4-27 Array Geometry and Excitation Definitions

The material in Chapter 4 can be used in a computer program to both define element positions and their excitations to be used in computational electromagnetic tools. Program XADEF can be used to define the positions and orientation of array elements. XADEF applies various amplitude distributions to lower sidelobes. Various commands allow element edits, element removal, and lists of positions, orientations, and excitations. Input files can be prepared to run XADEF and serve as a record of the array design.

Rectangular Arrays

```
C:\Arrays>xadef
Enter input 0 keyboard, 1 file 0
Enter File Name sq225t.arr
Enter units: 1 in., 2 ft, 3 cm, 4 m 3
New File? y
Enter Frequency (GHz) 1
xadef: ??

Directive Array File Definition Directives

 1. (??,(pr)). . . . List Directives
 2. (ad,el,#). . . . Add, Elements,<Number>
 3. (ad,re,#,(ar)) . Add, Rectangular Array
 4. (ad,he,#,(ar)) . Add, Hexangular Array
 5. (ad,ci,#,(ar)) . Add, Circular Array
 6. (ad,ep,#,(ar)) . Add, Elliptical Array
 7. (ad,ar,#,(ar)) . Add, Arc Array
 8. (ad,ca,#,(ar)) . Add, Cap Array
 9. (ad,co,#,(ar)) . Add, Cone Array
10. (ad,tr,#,(ar)) . Add, Triangular Spaced Array
11. (ad,ri,#,(ar)) . Add, Ring filled Array
12. (ad,sp,#,(ar)) . Add, Spherical Array
12. (ad,ic,#,(ar)) . Add, Icosahedron Spherical Array
13. (li,fi,(pr)) . . List File
14. (li,<NUMBER>). . List Element <NUMBER>
15. (ed,lo,#). . . . Edit Location <NUMBER>
16. (ed,am,#). . . . Edit Amplitude <NUMBER>
17. (ed,am,gr,#,#) . Edit Amplitude Group,<#>,to<#>
18. (ed,ph,#). . . . Edit Phase <NUMBER>
19. (ed,eu,#). . . . Edit Euler Angles <NUMBER>
20. (ch,fr). . . . . Change Frequency
21. (ex) . . . . . . Exit xadef

xadef: ad,re,225
Enter Ampl (dB), Phase 0,0
Enter Number of Elements along X-axis 15
Enter initial spacings in X,Y axes cm  15,15
Array along X-axis
Enter Axis Spacing: 1 Uniform, 2 Bratkovic  3 Geometric, 4 Uneven Taylor 1
Enter 1) Uniform Amplitude Distribution
      2) Area Sampling of Taylor Distribution
      3) Point Sampling of Taylor Distribution
      4) Zero Sampled Taylor Distribution
      5) Point Sampled Bayliss Distribution
      6) Zero Sampled Bayliss Distribution
      7) Chebyshev array 4_
```

```
Enter Sidelobe Level (dB) 30
Enter # of First Unchanged Zero of Uniform Distr. 6
Enter Quadratic Phase Factor, S 0
Array along Y-axis
Enter Axis Spacing: 1 Uniform, 2 Bratkovic  3 Geometric, 4 Uneven Taylor 1
Enter 1) Uniform Amplitude Distribution
      2) Area Sampling of Taylor Distribution
      3) Point Sampling of Taylor Distribution
      4) Zero Sampled Taylor Distribution
      5) Point Sampled Bayliss Distribution
      6) Zero Sampled Bayliss Distribution
      7) Chebyshev array 4
Enter Sidelobe Level (dB) 30
Enter # of First Unchanged Zero of Uniform Distr. 6
Enter Quadratic Phase Factor, S 0
move: ??
MOVE Commands:
 1. (tw) . . . . . . Twist Array
 2. (ta) . . . . . . Sample Circular Taylor Distr.
 3. (li) . . . . . . Sample Linear Distribution
 4. (sp) . . . . . . Space Feed Distribution
 5. (ti) . . . . . . Tilt Elements about plane Z axis
 6. (ro,pl). . . . . Rotate Elements in Plane
 7. (ro,ax). . . . . Rotate Array about Axis
 8. (tr,pl). . . . . Translate elements in plane
 9. (tr,ge). . . . . Translate elements in general
10. (pr,co). . . . . Project Array on to Cone
11. (pr,he). . . . . Project on to Cylindrical Helix
12. (pr,ep). . . . . Project on to Elliptical Helix
13. (ci) . . . . . . Confine array to circle
14. (re) . . . . . . Confine array to rectangle
15. (co,sp). . . . . Confine using spline of Y
16. (po) . . . . . . Confine array in polygon
17. (no) . . . . . . Normalize Array to power = 1
18. (mo) . . . . . . Apply spiral mode
19. (qa) . . . . . . Quantize amplitudes
20. (ex) . . . . . . Exit and convert array
move: no
move: ex
Enter Final New Z axis Rotation of Antennas (array) 0
xadef: _
```

Program XADEF generates array definition file of 225 element rectangular array. After preliminaries, the program prompts with its name. By typing '??', the program returns a list of text commands delimitated by comma or space. For example, "array rectangular 225" or equivalently "ar,re,225" for a 225 element square array (15 on each axis). The elements are uniformly spaced and excited with a zero sampled (Section 4-12) 30 dB, n-bar = 6 distribution along both *x*- and *y*-axes.

After the array is defined in the *XY*-plane, the "move:" allows various rotations, translations, projects, etc. on the array. It lists its possible operations when "??" is entered. The "no" command normalizes the array feeding coefficients to an input power of one.

All commands for XADEF can be listed in a text file and used to run the program and document the array definition. The array can be pictured in perspective as squares for elements with a short normal along *z*-axis.

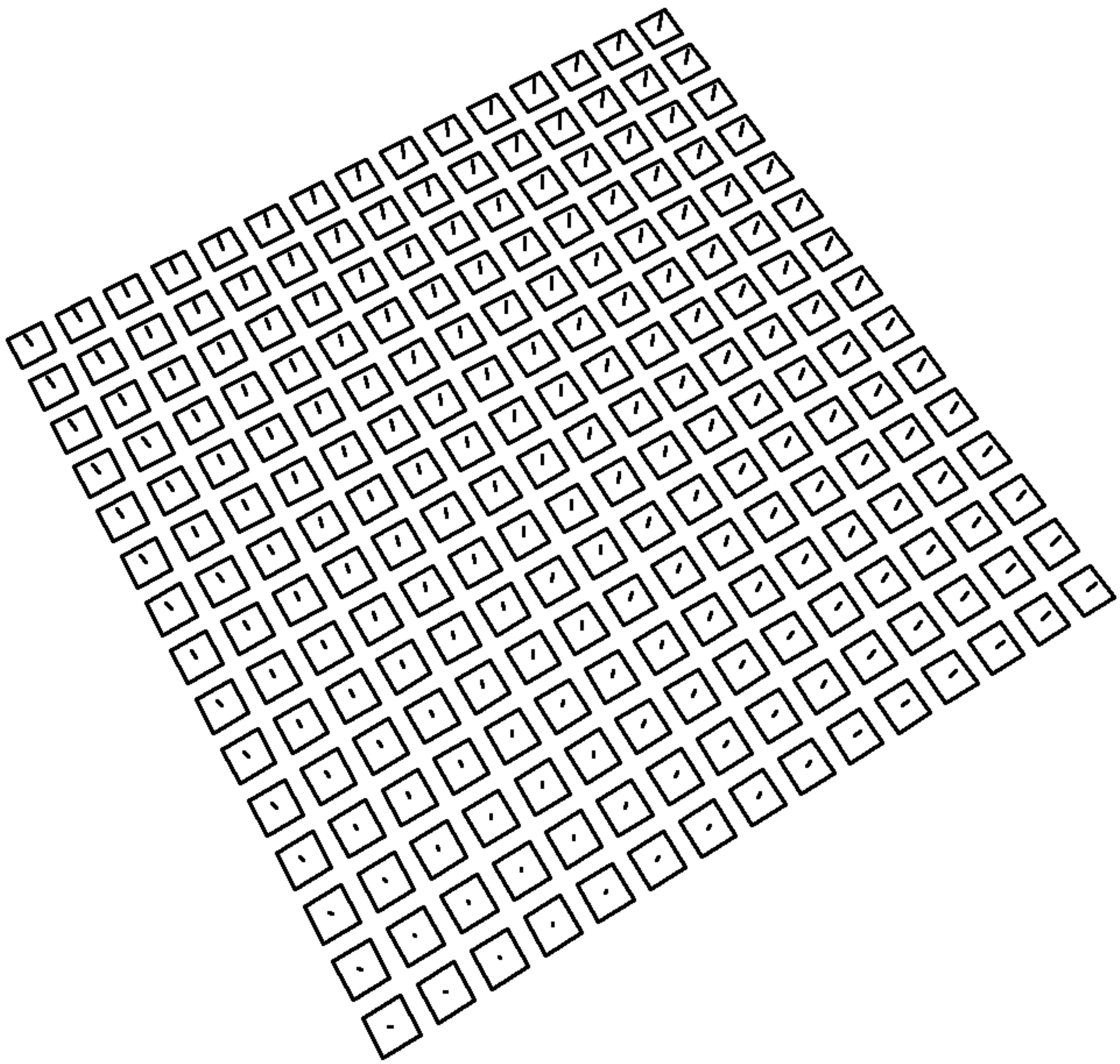

Figure 1 Square 225 element array with 15 cm spacing in the *XY*-plane

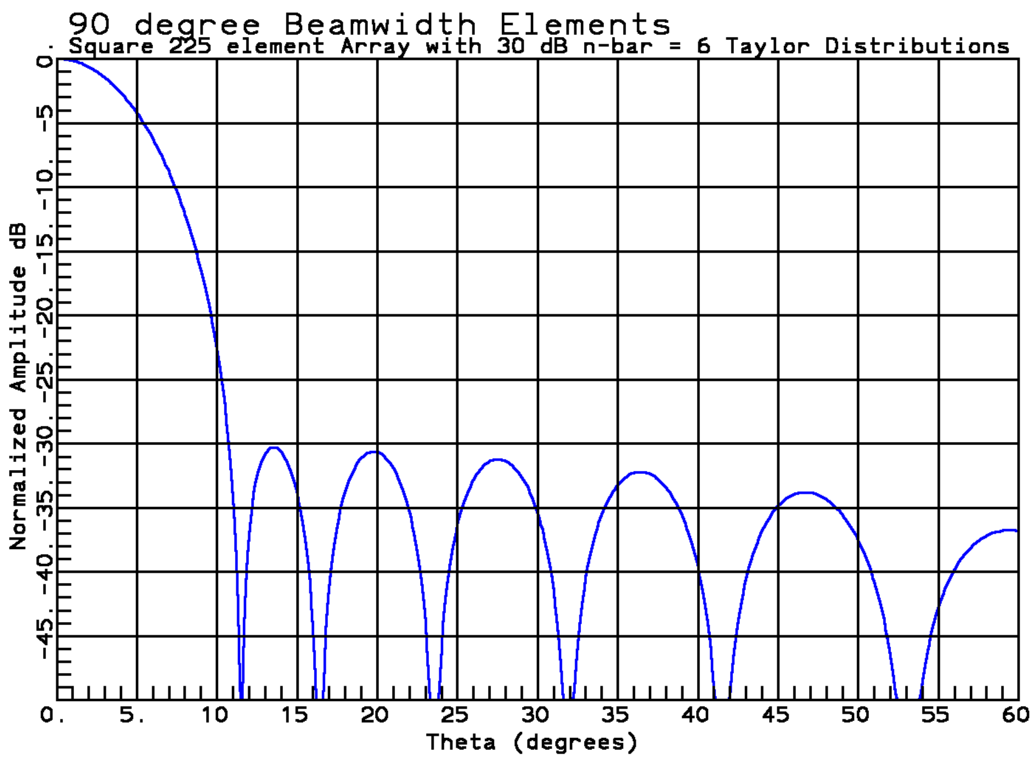

Figure 2 Pattern of 225 elements Square Array with 30 dB Taylor Distribution

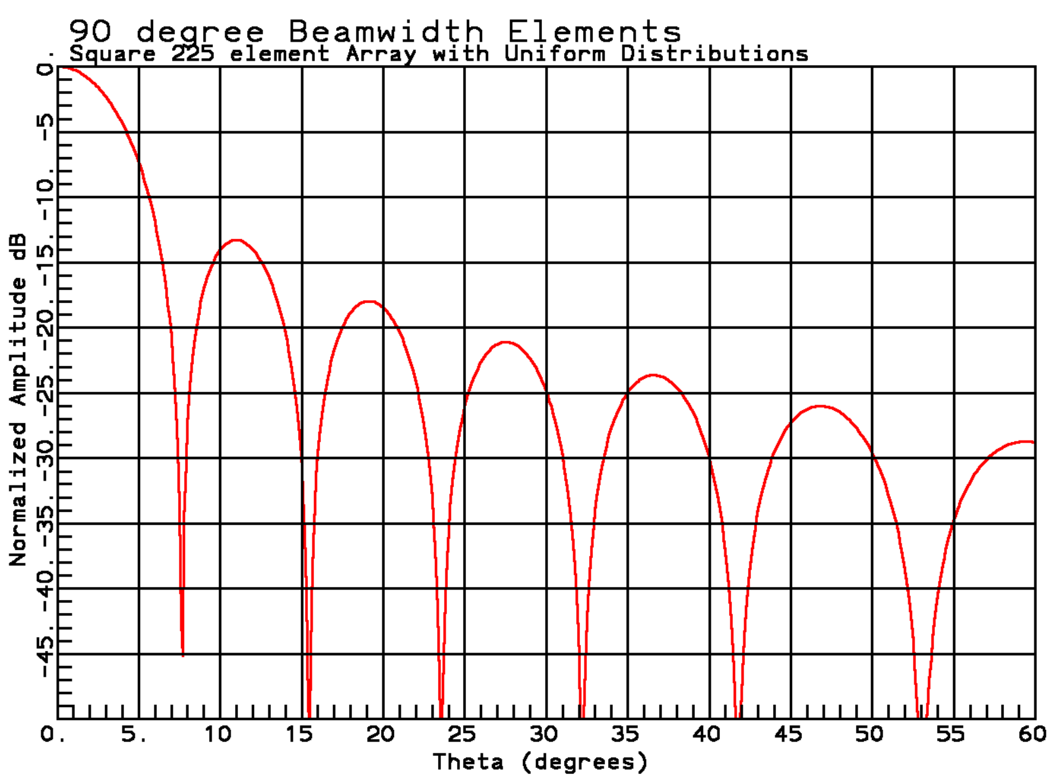

Figure 3 Pattern of 225 elements Square Array with Uniform Distribution

Bratkovic Array

Bratkovic spacing is used with arrays of frequency independent antennas to minimize the maximum sidelobe of patterns over a large frequency range. The element spacing exceeds a wavelength as frequency increases and the total power radiated in grating lobes is spread over the sidelobes. The second element spacing option in XADEF for a linear array is the Bratkovic array.

Franc Bratkovic, 'Synthesis of Broad-band Arrays with Arbitrary Frequency-Independent Elements', IEEE Trans. on Antennas Propagat., March 1973, pp. 211-213.

```
C:\Arrays>xadef
Enter input 0 keyboard, 1 file 0
Enter File Name bratk25.arr
Enter units: 1 in., 2 ft, 3 cm, 4 m 3
New File? y
Enter Frequency (GHz) 1
xadef: ad,re,25
Enter Ampl (dB), Phase 0,0
Enter Number of Elements along X-axis 25
Enter initial spacings in X,Y axes cm  15,15
Array along X-axis
Enter Axis Spacing: 1 Uniform, 2 Bratkovic  3 Geometric, 4 Uneven Taylor 2
Enter Frequency Ratio 5
Recommended Spacing Ratio for Frequency Ratio, Rf
1<Rf<6: 2, 6<Rf<10: 1.5, 10<Rf<27: 1.33, Rf>27: 1.28
Rf*Rs>4 Enter Spacing Ratio 2
Enter Constant K of Cos(Theta)**K/2 Element Pattern (0 - 2) 0
Enter 1) Uniform Amplitude Distribution
      2) Area Sampling of Taylor Distribution
      3) Point Sampling of Taylor Distribution
      4) Zero Sampled Taylor Distribution
      5) Point Sampled Bayliss Distribution
      6) Zero Sampled Bayliss Distribution
      7) Chebyshev array 1
Enter Quadratic Phase Factor, S 0
Array along Y-axis
move: ex
Enter Final New Z axis Rotation of Antennas (array) 0
xadef: li,fi
```

The spacing distribution asks for the frequency ratio (5 in the case above) over which the array will operate. The program suggests a spacing ratio given frequency ratio and the case above uses isotropic elements (k=0) in a dynamic programming to compute the unequally spaced array. We can apply an amplitude distribution across the linear array, but the example above uses uniform. Instead of an array width of 360 cm (15x24) the elements are spread to a total width ~520.3 cm.

xadef: li,fi

File:bratk25.arr

No	Location X	Y	Z	Element Ampl(dB)	Phase	Euler Angles		
1	-260.134	0.000	0.000	0.00	0.00	0.00	0.00	0.00
2	-245.134	0.000	0.000	0.00	0.00	0.00	0.00	0.00
3	-229.675	0.000	0.000	0.00	0.00	0.00	0.00	0.00
4	-213.743	0.000	0.000	0.00	0.00	0.00	0.00	0.00
5	-197.324	0.000	0.000	0.00	0.00	0.00	0.00	0.00
6	-180.402	0.000	0.000	0.00	0.00	0.00	0.00	0.00
7	-162.962	0.000	0.000	0.00	0.00	0.00	0.00	0.00
8	-144.989	0.000	0.000	0.00	0.00	0.00	0.00	0.00
9	-126.466	0.000	0.000	0.00	0.00	0.00	0.00	0.00
10	-107.377	0.000	0.000	0.00	0.00	0.00	0.00	0.00
11	-87.703	0.000	0.000	0.00	0.00	0.00	0.00	0.00
12	-67.428	0.000	0.000	0.00	0.00	0.00	0.00	0.00
13	-46.532	0.000	0.000	0.00	0.00	0.00	0.00	0.00
14	-24.996	0.000	0.000	0.00	0.00	0.00	0.00	0.00
15	-2.802	0.000	0.000	0.00	0.00	0.00	0.00	0.00
16	20.071	0.000	0.000	0.00	0.00	0.00	0.00	0.00
17	43.644	0.000	0.000	0.00	0.00	0.00	0.00	0.00
18	67.938	0.000	0.000	0.00	0.00	0.00	0.00	0.00
19	92.976	0.000	0.000	0.00	0.00	0.00	0.00	0.00
20	118.779	0.000	0.000	0.00	0.00	0.00	0.00	0.00
21	145.372	0.000	0.000	0.00	0.00	0.00	0.00	0.00
22	172.779	0.000	0.000	0.00	0.00	0.00	0.00	0.00
23	201.024	0.000	0.000	0.00	0.00	0.00	0.00	0.00
24	230.134	0.000	0.000	0.00	0.00	0.00	0.00	0.00
25	260.134	0.000	0.000	0.00	0.00	0.00	0.00	0.00

xadef:

Figure 4 25 element Bratkovic spaced linear array for 5:1 Frequency Range

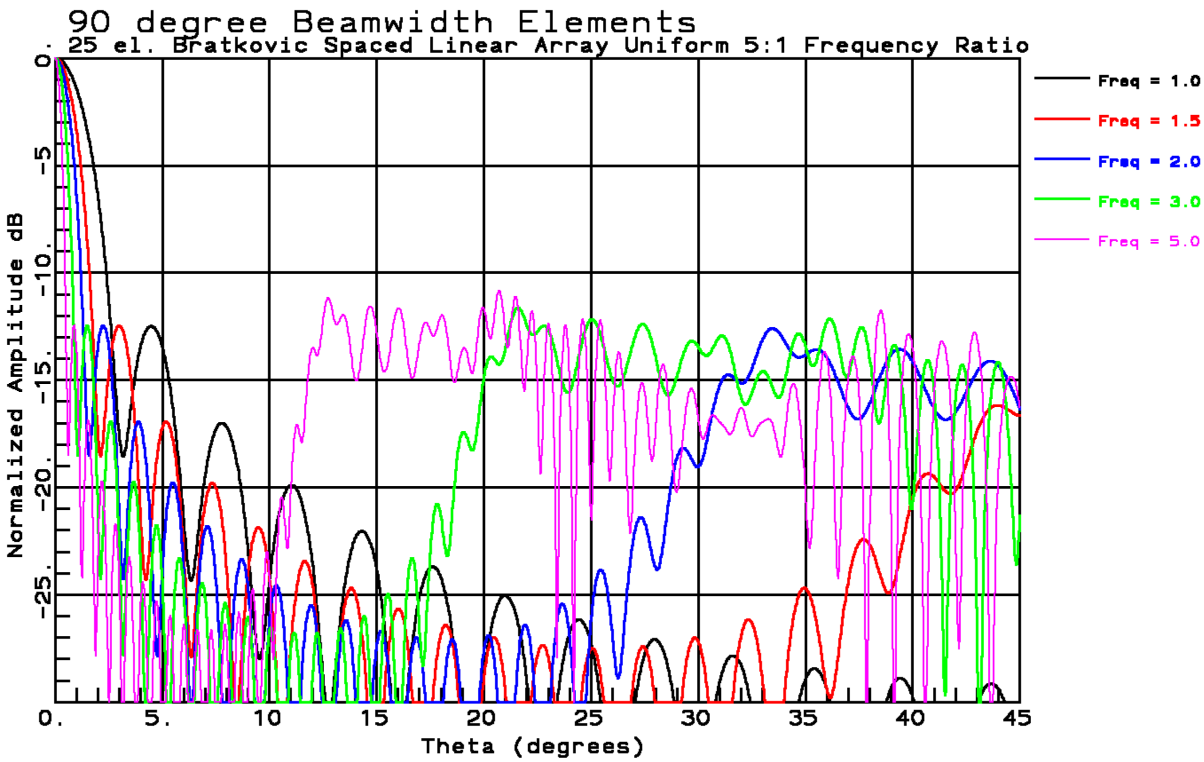

Figure 5 25 element Bratkovic spaced linear array for 5:1 Frequency Range Uniform Distribution

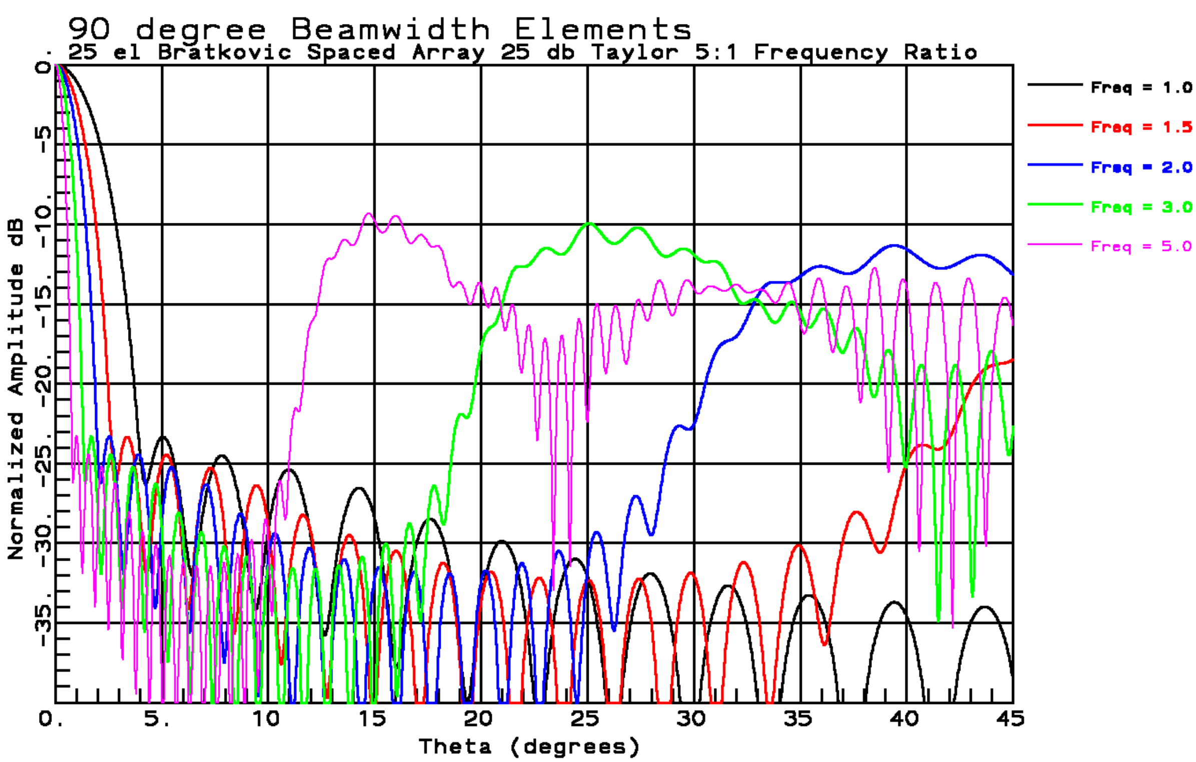

Figure 6 25 element Bratkovic spaced linear array for 5:1 Frequency Ratio 25 dB Taylor Distr.

The 25 dB n-bar = 5 pointed sampled Taylor amplitude distribution (Section 4-4) reduces the inner sidelobes in figure 6, but fails to reduce the spread out grating lobes due to the Bratkovic spacing.

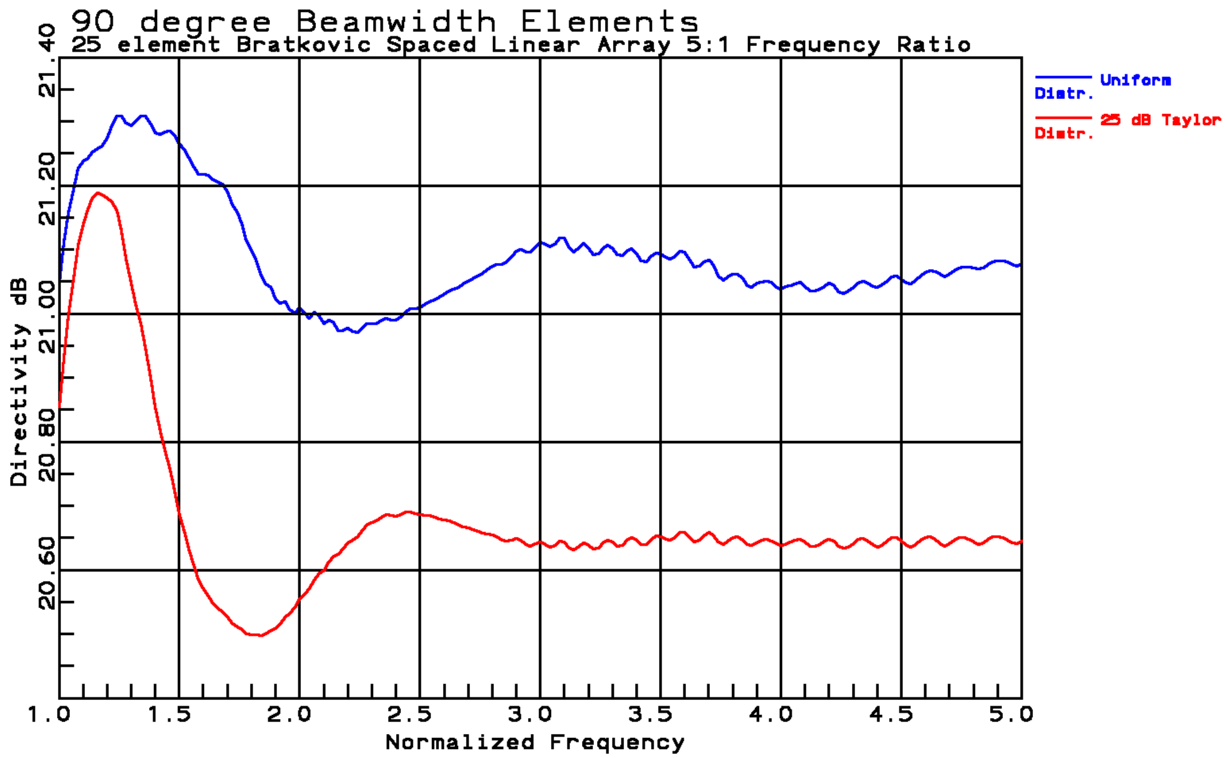

Figure 7 25 element Bratkovic spaced linear array for 5:1 Frequency Ratio Directivity

Figure 7 illustrates that directivity of the Bratkovic linear array is nearly constant over the 5:1 frequency range. Figure 7 computations use the normalized mutual resistance method of section 3-12. The average directivity difference between the two curves of Figure 7 matches the value given in Table 4-5.

Geometric Spaced Array

The third spacing choice for a linear array is geometric. The program spaces elements log-periodically along the axis to fill a longer length than equal spacing. It has the property of tapering the aperture distribution which can lower sidelobes for a symmetric arrangement.

Figure 8 Symmetric 25 element Linear Array Spaced **geometric** over 35 normalized spacing lengths (τ = 0.938 of log-periodic spacing)

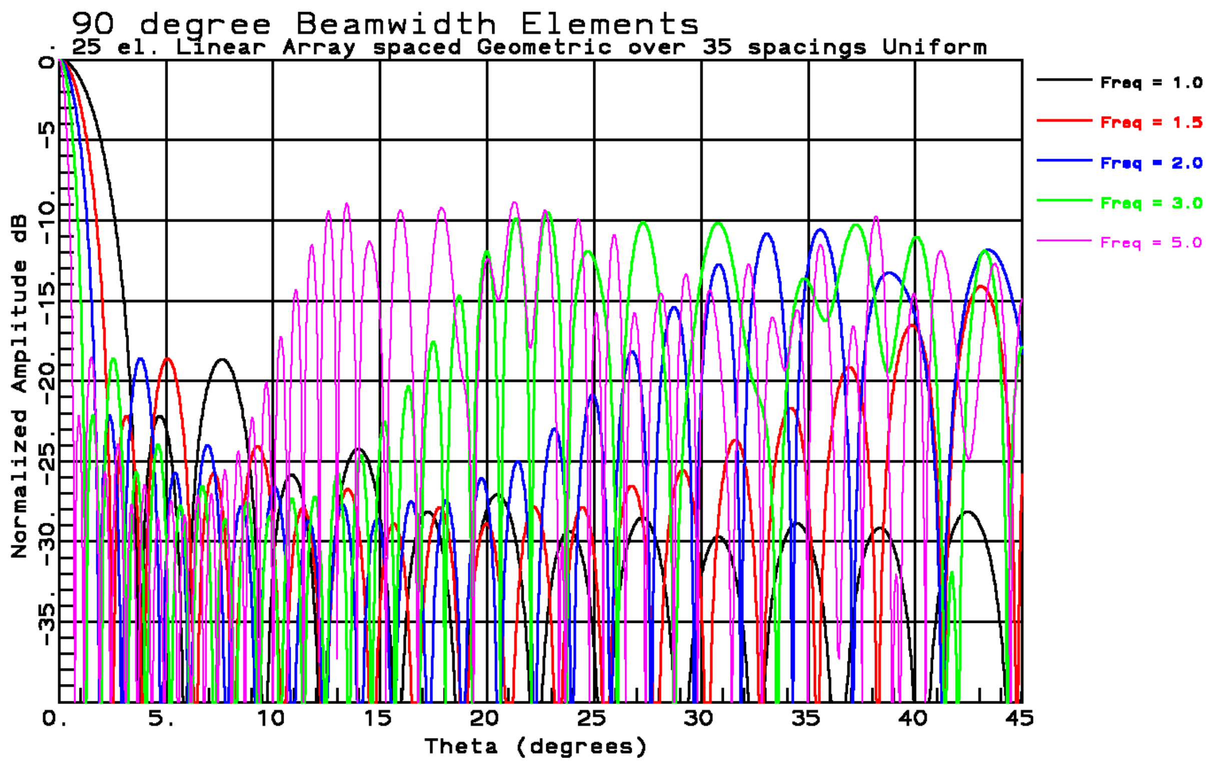

Figure 9 25 element Geometric spaced linear array over 35 element spacings Uniform Distribution

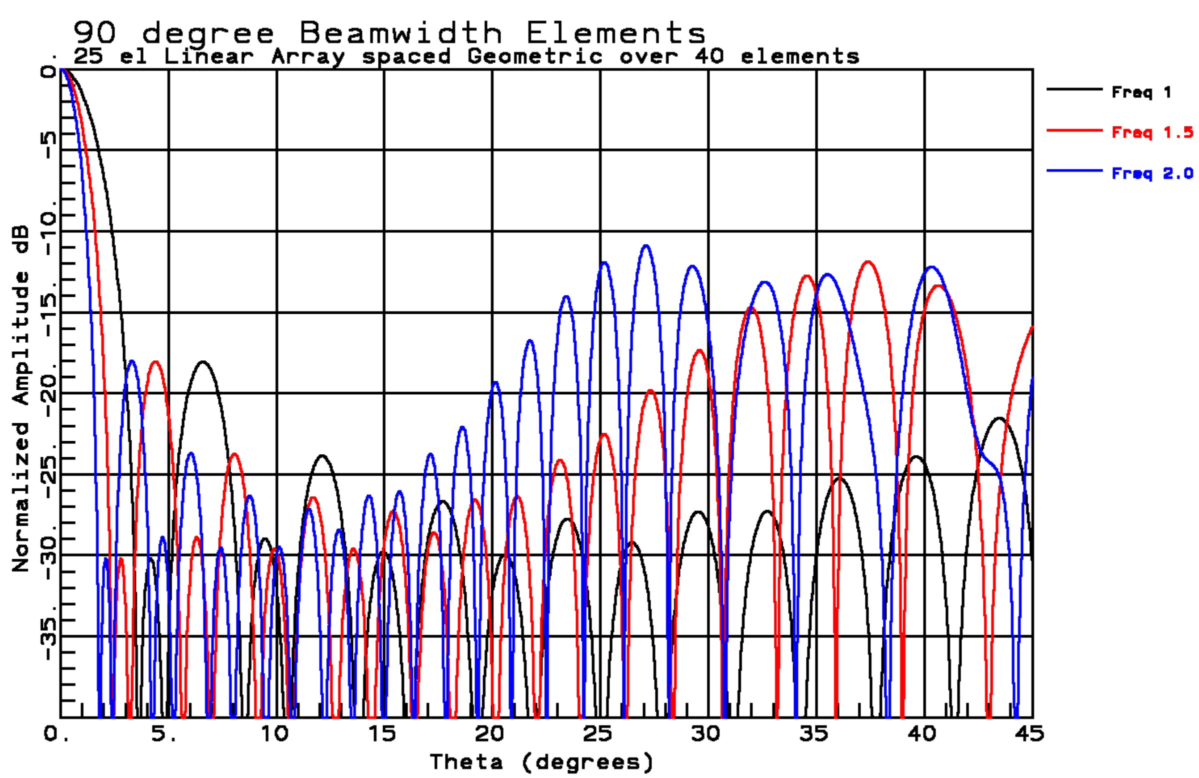

Figure 10 25 element Geometric spaced linear array over 40 element spacings Uniform Distribution

Increasing the length of the geometric spaced array helps the first sidelobe, but the second one is approximately the same in both arrangements.

Unevenly Spaced Linear Taylor Distribution

The position of elements can be located by area sampling a linear distribution (Section 12-4.1). The fourth option in spacing elements of a linear array implements this method.

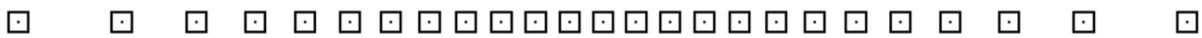

Figure 11 Symmetric 25 elements Linear Array Unevenly Spaced to Area Sample 30 dB Taylor Distribution

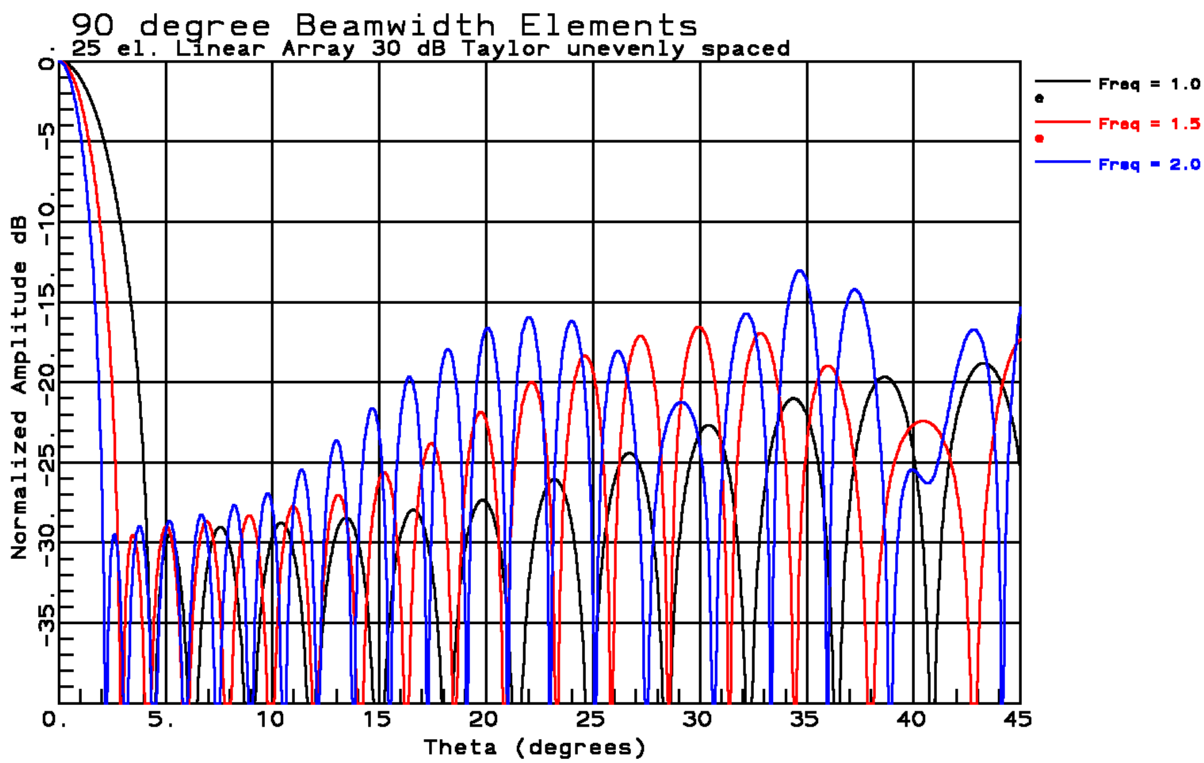

Figure 12 Symmetric 25 elements Linear Array Unevenly Spaced to Area Sample 30 dB Taylor Distribution

Area sampling a linear Taylor distribution for an unevenly spaced array does a better job of controlling the near-in sidelobes of the array. However, even with the inner most array spacing starting at λ/2, the array sidelobes start to rise after a few of the inner sidelobes. Increasing the number of elements in the array improves its ability to control more of the sidelobes (Figure 12-8).

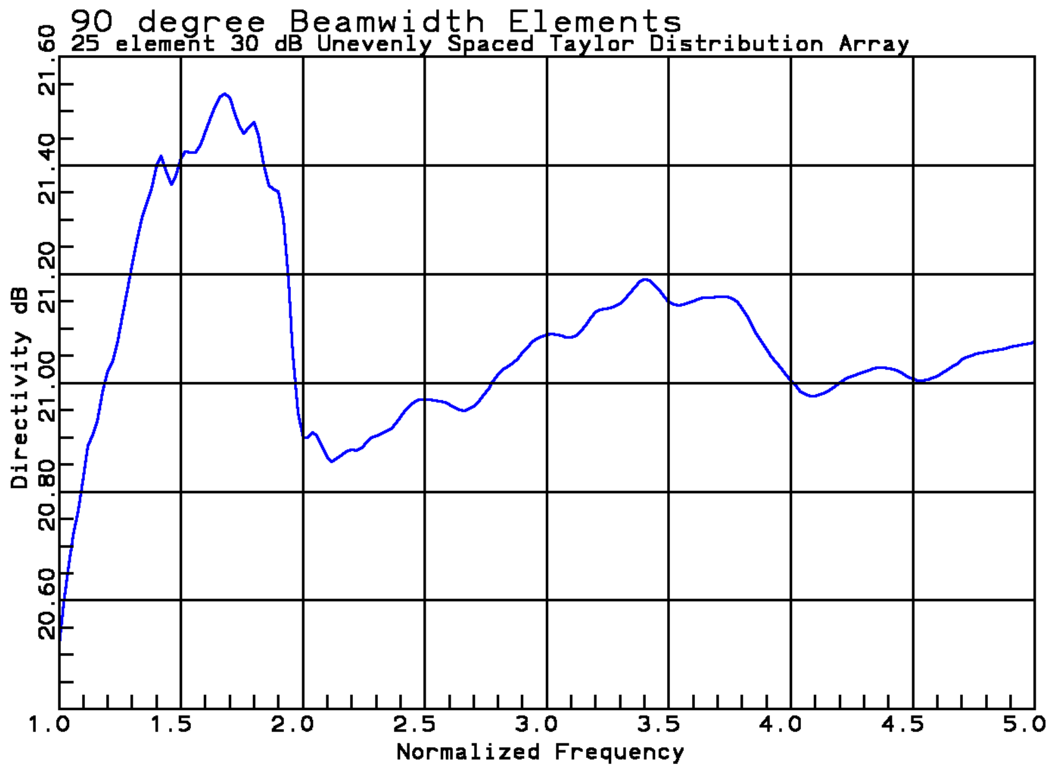

Figure 13 25 element 30 dB Taylor Distribution Unevenly Spaced Linear Array Directivity

Figure 13 computations use the normalized mutual resistance method of section 3-12 to find the directivity of unevenly spaced array with uniform amplitude distribution. Because more elements nearly equally spaced in Figure 11 than the Bratkovic array (Figure 4), the curve on Figure 13 has greater variation than the upper one of Figure 7. Less than a normalized frequency of 1.5, the efficiency degrades more quickly. Nevertheless, the curve in Figure 13 is fairly flat compared to evenly spaced array directivities.

