Chapter 12
Pattern Calculation and Phase Analysis
Paul Wade W1GHZ ©1998,1999

12.0 Introduction

Antenna measurement is the most reliable way to verify antenna performance, but it is never easy to make accurate antenna measurements. When a new antenna or dish feed comes along, great claims are made, but how can we compare it to previous ones? While we might like to build and test every new design, it would soon become an overwhelming task. Finding time and good weather for antenna gain or pattern measurements isn't always easy. Sun noise measurements are a good alternative, but in New England, the sun only gets high enough for sun noise measurements about half the year.

On the other hand, if we can do initial comparisons with a computer and attempt to predict performance, we can avoid a lot of unnecessary work and only build the promising antennas. Then we can make measurements to verify that the performance meets predictions, and prove that it really works. We can also exchange computer models, try out different variations, and collaborate on improvements. And we can do the computer work in any season or weather.

With today's fast personal computers, it is possible to calculate the radiation pattern of most common feed antennas, including both amplitude and phase. While measurement of the phase pattern of an antenna is extremely difficult, it is impossible to calculate an antenna pattern without using phase — the electromagnetic field is described using complex vectors, which have both magnitude and phase. Once we have calculated the phase, why not extract it and make use of it? An antenna radiation pattern may be calculated using a personal computer with a fast Pentium™ or even faster Alpha™ microprocessor in a few minutes — a few years ago it would have taken longer even on a supercomputer, and at a prohibitive cost.

This chapter is not a thorough treatment of antenna analysis, but rather an overview with a few useful examples that could be a starting point.

12.1 Software

I have used two techniques to calculate antenna patterns. The first, for wire-like antennas and simple horns, uses the **NEC2** program[1] which uses the method-of-moments to calculate radiation patterns. The original **FORTRAN** program has phase information available in the output, unlike some of the derivative versions with Windows™ interfaces and graphical data entry and display. The **FORTRAN** code has been compiled for DOS for different memory sizes and is available[2] on the web.

For larger antennas like horns and dishes, I used Physical Optics (P.O.) routines from Milligan and Diaz[3]. (A description by Rusch[4]: "Physical optics, whereby the free-space dyadic Green's function is integrated over the geometrical-optics current distribution, is commonly used to analyze high-frequency reflectors, particularly, focusing reflectors.") The P.O. calculations require far less memory and computation time than **NEC2**, but require an understanding of the aperture current distribution; this is known for common structures like horns and dishes, but is not easily calculated for some of the more elaborate feeds. The techniques used in **NEC2** are applicable to arbitrary shapes, but require orders-of-magnitude larger compute times. In Chapter 6, several examples show good correlation between measured data and both modeling techniques, so we may use whichever is best suited for a particular antenna.

The P.O. routines are provided in two forms: **FORTRAN** and **MATLAB**[5]. These are not complete programs, but a set of routines that do bits of antenna analysis. The **FORTRAN** would have to be augmented with some IO routines into a complete program, while the **MATLAB** routines may be run sequentially and easily customized; I chose the latter route. Unfortunately, the **MATLAB™** software[5] is rather expensive, so I will not elaborate further (Note: the student version is restricted to matrix sizes which are much too small for these antenna models). If you have access to **MATLAB** and buy the book by Milligan and Diaz, I'll be glad to provide my customizations.

Of course, a computer model of an antenna is only an approximation of a real antenna, achieved by segmenting the antenna into a number of small pieces for purposes of calculation. The calculated patterns may be compared with published results and with measurements, which have their own inaccuracies. What we find, for a reasonably detailed model, is that the calculated forward patterns, out to about 90º rotation from the axis, are fairly accurate in amplitude and phase. The back half of the patterns, from 90º to 180º, are less accurate, particularly for the Physical Optics technique, which usually finds spurious sidelobes at about ±150º and a null at 180º. However, it is only the forward half of the feed pattern that illuminates a dish — even a very deep dish, with ***f*/D**=0.25, has an illumination angle of 180º, or ±90º from the axis. The back half of the pattern is just spillover that does not contribute to useful radiation. Thus the amplitude and phase of the spillover at any particular angle does not matter; only the total amount of power lost is needed for efficiency calculation. If the forward half of the pattern is accurate, then, by conservation of energy, the total power in the back half of the pattern is known, so we can also calculate antenna efficiency with reasonable accuracy.

12.2 Antenna modeling

The first step in making computer antenna simulations is to make an antenna model, a description that the software can understand. The newer **NEC** derivatives allow you to graphically draw a sketch of the antenna, then generate the actual model from the sketch. For **NEC2**, no such sophistication is available — we must describe the geometry as a series of three-dimensional coordinates. For example, a 1296 MHz dipole might be described as a wire from 0.049 meters below the tip of my nose to 0.049 meters above. My nose is just an arbitrary reference point, with x,y,z coordinates = 0,0,0. The three directions are, again arbitrarily, x = right, y = up, and z = forward; it doesn't really matter as long as you are consistent, but this orientation makes the *phi* and *theta* pattern orientations reported by **NEC2** come out right. The dipole would then be from 0, -0.049, 0 to 0, 0.049, 0. We could add a reflector, from 0, -0.057, 0.055 to 0, 0.057, 0.055; notice that it is longer than the dipole and is spaced 0.055 meters away. The dipole can be excited in the center, and we've just described a simple feed antenna. For **NEC2**, we can use the same description, but the syntax is much more rigorous — FORTRAN programs are notoriously fussy. We must also supply some additional details, like breaking up the wires into segments much smaller than a wavelength, so that **NEC2** can do calculations for each segment rather than a large structure.

Most of the antennas I have modeled are low-Q structures, like feedhorns. Low-Q structures are much more forgiving of small dimensional errors and modeling approximation. High-Q structures like Yagi-Uda antennas, on the other hand, are very sensitive to small differences in dimensions. As a result, specialized versions of **NEC**, like **Yagimax**, have been developed to deal with these antennas.

