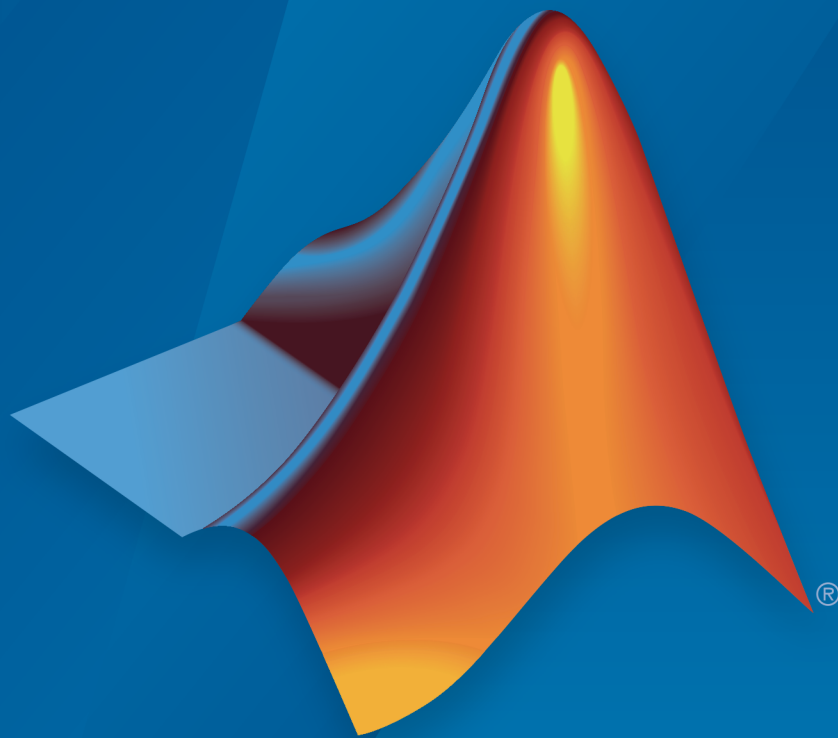


# Antenna Toolbox™

## Getting Started Guide



# MATLAB®

R2017a



# How to Contact MathWorks



Latest news: [www.mathworks.com](http://www.mathworks.com)  
Sales and services: [www.mathworks.com/sales\\_and\\_services](http://www.mathworks.com/sales_and_services)  
User community: [www.mathworks.com/matlabcentral](http://www.mathworks.com/matlabcentral)  
Technical support: [www.mathworks.com/support/contact\\_us](http://www.mathworks.com/support/contact_us)



Phone: 508-647-7000



The MathWorks, Inc.  
3 Apple Hill Drive  
Natick, MA 01760-2098

## *Antenna Toolbox™ Getting Started Guide*

© COPYRIGHT 2015–2017 by The MathWorks, Inc.

The software described in this document is furnished under a license agreement. The software may be used or copied only under the terms of the license agreement. No part of this manual may be photocopied or reproduced in any form without prior written consent from The MathWorks, Inc.

**FEDERAL ACQUISITION:** This provision applies to all acquisitions of the Program and Documentation by, for, or through the federal government of the United States. By accepting delivery of the Program or Documentation, the government hereby agrees that this software or documentation qualifies as commercial computer software or commercial computer software documentation as such terms are used or defined in FAR 12.212, DFARS Part 227.72, and DFARS 252.227-7014. Accordingly, the terms and conditions of this Agreement and only those rights specified in this Agreement, shall pertain to and govern the use, modification, reproduction, release, performance, display, and disclosure of the Program and Documentation by the federal government (or other entity acquiring for or through the federal government) and shall supersede any conflicting contractual terms or conditions. If this License fails to meet the government's needs or is inconsistent in any respect with federal procurement law, the government agrees to return the Program and Documentation, unused, to The MathWorks, Inc.

### **Trademarks**

MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See [www.mathworks.com/trademarks](http://www.mathworks.com/trademarks) for a list of additional trademarks. Other product or brand names may be trademarks or registered trademarks of their respective holders.

### **Patents**

MathWorks products are protected by one or more U.S. patents. Please see [www.mathworks.com/patents](http://www.mathworks.com/patents) for more information.

### **Revision History**

March 2015	Online only	New for Version 1.0 (R2015a)
September 2015	Online only	Revised for Version 1.1 (R2015b)
March 2016	Online only	Revised for Version 2.0 (R2016a)
September 2016	Online only	Revised for Version 2.1 (R2016b)
March 2017	Online only	Revised for Version 2.2 (R2017a)

## Introduction to Antenna Toolbox

<b>Antenna Toolbox Product Description</b> .....	<b>1-2</b>
Key Features .....	1-2
<b>Antenna Modeling and Analysis</b> .....	<b>1-3</b>
<b>Antenna Toolbox Library Elements</b> .....	<b>1-21</b>
Dipole Antennas .....	1-21
Monopole Antennas .....	1-24
Loop Antennas .....	1-25
Spiral Antennas .....	1-25
Patch Antennas .....	1-26
Other Antennas .....	1-27
Arrays .....	1-29
<b>Antenna Classification</b> .....	<b>1-32</b>
Radiation Pattern .....	1-32
Antenna Feeding Mechanism .....	1-35
<b>Antenna Toolbox Coordinate System</b> .....	<b>1-37</b>
Rectangular Coordinate System .....	1-37
Spherical Coordinate System .....	1-40
Conversion Between Rectangular and Spherical Coordinates .....	1-44
<b>Antenna Radiation Patterns</b> .....	<b>1-46</b>
<b>Antenna Toolbox Limitations</b> .....	<b>1-76</b>
Antenna Library .....	1-76
Array Library .....	1-76
<b>Interact with Polar Plot</b> .....	<b>1-77</b>
<b>Antenna Design and Analysis Using Antenna Designer App</b> .....	<b>1-87</b>

<b>Design Variations On Microstrip Patch Antenna Using PCB Stack .....</b>	<b>1-93</b>
--	-------------

## **Introduction to Arrays**

# **2**

<b>Array Modeling and Analysis .....</b>	<b>2-2</b>
--	------------

# Introduction to Antenna Toolbox

---

- “Antenna Toolbox Product Description” on page 1-2
- “Antenna Modeling and Analysis” on page 1-3
- “Antenna Toolbox Library Elements” on page 1-21
- “Antenna Classification” on page 1-32
- “Antenna Toolbox Coordinate System” on page 1-37
- “Antenna Radiation Patterns” on page 1-46
- “Antenna Toolbox Limitations” on page 1-76
- “Interact with Polar Plot” on page 1-77
- “Antenna Design and Analysis Using Antenna Designer App” on page 1-87
- “Design Variations On Microstrip Patch Antenna Using PCB Stack” on page 1-93

## Antenna Toolbox Product Description

### Design, analyze, and visualize antenna elements and antenna arrays

Antenna Toolbox™ provides functions for the design, analysis, and visualization of antenna elements and arrays. You can design standalone antennas and build linear, rectangular, and conformal arrays of antennas using predefined elements with parameterized geometry or custom elements.

Antenna Toolbox uses the method of moments (MoM) to compute port properties such as impedance, surface properties such as current and charge distribution, and field properties such as the near-field and far-field radiation pattern. You can visualize antenna geometry and analysis results in 2D and 3D.

You can integrate antennas and arrays into wireless systems, and use impedance analysis to design matching networks. Antenna Toolbox provides radiation patterns for simulating beam forming algorithms.

### Key Features

- Rapid design and visualization of antennas using predefined or custom elements
- Design of linear, rectangular, conformal, and custom antenna arrays
- Large array analysis using the infinite array or embedded element pattern approach
- Infinite ground plane specification for analyzing antennas mounted on very large structures
- Port analysis of impedance, return loss, and S-parameters of antennas and antenna arrays
- Radiation field analysis of the pattern, E-H fields, and beam width of antennas, antenna arrays, and custom data
- Surface analysis of antenna and antenna array current, charge, and meshing

# Antenna Modeling and Analysis

This example shows how to construct, visualize and analyze the antenna elements in the Antenna Toolbox.

## Define Antenna Element Using the Antenna Library

Define a helix antenna using the helix antenna element in the Antenna Modeling and Analysis library.

```
hx = helix
```

```
hx =
```

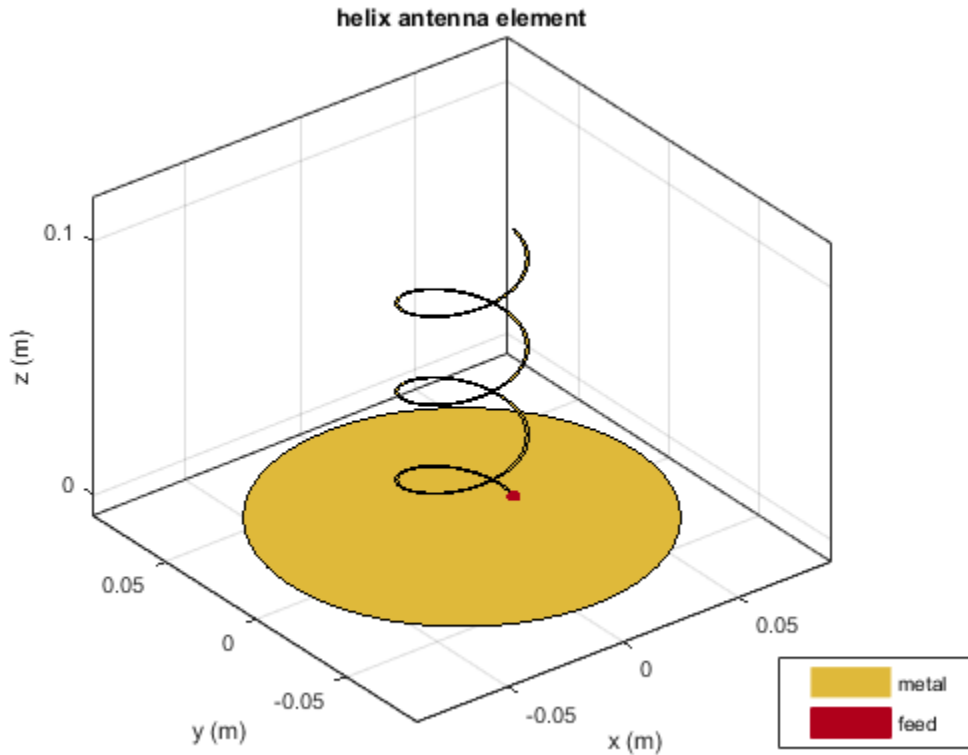
```
helix with properties:
```

```
    Radius: 0.0220
    Width: 1.0000e-03
    Turns: 3
    Spacing: 0.0350
    WindingDirection: 'CCW'
    GroundPlaneRadius: 0.0750
    Tilt: 0
    TiltAxis: [1 0 0]
    Load: [1×1 lumpedElement]
```

## Show Structure of Antenna

Use the show function to view the structure of the helix antenna. A helical antenna consists of a helical shaped conductor on a ground plane. The ground plane of the antenna is located in the X-Y plane.

```
show(hx)
```



### Modify Properties of Antenna

Modify the following properties of the helix antenna: Radius =  $28 \times 10^{-3}$ , Width =  $1.2 \times 10^{-3}$ , Number of Turns = 4. Display the properties of the antenna. View the antenna to see the change in structure.

```
hx = helix('Radius',28e-3,'Width',1.2e-3,'Turns',4)
show(hx)
```

