035067-7-T

USERS MANUAL FOR FSS-PRISM

JULY 1998 REVISION

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

35067-7-T = RL-2487

PROJECT INFORMATION

PROJECT TITLE:	Hybrid Finite Element Design Codes for the SERAT Array
REPORT TITLE:	Users Manual for FSS-PRISM
U-M REPORT No.:	035067-7-T
CONTRACT	
START DATE:	October 1996
END DATE:	September 1998
DATE:	July 1998
SPONSOR:	Roland Gilbert SANDERS, INC, A Lockheed Martin Co. MER 24-1583 PO Box 868 Nashua, NH 030601-0868 Phone: (603) 885-5861 Email: roland.a.gilbert@lmco.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:	T. Eibert, Y. Erdemli, D. Giszczak and J. Volakis.

1. General Code Description

FSS_PRISM is used to analyze electromagnetic scattering and radiation characteristics of infinite periodic planar antenna arrays and frequency selective surface (FSS) configurations, as illustrated in Figure 1, with an arbitrary number of FSS layers (metallic patches or slots) and combinations of both. The code is capable of dealing with commensurate as well as non-commensurate structures. For improved computational efficiency, besides the general code version, a second version is optimized for commensurate structures only.

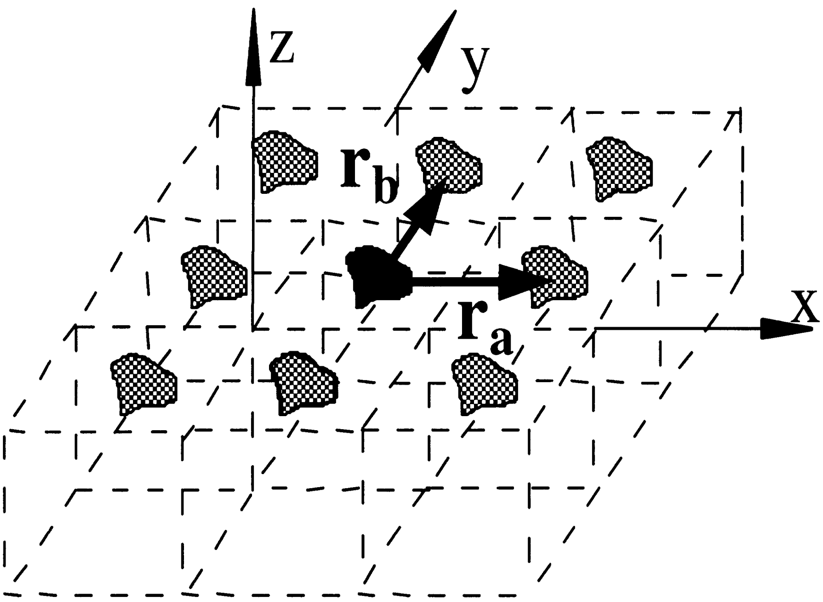


Figure 1: Periodic array configuration.

The hybrid finite element-boundary integral (FE-BI) method is used for field calculation. The finite element formulation is employed within the volumetric part and the boundary integral is used for terminating the mesh. The code works with prismatic elements in the FE-volume and triangular elements in the BI-surface. The prismatic elements can be right-angled as well as distorted (in the current FSS_PRISM versions only right-angled prisms are employed). First, the code generates triangular surface meshes with all geometrical adaptability for the individual layers while the volumetric FE-mesh is grown along the depth of the volume. The code has the option to deal with metal backed periodic configurations and with periodic configurations which are open at the top as well as at the bottom surface of the FE-mesh. In the first case, the BI is applied only on the top surface whereas in the latter, the BI-method is used to terminate both surfaces.

To model the infinite array problem, the periodicity condition for the fields in the infinite periodic array is employed using only one unit cell of the array. That is, within the FE-model of this unit cell, the periodic boundary condition (PBC) is enforced on the vertical walls of the mesh and on the boundary edges of the BI-surfaces where also an appropriate periodic Green's function (PGF) must be used. A typical FE mesh for a commensurate unit cell is shown in Figure 2. For modeling non-commensurate structures, the individual layer periodicities are decoupled as illustrated in Figure 3. Of course, this is only an approximate model but its accuracy can arbitrarily be improved by grouping several cells in the individual layers. For further details of the analysis and the theory of the formulation the user is referred to [1-10].

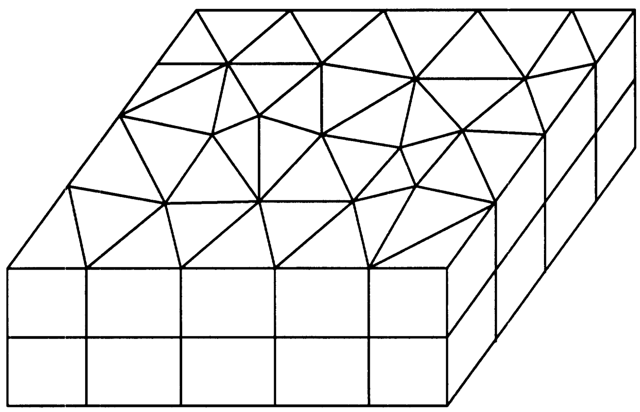


Figure 2: Prismatic FE mesh for commensurate unit cell.

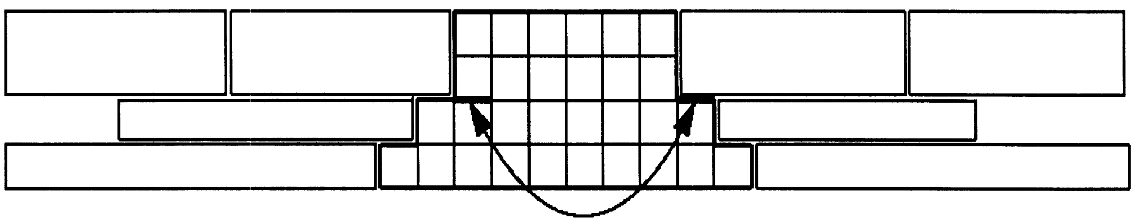


Figure 3: Decoupling of individual layer periodicities for non-commensurate structures.

FSS_PRISM is written in Fortran 77 and has been verified to run on PC's and Unix platforms for HP, Sun, SGI workstations. In this manual, we describe input/output (I/O) operations. Also, several computational examples are presented to demonstrate usage and performance of the code. The general code version is designated as FSS_PRISM_NC indicating that non-commensurate structures can be handled while the version optimized for commensurate problems is designated as FSS_PRISM_C.

2. Features of the FSS_PRISM Driver

Each FSS_PRISM version includes two separate codes: the geometry driver (FORTRAN source file "Driver.f") and the analysis (main) code comprised of the FORTRAN source files "femap.f, elmmat.f, bint.f, solver.f, aim.f, invers.f cfft2-D.f". In addition, there is a parameter file "fema.dm1" which is used by both parts of the code and defines the array dimensions for compilation. The driver is an I/O tool that enables users to enter the necessary geometry and excitation information for antenna and/or FSS analysis. Using the geometrical input data, the driver generates the antenna mesh consisting of prismatic elements without a need for an external meshing package. The surface mesh for each FSS layer interface (2-D cross-section), and the whole geometry in 3-D can be viewed using MatLab. Thus, the user can visualize and check the geometry before running the analysis code. The PRISM driver requires the physical dimensions of the geometry in centimeter, angles in degrees, and frequency in GHz. The driver stores the necessary mesh and excitation information in two data files to be used by the analysis code. After executing the driver code, the prism code is run to compute input impedances of radiating elements and/or reflection and transmission coefficients of the corresponding structure. Of course, the input data files for the actual code can also be generated without the delivered driver (using external or specialized gridding packages). This allows the code to be used for more general structures than those supported by the driver.

The driver can be used to generate the input files for the NC version as well as the C version of the code (selected by flag) and has the following I/O features:

1) All input data provided by the user is stored in a file with a pre-specified name so that it is convenient to rerun the code using the input file with the same data or modifications without reentering the entire input data set.

2) The user can choose either transmission/reflection coefficient evaluation of FSS structures or radiation computations for antenna configurations on top of multilayered FSS structures. After running the analysis code, for FSS analysis, the transmission and reflection coefficients of the structure for a given scan angle and frequencies are displayed on the screen and stored in corresponding output files ("Reflect", "Trans"). Similarly, plane wave amplitudes in scan direction and input impedances (in Ohm) (file "Imp") are displayed and stored for antenna analysis.

3) FSS_PRISM is capable of analyzing commensurate as well as non-commensurate infinite periodic arrays. The FEM sample size and dimensions for each unit cell must be specified in centimeter. The driver also asks for the thickness or depth of the structure, including the number of layers and their material properties. In addition, beginning, end, and incremental frequencies (in GHz) and angles (in degrees) are specified by the user.

4) The code can employ the adaptive integral method (AIM) to speed up the BI computation, especially for large problems. Therefore, the driver asks for AIM parameters to be set for optimal efficiency.

5) Strip dipoles, slots, crossed-strip, and crossed-slot antenna or FSS elements are available in the driver as primitives. These elements can be placed at any location within the unit cell. The physical dimensions and the lower-left corner location of elements must be specified for each layer interface. It is assumed that the lower-left corner of each layer interface is the origin (0.,0.). Moreover, the user can generate other types of elements by specifying the size and location of rectangular metallic patches that would form the specific FSS element.

6) The driver allows horizontal (*x*- or *y*-directed) as well as vertical (*z*-directed) probe feeds at any node location within the structure where the user can specify locations, amplitudes, and phases of the probe currents.

7) Vertical short-circuit pins can be placed in all layers by specifying their (*x*,*y*) coordinates.

8) Vertical (z-directed) lumped impedance loads can be situated in each layer; their admittance values (in 1/Ohm) can be specified.

9) Horizontally placed lumped impedance loads are also available in the driver. The user can place x- or y- directed loads on each FSS layer as well as at the aperture where antenna elements may reside. The admittance values and locations of the loads are entered by the user.

10) As another option, resistive cards can be selected. The cards can be in all FSS layers and in the aperture. Surface conductance (1/Ohm), location, and size of the cards can be specified.

3. Running the Code

Before compiling the driver source file "Driver.f", the dimension parameters "NdmTri" (maximum number of triangles), "NdmSEd" (maximum number of edges), "NdmLay" (number of prism layers), and "NdmSNo" (maximum number of nodes expected in the final mesh) in the file called "fema.dm1" should be adjusted properly. Next the driver code is compiled by typing

f77 Driver.f -o *executable_filename1*

on UNIX platforms. To run the code, just type *executable_filename1* and follow the instructions for standard I/O operations. If an input file for the driver is already available, the user only needs to enter the name of that file; otherwise, the user is asked to enter the necessary input information one by one on the screen. All information is also stored in a user-specified input file allowing simplified reruns. After executing the driver, two data files "Data1" and "Data2" are generated to be used by the analysis code.

The main code can be compiled using the delivered "Makefile" for UNIX systems. Typing "make" at the prompt starts the compiler for the individual source files and finally links them together to a executable called "FSS_PRISM_C" or "FSS_PRISM_NC", respectively. For optimal use of the "Makefile" it might be necessary to adopt compiler and linker options, for instance with respect to code optimization, to the used system. On PC based systems, typically a project is defined including all the necessary source files. Before the code is compiled the array dimensions defined in the parameter file "fema.dm1" must be set sufficiently large. The meaning of the different parameters is explained in the file "fema.dm1" and also later in this manual. To make sure that no array dimension is exceeded, FORTRAN compilers usually have a compiler option enabling array range checking. Also, most of the array dimensions are checked during run time and the code will terminate with an error message.

After executing the prism code, the output data is displayed on the screen and also stored in the output files "Imp", "Trans", and "Reflect", whose formats will be described later in this manual. Sometimes it might be useful to redirect the output from the screen into an ASCII file by typing "> out.txt" at the end of the command (on Unix), viz.

```
FSS_PRISM_... > out.txt
```

This causes the screen output to be written in the file "out.txt".

4. I/O Operations

After starting the driver code, the user must specify the input data format. If an input file is available, the user is only asked to enter the name of that file. Otherwise, the user must enter all input data one by one on the screen. However, in this case all input data is stored in a file specified by the user. The driver finally generates two data files, namely "Data1" and "Data2" to be used as input files for the analysis code. The data format of these files is described later in this section.

In case of entering the input data on the screen, the user will be asked specifically for all necessary information where the questions are self-explanatory. However, if the user wants to generate an input file by himself rather than entering each input on the screen, it is necessary to know the input file format of the geometry driver. For this purpose, all input variables appearing in the input file are described in the next subsection.

4.1 Input File Format

Each line in the input file includes at least one entry and the entries must be separated by either a blank or a comma. It is important to pay attention to the type of the individual variables, whether it is a real, integer or complex variable, and specify it accordingly; otherwise, the driver may give error message and terminate. All entries specified in an input file are described below and illustrated by an example. Note that the variables are designated by its representative names.

`1)` Problem type and configuration: **`ITYP,IBOT,ICOM`**

`ITYP` (integer): Problem = 1 → Radiation *or* = 2 → Scattering

`IBOT` (integer): Bottom aperture = 1 → Open *or* = 2 → Closed (metal-backed)

`ICOM` (integer): = 1 → Commensurate *or* = 2 → Non-commensurate

Example: `1,2,2` [Radiation problem; Non-commensurate configuration with metal-backed bottom surface]

`2)` Polarization type of incident plane wave and scan direction: **`IPOL,PHI,THETA`**

`IPOL` (integer): Polarization = 0 → No plane wave excitation, Radiation (`ITYP=1`)
Polarization = 1→ TE-pol *or* = 2 → TM-pol (Scattering: `ITYP=2`)
`PHI,THETA` (real): Array scan angles (degree)

Example: `1,0.,90.` [TE-polarization; $\phi=0^{\circ}$, $\theta=90^{\circ}$]

`3)` Frequency of operation: **`FRBEG,FREND,FRSTP`**

`FRBEG,FREND,FRSTP` (real) : Beginning, end, and incremental frequencies (GHz)

Example: `1.,10.,0.25` [f_1=1 GHz, f_2=10 GHz, δf =0.25 GHz]

`4)` AIM computation parameters: **`NSB1,NFBX,NFBY`**

`NSB1` (integer): Number of samples per dimension in the regular grid (Commensurate: `ICOM=1`) *or* in the bottom regular grid (Non-commensurate: `ICOM=2`)
`NFBX,NFBY` (integer): Nearfield threshold in the regular *or* in the bottom regular grid for the x- and y-directions

Only for non-commensurate case, additionally: **`NST1,NFTX,NFTY`**

`NST1` (integer): Number of samples per dimension in the top regular grid

`NFTX,NFTY` (integer): Nearfield threshold in the top regular grid for x- and y-directions

Example: `50,18,18` [Commensurate case: `NSB1=50`, `NFBX=NFBY=18`]

The regular AIM grid overlays the possibly irregular triangular surface mesh. In the commensurate case, the surface mesh is equal throughout the whole structure and therefore the BI matrices on the top and bottom surfaces must be calculated only once.