12.2.1 NEC2 example — dipole with rod reflector, or 2-element Yagi

We will start with a very simple example of an **NEC2** file for pattern calculation, a dipole with a rod reflector. The **NEC2** program is in **FORTRAN**, and so the file must pretend that it is a series of IBM punch cards – each line is equivalent to an 80-column card. Fortunately, the PC versions of **NEC2** aren't quite as fussy, and will accept a space between data fields instead of having to line up data in exact column format. Here is the file DIPROD.NEC:

```
CM 2el Yagi as simple feed DIPROD.NEC
CM leave out freq, do it in wavelengths
CM pointed in pos Z direction
CM in free space
CE
GW 1 21 0 -.214 0 0 .214 0  .01
GW 2 21 0 -.25 -.24 0 .25 -.24  .01
GS 0 0 1
GE 0
EK
EX 0 1 11 0 1 0
RP 0 19 3 1500 0 0 10 45
EN
```

That should be clear as mud! Let's try it again with some explanation:

CM lines are comments, ignored by program, so you can figure out what the file does

```
CM 2el Yagi as simple feed
CM leave out freq, do it in wavelengths
CM pointed in pos Z direction
CM in free space
```

CE line tells the program that the comments have ended

```
CE
```

GW is a wire:
GW (#)1 segments (from) X Y Z (to) X Y Z diameter
GW 1 is the dipole, GW 2 is the reflector, longer, and z = -0.24λ

```
GW 1 21 0 -.214 0 0 .214 0  .01
GW 2 21 0 -.25 -.24 0 .25 -.24  .01
```

GS is a scaling factor: all dimensions are multiplied by the third number. All dimensions are calculated in meters, so the scaling factor allows entry in other units.

```
GS 0 0 1
```

GE 0 line tells the program that the geometry is finished

```
GE 0
```

EK specifies the "Extended Thin-Wire Kernel" - for very thin wires

```
EK
```

EX is the excitation: 1 volt at segment #11 of 21, or middle of dipole
0 — voltage source
1 — tag number of source segment — line starting GW 1
11 — 11th segment of tag number 1 — 11 of 21 is center
0 — no action for this parameter
1 — 1 volt excitation

```
EX 0 1 11 0 1 0
```

RP specifies the output Report:
0
19 theta cuts — 0 to 180 degrees at 10 degree increments (below)
3 phi cuts — 0, 45, and 90 degrees (below)
1500 some arcane parameters
0
0
10 degree theta increments
45 degree phi increments

```
RP 0 19 3 1500 0 0 10 45
```

EN line ends program

```
EN
```

I'm sure that didn't clear things up much, but is should provide the flavor of an **NEC2** file. In fact, my explanations may not even be right, but these are the incantations that worked! The real explanation may be found in the **NEC2** instruction manual; I found it online at http://members.home.net/nec2/ A printed version may be ordered from the US government.

12.3 Running the model

Once the antenna model is complete, we'd like to run it and see how it works (or usually doesn't, on the initial attempt!). After downloading and unzipping **NEC2**, we run it from a **DOS** prompt:

```
>NEC2D
  [ some words ]

ENTER NAME OF INPUT FILE > mymodel.nec
ENTER NAME OF OUTPUT FILE> mymodel.out
>
```

For our simple dipole feed, the whole run takes less than a second on any sort of Pentium™ PC. Then we can look at the patterns in the output file, and we can look at the geometry to see if **NEC2** interpreted it correctly. I use a shareware viewer called **NECDRAW**[6]:

```
>NECDRAW MYMODEL.OUT mymodel.err
```

The second filename, *mymodel.err*, is optional, but it provides a list of any errors or warnings, which can be helpful. If there is an error in the model that prevents it from running, then you'll be in for some headscratching, because **NEC2** error messages are pretty cryptic.

The "D" in **NEC2D** is for double precision, which allows for more accurate floating-point computation; some of the **BASIC** versions of **NEC** may not have this accuracy. One of the problems with computer arithmetic is that very small errors can accumulate into a significant error, especially in long calculations like those needed for large antennas.

The other PC versions have a number after the "D" indicating how many segments they can handle — larger models will need more segments. They also require more memory — lots of RAM. From my experience, 64 Meg of RAM will run the **NEC2D960** version for up to 960 segments, while 128M is needed for the largest version, **NEC2D2K8** for up to 2800 segments. The large versions take much longer to load memory, so don't use more than you need – the 512 segment version, **NEC2D512**, is suitable for many smaller antennas and requires less than 16M of RAM. Compute time is also a function of size: a small model may run in less than a second, but the larger ones need tens of minutes on a fast PC.

12.4 Interpreting the NEC2 output file

After an **NEC2** run is complete, the output is in a long file full of numbers. The useful information is near the bottom; use a text editor to search for “RADIATION PATTERN.” The pattern information we need starts here. The simple amplitude pattern is at the very bottom, after the heading “NORMALIZED GAIN,” but we need the detailed radiation pattern to also determine the phase. The following table is the useful part of the output file, DIPROD.OUT from our simple dipole feed; I have added the color-coding for the explanation that follows.

