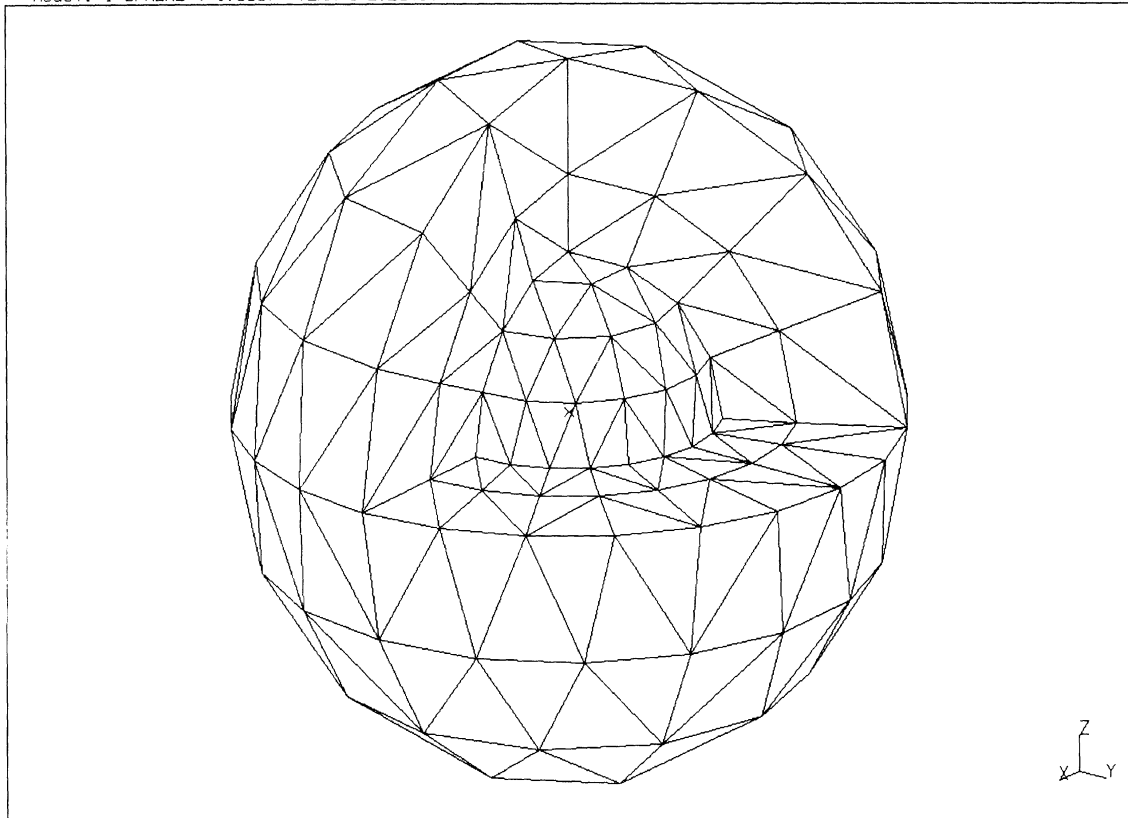
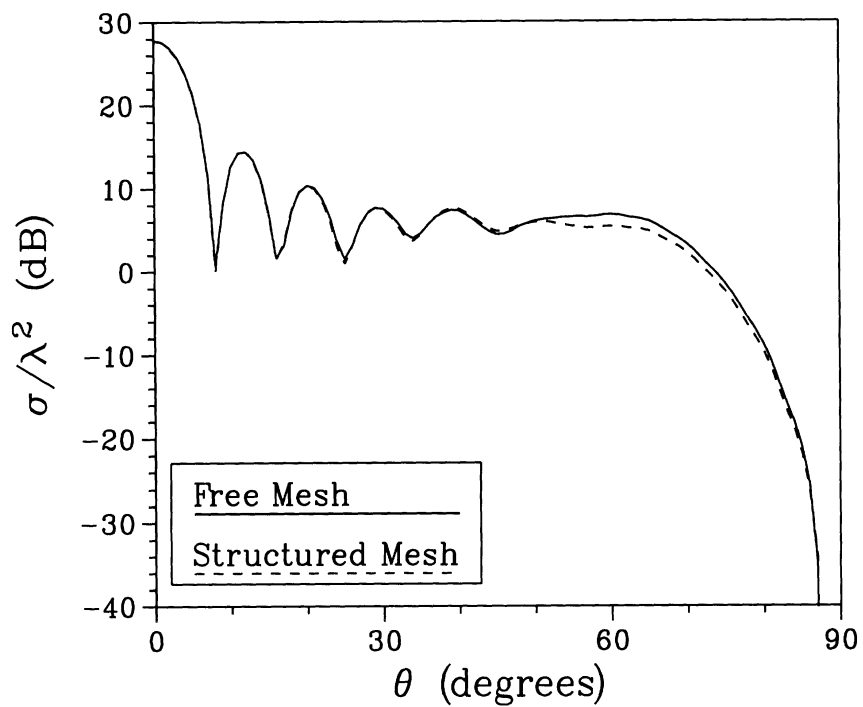
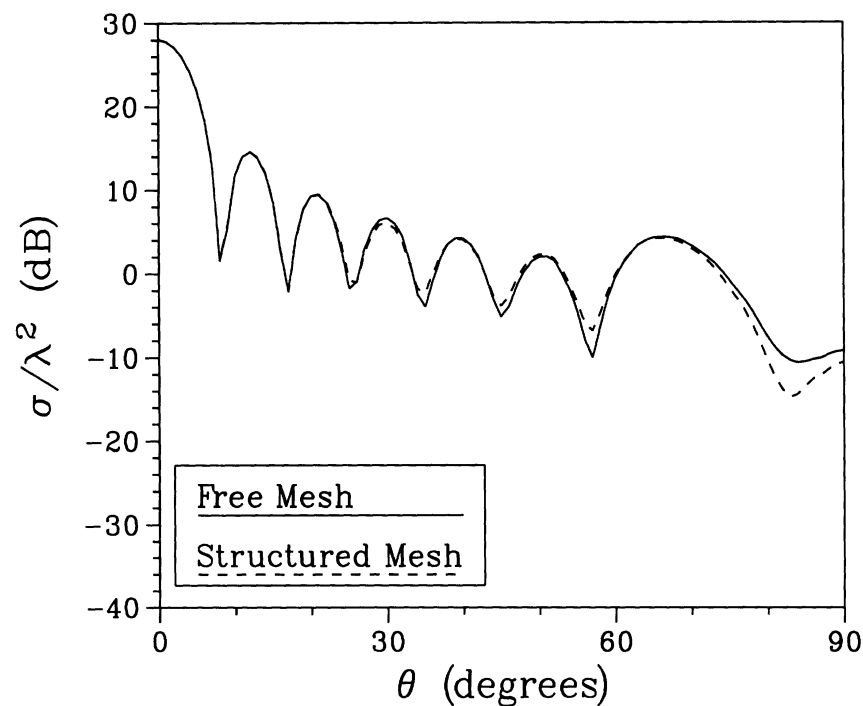