Hexagonal Array

The equally spaced hexagonal array is arranged in rings. Given the number of rings, N_R, the total number of elements equals $3N_R(N_R+1)+1$. For example, a 9 ring array has 3(9)10+1 = 271 elements. XADEF has a hexagonal array input. The maximum element spacing is 0.577λ for a hexagonal array without grating lobes (Section 3-8).

```
C:\Arrays>xadef
Enter input 0 keyboard, 1 file 0
Enter File Name hex271.arr
Enter units: 1 in., 2 ft, 3 cm, 4 m 3
New File? y
Enter Frequency (GHz) 1
xadef: ad,he,271
Enter Ampl (dB), Phase 0,0
Enter Rotation of Hex 0
Enter Element Spacing cm  18
move: ex
Enter Final New Z axis Rotation of Antennas (array) 0
xadef: _
```

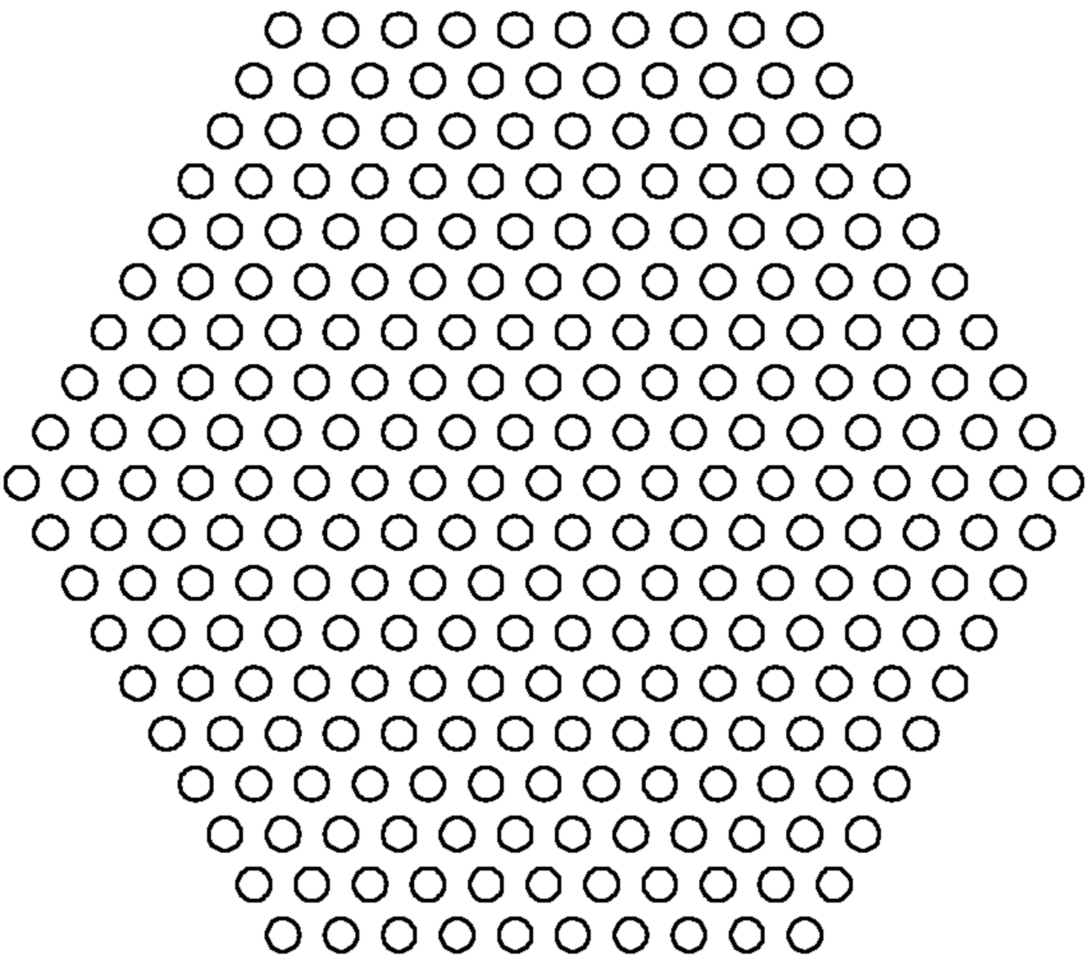

Figure 14 271 element Equally Spaced Hexagonal Array

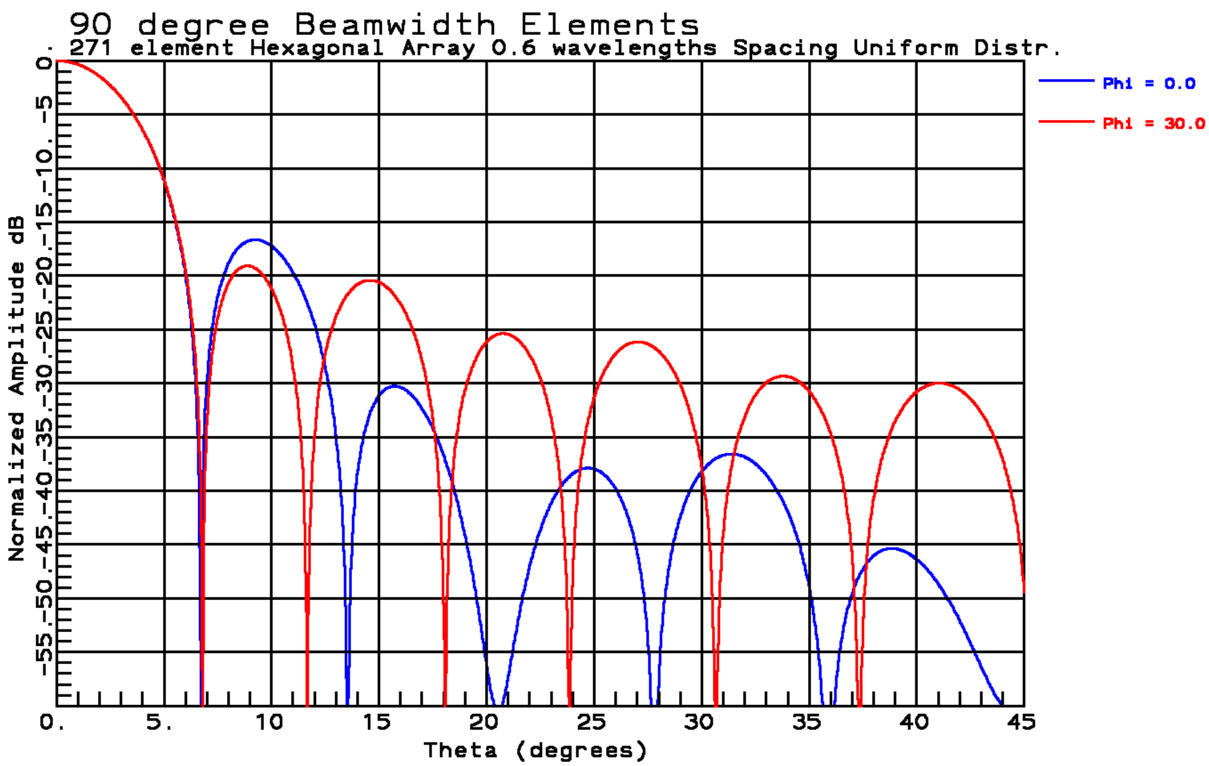

Figure 15 271 Element Hexagonal Array Uniform Amplitude Distribution

Ring Filled Array

The array definition command "ar,ri,*" generate a ring filled array. A command: ar,ri,1 is sufficient to produce an array which fills the circular ring with approximately equally spaced elements in a series of rings and the number of elements specified does not matter. See the listing of the XADEF program inputs below for a filled circle array starting at a radius = 0 and continuing to a radius = 162 cm. Since we want elements spaced ~15 cm, we specify 11 rings in the array. XADEF generates array geometry with no ϕ symmetry, 2 way, or 4 way. The program computes that 373 elements are required and lists the number of elements in each ring.

```
C:\Arrays>xadef
Enter input 0 keyboard, 1 file 0
Enter File Name fcir373.arr
Enter units: 1 in., 2 ft, 3 cm, 4 m 3
New File? y
Enter Frequency (GHz) 1
xadef: ad,ri,1
Enter Ampl (dB), Phase 0,0
Filled Ring Array in X-Y Plane
Enter Initial, Final Radius cm  0,162
Enter Start Rotation Angle 0
Enter number of rings 11
Enter spacing between elements in rings cm  15
Enter phi symmetry 1 none, 2 way, 4 way 1
Enter spacing: 1 uniform, 2 Uneven Taylor 1

Ring   Radius   NP  Spacing
  1    7.3636    3  15.0000
  2   22.0909    9  15.0000
  3   36.8182   15  15.0000
  4   51.5455   22  15.0000
  5   66.2727   28  15.0000
  6   81.0000   34  15.0000
  7   95.7273   40  15.0000
  8  110.4545   46  15.0000
  9  125.1818   52  15.0000
 10  139.9091   59  15.0000
 11  154.6364   65  15.0000
Number of elements =   373
move: ex
Enter Final New Z axis Rotation of Antennas (array) 0
xadef: ex
Another file? n
Stop - Program terminated.

C:\Arrays>_
```

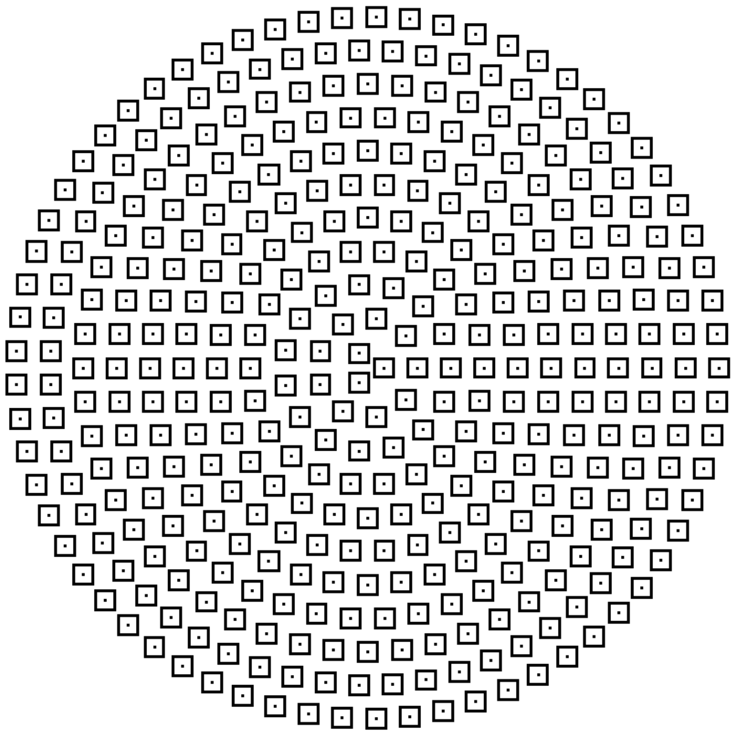

Figure 16 373 element Filled Circle Array

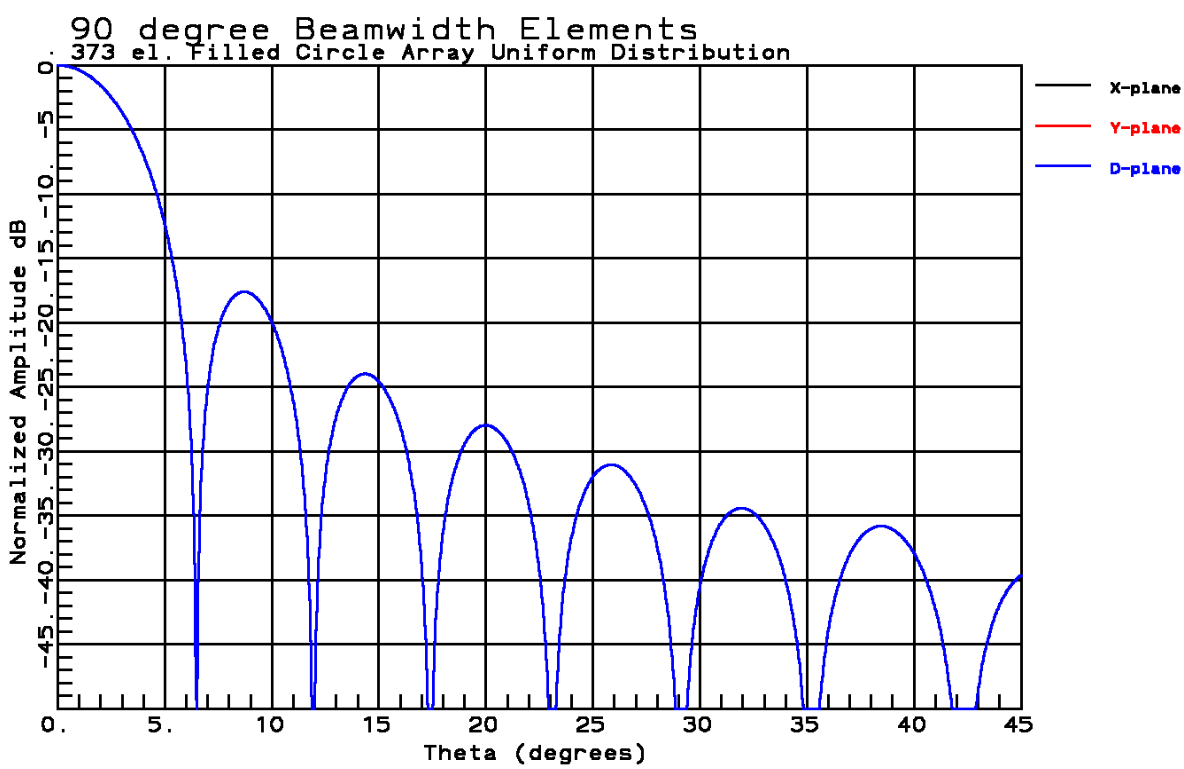

Figure 17 373 element Filled Circle Array Pattern

All three plots are identical and have a similar shape to the uniform circular aperture. Figure 17 plots the horizontal (*X*-plane), vertical (*Y*-plane), and diagonal (*D*-plane) of the filled array confined to a circle.

Ring Filled Array for Circular Taylor Distribution using Uneven Spacing

```
C:\Arrays>del fcirt30.arr

C:\Arrays>xadef
Enter input 0 keyboard, 1 file 0
Enter File Name fcirt30.arr
Enter units: 1 in., 2 ft, 3 cm, 4 m 3
New File? y
Enter Frequency (GHz) 1
xadef: ad,ri,1
Enter Ampl (dB), Phase 0,0
Filled Ring Array in X-Y Plane
Enter Initial, Final Radius cm  0,162
Enter Start Rotation Angle 0
Enter number of rings 11
Enter spacing between elements in rings cm  15
Enter phi symmetry 1 none, 2 way, 4 way 1
Enter spacing: 1 uniform, 2 Uneven Taylor 2
Circular Taylor Distribution Sampling
Enter Sidelobe Level (dB) 30
Enter Number of First Unchanged Zero of Uniform Distribution 6
Adjust for non-uniform spacing in rings? n
Generate plot of method? n

Ring   Radius   NP  Spacing
  1     4.9577    2  15.0000
  2    14.8831    6  15.0000
  3    24.8806   10  15.0000
  4    35.1081   15  15.0000
  5    45.8120   19  15.0000
  6    57.2688   24  15.0000
  7    69.7137   29  15.0000
  8    83.4524   35  15.0000
  9    99.5420   42  15.0000
 10   121.3913   51  15.0000
 11   149.1169   62  15.0000
Number of elements =   295
move: ex
Enter Final New Z axis Rotation of Antennas (array) 0
xadef: ex
Another file? n
Stop - Program terminated.

C:\Arrays>_
```

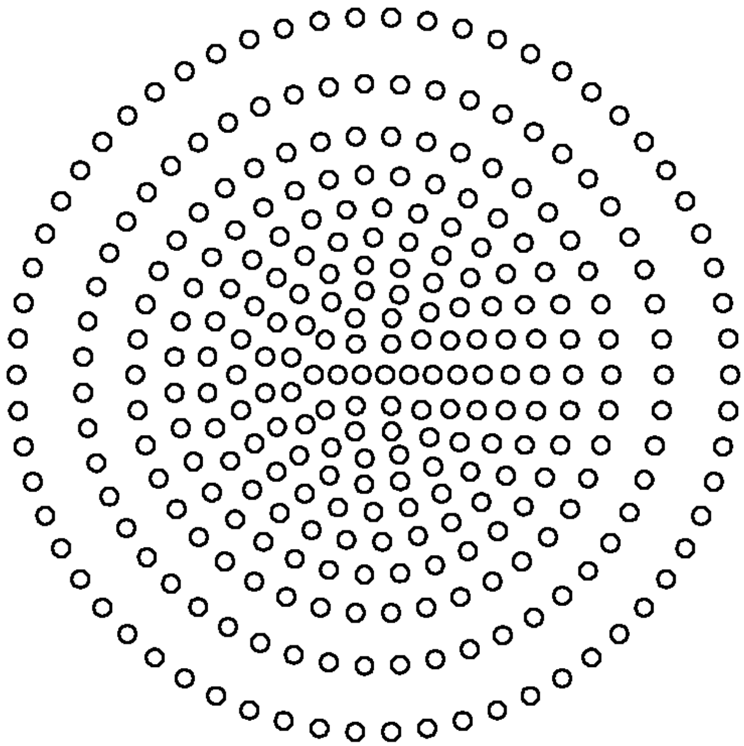

Figure 18 295 element Filled Circle Array Unevenly Spaced for 30 dB Taylor

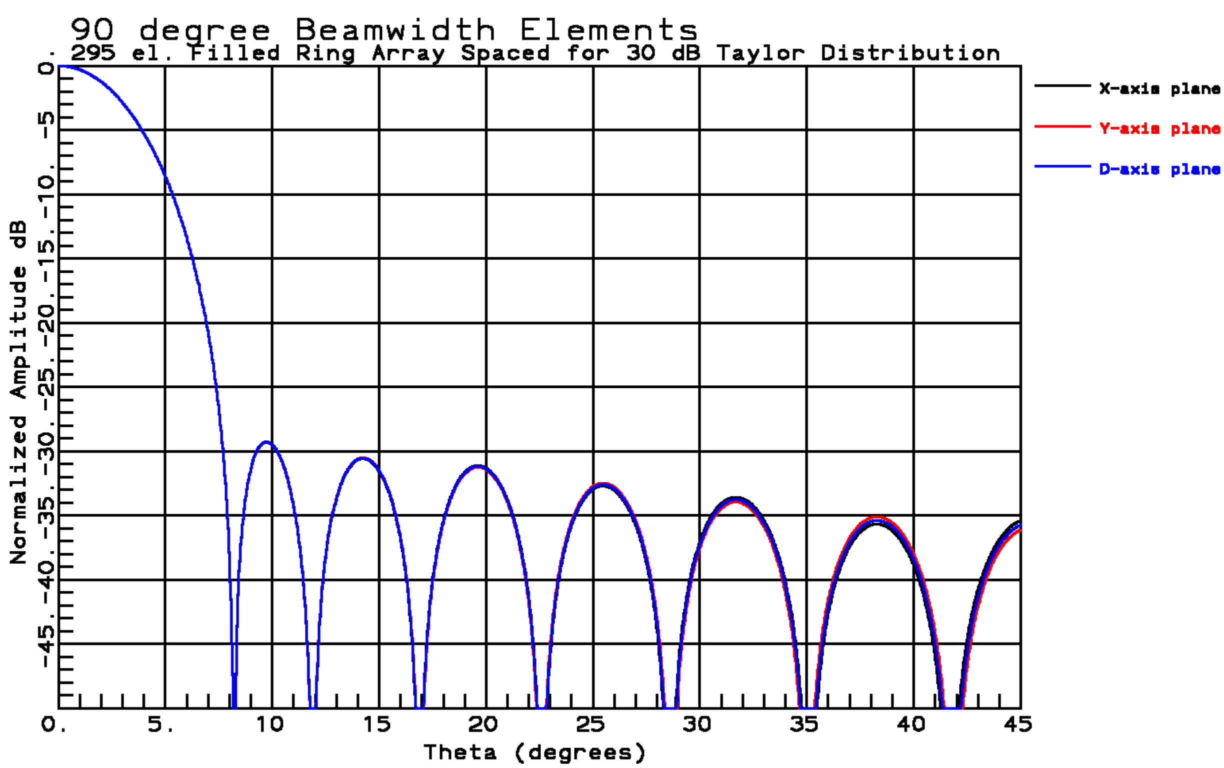

Figure 19 295 element Filled Circle Array Unevenly Spaced for 30 dB Taylor Uniform Feeding

Ring Filled Array with Central Hole

```
C:\Arrays>xadef
Enter input 0 keyboard, 1 file 0
Enter File Name frin300.arr
Enter units: 1 in., 2 ft, 3 cm, 4 m 3
New File? y
Enter Frequency (GHz) 1
xadef: ad,ri,1
Enter Ampl (dB), Phase 0,0
Filled Ring Array in X-Y Plane
Enter Initial, Final Radius cm  32,162
Enter Start Rotation Angle 0
Enter number of rings 9
Enter spacing between elements in rings cm  15
Enter phi symmetry 1 none, 2 way, 4 way 4
Enter spacing: 1 uniform, 2 Uneven Taylor 1

Ring   Radius   NP  Spacing
  1   39.2222   20  15.0000
  2   53.6667   24  15.0000
  3   68.1111   32  15.0000
  4   82.5556   36  15.0000
  5   97.0000   44  15.0000
  6  111.4444   48  15.0000
  7  125.8889   56  15.0000
  8  140.3333   60  15.0000
  9  154.7778   68  15.0000
Number of elements =   388
move: ex
Enter Final New Z axis Rotation of Antennas (array) 0
xadef: ex
Another file? n
Stop - Program terminated.
```

The inputs to XADEF generate an array with an inner hole of at least 64 cm diameter and 162 cm maximum diameter. This input requires 4 way symmetry which alters the input and output radiuses so that the number of elements in all rings is an integer multiplier of four.