						RADIATION PATTERN					
	ANGLES - - -		POWER	GAINS -	- - -	POLA	RIZATION	E(THETA)		E(PHI)	
THETA	PHI	VERT.	HOR.	TOTAL	AXIAL	TILT	SENSE	MAGNITUD	PHASE	MAGNITUD	PHASE
DEGREES	DEGREES	DB	DB	DB	RATIO	DEG.		VOLTS/M	DEGREES	VOLTS/M	DEGREES
											H-plane phase
0	0	-999.99	5.61	5.61	0	90	LINEAR	0.00E+00	0	1.24E+00	-88.41
10	0	-999.99	5.59	5.59	0	90	LINEAR	0.00E+00	0	1.24E+00	-87.94
20	0	-999.99	5.51	5.51	0	90	LINEAR	0.00E+00	0	1.23E+00	-86.54
30	0	-999.99	5.37	5.37	0	90	LINEAR	0.00E+00	0	1.21E+00	-84.28
40	0	-999.99	5.13	5.13	0	90	LINEAR	0.00E+00	0	1.17E+00	-81.28
50	0	-999.99	4.76	4.76	0	90	LINEAR	0.00E+00	0	1.12E+00	-77.69
60	0	-999.99	4.22	4.22	0	90	LINEAR	0.00E+00	0	1.06E+00	-73.76
70	0	-999.99	3.48	3.48	0	90	LINEAR	0.00E+00	0	9.71E-01	-69.8
80	0	-999.99	2.49	2.49	0	90	LINEAR	0.00E+00	0	8.67E-01	-66.25
90	0	-999.99	1.23	1.23	0	90	LINEAR	0.00E+00	0	7.50E-01	-63.73
100	0	-999.99	-0.31	-0.31	0	90	LINEAR	0.00E+00	0	6.27E-01	-63.11
110	0	-999.99	-2.09	-2.09	0	90	LINEAR	0.00E+00	0	5.11E-01	-65.64
120	0	-999.99	-3.89	-3.89	0	90	LINEAR	0.00E+00	0	4.16E-01	-72.77
130	0	-999.99	-5.23	-5.23	0	90	LINEAR	0.00E+00	0	3.56E-01	-84.86
140	0	-999.99	-5.68	-5.68	0	-90	LINEAR	0.00E+00	0	3.38E-01	-98.95
150	0	-999.99	-5.34	-5.34	0	-90	LINEAR	0.00E+00	0	3.52E-01	-110.48
160	0	-999.99	-4.76	-4.76	0	-90	LINEAR	0.00E+00	0	3.76E-01	-117.74
170	0	-999.99	-4.32	-4.32	0	-90	LINEAR	0.00E+00	0	3.96E-01	-121.45
180	0	-999.99	-4.16	-4.16	0	-90	LINEAR	0.00E+00	0	4.03E-01	-122.57
0	45	2.6	2.6	5.61	0	45	LINEAR	8.78E-01	-88.41	8.78E-01	-88.41
10	45	2.42	2.55	5.49	0	45.44	LINEAR	8.59E-01	-87.93	8.72E-01	-87.93
20	45	1.85	2.39	5.14	0	46.78	LINEAR	8.05E-01	-86.54	8.57E-01	-86.54
30	45	0.87	2.11	4.55	0	49.11	LINEAR	7.19E-01	-84.3	8.30E-01	-84.3
40	45	-0.6	1.72	3.72	0	52.55	LINEAR	6.07E-01	-81.33	7.93E-01	-81.33
50	45	-2.66	1.18	2.68	0	57.27	LINEAR	4.79E-01	-77.82	7.45E-01	-77.82
60	45	-5.53	0.49	1.46	0	63.43	LINEAR	3.44E-01	-74.01	6.89E-01	-74.01
70	45	-9.69	-0.37	0.11	0	71.12	LINEAR	2.13E-01	-70.22	6.24E-01	-70.22
80	45	-16.62	-1.41	-1.29	0	80.15	LINEAR	9.60E-02	-66.87	5.53E-01	-66.87
90	45	-999.99	-2.67	-2.67	0	-90	LINEAR	2.44E-12	115.45	4.78E-01	-64.55
100	45	-19.35	-4.14	-4.01	0	-80.15	LINEAR	7.02E-02	115.91	4.04E-01	-64.09
110	45	-15.09	-5.77	-5.29	0	-71.12	LINEAR	1.15E-01	113.35	3.35E-01	-66.65
120	45	-13.4	-7.37	-6.41	0	-63.43	LINEAR	1.39E-01	106.48	2.78E-01	-73.52

130	45	-12.39	-8.55	-7.05	0	-57.27	LINEAR	1.56E-01	94.96	2.43E-01	-85.04
140	45	-11.21	-8.89	-6.89	0	-52.55	LINEAR	1.79E-01	81.35	2.34E-01	-98.65
150	45	-9.74	-8.49	-6.06	0	-49.11	LINEAR	2.12E-01	69.9	2.45E-01	-110.1
160	45	-8.38	-7.84	-5.1	0	-46.78	LINEAR	2.48E-01	62.48	2.64E-01	-117.52
170	45	-7.48	-7.35	-4.4	0	-45.44	LINEAR	2.75E-01	58.61	2.79E-01	-121.39
180	45	-7.17	-7.17	-4.16	0	-45	LINEAR	2.85E-01	57.43	2.85E-01	-122.57
									E-plane phase		
0	90	5.61	-999.99	5.61	0	0	LINEAR	1.24E+00	-88.41	6.34E-12	91.59
10	90	5.4	-999.99	5.4	0	0	LINEAR	1.21E+00	-87.93	6.28E-12	92.07
20	90	4.75	-999.99	4.75	0	0	LINEAR	1.12E+00	-86.54	6.10E-12	93.46
30	90	3.63	-999.99	3.63	0	0	LINEAR	9.88E-01	-84.31	5.83E-12	95.69
40	90	2.01	-999.99	2.01	0	0	LINEAR	8.20E-01	-81.38	5.46E-12	98.62
50	90	-0.22	-999.99	-0.22	0	0	LINEAR	6.34E-01	-77.95	5.04E-12	102.05
60	90	-3.25	-999.99	-3.25	0	0	LINEAR	4.48E-01	-74.26	4.57E-12	105.74
70	90	-7.53	-999.99	-7.53	0	0	LINEAR	2.73E-01	-70.63	4.08E-12	109.37
80	90	-14.53	-999.99	-14.53	0	0	LINEAR	1.22E-01	-67.49	3.59E-12	112.51
90	90	-999.99	-999.99	-999.99	0	0		3.11E-12	114.62	3.11E-12	114.62
100	90	-17.17	-999.99	-17.17	0	0	LINEAR	9.01E-02	114.93	2.65E-12	114.93
110	90	-12.76	-999.99	-12.76	0	0	LINEAR	1.50E-01	112.38	2.23E-12	112.38
120	90	-10.88	-999.99	-10.88	0	0	LINEAR	1.86E-01	105.78	1.90E-12	105.78
130	90	-9.7	-999.99	-9.7	0	0	LINEAR	2.13E-01	94.8	1.69E-12	94.8
140	90	-8.41	-999.99	-8.41	0	0	LINEAR	2.47E-01	81.66	1.65E-12	81.66
150	90	-6.87	-999.99	-6.87	0	0	LINEAR	2.95E-01	70.28	1.74E-12	70.28
160	90	-5.45	-999.99	-5.45	0	0	LINEAR	3.48E-01	62.7	1.89E-12	62.7
170	90	-4.49	-999.99	-4.49	0	0	LINEAR	3.88E-01	58.67	2.01E-12	58.67
180	90	-4.16	-999.99	-4.16	0	0	LINEAR	4.03E-01	57.43	2.06E-12	57.43

NORMALIZED
GAIN
TOTAL
GAIN
NORMALIZATION FACTOR 5.61 DB

	H-plane amplitude						E-plane amplitude	
	ANGLES - -	GAIN		ANGLES	GAIN		ANGLES -	GAIN
THETA	PHI	DB	THETA	PHI	DB	THETA	PHI	DB
DEGREES	DEGREES		DEGREES	DEGREES		DEGREES	DEGREES	
0	0	0	0	45	0	0	90	0
10	0	-0.02	10	45	-0.12	10	90	-0.21
20	0	-0.1	20	45	-0.47	20	90	-0.87
30	0	-0.24	30	45	-1.07	30	90	-1.98
40	0	-0.49	40	45	-1.89	40	90	-3.6
50	0	-0.86	50	45	-2.93	50	90	-5.83
60	0	-1.39	60	45	-4.15	60	90	-8.86
70	0	-2.13	70	45	-5.5	70	90	-13.14
80	0	-3.12	80	45	-6.9	80	90	-20.14
90	0	-4.38	90	45	-8.28	90	90	-1005.6
100	0	-5.93	100	45	-9.62	100	90	-22.78
110	0	-7.7	110	45	-10.9	110	90	-18.37
120	0	-9.5	120	45	-12.02	120	90	-16.49
130	0	-10.85	130	45	-12.66	130	90	-15.31
140	0	-11.29	140	45	-12.5	140	90	-14.02
150	0	-10.95	150	45	-11.67	150	90	-12.48
160	0	-10.37	160	45	-10.71	160	90	-11.06
170	0	-9.93	170	45	-10.01	170	90	-10.1
180	0	-9.77	180	45	-9.77	180	90	-9.77