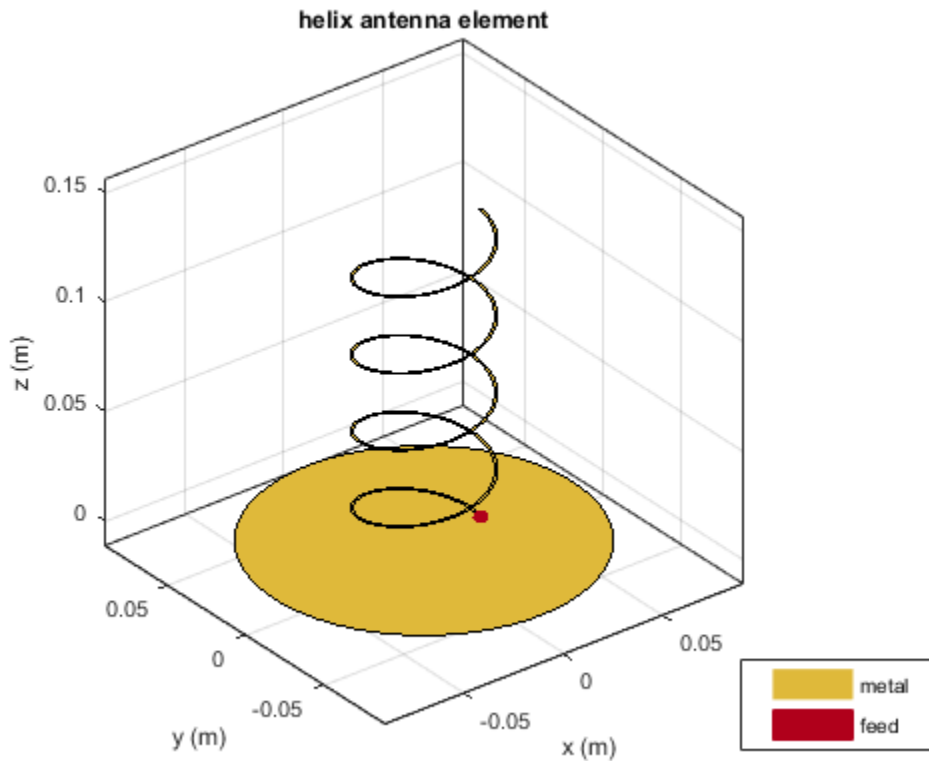
hx =

helix with properties:

Radius: 0.0280



```
Width: 0.0012
Turns: 4
Spacing: 0.0350
WindingDirection: 'CCW'
GroundPlaneRadius: 0.0750
Tilt: 0
TiltAxis: [1 0 0]
Load: [1×1 lumpedElement]
```

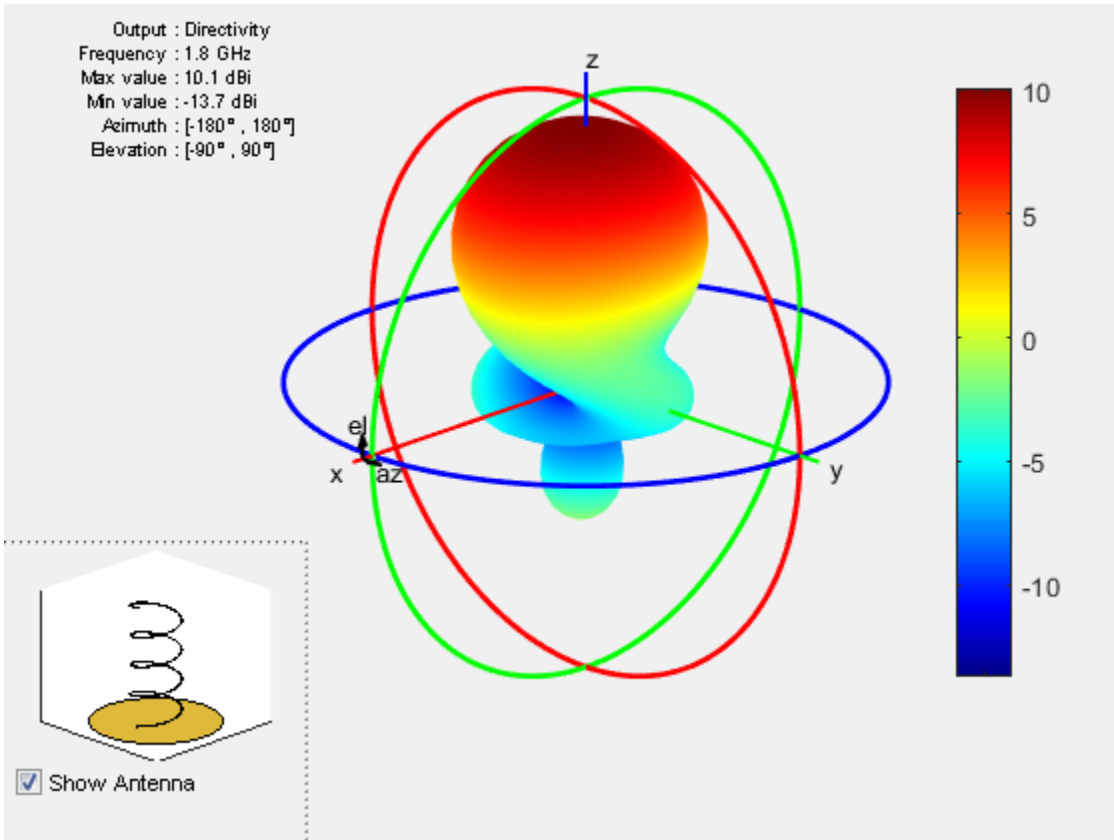


### Plot Radiation Pattern Of Antenna

Use pattern function to plot the radiation pattern of the helix antenna. The radiation pattern of an antenna is the spatial distribution of power of an antenna. The pattern

displays the directivity or gain of the antenna. By default, the pattern function plots the directivity of the antenna.

```
pattern(hx,1.8e9)
```

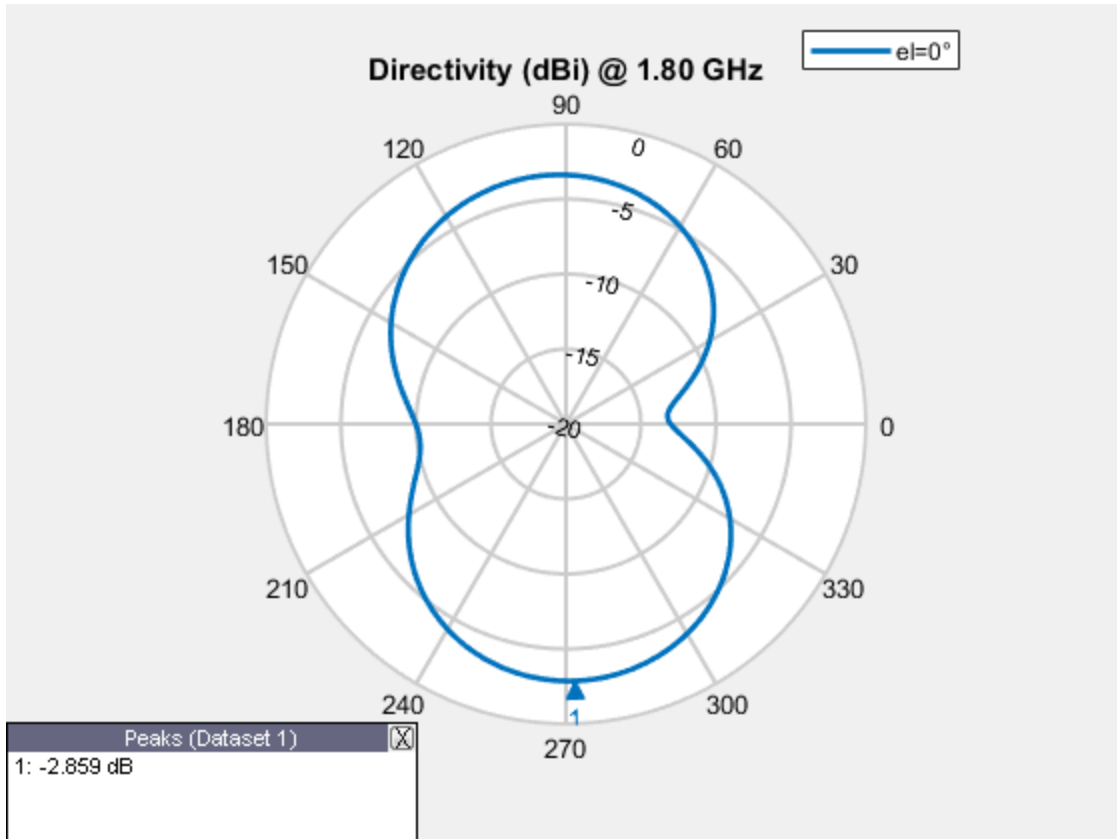


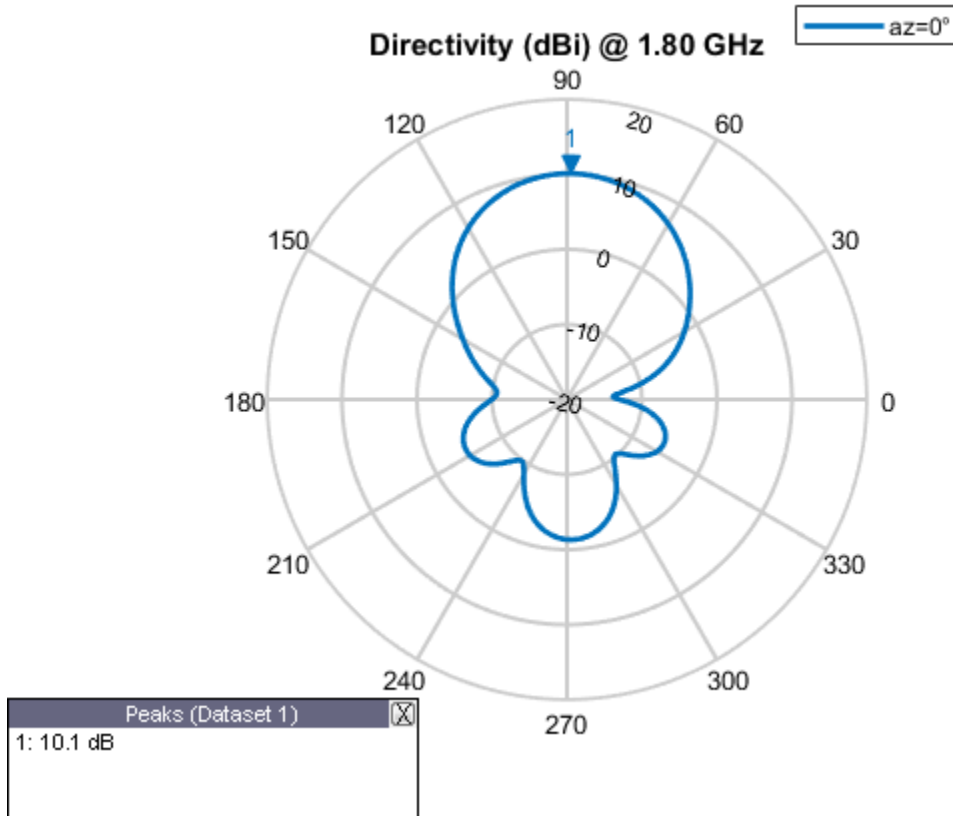
### Plot Azimuth and Elevation Pattern of Antenna

Use `patternAzimuth` and `patternElevation` functions to plot the azimuth and elevation pattern of the helix antenna. This is the 2D radiation pattern of the antenna at a specified frequency.

```
patternAzimuth(hx,1.8e9)
```

figure  
patternElevation(hx,1.8e9)