Similarly, only one regular AIM grid is needed. In the non-commensurate case, the surface meshes on the top and bottom BI terminations can have different extent and the employed Green function grid can be different as well. Therefore, different regular AIM grids can be defined on the top and bottom surfaces. Good AIM performance can typically be achieved by three to four regular grid points per triangle side length in the original mesh. In this case, the nearfield thresholds in x- and y-directions should not be smaller than 15 (typically 18). For instance, if the extent of the original mesh in x- and y-directions is 10 cm and the average side length of the triangles is 0.5 cm, a total number of 60 regular grid points per dimension would lead to three regular grid points per triangle side length. In the current implementation of the code, a 2-D FFT algorithm is used which requires the same number of elements in both dimensions. Therefore, the number of samples per dimension equally applies to x- and y-directions.

5) <u>FSS unit cell size and FEM sample size:</u> **`XL,YL,DX,DY`**

`XL,YL` (real): FSS unit cell size (`ICOM=1`) *or* Largest unit cell size (`ICOM=2`) (cm)

`DX,DY` (real): FEM sample size (cm)

Example: `2.,2.,0.05,0.05` [FSS unit cell size: 2x2 cm^2,
FEM sample size: 0.05x0.05 cm^2]

6) <u>Depth of structure and number of layers:</u> **`ZL,NZ`**

`ZL` (real): Depth of structure (cm)

`NZ` (integer): Number of layers along the depth

Example: `1.,3` [3-layered structure with 1 cm in depth]

7) <u>FSS unit cell size:</u> **`FXL,FYL`**

Only if `ICOM=2` (Non-commensurate case), for each layer from the bottom to the top:

`FXL,FYL` (real): x-y dimensions of FSS unit cell in cm (each layer's top surface)

Example: `2.,2.` [FSS unit cell size for Layer # 1 (bottom layer): 2x2 cm^2]
`1.5,1.25` [FSS unit cell size for Layer # 2: 1.5x1.25 cm^2]
`1.25,1.` [FSS unit cell size for Layer # 3 (top layer): 1.25x1 cm^2]

8) <u>Layer parameters:</u> **`DT,EPS,MU`**

For each layer from the bottom to the top:

DT (real): Layer thickness (cm)

EPS, MU (complex): Relative permittivity and permeability

Example:	`0.5,(1.,0.),(1.,0.)`	[t_1=0.50 cm, $\varepsilon_{r1}=\mu_{r1}$=1.+j0.]
	`0.25,(2.,0.),(1.,0.)`	[t_2=0.25 cm, ε_{r2}=2.+j0., μ_{r2}=1.+j0.]
	`0.25,(1.,0.),(1.,-0.05)`	[t_3=0.25 cm, ε_{r3}=1.+j0., μ_{r3}=1.-j0.05]

9) FSS elements:

For each layer from the bottom to the top:

Flag for FSS elements: **IYNFS**

IYNFS (integer): = 1 → FSS elements on the top of layer *or* = 2 → No FSS elements

If *only* IYNFS=1:

Type of FSS elements and number of sections: **NFSEL,NMET**

NFSEL (integer): FSS element type = 1 → Strip *or* = 2 → Slot *or* = 3 → Crossed-strip *or* = 4 → Crossed-slot *or* = 5 → User-specified element

NMET (integer): Number of sections = 1 (NFSEL=1,2) *or* = 2 (NFSEL=3,4) *or* = user-specified (NFSEL=5)

Size of each section: **DPX,DPY**

DPX, DPY (real): *x-y* dimensions of each section for each FSS element (cm)

Location of each section: **X1,Y1**

X1, Y1 (real): Lower-left corner coordinate of each section in cm (for each layer, the origin is assumed to be the lower-left corner of the corresponding unit cell)

Example:	`1`	[FSS elements exist on Layer # 1 (bottom layer)]
	`1,1`	[FSS element type: Strip with one section only]
	`0.5,0.75`	[Size of Strip element: 0.5x0.75 cm^2]
	`0.25,0.35`	[(x,y) coordinate of Strip element in cm]
	`2`	[There is no FSS element on Layer # 2]
	`1`	[FSS elements exist on Layer # 3 (top layer)]
	`3,2`	[FSS element type: Crossed-strip with two sections]
	`0.5,0.5`	[Size of 1^{st} section: 0.5x0.5 cm^2]
	`0.25,0.35`	[(x,y) coordinate of 1^{st} section in cm]
	`0.75,0.75`	[Size of 2^{nd} section: 0.75x0.75 cm^2]
	`0.35,0.25`	[(x,y) coordinate of 2^{nd} section in cm]

10) Resistive Cards (R-cards):

For each layer from the bottom to the top:

Number of R-cards: **NRES**

NRES (integer): Number of R-cards placed on each layer's top surface

Size of each R-card: **RCX,RCY**

RCX, RCY (real): *x-y* dimensions of each R-card in cm

Location of each R-card: **RX1,RY1**

RX1, RY1 (real): Lower-left corner coordinate of each R-card in cm (for each layer, the origin is assumed to be the lower-left corner of the corresponding unit cell)

Surface Conductance of each R-card: **SCN**

SCN (complex): Surface conductance of the R-card (1/Ohm)

Example:	0	[There is no R-card on Layer # 1 (bottom layer)]
	0	[There is no R-card on Layer # 2]
	2	[There are two R-cards on Layer # 3 (top layer)]
	0.5,0.5	[Size of R-card # 1: 0.5x0.5 cm^2]
	0.25,0.25	[(*x*,*y*) coordinate of R-card # 1 in cm]
	(100.,0.)	[Conductance value of R-card # 1: 100+j0 (1/Ohm)]
	0.25,0.25	[Size of R-card # 1: 0.25x0.25 cm^2]
	0.25,0.25	[(*x*,*y*) coordinate of R-card # 2 in cm]
	(10.,-0.5)	[Conductance value of R-card # 2: 10-j0.5 (1/Ohm)]

11) Lumped Impedance Loads:

x-directed loads (*x*-loads):

For each layer from the bottom to the top:

Number of *x*-loads: **NXD**

NXD (integer) : Number of *x*-loads located on each layer's top surface

Location of each *x*-load: **X1,Y1,X2,Y2**

X1, Y1 (real) : Beginning coordinate of each *x*-load in cm

X2, Y2 (real) : End coordinate of each *x*-load in cm

Note that an *x*-load is located between its beginning and end coordinates, and also Y1 should be equal to Y2 .

Admittance of each *x*-load: **AXLD**

AXLD (complex): Admittance value of each *x*-load (1/Ohm)

Example: 0 [There is no *x*-load on Layer # 1]

`1`	[There is only one x-load on Layer # 2]
`0.1,0.1,0.3,0.1`	[$(x_1,y_1) = (0.1,0.1)$ and $(x_2,y_2) = (0.3,0.1)$ cm]
`(150.,0.)`	[Admittance of the x-load: 150+j0 (1/Ohm)]
`0`	[There is no x-load on Layer # 3]

<u>y-directed loads (y-loads) :</u>

For each layer from the bottom to the top:

<u>Number of y-loads:</u> **`NYD`**

`NYD` (integer) : Number of y-loads located on each layer's top surface

<u>Location of each y-load:</u> **`X1,Y1,X2,Y2`**

`X1,Y1` (real) : Beginning coordinate of each y-load in cm

`X2,Y2` (real) : End coordinate of each y-load in cm

Note that a y-load is located between its beginning and end coordinates, and also `X1` should be equal to `X2`.

<u>Admittance of each y-load:</u> **`AYLD`**

`AYLD` (complex) : Admittance value of each y-load (1/Ohm)

Example:	`0`	[There is no y-load on Layer # 1]
	`1`	[There is only one y-load on Layer # 2]
	`0.1,0.3,0.1,0.3`	[$(x_1,y_1) = (0.1,0.1)$ and $(x_2,y_2) = (0.3,0.1)$ cm]
	`(10.,10.)`	[Admittance of the y-load: 10+j10 (1/Ohm)]
	`0`	[There is no y-load on Layer # 3]

<u>z-directed loads (z-loads) :</u>

For each layer from the bottom to the top:

<u>Number of z-loads:</u> **`NZD`**

`NZD` (integer) : Number of z-loads located in each layer

<u>Location of each z-load:</u> **`X1,Y1`**

`X1,Y1` (real) : (x,y) coordinate of each z-load within the unit cell (cm)

<u>Admittance of each z-load:</u> **`AZLD`**

`AZLD` (complex) : Admittance value of each z-load (1/Ohm)

Example:	`1`	[There is only one z-load in Layer # 1]
	`0.15,0.3`	[Location of the load: $(x,y) = (0.15,0.3)$ cm]
	`(10.,1.5)`	[Admittance of the z-load: 10+j1.5 (1/Ohm)]
	`1`	[There is only one z-load in Layer # 2]

`0.15,0.3`	[Location of the load: (x,y) = (0.15,0.3) cm]
`(10.,1.5)`	[Admittance of the z-load: 10+j1.5 (1/Ohm)]
`1`	[There is only one z-load in Layer # 3]
`0.15,0.3`	[Location of the load: (x,y) = (0.15,0.3) cm]
`(10.,1.5)`	[Admittance of the z-load: 10+j1.5 (1/Ohm)]

Note that a cascade of three vertical loads located at (x,y) = (0.15,0.3) are formed along the depth of the structure in the above example.

12) <u>Short-Circuit Pins:</u>

For each layer from the bottom to the top:

<u>Number of pins:</u> **`NPIN`**

`NPIN` (integer) : Number of short-circuit pins inserted in each layer

<u>Location of each pin:</u> **`X1,Y1`**

`X1,Y1` (real) : (x,y) coordinate of each pin within the unit cell (cm)

Example:

`1`	[There is only one pin in Layer # 1 (bottom layer)]
`0.3,0.3`	[Location of the load: (x,y) = (0.3,0.3) cm]
`1`	[There is only one pin in Layer # 2]
`0.3,0.3`	[Location of the pin: (x,y) = (0.3,0.3) cm]
`1`	[There is only one pin in Layer # 3 (top layer)]
`0.3,0.3`	[Location of the pin: (x,y) = (0.3,0.3) cm]

The top and bottom surfaces of the structure can be short-circuited by placing pins at the same (x,y) coordinate in each layer as done in the above example.

13) <u>Probe-Current Feeds:</u>

<u>x-directed probe-current feeds (x-feeds) :</u>

For each layer from the bottom to the top:

<u>Number of x-feeds:</u> **`NXFED`**

`NXFED` (integer) : Number of x-feeds located on each layer's top surface

<u>Location of each x-feed:</u> **`X1,Y1,X2,Y2`**

`X1,Y1` (real) : Beginning coordinate of each x-feed in cm

`X2,Y2` (real) : End coordinate of each x-feed in cm

Note that an x-feed is located between its beginning and end coordinates, and also `Y1` should be equal to `Y2`.

Amplitude and phase of each x-feed: **XAMP,XPHS**

`XAMP` (real) : Amplitude of each x-feed (Amp)

`XPHS` (real) : Phase of each x-feed (degree)

Example:

`0`	[There is no x-feed on Layer # 1]
`0`	[There is no x-feed on Layer # 2]
`1`	[There is only one x-feed on Layer # 3]
`0.2,0.1,0.5,0.1`	[$(x_1,y_1) = (0.2,0.1)$ and $(x_2,y_2) = (0.5,0.1)$ cm]
`1.,45.`	[Probe-current: $I_{x\text{-feed}} = 1.\ e^{j\pi/4}$ Amp]

y-directed probe-current feeds (y-feeds) :

For each layer from the bottom to the top:

Number of y-feeds: **NYFED**

`NYFED` (integer) : Number of y-feeds located on each layer's top surface

Location of each y-feed: **X1,Y1,X2,Y2**

`X1,Y1` (real) : Beginning coordinate of each y-feed in cm

`X2,Y2` (real) : End coordinate of each y-feed in cm

Note that an y-feed is located between its beginning and end coordinates, and also `X1` should be equal to `X2`.

Amplitude and phase of each y-feed: **YAMP,YPHS**

`YAMP` (real) : Amplitude of each y-feed (Amp)

`YPHS` (real) : Phase of each y-feed (degree)

Example:

`0`	[There is no y-feed on Layer # 1]
`0`	[There is no y-feed on Layer # 2]
`1`	[There is only one y-feed on Layer # 3]
`0.3,0.1,0.3,0.4`	[$(x_1,y_1) = (0.3,0.1)$ and $(x_2,y_2) = (0.3,0.4)$ cm]
`1.,0.`	[Probe-current: $I_{y\text{-feed}} = 1.\ e^{j0.}$ Amp]

z-directed probe-current feeds (z-feeds) :

For each layer from the bottom to the top:

Number of z-feeds: **NZFED**

`NZFED` (integer) : Number of z-feeds inserted in each layer

Location of each z-feed: **`X1,Y1`**

`X1,Y1` (real) : (x,y) coordinate of each z-load within the unit cell (cm)

Amplitude and phase of each z-feed: **`ZAMP,ZPHS`**

`ZAMP` (real) : Amplitude of each y-feed (Amp)

`ZPHS` (real) : Phase of each y-feed (degree)

Example:

`1`	[There is only one z-feed in Layer # 1 (bottom)]
`0.2,0.2`	[Location of the z-feed: $(x,y) = (0.2,0.2)$ cm]
`1.,90.`	[Probe-current: $I_{z\text{-feed}} = 1.\ e^{j\pi/2}$ Amp]
`1`	[There is only one z-feed in Layer # 3]
`0.2,0.2`	[Location of the z-feed: $(x,y) = (0.2,0.2)$ cm]
`1.,90.`	[Probe-current: $I_{z\text{-feed}} = 1.\ e^{j\pi/2}$ Amp]
`1`	[There is only one z-feed in Layer # 3 (top)]
`0.2,0.2`	[Location of the z-feed: $(x,y) = (0.2,0.2)$ cm]
`1.,90.`	[Probe-current: $I_{z\text{-feed}} = 1.\ e^{j\pi/2}$ Amp]

As shown in the above example, one can construct a continuous current feed between the bottom and top surfaces of the structure by placing probe-currents with same amplitude and phase value at the same (x,y) coordinate in each layer.