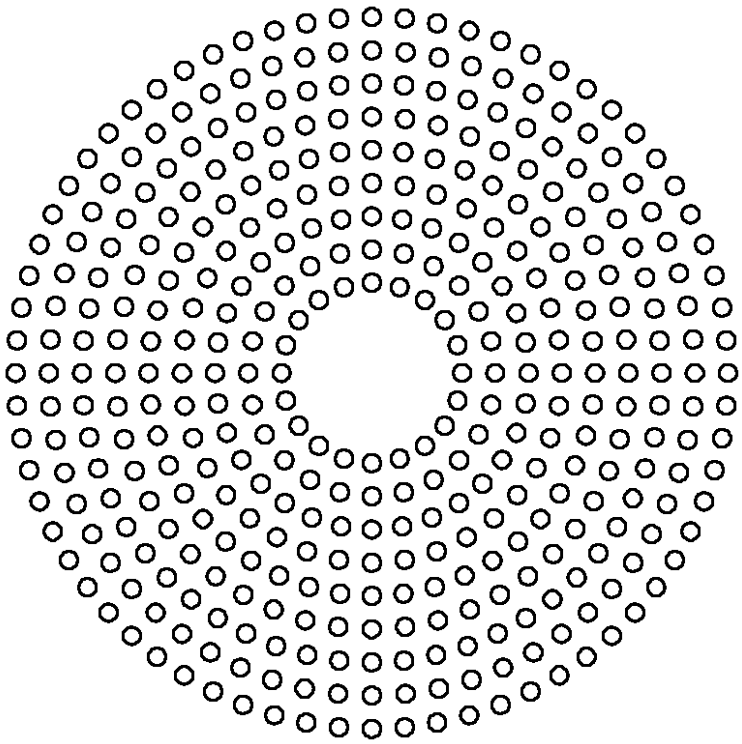

Figure 20 388 element Filled Ring Array with 4 way symmetry

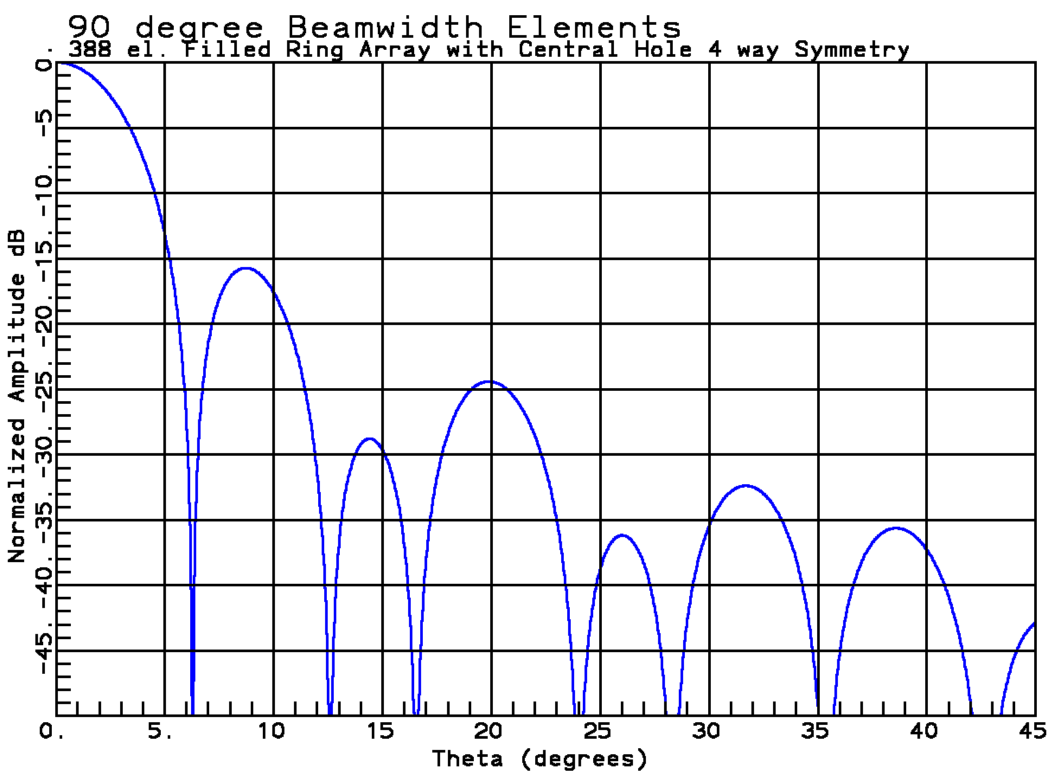

Figure 21 388 element Filled Ring Array with 4 way symmetry

We can add the pattern of the central hole to the pattern of the filled circular array to obtain Figure 21. The wide beamwidth of the hole radiation adds to the first sidelobe, subtracts from the second sidelobe, adds to the third sidelobe, subtracts from the fourth sidelobe because it is 180° out-of-phase with the filled aperture phase. The sidelobe phases of the filled array alternate phase 180°, 0, 180°, 0 relative to the central lobe.

Triangularly Spaced Array

The hexagonal array is a special case of the triangularly spaced array with the angle with respect to the *x*-axis of 60°. The hexagonal array generates a six-sided ring structure which can be modified in the "move" commands that confine it to various shapes: circle, rectangle, polynomial, etc. This command generates a triangular array confined in a trapezoid. Later we can alter the number of elements and exterior rim shape in the "move" commands.

We can specify any number of elements in the command, but it is ignored. The command generates linear arrays along the *x*-axis with alternating rows shifted by the triangle angle.

```
C:\Arrays>xadef
Enter input 0 keyboard, 1 file 0
Enter File Name tri240.arr
Enter units: 1 in., 2 ft, 3 cm, 4 m 3
New File? y
Enter Frequency (GHz) 1
xadef: ad,tr,1
Enter Ampl (dB), Phase 0,0
Triangularly Spaced Array within Trapezoid
Enter Spacings: X-axis, Y-axis cm  15,10.607
Enter Lower, Upper X-axis widths cm  217.5,217.5
Enter Y-axis Width cm  159.11
Enter initial position X,Y cm  0,0
240 Elements Added
move: ex
Enter Final New Z axis Rotation of Antennas (array) 0
xadef: _
```

XADEF centers the array in the *x*-*y* plane within the trapezoid. The initial position input is ignored. The array is not exactly center, but the "translate elements in plane" of the "move" commands can be used to refine the array position.

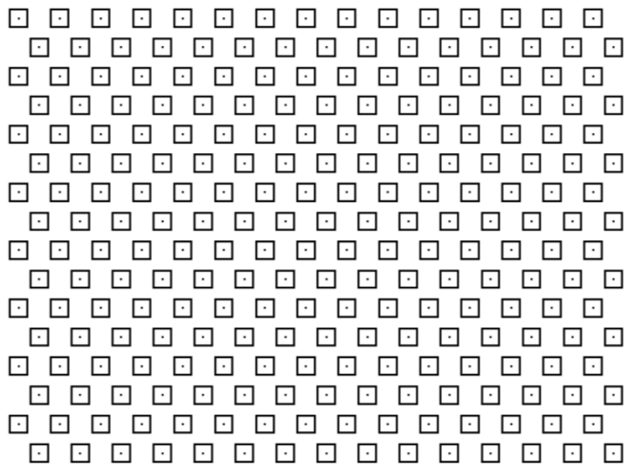

Figure 22 240 Elements in Triangularly Spaced Array Confined to a Rectangle

Array of Arrays

XADEF can be used to define a small array which we array in a larger array by duplicating the small array and translating as a group as elements in an array. We will generate a 333 element array by first generating a 37 element hexagonal array.

```
C:\Arrays>xadef
Enter input 0 keyboard, 1 file 0
Enter File Name sqh333.arr
Enter units: 1 in., 2 ft, 3 cm, 4 m 3
New File? y
Enter Frequency (GHz) 1
xadef: ad,he,37
Enter Ampl (dB), Phase 0,0
Enter Rotation of Hex 0
Enter Element Spacing cm  18
move: ta
Circular Taylor Distribution Sampling
Enter Sidelobe Level (dB) 25
Enter Number of First Unchanged Zero of Uniform Distribution 6
Maximum Radius =     54.000
Enter Normalizing Ellipse Radii, Angle of 1st Radius 54,54,0
Density taper w/ uniform amplitude? n
move: ex
Enter Final New Z axis Rotation of Antennas (array) 0
xadef: ad,re,9,ar
Enter Ampl (dB), Phase 0,0
Enter Number of Elements along X-axis 3
Enter initial spacings in X,Y axes cm  150,120
Array along X-axis
Enter Axis Spacing: 1 Uniform, 2 Bratkovic  3 Geometric, 4 Uneven Taylor 1
Enter 1) Uniform Amplitude Distribution
      2) Area Sampling of Taylor Distribution
      3) Point Sampling of Taylor Distribution
      4) Zero Sampled Taylor Distribution
      5) Point Sampled Bayliss Distribution
      6) Zero Sampled Bayliss Distribution
      7) Chebyshev array 1
Enter Quadratic Phase Factor, S 0
Array along Y-axis
Enter Axis Spacing: 1 Uniform, 2 Bratkovic  3 Geometric, 4 Uneven Taylor 1
Enter 1) Uniform Amplitude Distribution
      2) Area Sampling of Taylor Distribution
      3) Point Sampling of Taylor Distribution
      4) Zero Sampled Taylor Distribution
      5) Point Sampled Bayliss Distribution
      6) Zero Sampled Bayliss Distribution
      7) Chebyshev array 1
Enter Quadratic Phase Factor, S 0
move:
```

The “move” command “ta” is used to taper the three ring hex array to approximate the 30 dB circular Taylor amplitude distribution. This subarray is used in a 3 x 3 uniform amplitude rectangular array. Initially XADEF generates a 370 element array because the first 37 elements are duplicated. It is necessary to remove these elements. We use the command: ‘ad,el,-37’ to erase the extra subarray.

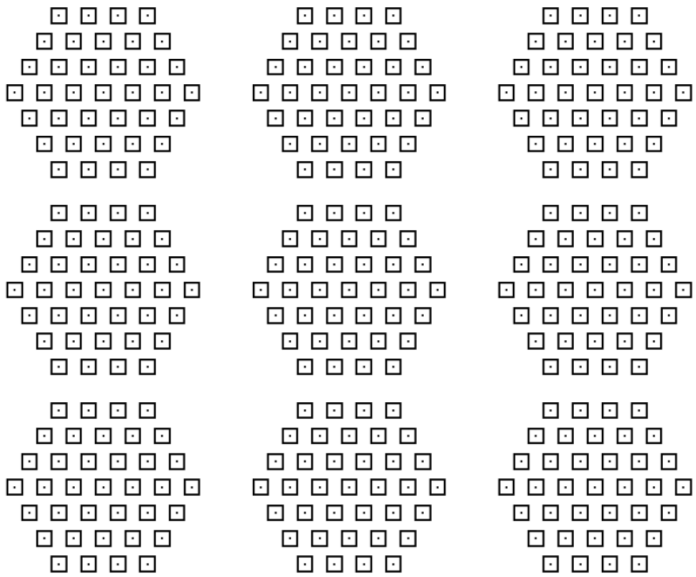

Figure 23 3 x 3 Array of 37 element Hexagonal Arrays

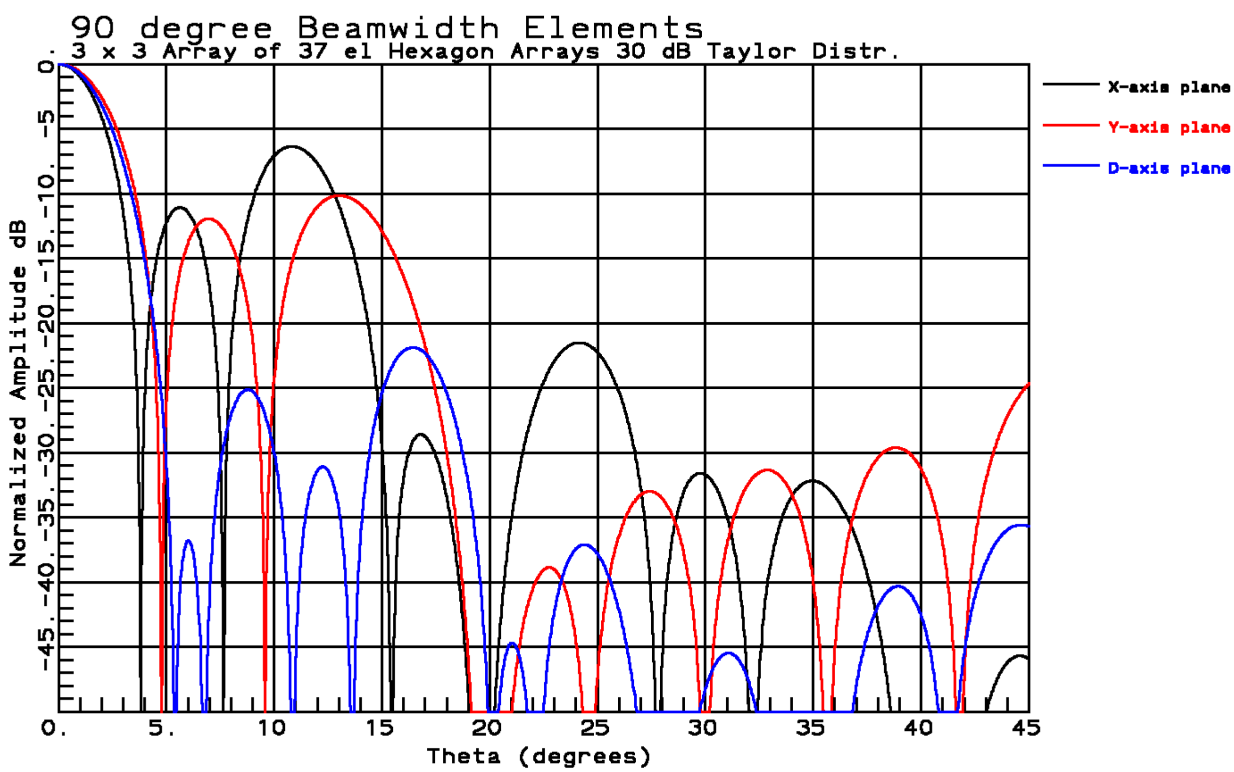

Figure 24 3 x 3 Array of 37 element Hexagonal Arrays

Figure 24 shows the first grating lobes in the *X*-axis and *Y*-axis planes between 10° and 15°. The array is not practical but illustrates how XADEF can form larger arrays using subarrays.

