HYBRID FINITE ELEMENT AND MOMENT METHOD SOFTWARE FOR THE SERAT ARRAY

3rd Quarterly Report

Sanders, A Lockheed Martin Co.
95 Canal Street NCA1-6268
P.O. Box 868
Nashua, NH 030601-0868

PROJECT INFORMATION

PROJECT TITLE: Hybrid Finite Element Design Codes for the SERAT Array

REPORT TITLE: 3rd Quarterly Report

U-M REPORT No.: 035067-3-T

CONTRACT
START DATE: October 1996
END DATE: September 1998

DATE: July 1, 1997
(3rd Quarterly Report)

SPONSOR: Roland Gilbert
SANDERS, INC, A Lockheed Martin Co.
MER 24-1583
PO Box 868
Nashua, NH 030601-0868
Phone: (603) 885-5861
Email: RGILBERT@mailgw.sanders.lockheed.com

SPONSOR
CONTRACT No.: P.O. QP2047

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
volakis@umich.edu
http://www-personal.engin.umich.edu/~volakis/

CONTRIBUTORS
TO THIS REPORT: Y. Erdemli (UM), Thomas Eibert(UM)
D. Jackson(UH), J. Volakis(UM) and D. Wilton(UH)

TABLE OF CONTENTS

LIST OF FIGURE CAPTIONS IN SUMMARY SECTION

CHRONOLOGY of Events (Updated Every Quarter)

- April 1996 — Proposal Submission
- July 1996 — Answers to Proposal Questions
- August 1996 — Began Contract Negotiations
- 20 Sept. 1996 — Kickoff meeting at Ann Arbor (attended by Sanders, UM and UH)
- October 1996 — Contract Signed between U-M and Sanders in Mid October
- October 1996 — Subcontract to the Univ of Houston (formalized in early November)
- 15 Nov. 1996 — **SERAT Review meeting** (at Nashua)
- 9 January 1997 — **Submitted First Quarterly Report**
  Report Described Code Plan and Progress on the Moment Method FSS Code. Specifically, a new scheme was developed to accelerate the convergence of the periodic Green's function
- 28 February 1997 — **Prepared viewgraphs on the project's progress review.**
  Showed first validation results for the moment method FSS code with the new accelerated Green's function; showed results for a new algorithm to accelerate the boundary integral truncation of the planar and curved FSS hybrid FEM code using the Adaptive Integral Method(AIM) and CVSS, the new LU solver specialized to sparse matrices
- 5 April 1997 — **Submission of Second Quarterly Report.**
  Report included the first validation results for the stand alone small array FEM code (with dipole FSS elements and dipole antenna elements). A similar validation was done for the moment method FSS developed at Houston. The fast AIM algorithm was described for boundary truncation and the TRIANGLE surface mesher was introduced to generate the aperture mesh, subsequently grown down to the FSS.
- 30 May 1997 — **Semi-annual review at the Univ. of Michigan**
  (attended by all parties)
  Review covered progress up-to-date. At this meeting, emphasis was on the validation of the three codes which took 'shape and form' between March-May 1997 in accordance with the proposed schedule. Theory, validations and comparisons among the codes were presented.

- 25 June 1997 **SERAT review Meeting** (Nashua)
- 5 July 1997 **Submission of Third Quarterly Report**
  The major component of this report was the description and validation of the periodic hybrid FEM code, FSS-PRISM. Comparisons among the FSS-EIGER, FSS-BRICK and FSS-PRISM were given for the first time. Also, the geometry drivers for the FSS-BRICK and FSS-EIGER were given.

MEETINGS

Two meetings were held this period
See the Chronology list above

SUMMARY OF 3RD QUARTER PROGRESS

1. Previous Quarter Progress Summary

During the previous quarter we reported on

- Small array code (FSS-BRICK) along with validation data for dipoles on multilayered FSS. It was primarily noted that this code was fast because it employed mesh compression algorithms for the boundary integral truncation. However, its geometrical generality was limited due to the brick elements used for mesh modeling.
- Simple moment method code FSS-EIGER for simulation of antenna elements on commensurate FSS structures. Some preliminary validations were given.
- Method for extracting reflection and transmission coefficient parameters in connection with the FSS-BRICK
- Performance of a matrix compression scheme based on the Adaptive Integral (AIM) method. A ten-fold decrease in the CPU and memory requirements of this fast integral method was shown. Therefore, AIM is particularly attractive for simulating the doubly curved FSS structure where it would be necessary to treat the entire array surface without making of the periodic Green's function.

2. Summary of 3rd Quarter's Progress.

This quarter has been extremely productive and pushed us ahead of schedule with the successful completion of the hybrid FEM code FSS-PRISM. The latter was developed by Dr. T. Eibert, a new addition to the SERAT code development team. Dr. Eibert will be responsible for the doubly curved array simulation during the second year. More details on FSS-PRISM are given in the appropriate section of this report.

Below we summarize this quarter's activity. However, the reader is also referred to Appendix 2 which contains the entire slide presentation given at the June 25, 1997 review of the project. These viewgraphs give a concise overview of the code development and validation status. Also, Appendices 3-5 include more detailed viewgraphs of the presentation given on May 30, 1997 during the review held in Ann Arbor. Appendices 3-5 cover the capabilities of the codes FSS-EIGER, FSS-PRISM and FSS-BRICK in this respective order.

2.1 Geometry Driver for FSS-BRICK

A geometry Driver was developed for FSS-BRICK. As noted above and in the previous report, FSS-BRICK is our small array code and its function is to provide a fast finite array analysis tool. As note in the attached short description of the FSS-BRICK, the geometry Driver developed this quarter uses antenna and FSS element primitives to simplify its utility and a need to interface with external meshing packages. The meshing is done automatically and is based on the primitive element choices, FSS layers and their

composition. Because the code uses bricks as the basic element for volume and element modeling, it may be necessary to make approximation for modeling the entire SERAT panel. Here is a list of its present capabilities:

- Uses slots, dipoles, crossed-slots and crossed-dipoles for primitives (see Figure 1)
- Can accommodate resistive cards of rectangular shape, placed at any layer interface
- Feeds are horizontal (x or y directed) current probes placed at any location on the uniform grid. Vertical probes are also available but not through the present Driver.
- Lumped loads can be placed at any node location between two horizontal or two vertical nodes.
- Conducting posts can be placed at any node location in much the same way done for the lumped loads. These posts can be concatenated to form long feeds and wires running from the base of the FSS panel to the surface.

FSS-BRICK has been validated for slots and gives the same results as the FSS-EIGER and the FSS-PRISM. Figure 2 shows such as a comparison of all the codes under development. Comparisons among dipole arrays is still in progress. We are improving the reflection/transmission coefficient extraction process for finite array apertures. So far, good agreement is obtained for near resonance computations and this is to be expected because of the finite aperture.

In the next three months , we will

1. Complete testing of the FSS-BRICK and its geometry Driver for dipole array simulations (input impedance vs. scan angle, FSS reflection/transmission coeff., etc.), resistive cards and lumped loads

2. Package the code with a short manual and plotting using XMGR

The graduate student Y. Erdemli will then continue to implement a similar geometry Driver for FSS-PRISM. Depending on the success of FSS-PRISM and FSS-EIGER for modeling non-commensurate FSS, we may also decide to extend the application of FSS-BRICK to non-commensurate FSS and periodic array structures by adding phase boundary conditions and mesh truncation using periodic Green's function. Coupled with the FFT for performing the matrix-vector products in the solver, FSS-BRICK may become our fastest code.

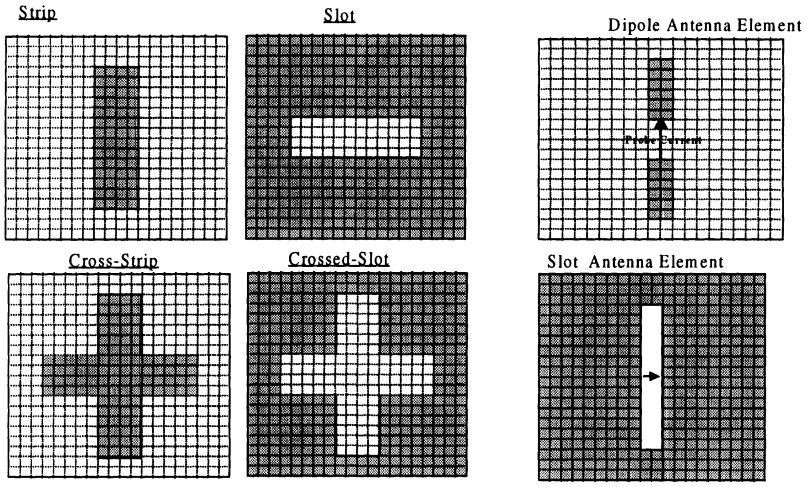


Figure 1. Illustration of the primitive elements used in FSS-BRICK. Note that the element must fit within the uniform grid.

Single Layer FSS : Slot Element

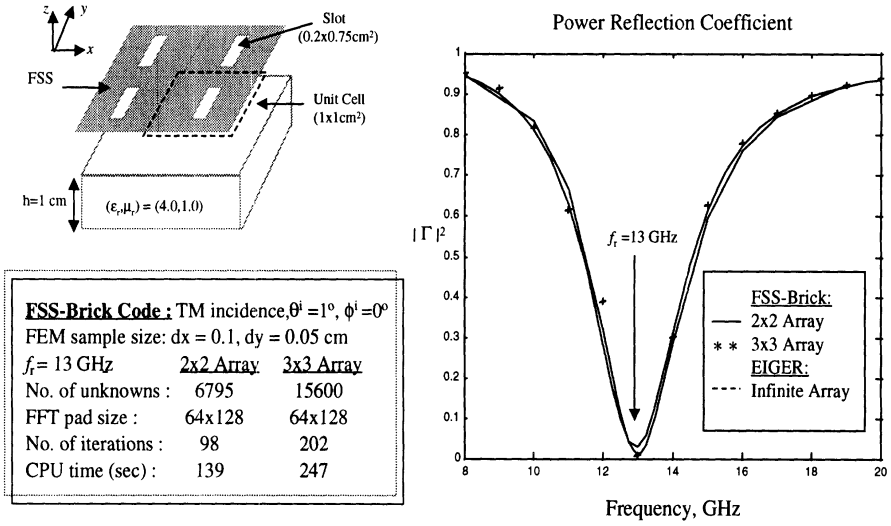


Figure 2. Comparison of reflection coefficient for a slot FSS as computed by the finite array code FSS-BRICK and the infinite array code FSS-EIGER.

2.2 Geometry Driver and Progress in the development of FSS-EIGER

As noted in earlier reports, FSS-EIGER is a generalization of the general purpose moment method code EIGER developed by the Univ. of Houston and Lawrence Livermore Labs. under Navy sponsorship. During the first quarter of this contract, a fast

periodic Green's function was developed which delivered two orders of magnitude in improved speed. This Green's function was generalized to multilayered structures during the second quarter of the contract and was ported into the original EIGER code to create FSS-EIGER. The remaining effort during the second quarter was devoted to testing and code validation and this continued into the third quarter. In our second quarter report we showed some validations for stand-alone multilayered dipole FSS (see Appendices 2 and 3). During this quarter

- we continued the validation of FSS-EIGER for slot arrays and for dipole antennas on multilayered FSS. Some results are shown in Figures 3 and 4. The results in Figure 4 refer to a dipole array on a dual layer dipole FSS and will be validated soon using FSS-BRICK and FSS-PRISM. It should be noted that these results refer to configurations that have not been considered in the literature due to their generality in terms of geometry and layered structure. Also note that the slot element in Figure 3 is identical to that used in Figure 2 for validating FSS-BRICK. The same slot element will be used later for validating FSS-PRISM.
- Developed a geometry Driver for the FSS-EIGER using dipoles, slots, crossed slots and crossed-dipoles as primitives. The description of this Driver will be given as an independent report from the Univ. of Houston.
- FSS-EIGER can model resistive cards and horizontal resistive loads which were already present in EIGER. However, testing of these capabilities has not yet been completed.

The emphasis over the next quarter will be devoted to the non-commensurate arrays and FSS.

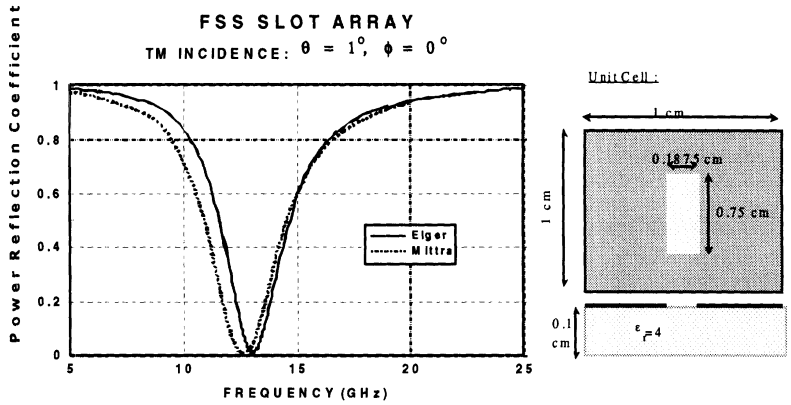


Figure 3. Validation of the FSS-EIGER for slot arrays.

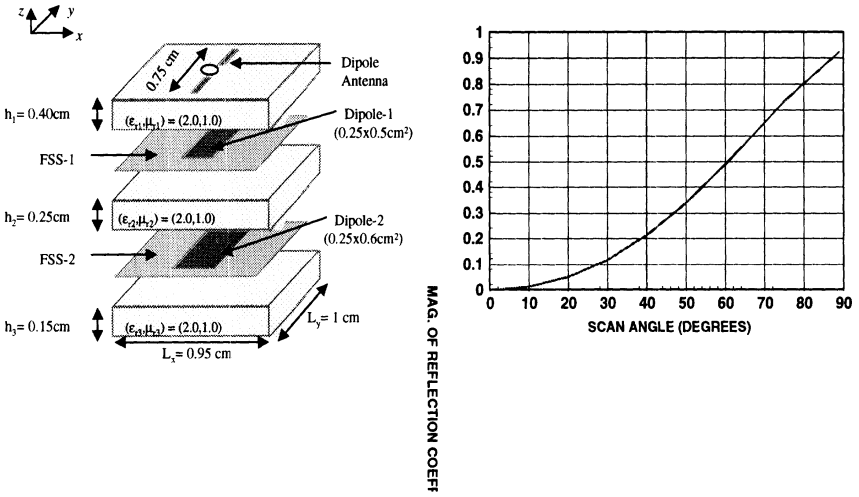


Figure 4. Broadside E-plane active reflection coefficient for a dipole array on a dual dipole FSS.

Aleko

2.3 Development and Validation of FSS-PRISM

FSS-PRISM refers to the hybrid FEM code. This code is the centerpiece of the proposed development and will combine the geometrical adaptability and generality of the FEM for modeling materials and inhomogeneities with the rigor of the moment method for mesh truncation. This code will therefore be the most general and capable among the all codes developed under this project. The code employs distorted prisms and will evolve to the doubly curved array code during the second year of this development.

FSS-PRISM took shape and form during this quarter and was developed by Dr. Eibert who joined the project team in March and will be the responsible for the development of the doubly curved array code during the second year.

The development of FSS-PRISM began with the existing code FEMA-PRISM which was a single element code for the analysis of conformal antennas on doubly curved surfaces. FEMA-PRISM was supported by Air Force contracts (Rome and Wright Laboratories) and was mentioned in the delivered proposal.

During the quarter the Dr. Eibert modified FEMA-PRISM as follows (see also attached report in section 3 and the viewgraphs in the Appendices):

- Incorporated periodic boundary conditions into the FEM domain to simulate the periodic nature of the FSS.
- Replaced the free space Green's function in FEMA-PRISM with the faster Ewald Periodic green's function developed by the Univ. of Houston.
- Validated the code for several trivial geometry for robustness
- Validated the code for slot and dipole arrays on commensurate FSS.

Figure 5 shows a validation of FSS-PRISM for a slot FSS. This figure displays results from all codes and it is pleasing to see that all codes are in excellent agreement. In our subsequent report we will demonstrate applications for dipoles and more complex configurations.

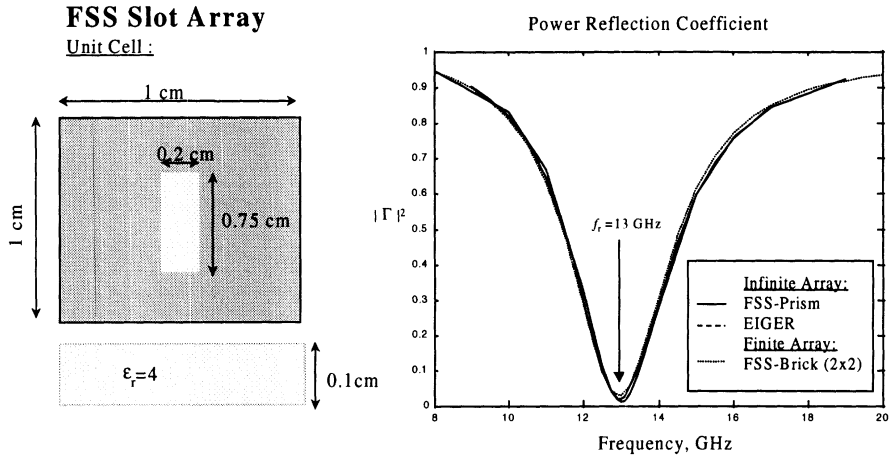


Figure 5. Comparison of FSS-PRISM, FSS-EIGER and FSS-BRICK for computing the reflection coefficient of a slot FSS on a dielectric layer.

In the following months, our goal will be

- generalize the current version of FSS-PRISM to non-commensurate arrays
- develop a geometry driver similar to that in FSS-BRICK
- compare FSS-PRISM and FSS-EIGER in terms of CPU and memory requirements for modeling multilayered FSS.

3.1 Theory and Description of FSS-PRISM for Commensurate FSS: Upgrading FEMA-PRISM to Deal with Infinite Periodic Planar Structures

1 Introduction

FEMA-PRISM–BI is a finite element (FE)/boundary element (BE)–hybrid code that works with prismatic elements in the volumetric FE–part and triangular elements in the BE–part. The prismatic elements can be right angled as well as distorted. So, the code vertically surface meshes at each layer with all geometrical adaptability whereas the volumetric FE-mesh can be easily generated by growing the mesh along the normals to the adjacent layer.
The original version of FEMA-PRISM was implemented as a pure FE–code with artifical absorbers used to terminate the interface to free space. Because of this, no Green's function was needed to truncate the mesh, and as a result doubly curved finite surfaces could be modeled.
During the first quarter of this project FEMA–PRISM was upgraded to a FE/BE–hybrid code with the BE–part implemented for planar surfaces based on the free space Green's function. In this stage FEMA–PRISM with BI can deal with planar cavity backed antenna configurations as illustrated in Fig. 1.
During the third quarter of this project FEMA-PRISM-BI was upgraded to a periodic FE/BE–hybrid code (PPRISM), capable of dealing with infinite periodic

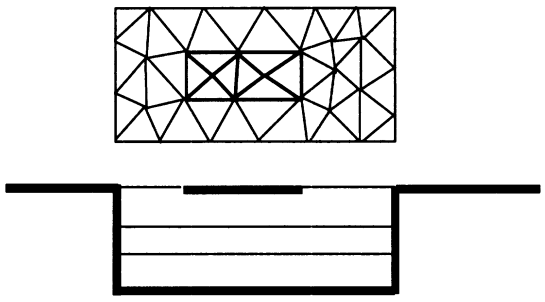


Figure 1: Cavity backed planar antenna configuration

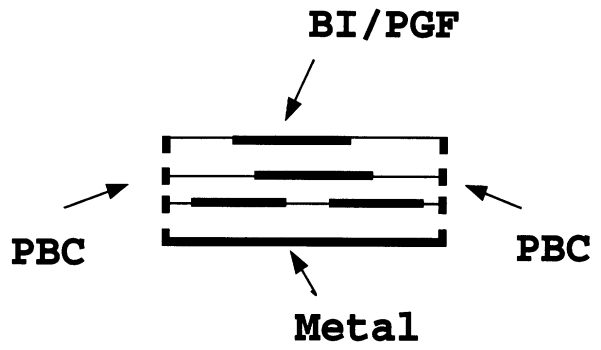


Figure 2: Metal backed periodic configuration

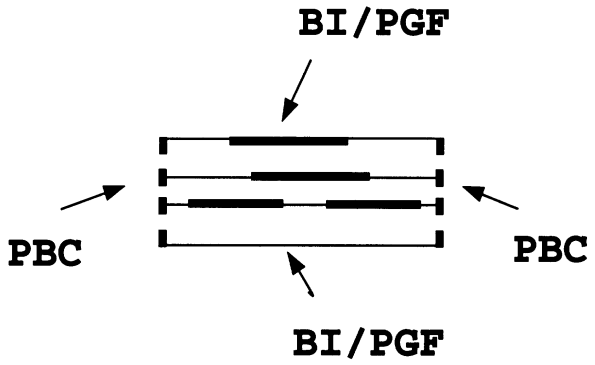


Figure 3: Open periodic configuration

planar antenna configurations as well as frequency selective surface (FSS) configurations with an arbitrary number of FSS layers (metallic patches or slots) and combinations of both. The code has the option to deal with metal backed configurations as illustrated in Fig. 2 and with configurations that are open at the top as well as the bottom surface of the FE–mesh (see Fig. 3). In this case the BE–method (or BI–method (boundary integral)) is used to terminate the top surface and the bottom surface. Due to the application of the BE–method on both surfaces the computational effort is larger and most often doubled.
In the following, it is shown how the periodicity condition for the fields in the infinite periodic array can be used to model the infinite array problem using only with one cell of the array. This means that within the FE–model of this unit cell, the periodic boundary condtion (PBC) has to be enforced on the vertical walls of the mesh and on the boundary edges of the BE surface. Also in the BE surface the appropriate periodic Green's function (PGF) must be used. For this application, the PGF developed by Prof. D. Wilton and Prof. D. Jackson will be used.
After a brief presentation of the formulation, numerical results are given to validate

the code.

2 Formulation of the Infinite Periodic Problem

The infinite periodic problem is illustrated in Fig. 4. For the electromagnetic

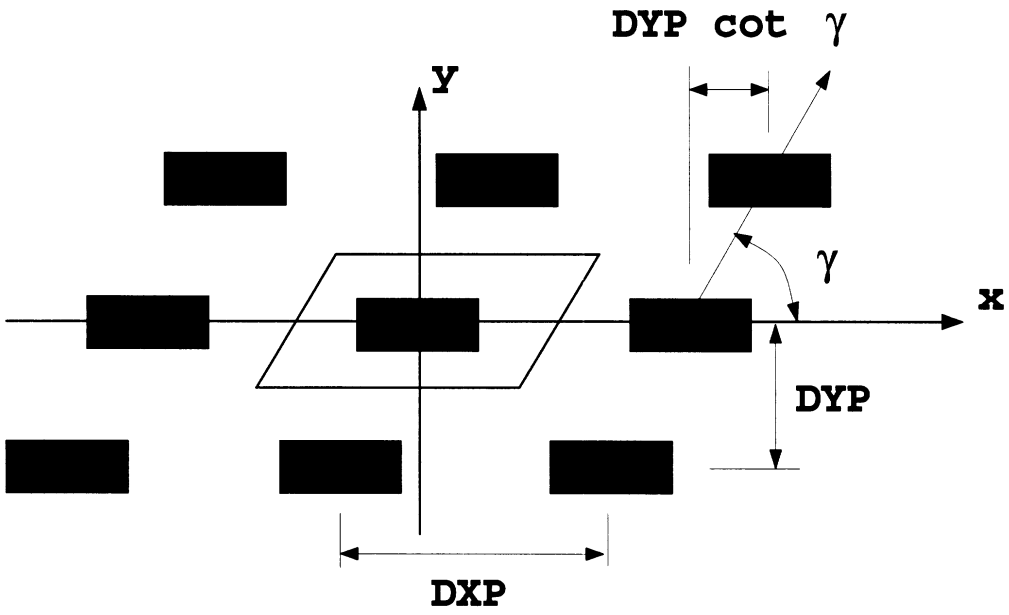


Figure 4: Infinite periodic configuration

analysis of this configuration, it is sufficient to consider one single cell (unit cell) of the infinite periodic array because the fields in the array fulfill a periodicity condition given by

$$\mathbf{E}(x + m\,DXP + n\,DYP\,\cot(\gamma),\, y + n\,DYP) = \mathbf{E}(x,y)\,\mathrm{e}^{-j\beta_x(m\,DXP + n\,DYP\,\cot(\gamma))}\,\mathrm{e}^{-j\beta_y n\,DYP}, \tag{1}$$

$$\mathbf{H}(x + m\,DXP + n\,DYP\,\cot(\gamma),\, y + n\,DYP) = \mathbf{H}(x,y)\,\mathrm{e}^{-j\beta_x(m\,DXP + n\,DYP\,\cot(\gamma))}\,\mathrm{e}^{-j\beta_y n\,DYP}. \tag{2}$$

DXP and DYP are the distances between adjacent cells in the array in x- and y-directions, respectively. Also, γ is the skewness angle of the lattice measured from the x-axis as illustrated in Fig. 4; β_x and β_y are given by

$$\beta_x = k_0 \sin\theta_0 \cos\phi_0 \tag{3}$$

$$\beta_y = k_0 \sin\theta_0 \sin\phi_0, \tag{4}$$

in which θ_0 and ϕ_0 are the angles describing the propagation direction of an incident plane wave or the scan direction of the array for antenna radiation, and k_0 is

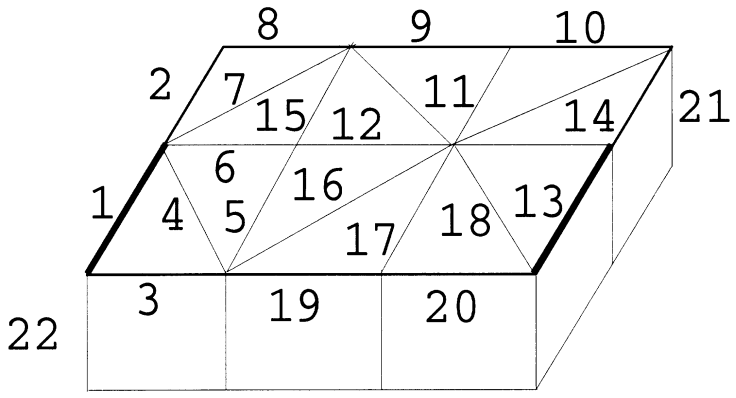


Figure 5: FE–mesh of unit cell

the free space wave number.
The periodicity condition (1) has to be enforced on the vertical walls of the FE–mesh and on the boundary edges of the surface meshes for the BE–method. Additionally, in the BE mesh the appropriate Green's function for the infinite periodic case must be employed. For the implementation of the periodic boundary condition into the FE volume let us consider the FE–mesh of a unit cell in Fig. 5. First, the FE–matrix is generated without considering the boundary condition at all, including all edges of the unit cell. Afterwards the right side wall is wrapped onto the left side wall and the upper side wall is wrapped onto the lower side wall via a matrix transformation. In this matrix transformation, all matrix entries involving unknowns associated with edges on the right and the upper vertical walls are eliminated and are replaced by new matrix entries involving the unknowns for the edges on the left and lower vertical walls. Consider for example edge #13 on the right boundary of the unit cell in Fig. 5. It can be seen that the corresponding periodic edge on the left boundary is edge #1. Therefore in all matrix entries the unknowns for edge #13 are replaced by

$$\begin{aligned} S_{i,13} &\longrightarrow S_{i,1}\, \mathrm{e}^{-j\beta_x DXP}, \\ S_{13,i} &\longrightarrow S_{1,i}\, \mathrm{e}^{j\beta_x DXP}, \end{aligned} \tag{5}$$

where i is an arbitrary edge number (the testing edge) that is also not on the right or the upper boundary wall (periphery) of the cell. In the latter case, the matrix entries have to be transformed with respect to both edge numbers. For instance the self–term for edge #13 is replaced as

$$S_{13,13} \longrightarrow S_{1,1}\,. \tag{6}$$

Edges that are located on the top vertical periphery of the cell have to be wrapped onto the corresponding edges on the lower periphery with the phase term ($\beta_x\, DY P\, \cot(\gamma)+$

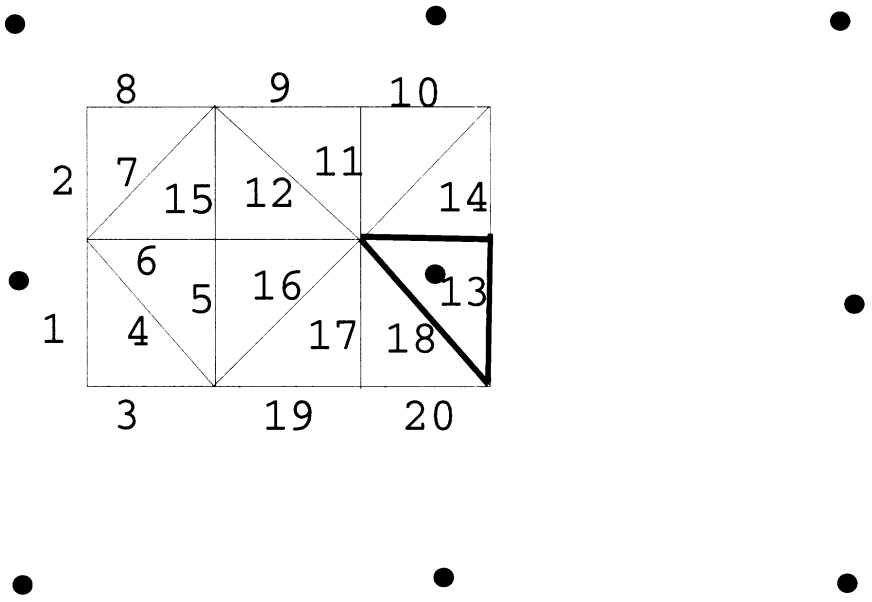


Figure 6: Configuration for BE–calculations

$\beta_y\, DYP$) and edges that are located at corners of the periodic cell (for instance edge #21) must be phase modified with respect to the x– and y–directions.
The strategy for the calculation of the BE–matrices is similar to that for the calculation of the FE–matrix. First the matrix is calculated for all edges of the unit cell no matter whether they are located on the boundary of the unit cell or not. Then the matrix entries for the edges on the right and upper periphery section of the boundary are wrapped onto the associated matrix entries for the corresponding edges on the left and lower portion of the periphery with the help of the appropriate phase terms (see Fig. 6). Additionally, for the calculation of the matrix elements, the Green's function for an infinite periodic array must be used. In the given case, matrix elements of the form

$$\begin{aligned} Z_{mn} &= -2k_0^2 \iint_{T_i} \iint_{T_j} G(\mathbf{r},\mathbf{r}')\,(\mathbf{W}_n \times \hat{z})\,ds' \cdot (\mathbf{W}_m \times \hat{z})\,ds \\ &\quad +2 \iint_{T_i} \iint_{T_j} G(\mathbf{r},\mathbf{r}')\,\nabla' \cdot (\mathbf{W}_n \times \hat{z})\,ds'\,\nabla \cdot (\mathbf{W}_m \times \hat{z})\,ds \end{aligned} \tag{7}$$

must be evaluated. As usual, $\mathbf{W}$ are the edge element expansion functions, T_i and T_j are the test and source triangles, respectively, and G is the periodic Green's function. In the original version of the PRISM–BI code, the free space Green's function is employed,

$$G_0 = \frac{1}{4\pi} \frac{\mathrm{e}^{-jk_0|\mathbf{r}-\mathbf{r}'|}}{|\mathbf{r}-\mathbf{r}'|}, \tag{8}$$

where $\mathbf{r}$ is the observation point and $\mathbf{r}'$ is the source point. In the general case these coupling integrals are evaluated with a numerical quadrature technique, but for

near–coupling and especially self–coupling terms, the integration of the singularity of the Green's functions must be carefully performed to obtain accurate results. In the infinite periodic case, the proper Green's function can be obtained as an infinite sum over terms of the form G_0 with the coordinates of the source points of each term adjusted to the locations in the different cells in the lattice and with considerations for the phase terms due to the scan direction of the array. For practical implementations, this space domain summation is not applicable because of its very slow convergence of the series.