Figure 3: Structured mesh for a spherical region.

transition region between the aperture ($\theta = 0$) and side ($\theta = 90^\circ$) normal incidences. The jaggedness of the curves in Fig. 7 also reveals a limitation to the bi- to monostatic conversion which was used to generate the additional four data points within $\pm 2^\circ$ for each incidence angle. For such scattering geometries, the bi- to monostatic conversion range should be reduced to $\pm 1^\circ$ or possibly not invoked at all over certain aspects of incidence.

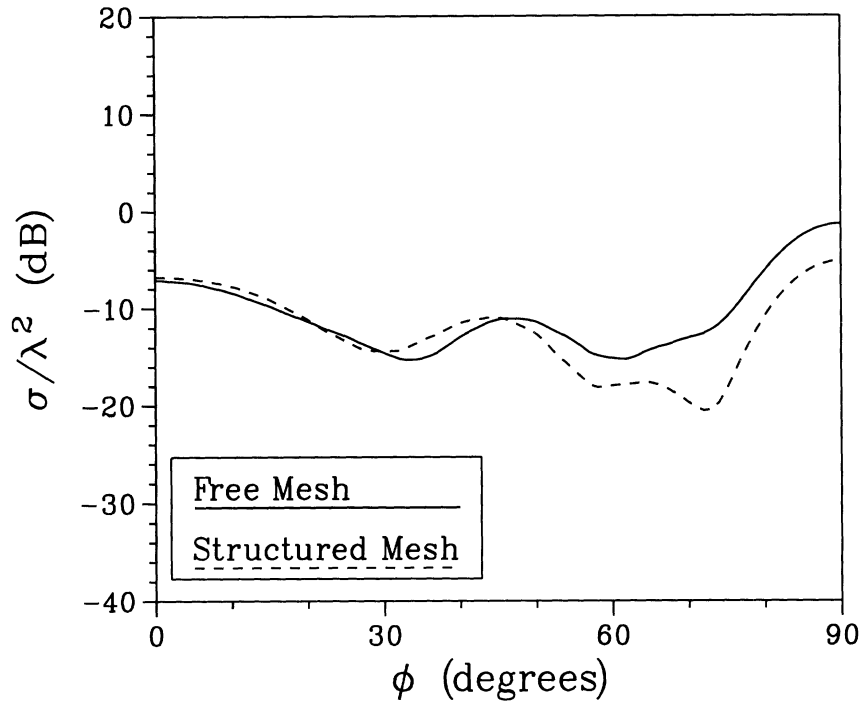


(a)