Move commands

Circular Rim Confined Array

Start with a 20 x 20 rectangular array with a uniform distribution. Remove the corners to limit the array to a circle with a diameter 285 cm with 15 cm spaced elements using the move command: 'ci'.

```
C:\Arrays>xadef
Enter input 0 keyboard, 1 file 0
Enter File Name sqci284.arr
Enter units: 1 in., 2 ft, 3 cm, 4 m 3
New File? y
Enter Frequency (GHz) 1
xadef: ad,re,400
Enter Ampl (dB), Phase 0,0
Enter Number of Elements along X-axis 20
Enter initial spacings in X,Y axes cm  15,15
Array along X-axis
Enter Axis Spacing: 1 Uniform, 2 Bratkovic  3 Geometric, 4 Uneven Taylor 1
Enter 1) Uniform Amplitude Distribution
      2) Area Sampling of Taylor Distribution
      3) Point Sampling of Taylor Distribution
      4) Zero Sampled Taylor Distribution
      5) Point Sampled Bayliss Distribution
      6) Zero Sampled Bayliss Distribution
      7) Chebyshev array 1
Enter Quadratic Phase Factor, S 0
Array along Y-axis
Enter Axis Spacing: 1 Uniform, 2 Bratkovic  3 Geometric, 4 Uneven Taylor 1
Enter 1) Uniform Amplitude Distribution
      2) Area Sampling of Taylor Distribution
      3) Point Sampling of Taylor Distribution
      4) Zero Sampled Taylor Distribution
      5) Point Sampled Bayliss Distribution
      6) Zero Sampled Bayliss Distribution
      7) Chebyshev array 1
Enter Quadratic Phase Factor, S 0
move: ci
Maximum Radius =    201.525
Enter Normalizing Ellipse Radii, Angle of 1st Radius 142.8,142.8,0
 284 Elements in array, OK? y
move: ex
Enter Final New Z axis Rotation of Antennas (array) 0
xadef: ex
Another file? n
Stop - Program terminated.

C:\Arrays>
```

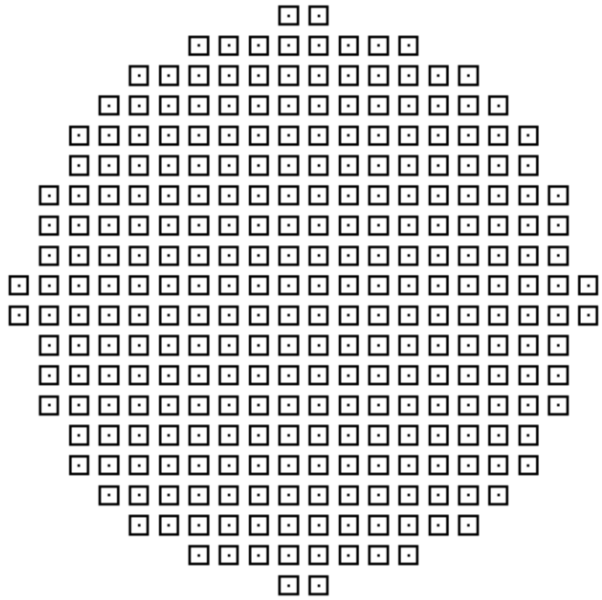

Figure 25 Circularized Uniform Square Array to 284 elements

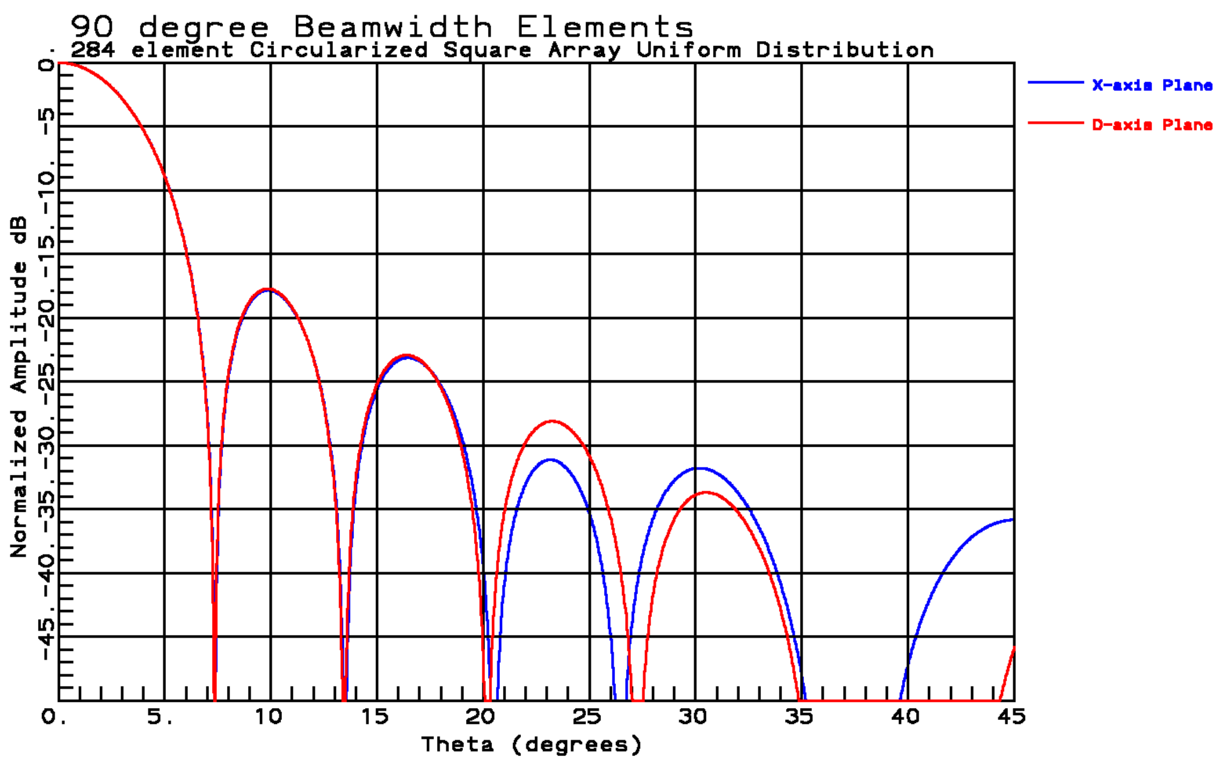

Figure 26 Pattern of Circularized Uniform Square Array to 284 elements

Figure 26 shows that circularizing the 284 elements in square array produce approximately a uniform circular aperture pattern.

Taylor Distribution Applied to Circularized Square Array

We start with a 20 x 20 uniformly space square array and use the move command to circularize it to 284 elements as above. The move command: 'ta', alters the element amplitude distribution to approximate a circular Taylor distribution with 30 dB sidelobes.

```
move: ci
Maximum Radius =    201.525
Enter Normalizing Ellipse Radii, Angle of 1st Radius 142.8,142.8,0
 284 Elements in array, OK? y
move: ta
Circular Taylor Distribution Sampling
Enter Sidelobe Level (dB) 30
Enter Number of First Unchanged Zero of Uniform Distribution 6
Maximum Radius =    142.697
Enter Normalizing Ellipse Radii, Angle of 1st Radius 142.7,142.7,0
Density taper w/ uniform amplitude? n
move: ex
Enter Final New Z axis Rotation of Antennas (array) 0
xadef: ex
Another file? n
Stop - Program terminated.
```

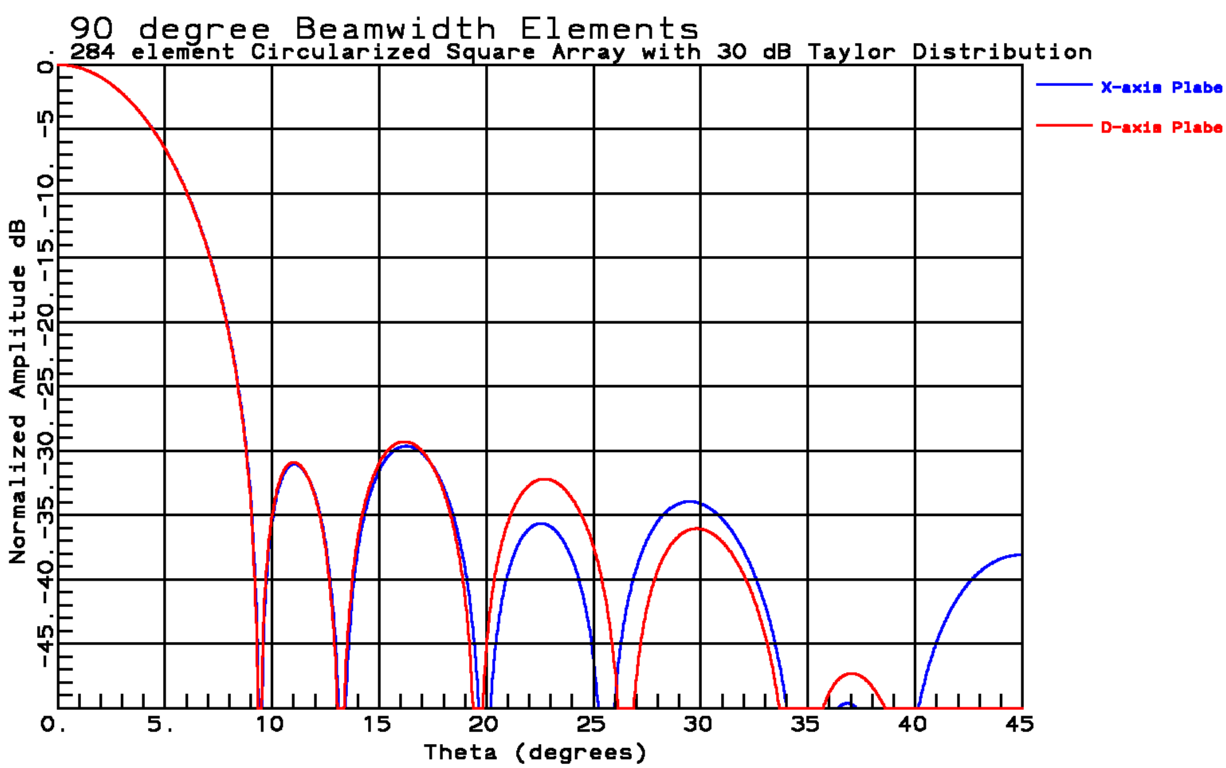

Figure 27 Pattern of Circularized Uniform Square Array to 284 elements

Figure 27 illustrates that the 284 elements in the circularized array using uniformly spaced element with a circular 30 dB Taylor amplitude distribution applied produces close to 30 dB sidelobes.

Taylor Distribution Applied to Circularized Square Array by Thinning

We start with a 20 x 20 uniformly space square array and use the move command to circularize it to 284 elements as above. The move command: 'ta', thins the array using uniform amplitude elements to approximate a circular Taylor distribution with 30 dB sidelobes.

```
move: ci
Maximum Radius =    201.525
Enter Normalizing Ellipse Radii, Angle of 1st Radius 142.8,142.8,0
 284 Elements in array, OK? y
move: ta
Circular Taylor Distribution Sampling
Enter Sidelobe Level (dB) 30
Enter Number of First Unchanged Zero of Uniform Distribution 6
Maximum Radius =    142.697
Enter Normalizing Ellipse Radii, Angle of 1st Radius 142.7,142.7,0
Density taper w/ uniform amplitude? y
Symmetrical thinning? y
Enter density factor k (k=1 natural) 1
 140 Elements in array, OK? n
Enter density factor k (k=1 natural) 1.2
 180 Elements in array, OK? y
move: ex
Enter Final New Z axis Rotation of Antennas (array) 0
xadef: ex
Another file? n
Stop - Program terminated.
```

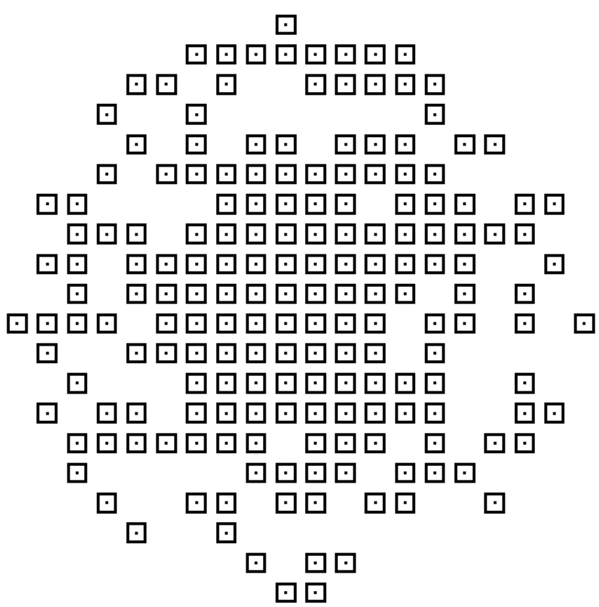

Figure 28 Thinned Array for 25 dB Taylor Distribution using Uniform Amplitude elements starting with Circularized Square Array by reducing to 183 elements from 284

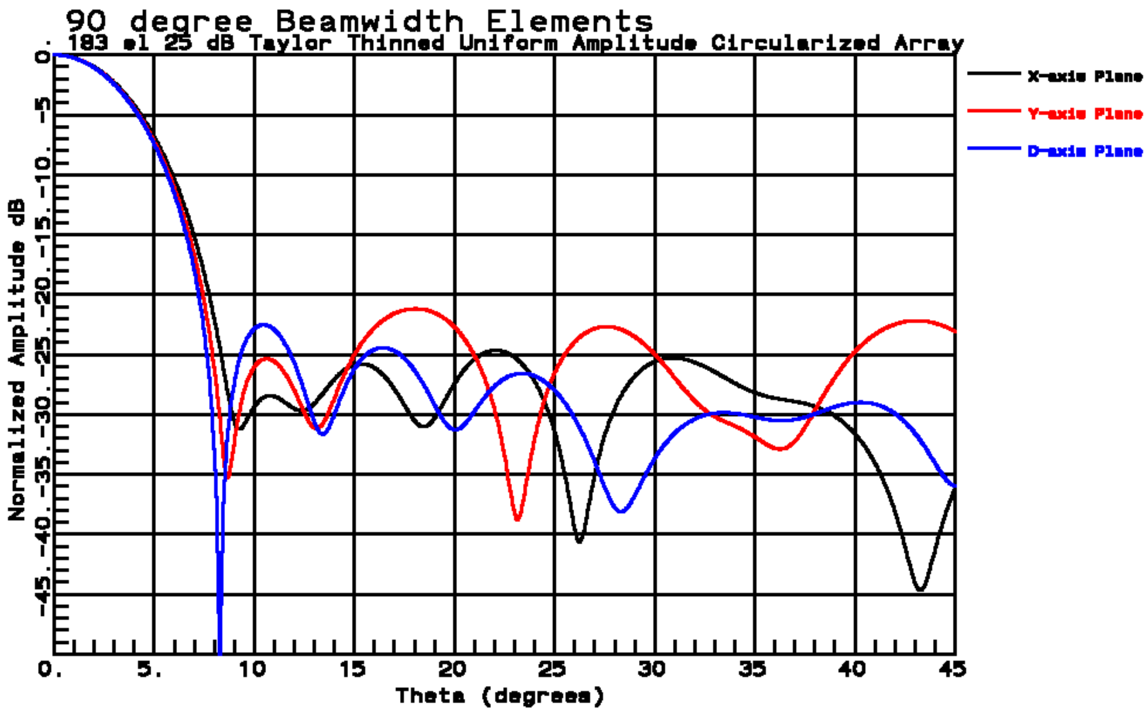

Figure 29 Thinned Array for 25 dB Taylor Distribution using Uniform Amplitude elements starting with Circularized Square Array by reducing to 183 elements from 284

Density thinning of the circularized array fails to generate a pattern with the specified sidelobes. The method works better for arrays starting with a larger number of elements. However, the thinning did reduce the sidelobes compared to a uniform circular distribution.

Array Confined in Polygon

The move command 'po' confines the array by a predefined polygon where it is defined in a file listing its vertices. For this example, we start with a 21 x 15 element rectangular array with 15 cm spacing. The confining polygon has 8 vertices to remove element from the corners by using miters.

```
8      Rectangle with mitered corners
-80.  -105.0
80.0  -105.0
150.   -35.0
150.    35.0
80.0   105.0
-80.   105.0
-150.   35.0
-150.  -35.0
```

```
move: po
X range:  -1.50000E+02    1.50000E+02
Y range:  -1.05000E+02    1.05000E+02

Polygon defined by vertices listed in file:
Number of vertices arranged CCW in X-Y plane

Number
X-axis, Y-axis vertex, etc.

Enter Polygon file rectmiter.txt
Polygon ranges
X range:  -1.50000E+02    1.50000E+02
Y range:  -1.05000E+02    1.05000E+02
Enter scale factor 1
 255 Elements in array, OK? y
move: ex
Enter Final New Z axis Rotation of Antennas (array) 0
xadef: ex
Another file? n
Stop - Program terminated.

C:\Arrays>
```

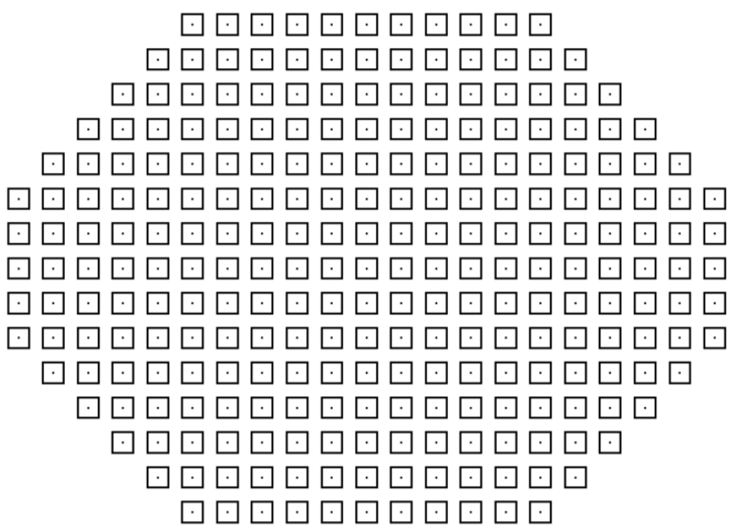

Figure 30 Originally 21 x 15 Rectangular Array Confined by Polygon to Miter Corners to 255 el.

The spacing arrangement is independent of the confining polynomial. Below Figure 31 illustrates using the same polygon with hexagonally spaced elements

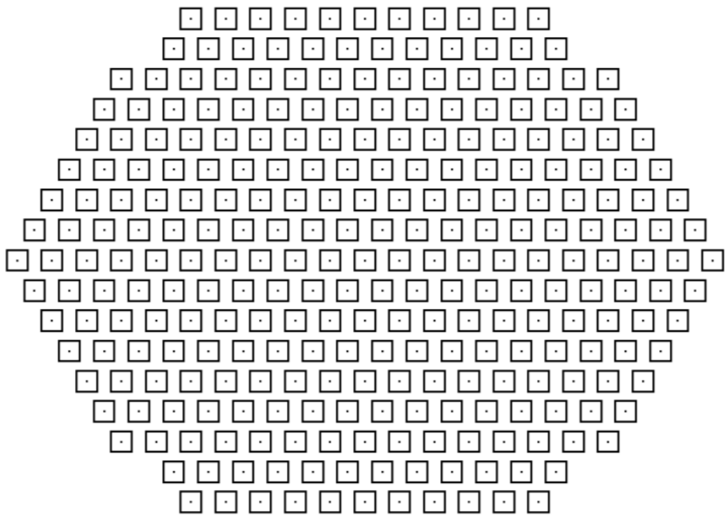

Figure 31 Originally 331 element Hexagon Array Confined by Polygon to Miter Corners to 277 el.

Array Confined by Spline Curves

We can confine an array using spline curves along the *x*-axis. We write a text file containing columns of *x*- and *y*-axis points of the spline curve. This example starts with a 315 element (21 x 15) array width with 15 cm spacing that XADEF centers on the origin.

-150.000000	10.500000
-135.000000	12.812577
-120.000000	19.523945
-105.000000	29.977148
-90.000000	43.148945
-75.000000	57.750000
-60.000000	72.351051
-45.000000	85.522850
-30.000000	95.976051
-15.000000	102.687424
0.000000E+00	105.000000
15.000000	102.687424
30.000000	95.976051
45.000000	85.522850
60.000000	72.351051
75.000000	57.750000
90.000000	43.148945
105.000000	29.977148
120.000000	19.523945
135.000000	12.812577
150.000000	10.500000

A similar curve is written with negative *y*-axis values used for the lower confinement.

```
move: co,sp
Maximum Radius =   1.83098E+02
X-axis  Min:  -1.50000E+02  Max:   1.50000E+02
Y-axis  Min:  -1.05000E+02  Max:   1.05000E+02

Spline of Y points along X-axis
X, Y values with points counted
Enter file with table of X and Y spc2u.txt
Enter columns with X, Y data 1,2
File X-axis Max:  -1.50000E+02 Min:   1.50000E+02
     Y-axis Max:   1.05000E+01 Min:   1.05000E+02
Enter X-axis shift, Y boundary increase 0,0
Enter size of array element X,Y 0,0
Remove elements: +1 above, -1 below spline curve 1
 230 Elements in array, OK? y
move: co,sp
Maximum Radius =   1.83098E+02
X-axis  Min:  -1.50000E+02  Max:   1.50000E+02
Y-axis  Min:  -1.05000E+02  Max:   9.00000E+01

Spline of Y points along X-axis
X, Y values with points counted
Enter file with table of X and Y spc2d.txt
Enter columns with X, Y data 1,2
File X-axis Max:  -1.50000E+02 Min:   1.50000E+02
     Y-axis Max:  -1.05000E+02 Min:  -1.05000E+01
Enter X-axis shift, Y boundary increase 0,0
Enter size of array element X,Y 0,0
Remove elements: +1 above, -1 below spline curve -1
 145 Elements in array, OK? y
move: ex
Enter Final New Z axis Rotation of Antennas (array) 0
xadef: ex
Another file? n
Stop - Program terminated.
```

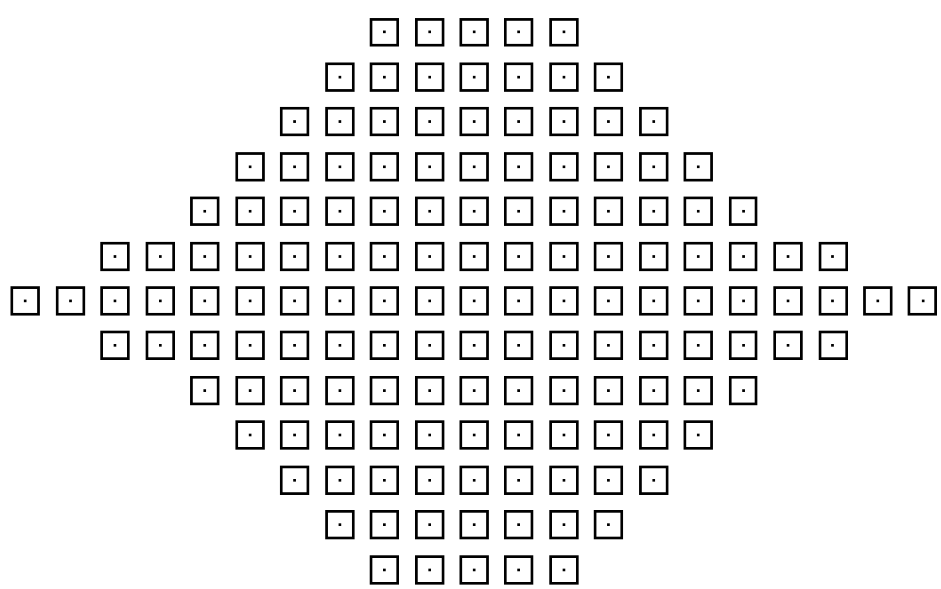

Figure 32 Original 315 (21 x 15) element Array Confined by Upper and Lower Spline Curves to 145 el