For implementation into the FSS–PRISM code, the Green's function supplied by Prof. D. Wilton and Prof. D. Jackson, the University of Houston, was used. As noted in earlier reports, the EWALD acceleration scheme for the evaluation of doubly infinite series was employed. Using the EWALD summation scheme, one part of the infinite series is evaluated in the space domain whereas the other part is evaluated in the spectral domain. With a proper adjustment of the space and spectral domain parts of the series, a very fast convergence can be obtained. At the moment, the infinite array Green's function is available only for a lattice skew angle $\gamma = 90^o$ (right–angled lattice).

If both the source and the test triangles are close to be boundaries of the unit cell, it can be observed (see Fig. 6) that additional singular terms of the Green's function series can be situated close to the integration domain. These situations have to be recognized for the evaluation of the integrals in eq. 7 and the additional singularities have to be dealt with carefully.

3 Validation of the Code

3.1 Example 1 :

As a first validation example, the trivial configuration in Fig. 7 was considered. The air layer was discretized with finite elements and a boundary integral with the periodic Green's function was applied on the top and bottom layers. The structure was excited by a plane wave propagating in the z–direction, and from Fig. 8 it is seen that we recover the expected unity transmission coefficient.

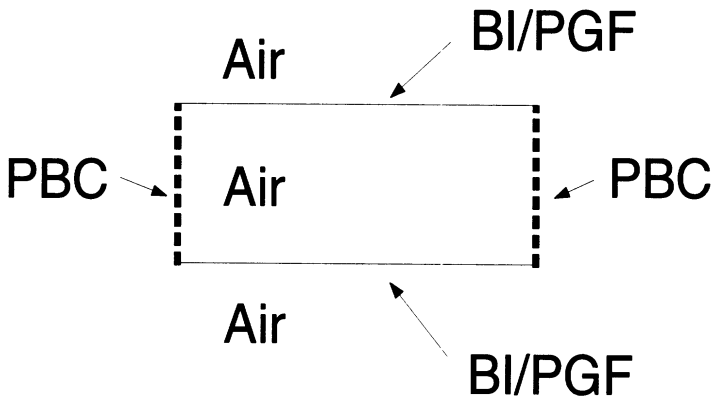


Figure 7: Infinite air layer

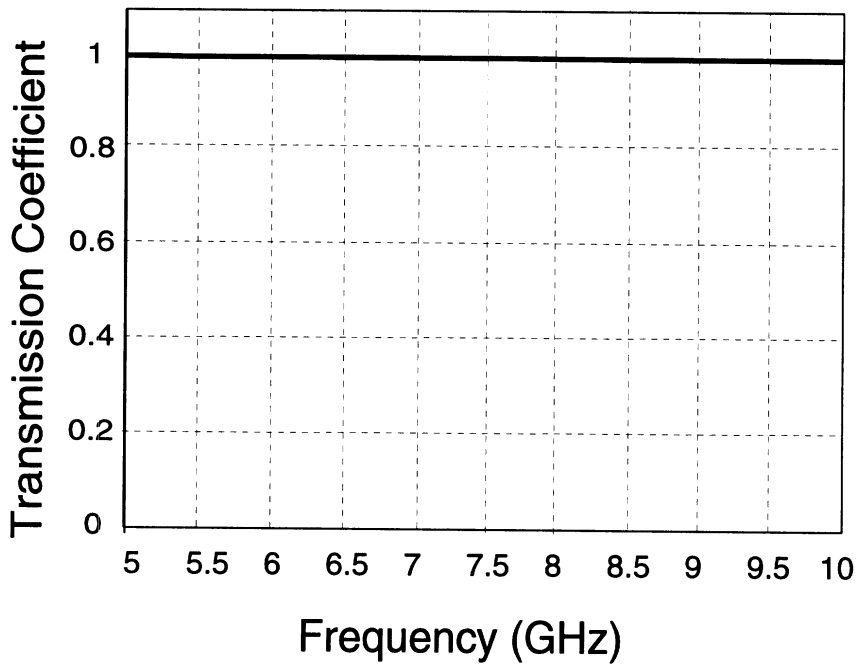


Figure 8: Power transmission coefficient for configuration in Fig. 7

3.2 Example 2 :

In the second validation example illustrated in Fig. 9, the air layer was replaced by a dielectric layer with a permittity of $\varepsilon_r = 5$. In this case, the reflection and transmission coefficients for plane waves can be calculated analytically by recursively applying the reflection and transmission coefficients at the layer interfaces. A comparision between the analytical and FSS–PRISM values for the transmission coefficients of a normally incident plane wave is given in Fig. 10.

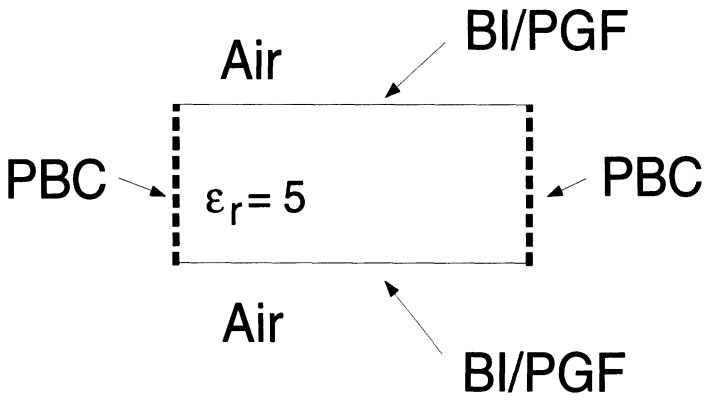


Figure 9: Infinite dielectric layer

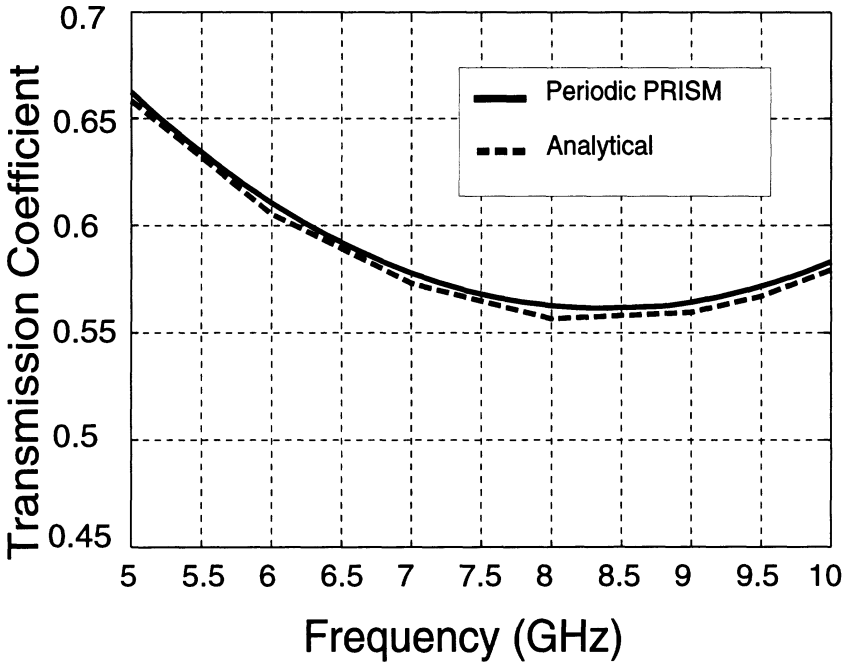


Figure 10: Power Tansmission coefficient for configuration in Fig. 9

3.3 Example 3 :

The geometry of the third example is given in Fig. 11. A rectangular periodic slot is backed by a dielectric layer with a permittivity of $\varepsilon_r = 4$. The configuration is excited by a plane wave propagating along the z–direction with the electric field vector oriented parallel to the long side of the slot. In Fig. 12 the numerical results of FSS–PRISM are compared to those generated by FSS–EIGER, and the results from the literature (Mittra et.al., Proc. IEEE, 1989).

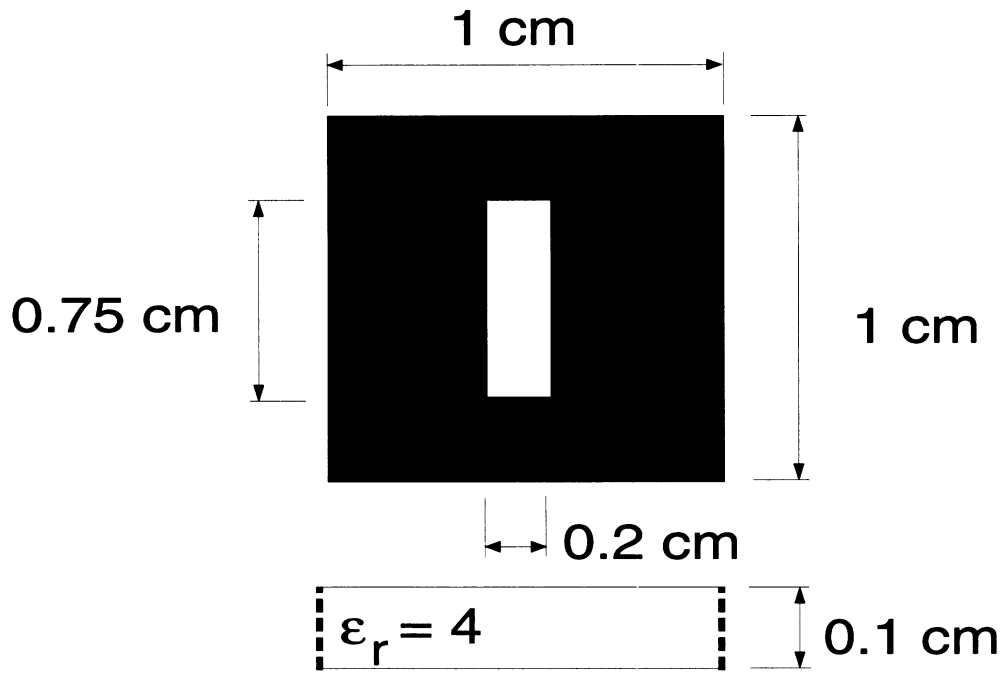


Figure 11: Slot–FSS on dielectric layer

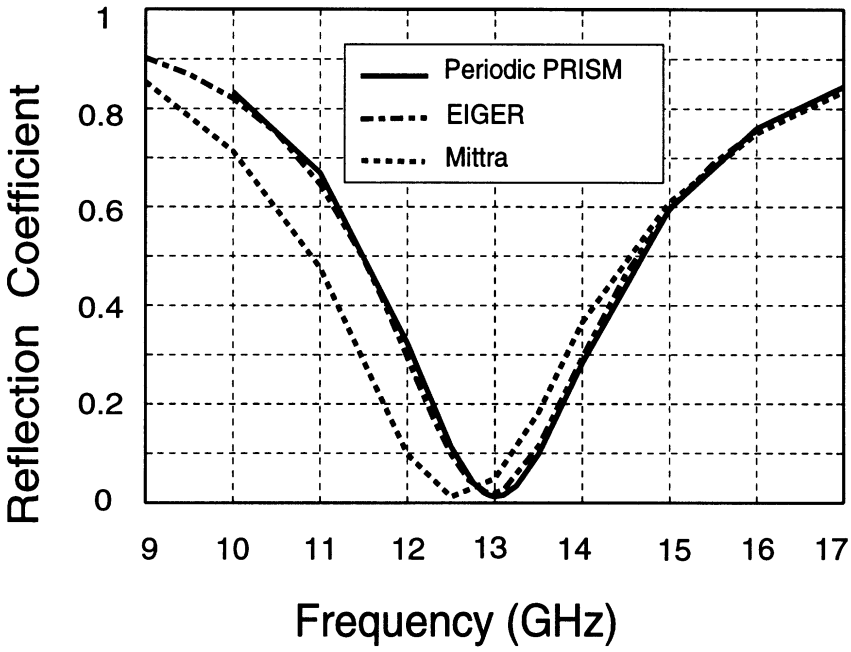


Figure 12: Power reflection coefficient for configuration in Fig. 11

APPENDICIES

APPENDIX 1: Project Goals

The goal of the SERAT project at the University of Michigan (with subcontract to Univ. Of Houston) is to develop a suite of software for the analysis of strip and slot dipoles on multilayered substrates backed by a frequency selective surface. The dipoles are equipped with photonic switches permitting variable electrical dipole lengths for broadband performance and the FSS is suitably designed to simulate a variable substrate thickness for optimal operation. A general view of the geometry is given in Figure 1.

The UM/UH team proposed to construct a code which combines various computational modules interfaced with appropriate pre-processors and post-processors. The computational modules include:

- Stand-alone moment method simulation of the FSS with up to 10 layers with commensurate and non-commensurate periodicities.
- Simple moment method simulation of the antenna elements on the FSS panels
- Hybrid FEM simulation modules for small arrays, planar periodic arrays and curved arrays on FSS panels.

Various options for modeling the FSS and for mesh truncation were proposed to provide a compromise between speed and accuracy. These are outlined in the proposal and summarized in the attached milestone chart (repeated from the proposal).

As called in the milestone chart, we are proceeding in accordance with the schedule in our proposal. Specifically:

1. The commensurate moment method FSS and moment method periodic array code was completed and tested by Houston and delivered to UM. This code is referred to as FSS-EIGER

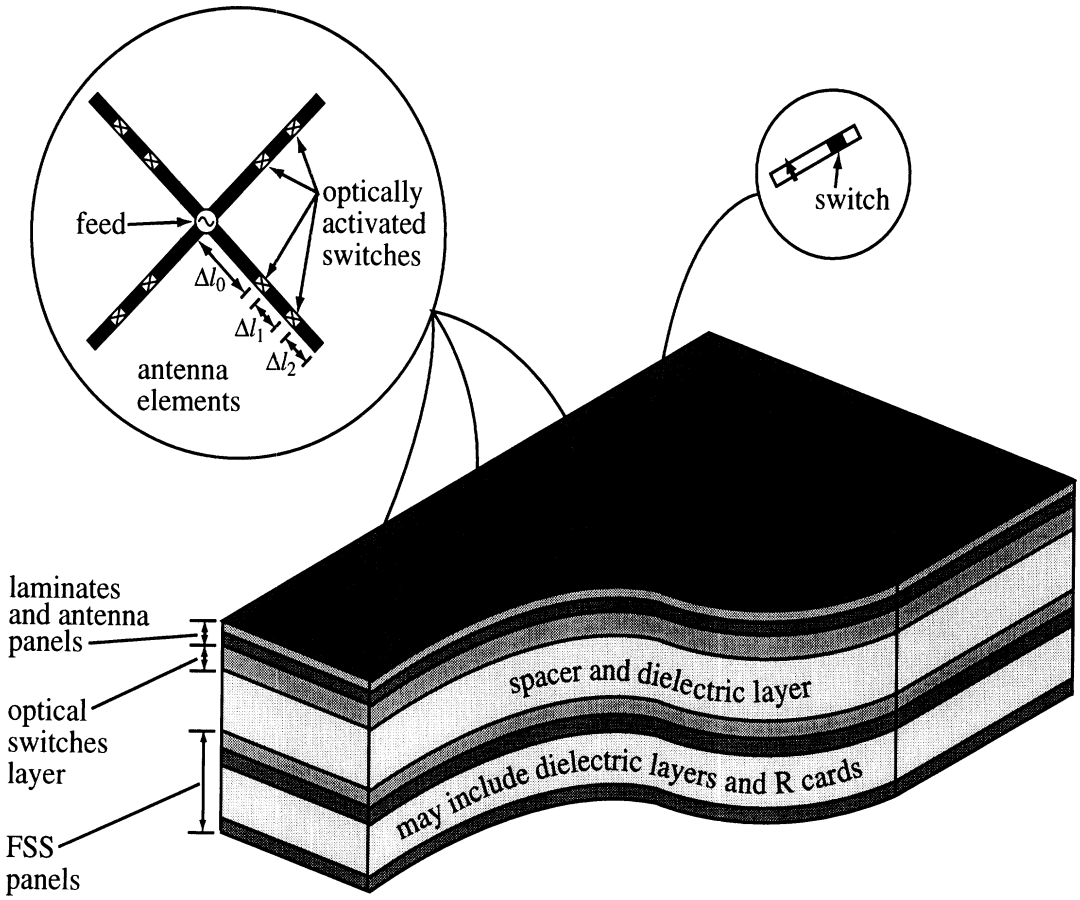
2. A geometry driver is also being developed for FSS-EIGER to handle the dipole, crossed-dipole, slot and crossed-slot elements.

3. UM completed ahead of schedule the small array FSS code and has tested it for slot and dipole array.

4. An automatic geometry generation Driver was developed for FSS-BRICK applicable to commensurate FSS panels

5. A first version of the centerpiece of this Code-Suite, the hybrid FEM code, is completed for commensurate arrays and combines the powerful FEM for material modeling with the robust boundary integral method. A Driver is now under development.

6. Several code validations have been given for slot/dipole periodic arrays on FSS structures.

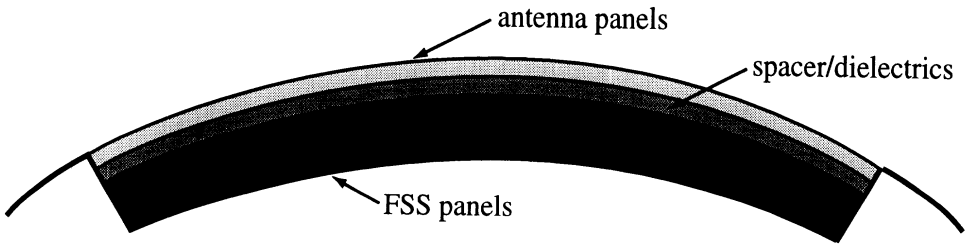


SERAT panel

single strip
on different
laminates

crossed
slots

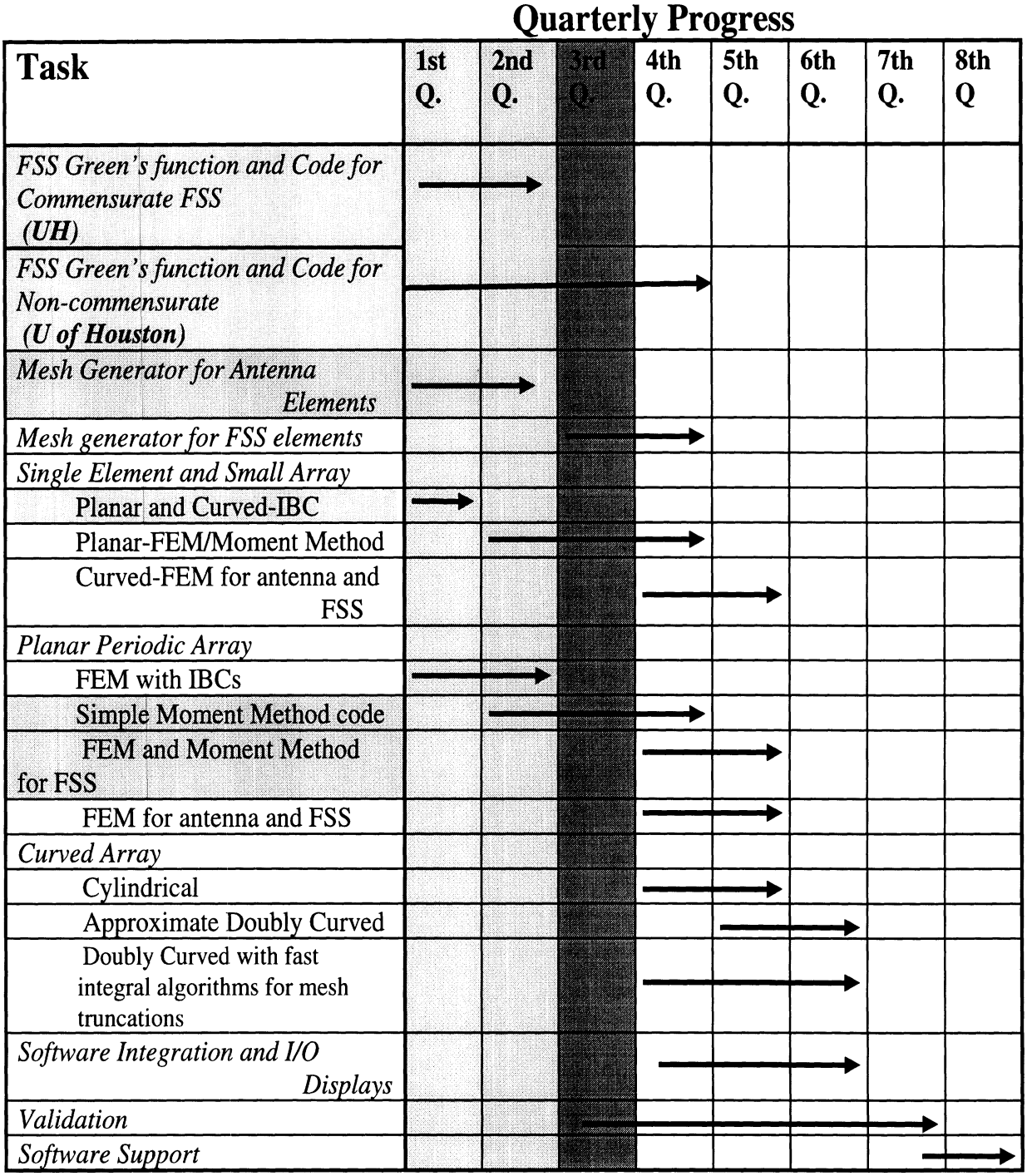
FSS elements



Curved SERAT

Figure 1: General SERAT configuration.

Milestone Chart for EM Model Development (Tasks 1 and 2)



APPENDIX 2: Presentation Given on June 25, 1997 at Nashua, N.H.

Hybrid FEM Software for SERAT Simulation

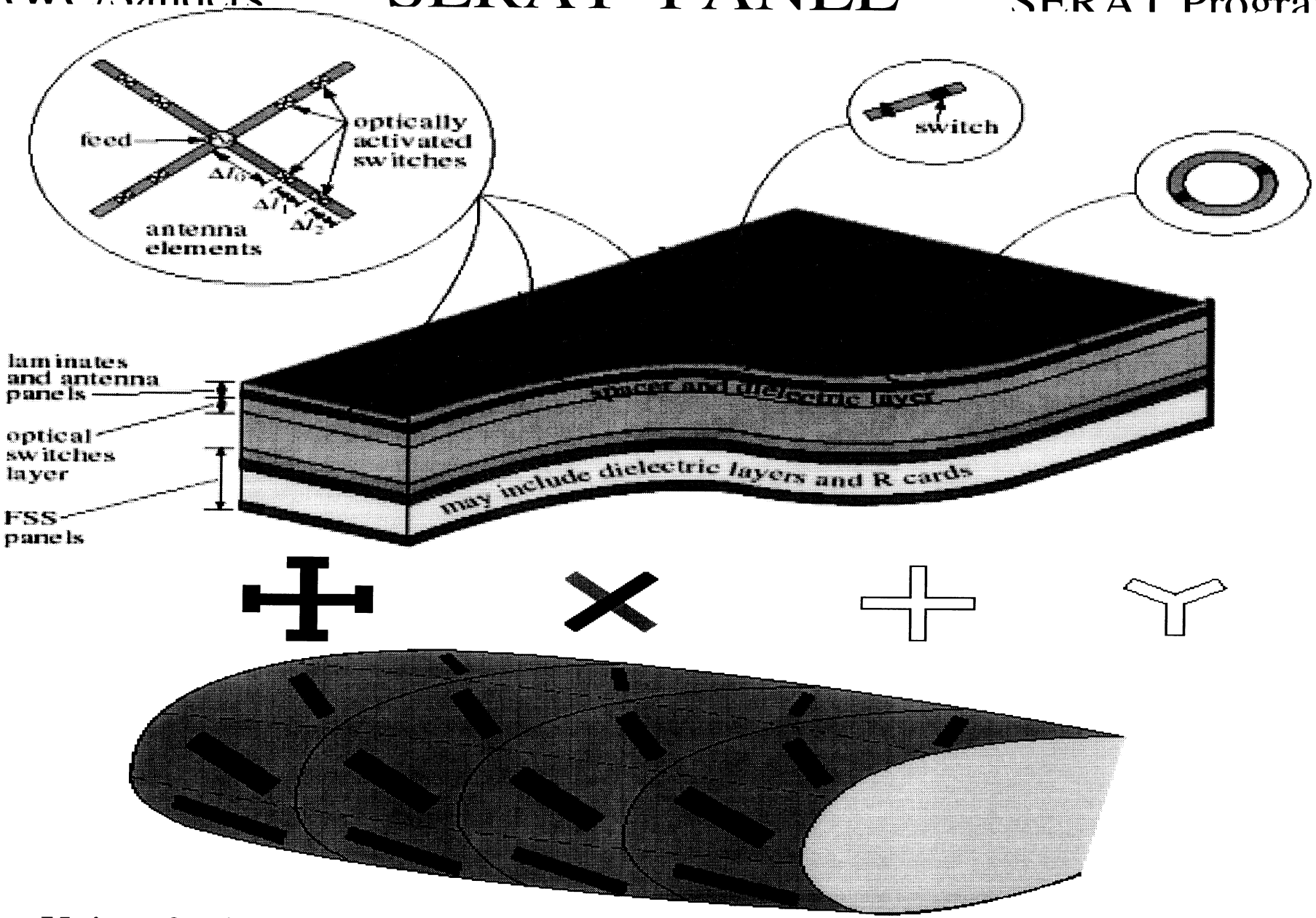
Project Team

T. Eibert, Y. Erdemli and J. L. Volakis
Radiation Laboratory
Dept. of Electrical Engin. and Computer Science
University of Michigan
Ann Arbor, MI 48109-2122

D.R. Wilton and D.R. Jackson
Dept. of Electrical and Computer Engineering
University of Houston

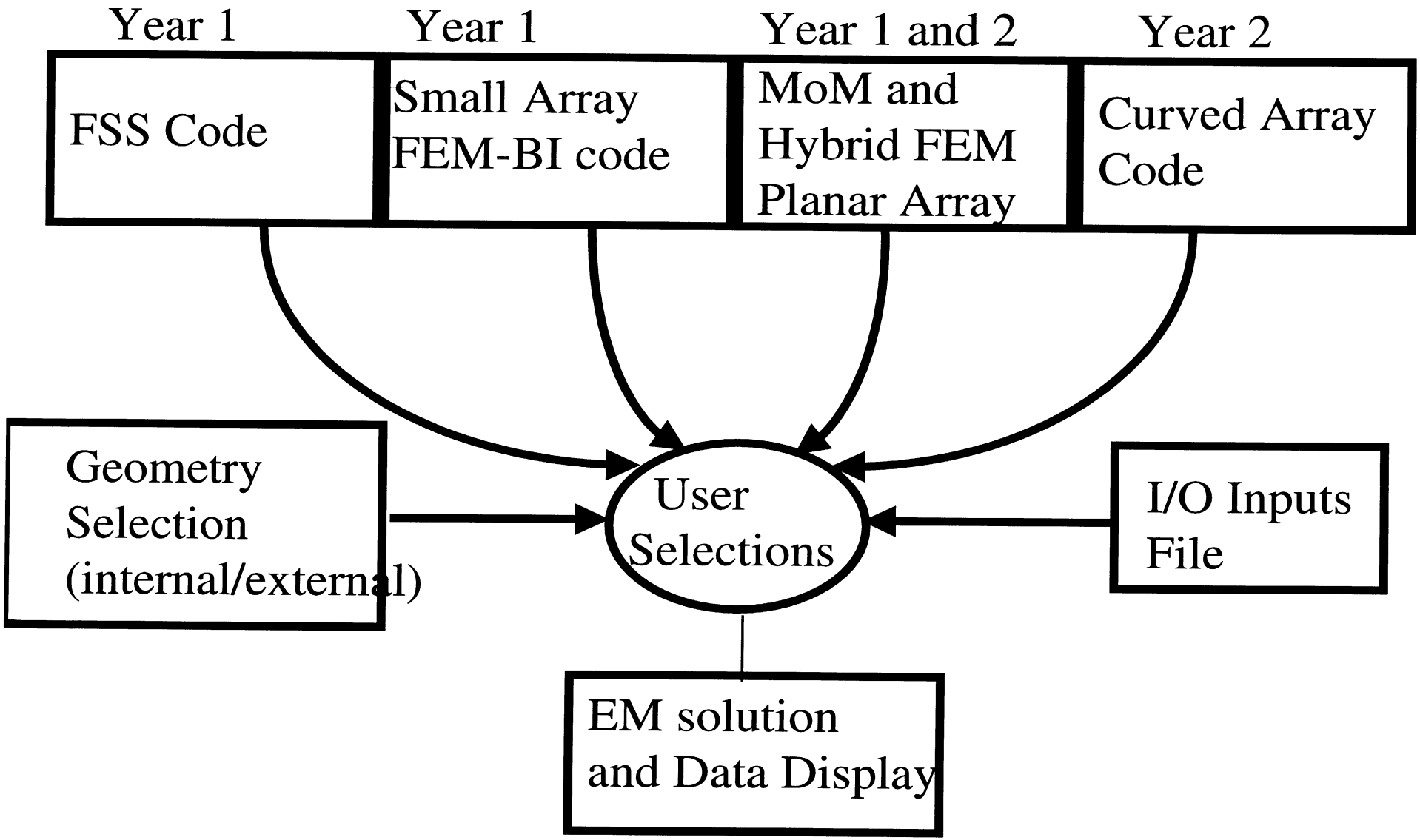
Univ. of Michigan/Univ. of Houston SERAT Code Development Team

SERAT PANEL


feed
optically activated switches
antenna elements
switch
laminates and antenna panels
optical switches layer
FSS panels
spacer and dielectric layer
may include dielectric layers and R cards

Overall Project Goals

- **Develop a Suite of Codes**
  - **Combining Moment Methods, Finite Element and Hybrid Formulations for SERAT Modeling**
- **Univ of Michigan**
  - **Develop Hybrid FEM Software and Integrate Univ. of Houston Codes into a single SERAT Simulation Package Addressing Geometrical Adaptability, Speed and Accuracy.**
- **Univ of Houston**
  - **Develop Moment Method FSS Simulators and Modules for the Hybrid FEM Software Package**


