Array Gain

This notebook computes and displays the H-plane gain for a vertical stack of horizontally-polarized antennas over perfect flat ground, relative to a single antenna in free space. The algorithm is straight-forward pattern multiplication, and thus does not account for interaction between the antennas.

The ArrayGain function has 3 arguments, **a**, **h**, and **alpha**. The first two arguments are lists of antenna drive voltages and antenna heights above ground, respectively. The lists must have the same length. The sum of the magnitudes of the drive voltages must equal the list length to achieve proper gain normalization. The unit of height is wavelength. **alpha** is the vertical angle measured from the horizontal plane; typically this is the independent variable in creating pattern plots.

ArrayGain(a_, h_, alpha_) :=

$$20 \log_{10}\left(\left|\text{Plus} @@ \left(\text{sgn}(a) \sqrt{|a|}\, e^{i\, h \sin(\text{alpha})}\right) - \text{Plus} @@ \left(\text{sgn}(a) \sqrt{|a|}\, e^{-i\, h \sin(\text{alpha})}\right)\right|\right) - 10 \log_{10}(\text{Length}[a])$$

The following cells create a grid for the polar pattern plots. The outermost gain level is set by the maxGain parameter. With this display, it is easy to quickly assess the relative merits of different phasing choices, which is rather more difficult with the ARRL normalized non-linear radial scaling scheme for antenna pattern plots.

```
maxGain = 11;
circles = Table[x, {x, maxGain}];
gainGrid = PolarPlot[circles, {x, 0, Pi / 2}, PlotStyle → {GrayLevel[.5, .5]}];

radialGrid = ListPlot[{
    {{0, 0}, maxGain {Cos[Pi / 18], Sin[Pi / 18]}}, {{0, 0}, maxGain {Cos[Pi / 9], Sin[Pi / 9]}},
    {{0, 0}, maxGain {Cos[Pi / 6], Sin[Pi / 6]}}, {{0, 0}, maxGain {Cos[2 Pi / 9], Sin[2 Pi / 9]}},
    {{0, 0}, maxGain {Cos[5 Pi / 18], Sin[5 Pi / 18]}},
    {{0, 0}, maxGain {Cos[Pi / 3], Sin[Pi / 3]}}, {{0, 0}, maxGain {Cos[7 Pi / 18], Sin[7 Pi / 18]}},
    {{0, 0}, maxGain {Cos[4 Pi / 9], Sin[4 Pi / 9]}}},
   PlotRange → {0, maxGain}, PlotStyle → {GrayLevel[.5, .5]}, Joined → True];

grid = Show[gainGrid, radialGrid];
```

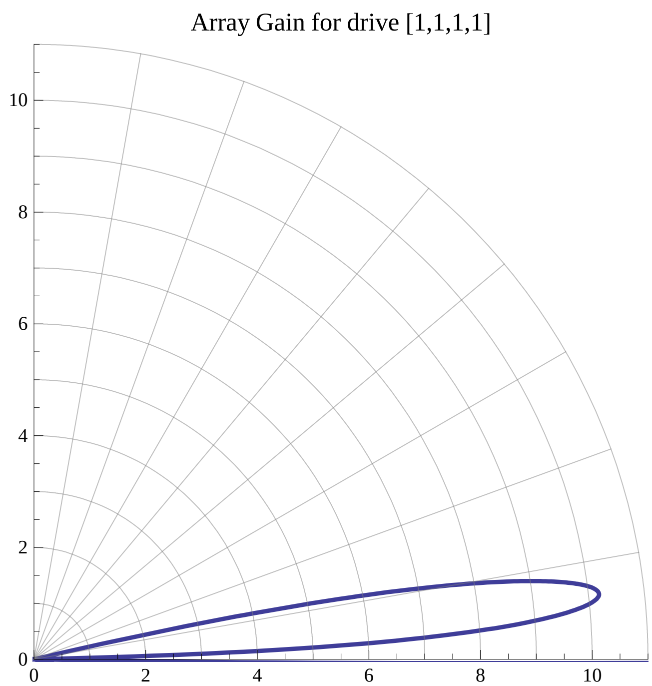
Example

Set up the height list for the KC1XX 4-stack of 20m yagis. The list is bottom-up, but order doesn't matter as long as you know which slot is which antenna when you fuss with the drives list.