### Calculate Directivity of Antenna

Use Directivity name-value pair in the output of the pattern function to calculate the directivity of helix antenna. Directivity is the ability of an antenna to radiate power in a particular direction. It can be defined as ratio of maximum radiation intensity in the desired direction to the average radiation intensity in all other directions.

```
[Directivity] = pattern(hx,1.8e9,0,90)
```

```
Directivity =
```

```
10.0915
```

## Calculate EHfields Of Antenna

Use the EHfields function to calculate the EH fields of the helix antenna. EH fields are the x, y,z componenets of electric and magnetic fields of an antenna. These components are measured at a specific frequency and at specified points in space.

```
[E,H] = EHfields(hx,1.8e9,[0;0;1])
```

E =

```
-0.5321 - 0.5976i  
-0.9096 + 0.5331i  
-0.0041 + 0.0007i
```

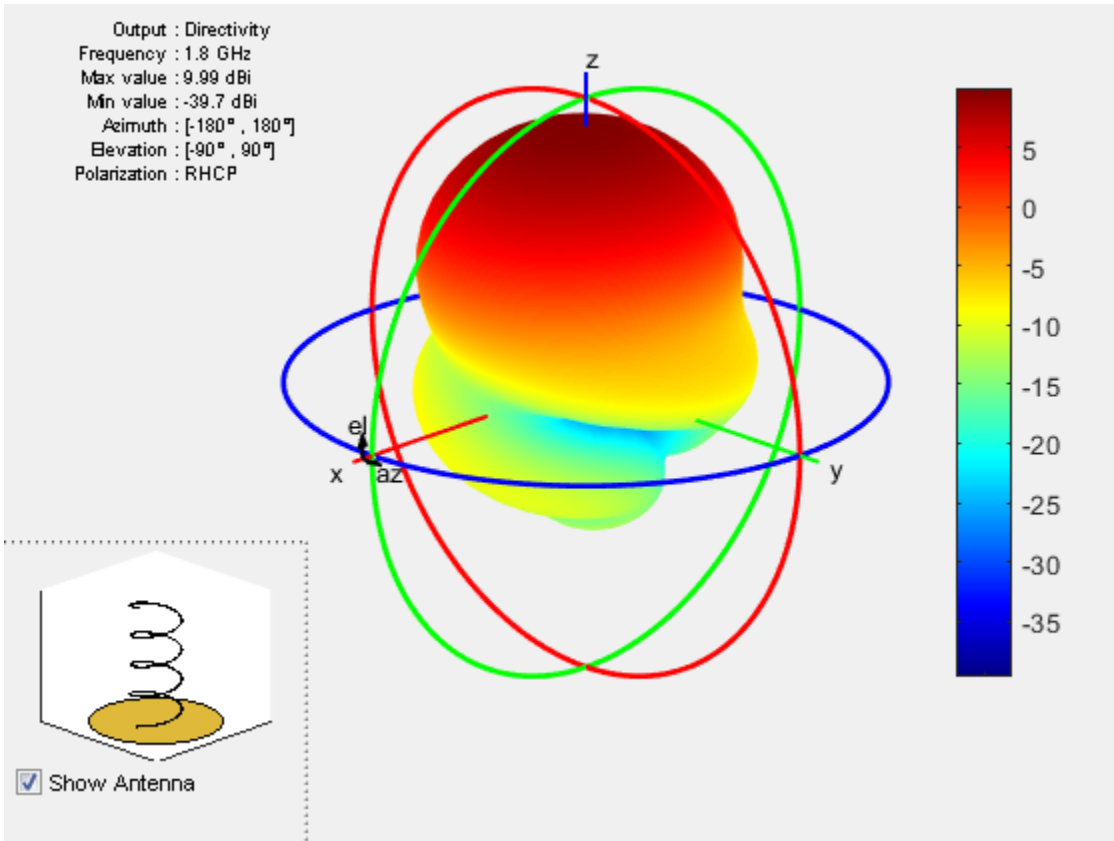
H =

```
0.0024 - 0.0014i  
-0.0014 - 0.0016i  
0.0000 - 0.0000i
```

## Plot Different Polarizations of Antenna

Use the Polarization name-value pair in the pattern function to plot the different polarization patterns of the helix antenna. Polarization is the orientation of the electric field, or E-field, of an antenna. Polarization is classified as elliptical, linear, or circular. This example shows the Right-Hand Circularly Polarized (RHCP) radiation pattern of the helix.

```
pattern(hx,1.8e9,'Polarization','RHCP')
```



### Calculate Axial Ratio Of Antenna

Use the `axialRatio` function to calculate the axial ratio of the helix antenna. Antenna axial ratio (AR) in a given direction quantifies the ratio of two orthogonal field components radiated in a circularly polarized wave. An axial ratio of infinity, implies a linearly polarized wave. The unit of measure is dB.

```
ar = axialRatio(hx,1.8e9,20,30)
```

```
ar =
```

```
25.4168
```

### Calculate Beamwidth Of Antenna

Use the beamwidth function to calculate the beamwidth of the antenna. Antenna beamwidth is the angular measure of the antenna pattern coverage. Beamwidth angle is measured in plane containing the direction of main lobe of the antenna.

```
[bw, angles] = beamwidth(hx,1.8e9,0,1:1:360)
```

```
bw =
```

```
57.0000
```

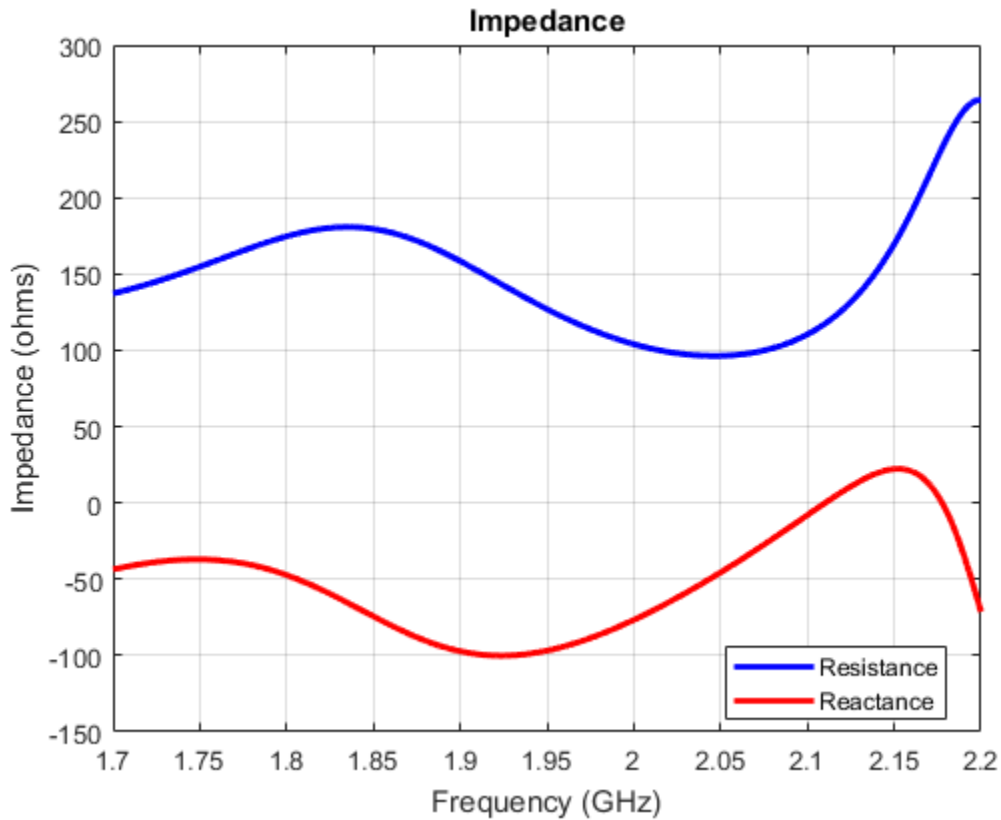
```
angles =
```

```
60 117
```

### Calculate Impedance Of Antenna

Use the impedance function to calculate and plot the input impedance of helix antenna. Input impedance is a ratio of voltage and current at the port. Antenna impedance is calculated as the ratio of the phasor voltage( which is 1V at a phase angle of 0 deg as mentioned earlier) and the phasor current at the port.

```
impedance(hx,1.7e9:1e6:2.2e9)
```



### Calculate Reflection Coefficient of Antenna

Use the `sparameters` function to calculate the S11 of the helix antenna. Antenna reflection coefficient, or `S_1_1`, describes a relative fraction of the incident RF power that is reflected back due to the impedance mismatch.

```
S = sparameters(hx, 1.7e9:1e6:2.2e9, 72)
rfplot(S)
```

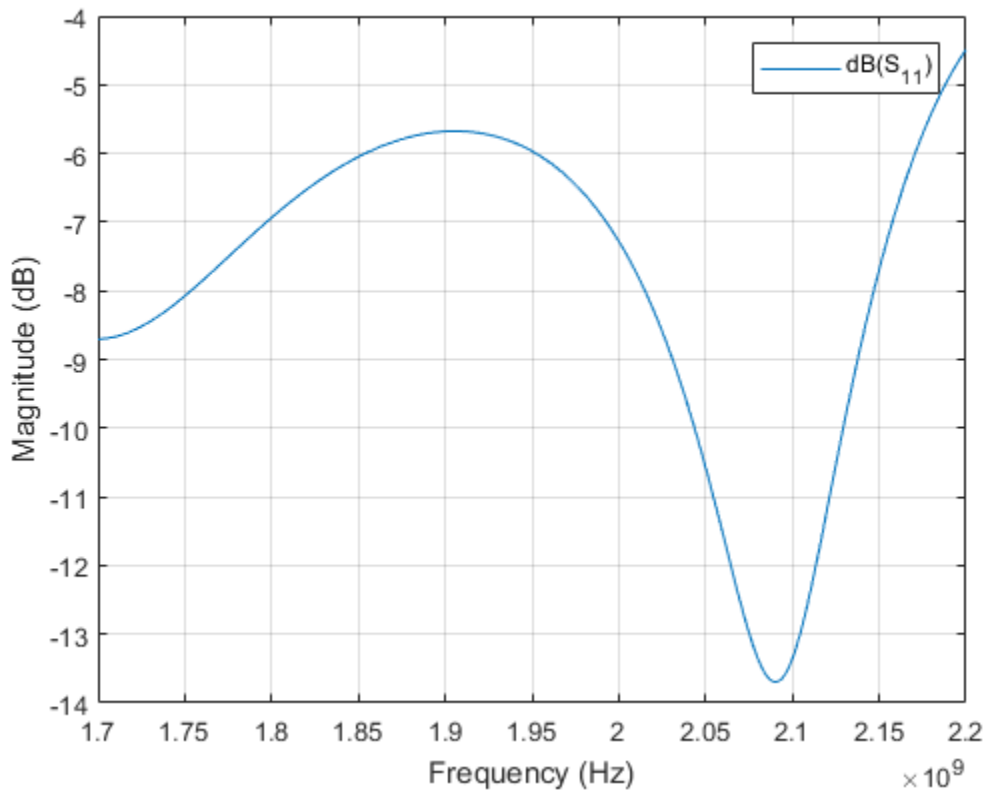
S =

```
sparameters: S-parameters object
```



```
NumPorts: 1
Frequencies: [501×1 double]
Parameters: [1×1×501 double]
Impedance: 72
```