Year 1
Year 1
Year 1 and 2
Year 2
FSS Code
Small Array FEM-BI code
MoM and Hybrid FEM Planar Array
Curved Array Code
Geometry Selection (internal/external)
User Selections
I/O Inputs File
EM solution and Data Display

First Year Goals(Summary)-Planar

- Develop accelerated (simple) Moment Method FSS codes
- Integrate Moment Method FSS code with antenna elements
- Develop small array Hybrid FEM code as a testbed for the Hybrid FSS code
- Integrate FSS Green's function with FEM for a general purpose planar SERAT array hybrid code

Second Year Goals(Summary)-Curved

- Generalize hybrid FEM codes for SERAT arrays on doubly curved platforms
- Introduce fast algorithms and new solvers to speed-up computations for planar and curved SERAT arrays

Code Development Status

- **FSS_BRICK (UM) Small array code**
  - Capable of large finite array analysis using fast BI solvers
  - Fast but more restricted geometry due to brick modeling
  - Will be upgraded to periodic arrays
- **FSS_EIGER (UH) Simple moment code**
  - Upgraded from EIGER to handle arrays on FSS
  - Includes fast evaluation/Ewald of periodic Green's function
  - Validated for commensurate slot and dipole arrays on FSS
  - Delivered to UM and is being tested
- **FSS_PRISM (UM) Planar SERAT hybrid FEM code**
  - Our most capable code--combines FEM and MoM capabilities
  - Implemented with Periodic Boundary Conditions and Eiger free space Green's function.
  - Can handle FSS and antenna elements at any layer location.
  - Validated with FSS_EIGER for slot and dipole elements
  - Slower than FSS_BRICK but full geometrical adaptability

- **FSS-BRICK (Small Array Code)**

(original UM version of the Air-Force XBRICK)

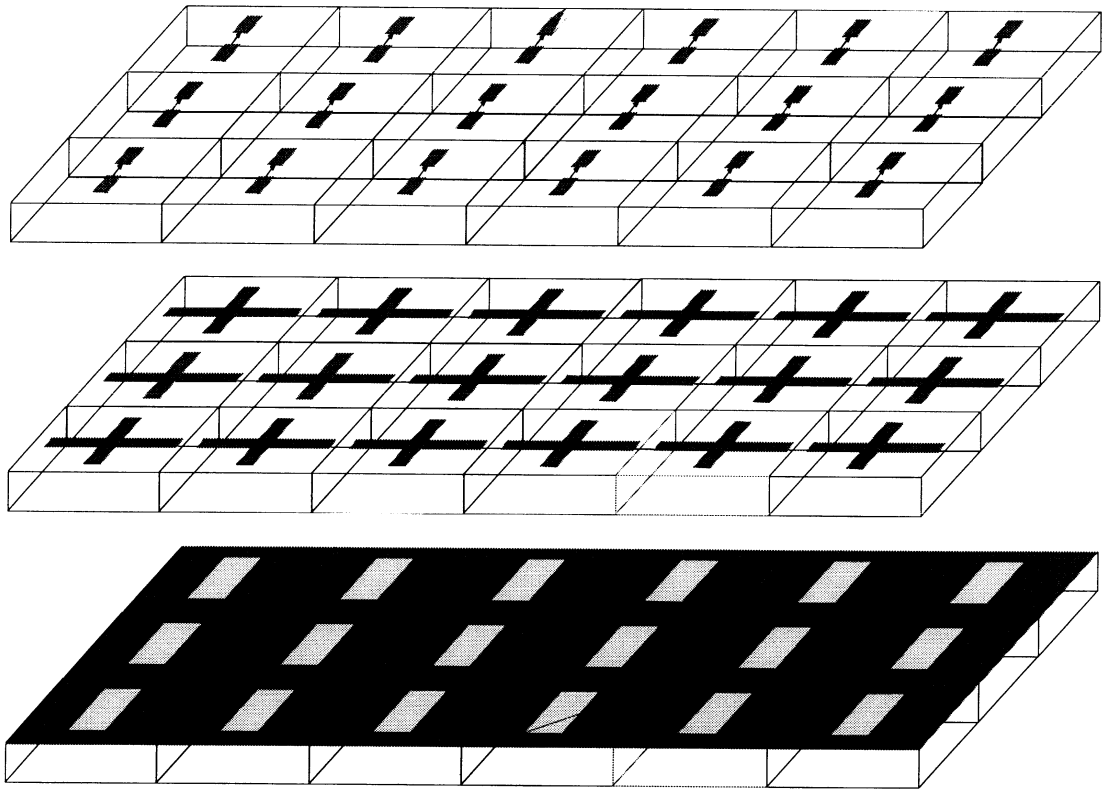
  - **Rectangular brick elements for modeling SERAT volume**
  - **Includes capabilities for lumped loads, vertical wires, resistive sheets and feeds of arbitrary orientation**
  - **Boundary integral for mesh truncation and the FFT for the fastest known matrix compression--THIS IS THE GIVES A MAJOR SPEED ADVANTAGE FOR BRICK**
  - **BRICK was demonstrated for slot and strip FSS elements, dipoles on multilayered FSS.**
  - **Geometry Driver was developed for the SERAT commensurate panels (antenna and multilayered FSS)**

This code was scheduled for completion end of first year.

It is ready, except for various upgrades in Geometry interface, lumped elements, more validation.

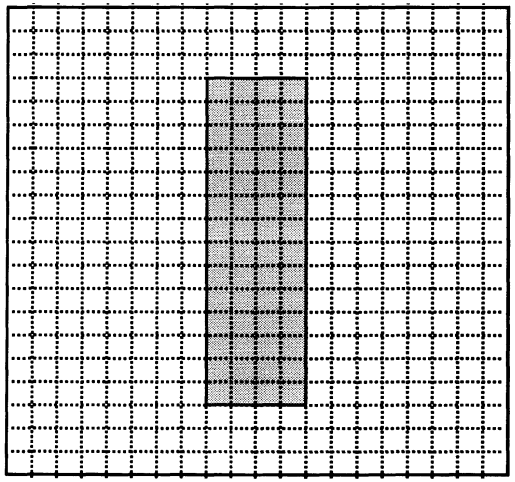
Goal is to extend BRICK to a Periodic Code and compare to the Periodic PRISM in terms of speed and capability.

FSS-BRICK GEOMETRY DRIVER

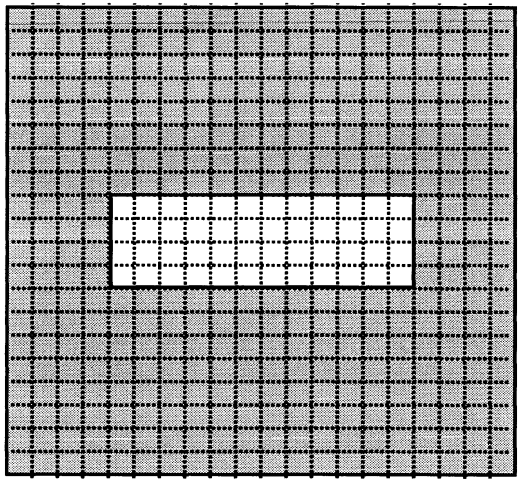


FSS Element Types Available in BRICK Driver

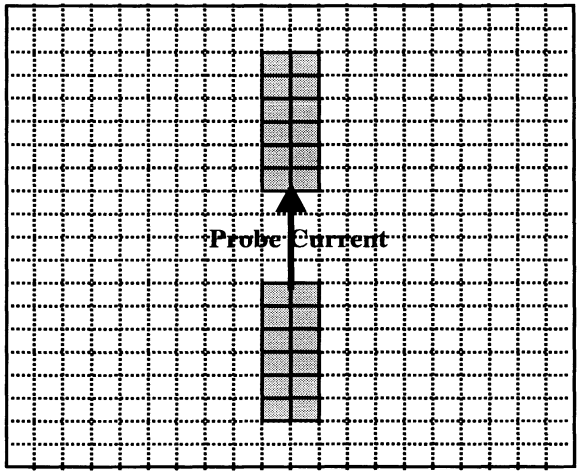
Strip



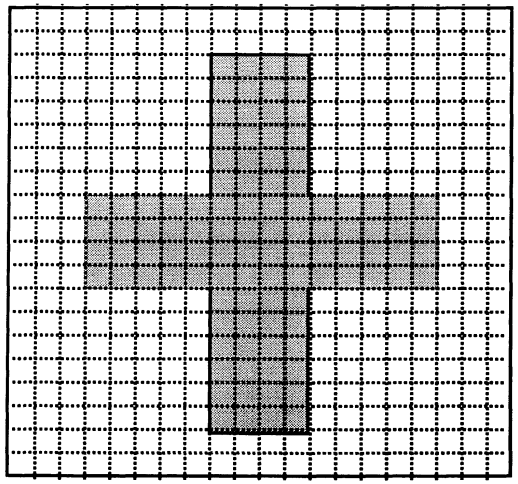
Slot



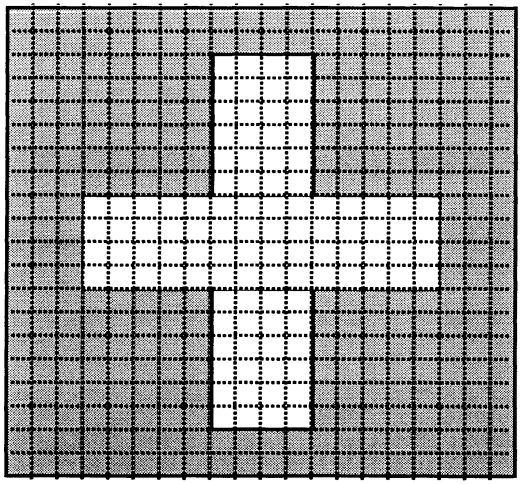
Dipole Antenna Element



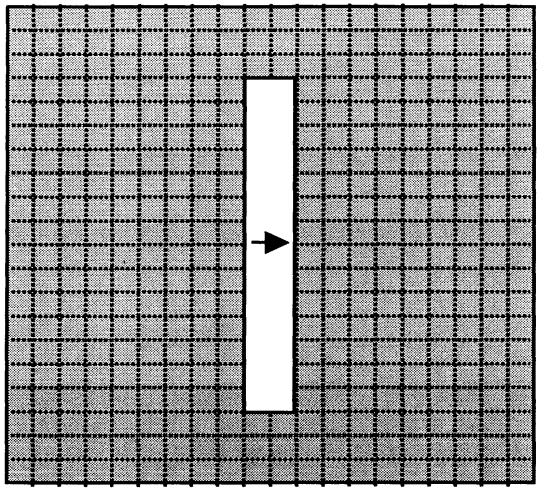
Cross-Strip



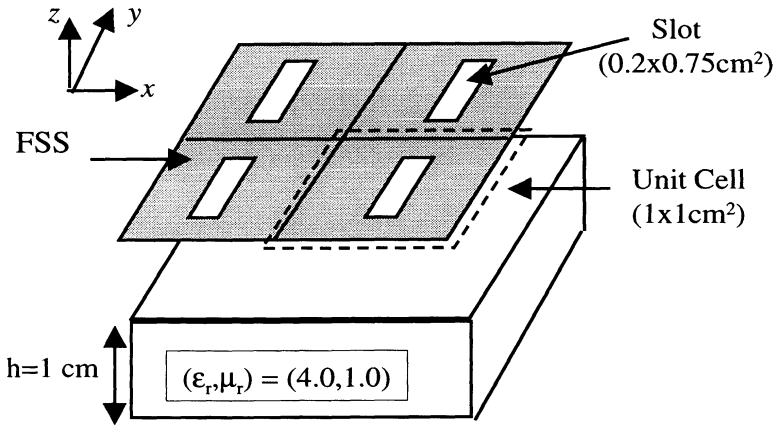
Crossed-Slot



Slot Antenna Element

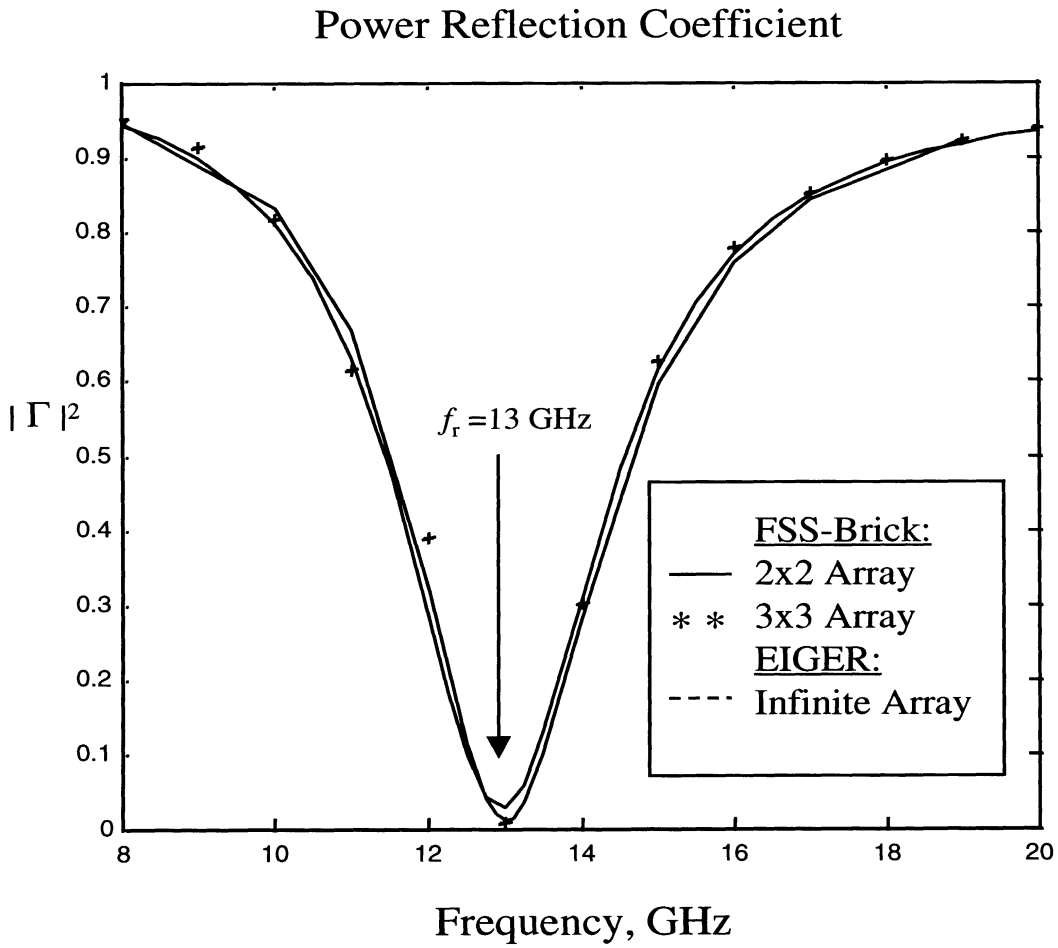


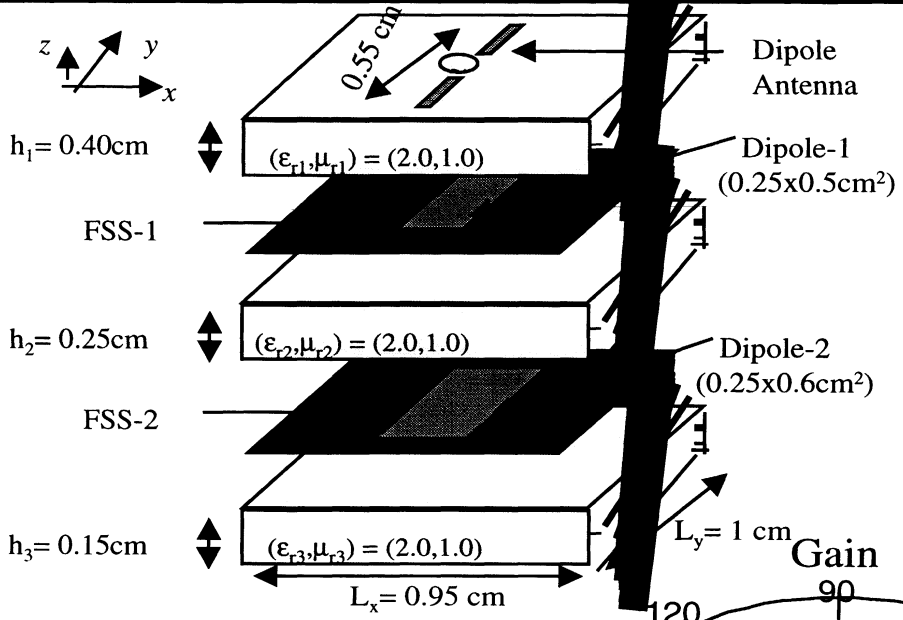
Single Layer FSS : Slot Element



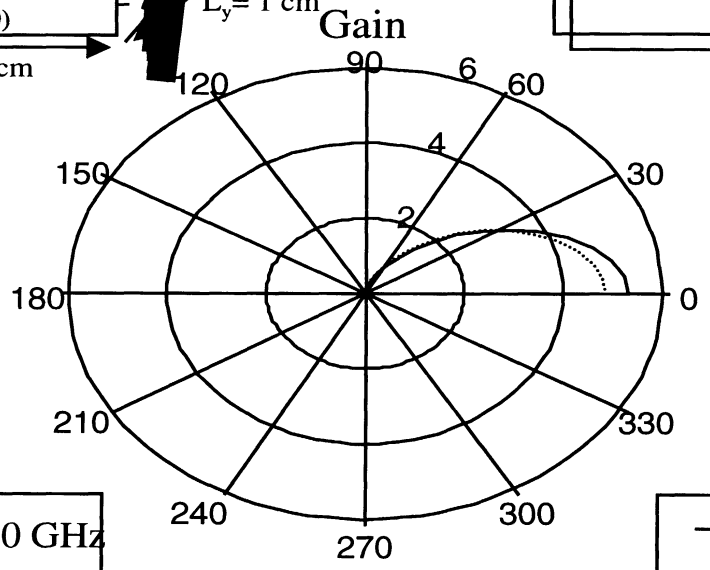
FSS-Brick Code : TM incidence, $\theta^i = 1^o$, $\phi^i = 0^o$
FEM sample size: dx = 0.1, dy = 0.05 cm

f_r = 13 GHz	2x2 Array	3x3 Array
No. of unknowns :	6795	15600
FFT pad size :	64x128	64x128
No. of iterations :	98	202
CPU time (sec) :	139	247





θ = 0-90°, ϕ = 0°
f_o = 17.25 GHz
FEM sample size = 0.05 cm
FSS with 1-cell :
No. of unknowns = 16702
FFT pad size = 64x64
No. of iterations = 323
CPU time = 333.25 sec



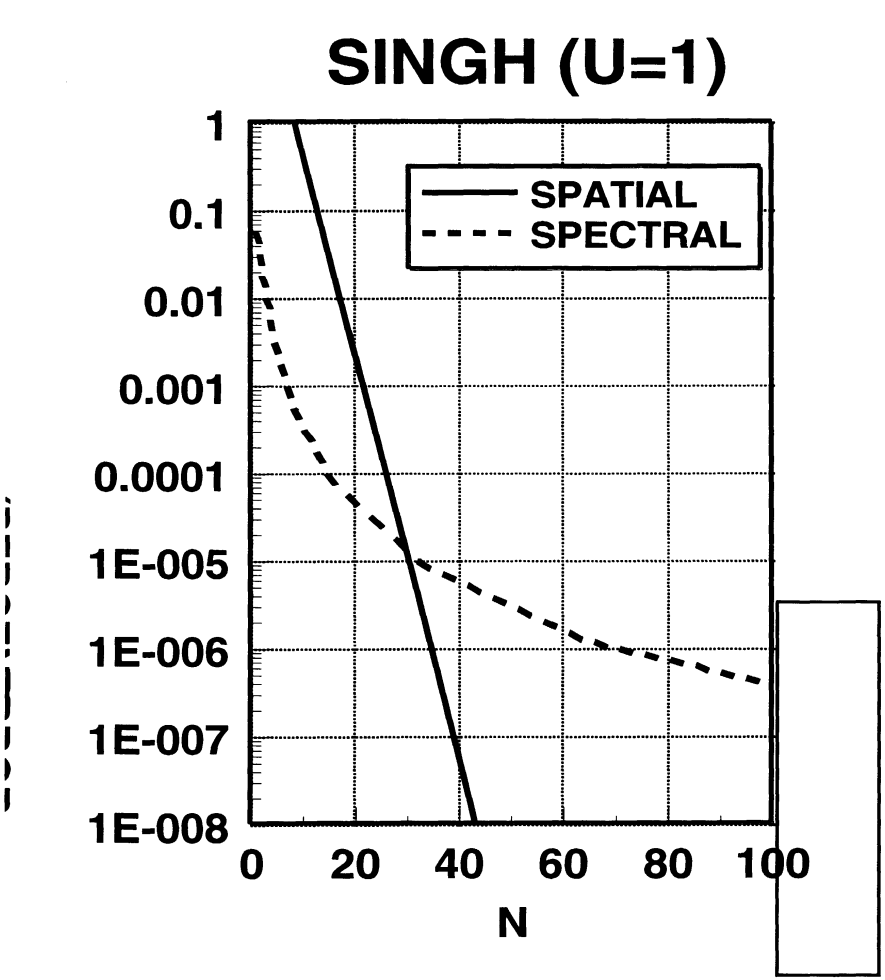
..... f_o= 13.50 GHz
Z_{in}= 22.6-j26.3 ohm

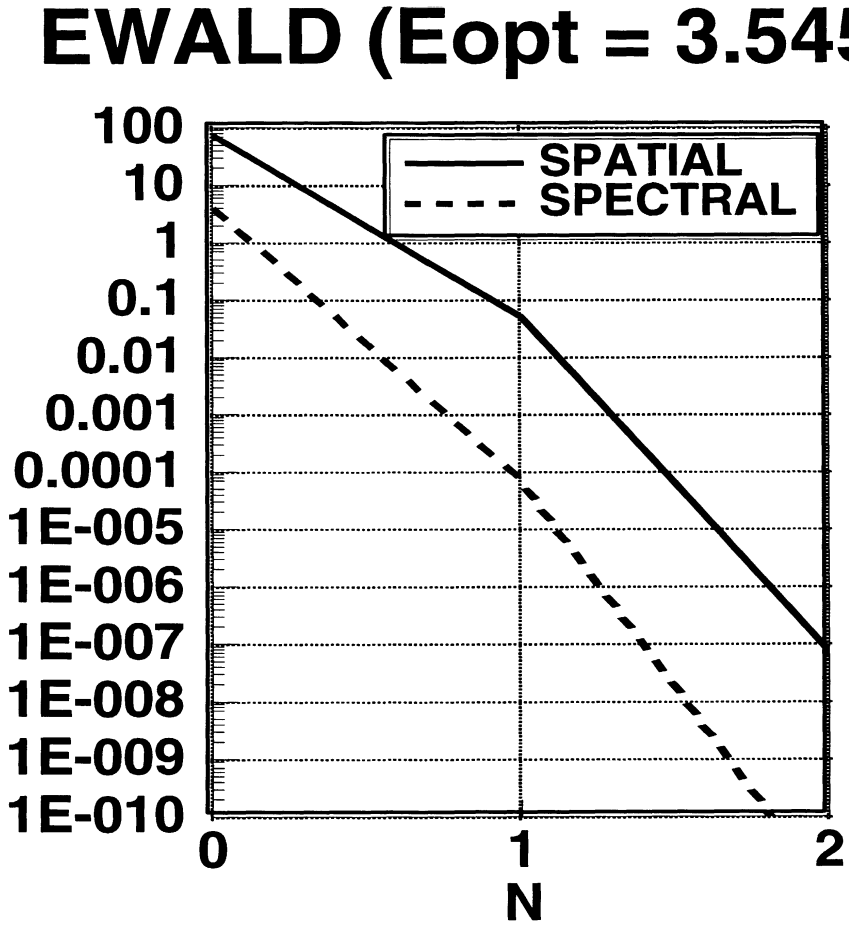
— f_o= 17.25 GHz
Z_{in}= 118.8+j23.2 ohm

- **FSS Green's function and Code**
  - **Efforts during the first quarter concentrated on accelarating the periodic Green's function using Ewald's method**
  - **Achieved substantial speed-ups using the Ewald method as demonstrated in the 1st Quarterly report**
  - **Speed-up is usually 2 orders of magnitude**
  - **Validated Green's function by comparison with printed dipole and slot FSS**
  - **Ewald free space periodic Green's function was delivered to UM for integration with Periodic-PRISM code.**

This task was to be completed by the end of first quarter and was done on schedule

COMPARISON OF ACCELERATION SCHEMES: SINGH AND EWALD METHODS





COMPARISON OF CALCULATION TIMES

define: $T = \dfrac{\text{Ewald calculation time}}{\text{Singh calculation time}} = \dfrac{2\times\left(2\left(N_{Ewald}+1\right)\right)^2\times 10}{2\times\left(2\left(N_{Singh}+1\right)\right)^2}$

(Assumes evaluation of erfc(z) is ten times more expensive than evaluation of the free-space Green's function.)

sample calculations:

- Assume a required error of 0.1% : $T = 1.11$
- Assume a required error of 10^{-7} %: $T = 0.0028$

• Simple MoM code Using New FSS Green's function

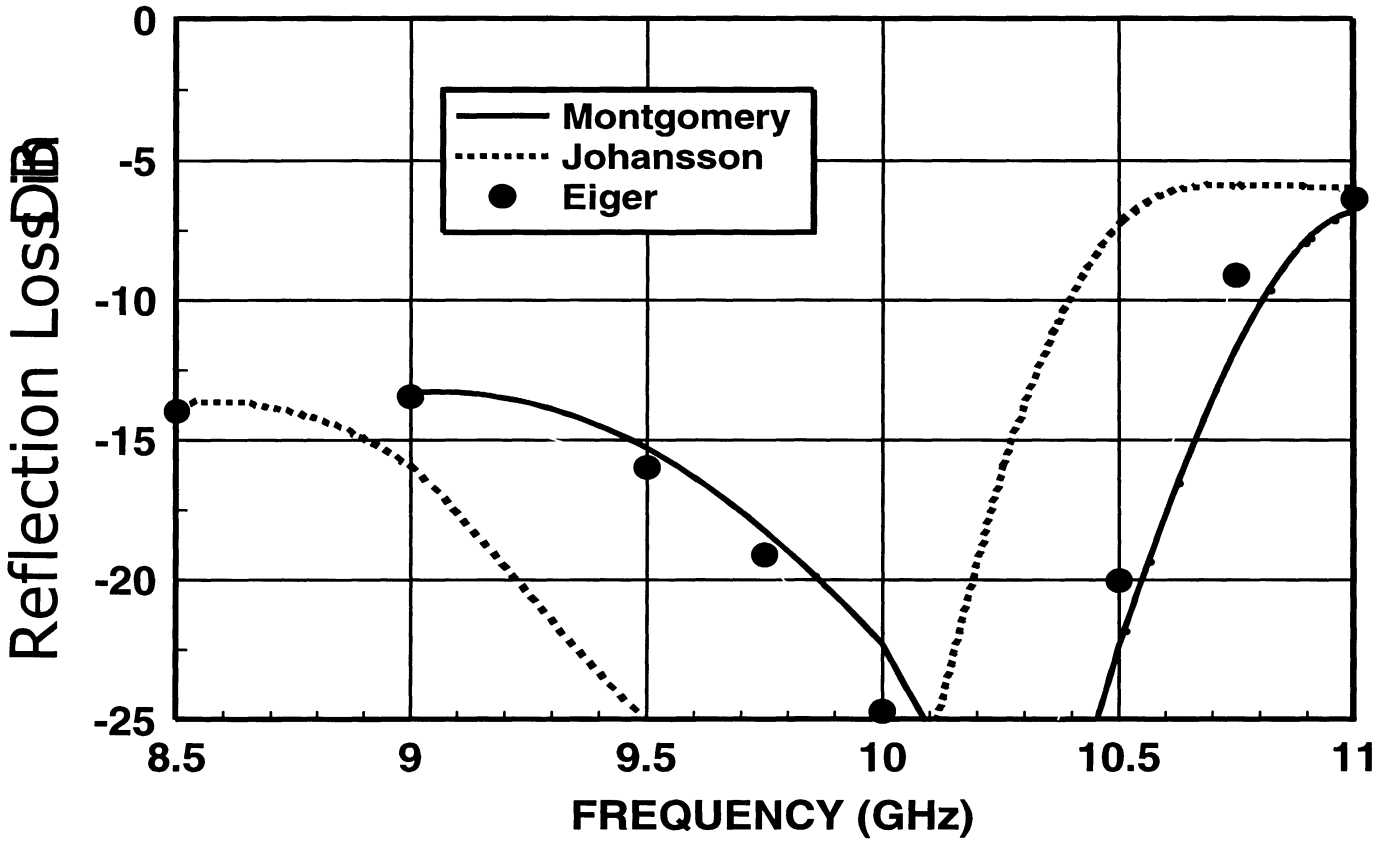
- Began with EIGER, a general-purpose code framework that allows flexible subdomain modeling with a wide variety of element types (triangles, rectangles, wires, bricks, etc.).
- Developed and imported Periodic Green's Function with EWALD acceleration.
- Added pre-processor (user interface) to generate SERAT geometries (dipoles, slots, crosses) and excitation data.
- Added post-processor to calculate figures of merit for SERAT anlaysis and design (reflection, transmission coeffs., impedance).
- Models any number of FSS and dielectric layers with lumped or distributed loads, arbitrary element shapes, linear or higher-order modeling, combined metallics/dielectrics.