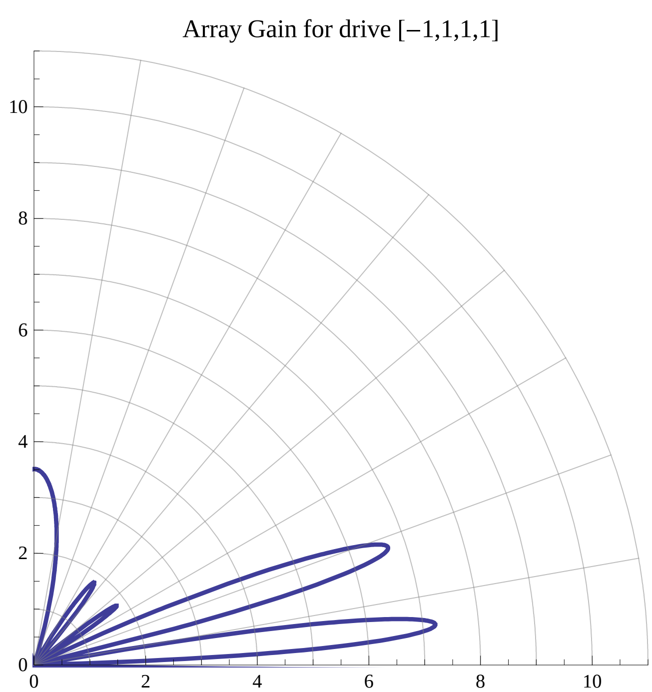
```
spacing = 50.0;
h0 = 50;
h = {h0, h0 + spacing, h0 + 2.0 * spacing, h0 + 3.0 * spacing};
height = h * 2 * Pi * 14.15 * 0.3048 / 299.79;
```

Each of the following cells sets a drive voltages lists and plots the resulting array gain. Decades ago, OH8OS found that the optimum (maximum gain) drive voltages for any vertical angle have phase angles quite close to either 0° or 180°. So, it makes sense to examine all combinations of equal power splits with these two phase angles. Thus, each drive voltage is either 1 or -1. There are 16 combinations, but, due to the symmetry of the antenna system and its perfect-ground image, only eight combinations have distinct patterns. They are enumerated here by keeping the top drive voltage phase always at 0° and evalating all combinations of phasing for the bottom three antennas.

```
drive = {1, 1, 1, 1};
low = PolarPlot[ArrayGain[drive, height, x], {x, 0, Pi}, PlotRange → {0, maxGain},
   PlotStyle → AbsoluteThickness[2], PlotLabel → "Array Gain for drive [1,1,1,1]"];
Show[
 low,
 grid]
```