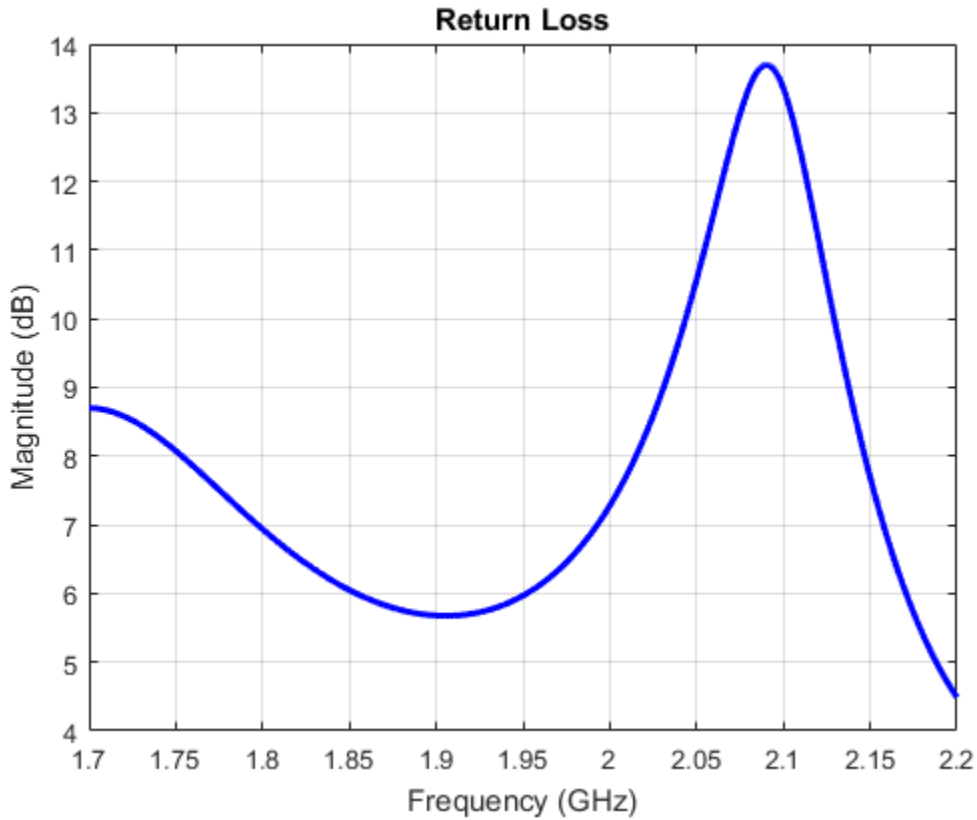
`rfparam(obj,i,j)` returns S-parameter  $S_{ij}$



### Calculate Return Loss Of Antenna

Use the `returnLoss` function to calculate and plot the return loss of the helix antenna. Antenna return loss is a measure of the effectiveness of power delivery from a transmission line to a load such as antenna. The calculations are displayed in logscale.

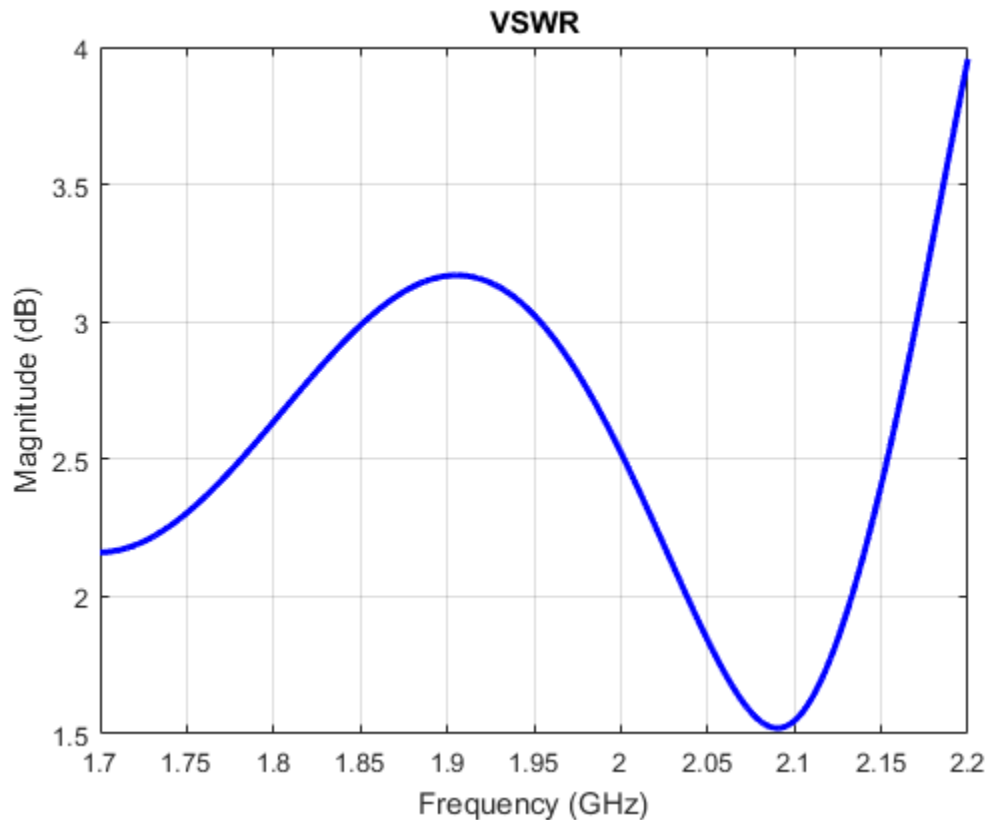
```
returnLoss(hx,1.7e9:1e6:2.2e9,72)
```



### Calculate Voltage Standing Wave Ratio (VSWR) of Antenna

Use the `vswr` function to calculate and plot the VSWR of the helix antenna. The antenna VSWR is another measure of impedance matching between transmission line and antenna.

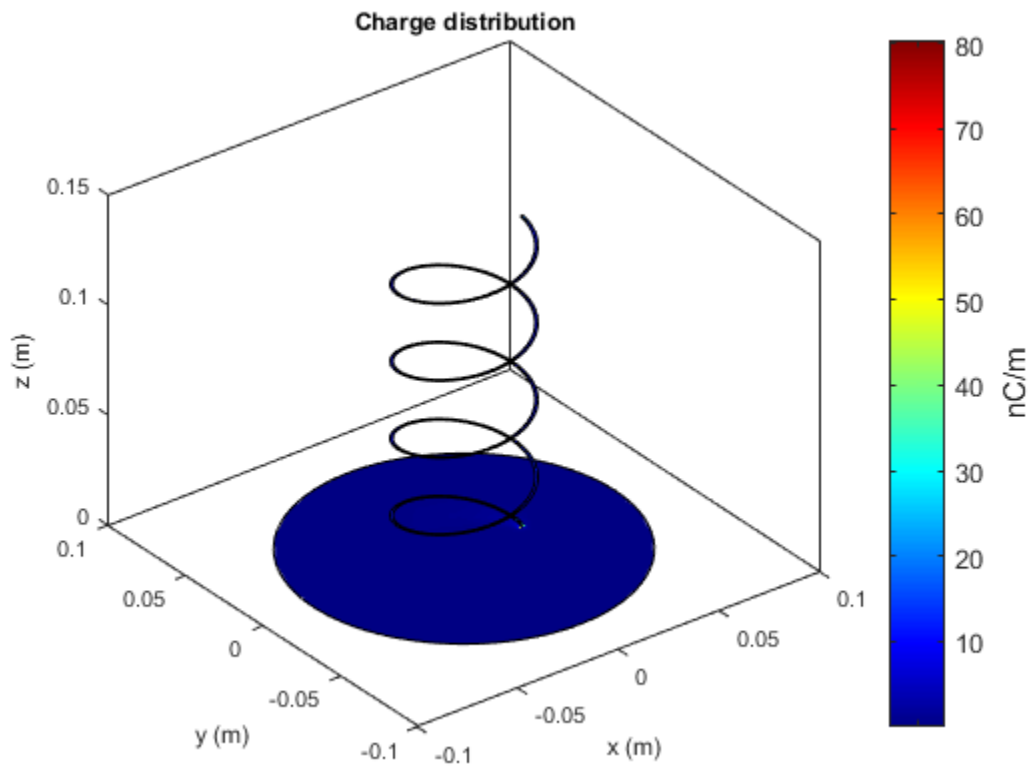
```
vswr(hx,1.7e9:1e6:2.2e9,72)
```

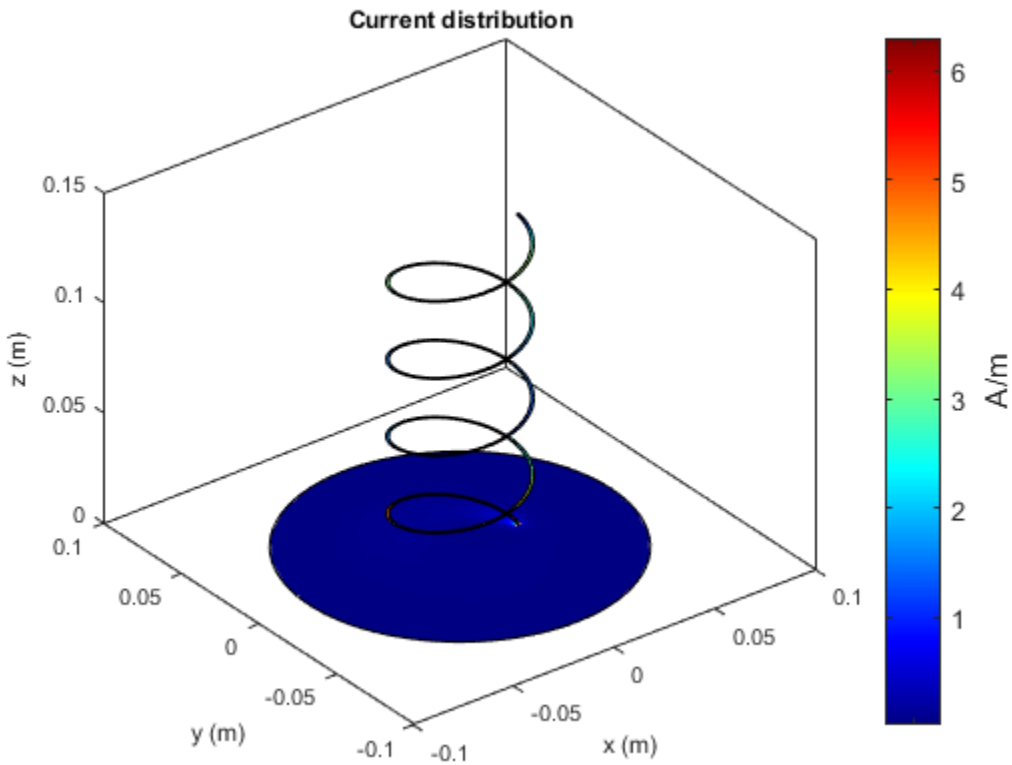


### Calculate Current and Charge Distribution Of Antenna

Use the charge function to calculate the charge distribution of the helix antenna. Charge distribution is the value of charge on the antenna surface at a specified frequency. Use the current function to calculate the current distribution of the helix antenna. Current distribution is the value of current on the antenna surface at a specified frequency.

```
charge(hx,2.01e9)
figure
current(hx,2.01e9)
```