Commensurate FSS/MoM code was completed end of 2nd quarter as scheduled

Non-commensurate FSS/MoM code and Drivers are in Progress

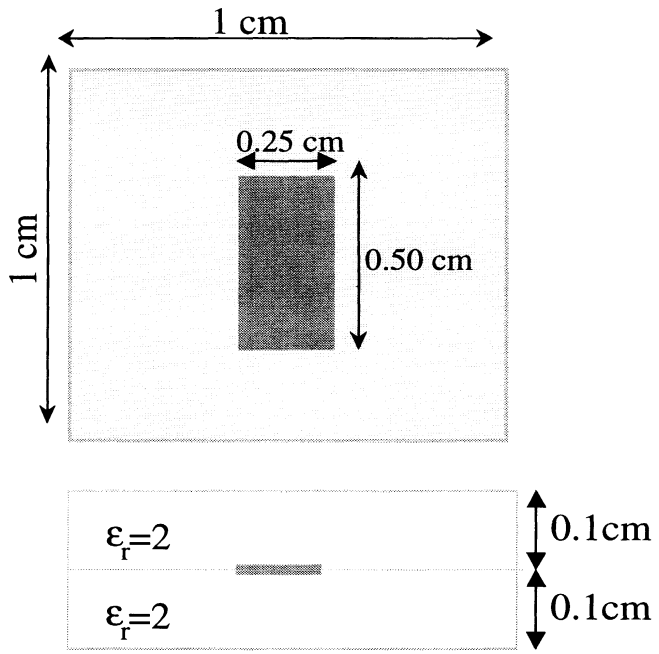
THREE-LAYER FSS IN FREE SPACE

REFLECTION LOSS OF (0,0) MODE

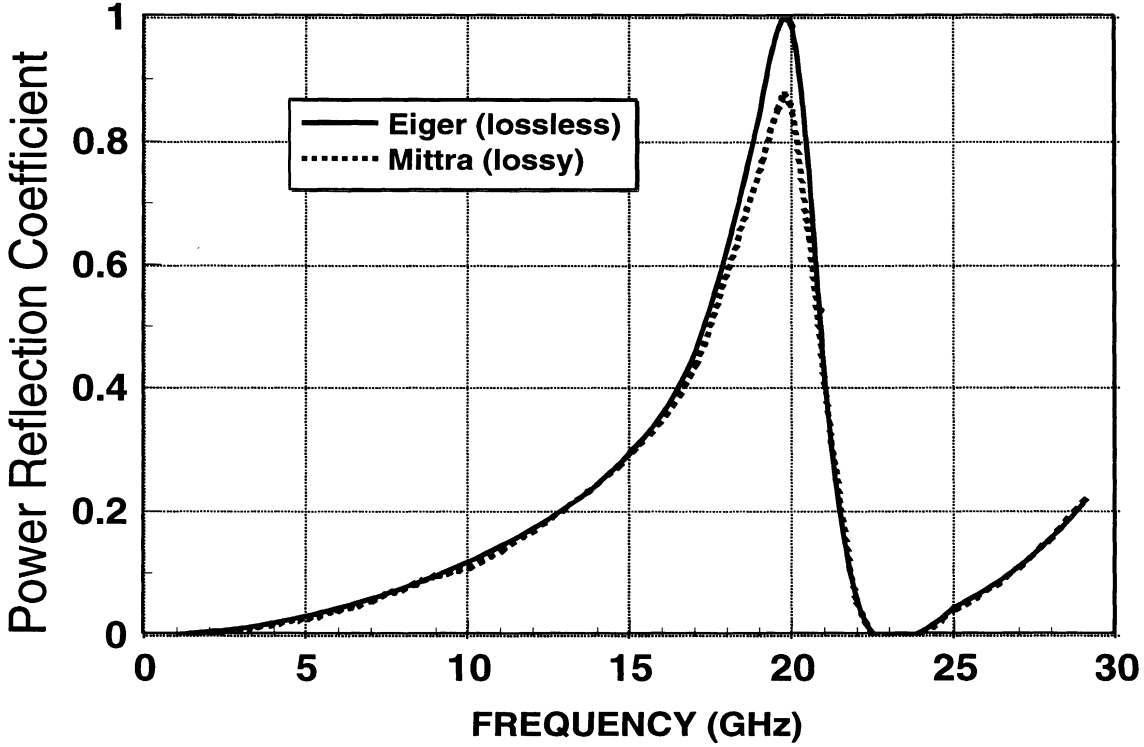


FSS EMBEDDED IN DIELECTRIC LAYER

Unit Cell :

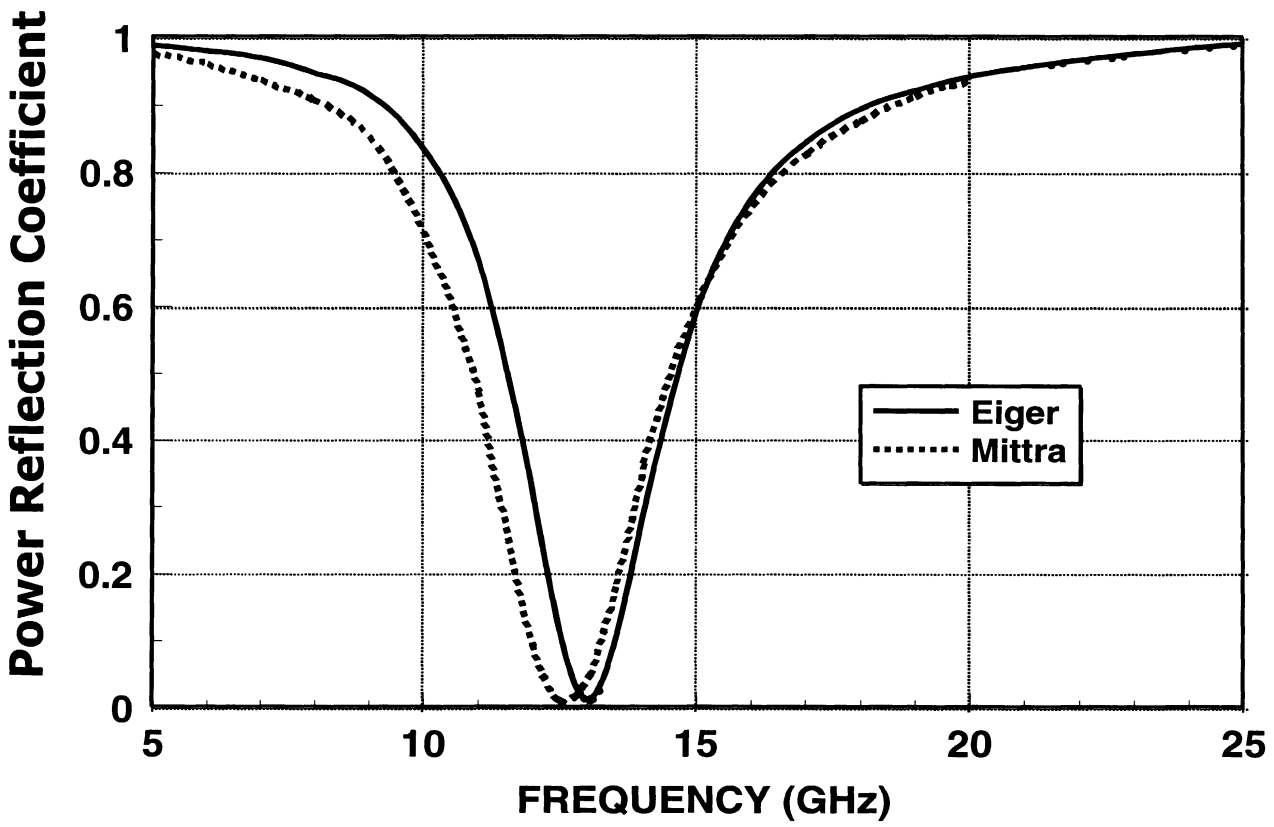


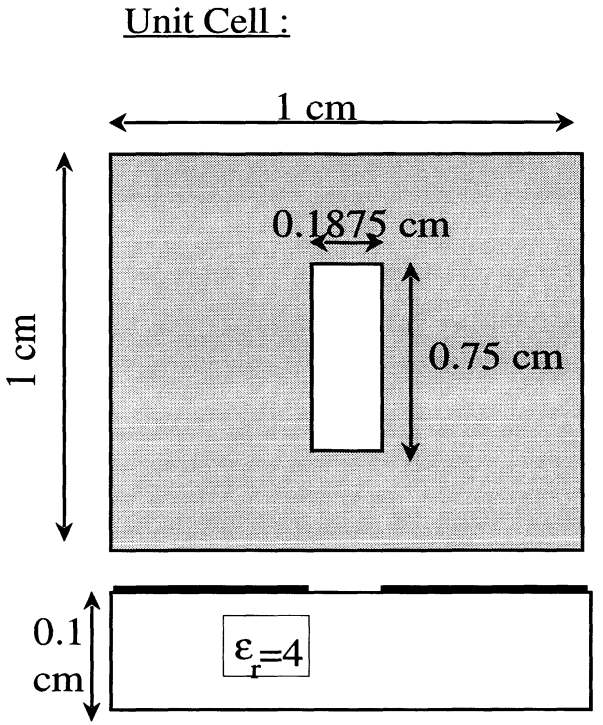
TE INCIDENCE: $\theta = 1^{\circ}$, $\phi = 90^{\circ}$



FSS SLOT ARRAY

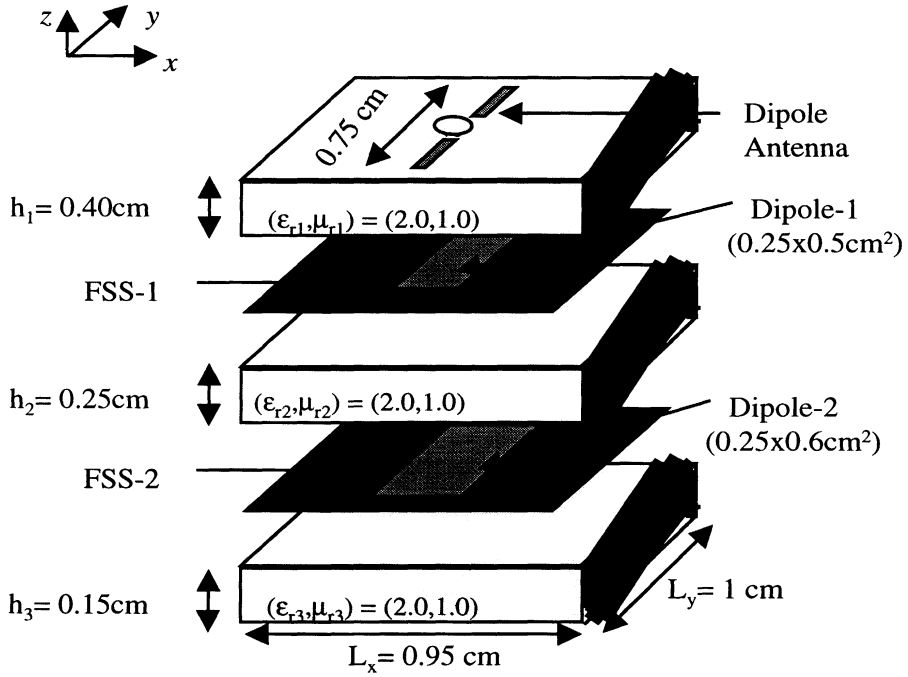
TM INCIDENCE: $\theta = 1,^{\circ}\ \phi = 0^{\circ}$

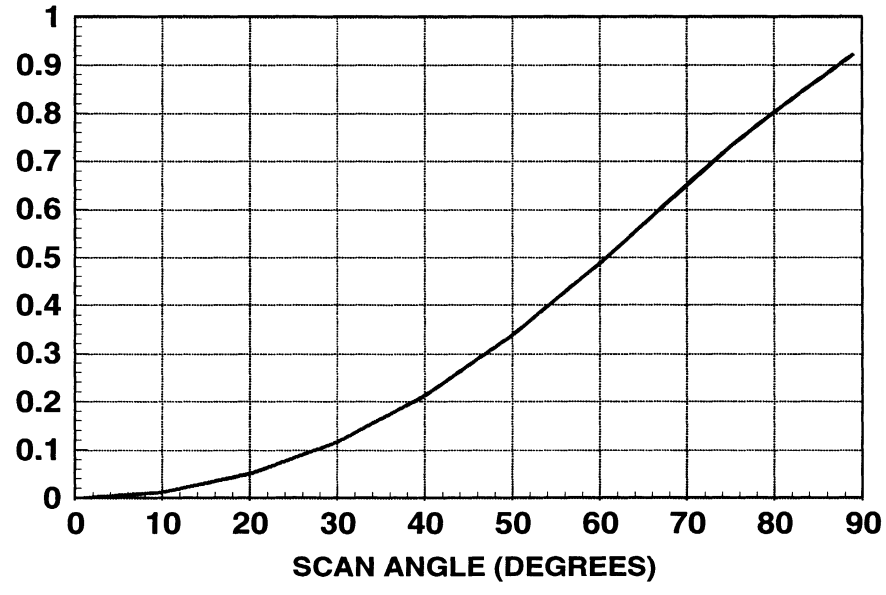




BROADSIDE-MATCHED ACTIVE REFLECTION COEFFICIENT

E-PLANE SCAN



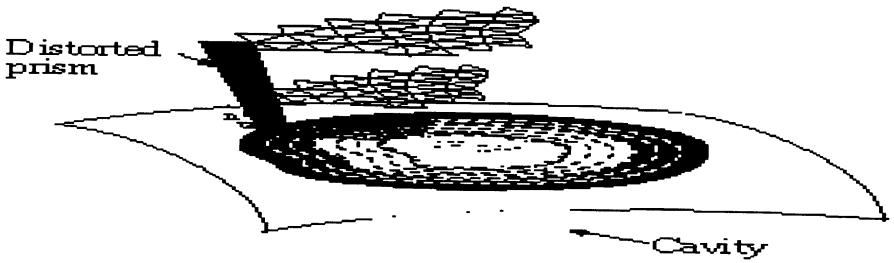


Original Version of PRISM as of October

FEMA-PRISM ATTRIBUTES

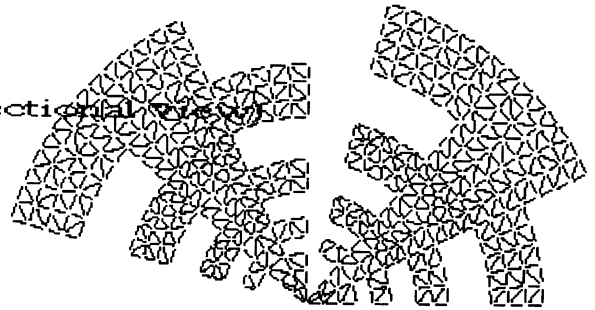
FEMA-PRISM was Developed under Air Force & Navy Sponsorship

Mesh Generation



(Perspective view)

(Cross-sectional view)



Features

Analyzes antennas on doubly curved platforms

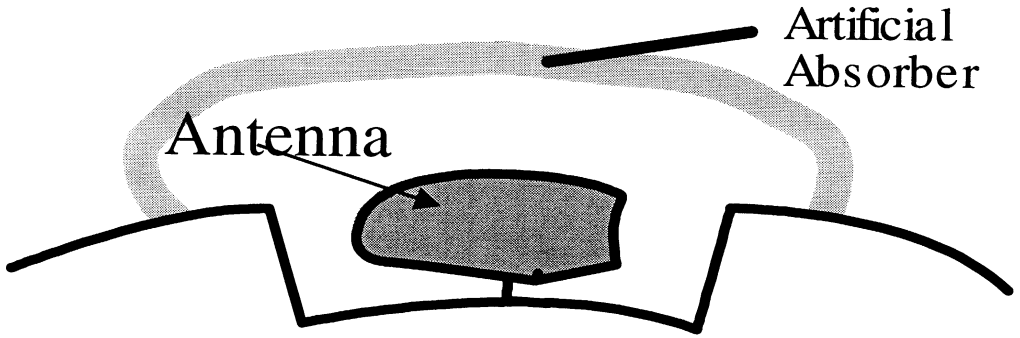
Anisotropic material coatings/substrates are accomodated

Grows volume mesh using distorted prisms. Given the surface mesh, volume mesh grown along surface normals

Built-in surface mesh generator for circular and rectangular patch antennas

Mesh turncated using isotropic/anisotropic artificial absorbers

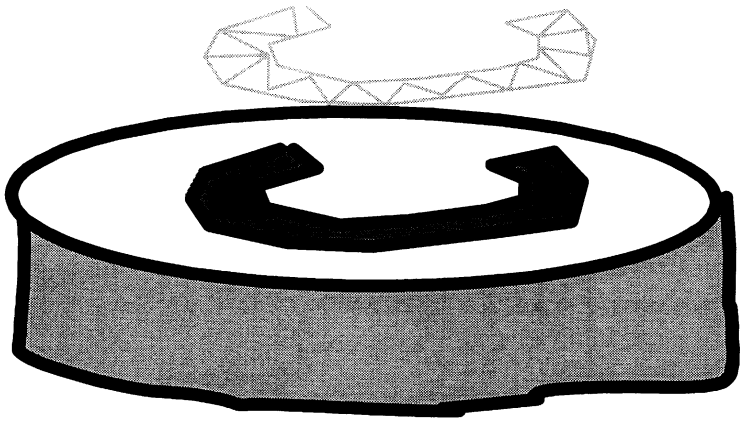
- **FEMA-PRISM was upgraded during 1st quarter for accurate boundary integral truncation and future interface with FSS Green's function**



Sparse Matrix

$$[A]\{E\}=\{f^{inc}\}$$

Boundary Integral Truncation



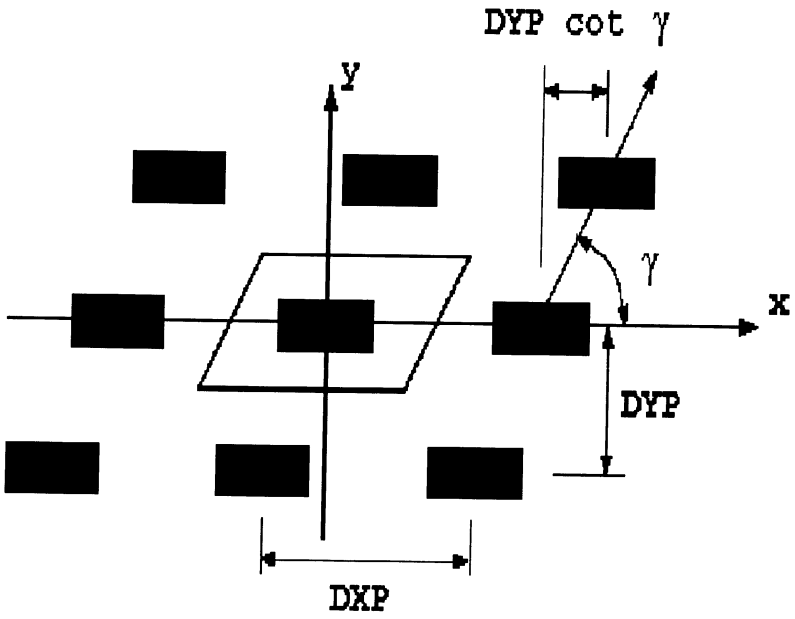
Dense

$$\begin{bmatrix} A^{\text{int}} & A^{cross} \\ A^{Tcross} & A^{bound} \end{bmatrix}\begin{Bmatrix} E^{\text{int}} \\ E^{bound} \end{Bmatrix}+\begin{bmatrix} 0 & 0 \\ 0 & G \end{bmatrix}\begin{Bmatrix} E^{\text{int}} \\ E^{bound} \end{Bmatrix}=\begin{Bmatrix} f^{\text{int}} \\ f^{ext} \end{Bmatrix}$$

FSS-PRISM(Planar SERAT Code: FEM+Moment Method)

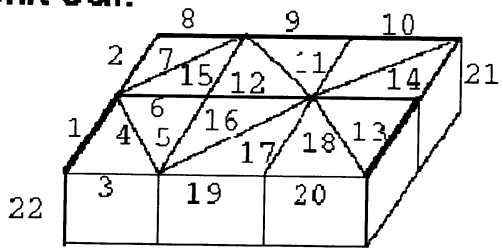
- Development of FSS-PRISM started during second quarter
- Implemented Periodic Boundary Conditions(PBCs) along vertical walls of the periodic element
- Modified code for FSS (open boundary at bottom surface) and antenna analysis
- Wrote interface and ported UH Periodic Green's function with Ewald Acceleration.
- Began testing code for simple FSS and antenna elements.
- Even with the Ewald speed-up, the periodic Green's function reduced speed by at least an order of magnitude

Infinite Periodic Array



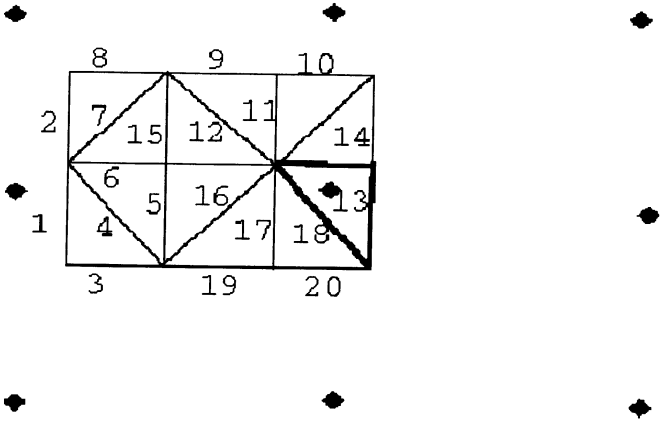
Periodic Boundary Condition (FE)

Unit Cell:



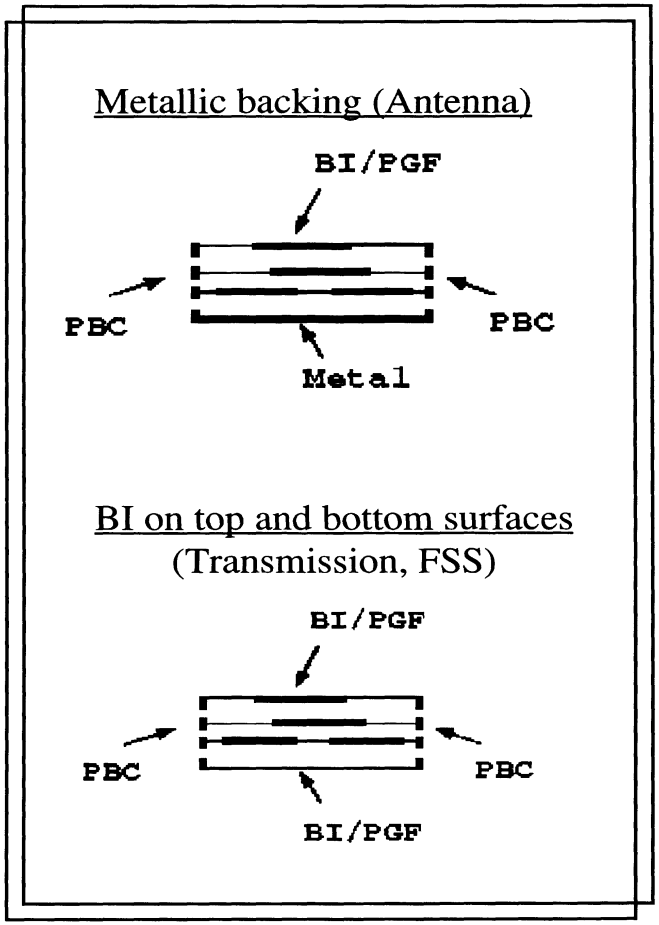
Periodic Free-Space Green's Function (BI)

Unit Cell with Image Sources:



FSS_PRISM Current Capabilities

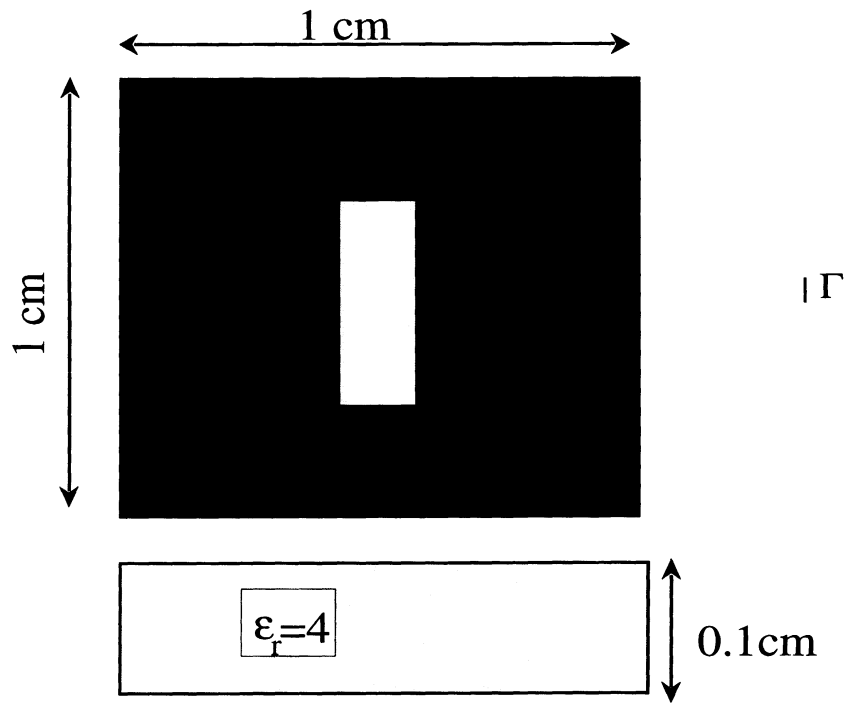
- Periodic Boundary Condition (PBC) for FE-part
- Periodic Green's Function (PGF) for BI-part
- Metallic patches in all layers possible (Antenna, FSS)
- Probe current feeds and plane wave excitation
- Lumped impedances / resistive sheets
  (to be done; already available in FSS-BRICK code)

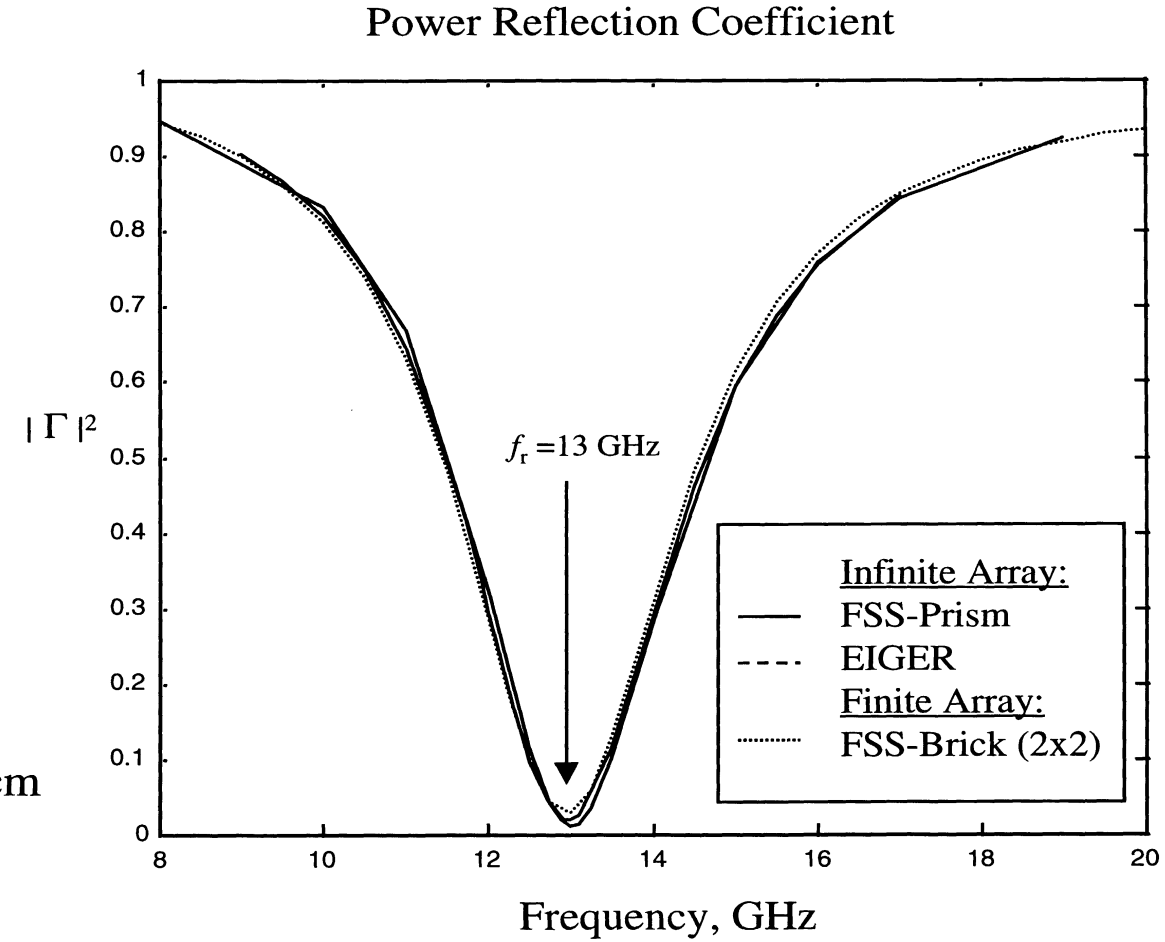


FSS_PRISM Validation

FSS Slot Array

Unit Cell :





Periodic-PRISM...next few months

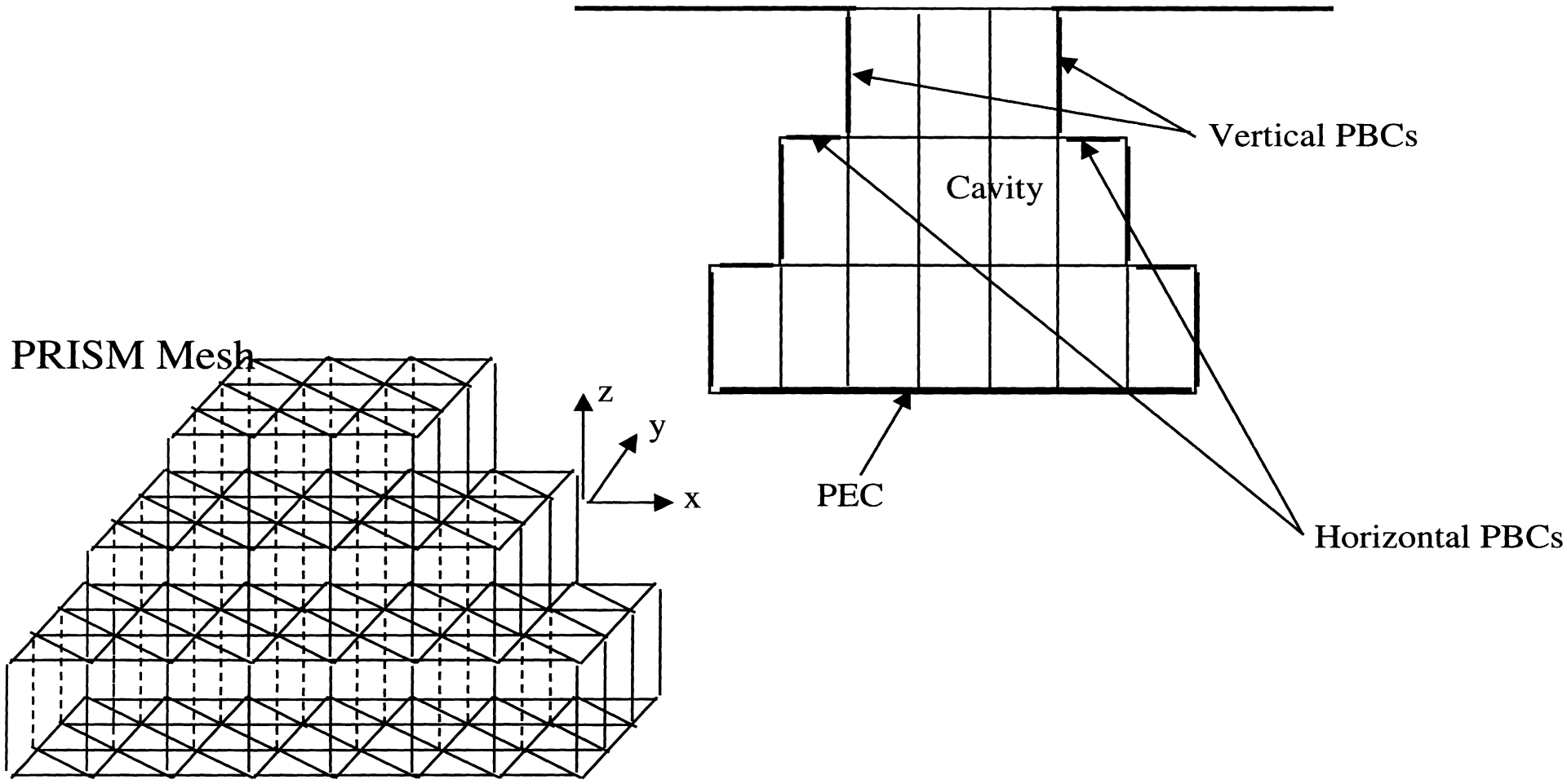
Planar hybrid FEM code is scheduled for delivery on the 5th quarter

We already have a working version at this point.