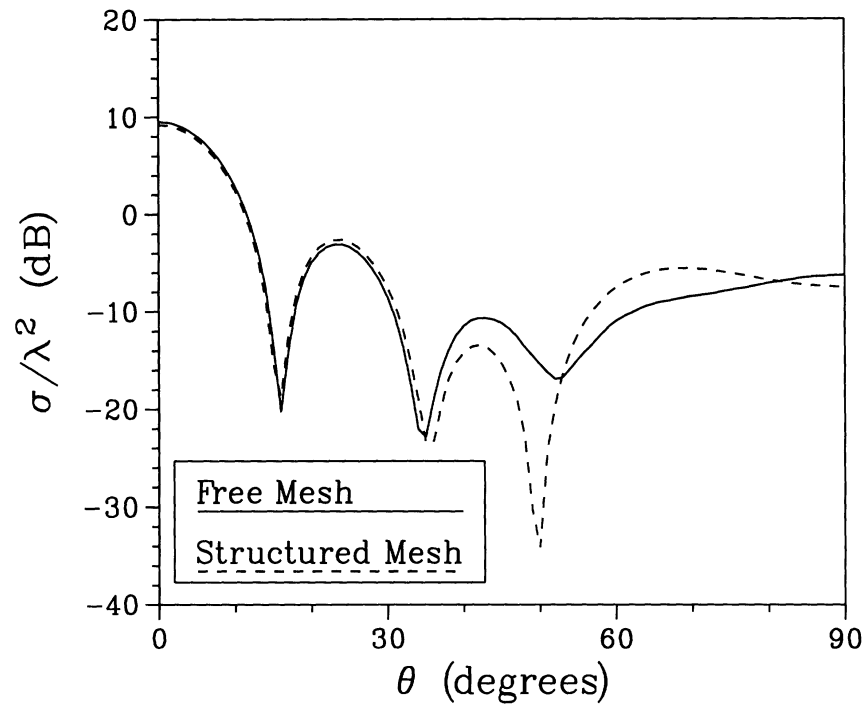


(b)

Figure 4: Comparison of backscatter patterns using free and structured meshes for the $3.5\lambda \times 2\lambda$ rectangular PEC plate with $\phi = 0$. (a) E_{θ}^i -polarization, (b) E_{ϕ}^i -polarization.

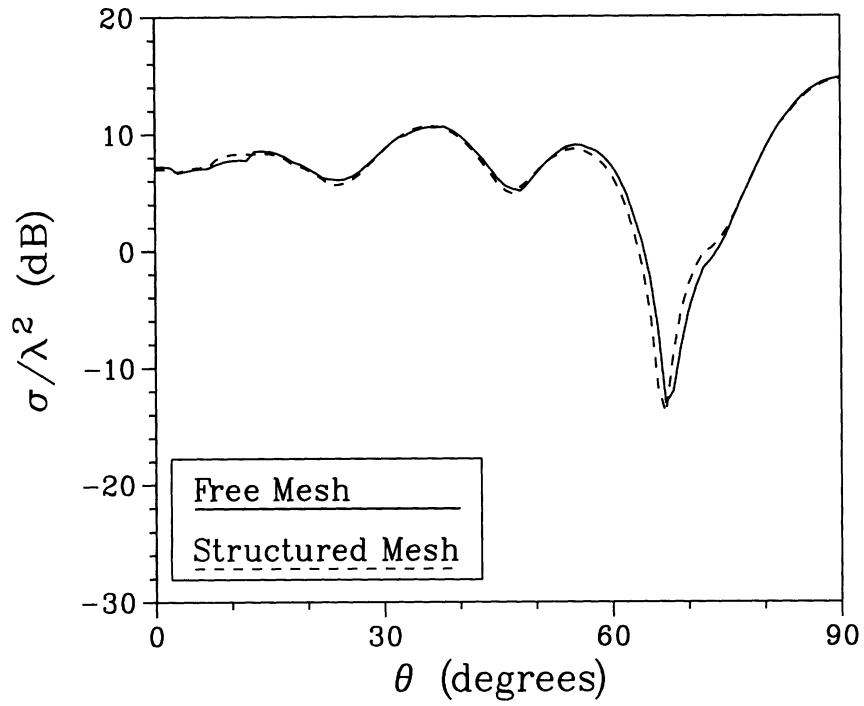


(a)