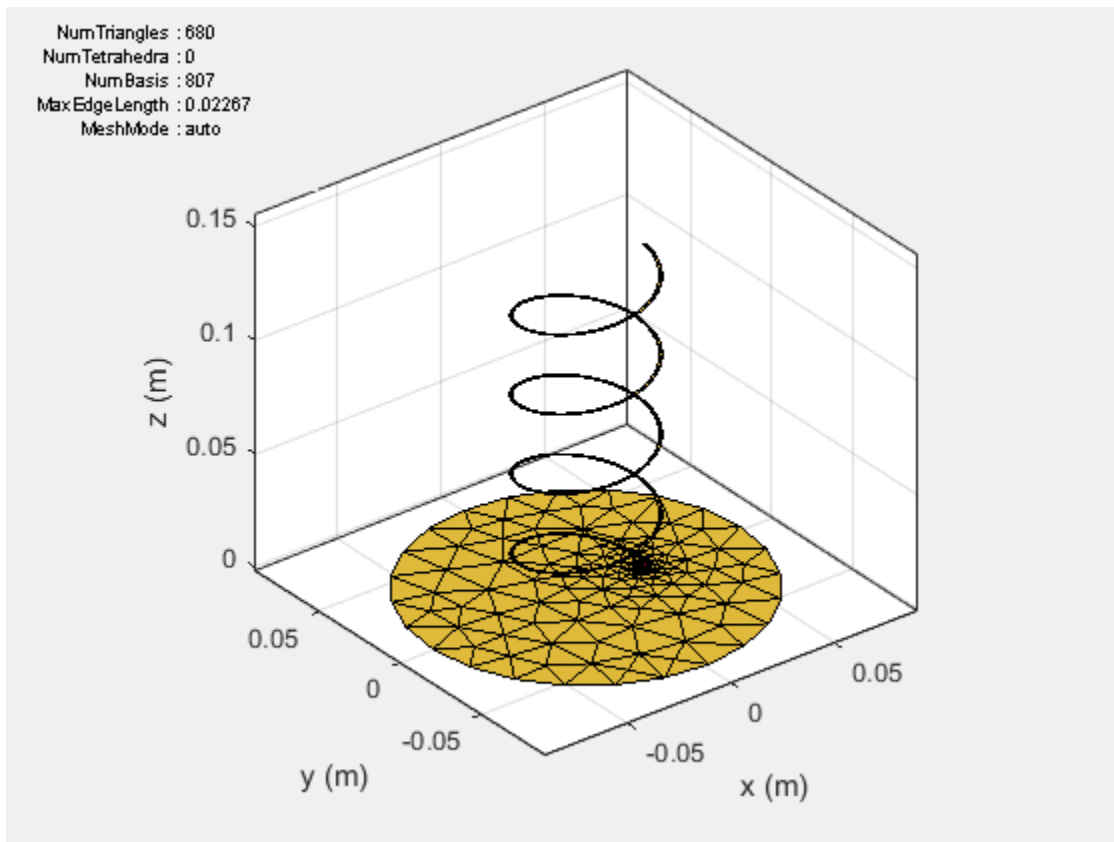




### Show Mesh of Antenna

Use the mesh function to create and show a mesh structure of the helix antenna. mesh is used to discretize antenna surface. In this process, the electromagnetic solver can process the geometry and material of the antenna. The shape of the basis or the discretizing element for subdividing the antenna surface is a triangle.

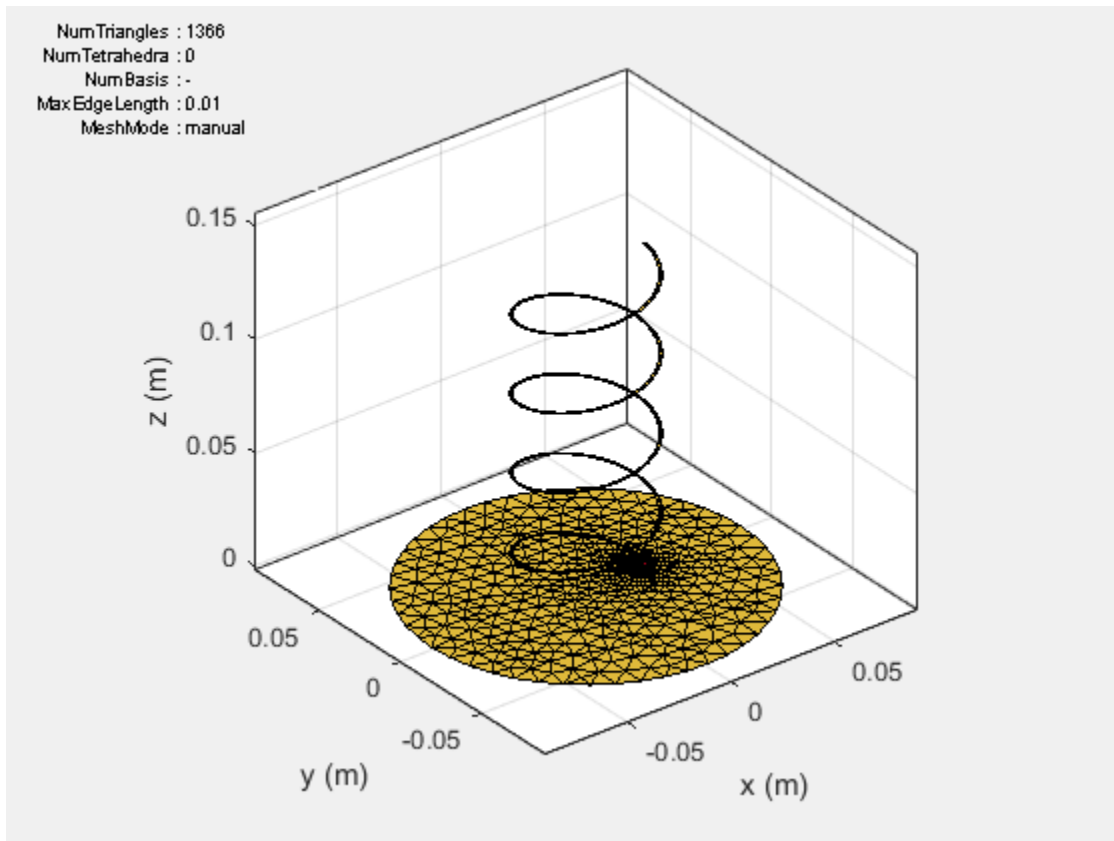
```
mesh(hx)
```



### Mesh Antenna Manually

Specify the maximum edge length for the triangles using the 'MaxEdgeLength' name-value pair. This name-value pair meshes the helix structure manually.

```
mesh(hx, 'MaxEdgeLength', 0.01)
```



### Change Meshing to Automatic

```
meshconfig(hx, 'auto')
```

```
ans =
```

```
struct with fields:
```

```
    NumTriangles: 1366  
    NumTetrahedra: 0  
           NumBasis: []  
    MaxEdgeLength: 0.0100  
           MeshMode: 'auto'
```

Please find the link to Antenna Near Field visualization example below: “Antenna Near-Field Visualization”

**References**

[1] Balanis, C.A. "Antenna Theory. Analysis and Design", p. 514, Wiley, New York, 3rd Edition, 2005.



# Antenna Toolbox Library Elements

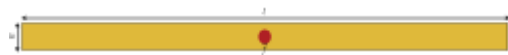
The Antenna Toolbox consists of two libraries: Antenna Library and Array Library.

## In this section...

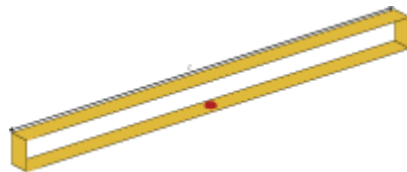
“Dipole Antennas” on page 1-21  
 “Monopole Antennas” on page 1-24  
 “Loop Antennas” on page 1-25  
 “Spiral Antennas” on page 1-25  
 “Patch Antennas” on page 1-26  
 “Other Antennas” on page 1-27  
 “Arrays” on page 1-29