4.2 Format of Data Files

In this subsection we describe the formats of the input data files “Data1” and “Data2”, the output files “Imp”, “Trans”, “Reflect”, “EqvCur”, “EqvCurB”, and “EdegUnk”, as well as the parameter file “fema.dm1”. “Data1” and “Data2” are generated by the driver, but can also be provided from another source. These files are the input files for the analysis code and contain all necessary data for the code to be run. “Data1” contains excitation and global geometry information and “Data2” contains surface mesh data (node and triangular element numbering), definitions for periodic boundary conditions, and locations of discrete elements (metallic sections, R-cards, loads, pins, feeds) in terms of node and triangle numbers. “Imp”, “Trans”, and “Reflect” are generated by the prism code and contain computed input impedances related to probe feeds, transmission and reflection coefficients, respectively, with respect to frequency for the specified scan direction. The files “EqvCur”, “EqvCurB”, and “EdgeUnk” are generated for the frequency specified by the parameter NFREQ in the file “Data1” and contain equivalent surface current densities and the values of the edge unknowns, respectively.

A. DATA1

1) <u>Problem type and configuration:</u> **ITYP,IBOT,ICOM**

ITYP (integer): Problem = 1 → Radiation *or* = 2 → Scattering

IBOT (integer): Bottom aperture = 1 → Open *or* = 2 → Closed (metal-backed)

ICOM (integer): = 1 → Commensurate *or* = 2 → Non-commensurate

2) <u>Polarization type of incident plane wave and scan direction:</u> **IPOL,PHI,THETA**

IPOL (integer): Polarization = 0 → No plane wave excitation, Radiation (ITYP=1)
Polarization = 1→ TE-pol *or* = 2 → TM-pol (Scattering: ITYP=2)
PHI, THETA (real): Array scan angles (degree)

3) <u>Frequency of operation:</u> **FRBEG,FREND,FRSTP**

FRBEG, FREND, FRSTP (real): Beginning, end and incremental frequencies (GHz)

4) <u>Miscellaneous parameters:</u>

i) **NFREQ,RACC,MCON,IDIS**

NFREQ (integer): Number of frequency for which the files “EqvCur”, “EqvCurB”, and “EdgeUnk” are saved ; NFREQ=1 (default)

RACC (real): Relative accuracy for the solver; RACC=0.01 (default)

In most cases, a relative accuracy RACC=0.01 gives good results, sometimes smaller values might be necessary.

MCON (integer): Monitor convergence; MCON=0 (Default: Monitoring → OFF),
MCON=1 (Monitoring → ON)

IDIS (integer): Type of prisms used for the mesh; IDIS=1 → Distorted
IDIS=2 → Right-angled (default)

ii) **NPRE,DEF,IDF1,IDF2**

NPRE (integer): Flag for preconditioning; NPRE=0 → No preconditioning
NPRE=1 → Diagonal precond. (default)
NPRE=2 → AIPC (not recommended)

For non-perpendicular scan direction no preconditioning usually gives better results than diagonal preconditioning. AIPC is not recommended, especially not in conjunction with AIM.

DEF (real): control parameter for AIPC

IDF1 (integer): flag for solver selection; IDEF1=**0** → BiCG solver (default)

IDEF1=1 → GMRES solver

IDF2 (integer): Number of search vectors per GMRES restart (for BiCG, value is ignored but must be present)

AIM computation parameters:

Commensurate: **IAM1,NSB1,NSB2,NFBX,NFBY**

IAM1 (integer) : Flag for AIM computation; IAM1=1 → AIM is active,
IAM1=0 → AIM is not active

NSB1,NSB2 (integer) : NSB1→ Number of samples in the regular grid; NSB2=2*NSB1

NFBX,NFBY (integer) : Nearfield threshold in the regular grid for the *x*- and *y*-directions

Non-commensurate: **IAM1,NSB1,NSB2,NFBX,NFBY,NST1,NST2,NFTX,NFTY**

IAM1 (integer) : Flag for AIM computation; IAM1=1 → AIM is active,
IAM1=0 → AIM is not active

NSB1,NSB2 (integer) : Number of samples in the bottom surface grid; NSB2=2*NSB1

NFBX,NFBY (integer) : Nearfield threshold in the top surface grid for both directions

NST1,NST2 (integer) : Number of samples in the top surface grid; NST2=2*NST1

NFTX,NFTY (integer) : Nearfield threshold in the top surface grid for both directions

For more information concerning the AIM parameters see the relevant description for the driver input. In addition to the parameters described there, "Data1" contains the parameters NSB2 and NST2 which are the sizes of the two-dimensional FTT pads and must be equal or larger than NSB2=2*NSB1 and NST2=2*NST1. The driver sets these parameters to NSB2=2*NSB1 and NST2=2*NST1. However, the speed of the utilized FFT algorithm differs with the size of the FFT pad so that the code may run faster for FFT pad sizes slightly larger than the minimum size. Relative CPU timings of the utilized 2-D FFT algorithm are given in the Appendix. According to the given curves, the size of the FFT pad should be chosen as a local minimum.

6) FSS unit cell size and FEM sample size: **`XL,YL,DX,DY,GAMMA`**

`XL,YL` (real): FSS unit cell size (`ICOM=1`) *or* Largest unit cell size (`ICOM=2`) (cm)

`DX,DY` (real): FEM sample size (cm)

`GAMMA`: Unit cell angle between the two lattice vectors. One of the vectors is always aligned with the *x*-axis. If `GAMMA` is not 90^0 the height `YL` in *y*-direction is shorter than the length of the second lattice vector.

7) Depth of structure and number of layers: **`ZL,NZ`**

`ZL` (real): Depth or thickness of structure (cm)

`NZ` (integer): Number of layers along the depth

8) FSS unit cell size: **`FXL,FYL,GAMMA`**

Only if `ICOM=2` (Non-commensurate case), for each layer from the bottom to the top:

`FXL,FYL` (real) : *x-/y*-dimensions of FSS unit cell in cm (each layer's top surface)

`GAMMA`: Unit cell angle between the two lattice vectors. One of the vectors is always aligned with the *x*-axis. If `GAMMA` is not 90^0 the height `YL` in *y*-direction is shorter than the length of the second lattice vector.

9) Layer parameters: **`DT,EPS,MU`**

For each layer from the bottom to the top:

`DT` (real): Layer thickness (cm)

`EPS,MU` (complex): Relative permittivity and permeability

B. DATA2

1) Surface mesh parameters:

i) Number of nodes and triangles: **`NNOD,NTRI`**

`NNOD` (integer): Number of nodes in the surface mesh

NTRI (integer): Number of triangular elements in the surface mesh

Note that the size of the surface mesh is one FEM sample larger (in both *x*- and *y*-directions) than the largest FSS unit cell size. That is, there is at least one additional row of triangular elements around each FSS unit cell of each layer; the number of additional rows can be more than one for non-commensurate configurations. The triangular elements forming these additional parts of the layers are named as "absorber triangles" and are dropped within the analysis code. The idea in the non-commensurate case is to define the surface mesh large enough so that it is larger than the largest unit cell and drop the parts of the mesh not necessary for smaller unit cells by defining them as "absorber triangles" in the corresponding layers.

ii) Triangle and node numbering: **ITR,NODE1,NODE2,NODE3**

ITR (integer): Triangle number (1 → NTRI)

NODE1,NODE2,NODE3 (integer): Node numbers for triangular element ITR
Note, all triangles must be numbered counterclockwise.

iii) Node number and (x,y,z) coordinate: **IN,XSNOD,YSNOD,ZSNOD**

IN (integer): Node number (1 → NNOD)

XSNOD,YSNOD,ZSNOD (real) : (x,y,z) coordinates of node IN; ZSNOD=0. (default)

2) Interface information:

Number of layer interfaces: **NZI**

NZI (integer): Number of layer interfaces along the depth of the structure;
NZI= NZ (Number of layers) + 1

Note that in the driver, surface elements (*x*- and *y*-directed impedance loads and feeds, resistive cards and metallic sections) can be placed on each layer's top surface, corresponding to interface # 2 → # NZI. However, in the "Data2" file also surface elements in the bottom surface of the lowest layer can be defined (interface # 1).

For each interface from the bottom to the top: **ZCO**

ZCO (real): *z*-coordinate of each interface in cm;
Top surface (interface number: NZI) → ZCO= 0.
Bottom surface (interface number: 1) → ZCO= -ZL (the depth)

3) Absorber triangles and periodic boundary condition (PBC) :

For *commensurate* case:

i) Number of absorber triangles: **NABTRI**

NABTRI (integer): Number of absorber triangles (as defined above) around the unit cell

ii) Absorber triangle numbering: **IAT**

IAT (integer): triangle number

iii) Number of PBC nodes: **NXPER**

NXPER (integer): Number of PBC nodes (outer nodes surrounding the unit cell) in x-direction

iv) Bottom and top PBC node list: **IPBOT,IPTOP**

For all PBC nodes along the x-direction:

IPBOT (integer): Node number along the lower side of the unit cell

IPTOP (integer): Corresponding node number along the upper side of the unit cell

v) Number of PBC nodes: **NYPER**

NYPER (integer): Number of PBC nodes (outer nodes surrounding the unit cell) in y-direction

vi) Left and right PBC node list: **IPLFT,IPRGT**

For each PBC node along the y-direction:

IPLFT (integer): Node number along the left side of the unit cell (along y-axis)

IPRGT (integer): Corresponding node number along the right side of the unit cell

For *non-commensurate* case: Repeat input options above (i-ii) for all layer interfaces. Follow with (iii-iv) for all layers and (v-vi) for all layers.

(1 → NZI). In addition, for each interface, specify the following:

vii) Number of periodic edges: **ICP**

ICP (integer) : Number of edges in the outer region of the FSS unit cell

viii) Periodic edge list: **N1,N2,PN1,PN2**

N1,N2 (integer) : Nodes located in the outer region of the cell and representing an edge

PN1,PN2 (integer) : Nodes located in the inner region of the cell and representing the corresponding periodic edge

In the non-commensurate case, the so-called periodic edges are necessary if the cross section of the unit cell changes from one prism layer to the other. The list of periodic edges relates the edges on the horizontal portions of the stairs from one unit cell size to the other to the corresponding edges inside the FE volume (see Figure 3) so that these edges are well-defined according to the applied principle of decoupled layer periodicities.

4) Metallic Triangles:

For each layer interface, repeat these steps as a pair:

Number of metallic triangles: **NOMT**

NOMT (integer): Number of metallic triangles in each interface's surface mesh

Triangle number: **INMT**

INMT (integer): metallic triangle number

5) Resistive Triangles:

For each layer interface, repeat these steps as a pair:

Number of resistive triangles: **NORT**

NORT (integer): Number of resistive triangles in each interface's surface mesh

Triangle number and Conductance: **INRES,SCN**

INRES (integer): resistive triangle number

SCN (complex): Surface conductance of resistive triangle in 1/Ohm

6) Lumped Impedance Loads:

x-directed loads (*x*-loads):

For each layer interface, repeat these steps as a triplet:

Number of *x*-loads: **NXD**

NXD (integer): Number of *x*-loads placed on each layer interface

Admittance: **AXLD**

AXLD (complex): Admittance value of each *x*-load in 1/Ohm

Node location: **LXB,LXE**

LXB, LXE (integer): Corresponding beginning and end node numbers in the surface mesh for the *x*-load

y-directed loads (*y*-loads):

For each layer interface, repeat these steps as a triplet:

Number of *y*-loads: **NYD**

`NYD` (integer): Number of y-loads placed on the layer interface

<u>Admittance:</u> **`AYLD`**

`AYLD` (complex): Admittance value of y-load in 1/Ohm

<u>Node location:</u> **`LYB,LYE`**

`LYB, LYE` (integer): Corresponding beginning and end node numbers in the surface mesh for the y-load

<u>z-directed loads (z-loads):</u>

For each layer, repeat these steps as a triplet:

<u>Number of z-loads:</u> **`NZD`**

`NZD` (integer): Number of z-loads embedded in the layer

<u>Admittance:</u> **`AZLD`**

`AZLD` (complex): Admittance value of the z-load in 1/Ohm

<u>Node location:</u> **`LZC`**

`LZC` (integer): Corresponding node number in the surface mesh for the z-load

7) <u>Short-Circuit Pins:</u>

For each layer, repeat these steps as a pair:

<u>Number of pins:</u> **`NPIN`**

`NPIN` (integer): Number of short-circuit pins inserted in the layer

<u>Node location:</u> **`LPC`**

`LPC` (integer): Corresponding node number in the surface mesh for the pin

8) <u>Current-Probe Feeds:</u>

Only for radiation problems:

<u>x-directed feeds (x-feeds):</u>

For each layer interface, repeat these steps as a triplet:

<u>Number of x-feeds:</u> **`NXFED`**

`NXFED` (integer): Number of x-feeds placed on the layer interface

<u>Current:</u> **`XCUR`**

`XCUR` (complex): Current of the probe feed in Amp

<u>Node location:</u> **`FXB,FXE`**

FXB, FXE (integer) : Corresponding beginning and end node numbers in the surface mesh for the *x*-feed

y-directed feeds (*y*-feeds):

For each layer interface, repeat these steps as a triplet:

Number of *y*-feeds: **NYFED**

NYFED (integer): Number of *y*-feeds placed on each layer interface

Current: **YCUR**

YCUR (complex): Current value of each *y*-feed in Amp

Node location: **FYB,FYE**

FYB, FYE (integer) : Corresponding beginning and end node numbers in the surface mesh for the *y*-feed

z-directed feeds (*z*-feeds):

For each layer, repeat these steps as a triplet:

Number of *z*-feeds: **NZFED**

NZFED (integer): Number of *z*-feeds embedded in each layer

Current: **ZCUR**

ZCUR (complex): Current value of each *z*-feed in Amp

Node location: **FZC**

FZC (integer): Corresponding node number in the surface mesh for each *z*-feed

C. Imp

The output file "Imp" contains the input impedances for the individual probe feeds. For all feeds it contains a line with the data:

```
f      FEEDNR      Z_REAL      Z_IMAG      ITER
```

f: Frequency

FEEDNR: Number of feed

Z_REAL: Real part of input impedance

Z_IMAG: Imaginary part of input impedance

ITER: Number of iterations

D. Reflect

The output file "Reflect" contains the reflection coefficients at the top BI surface for plane wave excitation and the radiated field amplitude in scan direction for radiation problems. For all frequencies the files contains a line with the data:

```
f    TM_AMP    TM_PHASE  TE_AMP    TE_PHASE
```