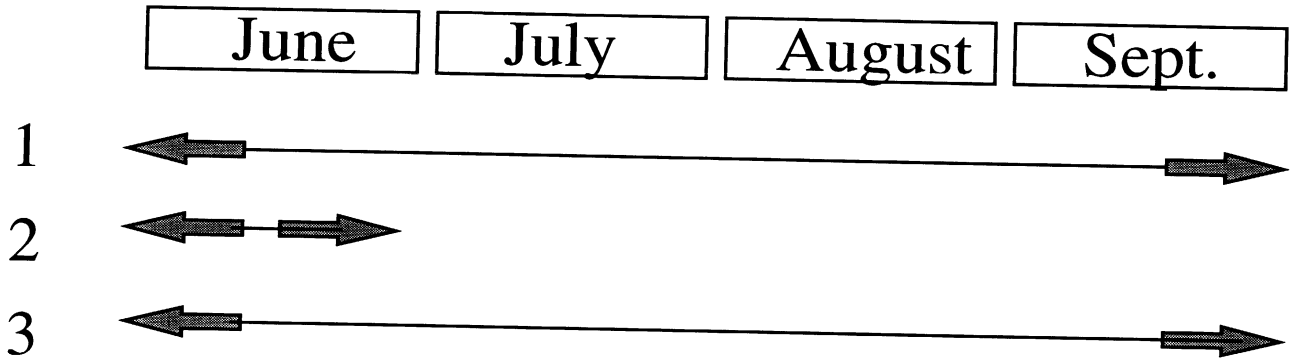
Goals for next few months:

- **develop Geometry Driver for commensurate and non-commensurate elements**
- **validate and compare (speed and accuracy) with Houston's Periodic-Eiger code**
- **optimize code**
- **port multilayer FSS Green's function**
- **Enhance modeling capabilities (loads, feeds, sheets, etc)**
- **implement fast algorithms for BI speed-up**

POSSIBLE APPROACH FOR NON-COMMENSURATE MODELING IN FSS_PRISM



TASK SCHEDULE FOR FSS_EIGER (1997)



1 Further code validation, improvements, documentation, driver development, loaded elements, calculation of additional figures of merit for antenna and FSS structures.

2 Numerical Green's function and integration with FSS_PRISM

3 Non-commensurate periodic structures.

At this point we have available:

- 1. Periodic FEM codes(FSS-BRICK and FSS-PRISM) for SERAT modeling with geometry Drivers for commensurate periods

PERIODIC FEM CODES ARE AHEAD OF SCHEDULE

- 2. Periodic MoM code (FSS-EIGER) for commensurate FSS periods

THIS TASK IS ON SCHEDULE

- These codes are currently more capable than other Periodic codes available at the start of the project.
- In principle, both codes have the capability of modeling arbitrary antenna and FSS elements
- The FEM code has the potential for modeling any non-planar details such as feeds and loads or material inhomogeneities

Activity That Will Affect Next Year's Effort

–New results on fast algorithms hold much promise for substantial speed-up above the present state-of-the-art. These algorithms handle the boundary integral (full) section of the hybrid system matrix

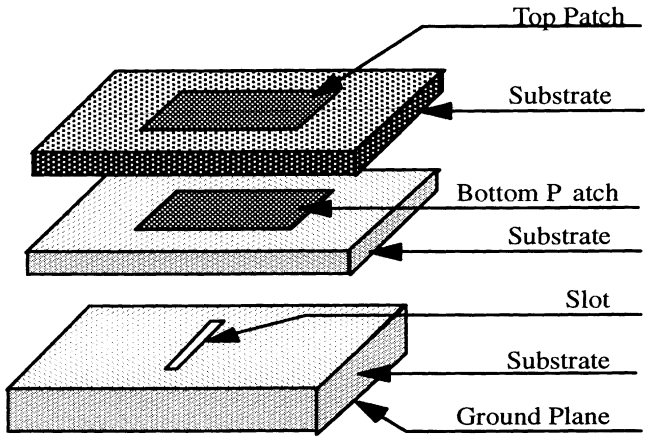
–CVSS, a new sparse LU solver was examined for handling the large FEM matrix section of the hybrid matrix. The dramatic speed-up of this solver will allow full scale FEM implementations of the SERAT.

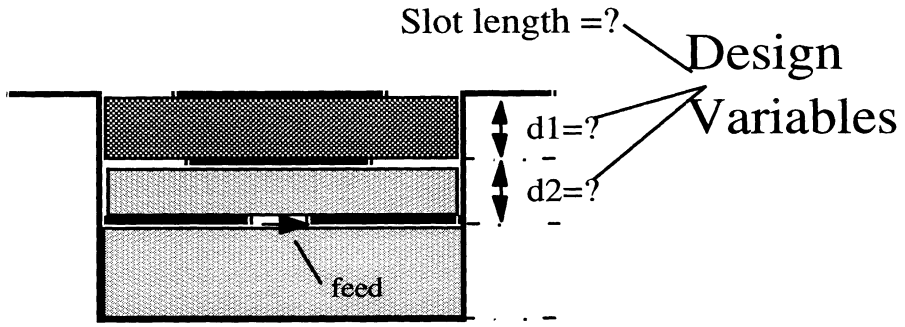
–Compared several matrix compression schemes to improve speed of iterative solvers for hybrid FEM systems

–Frequency extrapolation methods can reduce CPU time for broadband response evaluations

–We made substantial advances in integrating design optimization methods with antenna codes.

Slot Fed Dual Patch Antenna Design

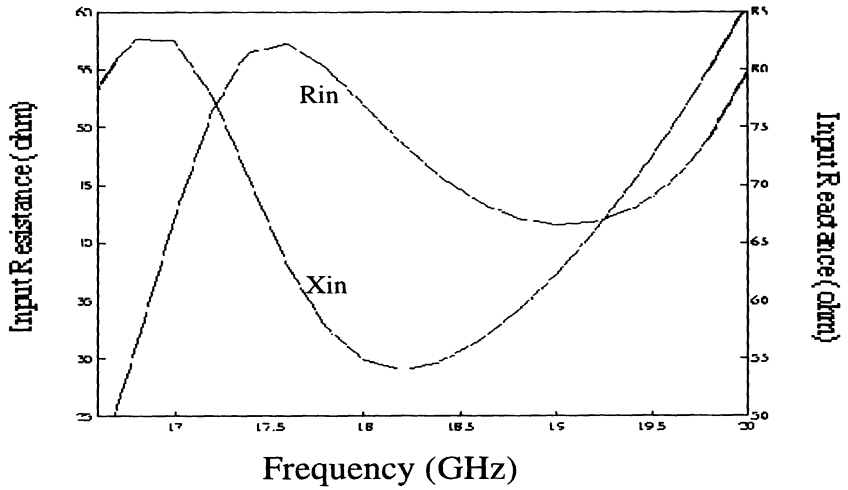




Performance at d1=0.85 mm, d2=0.55mm, slot length=4.0 mm

FINAL DESIGN:

VSWR = 1.414
BW = 18%



APPENDIX 3: Univ. of Houston (FSS-EIGER) Presentation on the May 30, 1997 review held in Ann Arbor.

FSS CODE DEVELOPMENT

- **FSS analysis is accomplished by modifying existing nonperiodic multilayer code "EIGER"**
- **EIGER has the capability to handle conductors of arbitrary shape and arbitrary element loading**
- **Efforts are focused on accelerating the convergence of the periodic series to improve efficiency:**

  Extraction of image terms from the spectral Green's function along with application of the "Ewald method"

EIGER USES A MIXED POTENTIAL INTEGRAL EQUATION FORMULATION

$$\boldsymbol{E} = -j\omega\boldsymbol{A} - \nabla\Phi - \frac{1}{\varepsilon}\nabla\times\boldsymbol{F}$$

$$\boldsymbol{H} = -j\omega\boldsymbol{F} - \nabla\Psi + \frac{1}{\mu}\nabla\times\boldsymbol{A}$$

$$\boldsymbol{A} = \mu\int_S \mathrm{G}^A\cdot\boldsymbol{J}(\boldsymbol{r}')dS'$$

$$\boldsymbol{F} = \varepsilon\int_S \mathrm{G}^F\cdot\boldsymbol{M}(r')dS' \quad , \qquad \mathrm{G}^A,\mathrm{G}^F \quad \text{dyadic potentials}$$

$$\Phi = \int_S \nabla\cdot\boldsymbol{J}(\boldsymbol{r}')K^{\Phi}(\boldsymbol{r},\boldsymbol{r}')\;dS' + \int_S J_z(\boldsymbol{r}')P_z(\boldsymbol{r},\boldsymbol{r}')dS'$$

$$\Psi = \int_S \nabla\cdot\boldsymbol{M}(\boldsymbol{r}')K^{\Psi}(\boldsymbol{r},\boldsymbol{r}')\;dS' + \int_S M_z(\boldsymbol{r}')Q_z(\boldsymbol{r},\boldsymbol{r}')dS'$$

EFFICIENT CALCULATION OF POTENTIALS REQUIRES A SOPHISTICATED SERIES ACCELERATION ALGORITHM

$$G^A = \sum_{p,q}\left[\tilde{G}^A_{pq} - \sum_i \tilde{G}^{Ai}_{pq}\right]$$

(spectral form of layered media Green's function with asymptotic form removed)

$$+\sum_i\left[\sum_{p,q}\tilde{G}^{'Ai}_{pq} + \sum_{r,s}\left[G^{'Ai}_{r,s} - \delta_{r0}\delta_{s0}G^{Ai}_{r,s}\right]\right]$$

(Ewald hybrid spectral - spatial summation method for terms representing periodic direct and quasi - static images)

$$+\sum_i G^{Ai}_{0,0}$$

(self terms removed from Ewald series for handling singularities)

EWALD ACCELERATION METHOD

- **The Ewald method is an efficient technique for calculating the periodic free-space Green's function**
- **The periodic free-space Green' s function is the slowest converging part of the multilayered Green's function**

  Ewald method converges much faster than conventional acceleration schemes

  Typically, the summation limits may be truncated at N = 1

OVERVIEW OF EWALD METHOD

$$G(\mathbf{r},\mathbf{r}') = \sum_{m=-\infty}^{\infty}\sum_{n=-\infty}^{\infty} e^{-jk_{t00}\cdot\rho_{mn}}\left(\frac{e^{-jkR_{mn}}}{4\pi R_{mn}}\right)$$

1) Use the following identity:

$$\frac{e^{-jkR_{mn}}}{4\pi R_{mn}} = \frac{2}{\sqrt{\pi}}\int_0^{\infty} e^{-R_{mn}^2 s^2 + \frac{k^2}{4s^2}}\, ds$$

2) Split the integral into two parts:

$$G_1(\mathbf{r},\mathbf{r}') = \sum_{m=-\infty}^{\infty}\sum_{n=-\infty}^{\infty} e^{-j\mathbf{k}_{t00}\cdot\rho_{mn}}\frac{2}{\sqrt{\pi}}\int_0^{E} e^{-R_{mn}^2 s^2 + \frac{k^2}{4s^2}}\, ds$$

$$G_2(\mathbf{r},\mathbf{r}') = \sum_{m=-\infty}^{\infty}\sum_{n=-\infty}^{\infty} e^{-j\mathbf{k}_{t00}\cdot\rho_{mn}}\frac{2}{\sqrt{\pi}}\int_E^{\infty} e^{-R_{mn}^2 s^2 + \frac{k^2}{4s^2}}\, ds$$

OVERVIEW OF EWALD METHOD

3) The second integral is cast into the form of the complimentary error function (which decays exponentially):

$$\frac{2}{\sqrt{\pi}}\int_E^\infty e^{-R_{mn}^2 s^2+\frac{k^2}{4s^2}}\,ds = \frac{1}{2R_{mn}}\left[e^{-jkR_{mn}}\operatorname{erfc}\left(R_{mn}E-\frac{jk}{2E}\right)+e^{jkR_{mn}}\operatorname{erfc}\left(R_{mn}E+\frac{jk}{2E}\right)\right]$$

Hence:

$$G_2(\mathbf{r},\mathbf{r}') = \sum_{m=-\infty}^{\infty}\sum_{n=-\infty}^{\infty}\frac{e^{-j\mathbf{k}_{t00}\cdot\rho_{mn}}}{8\pi R_{mn}}\frac{1}{2R_{mn}}\left[e^{-jkR_{mn}}\operatorname{erfc}\left(R_{mn}E-\frac{jk}{2E}\right)+e^{jkR_{mn}}\operatorname{erfc}\left(R_{mn}E+\frac{jk}{2E}\right)\right]$$

4) The Poisson transformation is used to transform the first sum into one that also involves the complimentary error function.

$$G_1(\mathbf{r},\mathbf{r}') = \frac{1}{A}\sum_{m=-\infty}^{\infty}\sum_{n=-\infty}^{\infty}\frac{1}{4jk_{zmn}}e^{-j\mathbf{k}_{tmn}\cdot(\rho-\rho')}\left[e^{-jk_{zmn}|z-z'|}\operatorname{erfc}\left(\frac{jk_{zmn}}{2E}-|z-z'|E\right)+e^{jk_{zmn}|z-z'|}\operatorname{erfc}\left(\frac{jk_{zmn}}{2E}+|z-z'|E\right)\right]$$

ACCELERATION RESULTS

Results are presented for the following case:

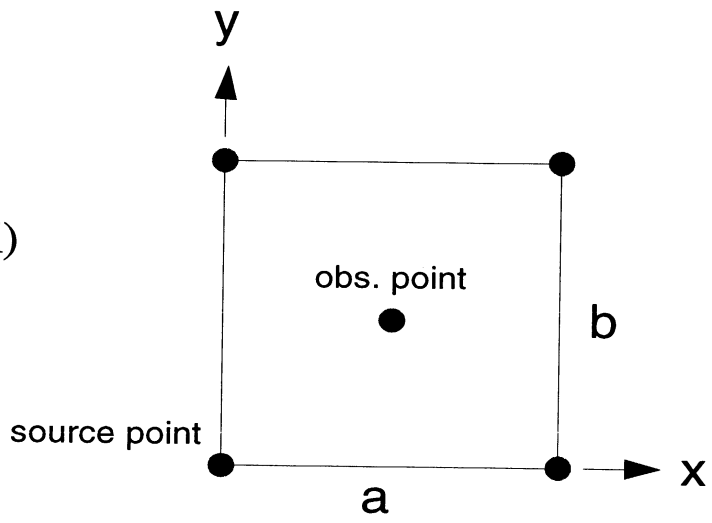
$a = b = 0.5\lambda_0$ (unit cell dimensions)

$x' = y' = z' = 0$ (source is at the origin)

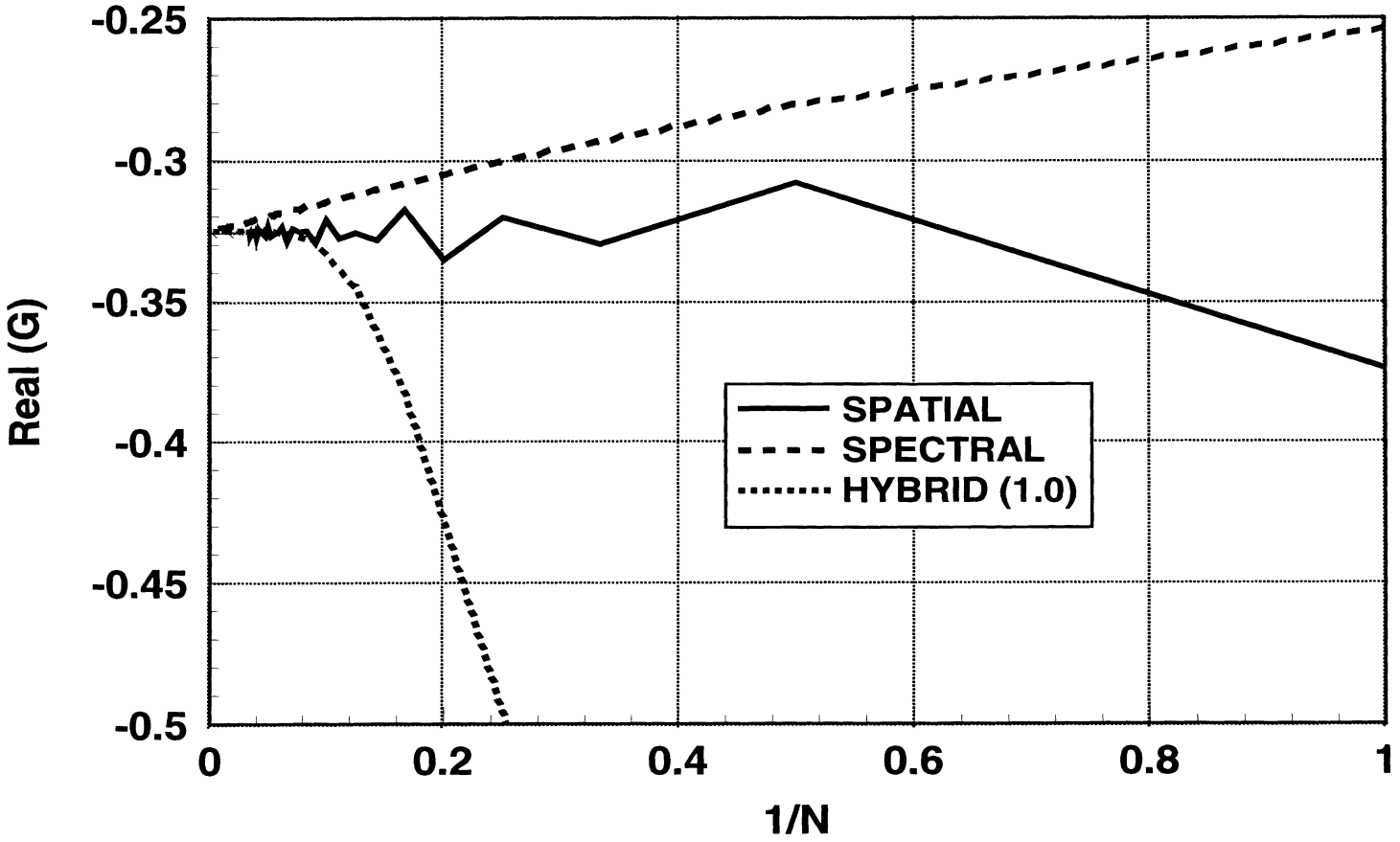
$x = y = 0.25\lambda_0$ (observation point is at the center of the unit cell)

$z = 0$ (observation point is in the source plane)

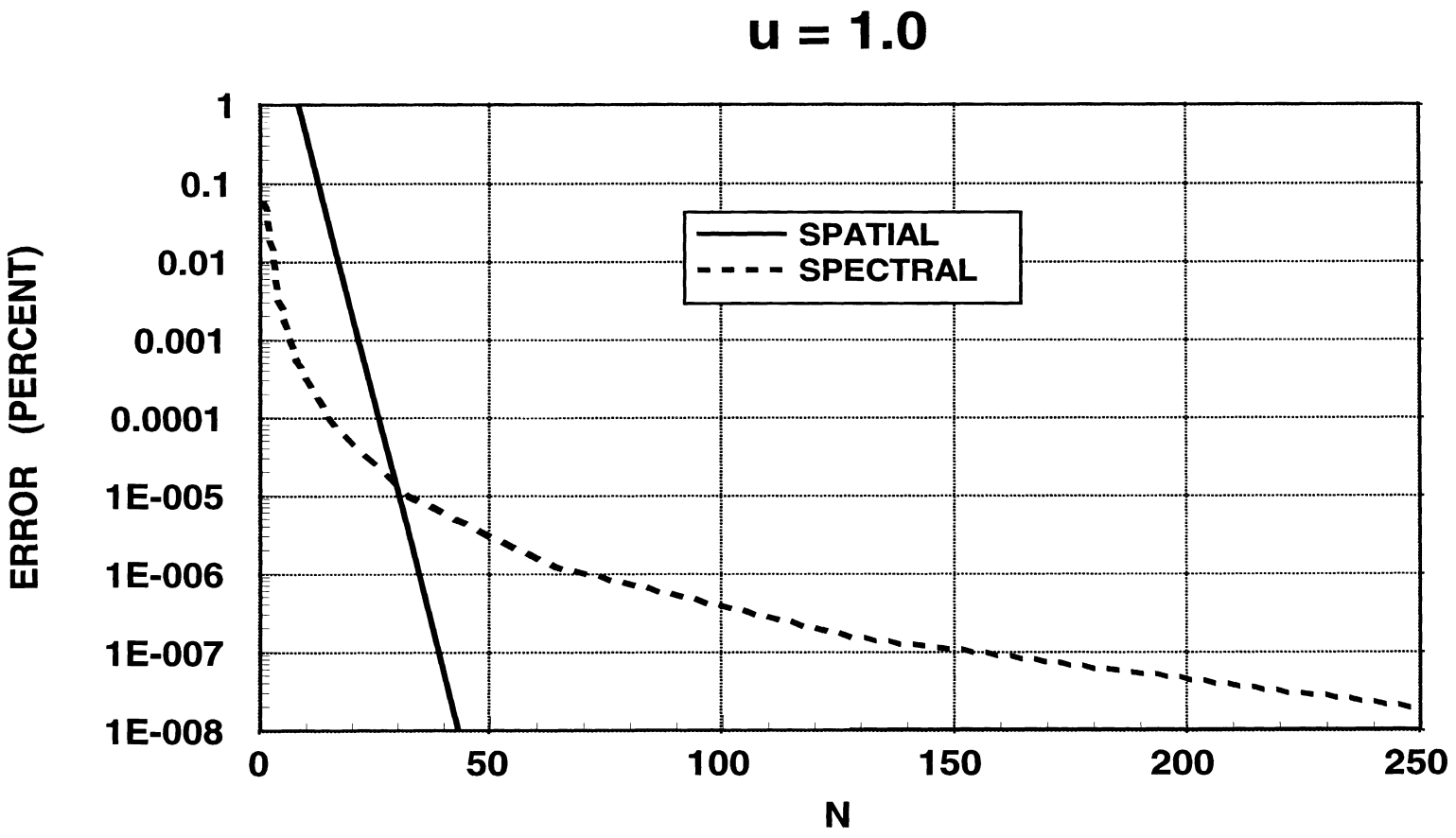
$k_{x0} = k_{y0} = 0$ (incident wavenumbers are zero)



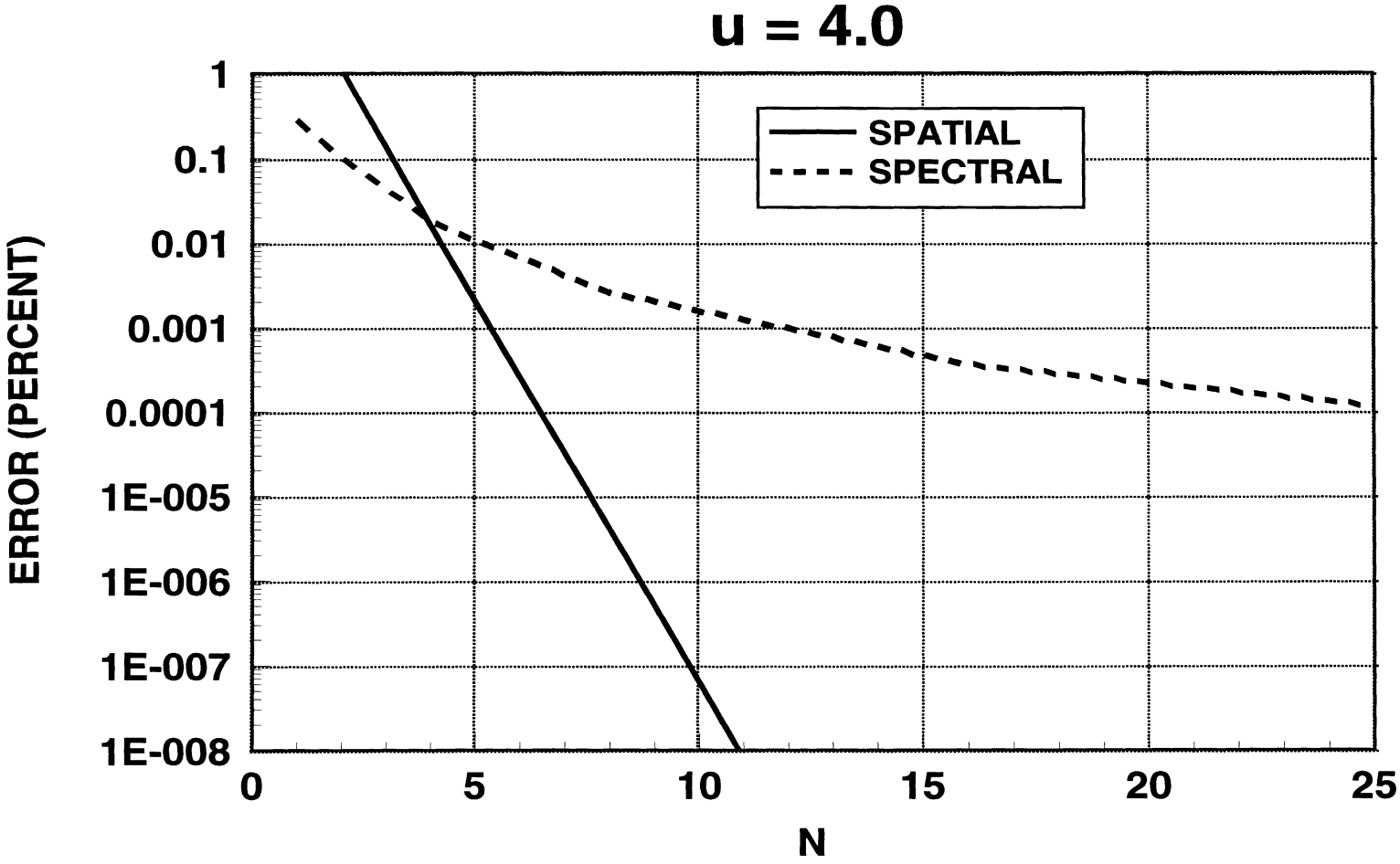
SINGH ACCELERATION SCHEME



SINGH ACCELERATION SCHEME

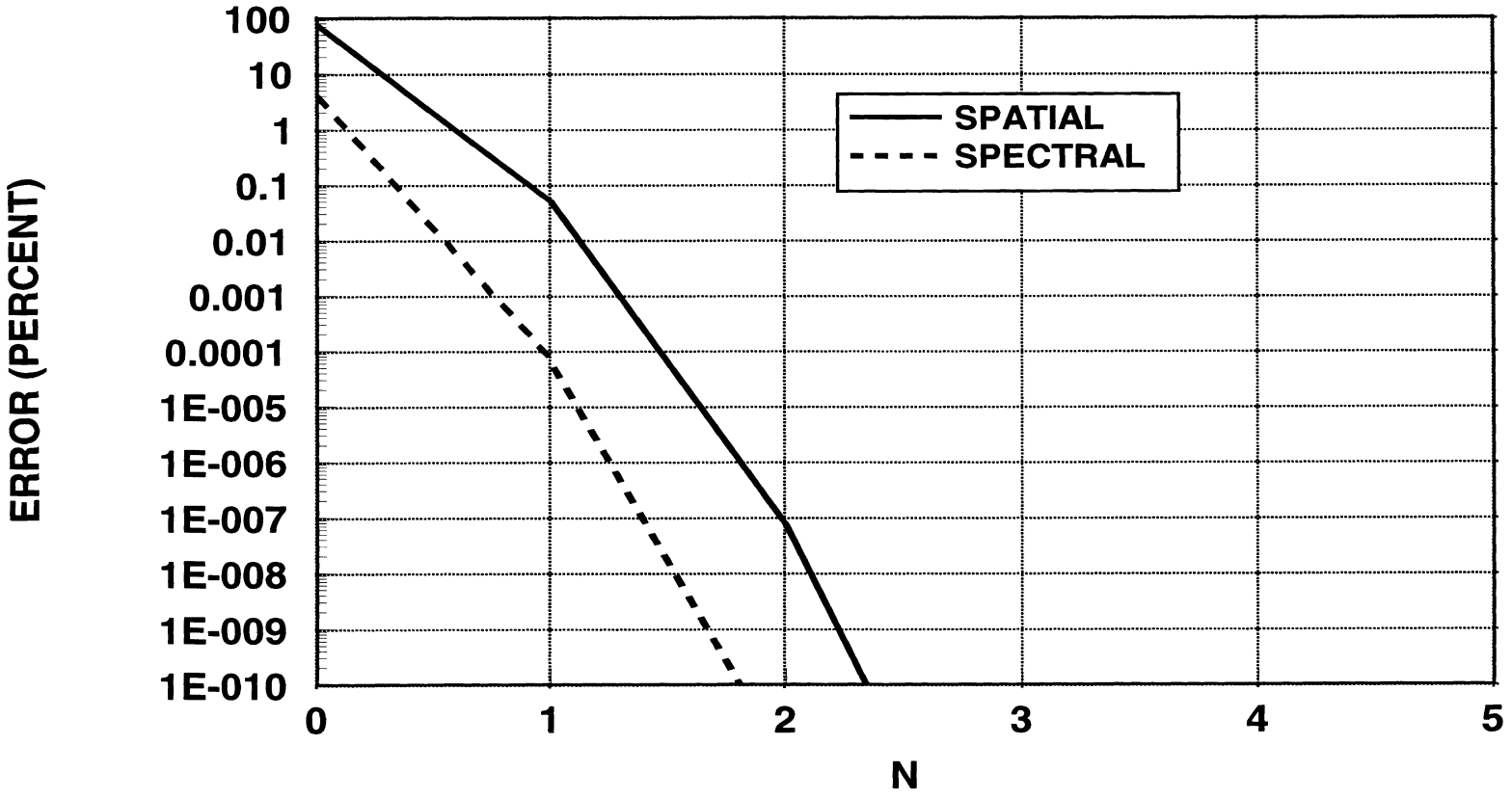


SINGH ACCELERATION SCHEME

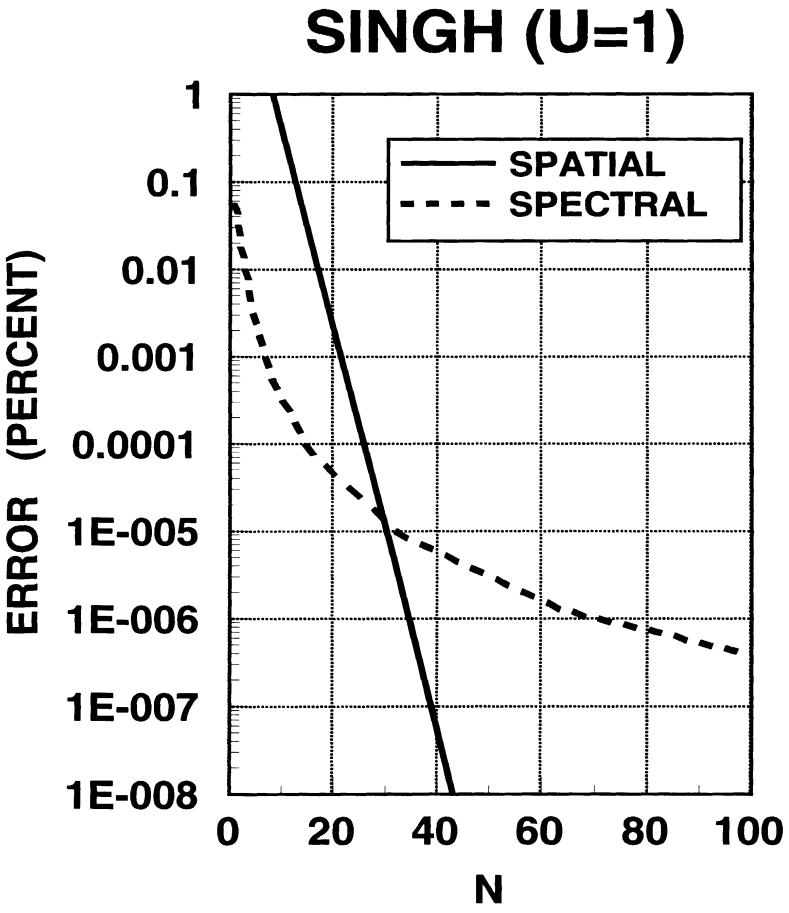


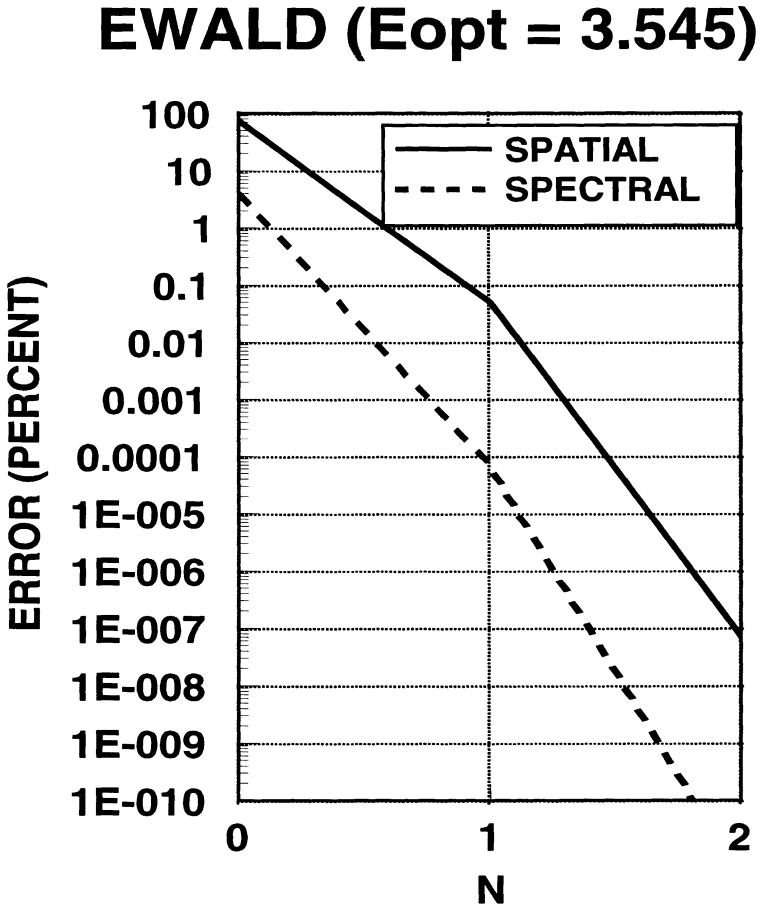
EWALD METHOD

E = Eopt = 3.545



COMPARISON OF ACCELERATION SCHEMES: SINGH AND EWALD METHODS





COMPARISON OF CALCULATION TIMES

define: $$T = \frac{\text{Ewald calculation time}}{\text{Singh calculation time}} = \frac{2\times\left(2\left(N_{Ewald}+1\right)\right)^2\times 10}{2\times\left(2\left(N_{Singh}+1\right)\right)^2}$$

(Assumes evaluation of erfc(z) is ten times more expensive than evaluation of the free-space Green's function.)