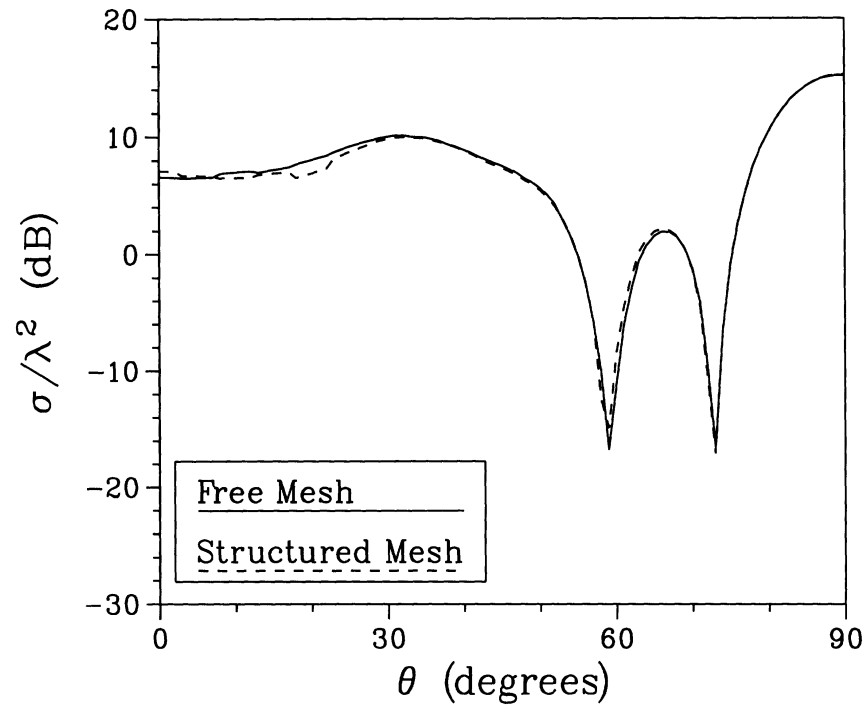


(b)

Figure 5: Comparison of backscatter patterns using free and structured meshes for the $1.75\lambda \times 1\lambda \times 0.125\lambda$ rectangular glass plate with $\epsilon_r = 3 - j0.09$ for E_ϕ^i -polarization. (a) $\theta = 80^\circ$ cut, (b) $\phi = 0^\circ$ cut.

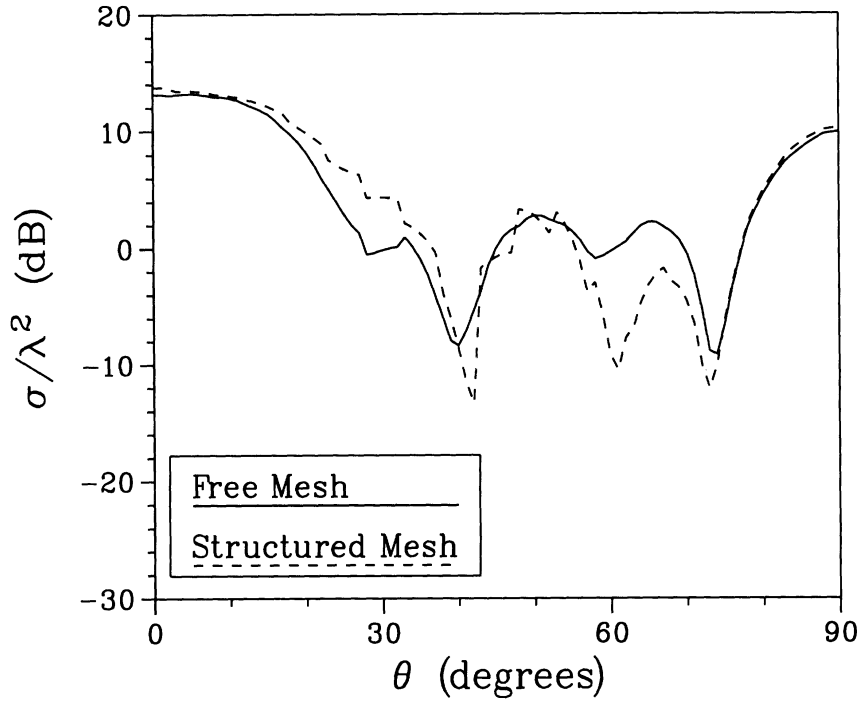


(a)

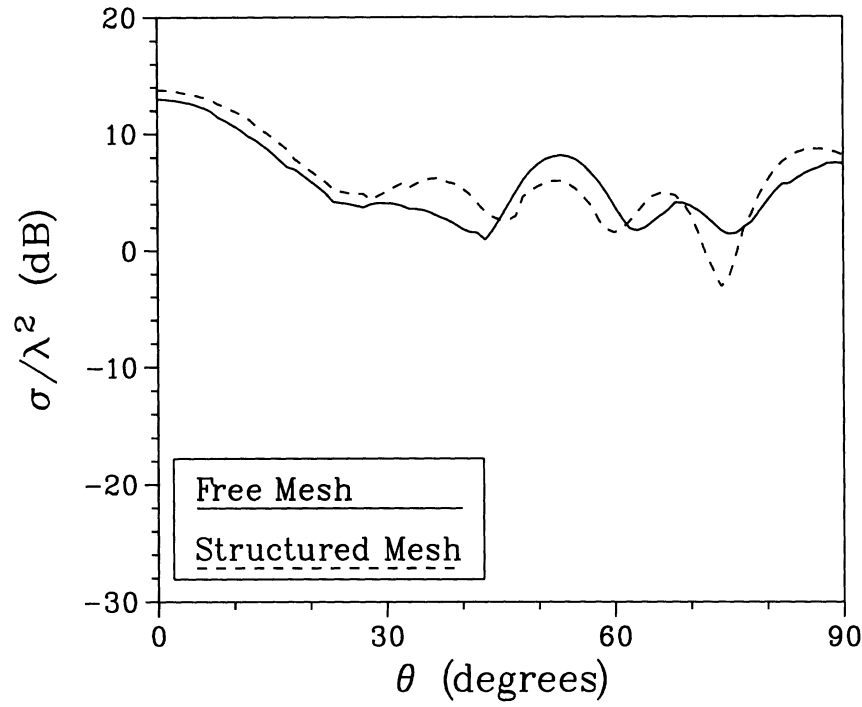


(b)

Figure 6: Comparison of backscatter patterns using free and structured meshes for the $1\lambda \times 1\lambda \times 1.5\lambda$ rectangular inlet with $\phi = 0$. (a) E_θ^i -polarization, (b) E_ϕ^i -polarization.