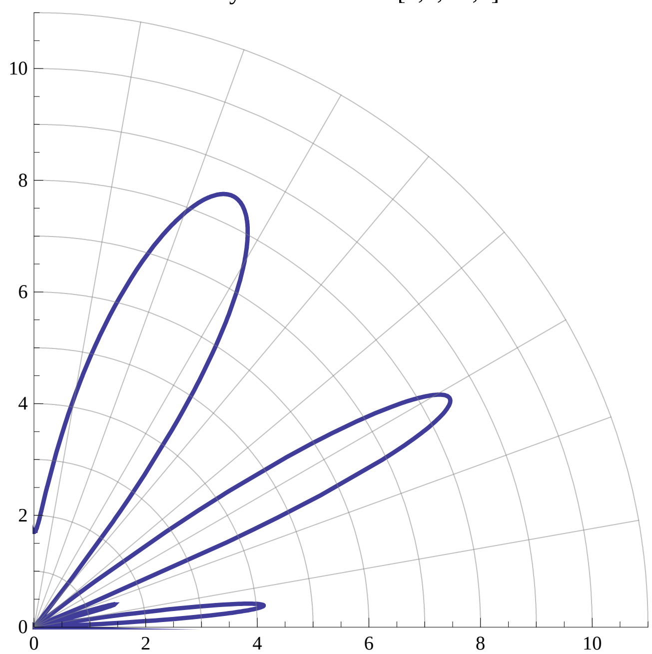



```
drive = {-1, 1, 1, 1};
Show[PolarPlot[ArrayGain[drive, height, x], {x, 0, Pi}, PlotRange → {0, maxGain},
  PlotStyle → AbsoluteThickness[2], PlotLabel → "Array Gain for drive [-1,1,1,1]"], grid]
```



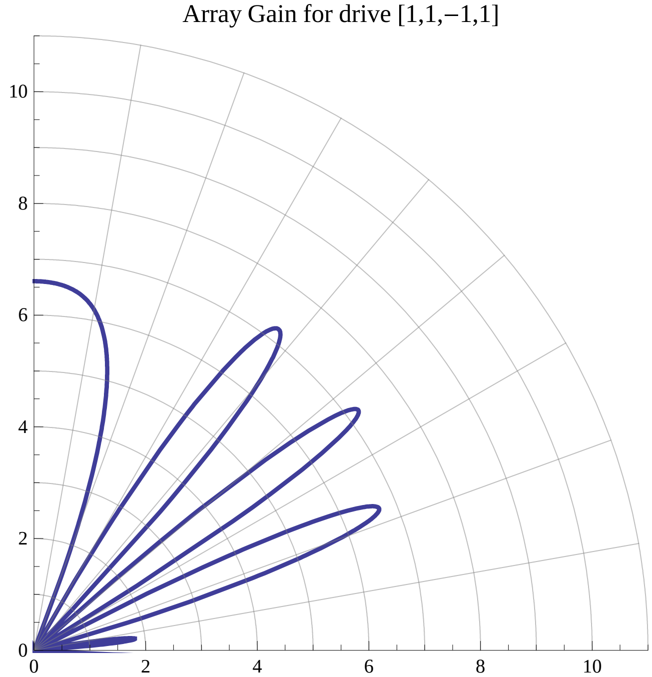
```
drive = {1, -1, 1, 1};
Show[PolarPlot[ArrayGain[drive, height, x], {x, 0, Pi}, PlotRange → {0, maxGain},
  PlotStyle → AbsoluteThickness[2], PlotLabel → "Array Gain for drive [1,1,-1,1]"], grid]
```

Array Gain for drive [1,1,−1,1]



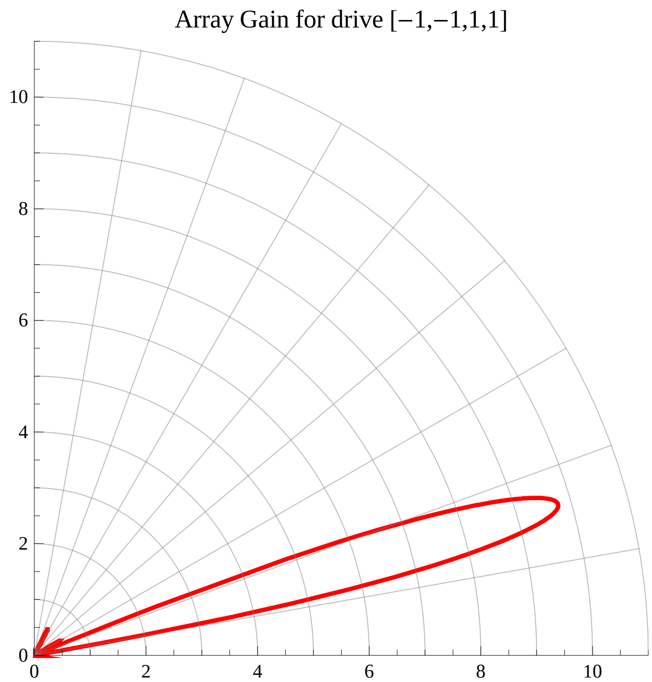

```
drive = {1, 1, -1, 1};
Show[PolarPlot[ArrayGain[drive, height, x], {x, 0, Pi}, PlotRange → {0, maxGain},
  PlotStyle → AbsoluteThickness[2], PlotLabel → "Array Gain for drive [1,1,-1,1]"], grid]
```