At the very bottom, we see the simple amplitude-only patterns for this antenna. The red numbers are for the **E-plane**, corresponding to the red curves in our pattern plots, and the blue numbers are for the **H-plane**, corresponding to the blue curves in our pattern plots. The yellow highlighted columns are for orientation: note that the **H-plane** is at a PHI angle of 0, while the **E-plane** is at a PHI angle of 90. There is also a PHI angle of 45°, which we do not use for dish analysis, but is worth a glance to see if there are any strange sidelobes.

We find the phase for each plane in the upper "RADIATION PATTERN" table by using the PHI angles – I've added the color coding to help, and green column headings as well. Using these phase angles without further calculation is possible only if the polarization is LINEAR, or close to it; note that the cross polarized amplitude in this example, highlighted in yellow, is either 0.00E+00 (= ZERO) or extremely small, like 6.84E-12, twelve orders of magnitude (120 dB) down. Some other feeds don't have quite as pure polarization as a dipole, but several orders of magnitude is enough to make the phase error negligible.

To use this data for analysis with the **PHASEPAT** program, it must be reduced to separate files for the **E-plane** and the **H-plane**. Here is the **H-plane** file, DIPROD_H.DAT, after sorting the data:

```
0         0.00   -88.41
10         0.02   -87.94
20         0.10   -86.54
30         0.24   -84.28
40         0.49   -81.28
50         0.86   -77.69
60         1.39   -73.76
70         2.13   -69.80
80         3.12   -66.25
90         4.38   -63.73
100         5.93   -63.11
110         7.70   -65.64
120         9.50   -72.77
130        10.85   -84.86
140        11.29   -98.95
150        10.95  -110.48
160        10.37  -117.74
170         9.93  -121.45
180         9.77  -122.57
// H-plane
// dipole with rod reflector feed
// (2 element yagi)
// dipole .428 wl long, .01 wl dia
// spaced .24 wl from 0.5 wl long reflector
// by NEC 7/21/98
```

The three columns are rotation angle, amplitude from the "NORMALIZED GAIN" table, and phase from the "RADIATION PATTERN" table; for the **H-plane**, the blue columns are used. The lines at the bottom are comments that I added – the program ignores anything preceded by "//". To extract the data from the **NEC2** output file, I use a text editor that is capable of cutting and pasting blocks and columns; an alternative would be to load sections of the file into a spreadsheet, like Excel™, and manipulate the data in the spreadsheet.

Of course, I eventually automated the process of extracting the radiation pattern data from the **NEC2** output file, using a **Perl** script, **nec2pat.pl**. Perl is an interpreted language designed for processing lists easily, but you must run the interpreter to use it:

```
>perl nec2pat.pl
```

a series of prompts will complete the process. Perl may be downloaded from www.activestate.com.