(a)



(b)

Figure 7: Comparison of backscatter patterns using free and structured meshes for the circular inlet with $h = 1.875\lambda$, $dia = 1.25\lambda$ and $\phi = 0$. (a) E_θ^i -polarization, (b) E_ϕ^i -polarization.

6 Prismatic Mesh Generator

The primary advantage of employing prismatic meshes is that it greatly simplifies the mesh generation process. Although the tutorial mesh generator described in the previous section uses this technique, its usefulness is limited in that it can only handle a few select geometries. A mesh generator which could handle more arbitrary geometries generated by CAD software such as ACAD was desired. An automatic prismatic mesh generator called PRISM was chosen to generate input files for FEMATS. PRISM was provided by Shishir Pandya of NASA Ames Research Center. PRISM provides several options for mesh generation, hyperbolic explicit, hyperbolic implicit, optimization, and averaged normals. For details of the techniques implemented in PRISM users should contact Shishir Pandya at Ames. In order to make PRISM more user-friendly a Graphical User Interface (GUI) was written. In addition a conversion routine was written to transform an ACAD facet file into a format compatible with PRISM. The output of PRISM is an IDEAS compatible universal file consisting of either prisms or the tetrahedrals needed in the current version of FEMATS.

Figure 8 shows the outer mesh surface of the NASA benchmark almond grown from an ACAD facet file with PRISM using the "average normals" selection. Figure 9 shows the mesh with the inner almond surface exposed. At this time the ACAD to PRISM convertor will only handle ACAD facet files consisting of a single entity. As Figure 8 shows the "averaged normals" selection produces a smoothly layered mesh by using the normals of surrounding elements as a guide for element shape and depth. At this time is is the only PRISM selection used for mesh generation.

Figure 10 shows the PRISM gui interface. Referring to Figure 10, the user must input the following information:

1. Name of Surface Datafile - This is the input .dat file which is generated by the ACAD to PRISM converter routine.
2. Number of Steps - This is simply the number of layers desired.
3. Number of Shells to Save - This is always one more then the number of steps. A shell is the surface between layers.
4. Initial Step Size - This is simply the desired thickness per layer.
5. Total Marching Distance - This is the total thickness of all layers if Stretch is on. For our applications Stretch is off, so this input is ignored.
6. Stretch - This is a method to vary layer thickness automatically. For our application Stretch is not used at this time.
7. Surface Triangulation Definition - This is the way normals are defined on the 3 sided patches (facets) on the surface of the target in the ACAD

```
SDPC  I-DEHS VI.1:  FE_Modeling_&_Analysis  20-OCT-95  12:28:40
Database: none
View: No stored view
Task: Mesh Creation
Model: 1-EPISM  RMS = 2) (HL-1-10.UM3)
Display: No stored Option
Model Bin: I-MAIN
Associated Workset: 1-MODKING_SET1
Units : IN
```

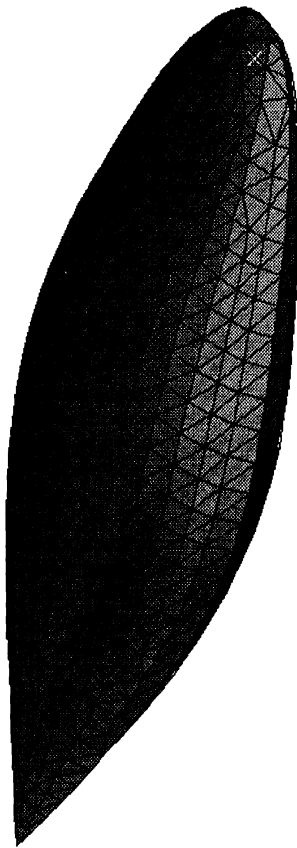


Figure 8: FEM Mesh - NASA Metallic Almond - Outer Surface

```
Database: none  
View: No stored View  
Task: Mesh Creation  
Model: 1-PRISM (MS = 2) (SHL-1-10,0HW)  
SUFC I-DEHS VI.1: FE_Modeling_8_Analysis  
20-OCT-95 12:31:27  
Units: IN  
Display: No stored Option  
Model Bin: 1-MBIN  
Associated Workset: 1-MDPF:INS_SET1
```