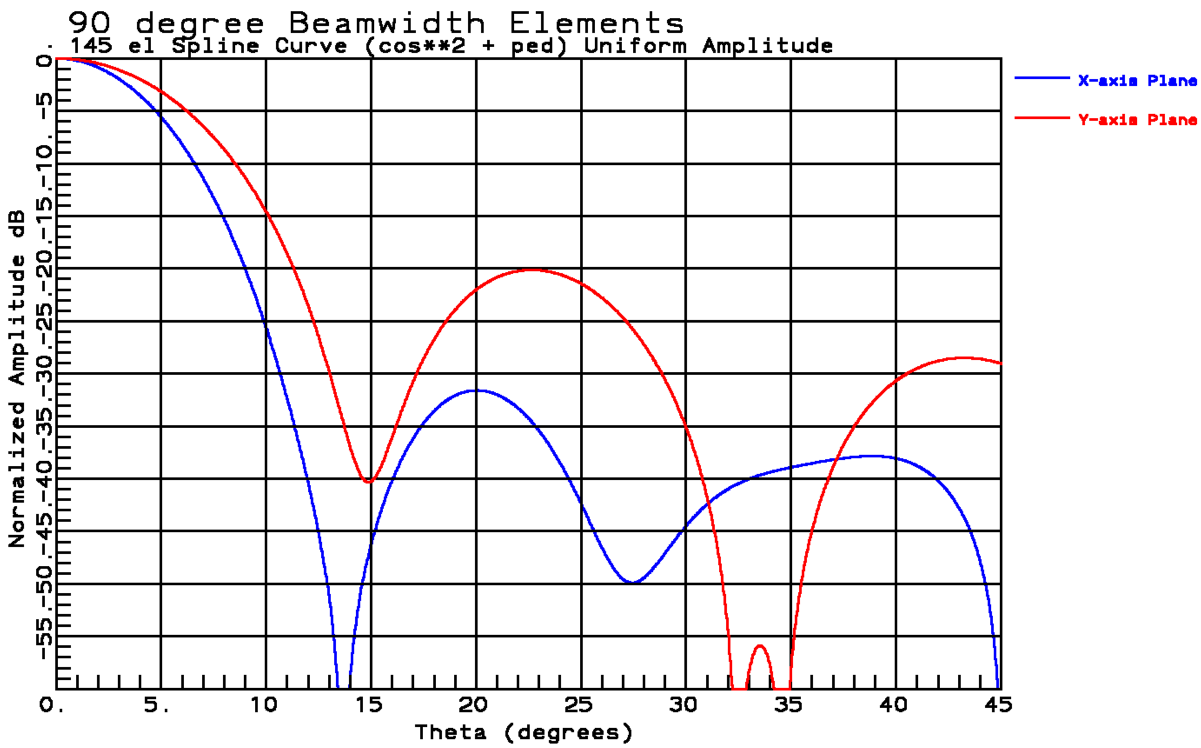

Figure 33 145 element Array Confined by Upper and Lower (1 – Ped)Cos^2(πx/a)+Ped Spline Curves

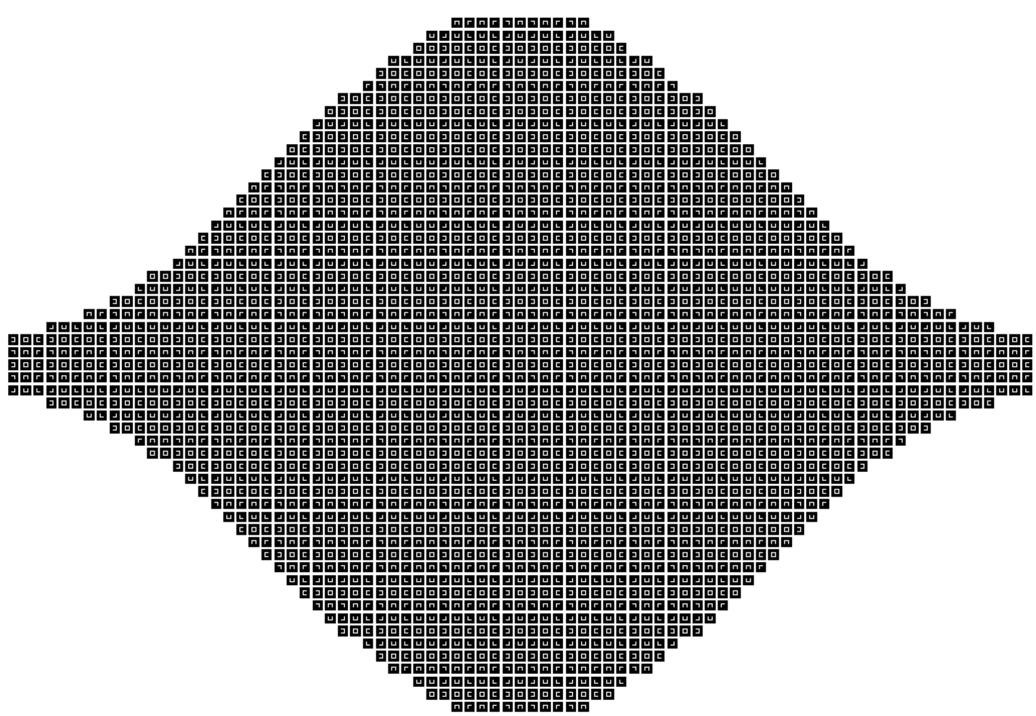

Figure 34 2463 element Array Confined by Upper and Lower (1 – Ped)Cos^2(πx/a)+Ped Spline Curves

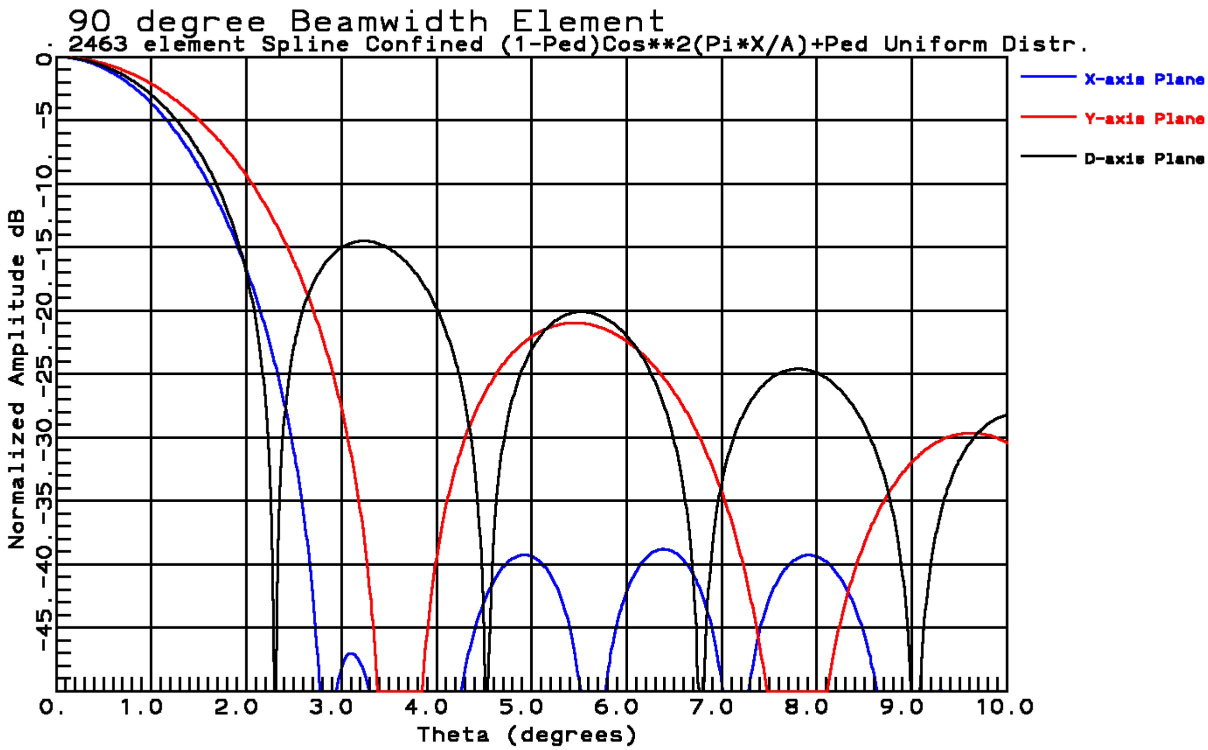

Figure 35 2463 element Array Confined by Upper and Lower (1 – Ped)Cos^2(πx/a)+Ped Spline Curves

Increasing the number of elements in the spline confined array produces a pattern close to the cosine squared plus 20 dB pedestal distribution that has low first two sidelobes and approximately 40 dB down next three sidelobes as illustrated in Figure 35 *x*-axis plane plot.

Circular Array

The command: ad,ci,# in XADEF generates a circular array with the elements evenly spaced.

```
C:\Arrays>xadef
Enter input 0 keyboard, 1 file 0
Enter File Name circle36.arr
Enter units: 1 in., 2 ft, 3 cm, 4 m 3
New File? y
Enter Frequency (GHz) 1
xadef: ad,ci,36
Enter Ampl (dB), Phase 0,0
Circular Array in X-Y Plane
Enter Radius cm  85.94
Enter Start Rotation Angle 0
move: ex
Enter Final New Z axis Rotation of Antennas (array) 0
xadef: ex
Another file? n
Stop - Program terminated.
```

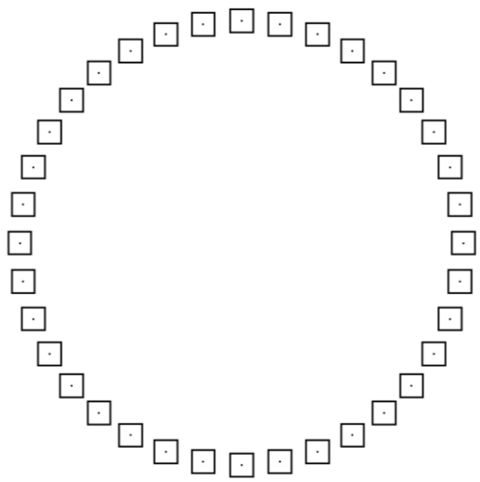

Figure 36 Circular Array with 36 evenly spaced elements

Elliptical Arc Array

The ad,ep,# command generates an evenly spaced elliptical array.

```
C:\Arrays>xadef
Enter input 0 keyboard, 1 file 0
Enter File Name ellip36.arr
Enter units: 1 in., 2 ft, 3 cm, 4 m 3
New File? y
Enter Frequency (GHz) 1
xadef: ad,ep,36
Enter Ampl (dB), Phase 0,0
Elliptical Array in X-Y Plane
Enter Major, Minor Diameter cm  222.94,111.47
move: ex
Enter Final New Z axis Rotation of Antennas (array) 0
xadef: ex
Another file? n
Stop - Program terminated.
```

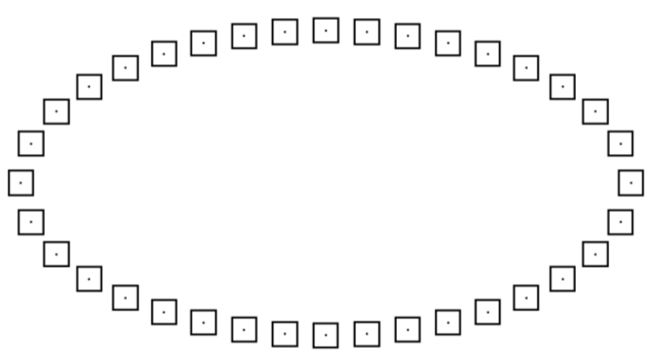

Figure 37 Elliptical Array with 36 evenly spaced elements

Cap Arrays

The geometry of cap arrays determines the number of elements.

Planar

```
C:\Arrays>xadef
Enter input 0 keyboard, 1 file 0
Enter File Name cicap1.arr
Enter units: 1 in., 2 ft, 3 cm, 4 m 3
New File? y
Enter Frequency (GHz) 1
xadef: ad,ca,1
Enter Ampl (dB), Phase 0,0
Cap Array
Enter Type: 1 Planar, 2 Spherical, 3 Cone 1
Enter Spacing between Elements cm  15
Enter Start Rotation Angle 0
Enter Radius cm  162
   347 Elements added
move: ex
Enter Final New Z axis Rotation of Antennas (array) 0
xadef: ex
Another file? n
Stop - Program terminated.
```

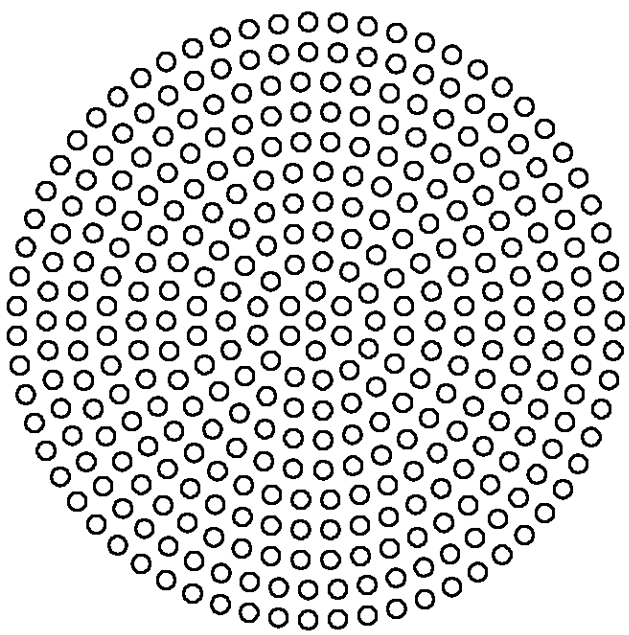

Figure 38 Planar Cap array with 347 elements and maximum radius = 162

The cap array starts with a hexangular array which is different from the ring filled array (Figure 16) with its 373 elements for a maximum radius of 162 cm. The move commands can be used to taper the amplitude across the array for lower sidelobes, for example, circular Taylor distribution.

3-d Arrays

The following XADEF array generation methods produce arrays in 3 dimensions where elements have individual pointing defined by Euler angles: 1) *z*-axis rotation, 2) new *y*-axis rotation, 3) new *z*-axis rotation. We continue with the cap array.

Spherical Cap

We will generate a spherical cap array rotated out to 30° with an arc length of 120 cm. The spherical cap has a radius = 229.2 cm and with the sphere centered at the origin. The array is translated by *z*-axis = -229.2 cm automatically to place the spherical cap center element at the origin.

```
C:\Arrays>xadef
Enter input 0 keyboard, 1 file 0
Enter File Name cicap2.arr
Enter units: 1 in., 2 ft, 3 cm, 4 m 3
New File? y
Enter Frequency (GHz) 1
xadef: ad,ca,1
Enter Ampl (dB), Phase 0,0
Cap Array
Enter Type: 1 Planar, 2 Spherical, 3 Cone 2
Enter Spacing between Elements cm  15
Enter Start Rotation Angle 0
Enter Radius of Cap cm  229.2
Enter Angle to Edge of Cap 30
   222 Elements added
move: ex
Enter Final New Z axis Rotation of Antennas (array) 0
xadef: li,1
Element No:      1
          Location: X =      0.000   Y =      0.000   Z =      0.000
          Ampl(dB) =    0.00  Phase =    0.00
Euler Angles =      0.00      0.00      0.00
xadef: li,2
Element No:      2
          Location: X =     12.981   Y =      7.495   Z =     -0.491
          Ampl(dB) =    0.00  Phase =    0.00
Euler Angles =     30.00      3.75    -30.00
xadef: _
```

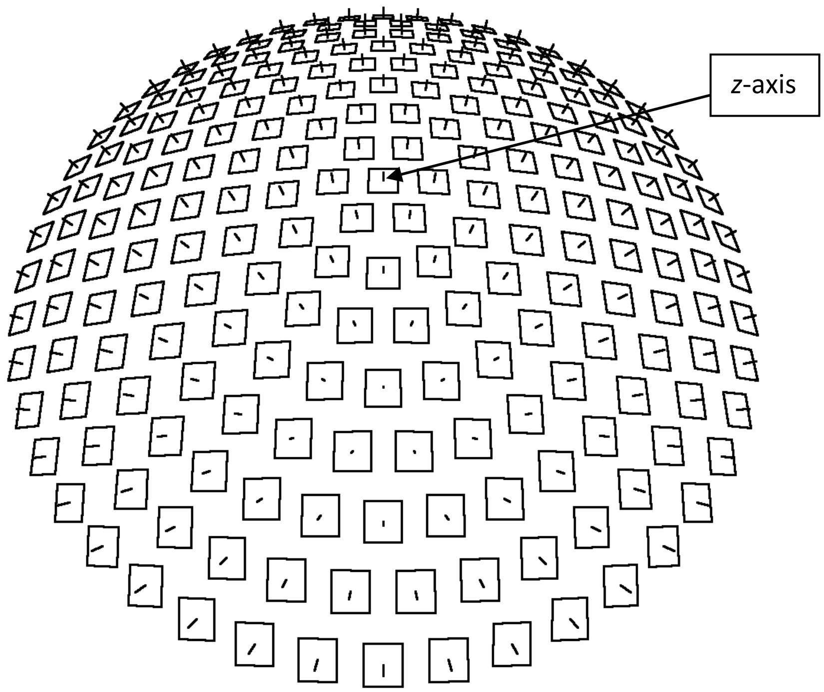

Figure 39 Spherical Cap array with 222 elements and on radius = 229.2 cm with 15 cm spacing

Partial listing of elements illustrates the rotation of the elements to point its boresight normal to the sphere and a final new *z*-axis rotation to align polarization planes. As specified the maximum angle is 30°.

File:cicap2.arr

		Location			Element			
No	X	Y	Z	Ampl(dB)	Phase	Euler Angles		
1	0.000	0.000	0.000	0.00	0.00	0.00	0.00	0.00
2	12.981	7.495	-0.491	0.00	0.00	30.00	3.75	-30.00
3	0.000	14.989	-0.491	0.00	0.00	90.00	3.75	-90.00
4	-12.981	7.495	-0.491	0.00	0.00	150.00	3.75	-150.00
5	-12.981	-7.495	-0.491	0.00	0.00	-150.00	3.75	150.00
6	0.000	-14.989	-0.491	0.00	0.00	-90.00	3.75	90.00
7	12.981	-7.495	-0.491	0.00	0.00	-30.00	3.75	30.00
8	29.914	0.000	-1.961	0.00	0.00	0.00	7.50	0.00
187	0.000	114.592	-30.703	0.00	0.00	90.00	30.00	-90.00
188	-14.957	113.612	-30.703	0.00	0.00	97.50	30.00	-97.50
189	-29.659	110.688	-30.703	0.00	0.00	105.00	30.00	-105.00
190	-43.853	105.870	-30.703	0.00	0.00	112.50	30.00	-112.50
191	-57.296	99.240	-30.703	0.00	0.00	120.00	30.00	-120.00
192	-69.759	90.912	-30.703	0.00	0.00	127.50	30.00	-127.50
193	-81.029	81.029	-30.703	0.00	0.00	135.00	30.00	-135.00
194	-90.912	69.759	-30.703	0.00	0.00	142.50	30.00	-142.50
195	-99.240	57.296	-30.703	0.00	0.00	150.00	30.00	-150.00
196	-105.870	43.853	-30.703	0.00	0.00	157.50	30.00	-157.50
197	-110.688	29.659	-30.703	0.00	0.00	165.00	30.00	-165.00

198	-113.612	14.957	-30.703	0.00	0.00	172.50	30.00	-172.50
199	-114.592	0.000	-30.703	0.00	0.00	-180.00	30.00	180.00
200	-113.612	-14.957	-30.703	0.00	0.00	-172.50	30.00	172.50
201	-110.688	-29.659	-30.703	0.00	0.00	-165.00	30.00	165.00
202	-105.870	-43.853	-30.703	0.00	0.00	-157.50	30.00	157.50
203	-99.240	-57.296	-30.703	0.00	0.00	-150.00	30.00	150.00
204	-90.912	-69.759	-30.703	0.00	0.00	-142.50	30.00	142.50
205	-81.029	-81.029	-30.703	0.00	0.00	-135.00	30.00	135.00
206	-69.759	-90.912	-30.703	0.00	0.00	-127.50	30.00	127.50
207	-57.296	-99.240	-30.703	0.00	0.00	-120.00	30.00	120.00
208	-43.853	-105.870	-30.703	0.00	0.00	-112.50	30.00	112.50
209	-29.659	-110.688	-30.703	0.00	0.00	-105.00	30.00	105.00
210	-14.957	-113.612	-30.703	0.00	0.00	-97.50	30.00	97.50
211	0.000	-114.592	-30.703	0.00	0.00	-90.00	30.00	90.00

Spherical Array (hemispherical)

The cap array with the spherical option can produce an array with maximum edge angle of 90° and produce a hemispherical array. The routine spaces elements equally in a ring which causes variations in the local environment of the elements.