## Dipole Antennas

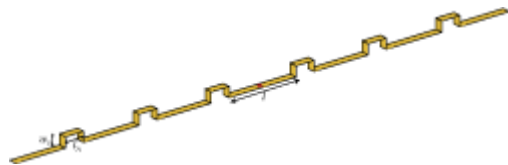
dipole



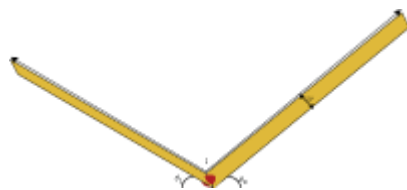
dipoleFolded



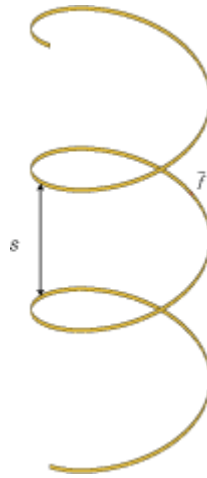
dipoleMeander



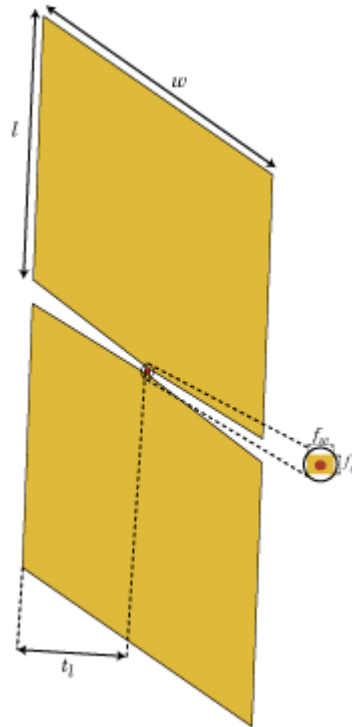
dipoleVee



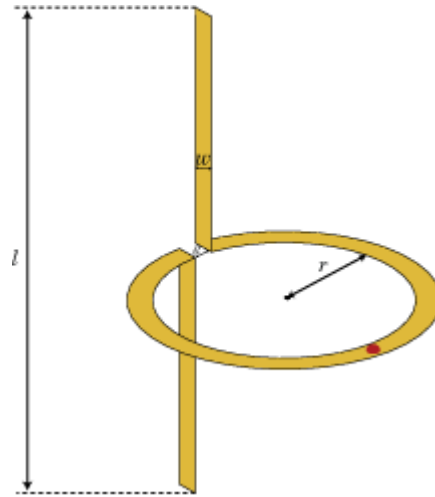
dipoleHelix



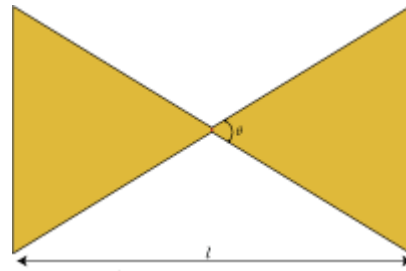
dipoleBlade



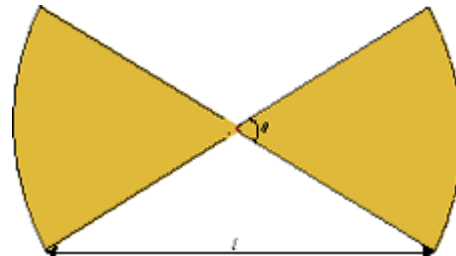
dipoleCycloid



bowtieTriangular

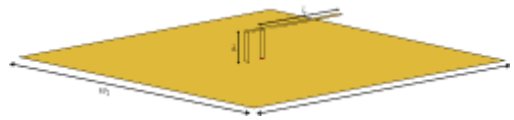


bowtieRounded

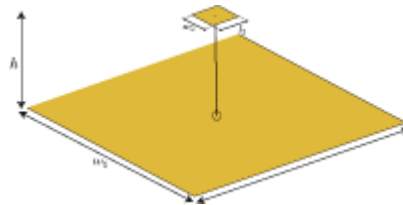


## Monopole Antennas

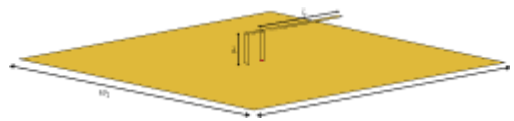
monopole



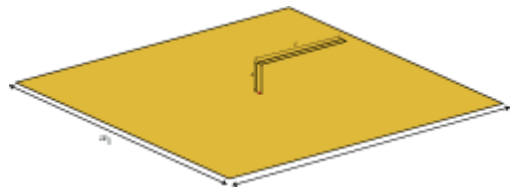
monopoleTopHat



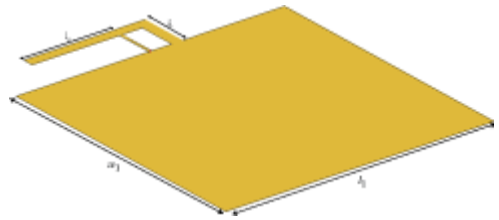
invertedF



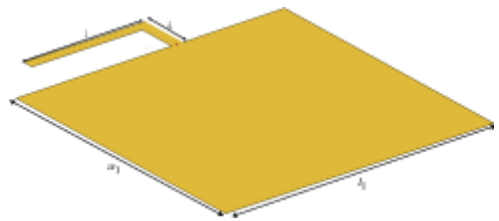
invertedL



invertedFcoplanar

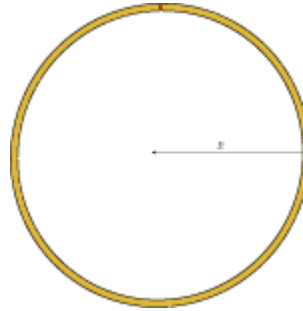


invertedLcoplanar



## Loop Antennas

loopCircular



loopRectangular



## Spiral Antennas

spiralArchimedean

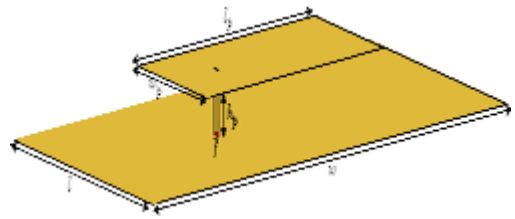


spiralEquiangular

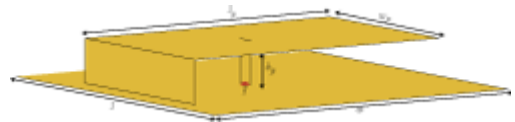


## Patch Antennas

patchMicrostrip

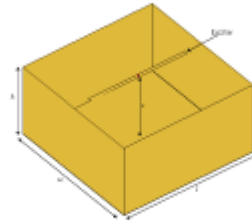


pifa

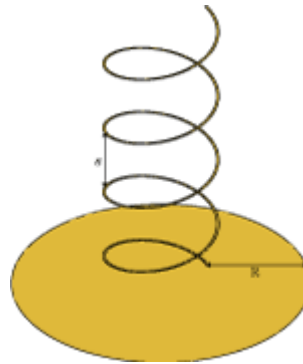


## Other Antennas

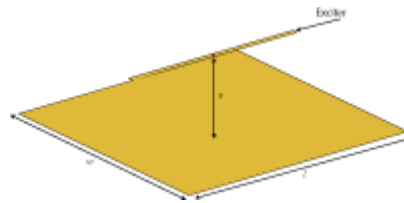
cavity



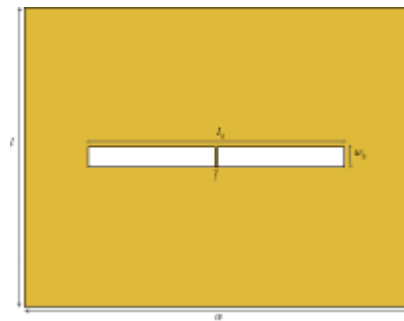
helix



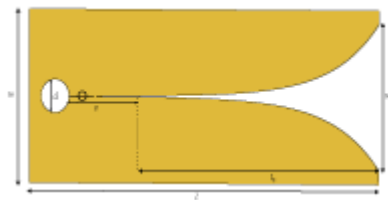
reflector



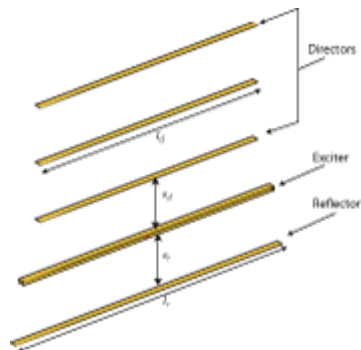
slot



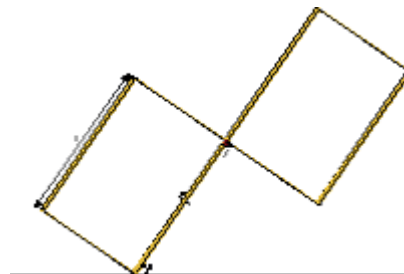
vivaldi



yagiUda

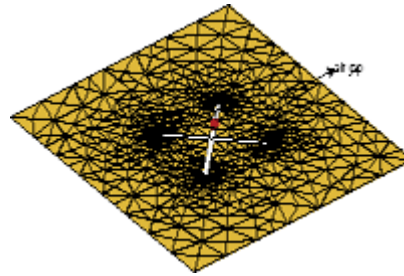


biquad

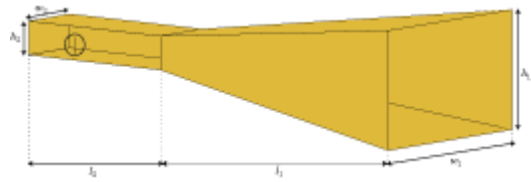




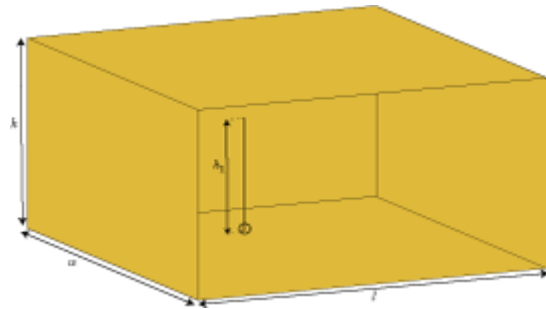
customAntennaMesh



horn

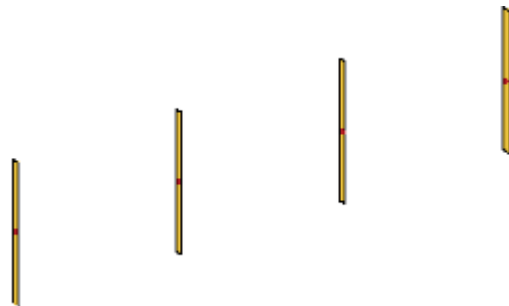


waveguide

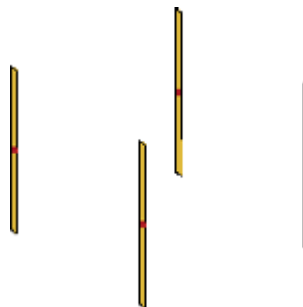


## Arrays

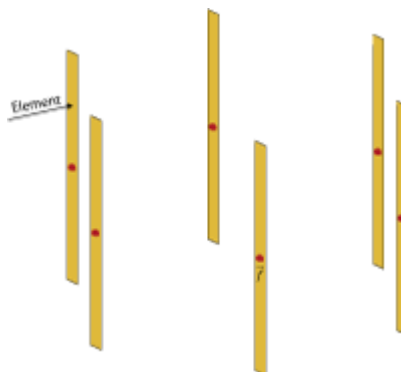
linearArray



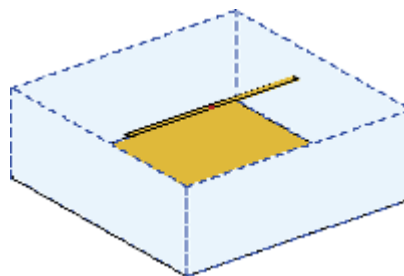
rectangularArray



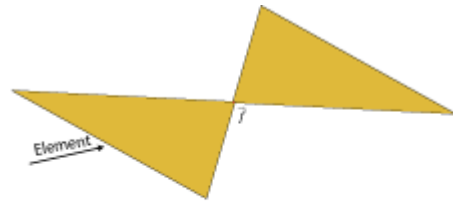
circularArray



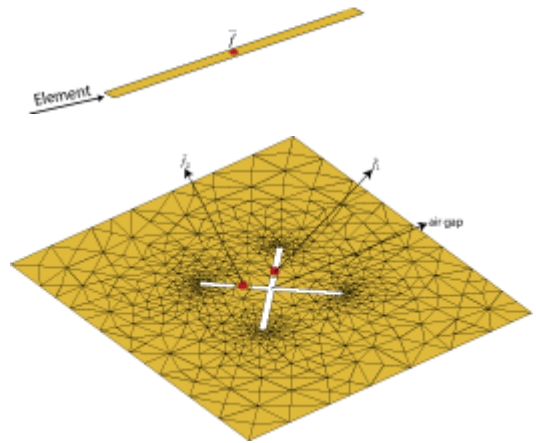
infiniteArray



conformalArray



customArrayMesh



## Antenna Classification

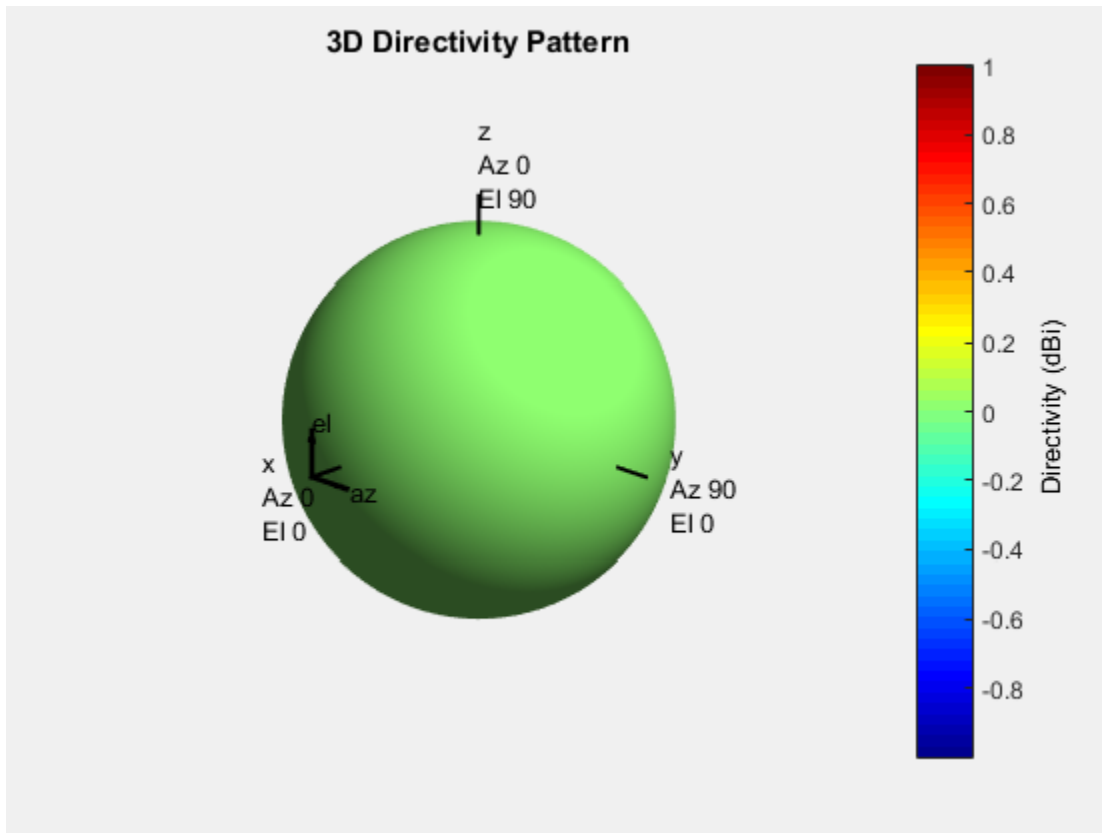
Antennas are classified based on the radiation pattern or the feeding mechanism. Antenna radiation pattern is the angular variation of signal strength around the antenna. Feeding mechanism defines the how the signal is fed into the antenna and the location of the feed point on the antenna.