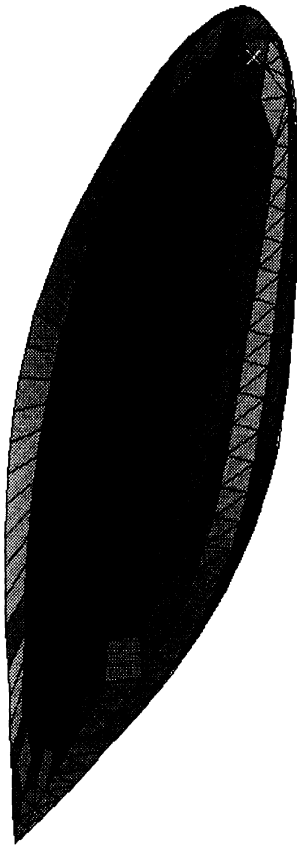


Figure 9: FEM Mesh - NASA Metallic Almond - Exposed Inner Surface

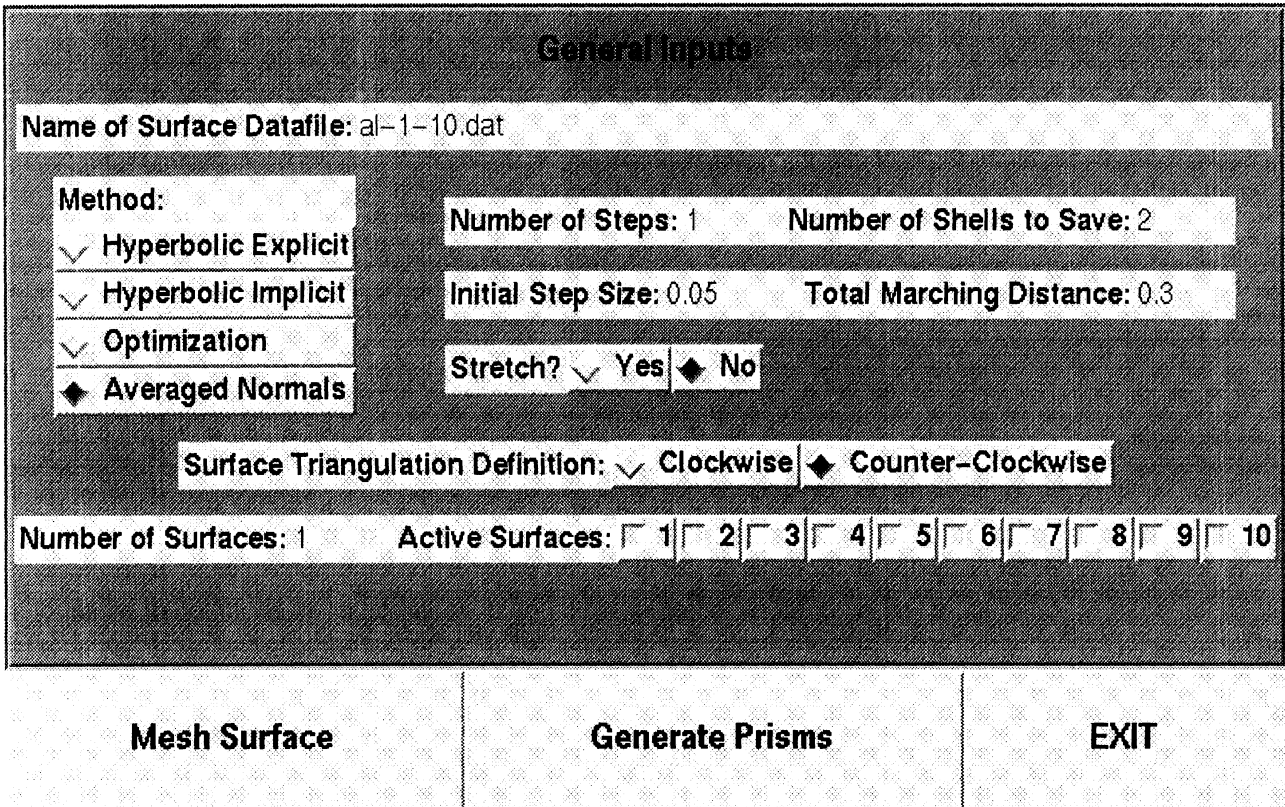


Figure 10: PRISM GUI

facet file. For ACAD facet files Counter-Clockwise corresponds to Right Hand Normal (RHN) convention defined in ACAD. This means that if you curl your fingers around a facet coincident with the node numbering increment, your thumb points in the direction of the normal. If you look in the ACAD facet file and follow the RHN convention you will find that the normals of the facets points outwards from the almond.

Once all inputs are set, simply click on Generate Prisms. A temporary window will open for user interaction. When PRISM is done this window will prompt the user on whether prisms or tetrahedrals are desired and on whether the user wants the almond surface layer to be saved as a different color. For FEMATS tetrahedrals are necessary.

When this step is done IDEAS must now be used to do the final processing on the universal file. THE INNER SURFACE LAYER (TARGET SURFACE) MUST BE DELETED AT THIS STAGE. This layer consists of 3 sided patches and not tetrahedrals. If left in, the FEMATS preprocessor will crash. Note that the inner surface must be deleted, not the inner most group of nodes. Next the user simply uses IDEAS to define nodal groups and elements materials as described previously in the FEMATS manual. The target is now ready for the FEMATS preprocessor.