`f`: Frequency

`TM_AMP`: Amplitude of TM component

`TM_PHASE`: Phase (degree) of TM component

`TE_AMP`: Amplitude of TE component

`TE_PHASE`: Phase (degree) of TE component

E. Trans

The output file "Trans" contains the transmission coefficients at the bottom BI surface (if open) for plane wave excitation and the radiated field amplitude in scan direction for radiation problems. For all frequencies the files contains a line with the data:

```
f    TM_AMP    TM_PHASE  TE_AMP    TE_PHASE
```

`f`: Frequency

`TM_AMP`: Amplitude of TM component

`TM_PHASE`: Phase (degree) of TM component

`TE_AMP`: Amplitude of TE component

`TE_PHASE`: Phase (degree) of TE component

F. EqvCur/EqvCurB

The file "EqvCur" contains the tangential magnetic current components in the top surface of the mesh and the file "EqvCurB" the corresponding data in the bottom surface. Both files are only generated for one frequency specified by the parameter `NFREQ` in the file "Data1". The first line of the files contains two integer numbers:

`Ntri` `Wnum`

`NTri`: Number of triangles in the surface

`Wnum`: Wavenumber

Then for each triangle a line follows with the data:

```
IT    Area    Xc    Yc    Zc    Mx    My    Mz
```

`IT`: Triangle number

`Area`: Area of triangle

`Xc`: x-coordinate of triangle center

`Yc`: y-coordinate of triangle center

`Zc`: z-coordinate of triangle center

`Mx`: x-component of magnetic current

`My`: y-component of magnetic current

`Mz`: z-component of magnetic current

G. EdgeUnk

The file "EdgeUnk" contains the values for the FE unknowns in the mesh. The file is only generated for one frequency specified by the parameter `NFREQ` in the file "Data1". For each edge in the volume mesh, a line is present with the data:

```
I    X1    Y1    Z1    X2    Y2    Z2    Amp    Real    Imag
```

`I`: Number of edge

`X1`: x-coordinate of first grid point

`Y1`: y-coordinate of first grid point

`Z1`: z-coordinate of first grid point

`X2`: x-coordinate of second grid point

`Y2`: y-coordinate of second grid point

`Z2`: z-coordinate of second grid point

`Amp`: Amplitude of the edge unknown

`Real`: Real part of the unknown

`Imag`: Imaginary part of the unknown

H. fema.dm1

"fema.dm1" is the parameter file for the FORTRAN codes. It defines parameters which determine the size of the data arrays in the codes. The file contains the same parameters for the driver as well as the analysis codes. However, in the driver not all parameters are used. The parameters must be larger than the largest possible size of the quantity for the given model. The list of these parameters with their descriptions are given in the following:

`NdmSNo`: Number of nodes in the surface mesh

`NdmVNo`: Number of nodes in the volume mesh (number of layers times surface nodes)

`NdmPNo`: Number of nodes on periodic boundary in a layer (per side)

`NdmTri`: Number of triangles in the surface mesh

`NdmPri`: Number of prisms in the volume mesh (number of layers times triangles)

`NdmSEd`: Number of edges in the surface mesh

`NdmNZS`: Number of non-zero edges in the surface mesh

`NdmVEd`: Number of edges in the volume mesh

`NdmNZE`: Number of non-zero edges in the volume mesh

`NdmPEd`: Number of edges on periodic boundary in a layer (per side)

`NdmLay`: Number of prism layers in the volume mesh

`NdmREd`: Number of x- and y-directed resistive edges in the volume mesh

`NdmRTri`: Number of resistive triangles in the volume mesh

`NdmZPin`: Number of vertical short circuit pins (edges) in the volume mesh

`NdmZEd`: Number of vertical resistive edges in the volume mesh

`NdmRow`: Number of matrix elements in the sparse matrix (FE + BI)

`NdmRowP`: Number of matrix elements in the AIPC preconditioner matrix (must be larger than `NNZE` if GMRES is applied in conjunction with diagonal preconditioner)

`NdmFFTPad`: Size of 2-D-FFTPad in one dimension (at least two times number of regular grid points)

`NdmLoIt`: Number of local GMRES iterations

4.3 A Sample Run for Driver.f

We now present a sample run for the driver code by entering input data on the screen. If there is an input file generated for the driver as described in section 4.1, then the user only needs to specify the name of that file.

Example 1:

```
************* FSS_PRISM DRIVER *************

CHOOSE: 1) READ  INPUT DATA FROM THE FILE
        2) ENTER INPUT DATA ON SCREEN
1

ENTER THE INPUT FILE NAME
Fss_Input
```

Then, the driver gets the data from the specified file and processes them to generate data files "Data1" and "Data2" to be used by the analysis code. On the other hand, if the user wants to enter the input information on the screen, he needs to choose the first input option as "2" referring to the above example. The necessary geometry and excitation information are then asked to enter one by one as shown in the following example.

Example 2:

```
************* FSS_PRISM DRIVER *************
CHOOSE: 1) READ  INPUT DATA FROM THE FILE
        2) ENTER INPUT DATA ON SCREEN
1

ENTER THE INPUT FILE NAME
Fss_Input

CHOOSE: 1) RADIATION PROBLEM
        2) SCATTERING PROBLEM
2

CHOOSE: 1) COMMENSURATE FSS
        2) NONCOMMENSURATE FSS
2

THE BOTTOM SURFACE IS OPEN OR CLOSED ?
 1) OPEN    2) CLOSED (METAL BACKED)
```

```
1

CHOOSE POLARIZATION TYPE
 1) TE-POL    2)TM-POL
1

ENTER INCIDENCE ANGLE: PHI,TETA (DEG)
0.,90.

ENTER BEGINNING,END AND STEP FREQUENCIES:
      FRBEG,FREND,FRSTP (GHZ)
1.,10.,1.

ENTER NO. OF SAMPLES IN THE BOTTOM SURFACE GRID
50

ENTER NEARFIELD THRESHOLD IN THE BOTTOM SURFACE
GRID FOR THE X- AND Y-DIRECTIONS: NFBX,NFBY
25,25

ENTER NO. OF SAMPLES IN THE TOP SURFACE GRID
30

ENTER NEARFIELD THRESHOLD IN THE TOP SURFACE
GRID FOR THE X- AND Y-DIRECTIONS: NFTX,NFTY
20,20

ENTER THE LARGEST UNIT CELL SIZE: XL,YL (CM)
2.,2.

ENTER THE FEM SAMPLE SIZE: DX,DY (CM)
0.05,0.05

ENTER THE CAVITY DEPTH: ZL (CM)
1.25

ENTER NUMBER OF LAYERS ALONG THE DEPTH
3

EACH LAYER IS NUMBERED STARTING FROM THE BOTTOM.
NOTE : THE CAVITY APERTURE IS AT Z=0 PLANE.
FOR EACH LAYER, ENTER INFORMATION REQUESTED.
FSS ELEMENTS, X-,Y- DIRECTED LUMPED LOADS,
X-,Y-DIRECTED FEEDS, AND RESISTIVE CARDS MAY
BE PLACED ON TOP SURFACE OF EACH LAYER. ALSO,
Z-DIRECTED LUMPED LOADS, FEEDS, AND SHORT-CKT.
PINS MAY BE EMBEDDED IN EACH LAYER.

******************* LAYER # 1 *******************

ENTER THICKNESS, RELATIVE PERMITTIVITY
AND PERMEABILITY: DT(CM),EPS,MU
0.5,(1.,0.),(1.,0.)

ENTER FSS UNIT CELL SIZE: FXL,FYL (CM)
2.,2.

ARE THERE FSS ELEMENTS ON LAYER # 1 ?
        1) YES          2) NO
2

ENTER NUMBER OF RESISTIVE CARDS ON LAYER # 1
0
```

```
ENTER NUMBER OF X-DIRECTED LOADS ON LAYER # 1
0

ENTER NUMBER OF Y-DIRECTED LOADS ON LAYER # 1
0

ENTER NUMBER OF Z-DIRECTED LOADS IN LAYER # 1
0

ENTER NUMBER OF SHORT CIRCUIT PINS IN LAYER # 1
1

FOR THE PIN # 1, ENTER:

(X,Y) COORDINATE OF THE PIN: PIX,PIY (CM)
NOTE: THE LOWER-LEFT CORNER OF THE UNIT
CELL IS THE ORIGIN (0.,0.)
1.,1.

******************* LAYER # 2 ******************

ENTER THICKNESS, RELATIVE PERMITTIVITY
AND PERMEABILITY: DT(CM),EPS,MU
0.25,(2.,0.),(1.,0.)

ENTER FSS UNIT CELL SIZE: FXL,FYL (CM)
1.5,1.5
ARE THERE FSS ELEMENTS ON LAYER # 2 ?
        1) YES            2) NO
1


CHOOSE AN FSS ELEMENT TYPE:
1) STRIP        3) CROSSED-STRIP
2) SLOT         4) CROSSED-SLOT
   5) USER-SPECIFIED
3

ENTER SIZE OF 1ST SECTION OF THE CROSS: DPX,DPY (CM)
0.7,0.3

ENTER SIZE OF 2ND SECTION OF THE CROSS: DPX,DPY (CM)
0.3,0.7

ENTER LOCATION OF THE LOWER-LEFT
CORNER OF THE 1ST SECTION: XF,YF (CM)
NOTE: THE LOWER-LEFT CORNER OF
THE UNIT CELL IS THE ORIGIN (0.,0.)
0.4,0.6

ENTER LOCATION OF THE LOWER-LEFT
CORNER OF THE 2ND SECTION: XF,YF (CM)
NOTE: THE LOWER-LEFT CORNER OF
THE UNIT CELL IS THE ORIGIN (0.,0.)
0.6,0.4

ENTER NUMBER OF RESISTIVE CARDS ON LAYER # 2
0

ENTER NUMBER OF X-DIRECTED LOADS ON LAYER # 2
1

FOR THE X-DIRECTED LOAD # 1, ENTER:
```

```
BEGINNING AND END COORDINATES OF
THE LOAD: CX1,CY1,CX2,CY2 (CM)
NOTE: THE LOWER-LEFT CORNER OF THE UNIT
CELL IS THE ORIGIN (0.,0.)
CY1 SHOULD BE EQUAL TO CY2.
0.4,0.4,0.6,0.4

ADMITTANCE OF THE LOAD: AXLD (1/OHM)
(20.,0)

ENTER NUMBER OF Y-DIRECTED LOADS ON LAYER # 2
1

FOR THE Y-DIRECTED LOAD # 1, ENTER:

BEGINNING AND END COORDINATES OF
THE LOAD: CX1,CY1,CX2,CY2 (CM)
NOTE: THE LOWER-LEFT CORNER OF THE UNIT
CELL IS THE ORIGIN (0.,0.)
CX1 SHOULD BE EQUAL TO CX2.
0.4,0.4,0.4,0.6

ADMITTANCE OF THE LOAD: AYLD (1/OHM)
(10.,0)

ENTER NUMBER OF Z-DIRECTED LOADS IN LAYER # 2
0

ENTER NUMBER OF SHORT CIRCUIT PINS IN LAYER # 2
1

FOR THE PIN # 1, ENTER:
(X,Y) COORDINATE OF THE PIN: PIX,PIY (CM)
NOTE: THE LOWER-LEFT CORNER OF THE UNIT
CELL IS THE ORIGIN (0.,0.)
0.75,0.75

******************* LAYER # 3 ******************

ENTER THICKNESS, RELATIVE PERMITTIVITY
AND PERMEABILITY: DT(CM),EPS,MU
0.5,(1.,0.),(1.,0.)

ENTER FSS UNIT CELL SIZE: FXL,FYL (CM)
1.,1.

ARE THERE FSS ELEMENTS ON LAYER # 3 ?
        1) YES           2) NO
2

ENTER NUMBER OF RESISTIVE CARDS ON LAYER # 3
1

FOR THE R-CARD # 1, ENTER:

SIZE OF THE R-CARD: RX,RY (CM)
1.,1.

THE LOWER-LEFT CORNER OF THE R-CARD: CRX,CRY (CM)
NOTE: THE LOWER-LEFT CORNER OF THE UNIT
CELL IS THE ORIGIN (0.,0.)
(0.,0.)

SURFACE CONDUCTANCE OF THE R-CARD: SCN (1/OHM)
```

```
(1000.,0)

ENTER NUMBER OF X-DIRECTED LOADS ON LAYER # 3
0

ENTER NUMBER OF Y-DIRECTED LOADS ON LAYER # 3
0

ENTER NUMBER OF Z-DIRECTED LOADS IN LAYER # 3
1

FOR THE Z-DIRECTED LOAD # 1, ENTER:

(X,Y) COORDINATE OF THE LOAD: ZLX,ZLY (CM)
NOTE: THE LOWER-LEFT CORNER OF THE UNIT
CELL IS THE ORIGIN (0.,0.)
0.5,0.5

ADMITTANCE OF THE LOAD: AZLD (I/OHM)
(100.,0)

ENTER NUMBER OF SHORT CIRCUIT PINS IN LAYER # 3
0

**************** END OF INPUT DATA ***************
```

In addition to the data files, "Data1" and "Data2", a MatLab file "Setup.m" with its data files "MeshDs" and "Attr" is generated by the driver. When this setup file is used along with the available MatLab files "Mesh2.m" and "Mesh3.m", the user can display a 2-D mesh for each layer interface in (x,y) and a 3-D view of the whole geometry in the (x,y,z) coordinate system, respectively. Each planar and volumetric discrete element in the structure is represented by different colors in these figures. This way, the user can view and check the location and size of all specified elements. The colors shown in the mesh view and what they stand for are described as follows:

- Black triangles: "Absorber" triangles. Also, for metal-backed structures, the bottom surface will be in black.
- Black lines: x- or y-directed feeds.
- Black stars (*): z-directed feeds.
- Red triangles: Metallic sections.
- Yellow triangles: Resistive cards.
- Blue lines: x- or y-directed impedance loads.
- Blue stars (*): z-directed impedance loads.
- Black circles(o): short-circuit pins.

The numbering of the nodes and triangles can be visualized using the available MatLab files "GloNod.m" and "TriNum.m", respectively.

5. Examples

In this section, we present some application examples for the FSS_PRISM code. We start with three simple commensurate examples for FSS_PRISM_C before three non-commensurate examples for FSS_PRISM_NC are discussed. The necessary input/data files to run the example and validate the codes are delivered with the codes.