In this section...
“Radiation Pattern” on page 1-32
“Antenna Feeding Mechanism” on page 1-35

### Radiation Pattern

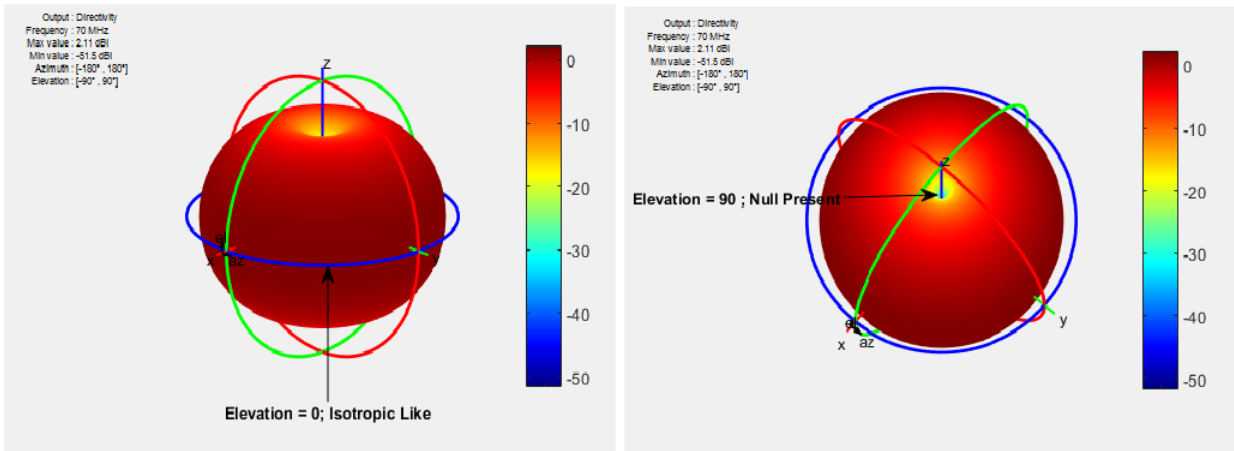
#### Isotropic Antenna

An *isotropic* antenna is an ideal lossless antenna that radiates uniformly in all directions. The antenna has no spatial selectivity or nulls. Practical antennas are compared against the isotropic antenna, but they rarely behaves like one.



### Omnidirectional Antenna

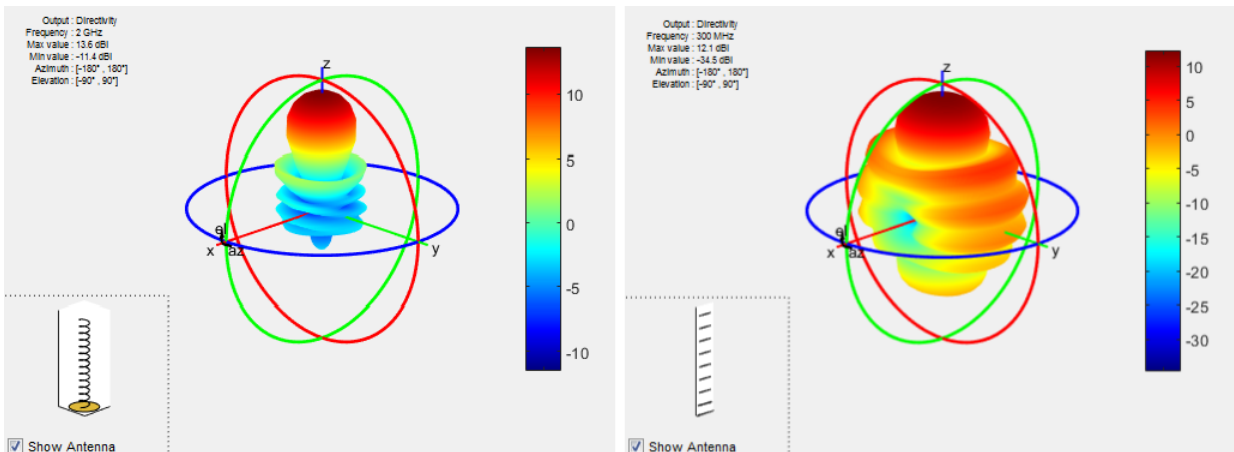
*Omnidirectional* antennas behave like isotropic antennas in one plane. These antennas have nulls in the orthogonal plane. A common example of an omnidirectional antenna is the dipole antenna.



The dipole is omnidirectional around the E-plane, or elevation angle. The null is present in the H-plane, or azimuth angle.

### Directional Antennas

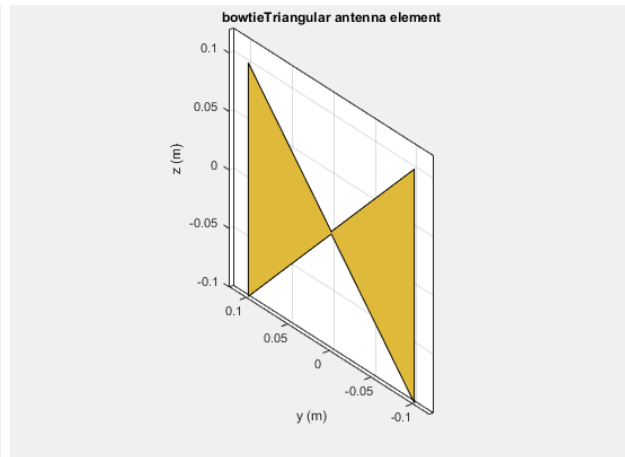
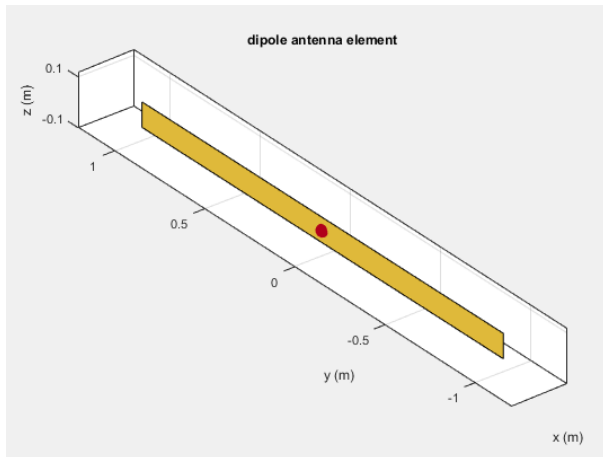
*Directional* antennas are highly directive in a given direction. These antennas show high spatial selectivity, narrow bandwidth. They also have well defined major, or main, beam in the desired directions. Common examples of directional antennas are helix and yagiUda.



## Antenna Feeding Mechanism

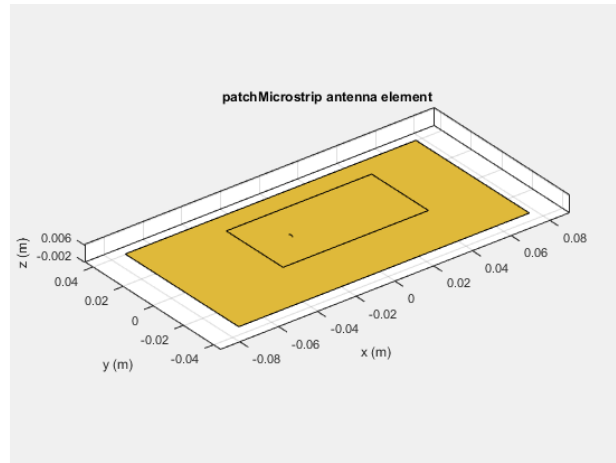
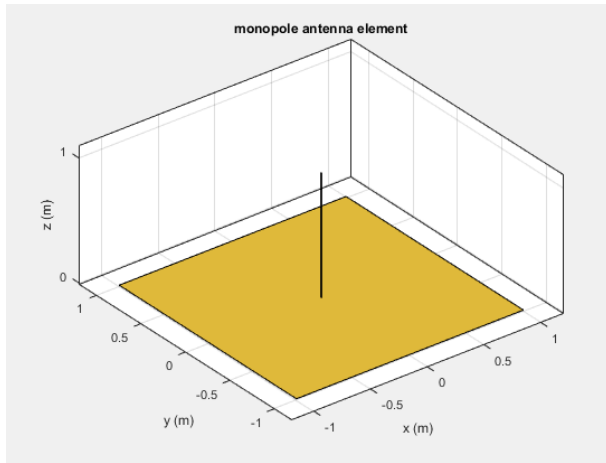
### Balanced Antennas

In *balanced* antennas, one side of the antenna is a mirror image of the other. These antennas require a balun to feed it, using a coaxial line. Common examples are: dipoles, bowties, spirals, and loops.



### Unbalanced Antennas

*Unbalanced* antennas are end fed and mounted on top of a ground plane. The coaxial shield is connected to the ground, and the center conductor is connected to the antenna element. Common examples are monopoles and patches.



## References

- [1] Balanis, C.A. *Antenna Theory: Analysis and Design*. 3rd Ed. New York: Wiley, 2005.



# Antenna Toolbox Coordinate System

Antenna Toolbox uses two types of coordinate system: *rectangular coordinate system* and *spherical coordinate system*.

Antenna Toolbox uses the *rectangular coordinate system* to visualize antenna or array geometry. The toolbox uses the *spherical coordinate system* to visualize antenna radiation patterns.