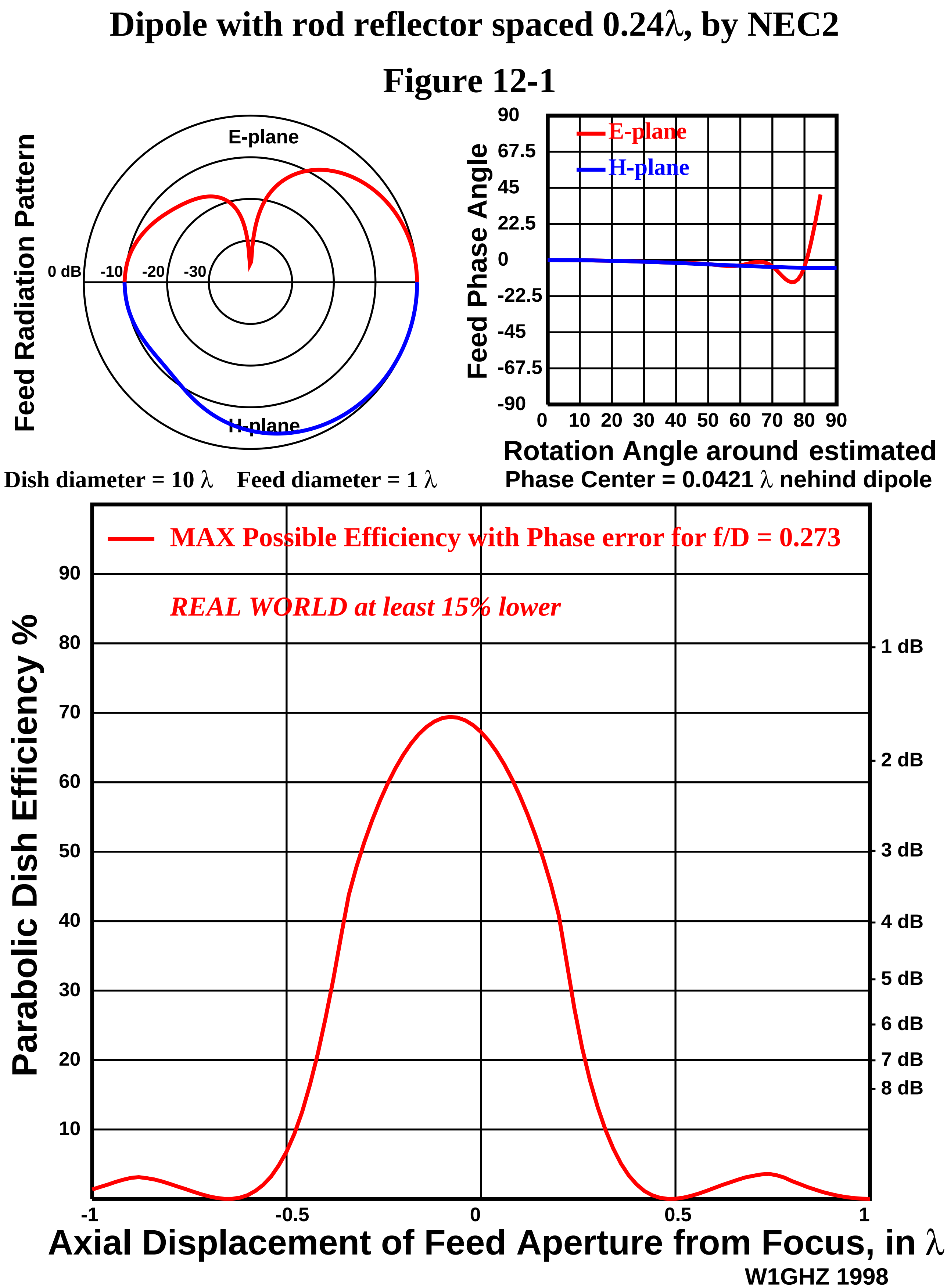
12.5 Interpreting the output

In Chapter 11, we saw how to use the **FEEDPATT** program to process amplitude-only radiation patterns for dish feeds and calculate dish efficiency. Once we add phase data to the patterns, we can use it to enhance the efficiency calculations and also to calculate phase center. The phase data is added to the E- and H-plane data files as a third column, after rotation angle and amplitude in dB; the phase should be in degrees. The **PHASEPAT** program uses these data files, with two output options: efficiency and phase center. Each option produces a graphical output. The normal sequence is to make a plot of phase center for some illumination angle, then use the plot to determine the best phase center. Figure 12-1 is the phase center plot for our dipole feed example; the best phase center is clearly at the peak of the curve, 0.06λ behind the dipole. Then the efficiency option is run, entering the phase center just obtained to produce a plot of dish efficiency, like Figure 12-2.

For some feeds, the phase center can vary significantly with illumination angle. In Chapter 6, we attempted to find the combination of phase center and ***f*/D** that gave best performance for each feed. However, if you are fitting a feed to a specific dish, the phase center calculations should use the illumination angle corresponding to the ***f*/D** of that dish.
Before we can trust the calculated phase center and efficiency, we should question the accuracy of the model. One good test is to examine the calculated radiation patterns in the **PHASEPAT** plots. Do they make sense? Are they counterintuitive? Are there large unexpected lobes or wildly varying phase? If the patterns don't feel right, then it is time to look for problems with the model. The **NECDRAW** error file may show a problem. If not, then examine the model carefully. Sometimes changing some patch sizes or wire segmentations and making another run can improve results or highlight a problem. For instance, having a large number of identical surface patches seems to create a spurious sidelobe pattern that is smoothed out by adding a few random variations in patch size.

12.6 More advanced models

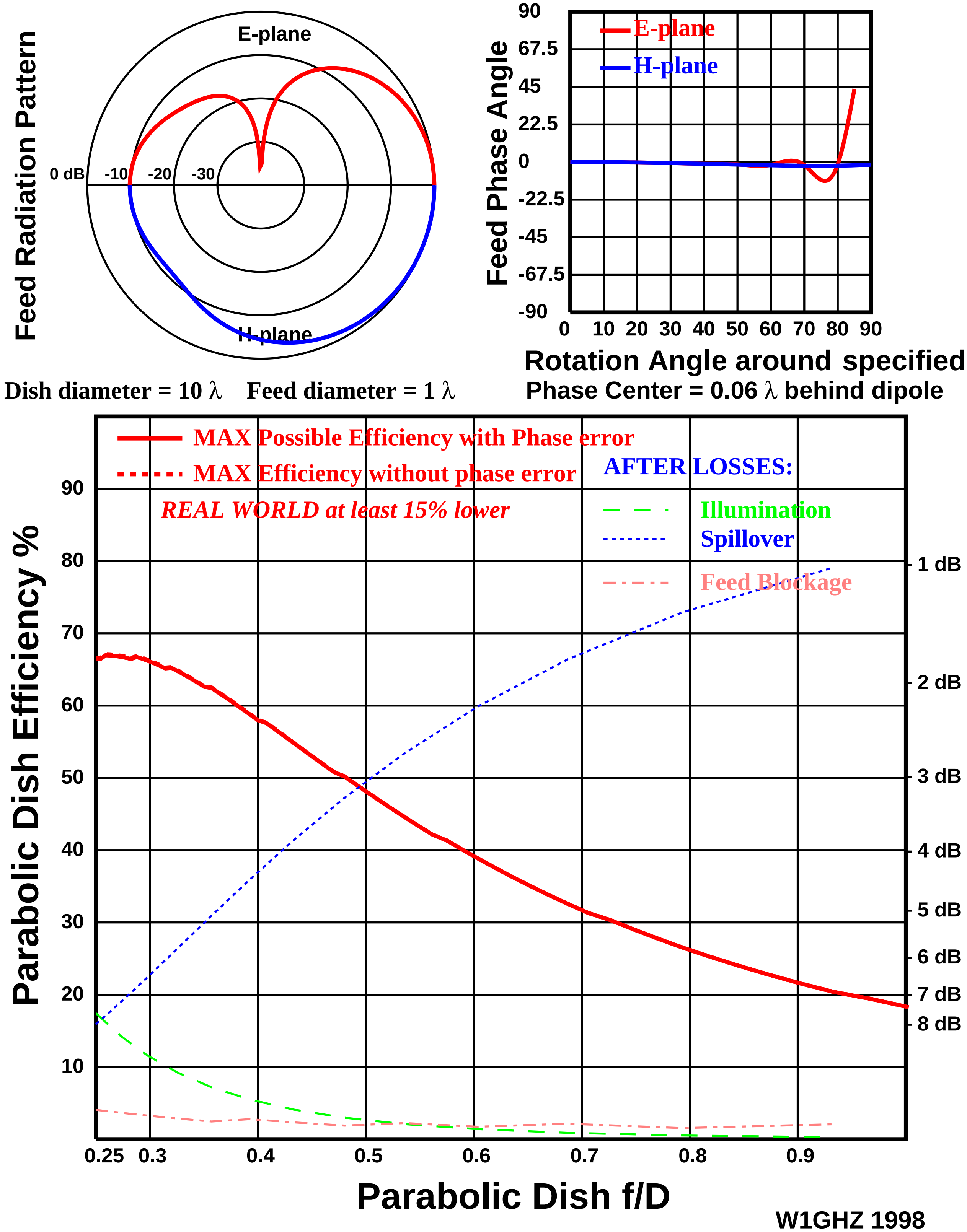
A logical extension of our simple dipole-reflector feed above replaces the rod reflector with a disk reflector, ½ λ in diameter, as seen in Figure 6.2-2. **NEC2** has no model for a metal plate, so we must simulate the reflector with a fine mesh of wires – we know that a mesh reflector is equivalent to solid surface if the holes are small enough. It is possible to form a circular disk using arcs of wire, but I had not figured them out when I modeled this feed, so instead I approximated the circle with a hexagon of straight wires connected together. The electrical connection is modeled by using the same endpoint for two or more wires. The only difference between the **NEC2** input file for this feed and the previous one is the reflector, so I've added explanations for that section only, *in red.*