5.1 Examples for FSS_PRISM_C

A. Air Layer as Sanity Check

In this example, a unit cell of size 6*6 cm^2 of an air layer of thickness 4 cm is modelled by finite elements with BI termination on the top and bottom surfaces. For plane wave excitation with a plane wave incident from the top half-space the reflection coefficient on the top surface must be 0.0 and the transmission coefficient must be 1.0.

The input for the driver is:

```
2           1           1

          2  0.0000000E+00  0.0000000E+00

   1.000000      1.000000      1.000000

         20          15          15


   6.000000      6.000000      1.000000      1.000000

   4.000000             4

   1.0000 (1.000000,0.0000000E+00) (1.000000,0.0000000E+00)
   1.0000 (1.000000,0.0000000E+00) (1.000000,0.0000000E+00)
   1.0000 (1.000000,0.0000000E+00) (1.000000,0.0000000E+00)
   1.0000 (1.000000,0.0000000E+00) (1.000000,0.0000000E+00)

          2

          2

          2
```

```
2
0
0
0
0
0
0
```

The above data defines a scattering problem with open bottom surface for the commensurate code version. Excitation is an incident TM plane wave from the top half-space with incident angles $\theta=0^0$ and $\phi=0^0$. The excitation frequency is 1 GHz. The regular AIM grid has 20 grid points per dimension and the nearfield thresholds in *x*- and *y*-directions are set to 15 regular grid points. This means that the use of AIM has no effect for this example because all BI interactions are considered to be in the nearzone. Therefore, for this example AIM could be deactivated in the file "Data1" without loss of efficiency.

The listing of the "Data1" file is given below:

```
2           1           1
   2  0.0000000E+00  0.0000000E+00
   1.000000       1.000000       1.000000
   1  9.9999998E-03           1           2
   1  0.7500000               0          39
   1           20          40          15          15
   6.000000  6.000000  1.000000  1.000000  90.00000
   4.000000         4
 1.000000 (1.000000,0.0000000E+00) (1.000000,0.0000000E+00)
 1.000000 (1.000000,0.0000000E+00) (1.000000,0.0000000E+00)
 1.000000 (1.000000,0.0000000E+00) (1.000000,0.0000000E+00)
 1.000000 (1.000000,0.0000000E+00) (1.000000,0.0000000E+00)
```

AIM can be switched off by setting the first number in line 6 to 0. Also, all the other data can directly be changed in the "Data1" file. However, the data must fit to the corresponding data in the file "Data2".

The contents of the file "Reflect" are:

```
1.0000      0.0043     34.9735      0.0000      27.3999
```

The contents of the file "Trans" are:

```
1.0000      1.0008    132.3373      0.0000    -117.8762
```

B. FSS Slot Array

The input file for the FSS slot array as suggested in [4] is given by:

```
2           1           1

          2  0.0000000E+00  0.0000000E+00

   13.000000       13.000000       1.000000

         60          15          15

   1.000000       1.000000       0.050000       0.0500000

   0.100000               2

   0.050000     (4.000000,0.0000000E+00) (1.000000,0.0000000E+00)
   0.050000     (4.000000,0.0000000E+00) (1.000000,0.0000000E+00)

          2

          1
          2    1
          0.2  0.75
          0.4  0.15

          0

          0

          0

          0

          0

          0

          0
```

```
0
0
0
0
0
```

This data defines the unit cell of a slot array on a dielectric substrate of height 0.1 cm with relative permittivity ε_r=4 (see Figure 4). The FSS has bandpass characteristics with a resonance frequency of about 13 GHz.

The contents of file "Data1" are:

```
2           1           1
2  0.0000000E+00  0.0000000E+00

   13.00000       13.00000       1.000000

           1  9.9999998E-03          1          2

           1  0.7500000              0         39

           1     60     120         15         15

   1.000      1.000     5.00001E-02  5.00001E-02   90.00000

  0.1000000             2

 5.0001E-02 (4.000000,0.00000E+00) (1.000000,0.0000E+00)
 5.0001E-02 (4.000000,0.00000E+00) (1.000000,0.0000E+00)
```

The contents of the file "Reflect" are:

```
13.0000       0.1114   -173.6919      0.0022    178.4277
```

The contents of the file "Trans" are:

```
13.0000       0.9942    162.3789      0.0030    -23.4589
```

In Figure 4, the results obtained by the FE/BI code are compared to results of a MoM code utilizing the Green's function of the layered structure.

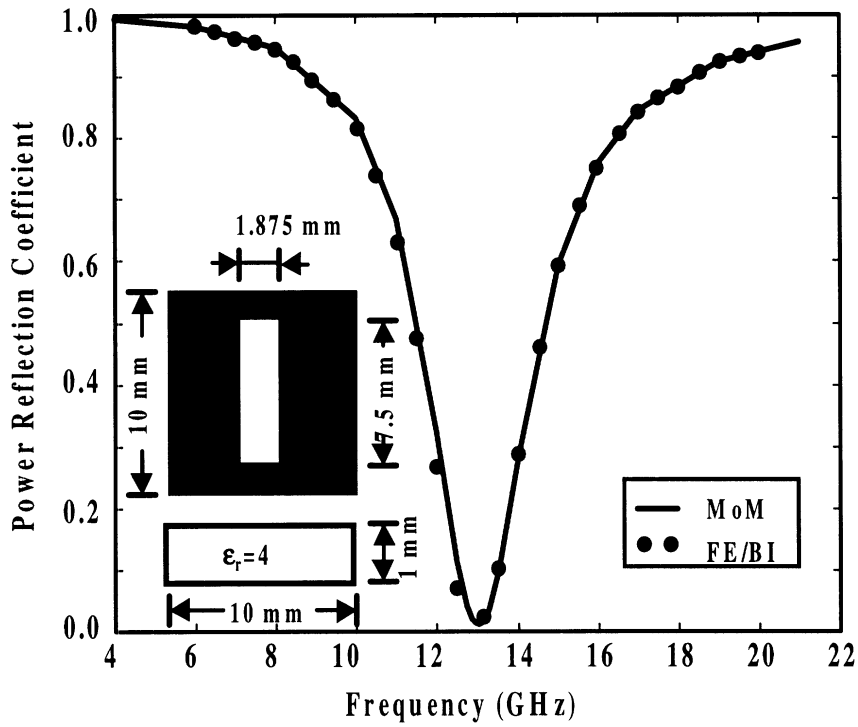


Figure 4: Power reflection coefficient of slot array, comparison of FE/BI and MoM results.

C. Strip Dipole Array

The input file for the strip dipole array is:

```
1            2            1

           0  0.0000000E+00  0.0000000E+00

   1.000000        1.000000        1.000000

          60           18           18

   10.00000        10.00000       0.5000000       0.5000000

   1.000000                2

 0.500000 (2.000000,0.00000E+00) (1.000000,0.00000E+00)
 0.500000 (2.000000,0.00000E+00) (1.000000,0.00000E+00)

           2

           1
           5            2
   1.000000       4.000000
   4.500000      0.5000000
   1.000000       4.000000
```

```
  4.500000       5.000000

           0

           0

           0

           0

           0

           0

           0

           0

           0

           0

           0

           0

           0

           1
  5.000000     4.500000     5.000000     5.000000
  1.000000      0.0000000E+00

           0

           0
```

This problem is defined as a radiation problem where the bottom of the mesh is closed and the structure is excited by a *y*-directed probe feed. The input impedance for the excitation frequency 1 GHz and normal scan direction is given in the file "Imp":

```
1.0000      1      4.9257    -27.2114       98
```

5.2 Examples for FSS_PRISM_NC

A. Air Layer as Sanity Check

As in the example for FSS_PRISM_C, a unit cell of an air layer of thickness 4 cm is modelled by finite elements with BI termination on the top and bottom surfaces. However, the unit cell size changes within the FE mesh from 6x6 cm^2 to 4x4 cm^2. For plane wave excitation with a plane wave incident from the top half-space the reflection coefficient on the top surface must still be 0.0 and the transmission coefficient must be 1.0.

The input file for the driver is:

```
2            1            2

           2  0.0000000E+00  0.0000000E+00

   1.000000       1.000000       1.000000

          18           15           15

          12           15           15

   6.000000       6.000000       1.000000       1.000000

   4.000000                4

   6.000000       6.000000
   6.000000       6.000000
   4.000000       4.000000
   4.000000       4.000000

   1.000000 (1.000000,0.00000E+00) (1.000000,0.00000E+00)
   1.000000 (1.000000,0.00000E+00) (1.000000,0.00000E+00)
   1.000000 (1.000000,0.00000E+00) (1.000000,0.00000E+00)
   1.000000 (1.000000,0.00000E+00) (1.000000,0.00000E+00)

           2

           2

           2

           2

           0

           0

           0
```

```
0
0
0
0
0
0
0
0
0
0
0
0
0
```

This is a scattering problem with open cavity bottom and the flag for the non-commensurate code option is set to 2. The AIM parameters are now set for the bottom and top BI surfaces according to their sizes. Also, the unit cell size is defined for all prism layers.

The "Data1" file is given by:

```
2            1            2
          2  0.0000000E+00  0.0000000E+00
  1.000000       1.000000       1.000000
          1  9.9999998E-03          1           2
          1  0.7500000              0          39
          1      18     36     15      15     12
                 24     15     15
  6.000   6.0000   1.0000   1.0000   90.000
  4.000000              4
  6.000000       6.000000       90.00000
  6.000000       6.000000       90.00000
  4.000000       4.000000       90.00000
```

```
   4.000000       4.000000       90.00000

   1.00    (1.000000,0.000000E+00) (1.000000,0.000000E+00)
   1.00    (1.000000,0.000000E+00) (1.000000,0.000000E+00)
   1.00    (1.000000,0.000000E+00) (1.000000,0.000000E+00)
   1.00    (1.000000,0.000000E+00) (1.000000,0.000000E+00)
```

The contents of the file "Reflect" is:

```
1.0000      0.0034     20.7327      0.0000     124.4750
```

The contents of the file "Trans" is:

```
1.0000      1.0008     84.2481      0.0000      -1.6739
```

B. 8-Layer Non-Commensurate Patch FSS

This example is a low-pass FSS structure which was investigated in [5]. It consists of 8 patch layers with different patch sizes and periodicities. The input file for the driver is given by (this listing omits lines with 0s for resistive triangles, edges, and short circuit pins.):

```
2          1          2

           2  0.0000000E+00  0.0000000E+00

   0.100       1.000       0.030

          65   18   18
          70   18   18

   20.0       20.0      1.0      1.0

   70.2          45

   18.0       18.0
   18.0       18.0
   18.0       18.0
   18.0       18.0
   18.0       18.0
   18.0       18.0
   18.0       18.0
   18.0       18.0
   18.0       18.0
   18.0       18.0
   18.0       18.0
   18.0       18.0
   18.0       18.0
   18.0       18.0
```

```
18.0       18.0
18.0       18.0
18.0       18.0
18.0       18.0
20.0       20.0
20.0       20.0
20.0       20.0
20.0       20.0
20.0       20.0
20.0       20.0
20.0       20.0
20.0       20.0
20.0       20.0
20.0       20.0
18.0       18.0
18.0       18.0
18.0       18.0
18.0       18.0
18.0       18.0
20.0       20.0
20.0       20.0
20.0       20.0
20.0       20.0
20.0       20.0
20.0       20.0
20.0       20.0
20.0       20.0
20.0       20.0
20.0       20.0
20.0       20.0
20.0       20.0

1.56 (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
1.56 (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
1.56 (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
1.56 (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
1.56 (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
1.56 (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
1.56 (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
1.56 (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
1.56 (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
1.56 (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
1.56 (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
1.56 (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
1.56 (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
1.56 (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
1.56 (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
1.56 (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
1.56 (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
1.56 (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
1.56 (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
1.56 (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
1.56 (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
1.56 (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
1.56 (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
1.56 (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
1.56 (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
```

```
 1.56 (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
 1.56 (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
 1.56 (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
 1.56 (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
 1.56 (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
 1.56 (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
 1.56 (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
 1.56 (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
 1.56 (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
 1.56 (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
 1.56 (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
 1.56  (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
 1.56  (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
 1.56  (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
 1.56  (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
 1.56  (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
 1.56  (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
 1.56  (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
 1.56  (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)
 1.56  (2.031400,-0.0500000E+00) (1.000000,0.0000000E+00)

          2

          2

          2

          2

           1
           5          9
 4.0         4.0
 1.0         1.0
 4.0         4.0
 7.0         1.0
 4.0         4.0
13.0         1.0
 4.0         4.0
 1.0         7.0
 4.0         4.0
 7.0         7.0
 4.0         4.0
13.0         7.0
 4.0         4.0
 1.0        13.0
 4.0         4.0
 7.0        13.0
 4.0         4.0
13.0        13.0


          2

          2

          2
```

2

1
5 9

4.0	4.0
1.0	1.0
4.0	4.0
7.0	1.0
4.0	4.0
13.0	1.0
4.0	4.0
1.0	7.0
4.0	4.0
7.0	7.0
4.0	4.0
13.0	7.0
4.0	4.0
1.0	13.0
4.0	4.0
7.0	13.0
4.0	4.0
13.0	13.0

2

2

2

2

1
5 4

6.0	6.0
1.0	1.0
6.0	6.0
10.0	1.0
6.0	6.0
1.0	10.0
6.0	6.0
10.0	10.0

2

2

2

2

1
5 4

7.0	7.0
1.0	1.0
7.0	7.0
11.0	1.0

7.0 7.0
1.0 11.0
7.0 7.0
11.0 11.0

2

2

2

2

1
5 4
7.0 7.0
1.0 1.0
7.0 7.0
11.0 1.0
7.0 7.0
1.0 11.0
7.0 7.0
11.0 11.0

2

2

2

2

1
5 4
6.0 6.0
1.0 1.0
6.0 6.0
10.0 1.0
6.0 6.0
1.0 10.0
6.0 6.0
10.0 10.0

2

2

2

2

1
5 16
3.0 3.0
1.0 1.0
3.0 3.0
6.0 1.0

3.0	3.0
11.0	1.0
3.0	3.0
16.0	1.0
3.0	3.0
1.0	6.0
3.0	3.0
6.0	6.0
3.0	3.0
11.0	6.0
3.0	3.0
16.0	6.0
3.0	3.0
1.0	11.0
3.0	3.0
6.0	11.0
3.0	3.0
11.0	11.0
3.0	3.0
16.0	11.0
3.0	3.0
1.0	16.0
3.0	3.0
6.0	16.0
3.0	3.0
11.0	16.0
3.0	3.0
16.0	16.0

2

2

2

2

1
5 16

3.0	3.0
1.0	1.0
3.0	3.0
6.0	1.0
3.0	3.0
11.0	1.0
3.0	3.0
16.0	1.0
3.0	3.0
1.0	6.0
3.0	3.0
6.0	6.0
3.0	3.0
11.0	6.0
3.0	3.0
16.0	6.0
3.0	3.0
1.0	11.0

```
  3.0        3.0
  6.0       11.0
  3.0        3.0
 11.0       11.0
  3.0        3.0
 16.0       11.0
  3.0        3.0
  1.0       16.0
  3.0        3.0
  6.0       16.0
  3.0        3.0
 11.0       16.0
  3.0        3.0
 16.0       16.0

          2

          2

          2

          2

          2
```

Instead of using the original µm-dimensions, the geometry is scaled to cm. Similarly, the frequency must be scaled by a factor of 10000. In Figure 5, the transmission coefficient of the structure is compared to results published in [5].