Array Gain for drive [1,1,−1,1]



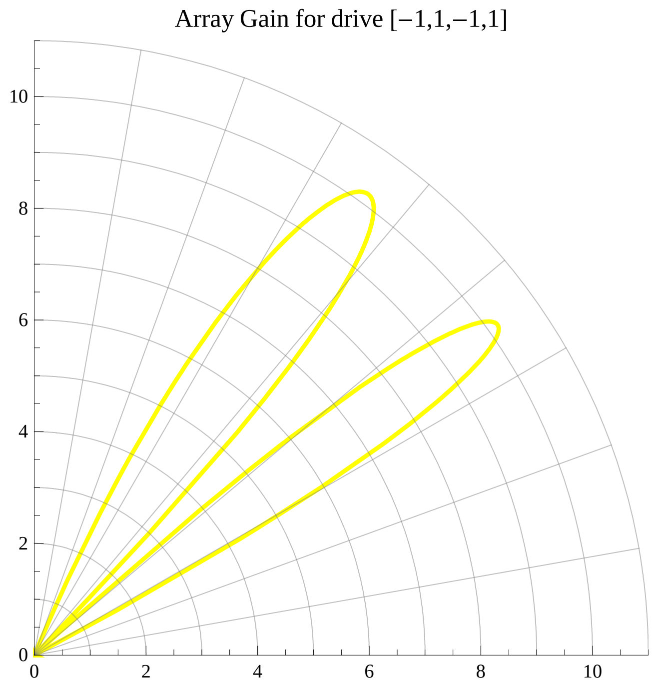
In[354]:=
```
drive = {-1, -1, 1, 1};
medium = PolarPlot[ArrayGain[drive, height, x], {x, 0, Pi},
   PlotRange → {0, maxGain}, PlotStyle → {RGBColor[1, 0, 0], AbsoluteThickness[2]},
   PlotLabel → "Array Gain for drive [-1,-1,1,1]"];
Show[
 medium,
 grid]
```

Out[356]=



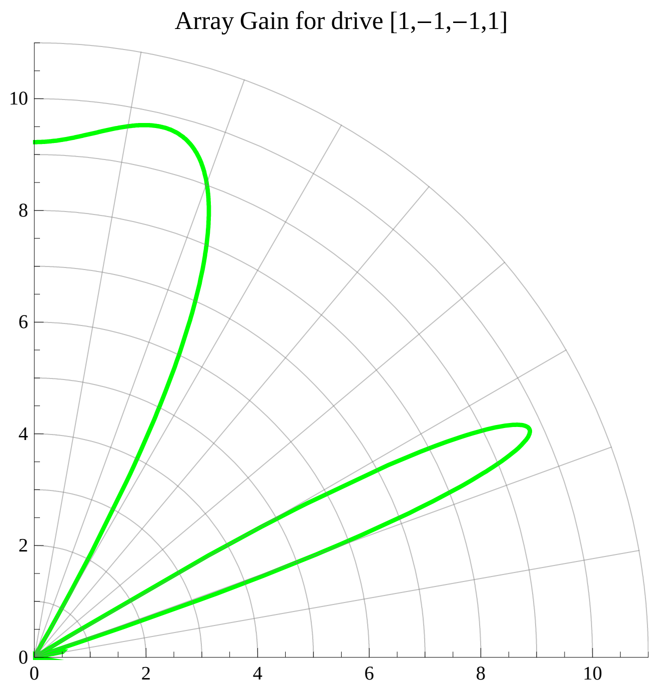
In[357]:=
```
drive = {-1, 1, -1, 1};
nosebleed = PolarPlot[ArrayGain[drive, height, x], {x, 0, Pi},
   PlotRange → {0, maxGain}, PlotStyle → {RGBColor[1, 1, 0], AbsoluteThickness[2]},
   PlotLabel → "Array Gain for drive [-1,1,-1,1]"];
Show[nosebleed, grid]
```