Ratio T depends on the level of accuracy required (T is smaller as the accuracy increases).

sample calculations:

- Assume a required error of 0.1%: $T = 1.11$
- Assume a required error of 10^{-7} %: $T = 0.0028$

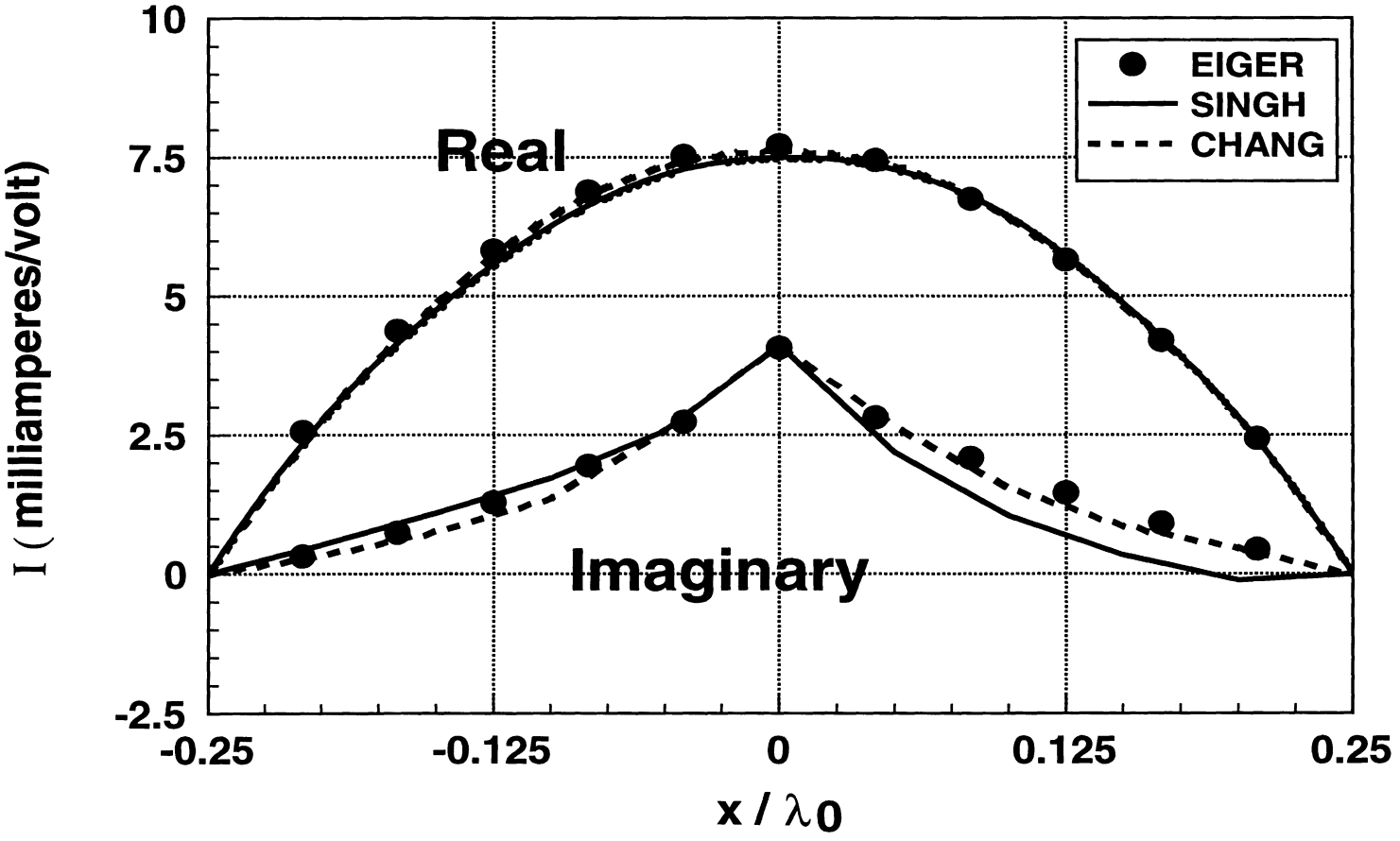
CODE VALIDATION

- **EIGER code with Ewald acceleration scheme has been validated for various cases:**

  - **arrays of dipole antennas in free space or over ground plane**
  - **arrays of dipole antennas printed on grounded substrate**
  - **FSS dipole arrays**
  - **FSS slot arrays**