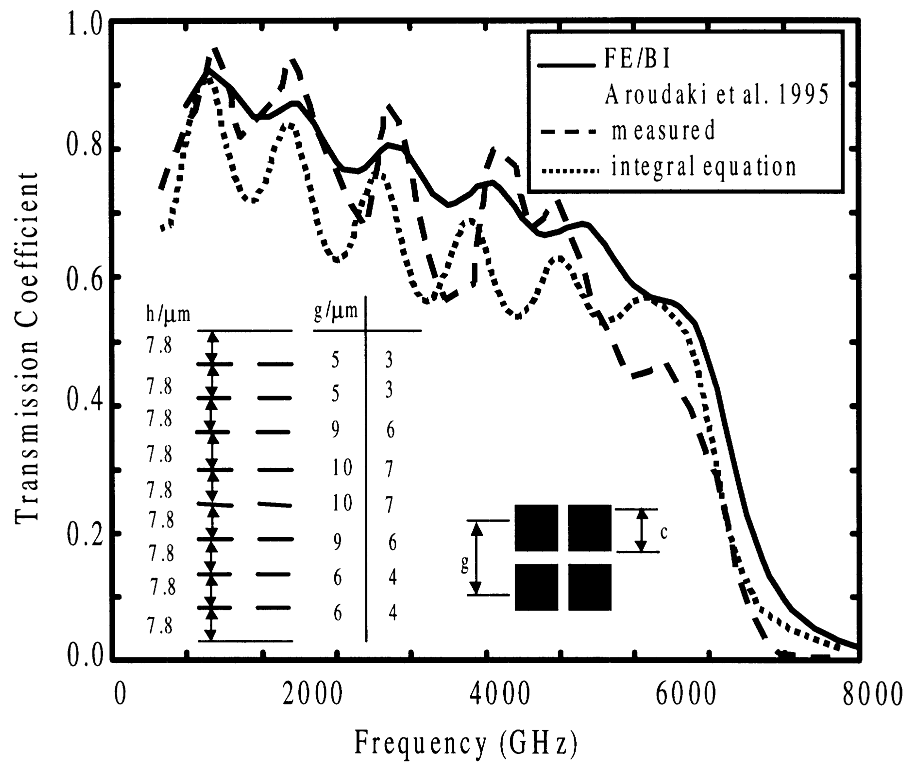


Figure 5: Transmission coefficient for 8 layer FSS compared to calculated and measured results in [5], normal incidence, ε_r=2.0324-j0.08 for FE/BI calculations.

C. Non-Commensurate Crossed Dipole FSS

This example deals with dipole arrays on top of a 3 layer FSS structure. The FSS is intended to have a frequency dependent reflection coefficient which gives optimal radiation efficiency of the dipole array over a wide frequency band. This structure was investigated in the SERAT program.

Together with the code, input files are delivered for the generation of the 3 layer FSS structure and for the FSS structure with dipole arrays on the top surface. File "Input2.txt" generates the 3 layer FSS with plane wave excitation. To suppress surface waves in the structure, in file "Input3.txt" resistive cards are added. The files "Input4.txt", "Input5.txt", and "Input6.txt" generate the 3 layer FSS including resistive cards with dipoles of different length in the top surface.

In Figure 6 and Figure 7, amplitudes and phases of the reflection coefficient in the top surface are given for increasing losses in the resistive cards placed in the three FSS layers. Figure 8 shows real and imaginary parts of the input impedance of the dipoles with various lengths on top of the FSS structures. It can be seen that the imaginary part of the input impedance is close to zero over a broad frequency band.

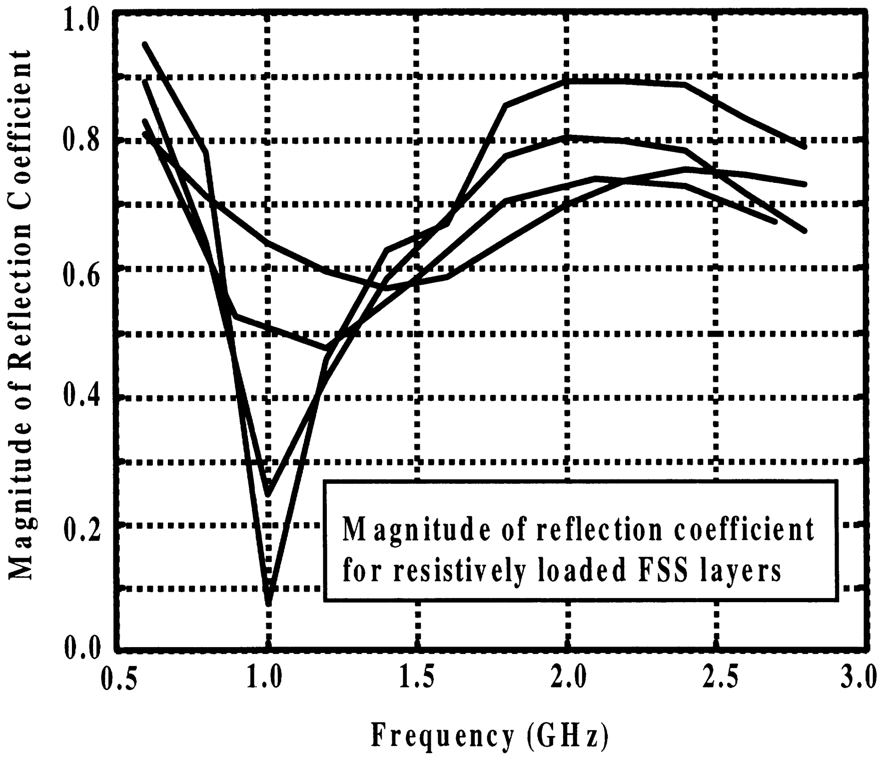


Figure 6: Magnitude of reflection coefficient of FSS structure for increasing losses in resistive cards.

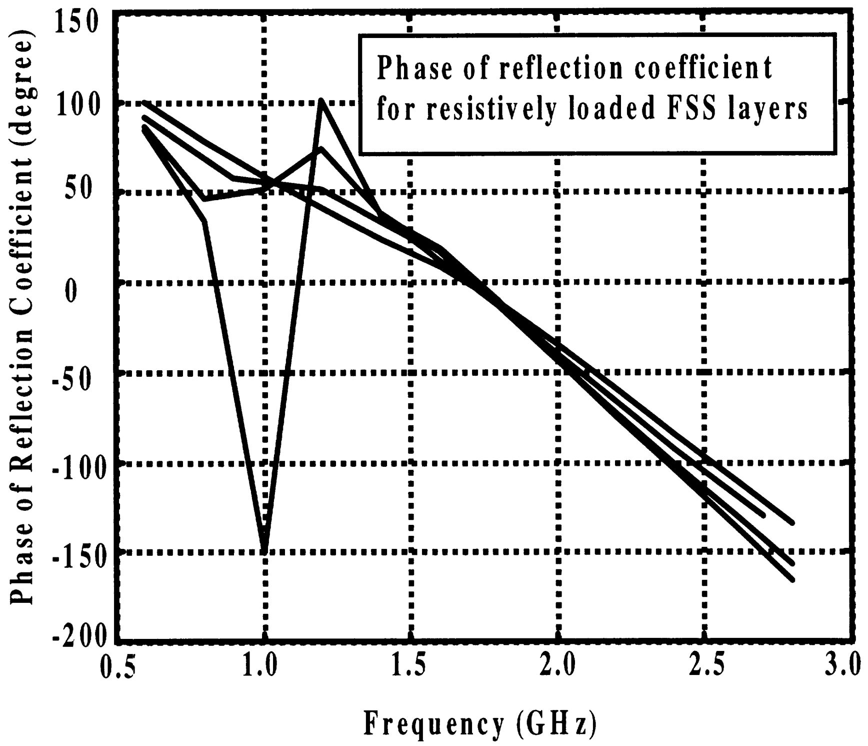


Figure 7: Phase of reflection coefficient for increasing losses in resistive cards.

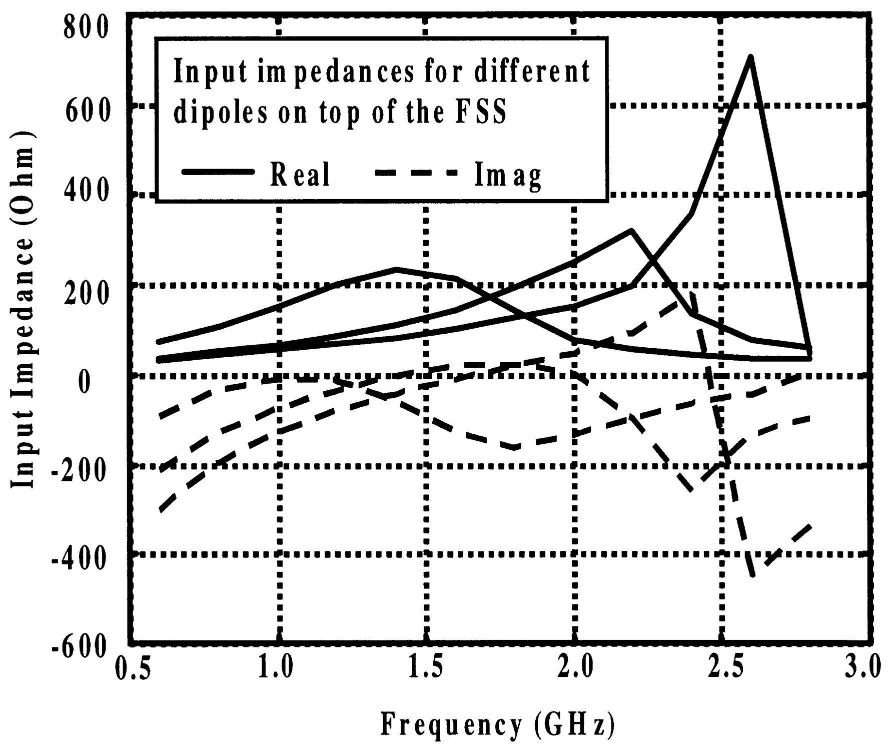


Figure 8: Input impedances of dipoles with varying length on top of the FSS (maximum attenuation of surface waves according to Figure 6 and Figure 7).

5.0 Electric field viewer for FSS-Prism by Dan Giszczak

This program was written for Windows 95 or Windows NT using OpenGL.

This program will view EdgeUnk files when a Data2 file is available. Large files may take a while to load. The status bar will show progress. The DOS window will prompt for the number of samples you wish per average edgelength. This fits a chosen number of samples across a distance equal to an average triangle edge length. For parts with few elements, say a hundred, a value like 10 will produce good results. For 10000 elements, 1.0 is good. In general, take 100/sqrt(number of elements) to get a good looking plot. Fraction values are permitted. Values under 0.5 produce no display.

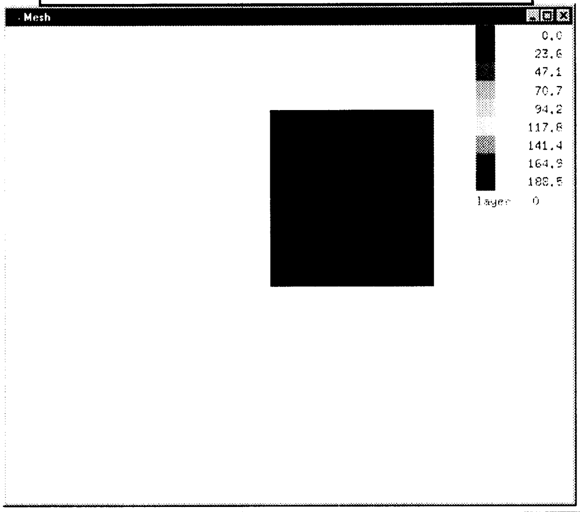
Once the view is loaded, press the first mouse button and drag the mouse on the display window to rotate to model. Press the right mouse button and drag to move the part around. To zoom in and out, either press both left and right buttons together or press the middle button and drag up or down.

T	T alternates between triangle and interpolated modes.
Q	Press q to display the quiver plot. This plot is a set of arrows showing the magnitude and direction of the real part of electric field.

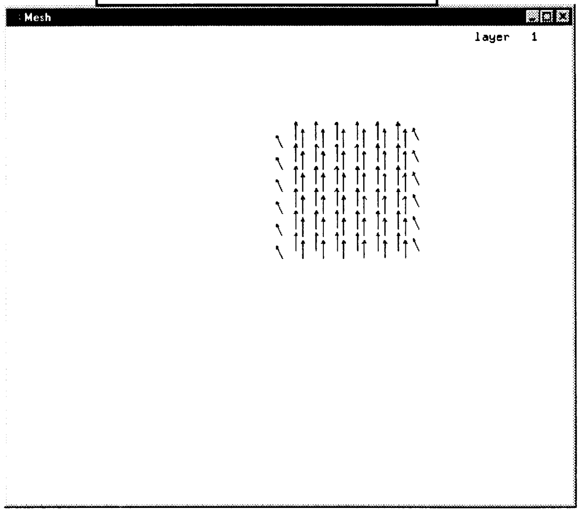
	While showing triangles, the following options are available
G	G will display the geometry. (T or Q returns to data display.)
X	Press x to display the x component of the electric field. Either the real part or the imaginary part may be shown depending upon which mode is current.
Y	Y displays the y component of the electric field. Both real and imaginary parts can be shown separately.
M	This displays the magnitude of the electric field, real or imaginary part.
R	Press r to display the real component x direction, y direction, or magnitude.
I	I will show the imaginary component x direction, y direction, or magnitude.
A	Animate the fields when overall magnitude is displayed.
	While showing interpolated data
R	R will bring up the real part magnitude.
I	I shows the imaginary magnitude.
	While in either mode
O	This displays the overall magnitude of electric field, combining both the real and imaginary magnitudes.
+	This key moves up a sample layer in the FSS.
-	- moves down a sample layer.
L	L displays all layers.
[	To decrease the distance between layers, press [.
]	To spread the layers apart more, press].