Dipole with rod reflector spaced 0.24λ, by NEC2
Figure 12-1
Feed Radiation Pattern
E-plane
H-plane
0 dB
-10
-20
-30
Feed Phase Angle
90
67.5
45
22.5
0
-22.5
-45
-67.5
-90
0
10
20
30
40
50
60
70
80
90
E-plane
H-plane
Rotation Angle around estimated
Phase Center = 0.0421 λ nehind dipole
Dish diameter = 10 λ
Feed diameter = 1 λ
MAX Possible Efficiency with Phase error for f/D = 0.273
REAL WORLD at least 15% lower
Parabolic Dish Efficiency %
90
80
70
60
50
40
30
20
10
1 dB
2 dB
3 dB
4 dB
5 dB
6 dB
7 dB
8 dB
-1
-0.5
0
0.5
1
Axial Displacement of Feed Aperture from Focus, in λ
W1GHZ 1998

Dipole with rod reflector spaced 0.24λ, by NEC2

Figure 12-2



12.6.1 NEC2 example — dipole with splashplate

Only differences from previous example are explained

```
CM dipole over 1/2 wave splasher feed DIPSPLSH.NEC
CM dimensions from RSGB Microwave Vol3 p14.25
CM leave out freq, do it in wavelengths
CM pointed in pos Z direction
CM in free space
CE
```

GW 1 thru 12 build up a wire mesh of 1/4 of a disk reflector
Notice how the coordinates of one end of a wire are
Are the other end of the next one, making a connection

```
GW 1 10 0.25 0 -0.3 .1768 .1768 -0.3 .001
GW 2 10  .1768 .1768 -0.3 0 0.25 -0.3 .001
GW 3 10 0.1875 0 -0.3 .1326 .1326 -0.3 .001
GW 4 10  .1326 .1326 -0.3 0 0.1875 -0.3 .001
GW 5 10 0.125 0 -0.3 .0884 .0884 -0.3 .001
GW 6 10  .0884 .0884 -0.3 0 0.125 -0.3 .001
GW 7 10 0.0625 0 -0.3 .0442 .0442 -0.3 .001
GW 8 10  .0442 .0442 -0.3 0 0.0625 -0.3 .001
GW 9 10 0 0 -0.3 0.0442 0.0442 -0.3 .001
GW 10 10 0.0442 0.0442 -0.3 .0884 .0884 -0.3 .001
GW 11 10 .0884 .0884 -0.3 .1326 .1326 -0.3 .001
GW 12 10 .1326 .1326 -0.3 .1768 .1768 -0.3 .001
```

GX Reflection in coordinate plane - adds a mirror image
12 is the tag number increment (start at 13, 25, etc)
110 reflection in both X and Y but not Z axis

```
GX 12 110
```

GW 51 is dipole, higher tag number to allow for duplications
Reflector is added first, duplicated, then dipole added afterward

```
GW 51 9 0 -.225 0 0 .225 0  .027
GS 0 0 1
GE 0
EK
EX 0 51 5 0 1 0
RP 0 19 3 1500 0 0 10 45
EN
```

12.6.2 NEC2 example — coffee-can feed

The easiest way to make an antenna model for **NEC2** is to start with a working model for a similar antenna and modify it. For complex models, make the modifications in steps, so that when it crashes, you'll know that it was the last change that did it. I learned to make models for circular waveguide feeds, like those in Chapter 6-3, by starting with a model of a VE2MA feed provided by Peter, PA3AEF. A simpler starting point is a coffee-can feed, or cylindrical horn. We model this by describing a narrow strip down one side, then duplicating the strip while rotating the copies to form a circle. For **NEC2**, the strip is formed by connecting a number of *surface patchs*, which are polygons much smaller than a wavelength. Patches are described by the coordinates of their corner points, and are connected together when they have the same corner points. They must be small because the software makes the assumption that current is constant across the patch. Since each patch is a flat surface, our circle is approximated by a number of straight segments to form a polygon — Peter used 18, which seems to be a pretty good approximation of a circle.

The dipole example used another simplification — no frequency was specified, so the wavelength is assumed to be one meter. As long as *all* dimensions are scaled proportionally, a model can be scaled to any frequency and yield the same radiation pattern. For convenience, I scale all my models to 1296 MHz; at this frequency, dimensions are reasonable and easy to type; a precision of 0.0001 meter, or 0.1 mm, is more than adequate. In the following model, the line starting "FR" specifies the frequency as 1296 MHz.

Another point to note is the feed, a monopole acting as a probe in the waveguide. We would like the coaxial ground return for this monopole to be the wall of the can – **NEC2** requires that the monopole start in the *center* of a surface patch to make this happen. It is easy to model a monopole or dipole that goes right through the wall of the waveguide, but rather hard to do it in metal without cutting a hole first. When the output file is viewed using **NECDRAW**, a properly connected monopole will have four dots around its base, like the mounting holes of a coax socket.

Strategy: make a strip of surface patches down one edge, centered on the Y-axis,
With X-dimension = radius = 88 mm

```
CM COFFEE.NEC
CM Coffee can feed - open cylindrical waveguide
CM Horn 300mm long, 176mm diameter
CM by W1GHZ from PA3AEF
CE **********************************************
```

SP surface patch
0 blank parameter
3 quadrilateral patch shape
.088 -.01394 .300 coordinates of first corner
.088 .01394 .300 coordinates of second corner

```
SP  0    3      .088   -.01394      .300      .088   0.01394      .300
```

SC surface patch continuation — third and fourth corners of first patch
Note change in Z dimension

```
SC   0    3      .088   0.01394      .260      .088   -.01394      .260
```