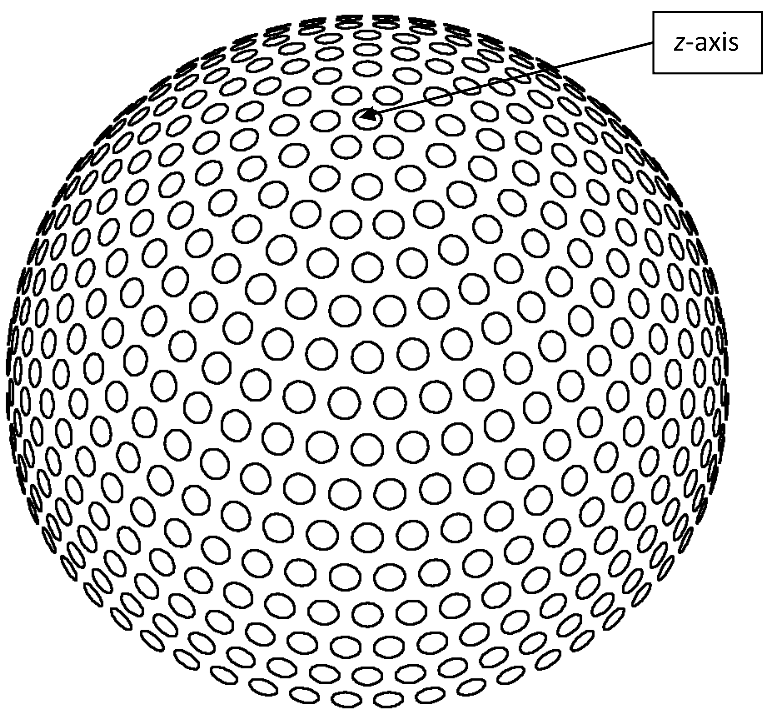

Figure 40 615 element hemispherical array generated from "cap, array" command

The hemispherical cap array starts with 6-way symmetry around the *z*-axis element which increases to 12 elements in the next ring. Figure 41 shows the base of the array at the 90° edge angle. Figure 40 and 41 illustrate that the elements are more or less uniformly distributed locally along the *x*-axis (vertical axis on illustrations) but starts to deviate from this when rotating ϕ away from zero.

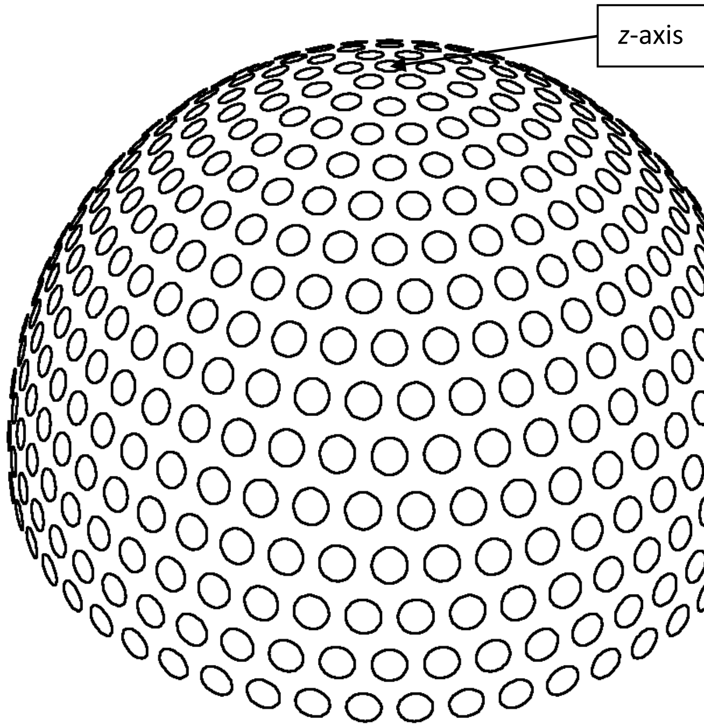

Figure 41 615 element hemispherical array generated from “cap, array” command illustrating base

Spherical Array command: ad,sp

This command produces spherical array, similar to the spherical cap, but allows specification of the symmetry. Using the spherical cap array as a guide, we see that 6-way symmetry produces with a uniform local distribution about the *z*-axis.

```
C:\Arrays>xadef
Enter input 0 keyboard, 1 file 0
Enter File Name spcap6a.arr
Enter units: 1 in., 2 ft, 3 cm, 4 m 3
New File? y
Enter Frequency (GHz) 1
xadef: ad,sp,1
Enter Ampl (dB), Phase 0,0
Spherical Array
Enter Sphere Radius cm  150
Enter symmetry order (6 for hexagon) 6
Enter spacing: 1 length, 2 Degrees 1
Enter spacing between elements cm  15
Enter Angular Extent of Array (theta) 90
Enter Final Rotation of E Plane 0
Elements in Array:   607
move: ex
Enter Final New Z axis Rotation of Antennas (array) 0
xadef: ex
Another file? n
Stop - Program terminated.
```

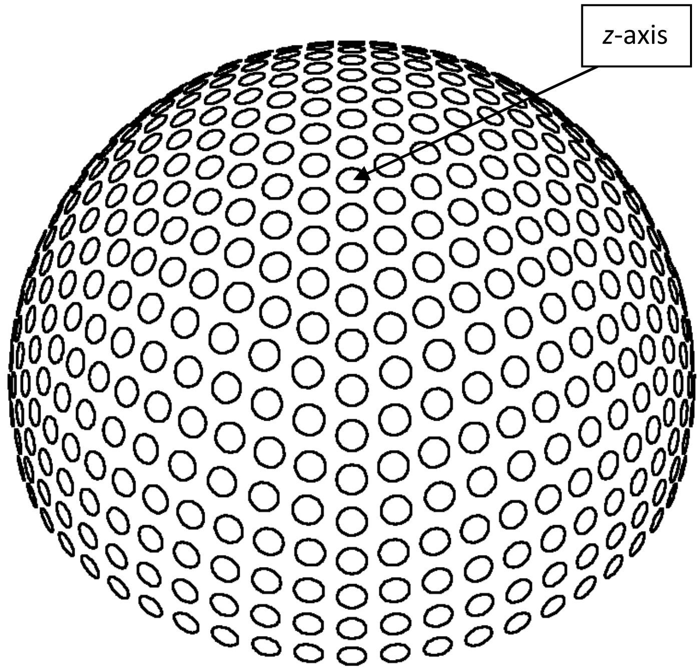

Figure 42 607 element hemispherical array generated from "add, spherical" command
The 'add, spherical" array command generates an array with hexagon symmetry about the *z*-axis. Looking from above we see the hexagon pattern continuing into every row easily seen in the 2nd row with 12 elements. Like the spherical cap array, generated from elements equally spaced in rings, this layout method produces far different local element distribution near θ = 90°. Figure 43 illustrates how the array which started as locally hexagonal near the *z*-axis has become locally rectangular at 90°.

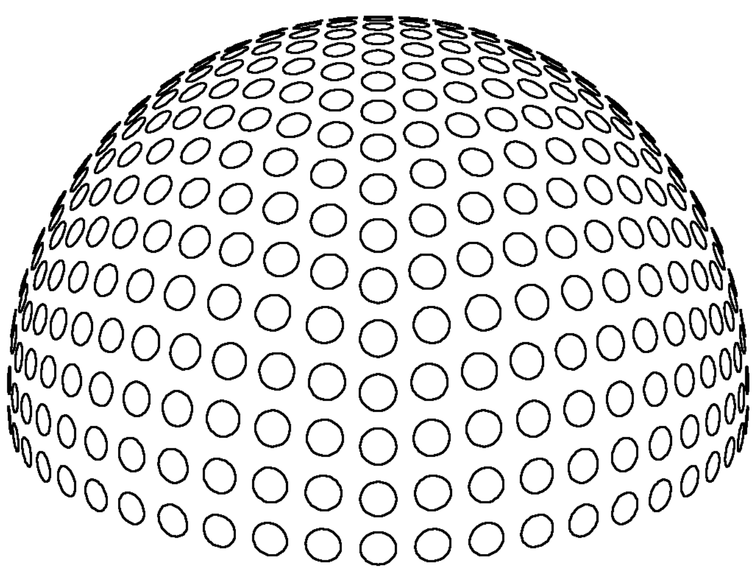

Figure 43 Base region of 607 element hemispherical array generated from "add, spherical" command

Subdivided Icosahedron Projected to Sphere Array

This method takes the regular solid: icosahedron and subdivides each of the twenty equilaterals into smaller triangles whose vertices are projected radically to a circumscribing sphere. This method is also used to locate the vertices of geodesic dome radomes. One method is to use the vertices as locations of the elements. The command "add, icosahedrons" uses the centers of the triangles project to the sphere for the location of elements. This method produces more locally equally spaced elements, but fails along the edges of the icoshedron. Its command listing is similar to the spherical array.

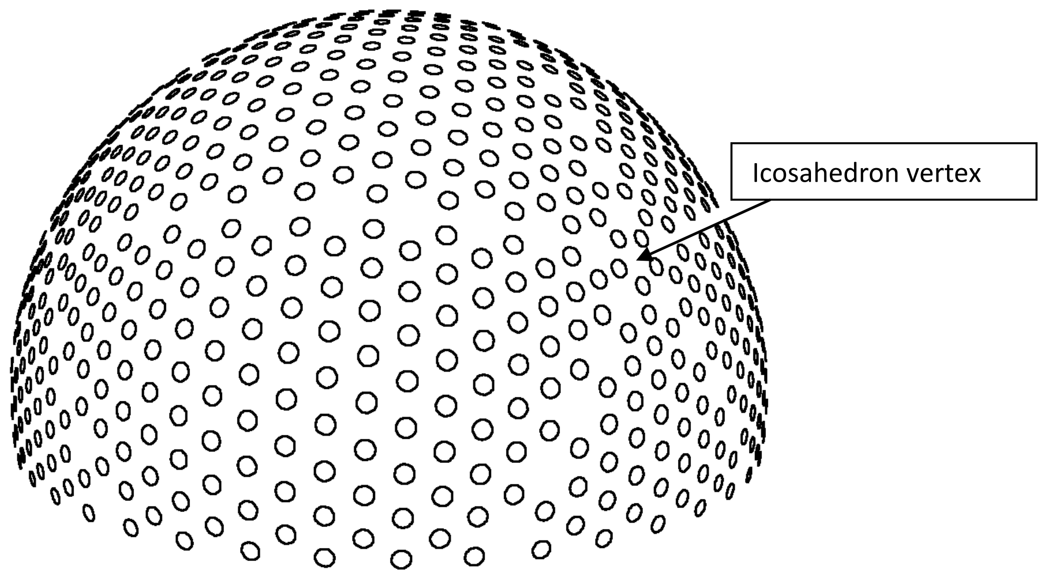

Figure 44 Hemispherical array generated from "add, icosahedron" command

Cone Cap

The cone cap places an element at its apex (origin) pointed along *z*-axis and locates additional elements on a cone sloping down from origin at a constant angle. We specify the maximum radius and cone height.

```
C:\Arrays>xadef
Enter input 0 keyboard, 1 file 0
Enter File Name concp1.arr
Enter units: 1 in., 2 ft, 3 cm, 4 m 3
New File? y
Enter Frequency (GHz) 1
xadef: ad,ca,1
Enter Ampl (dB), Phase 0,0
Cap Array
Enter Type: 1 Planar, 2 Spherical, 3 Cone 3
Enter Spacing between Elements cm  15
Enter Start Rotation Angle 0
Enter Radius cm  72
Enter Cone Height cm  76
    64 Elements added
move: ex
Enter Final New Z axis Rotation of Antennas (array) 0
xadef: li,fi

File:concp1.arr
```

	Location				Element			
No	X	Y	Z	Ampl(dB)	Phase	Euler Angles		
1	0.000	0.000	0.000	0.00	0.00	0.00	0.00	0.00
2	12.990	7.500	-15.833	0.00	0.00	30.00	46.55	-30.00
3	0.000	15.000	-15.833	0.00	0.00	90.00	46.55	-90.00
4	-12.990	7.500	-15.833	0.00	0.00	150.00	46.55	-150.00
5	-12.990	-7.500	-15.833	0.00	0.00	-150.00	46.55	150.00
6	0.000	-15.000	-15.833	0.00	0.00	-90.00	46.55	90.00
7	12.990	-7.500	-15.833	0.00	0.00	-30.00	46.55	30.00
8	30.000	0.000	-31.667	0.00	0.00	0.00	46.55	0.00
9	26.564	13.942	-31.667	0.00	0.00	27.69	46.55	-27.69
10	17.042	24.690	-31.667	0.00	0.00	55.38	46.55	-55.38
11	3.616	29.781	-31.667	0.00	0.00	83.08	46.55	-83.08
12	-10.638	28.050	-31.667	0.00	0.00	110.77	46.55	-110.77
13	-22.455	19.894	-31.667	0.00	0.00	138.46	46.55	-138.46

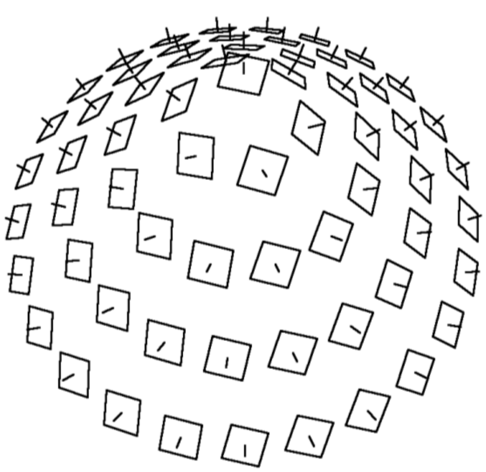

Figure 45 Conical Cap array with 64 elements and max radius = 72 cm, height = 76 with 15 cm spacing

Cone Array

The cone array starts at an initial radius and Phi angle in the *x-y* plane with the cone sloping down from this plane (negative z) given by a cone angle. A cone angle of 90° produces an array around a cylinder of a given number of rings with decreasing *z*-axis values in each new ring. The specified element spacing is measured along the cone. Negative cone angles produces increasing z with each ring, but the elements point inward.

The following inputs generate a 36 element array pointing outward with x-y positions of Figure 36.

```
C:\Arrays>xadef
Enter input 0 keyboard, 1 file 0
Enter File Name cyld36.arr
Enter units: 1 in., 2 ft, 3 cm, 4 m 3
New File? y
Enter Frequency (GHz) 1
xadef: ad,co,1
Enter Ampl (dB), Phase 0,0
Conical Array
Enter Initial Radius cm  85.94
Enter Cone Angle w.r.t. x-y plane 90
Enter Spacing between Elements cm  15
Enter Start Rotation Angle 0
Enter Number of Rings 1
Enter Limits of Phi for Array Start,Stop 0,360
Biconical Horn Feed? n
    36 Elements added
move: ex
Enter Final New Z axis Rotation of Antennas (array) 0
xadef: li,1
Element No:      1
          Location: X =     85.613   Y =      7.490   Z =      0.000
          Ampl(dB) =    0.00  Phase =    0.00
Euler Angles =       5.00     90.00      0.00
xadef: li,18
Element No:     18
          Location: X =    -85.613   Y =      7.490   Z =      0.000
          Ampl(dB) =    0.00  Phase =    0.00
Euler Angles =     175.00     90.00      0.00
```

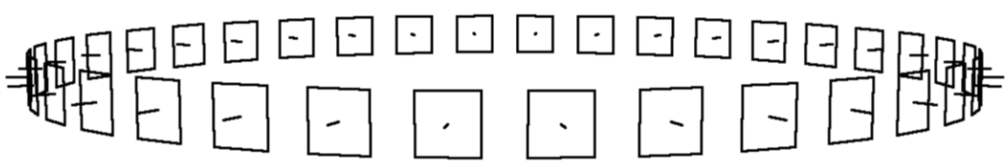

Figure 46 Cylinder (Cone = 90) array with 36 elements, radius = 85.94 cm with 15 cm spacing (perspective drawing, all elements same size)

The same array can be formed combining the circular array (Figure 36) with the “tilt, elements about plane Z-axis” move command.

```
C:\Arrays>xadef
Enter input 0 keyboard, 1 file 0
Enter File Name cyld3.arr
Enter units: 1 in., 2 ft, 3 cm, 4 m 3
New File? y
Enter Frequency (GHz) 1
xadef: ad,ci,36
Enter Ampl (dB), Phase 0,0
Circular Array in X-Y Plane
Enter Radius cm  84.75
Enter Start Rotation Angle 0
move: ti
Enter Angle w.r.t. Z axis 90
move: ex
Enter Final New Z axis Rotation of Antennas (array) 0
xadef: li,fi
```

File:cyld3.arr

	Location			Element				
No	X	Y	Z	Ampl(dB)	Phase	Euler Angles		
1	84.750	0.000	0.000	0.00	0.00	0.00	90.00	0.00
2	83.462	14.717	0.000	0.00	0.00	10.00	90.00	0.00
3	79.639	28.986	0.000	0.00	0.00	20.00	90.00	0.00
4	73.396	42.375	0.000	0.00	0.00	30.00	90.00	0.00
5	64.922	54.476	0.000	0.00	0.00	40.00	90.00	0.00
6	54.476	64.922	0.000	0.00	0.00	50.00	90.00	0.00
7	42.375	73.396	0.000	0.00	0.00	60.00	90.00	0.00
8	28.986	79.639	0.000	0.00	0.00	70.00	90.00	0.00
9	14.717	83.462	0.000	0.00	0.00	80.00	90.00	0.00
10	0.000	84.750	0.000	0.00	0.00	90.00	90.00	0.00
11	-14.717	83.462	0.000	0.00	0.00	100.00	90.00	0.00
12	-28.986	79.639	0.000	0.00	0.00	110.00	90.00	0.00
13	-42.375	73.396	0.000	0.00	0.00	120.00	90.00	0.00
14	-54.476	64.922	0.000	0.00	0.00	130.00	90.00	0.00
15	-64.922	54.476	0.000	0.00	0.00	140.00	90.00	0.00
16	-73.396	42.375	0.000	0.00	0.00	150.00	90.00	0.00
17	-79.639	28.986	0.000	0.00	0.00	160.00	90.00	0.00
18	-83.462	14.717	0.000	0.00	0.00	170.00	90.00	0.00
19	-84.750	0.000	0.000	0.00	0.00	180.00	90.00	0.00

Figure 46 illustrates this array also. The move command can also tilt the elements to another angle, such as 70°. The add cone array could also generate the same array by specifying a cone of 70° with respect to the *x-y* plane.

Array on 60° Cone

```
C:\Arrays>xadef
Enter input 0 keyboard, 1 file 0
Enter File Name cone60.arr
Enter units: 1 in., 2 ft, 3 cm, 4 m 3
New File? y
Enter Frequency (GHz) 1
xadef: ad,co,1
Enter Ampl (dB), Phase 0,0
Conical Array
Enter Initial Radius cm  47.747
Enter Cone Angle w.r.t. x-y plane 60
Enter Spacing between Elements cm  15
Enter Start Rotation Angle 0
Enter Number of Rings 6
Enter Limits of Phi for Array Start,Stop 0,360
Biconical Horn Feed? n
   167 Elements added
move: ex
Enter Final New Z axis Rotation of Antennas (array) 0
xadef: ex
Another file? n
Stop - Program terminated.
```

This produces an array starting with 20 elements in the first ring. The remaining rings have 23, 26, 29, 33, and 36 elements spaced approximately 15 cm as the ring diameter increases on the 60° cone.