Use the Data1 and Data2 files from the Prism example with the generated EdgeUnk. Put them in a directory with FSSEfield.exe and run the executable. When questioned, use 8.8 samples.

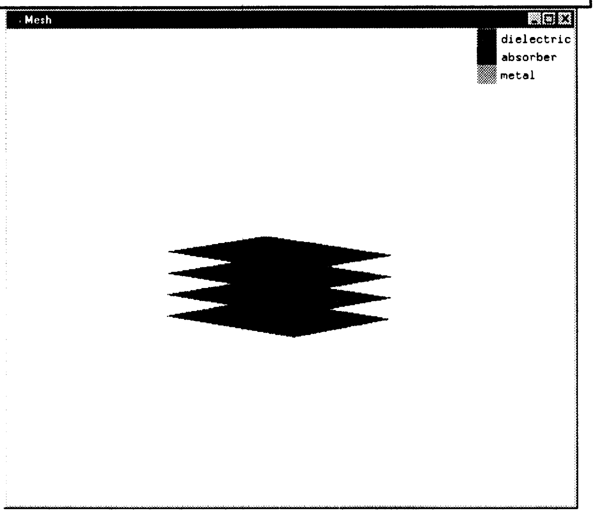
When it loads up, it should look like this.



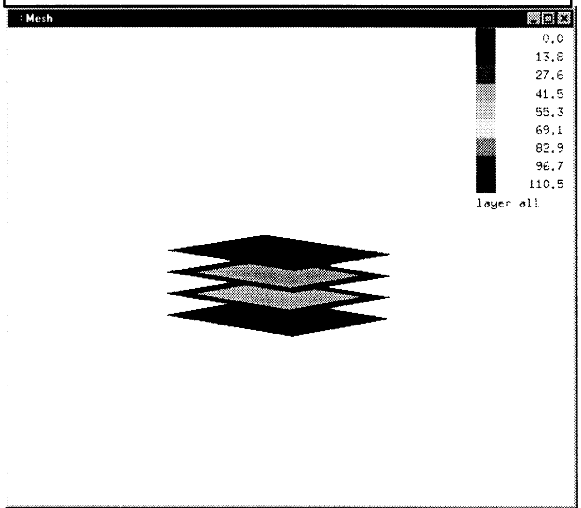
Hit ‘Q’ for the quiver plot.



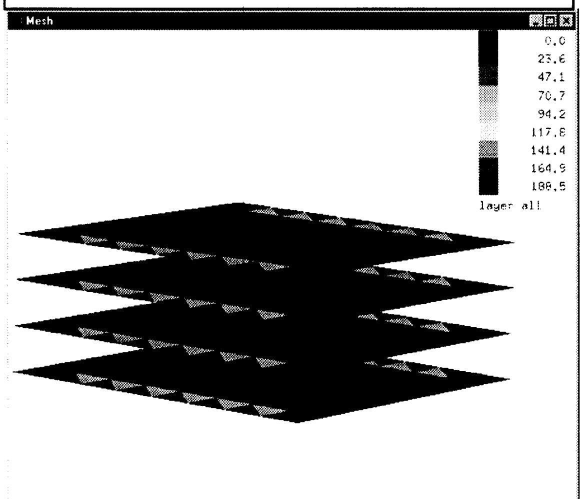
Hit ‘G’ then press the left mouse button on the window and rotate by dragging to see all the groups.



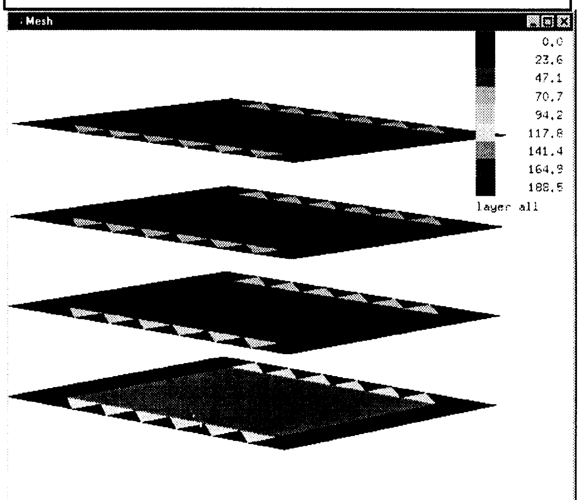
Hit ‘T’, ‘I’ to see the imaginary part of the fields.



Press ‘T’,’O’ to see the overall magnitude mesh data. Press the middle or left and right buttons and drag up to zoom in.



Press ‘]’ several times to increase the distance between the layers so that you can see them all. Then press the right mouse button and drag up until you can see the whole drawing.



Magnetic current viewer for FSS-Prism by Dan Giszczak

This program will view EqvCur and EqvCurB files when a Data2 file is available.

X	Show magnitude of the current in the x direction.
Y	Magnitude of the currents in the y direction.
Z	Z direction current magnitudes.
O	Show the overall <x,y,z> magnitude.
T	View magnitudes for the top layer.
B	Bottom layer magnitudes.

Bibliography

[1] T. F. Eibert, J. L. Volakis, D. R. Jackson, D. R. Wilton, "Hybrid FE/BI Modeling of 3-D Doubly Periodic Structures Utilizing Triangular Prismatic Elements and a MPIE Formulation Accelerated by the Ewald Transformation," *to be published.*

[2] T. F. Eibert, J. L. Volakis, "Adaptive Integral Method for Hybrid FE/BI Modeling of 3-D Doubly Periodic Structures," *to be published.*

[3] T. F. Eibert, J. L. Volakis, "FE/BI Modelling of 3-D Doubly Periodic Structures with Non-Commensurate Layer Periodicities," *to be published.*

[4] R. Mittra, C. H. Chan, T. Cwik, "Techniques for Analyzing Frequency Selective Surfaces - A Review," *Proc. IEEE*, Vol. 76, No. 12, pp. 1593-1615, Dec. 1988.

[5] H. Aroudaki, V. Hansen, H.-P. Gemünd, E. Kreysa, "Analysis of Low-Pass Filters Consisting of Multiple Stacked FSS's of Different Periodicities with Applications in the Submillimeter Radioastronomy," *IEEE Trans. AP*, Vol. 43, No. 12, pp. 1486-1491, Dec.1995.

[6] J. Gong, D. Jackson, J. Volakis, D. Wilton, "Hybrid Finite Element and Moment Method Software for the SERAT Array," *1st Quarterly Report 035067-1-T.*

[7] S. Bindiganavale, Y. Erdemli, J. Gong, D. Jackson, J. Volakis, D. Wilton, "Hybrid Finite Element and Moment Method Software for the SERAT Array," *2nd Quarterly Report 035067-2-T.*

[8] Y. Erdemli, T. Eibert, D. Jackson, J. Volakis, D. Wilton, "Hybrid Finite Element and Moment Method Software for the SERAT Array," *3rd Quarterly Report 035067-3-T.*

[9] T. Eibert, Y. Erdemli, K. Sertel, D. Jackson, J. Volakis, D. Wilton, "Hybrid Finite Element and Moment Method Software for the SERAT Array," *4th Quarterly Report 035067-4-T.*

[10] T. Eibert, Y. Erdemli, K. Sertel, D. Jackson, J. Volakis, D. Wilton, "Hybrid Finite Element and Moment Method Software for the SERAT Array," *5th Quarterly Report 035067-6-T.*

Appendix:

Relative CPU Timings for the Employed FFT Algorithm

The following diagrams, Figure 9 to Figure 18, show relative CPU times for the 2D FFT algorithm employed in the FSS_PRISM code. The algorithm is capable of calculating the FFT for arbitrary FFT pad sizes. However, as seen in the diagrams, the required CPU times can vary significantly for slightly different sizes of the FFT pad. Therefore, it is recommended to adjust the parameters `NSB2` and `NST2` in the file "Data1" so that they hit a local minimum on the timing curves. This can be done by increasing the values that were set by the driver without changing the corresponding parameters `NSB1` and `NST1`. Alternatively, also `NSB1` and `NST1` can be increased as long as they are smaller than half of `NSB2` and `NST2`, respectively, in order to utilize the size of the FFT pad most efficiently.

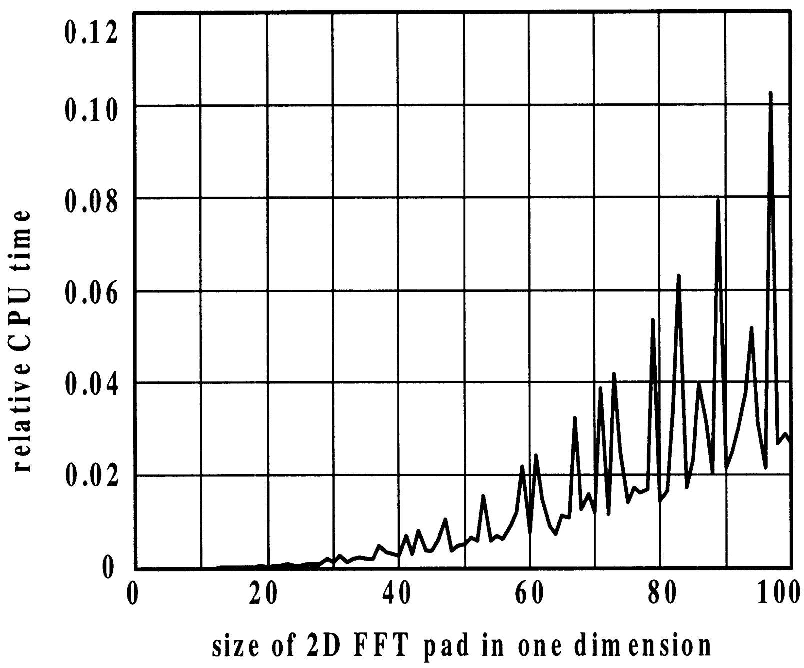


Figure 9: Relative CPU times for n=1 ... 100 .

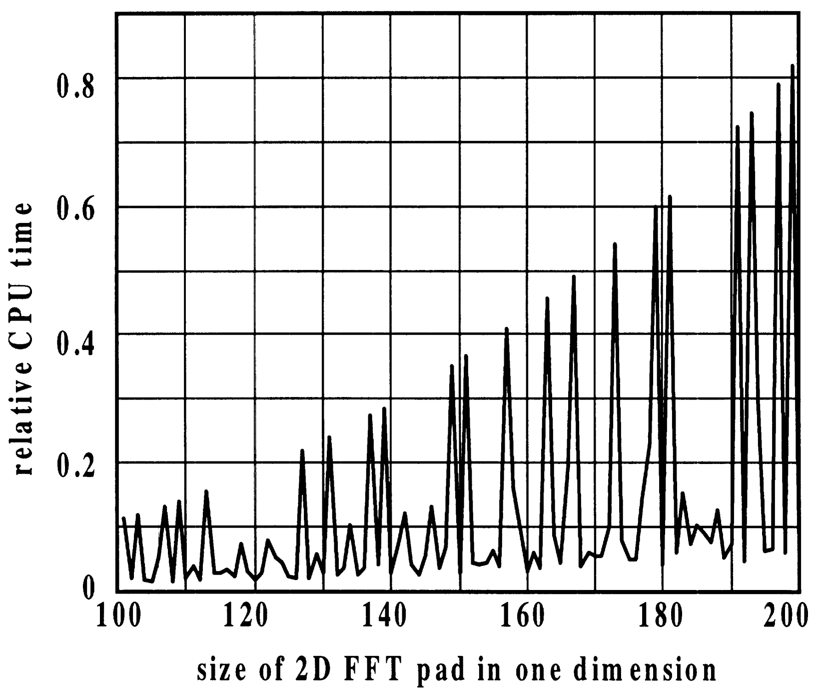


Figure 10: Relative CPU times for n=101 ... 200 .

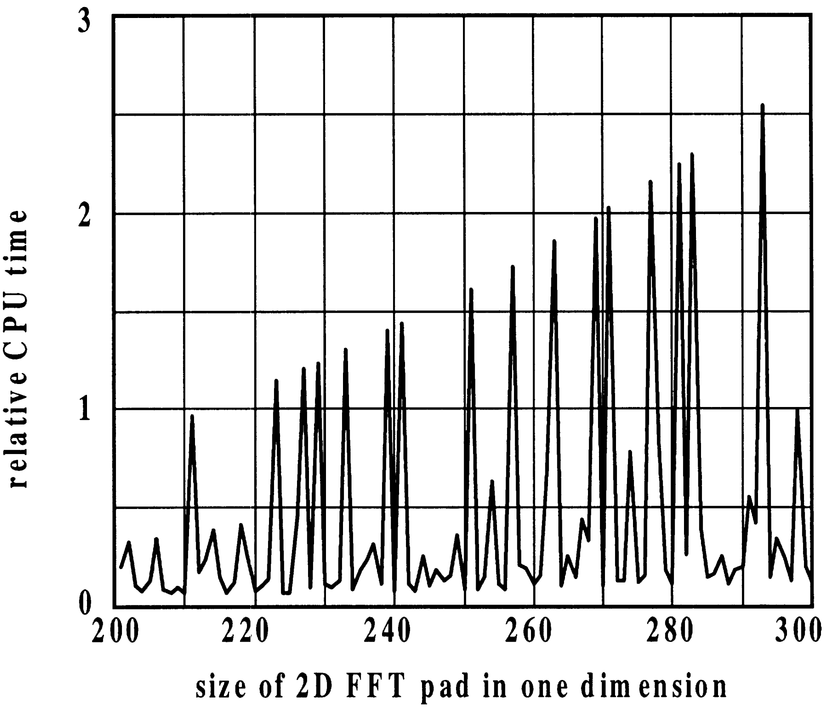


Figure 11: Relative CPU times for n=201 ... 300 .