7 FEM Artificial Absorbers (FEM-AA)

The previous version of FEMATS employed a 2nd order Absorbing Boundary Condition (ABC) for mesh termination. This type of termination requires curvature information over the entire target which can be quite cumbersome to implement for all but the simplest of shapes. Because of this, the previous version of FEMATS allowed only 3 mesh termination shapes, spherical, cylindrical, and flat, or a combination of these. A more conformal method of mesh termination is therefore desired to allow for scattering calculations from arbitrarily shaped geometries without having to make major changes in FEMATS. This led to the selection of the FEM-AA mesh truncation scheme. Readers are referred to [1] for the theory behind this method. Using this scheme, the mesh is terminated using a metal backed absorber. The absorber is composed of 3 layers each having the constitutive parameters $\epsilon_r = \mu_r = (1-j)2.7$. Note that because $\epsilon_r = \mu_r$ the wave impedance of the absorber (at normal incidence) is the same as that of free space, thus, providing minimal reflection at the air/absorber boundary at off normal incidences. To implement the absorber into FEMATS it was only necessary to turn off the incident field on the metal backing the absorber. To do this the absorber was assigned a unique identification number. If the user desires to use the FEM-AA option in FEMATS the following steps must be taken:

1. When grouping nodes in the universal file the outer layer of nodes must be designated conducting (Group Label Prefix C).
2. The elements in the absorber must be given material code "2". Material code "2" is now reserved for artificial absorbers only. As stated in the FEMATS users manual, material number 1 is reserved for air.
3. Do not group nodes in the absorbing layers or on the air/absorber interface as dielectric or any other material group.
4. Do not group any nodes as lying on the outer boundary (Group Prefix A).
5. When queried by the preprocessor about the type of outer boundary used to terminate mesh and associated boundary input (location of center of boundary etc.) simply put in dummy arguments.
6. When inputting material parameters use $\epsilon_r = \mu_r = (1-j)2.7$ for material 2.

If the user desires to make a FEMATS run using ABC's simply run as directed in the previous FEMATS documents and do not assign the material code "2" to any other dielectric.

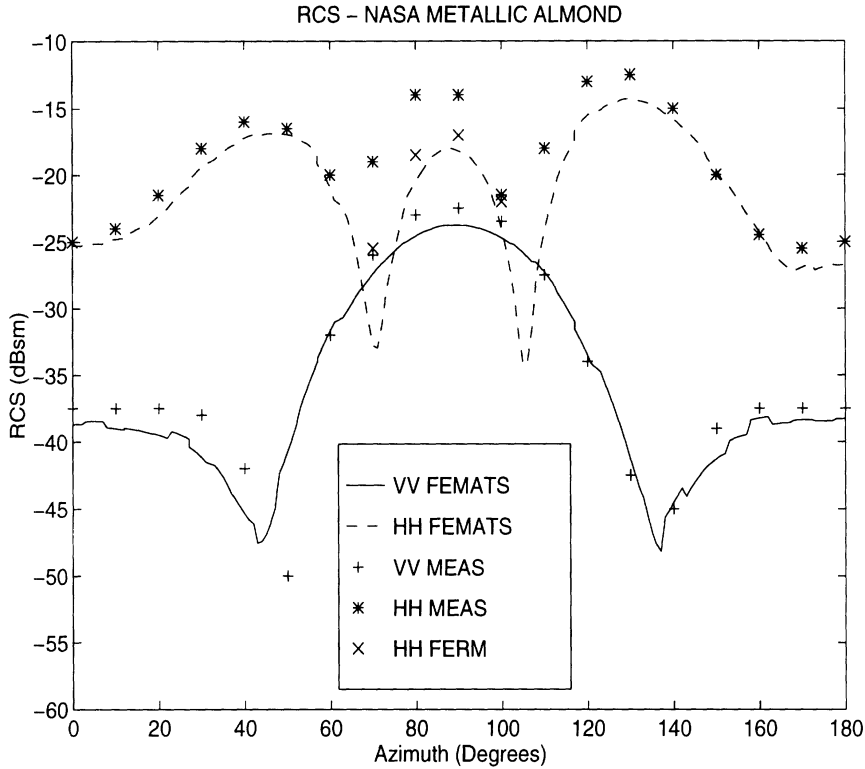


Figure 11: RCS - NASA METALLIC ALMOND - 1.19 GHz

8 Validation - NASA Metallic Almond

The NASA metallic almond shown in Figure 8 was selected as a further validation of both FEMATS and the FEM-AA mesh truncation scheme. The almond is 9.936 inches long. The target was interrogated at 1.19 GHz where it is 1λ long and is sampled at approximately $1/30$ of a wavelength. The mesh consists of 9 layers, each layer 0.05λ long for a total thickness of 0.45λ . The inner 6 layers are air and the outer 3 absorber. The mesh has of 46,878 edges with the outer shell conducting. Figure 11 shows the 1.19 GHz run from 0 to 180° azimuth for both VV and HH polarization, 0° being the point of the almond with the pattern cut along the narrow cross section of the almond. The FEMATS run is compared with both measured data and FERM (a Method of Moments (MoM) code). As can be seen, FEMATS tracks the measured data quite well. The only significant difference is near 90° aspect for the HH polarization. However the FEMATS data are nevertheless in agreement with the results from the FERM code.