File:cone60.arr

		Location			Element			
No	X	Y	Z	Ampl(dB)	Phase	Euler Angles		
1	47.159	7.469	0.000	0.00	0.00	9.00	60.00	0.00
21	55.247	0.000	-12.990	0.00	0.00	0.00	60.00	0.00
44	62.290	7.563	-25.981	0.00	0.00	6.92	60.00	0.00
70	70.247	0.000	-38.971	0.00	0.00	0.00	60.00	0.00
99	77.395	7.390	-51.962	0.00	0.00	5.45	60.00	0.00
132	85.247	0.000	-64.952	0.00	0.00	0.00	60.00	0.00
167	83.952	-14.803	-64.952	0.00	0.00	-10.00	60.00	0.00

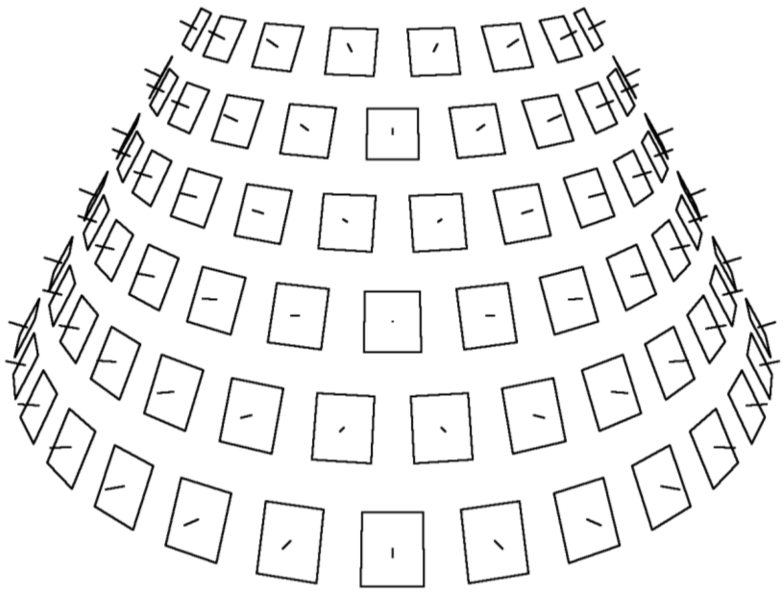

Figure 47 60° Cone array with 167 elements, radius = 47.75 cm with 15 cm spacing (perspective drawing, all elements same size) only front elements shown

3-d Arrays on Developable Surfaces

A developable object can be made from a flat surface rolled into the 3-d surface, such as, a cylinder, cone, or tape helix. We will add an array in the *x-y* plane while it is flat and the array is rolled into place. XADEF computes its new array element position and pointing when generating the array from the flat array positions. The "move" commands reposition elements of a flat array.

Project elements on a Cone: "pr, co" command

Generate the array in the *x-y* plane and the move will command will roll it into a cone.

```
C:\Arrays>xadef
Enter input 0 keyboard, 1 file 0
Enter File Name chex271.arr
Enter units: 1 in., 2 ft, 3 cm, 4 m 3
New File? y
Enter Frequency (GHz) 1
xadef: ad,he,271
Enter Ampl (dB), Phase 0,0
Enter Rotation of Hex 0
Enter Element Spacing cm  18
move: pr,co
Projection of Array from X-Y plane to Cone
Is the array orientated so that the X-axis will be around
the cone and Y axis along the cone axis? y
Enter Radius of Cone at Center of Array  cm  150
Enter Cone Angle (0 for cylinder) 30
Enter Rotation from X-axis of Center of array on Cone 0
Enter Orientation of Cone Axis: Theta, Phi 0,0
move: ex
Enter Final New Z axis Rotation of Antennas (array) 0
xadef: ex
```

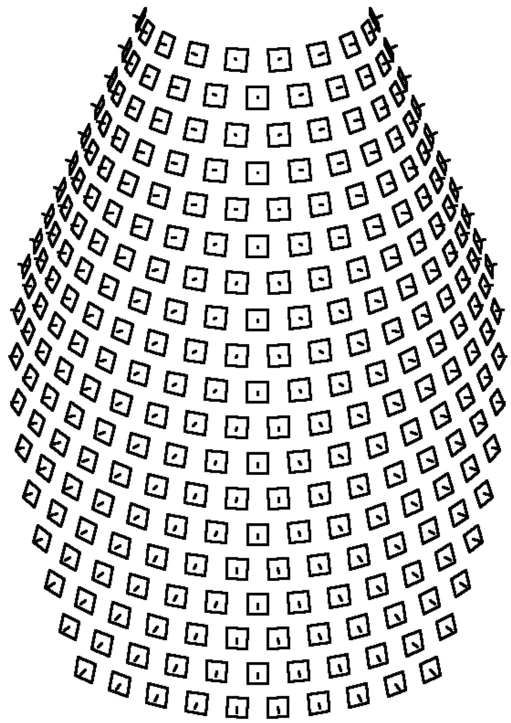

Figure 48 271 element Hexagonal Array Projected on 30° Cone

Project elements on a Cylinder: "pr, co" command

The same move command can roll an array on a cylinder

```
C:\Arrays>xadef
Enter input 0 keyboard, 1 file 0
Enter File Name cy271a.arr
Enter units: 1 in., 2 ft, 3 cm, 4 m 3
New File? y
Enter Frequency (GHz) 1
xadef: ad,he,271
Enter Ampl (dB), Phase 0,0
Enter Rotation of Hex 0
Enter Element Spacing cm  18
move: pr,co
Projection of Array from X-Y plane to Cone
Is the array orientated so that the X-axis will be around
the cone and Y axis along the cone axis? y
Enter Radius of Cone at Center of Array  cm  200
Enter Cone Angle (0 for cylinder) 0
Enter Rotation from X-axis of Center of array on Cone 0
Enter Orientation of Cone Axis: Theta, Phi 0,0
move: ex
Enter Final New Z axis Rotation of Antennas (array) 0
xadef: ex
Another file? n
Stop - Program terminated.
```

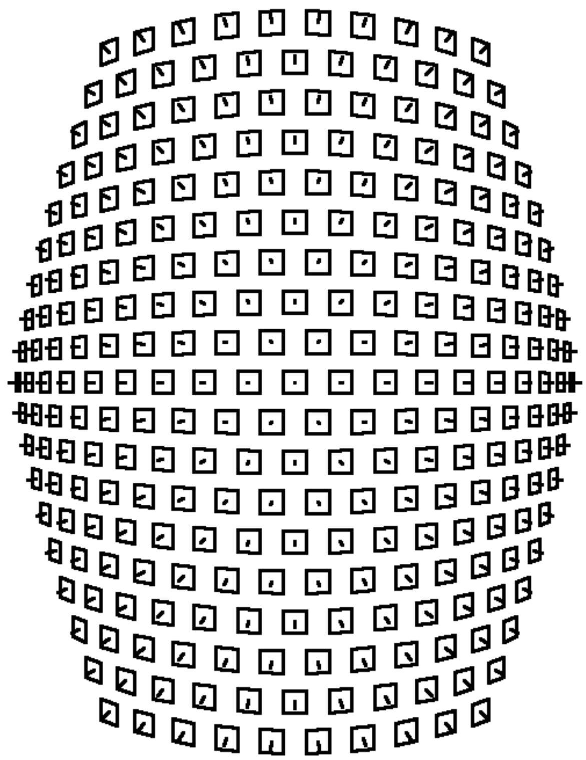

Figure 49 271 element Hexagonal Array Projected on a Cylinder

The maximum point lies on the +*x*-axis. It can be rotated around the *y*-axis to point the array in the *z* direction and translated along the *z*-axis to the origin.

The hexagonal array rolled on a cylinder (Figure 49) produces a distorted looking diagram. A rectangular array rolled on the cylinder produces a better image of the cylinder geometry.

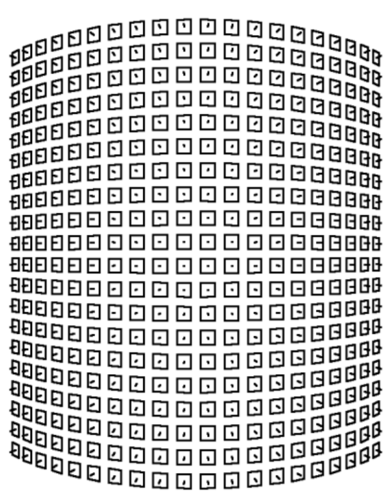

Figure 50 400 element Rectangular Array rolled on Cylinder

Un-phrased to form a beam the hexagonal array produces a smooth beam while the rectangular array produces a pattern with ripple (Figure 51).

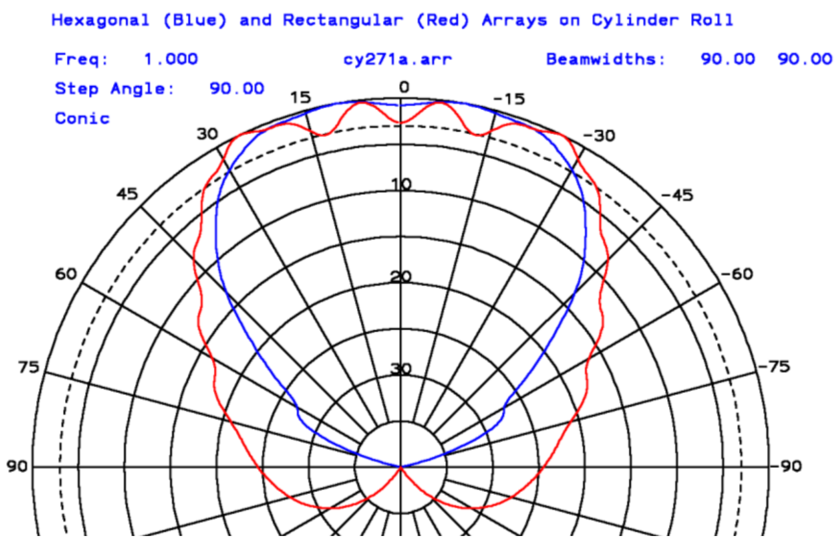

Figure 51 90° Roll Plane Pattern of un-phased Hexagonal (Blue) and Rectangular (Red) on Rolled on Cylinder

When we phase the array elements on the two arrays to form a beam at (θ, ϕ) = (90°, 0°), both arrays produce approximately the same beam. The hexagonal array on the cylinder produces lower sidelobe because of the taper of elements. Both arrays have equal amplitude elements.

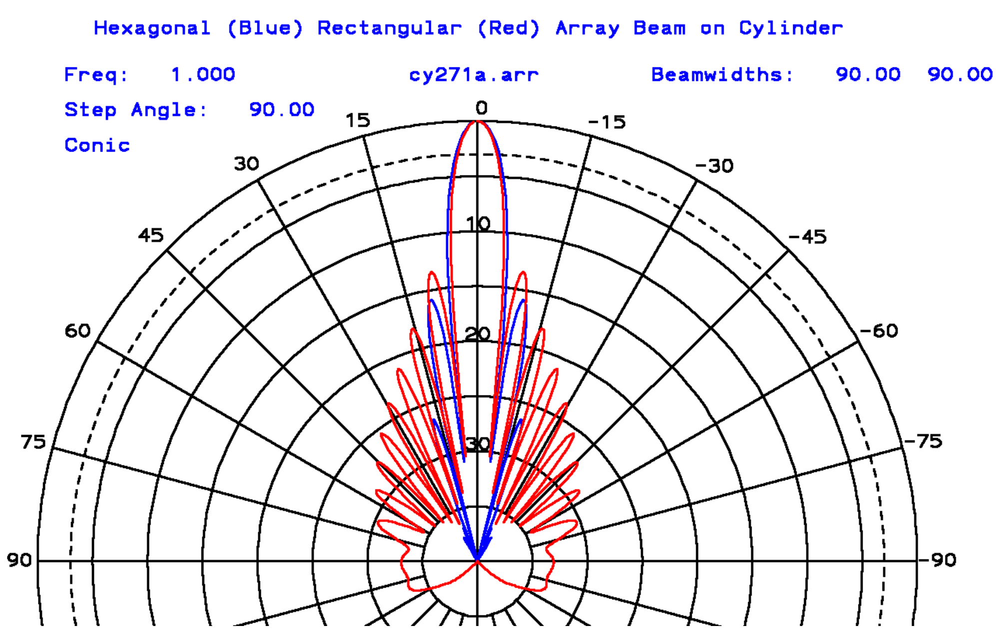

Figure 52 90° Roll Plane Pattern of Phased (90°, 0°) Hexagonal (Blue) and Rectangular (Red) on Rolled on Cylinder

Array Rolled Around Cylindrical Helix

This is another case of a flat array formed in the *x-y* plane rolled along a cylindrical helical path. We can generate the same cylindrical array using a 20 x 20 rectangular array using the “pr, he” move command.

```
C:\Arrays>xadef
Enter input 0 keyboard, 1 file 0
Enter File Name sq400h.arr
Enter units: 1 in., 2 ft, 3 cm, 4 m 3
New File? y
Enter Frequency (GHz) 1
xadef: ad,re,400
Enter Ampl (dB), Phase 0,0
Enter Number of Elements along X-axis 20
Enter initial spacings in X,Y axes cm  15,15
Array along X-axis
Enter Axis Spacing: 1 Uniform, 2 Bratkovic  3 Geometric, 4 Uneven Taylor 1
Enter 1) Uniform Amplitude Distribution
      2) Area Sampling of Taylor Distribution
      3) Point Sampling of Taylor Distribution
      4) Zero Sampled Taylor Distribution
      5) Point Sampled Bayliss Distribution
      6) Zero Sampled Bayliss Distribution
      7) Chebyshev array 1
Enter Quadratic Phase Factor, S 0
Array along Y-axis
Enter Axis Spacing: 1 Uniform, 2 Bratkovic  3 Geometric, 4 Uneven Taylor 1
Enter 1) Uniform Amplitude Distribution
      2) Area Sampling of Taylor Distribution
      3) Point Sampling of Taylor Distribution
      4) Zero Sampled Taylor Distribution
      5) Point Sampled Bayliss Distribution
      6) Zero Sampled Bayliss Distribution
      7) Chebyshev array 1
Enter Quadratic Phase Factor, S 0
move: pr,he
Projection of Array from X-Y plane to Helix Cylindrical Helix
Enter Radius of Cylinder cm  200
Enter Helix Pitch Angle of X-axis 0
Pointing of Element w.r.t. Axis (90 outward) 90
Enter Z-axis rotation of element w.r.t. Helix 0
Enter Rotation of Center of array around cylinder 0
Enter Orientation of Cylinder Axis: Theta, Phi 0,0
Enter Initial Z axis position on Cylinder cm  0
move: ex
Enter Final New Z axis Rotation of Antennas (array) 0
xadef: ex
Another file? n
```

The project on helix adds a few options, but the example above generated the same array shown in Figure 50.

```
C:\Arrays>xadef
Enter input 0 keyboard, 1 file 0
Enter File Name hep360.arr
Enter units: 1 in., 2 ft, 3 cm, 4 m 3
New File? y
Enter Frequency (GHz) 1
xadef: ad,re,360
Enter Ampl (dB), Phase 0,0
Enter Number of Elements along X-axis 60
Enter initial spacings in X,Y axes cm  15,15
Array along X-axis
Enter Axis Spacing: 1 Uniform, 2 Bratkovic  3 Geometric, 4 Uneven Taylor 1
Enter 1) Uniform Amplitude Distribution
      2) Area Sampling of Taylor Distribution
      3) Point Sampling of Taylor Distribution
      4) Zero Sampled Taylor Distribution
      5) Point Sampled Bayliss Distribution
      6) Zero Sampled Bayliss Distribution
      7) Chebyshev array 1
Enter Quadratic Phase Factor, S 0
Array along Y-axis
Enter Axis Spacing: 1 Uniform, 2 Bratkovic  3 Geometric, 4 Uneven Taylor 1
Enter 1) Uniform Amplitude Distribution
      2) Area Sampling of Taylor Distribution
      3) Point Sampling of Taylor Distribution
      4) Zero Sampled Taylor Distribution
      5) Point Sampled Bayliss Distribution
      6) Zero Sampled Bayliss Distribution
      7) Chebyshev array 1
Enter Quadratic Phase Factor, S 0
move: pr,he
Projection of Array from X-Y plane to Helix Cylindrical Helix
Enter Radius of Cylinder cm  142.92
Enter Helix Pitch Angle of X-axis 5.72
Pointing of Element w.r.t. Axis (90 outward) 90
Enter Z-axis rotation of element w.r.t. Helix 0
Enter Rotation of Center of array around cylinder 0
Enter Orientation of Cylinder Axis: Theta, Phi 0,0
Enter Initial Z axis position on Cylinder cm  0
move: ex
Enter Final New Z axis Rotation of Antennas (array) 0
xadef: ex
Another file? n
Stop - Program terminated.
```

Above a 60 x 6 element rectangular array has been rolled on to a helix with a 5.72° pitch angle. This is not a practical array, but shows in Figure 53 how an array can be tilted by the pitch angle and rolled on a helical path. XADEF move command has the command “pr, ep” which rolls the planar array along the path of an elliptical cross section helix. Its inputs are similar to the circular helix.

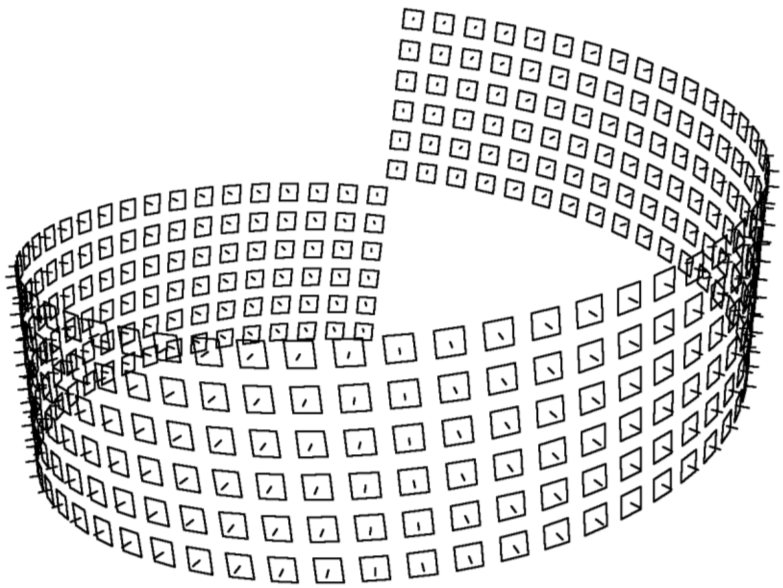

Figure 53 60 x 6 Rectangular Array with a Pitch rolled on Circular Helix showing front and rear elements

The material presented above illustrates the basic arrays which can be formed using XADEF. The move commands contain various translation and rotations not demonstrated. These allow the positioning and orientation of arrays. Of course, a single array definition file can contain combinations since the program cycles to the point where additional arrays can be added. Each array grouping needs to be orientated individually using the move commands.

XADEF inputs can be listed in a text file and used to generate an array and serve as a record or the array configuration. These can be tabulated in a text file as inputs are entered in a manual run. By annotating the inputs, this file can be duplicated and edited to generate similar arrays.

XADEF array file structure

The array geometry and excitation for each element is stored in a direct-access binary file whose RECL = 68 bytes. All records are written as single precision integers: IBUF(34)

OPEN(55,FILE=NAME,ACCESS='DIRECT',STATUS='OLD',RECL=68)

The integers in IBUF(34) are equated using EQUIVALENCE specifications to integers: IBUF(i)

The first record has IBUF(1) as the number of array elements, X(2) as frequency (GHz)

Array element(*I*) is stored in random direct-access record: *I+1.*

Element position: X(3): EQUIVALENCE (X(1),IBUF(1)) which means IBUF(1) – IBUF(6)

Element complex voltage excitation: E: EQUIVALENCE (E,IBUF(7)) which means IBUF(7) – IBUF(10)

Euler angle rotations: 1) about *Z*-axis (CCW), 2) about new *Y*-axis, 3) about new *Z*-axis: EU(3):

EQUIVALENCE (EU(1),IBUF(11)) which means IBUF(11) – IBUF(16)

Rotation matrix from Euler angles: AR(3,3): EQUIVALENCE (AR(1,1),IBUF(17)) which means IBUF(17) – IBUF(34) with array numbering in FORTRAN order AR(1,1), AR(2,1), AR(3,1), AR(1,2), AR(2,2), AR(3,2), AR(1,3), AR(2,3), AR(3,3)

Element positions are multiplied by 2π/λ where λ is determined from the specification frequency located in record one, positions are unit-less.