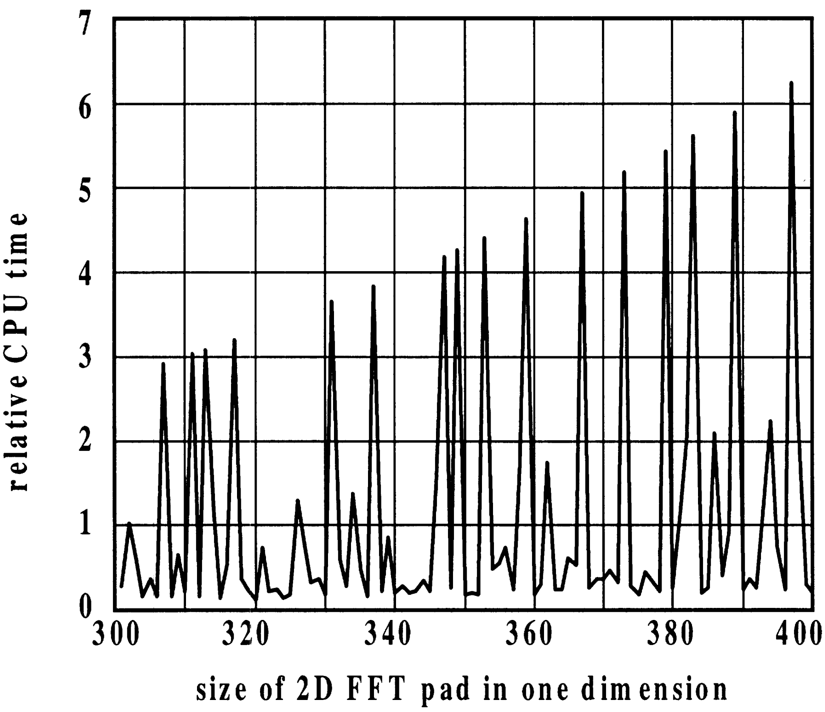


Figure 12: Relative CPU times for n=301 ... 400 .

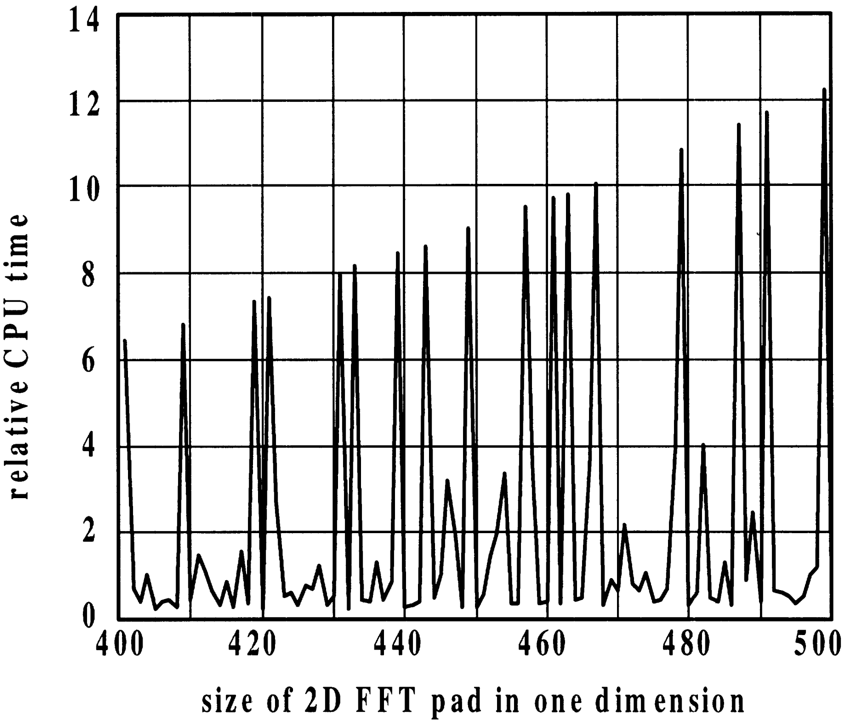


Figure 13: Relative CPU times for n=401 ... 500 .

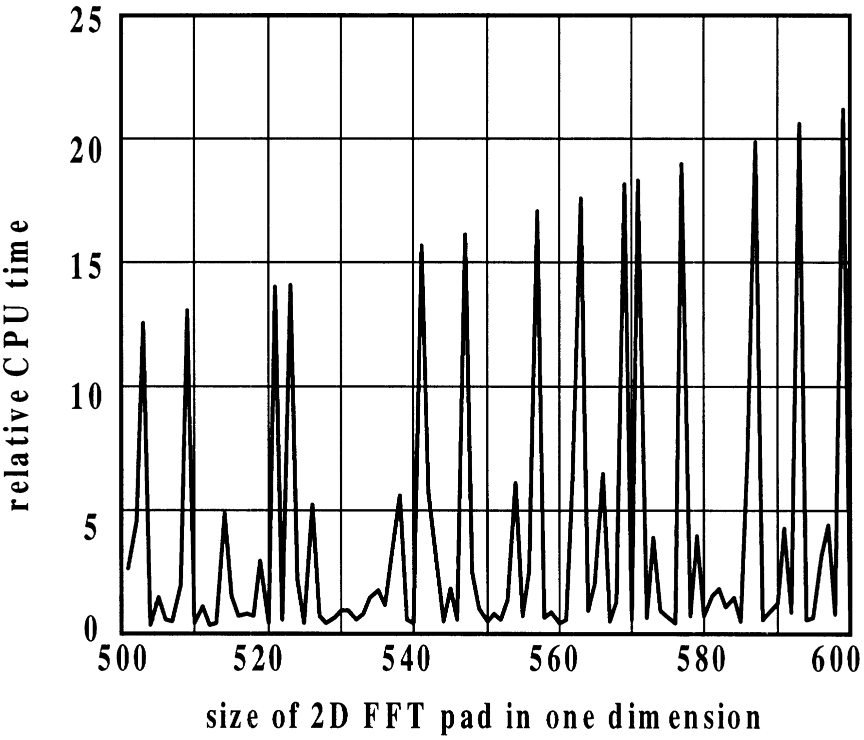


Figure 14: Relative CPU times for n=501 ... 600 .

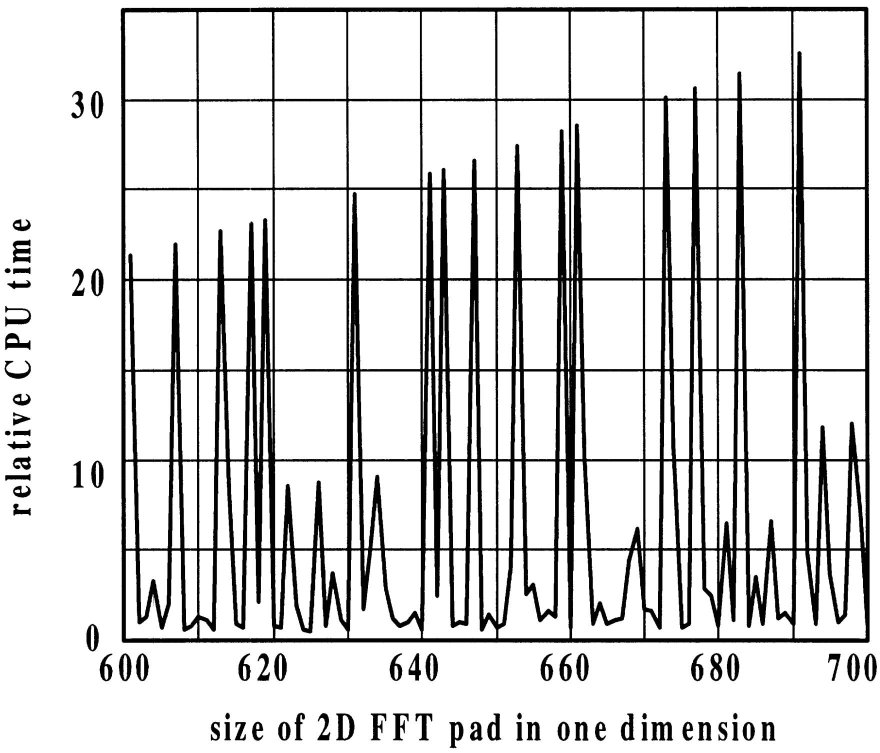


Figure 15: Relative CPU times for n=601 ... 700 .

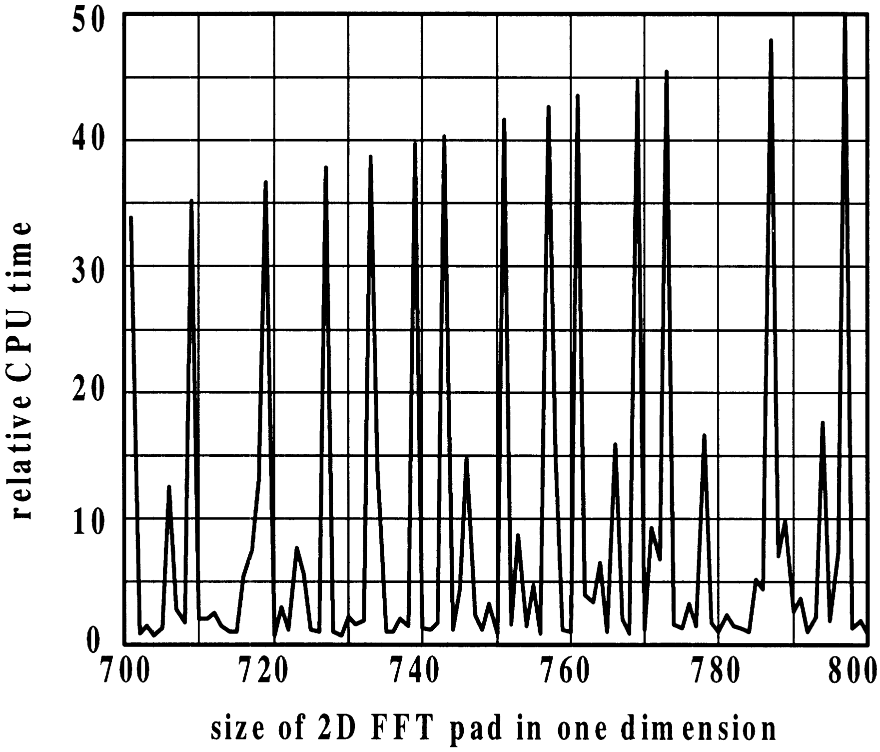


Figure 16: Relative CPU times for n=701 ... 800 .

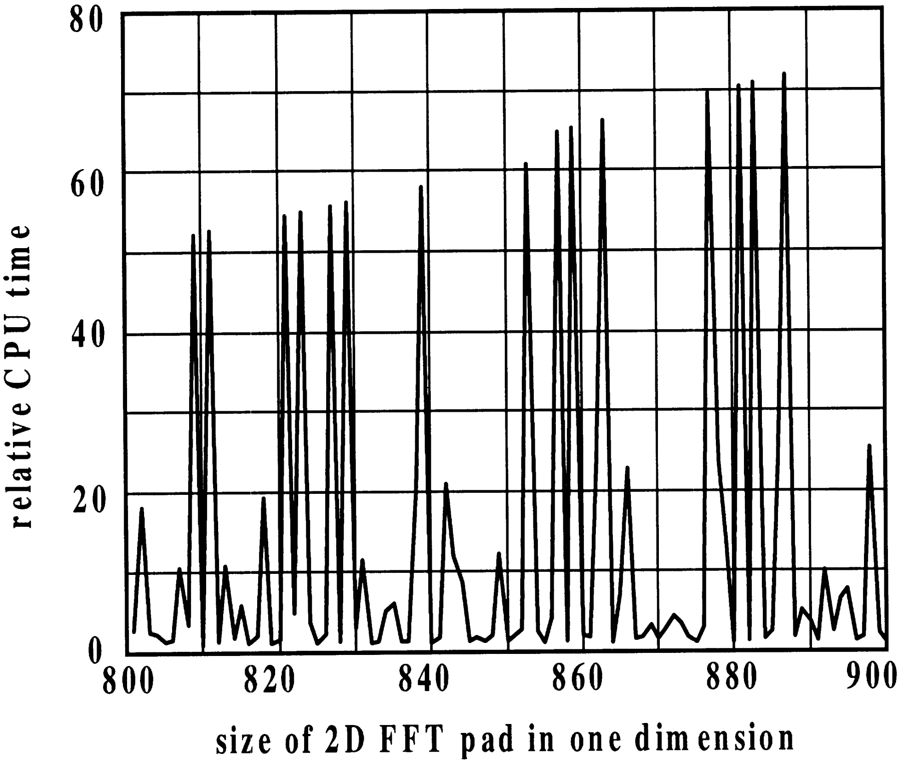


Figure 17: Relative CPU times for n=801 ... 900 .

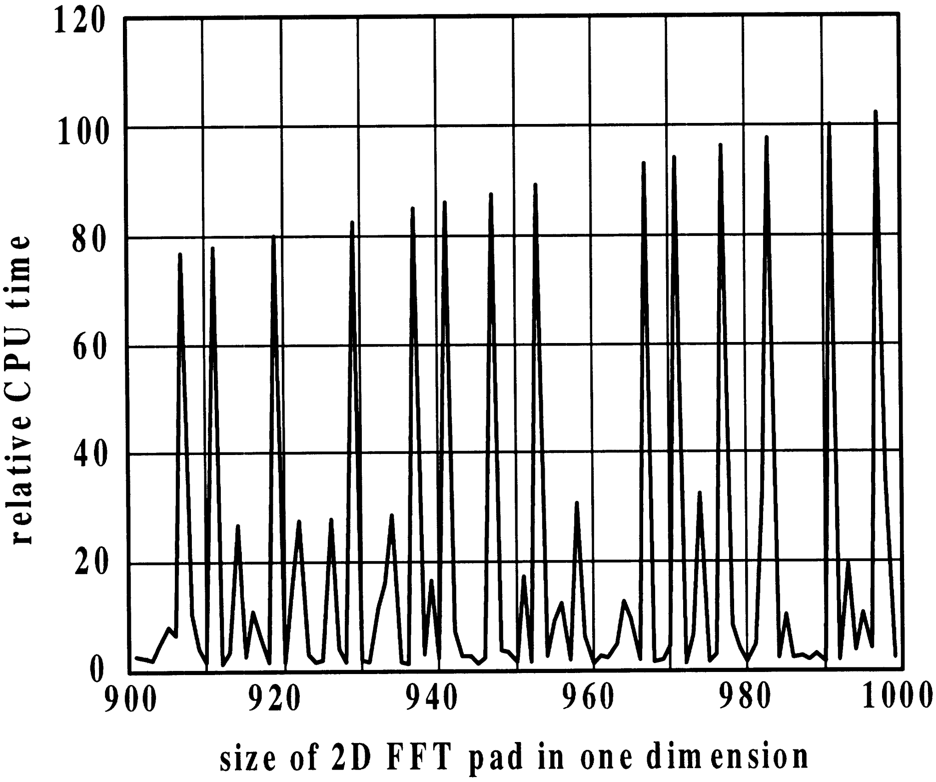


Figure 18: Relative CPU times for n=901 ... 999 .