The almond was also interrogated at 3.57 GHz where it is 3λ long and is sampled at approximately $1/10$ of a wavelength. The mesh dimensions are as in the previous example. The mesh has 43,578 edges. Figure 12 shows the 3.57 GHz run from $\theta = +90^\circ$ to -90° for HH polarization with ϕ held constant at 0° . -90° corresponds to the point of the almond. FEMATS is compared with an "exact" MoM code, also sampled at $1/10\lambda$, and compares

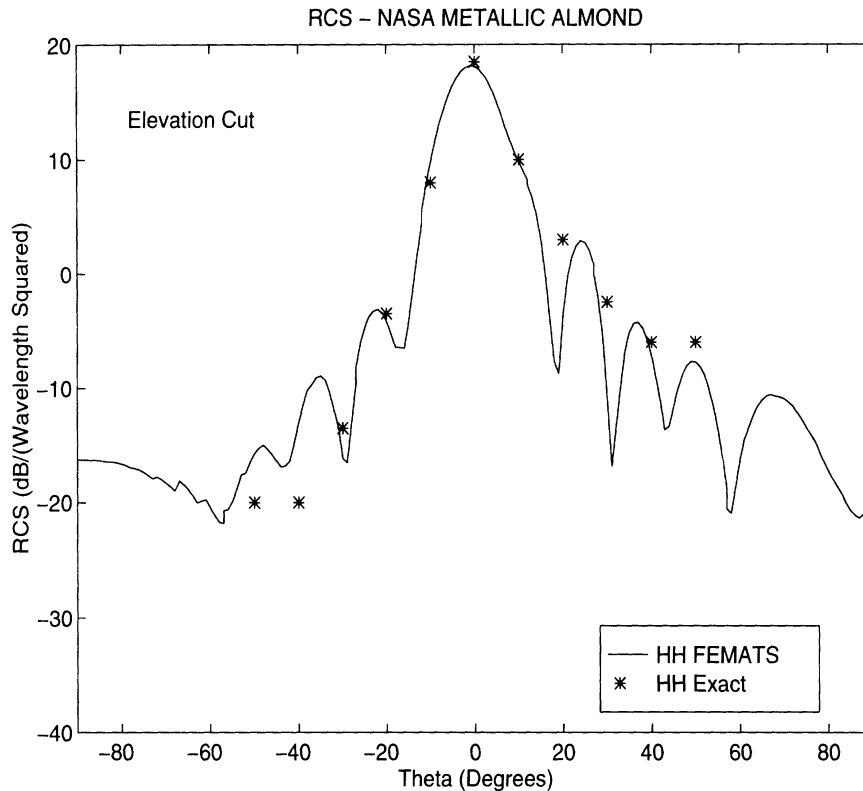


Figure 12: RCS - NASA METALLIC ALMOND - 3.57 GHz

favorably.

Computer Performance

All Almond runs were made on a CRAY Y-MP C90. For the 1λ almond the mesh contained 46,878 unknowns(edges). For VV polarization total CPU time was 464 seconds with an overall flop rate of 315 Mflops. For HH polarization total CPU time was 939 seconds with an overall flop rate of 361.7 Mflops. For the 3λ almond the mesh contained 43,578 unknowns. Total CPU time was 1010 seconds with an overall flop rate of 366.8 Mflops.

9 Future Work

Planned improvements to FEMATS include:

GUI

A gui interface is in the works and very close to completion. This will be a major improvement in the code from the users point of view.

Further Validation

With the implementation of FEM-AA mesh truncation in FEMATS the validation procedure can now be extended to larger and more complex objects. Planned validation runs to be performed include a coated target such as the coated NASA almond, and a complex object such as the VFY218 benchmark aircraft.

Portability

Plans to extend the portability of FEMATS include porting the code to both the IBM SP2 and CONVEX SPP-1000 parallel machines. The CONVEX machine uses the same HP processor that is in the HP workstations. With minor modifications in the makefile the HP serial version of FEMATS will compile and run on the CONVEX. The CONVEX compiler also has the option of automatic parallelization. While this is by no means the most efficient way to parallelize the code (automatic parallelization can actually slow the code down) it gives the user another platform to run the current serial version on. Due to the large amount of physical memory on the CONVEX the user can run serial problems that would not run on a standard HP workstation. Future plans do however include true parallelization of FEMATS for the CONVEX platform.

Anisotropic Absorbers

Implementation of anisotropic absorbers as an optional mesh truncation scheme. While the FEM-AA works quite well a perfect match is achieved at the air/absorber interface for normal incident only. Anisotropic absorbers will be matched at any wave incident angle on the air/absorber interface and should show good results with less layers.

Other future tasks include upgrades of the FEM assembly to make direct use of the prisms, adaptive error control and higher order elements.

References

1. T. Ozdemir and J.L. Volakis, "A comparative study of an absorbing boundary condition and an artificial absorber for truncating finite element meshes." *Radio Science.*, vol. 29, pp. 1255-63, September-October 1994.