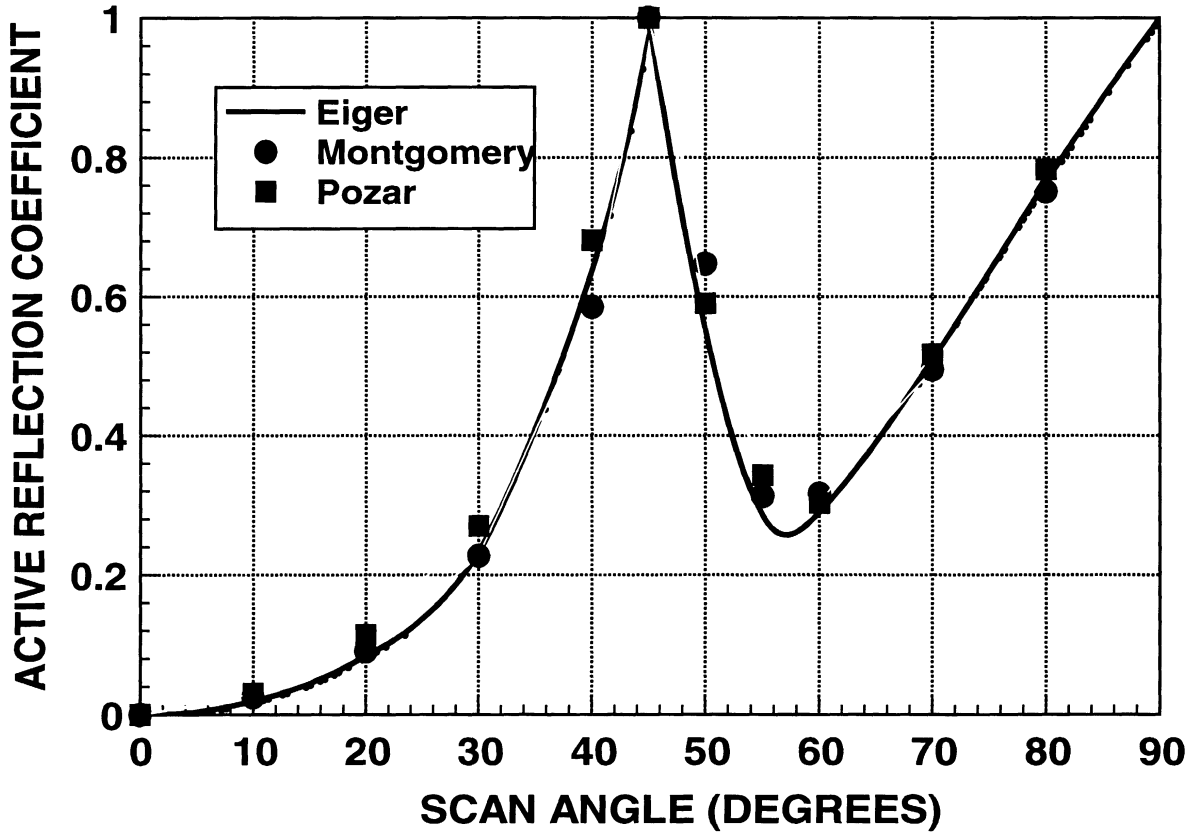
CODE VALIDATION

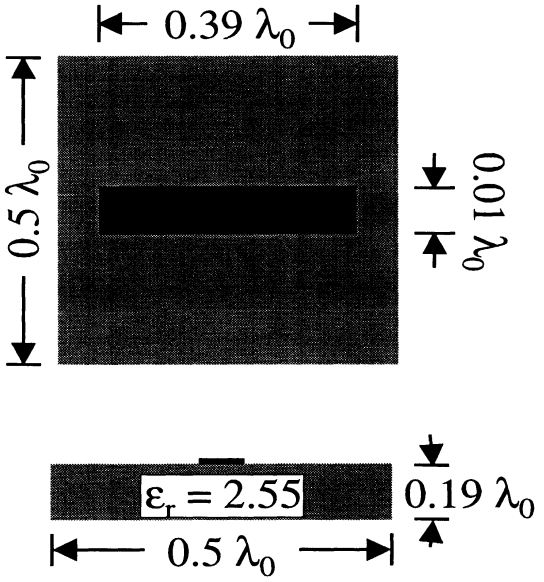
HALF-WAVE DIPOLE ARRAY IN FREE SPACE



CODE VALIDATION

ARRAY OF MICROSTRIP DIPOLES: E-PLANE SCAN

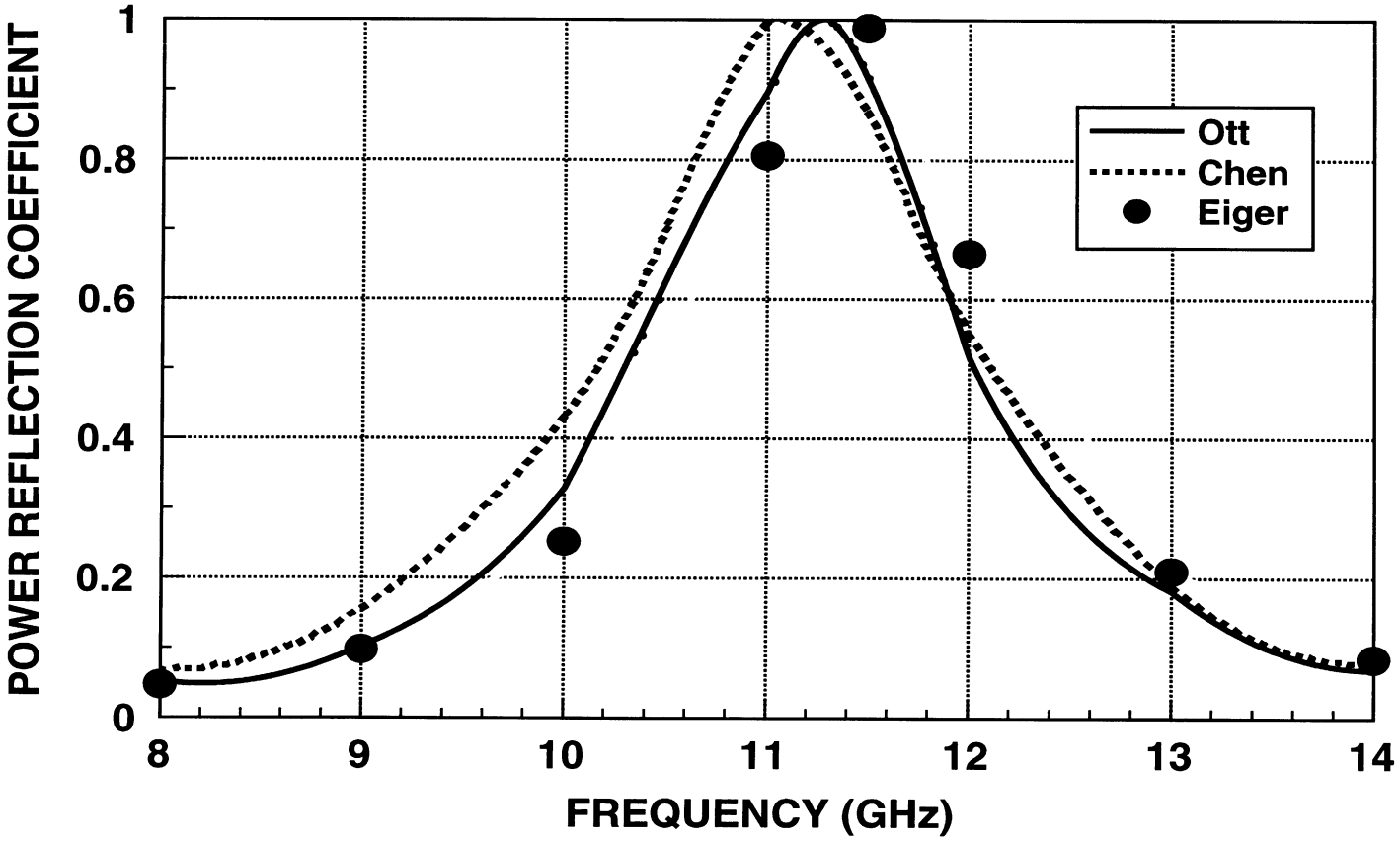




CODE VALIDATION

SINGLE-LAYER FSS IN FREE SPACE

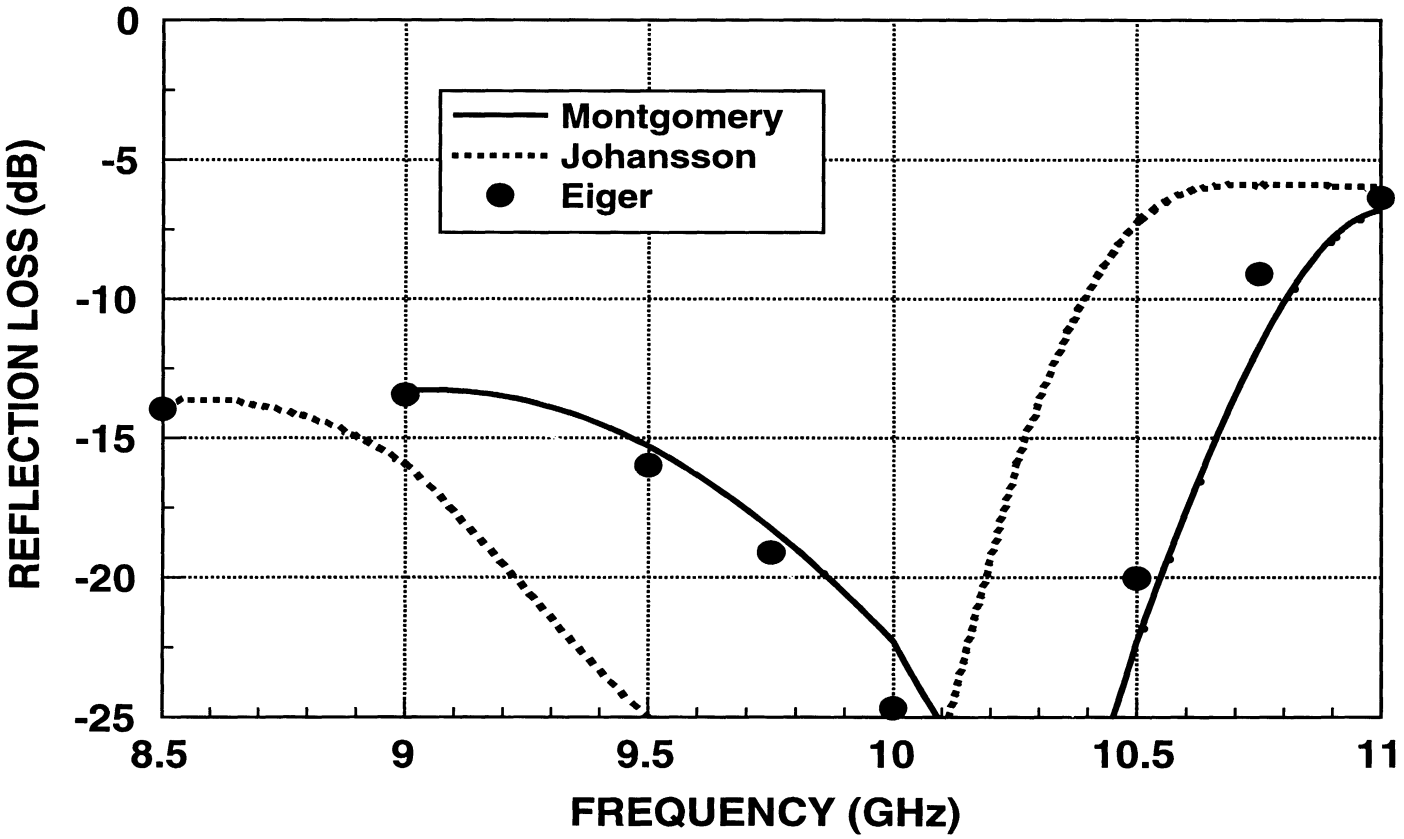
POWER REFLECTION OF (0,0) MODE



CODE VALIDATION

THREE-LAYER FSS IN FREE SPACE

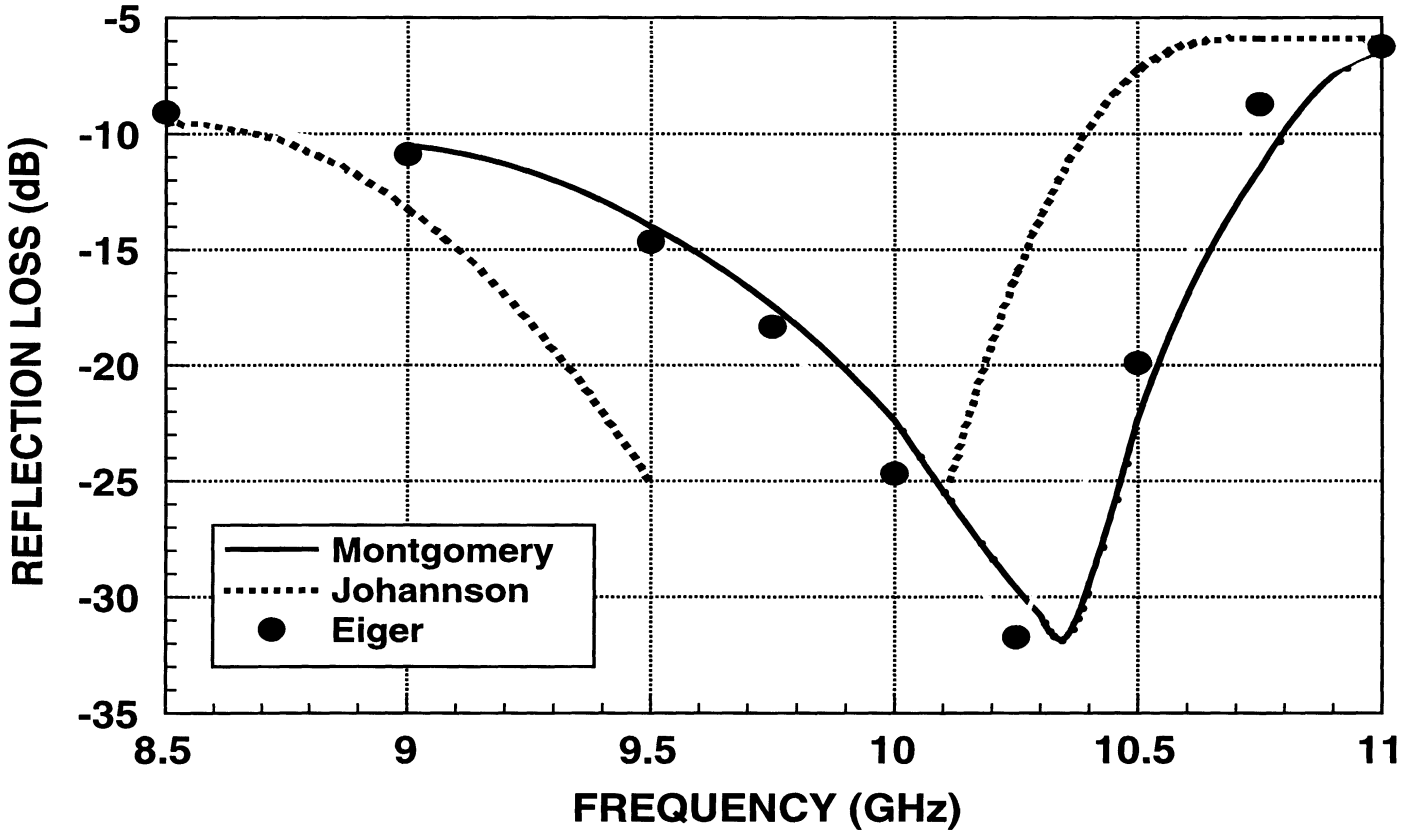
REFLECTION LOSS OF (0,0) MODE



CODE VALIDATION

THREE-LAYER FSS IN FREE SPACE

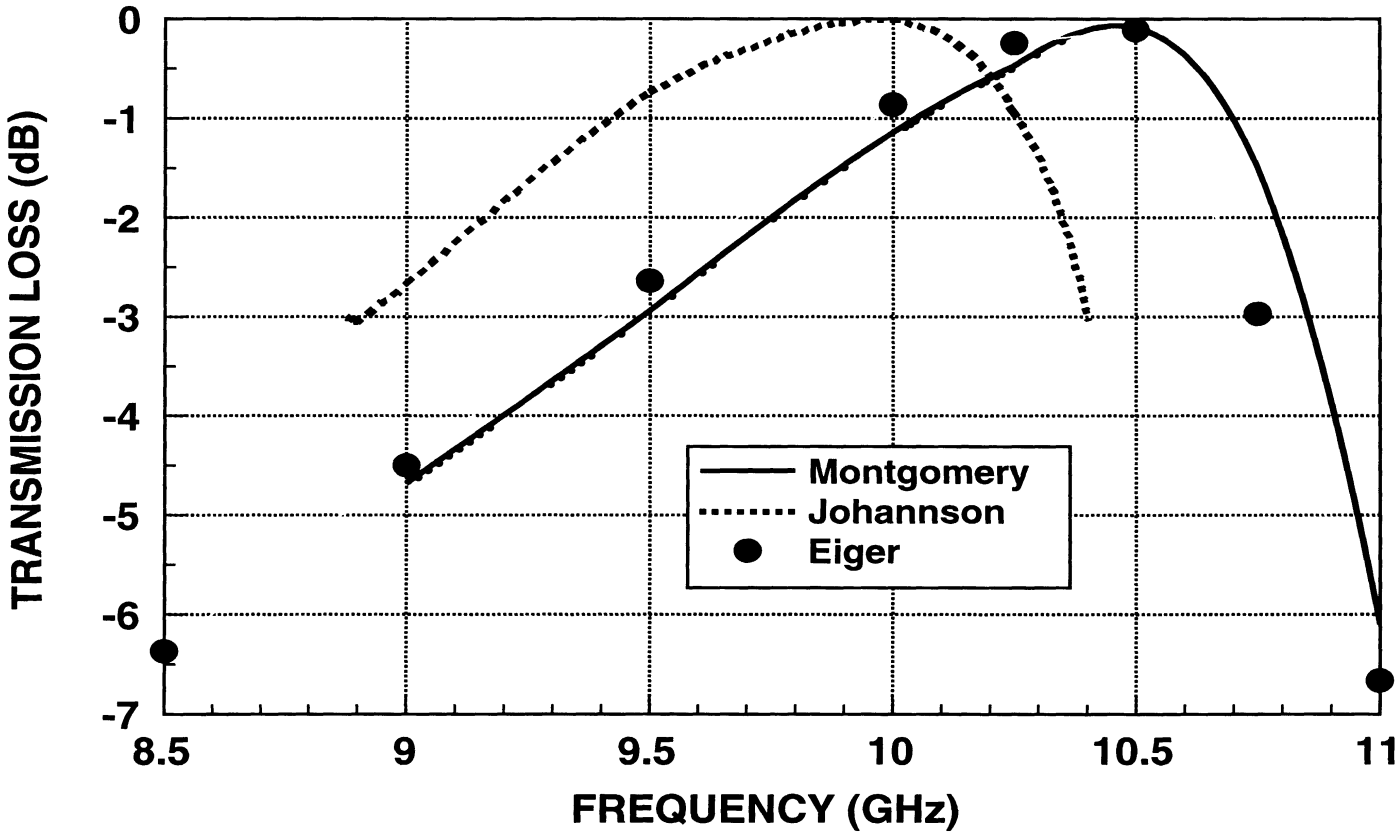
REFLECTION LOSS OF (-1,0) MODE



CODE VALIDATION

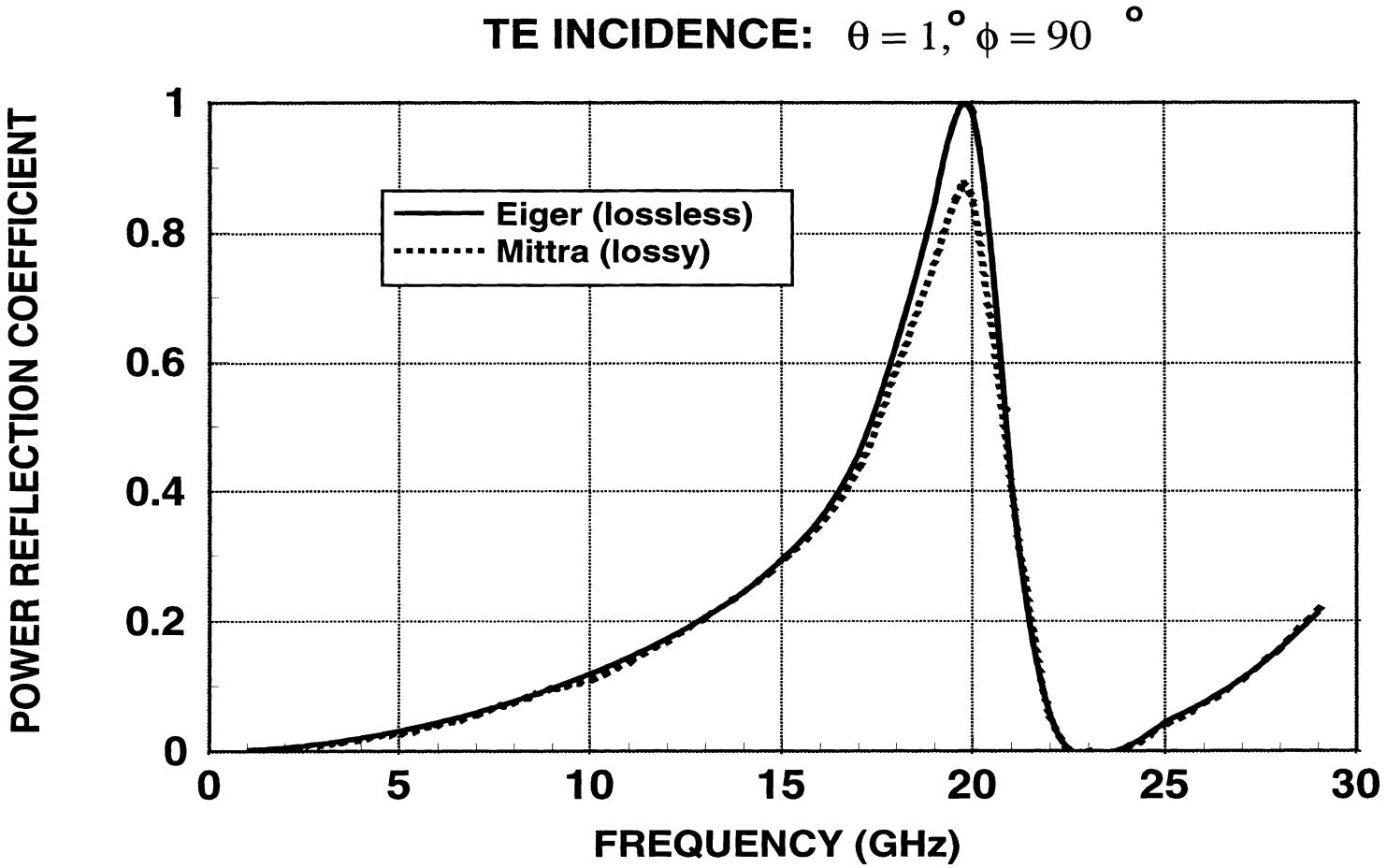
THREE-LAYER FSS IN FREE SPACE

TRANSMISSION LOSS OF (-1,0) MODE



CODE VALIDATION

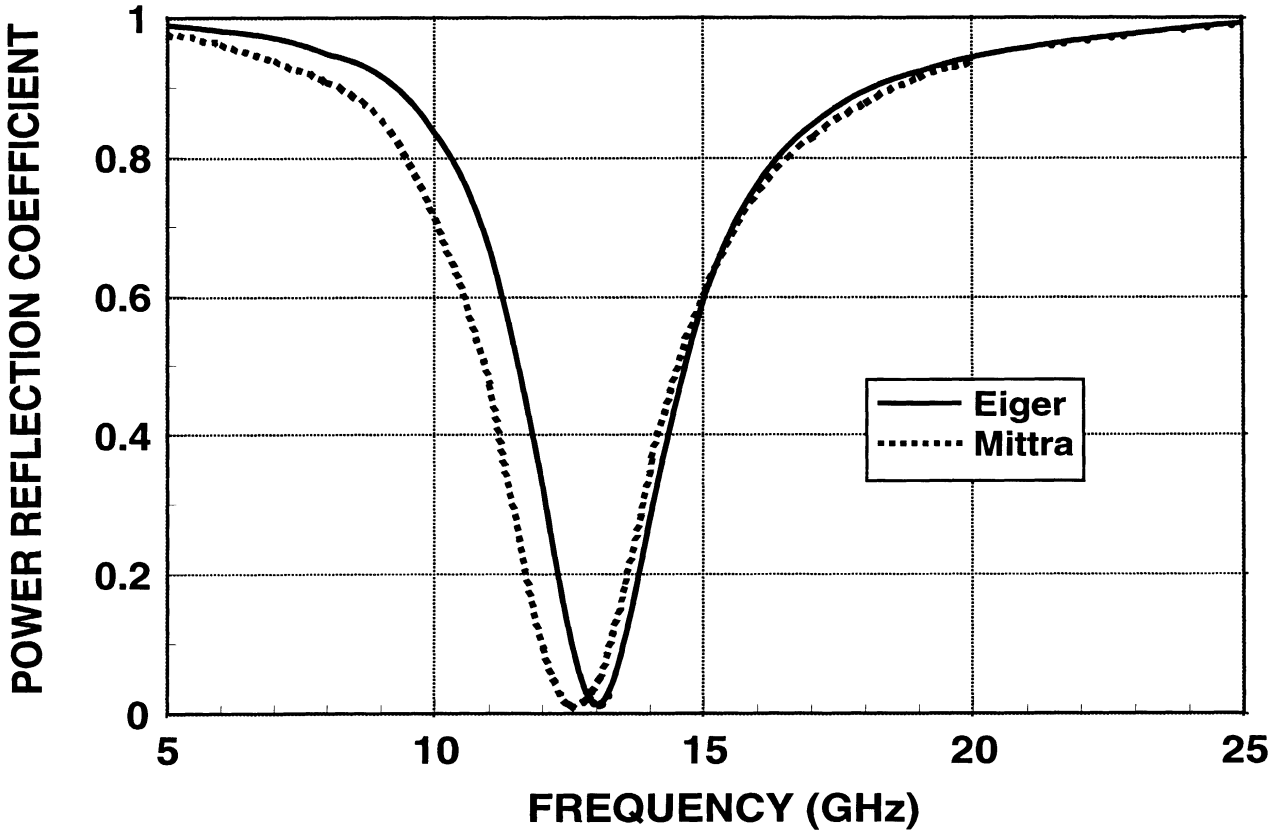
FSS EMBEDDED IN DIELECTRIC LAYER

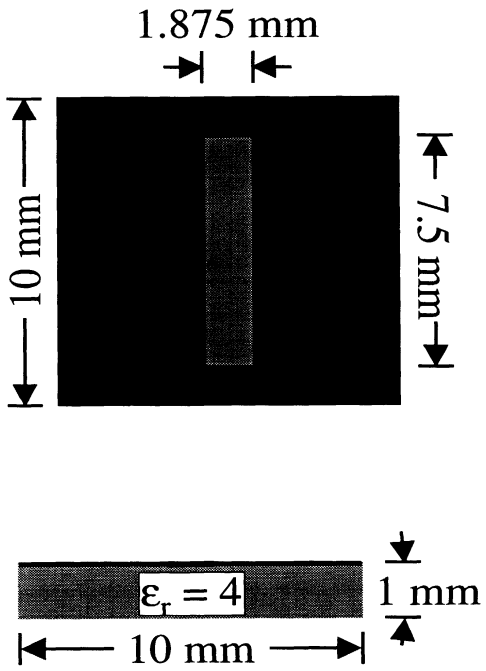


CODE VALIDATION

FSS SLOT ARRAY

TM INCIDENCE: $\theta = 1^{\circ}, \ \phi = 0^{\circ}$



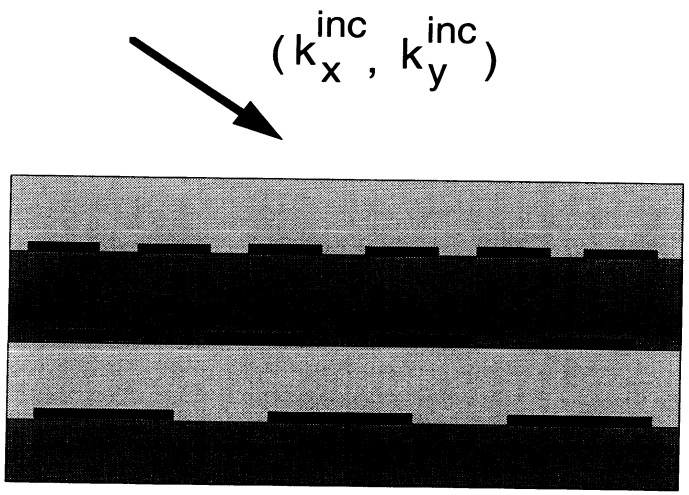


ANALYSIS OF FSS STRUCTURES WITH NONCOMMENSURATE PERIODS

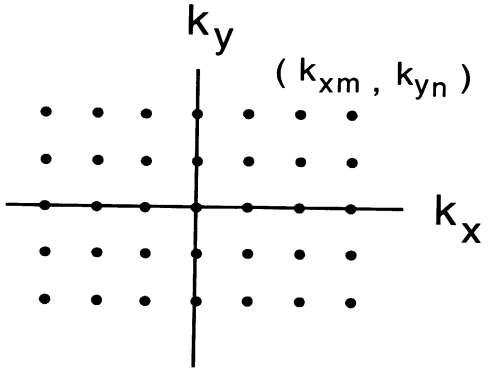
- **Formulation has been developed for analysis of FSS structures consisting of multiple screens with noncommensurate periods**
- **Analysis is based on a discrete spectral propagator method which cascades both propagating and evanescent parts of the spectrum between layers**
- **Method allows for arbitrary periodicity in the different FSS screens that are cascaded**

ANALYSIS OF FSS WITH NON-COMMENSURATE PERIODS

- The formulation of a systematic method for the general analysis of an FSS structure with non-commensurate periods has been developed.

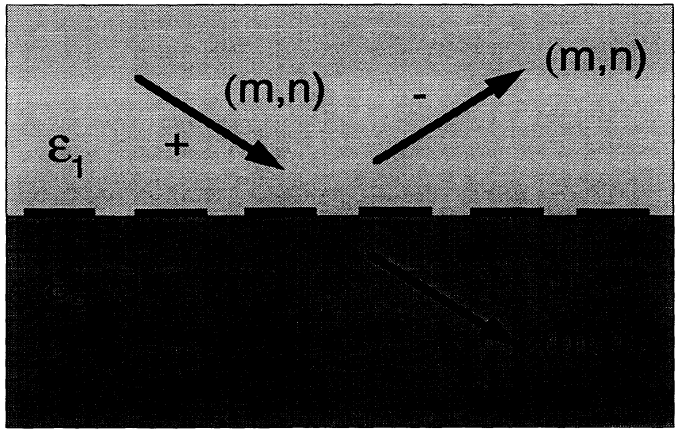


(m,n): $k_{xm} = m(\Delta k_x)$, $k_{yn} = n(\Delta k_y)$



- **Discrete Spectral Propagator** method: Introduce a *fixed* set of wavenumbers that discretizes wavenumber space (independent of structure). Propagating and evanescent wavenumbers are included in the set.

- Each screen is characterized by a generalized scattering matrix [S] in which all scattered wavenumbers are approximately recast into the *fixed* wavenumber set $\left(k_{xm}, k_{yn}\right)$.



$$(\psi) = \begin{pmatrix} A_z^{(m,n)} \\ --- \\ F_z^{(m,n)} \end{pmatrix}$$

$$\begin{pmatrix} \psi_1^- \\ --- \\ \psi_2^- \end{pmatrix} = \begin{bmatrix} S_{11} & | & S_{12} \\ - & - & - \\ S_{21} & | & S_{22} \end{bmatrix} \begin{pmatrix} \psi_1^+ \\ --- \\ \psi_2^+ \end{pmatrix}$$

- Each incident wavenumber $\left(k_{xm}, k_{yn}\right)$ produces an infinite set of wavenumbers $\left(k_{xp}, k_{yq}\right)$: $k_{xp}^{(m)} = k_{xm} + \frac{2\pi p}{d_x}$ $k_{yq}^{(n)} = k_{yn} + \frac{2\pi q}{d_y}$

- Therefore, to obtain the elements of the scattering matrix a renormalization procedure is used to recast the amplitudes of the scattered waves into an approximate representation in the discrete propagator set $\left(k_{xm}, k_{yn}\right)$.

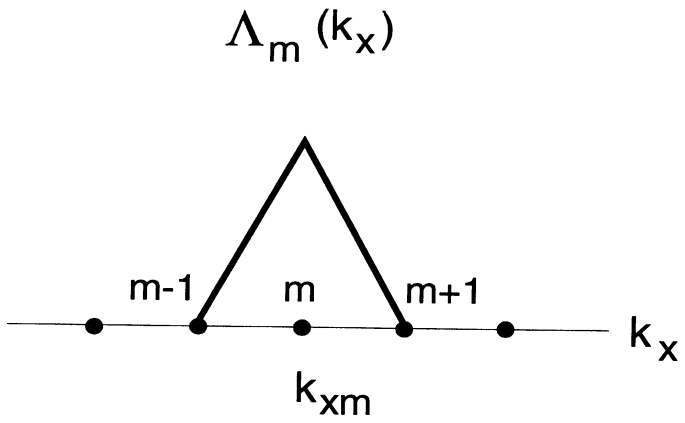
renormalization:

$$\psi_i^{+\,(m,n)} = \sum_{p,q} \psi_i^{+\,(p,q)}\, W_{(m,n)}^{(p,q)}$$

The weighting coefficients $W_{(m,n)}^{(p,q)}$ are determined from linear interpolation in wavenumber space:

$$W_{(m,n)}^{(p,q)} = \Lambda_m\left(k_{xp}\right)\Lambda_n\left(k_{yq}\right),$$

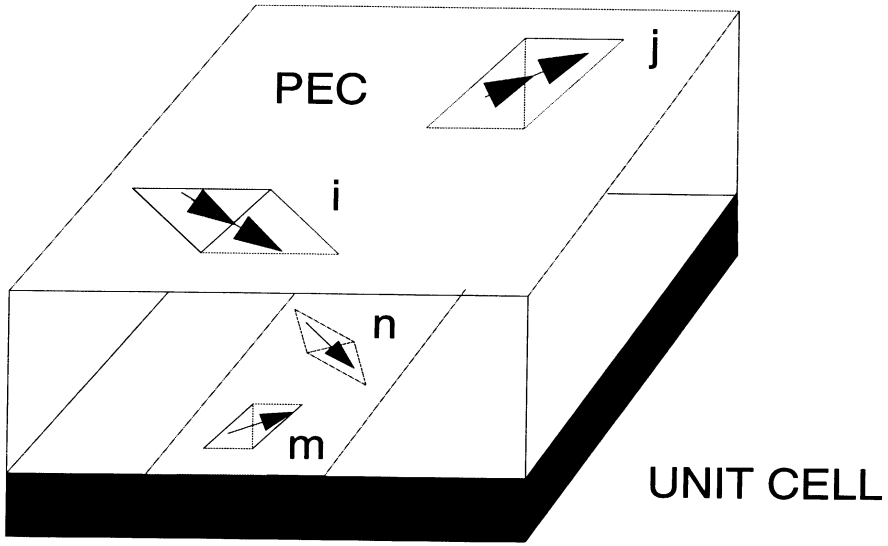
where, e.g., $\Lambda_m(k_x)$ is a linear (rooftop) interpolating function, centered at $k_x = k_{xm}$.



- The scattering matrices for different screens are cascaded to obtain the overall scattering matrix for the entire FSS structure.

- The scattering matrix can be used to help optimize the FSS design, and will also be used to terminate the FEM mesh for the analysis of the combined antenna / FSS structure.

NUMERICAL GREEN'S FUNCTION



Interaction needed between magnetic current j and testing function i:

$$\left[Y_{ij}\right]-\left[\beta_{in}\right]\left[Z_{mn}\right]^{-1}\left[\beta'_{mj}\right]$$

Matrix Element Definitions

DEFINE:

$\mathbf{E}(\mathbf{J},\mathbf{M})$ – electric field operator

$\mathbf{H}(\mathbf{J},\mathbf{M})$ – magnetic field operator

$\mathbf{J} = \sum_{n} \mathrm{I}_{n} \Lambda_{n}(\mathbf{r})$ – electric current on conductors

$\mathbf{M} = \sum_{j} \mathrm{V}_{j} \Lambda_{j}^{a}(\mathbf{r})$ – magnetic currents at interface

$$Y_{ij} = -\left\langle \Lambda_{i}^{a}, \mathbf{H}\left(0, \Lambda_{j}^{a}\right)\right\rangle \qquad \beta'_{mj} = \left\langle \Lambda_{m}, \mathbf{E}\left(0, \Lambda_{j}^{a}\right)\right\rangle$$

$$Z_{mn} = -\left\langle \Lambda_{m}, \mathbf{E}\left(\Lambda_{n}, 0\right)\right\rangle \qquad \beta_{in} = \left\langle \Lambda_{i}^{a}, \mathbf{H}\left(\Lambda_{n}, 0\right)\right\rangle$$

Obtain System Matrix for an Aperture at the Interface

- system matrix for an aperture problem:

$$\begin{bmatrix} [Y_{ij} + Y_{ij}^{ext}] & [\beta_{in}] \\ [\beta'_{mj}] & [Z_{mn}] \end{bmatrix} \begin{bmatrix} [V_j] \\ [I_n] \end{bmatrix} = \begin{bmatrix} [I_i^{sc}] \\ [0] \end{bmatrix}$$

- Fill upper half space with PMC so that $Y_{ij}^{ext} \rightarrow 0$.

- Write system matrix blocks to a data file and form

$$[Y_{ij}] - [\beta_{in}][X_{nj}]$$

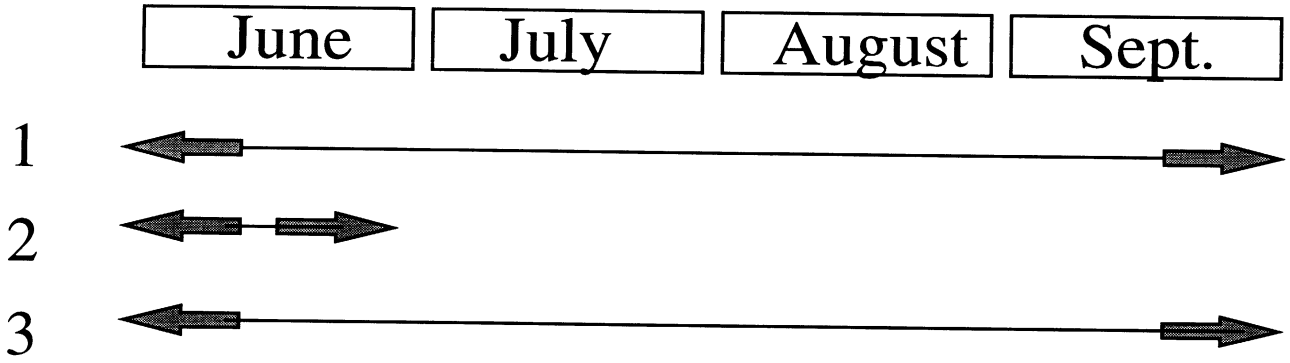
where

$$[Z_{mn}][X_{nj}] = [\beta'_{mj}]$$

NEAR-TERM GOALS

- **Calculation of reactions to provide interface between FSS and FEM codes**
- **Further code validation, improvements, documentation, driver development, loaded elements, calculation of additional figures of merit for antenna and FSS structures.**
- **Begin coding for FSS structures with noncommensurate periods**

TASK SCHEDULE (1997)



1 Further code validation, improvements, documentation, driver development, loaded elements, calculation of additional figures of merit for antenna and FSS structures.

2 Numerical Green's function.

3 Non-commensurate periodic structures.

Univ. of Michigan/Univ. of Houston SERAT Code Development Team

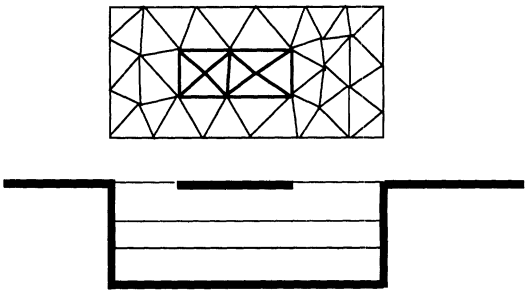
APPENDIX 4: FSS-PRISM presentation by T. Eibert given on the May 30, 1997 review held in Ann Arbor

PRISM-Development

PRISM: Finite Element/Boundary Integral Code Based on Right-Angled or Distorted Prismatic Finite Elements and Triangular Boundary Elements

- **Modeling Flexibility in FE-Part, (Arbitrary Materials, Arbitrary Geometries, ...)**
- **Geometrical Adaptability**
- **Ease of Meshing**

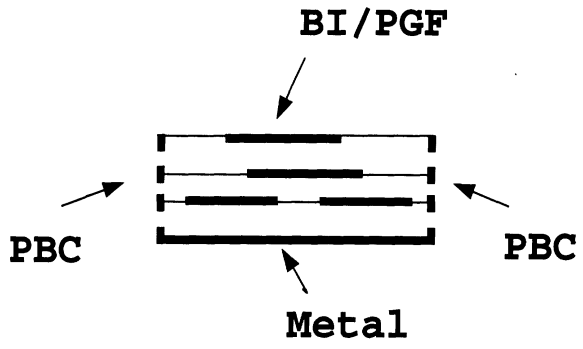
Starting Point: PRISM-code for Cavity-Backed Planar Antennas



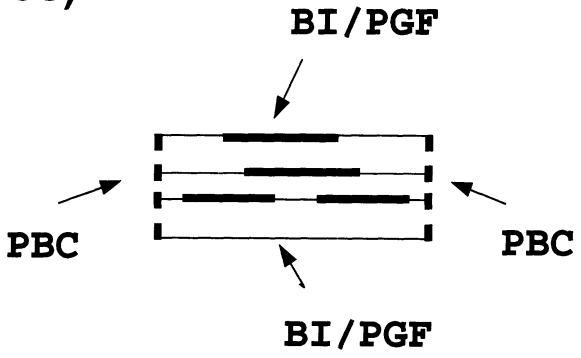
- Metallic Boundaries
- Antenna Elements in Top Surface
- Probe Current Feeds
- Vertically Grown Volume Mesh Starting from a Given Surface Mesh
- Boundary Integral Based on Free-Space Green's Function

Periodic PRISM-Code

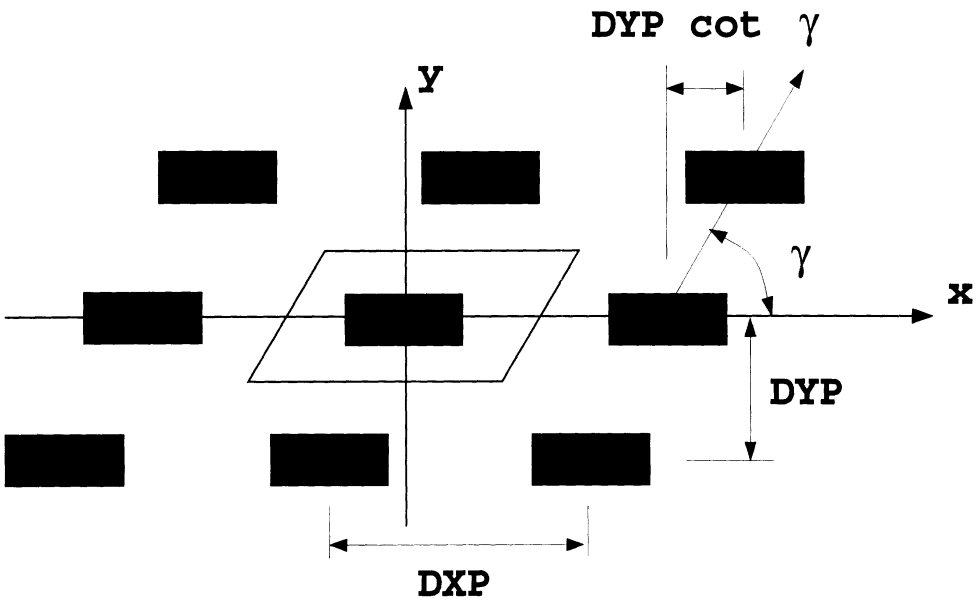
- Periodic Boundary Condition (PBC) for FE-part
- Periodic Green's Function (PGF) for BI-Part (from Houston)
- Metallic Patches in all Layers Possible (Antennas, FSS, ...)
- Probe Current Feeds and Plane Wave Excitation
- Lumped Impedances / Resisitive Sheets (to be done, already available in BRICK-Code)
- Metallic Backing (Antenna)



- BI on Top and Bottom Surface (Transmission, FSS)



Infinite Periodic Array



Periodicity Condition:

$$\mathbf{E}\,(x + m\,DXP + n\,DYP\,\cot\gamma,\; y\; n\,DYP) = \mathbf{E}\,(x,\; y)\; e^{-j\beta_x\,(m\,DXP\,+\,n\,DYP\,\cot\gamma)}\; e^{-j\beta_y\,n\,DYP}$$

$$\mathbf{H}\,(x + m\,DXP + n\,DYP\,\cot\gamma,\; y\; n\,DYP) = \mathbf{H}\,(x,\; y)\; e^{-j\beta_x\,(m\,DXP\,+\,n\,DYP\,\cot\gamma)}\; e^{-j\beta_y\,n\,DYP}$$

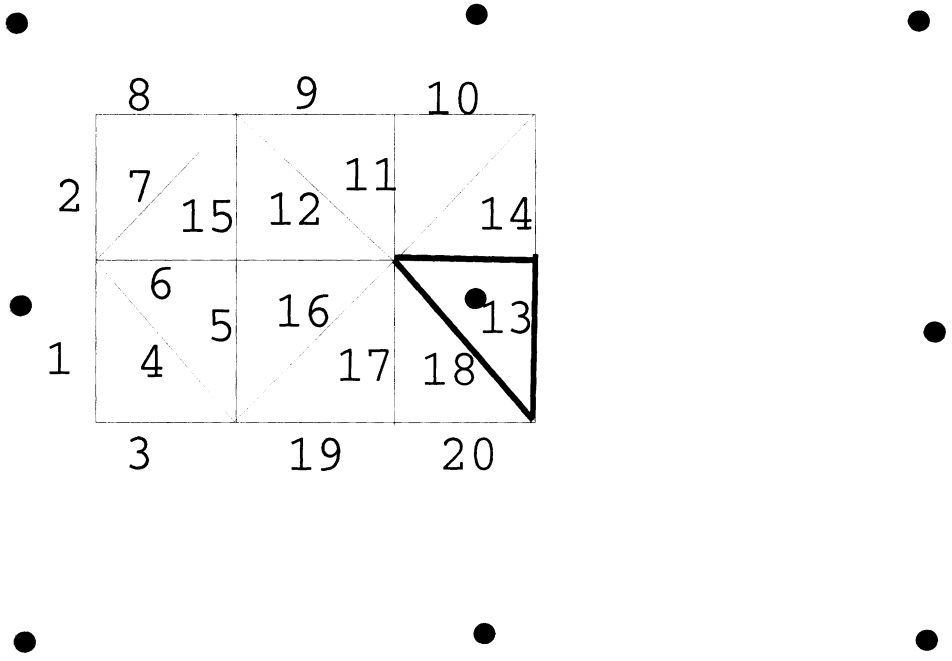
$$\beta_x = k_0 \sin\vartheta_0 \cos\varphi_0$$

$$\beta_x = k_0 \sin\vartheta_0 \sin\varphi_0$$

$\vartheta_0,\ \varphi_0$: Scan Angle

Periodic Free-Space Green's Function (BI) (from Houston)

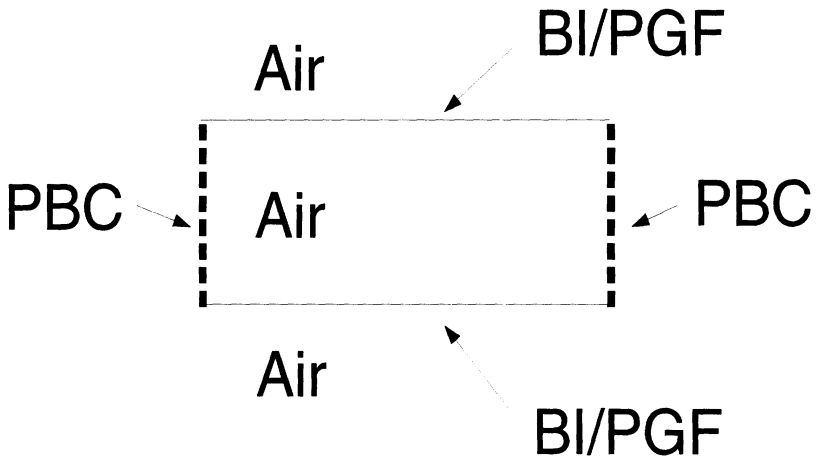
Unit Cell with Image Sources:



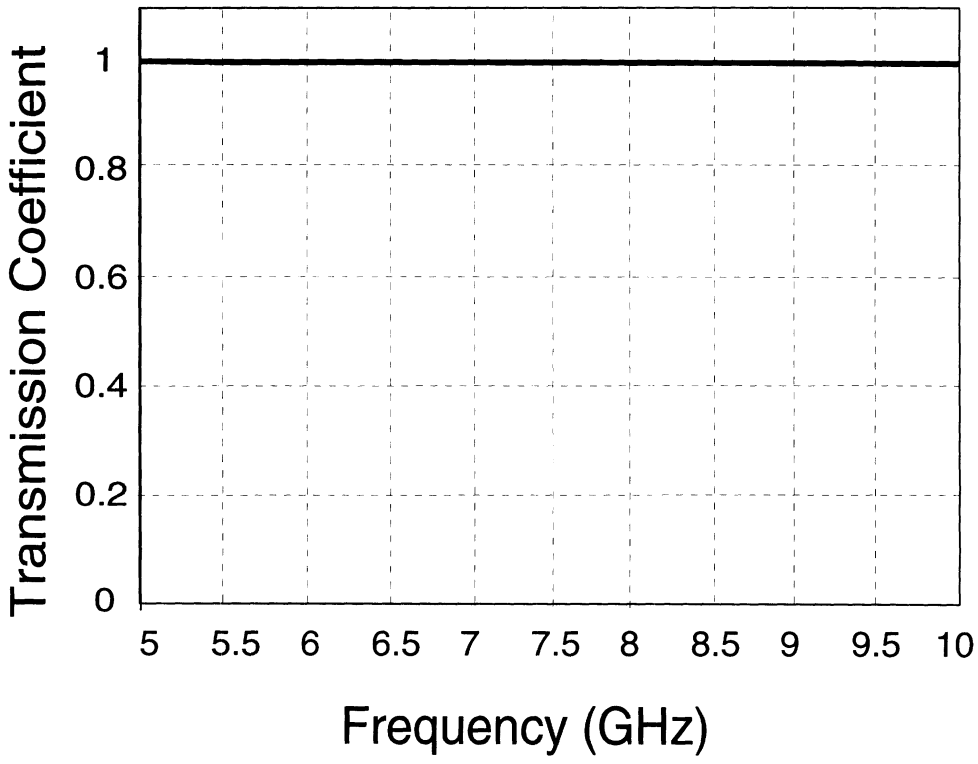
- Transformation of Matrix Elements analog to FE-Part
- Special Care for the Singular Image Sources
- Implementation of Plane-Wave Excitation
- Extraction of Transmission Coefficients and Radiation Patterns

Power Transmission Through an Air Layer

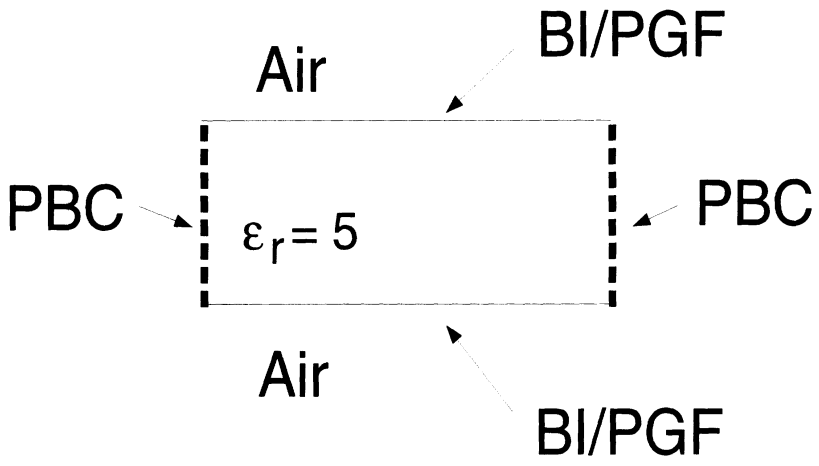
(sanity check)