SC surface patch continuation — third and fourth corners of second patch,
First and second corners are third and fourth of previous patch
SC pattern continues to make a string of patches with only Z dimension changing
SC patches turn corner to make shorted end of can

```
SC   0    3      .088   0.01394      .220      .088   -.01394      .220
SC   0    3      .088   0.01394      .180      .088   -.01394      .180
SC   0    3      .088   0.01394      .150      .088   -.01394      .150
SC   0    3      .088   0.01394      .120      .088   -.01394      .120
SC   0    3      .088   0.01394      .080      .088   -.01394      .080
SC   0    3      .088   0.01394      .040      .088   -.01394      .040
SC   0    3      .088   0.01394      .000      .088   -.01394      .000
```

SC patches turn corner to make shorted end of can

```
SC   0    3      .035   0.00554      .000      .035   -.00554      .000
```

GM coordinate transformation
0 tag number increment - surface patches don't need tag numbers
4 new structures to be generated — duplicates of string of patches
0 X-axis rotation
0 Y-axis rotation
18 degrees Z-axis rotation — new strips are rotated in 18º increments about Z-axis
total of 5 strips 18 degrees wide adds up to 90 degrees, a quarter circle

```
GM   0    4      .000   0.0          18.0
```

GM this line rotates everything another 9 degrees, to line up with Y-axis

```
GM   0    0      .000   0.0           9.0
```

SP surface patch — to close hole in shorted end of can
0 blank parameter
0 arbitrary patch shape

```
SP   0    0      .015   0.015        .000      90.0     .00000  .0009701
```

GX Reflection in coordinate plane - adds a mirror image of everything so far
0 is the tag number increment
110 reflection in both X and Y but not Z axis

```
GX   0  110
```

GW wire at 45 degrees for probe starting at wall of can

```
GW   1    4   .0622  .0622       .100     .0198     .0198      .100
.002
```

GM coordinate transformation
0 tag number increment - surface patches don't need tag numbers
0 new structures to be generated
0 X-axis rotation
0 Y-axis rotation
45 degrees Z-axis rotation to get probe in E-plane
0 meters translation in X-direction
0 meters translation in Y-direction

-0.300 meters translation in Z-direction — move aperture to 0,0,0

```
GM  0    0       0.0       0.0     45.0  0 0 -0.300
GE
```

FR frequency in Megahertz
0 linear steps
1 frequency step (single frequency)
0 required blank parameter
0 required blank parameter
1296.0 Megahertz

```
FR  0    1    0    0    1296.0
```

EX excited at segment 1 (end) of tag #1

```
EX  0    1    1    0       1.0         0
```

LD loading - specifies wire conductivity in mhos/meter

```
LD  5    0    0    0  3.72E+07
```

PT -1 is supposed to suppress printing of a lot of stuff in output file,
But doesn't seem to work

```
PT -1
```

RP specifies the output Report:
0
37 theta cuts — 0 to 360 degrees at 10 degree increments (below) — looking for asymmetry
3 phi cuts — 0, 45, and 90 degrees (below)

```
RP  0  37    3 1500       0.0       0.0       10.0      45.0
EN
```

12.6.3 NEC2 example — VE4MA feed

The VE4MA feed model is an extension of the coffee-can feed model above, plus the addition of a choke flange as shown in Figure 6.3-6. Just as we started the circular waveguide horn model with a strip down one side, we add a strip of the choke flange, then duplicate the whole thing in rotation into segments of a circle. To connect the choke flange to the wall of the horn, the end points of a choke surface patch must be the same points as the corners of a surface patch forming the horn. Two patches having one or more common corner points are considered to be connected together, but two patches that intersect at any other point are not.

The VE4MA feed is a coffee-can feed with a choke ring added

```
CM Dish-Feed VE4MA.NEC
CM Horn 300mm long, shorted end 176mm diameter
CM at 'mouth' cylinder 120mm long, 420mm diameter
CM horn flush with choke ring
CM by PA3AEF, modified by W1GHZ
CM smaller patches 11/98
CE **********************************************
SP  0    3    .210  -0.03326  .300      .210    0.03326    .300
```

Choke ring starts here, 210 mm radius

```
SC  0    3    .210  0.03326   .277      .210   -0.03326    .277
SC  0    3    .210  0.03326   .254      .210   -0.03326    .254
SC  0    3    .210  0.03326   .231      .210   -0.03326    .231
```

```
SC  0    3    .210  0.03326   .208      .210   -0.03326    .208
SC  0    3    .210  0.03326   .180      .210   -0.03326    .180
```

Surface patches turn corner for closed end of choke ring

```
SC  0    3    .170  0.02693   .180      .170   -0.02693    .180
SC  0    3    .130  0.02059   .180      .130   -0.02059    .180
```

Down to same radius as waveguide to make connection

```
SC  0    3    .088  0.01394   .180      .088   -0.01394    .180
```

Rest of file is identical to COFFEE.NEC

```
SP  0    3      .088   -.01394      .300      .088   0.01394      .300
SC  0    3      .088   0.01394      .277      .088   -.01394      .277
SC  0    3      .088   0.01394      .254      .088   -.01394      .254
SC  0    3      .088   0.01394      .231      .088   -.01394      .231
SC  0    3      .088   0.01394      .208      .088   -.01394      .208
SC  0    3      .088   0.01394      .185      .088   -.01394      .185
SC  0    3      .088   0.01394      .162      .088   -.01394      .162
SC  0    3      .088   0.01394      .139      .088   -.01394      .139
SC  0    3      .088   0.01394      .116      .088   -.01394      .116
SC  0    3      .088   0.01394      .093      .088   -.01394      .093
SC  0    3      .088   0.01394      .069      .088   -.01394      .069
SC  0    3      .088   0.01394      .046      .088   -.01394      .046
SC  0    3      .088   0.01394      .023      .088   -.01394      .023
SC  0    3      .088   0.01394      .000      .088   -.01394      .000
SC  0    3      .060   0.00950      .000      .060   -.00950      .000
SC  0    3      .030   0.00475      .000      .030   -.00475      .000
GM  0    4      .000   0.0          18.0
GM  0    0      .000   0.0           9.0
SP  0    0      .015   0.015        .000      90.0    .00000   .0009701
GX  0  110
GW  1    4    .0622  .0622       .100     .0198     .0198      .100
.002
GM  0    0       0.0       0.0     45.0 0 0 -0.300
GE
FR  0    1    0    0     1296.0
EX  0    1    1    0        1.0         0
LD  5    0    0    0   3.72E+07
PT -1
RP  0   37    3 1500       0.0       0.0       10.0      45.0
EN
```

12.6.4 NEC2 example — W2IMU dual-mode feed

Peter, PA3AEF, also provided one other feed model: a W2IMU dual-mode feedhorn. This feed, sketched in Figure 6.5-4, has a flared section connecting an section of circular waveguide, like the coffee-can feed, to a larger diameter output section. Once again, the model starts with a strip down one edge, then duplicates the strip into a circular structure. The surface patches for the larger diameter section must be larger than patches for the smaller diameter if the same number of duplicates is to form the circle. Then the flared section must connect between the smaller patches and the larger ones, connecting to the corners of both. Since the flare is in three-dimensional space, a bit of trigonometry is in order. As we saw in Chapter 6.5, the calculated radiation patterns quickly show whether a set of dimensions will provide good dual-mode operation.