Out[359]=



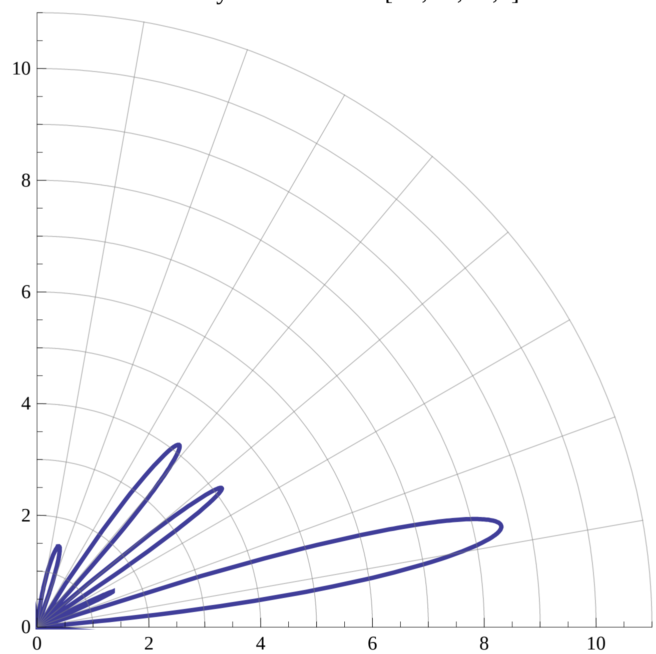
In[360]:=
```
drive = {1, -1, -1, 1};
high = PolarPlot[ArrayGain[drive, height, x], {x, 0, Pi},
   PlotRange → {0, maxGain}, PlotStyle → {RGBColor[0, 1, 0], AbsoluteThickness[2]},
   PlotLabel → "Array Gain for drive [1,-1,-1,1]"];
Show[
 high,
 grid]
```

Out[362]=



```
drive = {-1, -1, -1, 1};
Show[PolarPlot[ArrayGain[drive, height, x], {x, 0, Pi}, PlotRange → {0, maxGain},
  PlotStyle → AbsoluteThickness[2], PlotLabel → "Array Gain for drive [-1,-1,-1,1]"], grid]
```

Array Gain for drive [−1,−1,−1,1]



KC1XX uses three transfer switches to insert 180 ° delay lines in the bottom three feed lines for steering the vertical angle. The four choices available in the control box are Low (blue) = [1, 1, 1, 1] , Medium (red) = [-1, -1, 1, 1], High (yellow) = [1, -1, -1, 1], and Nosebleed (high) = [-1, 1, -1, 1]. These four combinations are plotted together below. The rather significant notch between the two lowest lobe is right at the typical arrival angle for signals from Europe when the band is open, but the foreground slope seen by the array pulls the second (Medium) lobe down into a more useful range. We also find the High choice sometimes useful in the afternoon.

In[363]:= `Show[low, medium, high, nosebleed, grid]`

Array Gain for drive [1,1,1,1]

Out[363]=