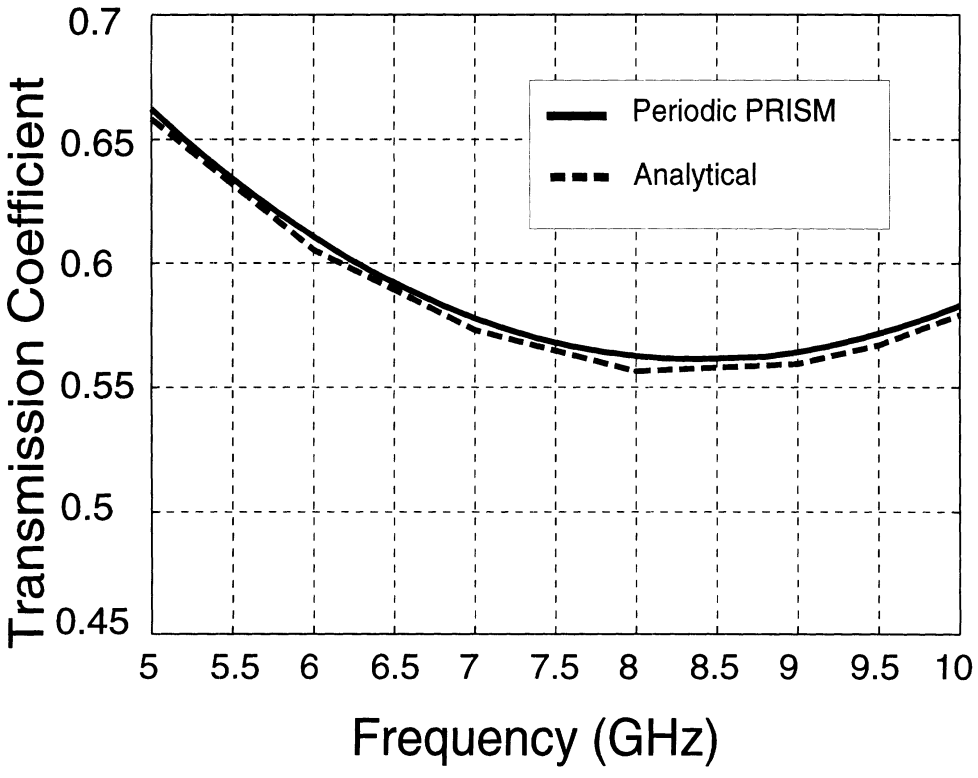


Transmission coefficient has to be 1.0

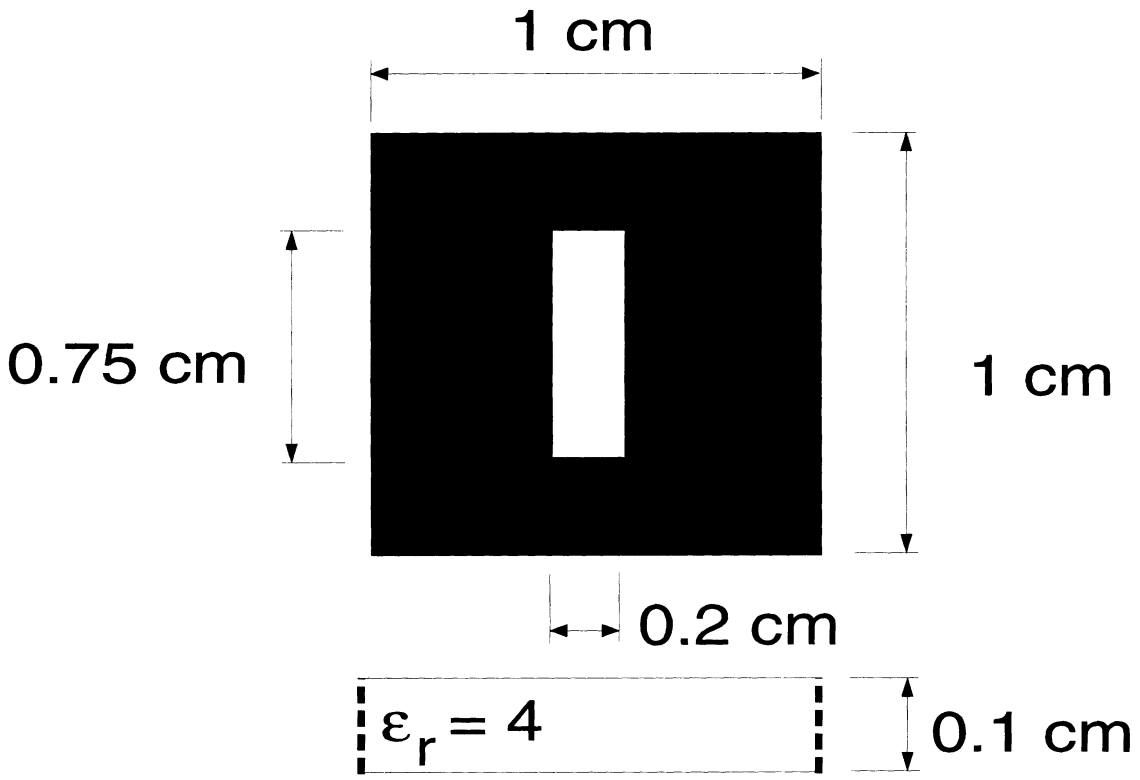


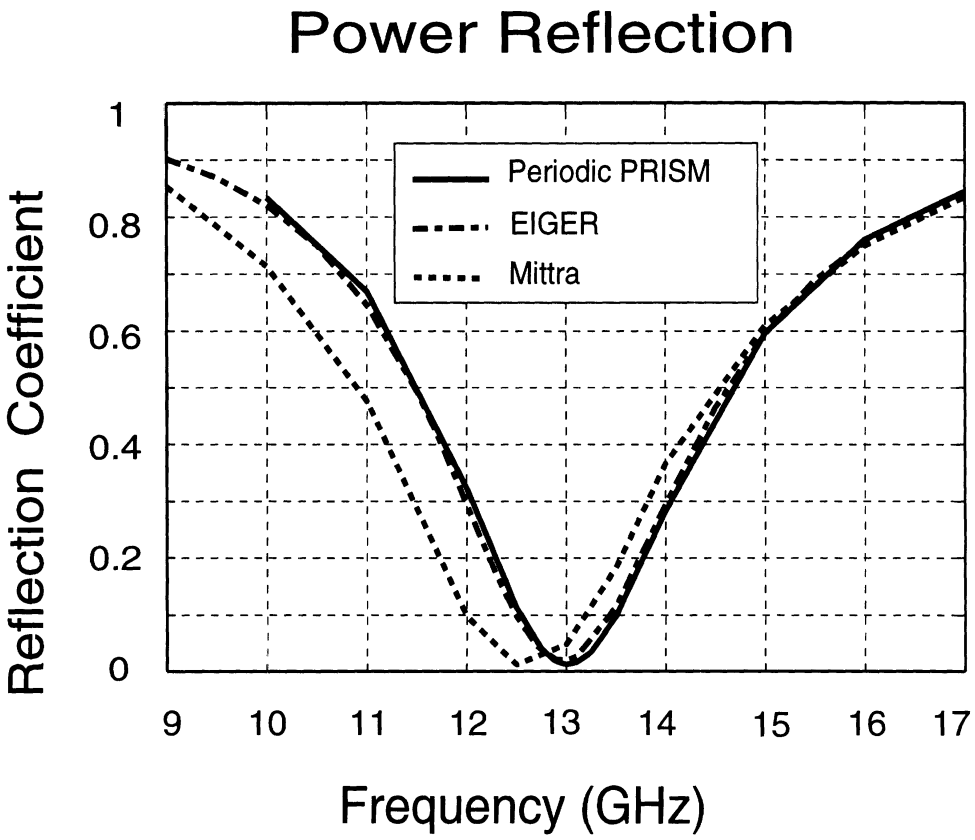
Power Transmission Through a Dielectric Layer





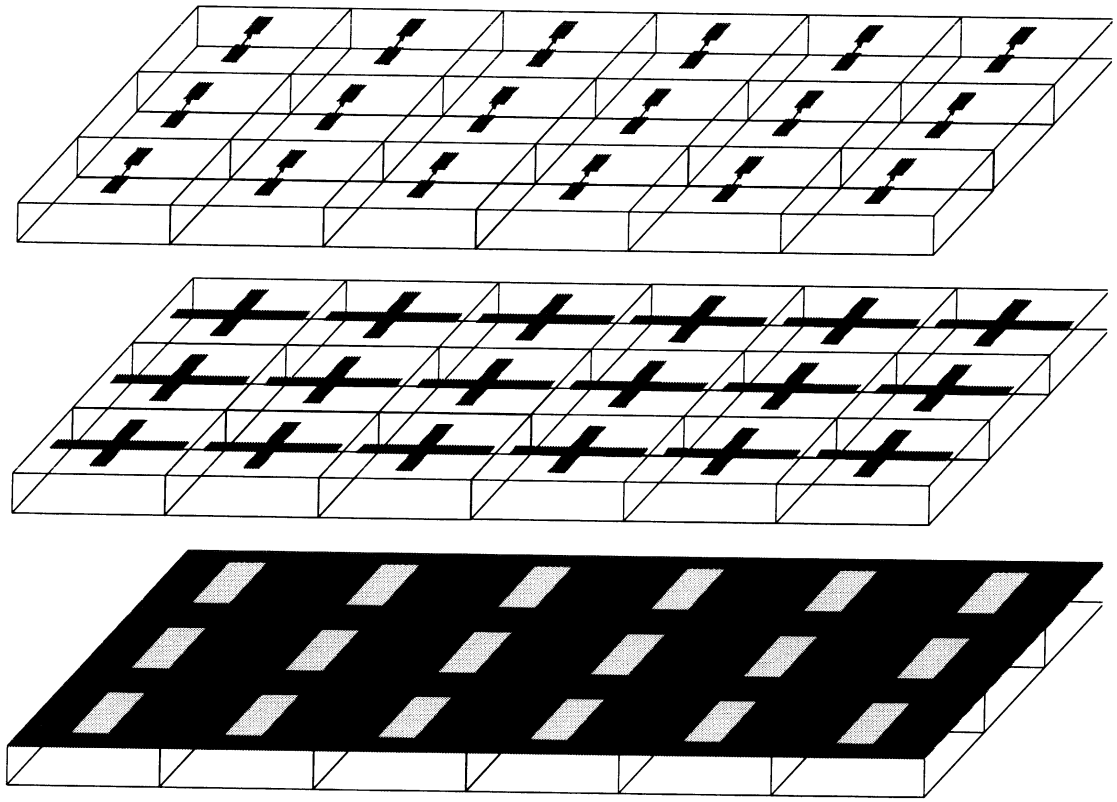
FSS Slot Array





APPENDIX 5: FSS-BRICK Presentation Given by Y. Erdemli on the May 30, 1997 review held in Ann Arbor

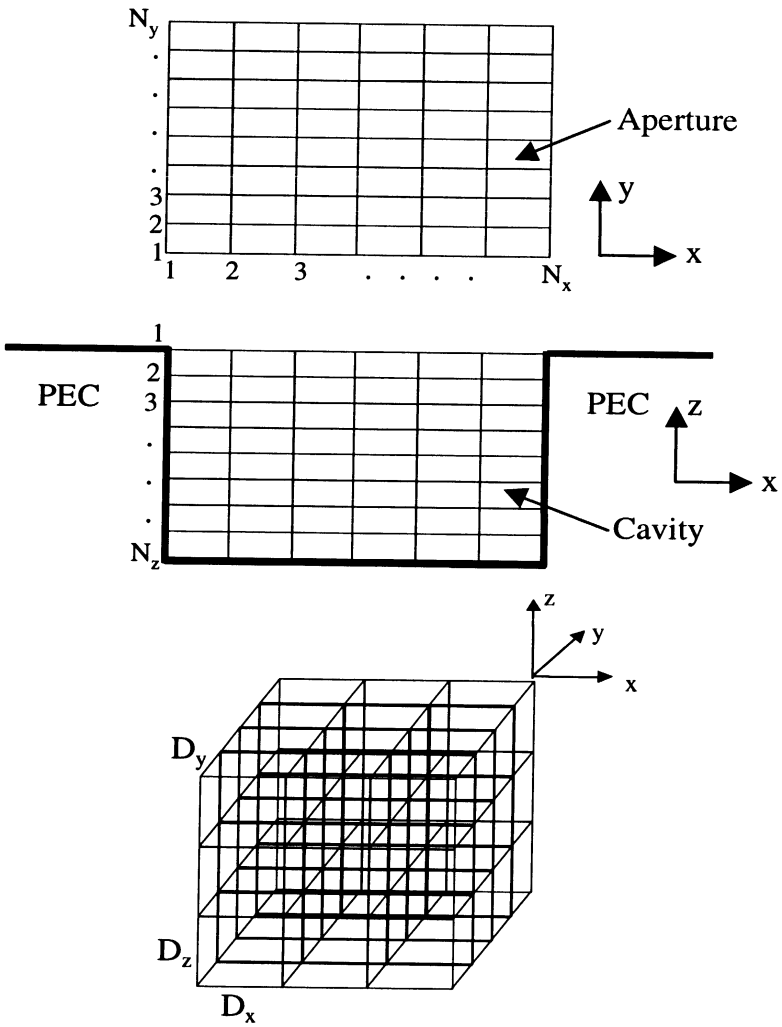
FSS-BRICK GEOMETRY DRIVER



FEATURES :

- Driver: Geometry I/O tool
- No need for external meshing package
- Mesh: Rectangular brick elements
- Driver requires the following inputs :
  - Computation type:
    - Bistatic scattering
    - Backscattering
    - Antenna radiation
  - Number of periodicity for commensurate FSS geometry (for both x- and y-direction)
  - Physical dimension of periodic FSS unit cell (all dimensions are in cm)
  - Sample size in x- and y-directions
  - Depth of cavity, and no. of layers along the cavity

Fss-Brick Code's Mesh



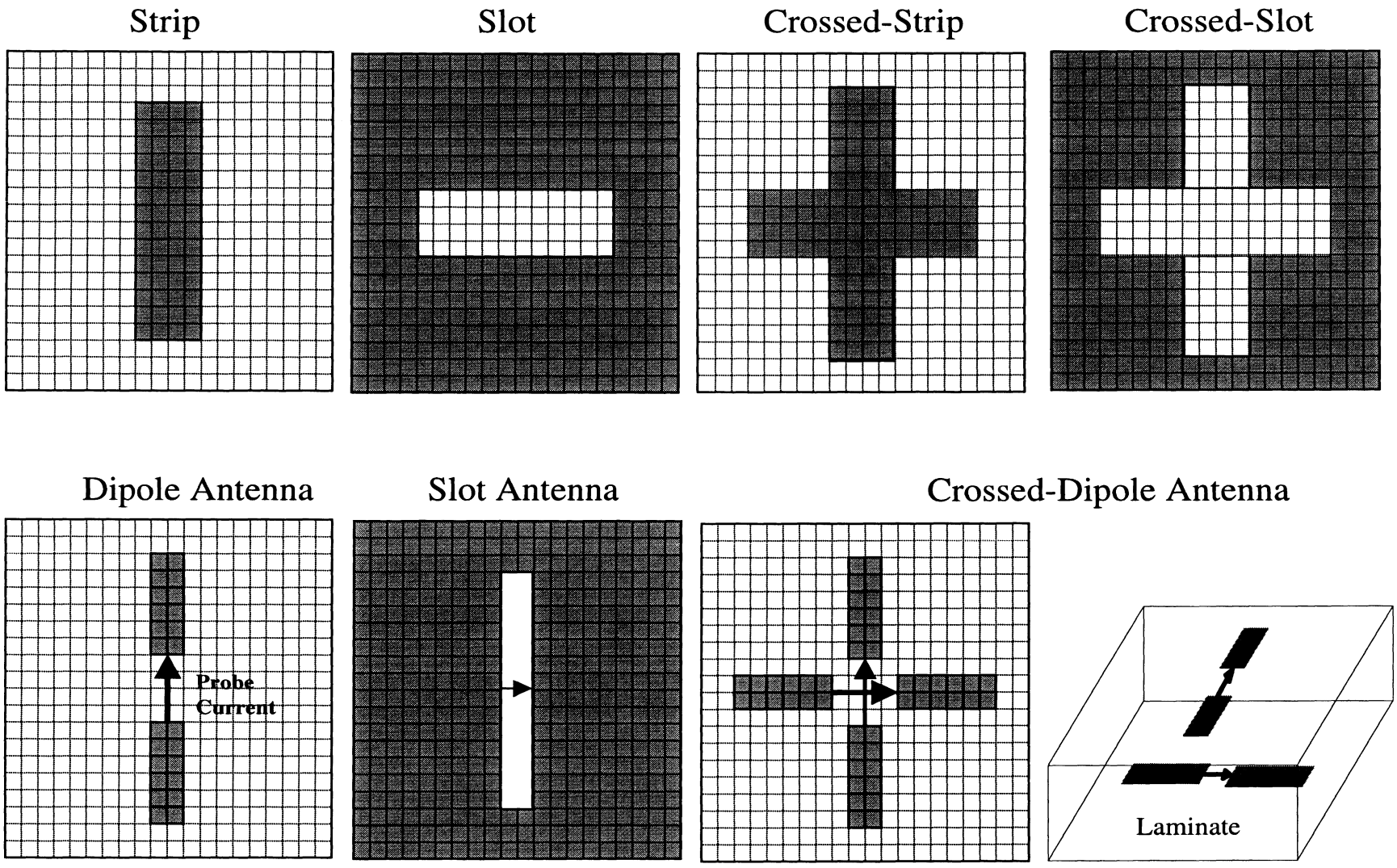
FEATURES (cont'd.):

- Thickness and electrical parameters (ε_r,μ_r) for each layer along the depth
- Operational frequencies in GHz
- Polarization, incidence, and scanning angles
- Type of antenna placed at the cavity aperture
    - Physical dimension of the center-fed antenna
    - Excitation information : Amplitude and phase of the probe-current (x- or y- directed)
- Physical dimension of each FSS layer's elements
- Geometry display of the antenna and FSS elements

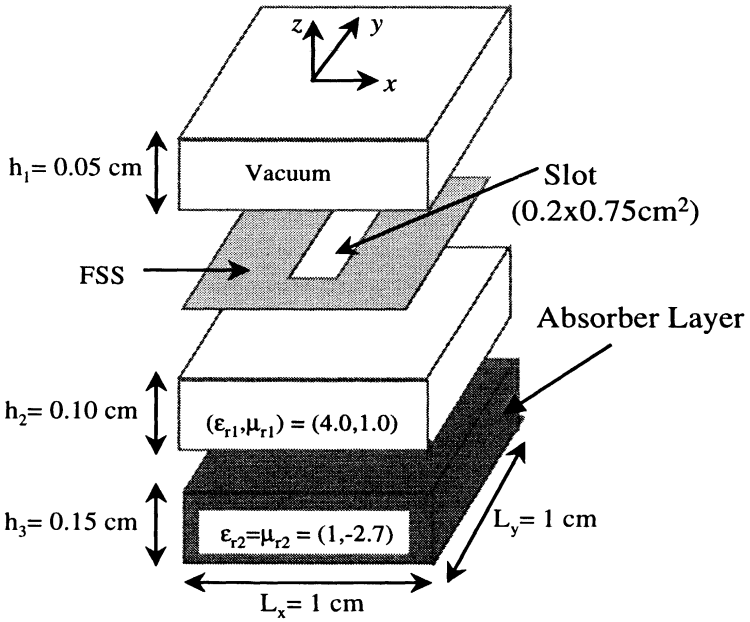
- Although they are not available in the driver as input options yet, the FSS-Brick code has also the following features :
    - z-directed probe feeds
    - Short-circuit pins
    - Lumped impedance loads
    - Resistive cards
    - Dielectric patches

- Transmission coefficient or Gain is output on screen.
- Similarly, a geometry driver for PRISM code will be developed.

FSS Element Types Available in BRICK Driver

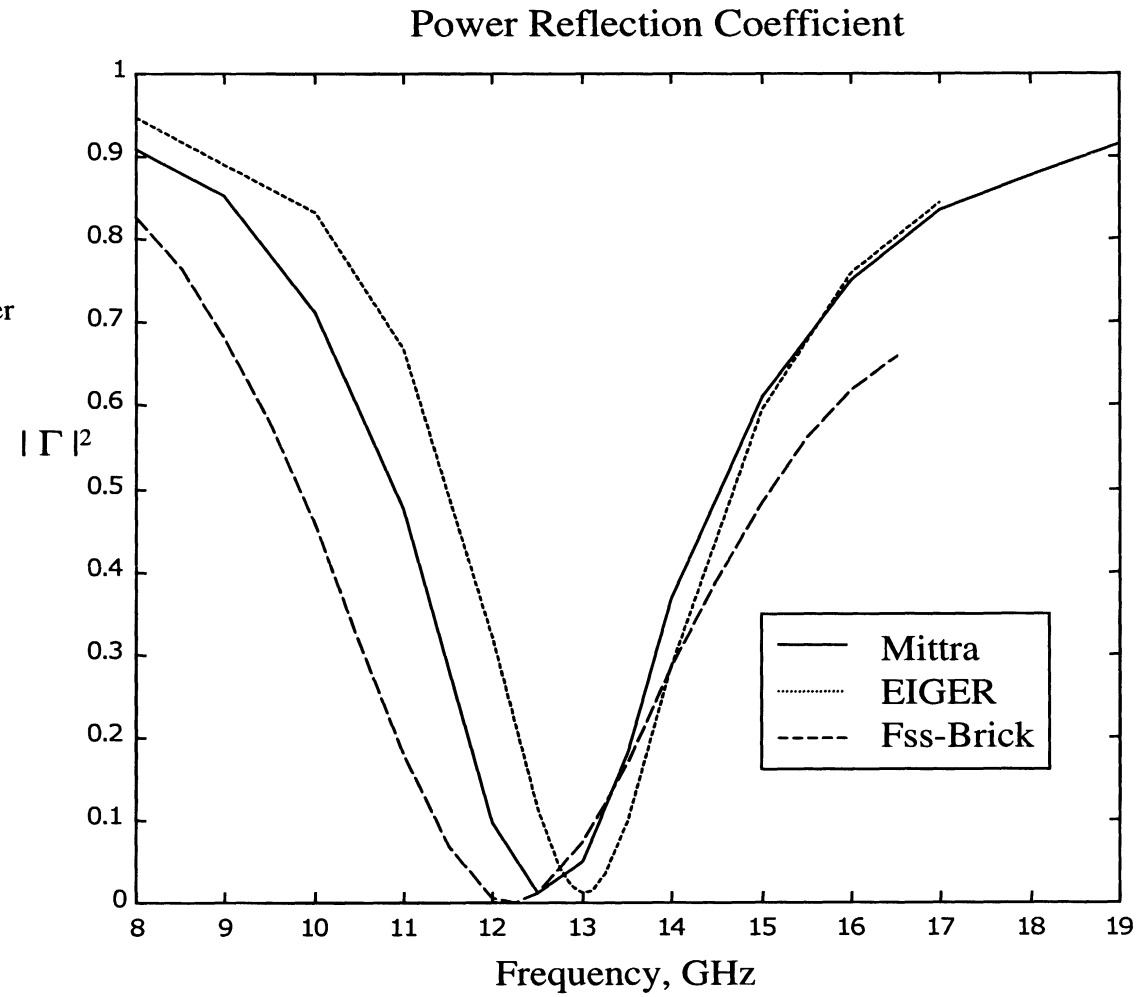


Single Layer FSS: Slot Element

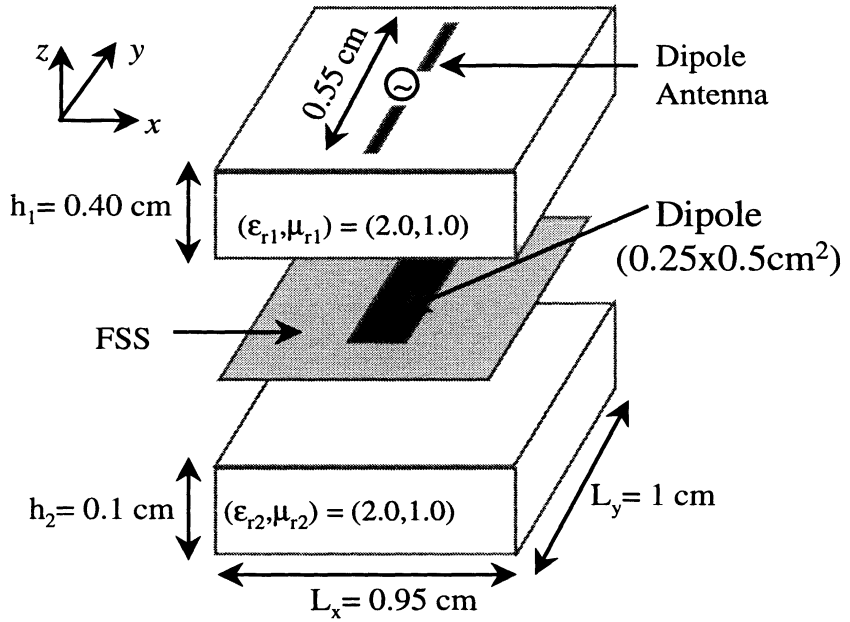


Fss-Brick Code :

TM incidence, θ^i = 1°, ϕ^i = 0°
f_r = 12.25 GHz
FEM sample size = 0.05 cm
FSS with 1-cell :
No. of unknowns = 5977
FFT pad size = 64x64
No. of iterations = 217
CPU time = 171.44 sec



Single Layer FSS with Antenna



Fss-Brick Code :

θ = 0-90°, ϕ = 0°

f_o = 18 GHz

FEM sample size = 0.05 cm

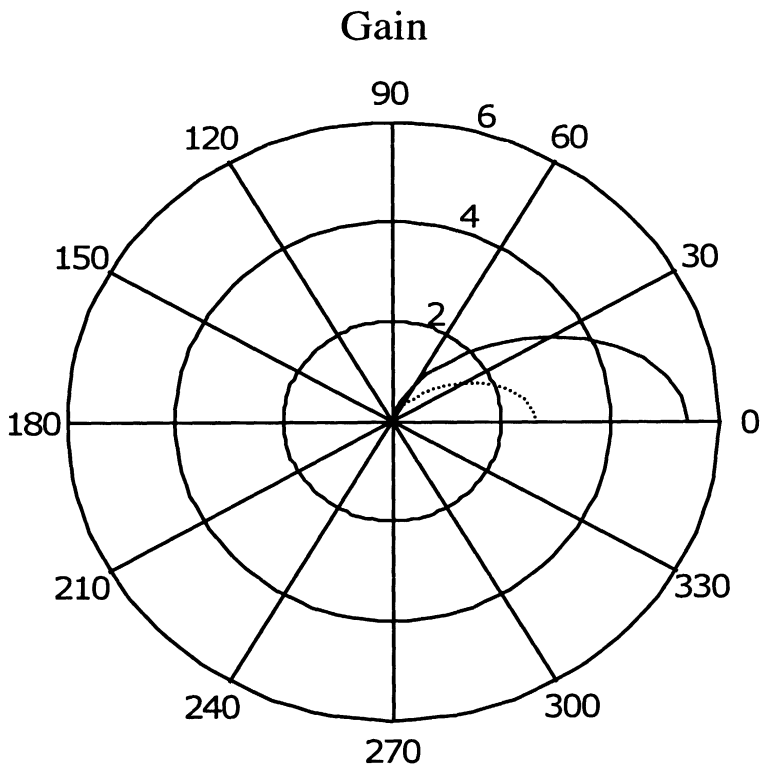
FSS with 1-cell :

No. of unknowns = 10461

FFT pad size = 64x64

No. of iterations = 306

CPU time = 224.63 sec



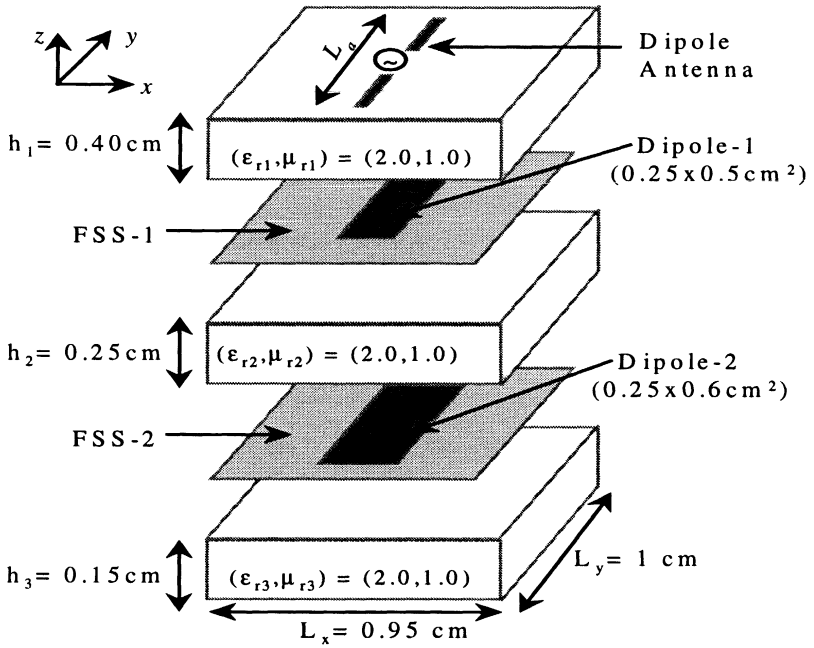
f_o= 16.0 GHz

Z_{in}= 1.7-j45.3 ohm

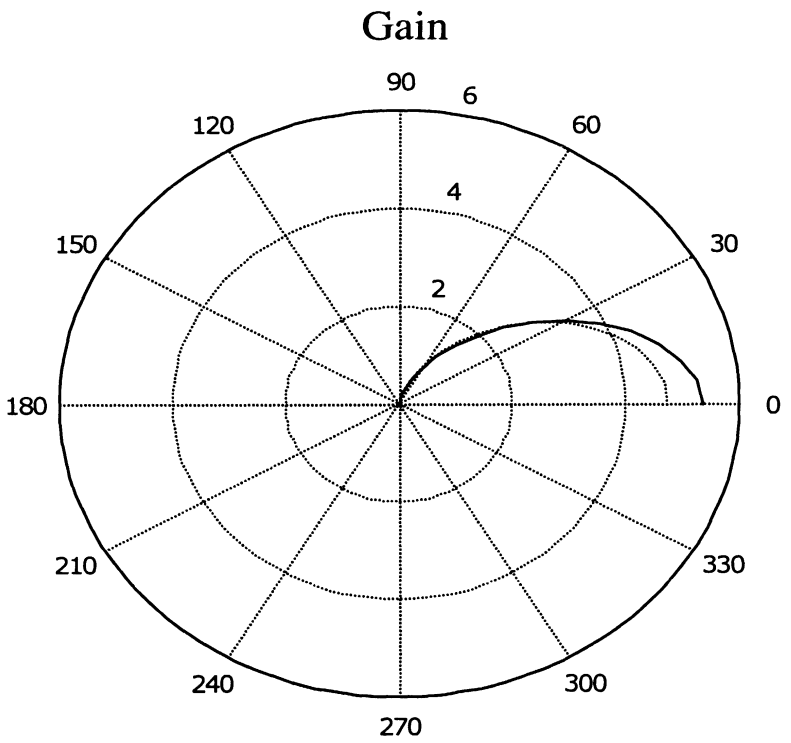
f_o= 18.0 GHz

Z_{in}= 135.3+j11.3 ohm

Double Layer FSS with Antenna



Fss-Brick Code :
θ = 0-90°, ϕ = 0°
f_o = 17.25 GHz
FEM sample size = 0.05 cm
FSS with 1-cell :
No. of unknowns = 16702
FFT pad size = 64x64
No. of iterations = 323
CPU time = 333.25 sec



..... f_o= 13.50 GHz
Z_{in}= 87.6+j109.9 ohm
L_a= 0.75 cm

—— f_o= 17.25 GHz
Z_{in}= 118.8+j23.2 ohm
L_a= 0.55 cm

POSSIBLE APPROACH FOR NON-COMMENSURATE FSS MODELING IN PRISM