This model is excited by a dipole inside the circular waveguide rather than a monopole transition. There is no transmission line connecting to the dipole – we can leave out details like that in a computer model and mgically drive the dipole in the center directly.

Like the coffee-can feed, the W2IMU dual-mode feed starts with a strip of surface patches Down one edge

```
CM Dish-Feed W2IMU.NEC (short version)
CM Dipole feed for test
CM Horn is 610mm long, narrow end 168mm diameter
CM 30 degree increase up to 302mm diameter
CM Cylinder 302mm diameter, 305mm long
CM by PA3AEF
CE *********************************************
```

Start of large aperture 151 mm radius

```
SP  0    3   0.15100  -0.02392     0.610   0.15100   0.02392     0.610
SC  0    3   0.15100   0.02392     0.585   0.15100  -0.02392     0.585
SC  0    3   0.15100   0.02392     0.550   0.15100  -0.02392     0.550
SC  0    3   0.15100   0.02392     0.515   0.15100  -0.02392     0.515
SC  0    3   0.15100   0.02392     0.480   0.15100  -0.02392     0.480
SC  0    3   0.15100   0.02392     0.445   0.15100  -0.02392     0.445
SC  0    3   0.15100   0.02392     0.410   0.15100  -0.02392     0.410
SC  0    3   0.15100   0.02392     0.375   0.15100  -0.02392     0.375
SC  0    3   0.15100   0.02392     0.340   0.15100  -0.02392     0.340
SC  0    3   0.15100   0.02392     0.305   0.15100  -0.02392     0.305
```

Diameter decreases at 30 degree angle — both X and Y dimensions change for flare

```
SC  0    3   0.13643   0.02161     0.280   0.13643  -0.02161     0.280
SC  0    3   0.12187   0.01930     0.255   0.12187  -0.01930     0.255
SC  0    3   0.10439   0.01653     0.225   0.10439  -0.01653     0.225
```

To 84 mm diameter waveguide section — rest of file is like coffee-can feed

```
SC  0    3   0.08396   0.01324     0.190   0.08396  -0.01324     0.190
SC  0    3   0.08396   0.01324     0.160   0.08396  -0.01324     0.160
SC  0    3   0.08396   0.01324     0.130   0.08396  -0.01324     0.130
SC  0    3   0.08396   0.01324     0.100   0.08396  -0.01324     0.100
SC  0    3   0.08396   0.01324     0.066   0.08396  -0.01324     0.066
SC  0    3   0.08396   0.01324     0.044   0.08396  -0.01324     0.044
SC  0    3   0.08396   0.01324     0.022   0.08396  -0.01324     0.022
```

```
SC  0    3   0.08396   0.01324     0.000   0.08396  -0.01324     0.000
SC  0    3   0.06550   0.01037     0.000   0.06550  -0.01037     0.000
SC  0    3   0.03000   0.00475     0.000   0.03000  -0.00475     0.000
GM  0    4      0.00      0.00      18.0
GM  0    0      0.00      0.00       9.0
SP  0    0   0.01500   0.01500     0.000   90.0000   0.00000 .00071273
GX  0  110
GM  0    0      0.00      0.00     -45.0
```

Except for dipole feed in E-plane after rotations complete

```
GW  1   15    0  0.047     0.040    0    -0.047      0.040    0.002
GM  0    0    0    0        0        0        0       -0.610
GE
FR  0    1    0    0    1296.0
```

EX excitation at segment 8 of 15 of tag #1, the dipole

```
EX  0    1    8    0        1.0         0
LD  5    0    0    0  3.72E+07
PT -1
RP  0   19    3 1500        0.0        0.0        10.0       45.0
EN
```

12.7 Antenna impedance

One of the quantities reported by **NEC2** is the input impedance of the antenna. The impedance can be more sensitive to changes in dimensions than the radiation pattern, particularly the exciting monopole or dipole. I don't worry too much about the impedance as long as it is somewhat reasonable, say a VSWR under 10. It is possible to find an excitation point that won't couple energy into the antenna – this usually results in an input impedance that is almost completely reactive, so it is worth glancing at the impedance but not obsessing about it.

12.8 Summary

The ability to calculate antenna patterns with both amplitude and phase allows us to more accurately estimate performance of various parabolic dish feeds. It also provides the ability to calculate phase centers of the feeds and to see the effects of axial displacement errors. Graphical presentation then enables us to visualize this data and use it to optimize the performance of our dishes.

The ease and flexibility of computer modeling allows comparison and optimization of many antennas and feeds. We should put this capability to use to enhance our creativity, not stifle it, and develop new and better antennas. Please take advantage of the internet to share new models – we can all learn from them. And since this is an *online* book, updates can readily added.

As we concluded previously, optimum dish performance is realized by matching the feed to the ***f*/D** of the dish and aligning the phase center of the feed at the focus of the parabola. Computer analysis is useful in both choosing the best feed and calculating its phase center.

12.9 Acknowledgements

The **NEC** models for many of the feedhorns were derived from models for the VE4MA and W2IMU feeds provided to me by Peter Beyer, PA3AEF. The **NEC2** program was compiled to run under **linux** on an Alpha™ PC by Matt Reilly, KB1VC.

12.10 References

1. G.J. Burke & A.J. Poggio, *Numerical Electromagnetic Code (NEC) — Method of Moments*, Lawrence Livermore Laboratory, 1981.

2. The *unofficial* Numerical Electromagnetic Code (NEC) Archives, by WB6TPU, http://www.qsl.net/wb6tpu/swindex.html

3. L. Diaz & T. Milligan, *Antenna Engineering Using Physical Optics*, Artech, 1996.

4. W.V.T. Rusch, "A Comparison of Geometrical and Integral Fields From High Frequency Reflectors," *Proceedings of the IEEE*, November 1974, pp. 1603-1604.

5. **MATLAB** is a product of The Mathworks, Inc., http://www.mathworks.com

6. The *unofficial* Numerical Electromagnetic Code (NEC) Archives, by WB6TPU, http://www.qsl.net/wb6tpu/swindex.html