## In this section...

“Rectangular Coordinate System” on page 1-37

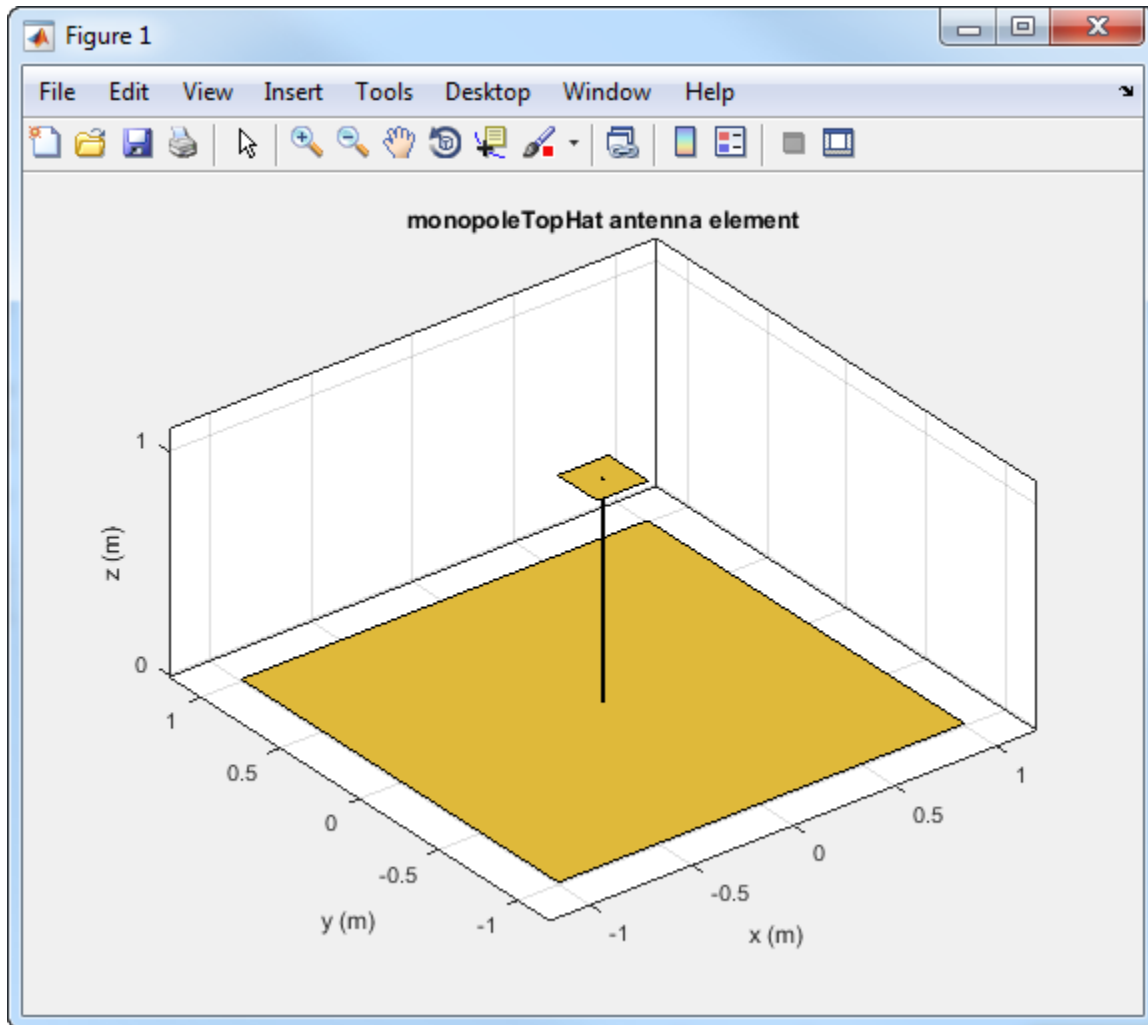
“Spherical Coordinate System” on page 1-40

“Conversion Between Rectangular and Spherical Coordinates” on page 1-44

## Rectangular Coordinate System

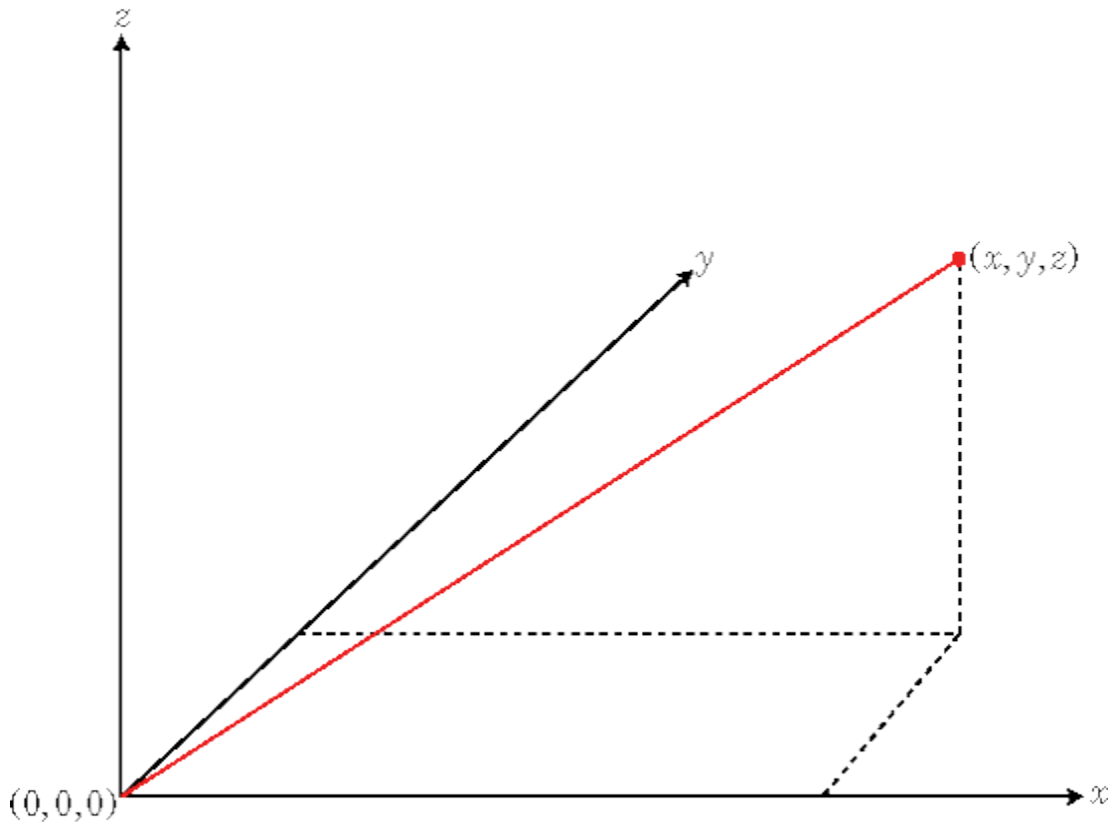
Visualize the geometry of a default `monopoleTopHat` antenna from the antenna library.

```
m = monopoleTopHat;  
show(m);
```



The toolbox displays the top-hat monopole antenna in the *rectangular* or *Cartesian* coordinate system.

The *rectangular* coordinate system also called *Cartesian* coordinate system specifies a position in space as an ordered 3-tuple of real numbers,  $(x, y, z)$ , with respect to the origin  $(0, 0, 0)$ .



You can view the 3-tuple as a point in space, or equivalently as a vector in three-dimensional Euclidean space. When viewed as a vector in space, the coordinate axes are basis vectors and the vector gives the direction to a point in space from the origin. Every vector in space is uniquely determined by a linear combination of the basis vectors. The most common set of basis vectors for three-dimensional Euclidean space are the standard unit basis vectors:

$$\{[1\ 0\ 0],[0\ 1\ 0],[0\ 0\ 1]\}$$

## Orthogonal Basis and Euclidean Norm

Any three linearly independent vectors define a basis for three-dimensional space. However, the Antenna Toolbox assumes that the basis vectors you use are orthogonal.

The standard distance measure in space is the  $l^2$  norm, or Euclidean norm. The Euclidean norm of a vector  $[x\ y\ z]$  is defined by:

$$\sqrt{x^2 + y^2 + z^2}$$

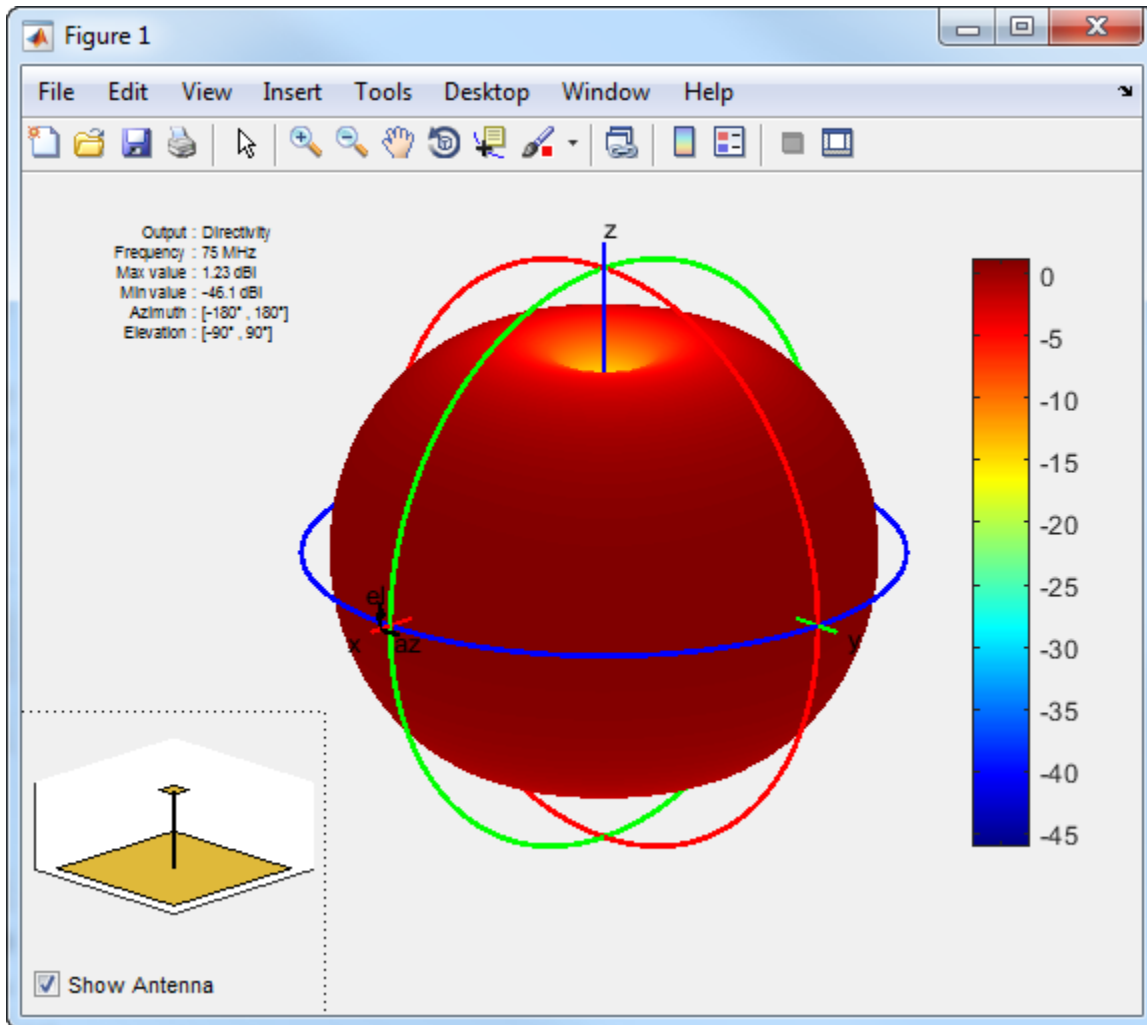
The Euclidean norm gives the length of the vector measured from the origin as the hypotenuse of a right triangle. The distance between two vectors  $[x_0\ y_0\ z_0]$  and  $[x_1\ y_1\ z_1]$  is:

$$\sqrt{(x_0 - x_1)^2 + (y_0 - y_1)^2 + (z_0 - z_1)^2}$$

## Spherical Coordinate System

Visualize the radiation pattern of the default `monopoleTopHat` antenna.

```
m = monopoleTopHat;  
pattern(m, 75e6);
```



The toolbox displays the radiation pattern of the top-hat monopole using *spherical* coordinate system represented by azimuth and elevation angles.

The *spherical* coordinate system defines a vector or point in space with a distance  $R$  and two angles. You can represent the angles in this coordinate system:

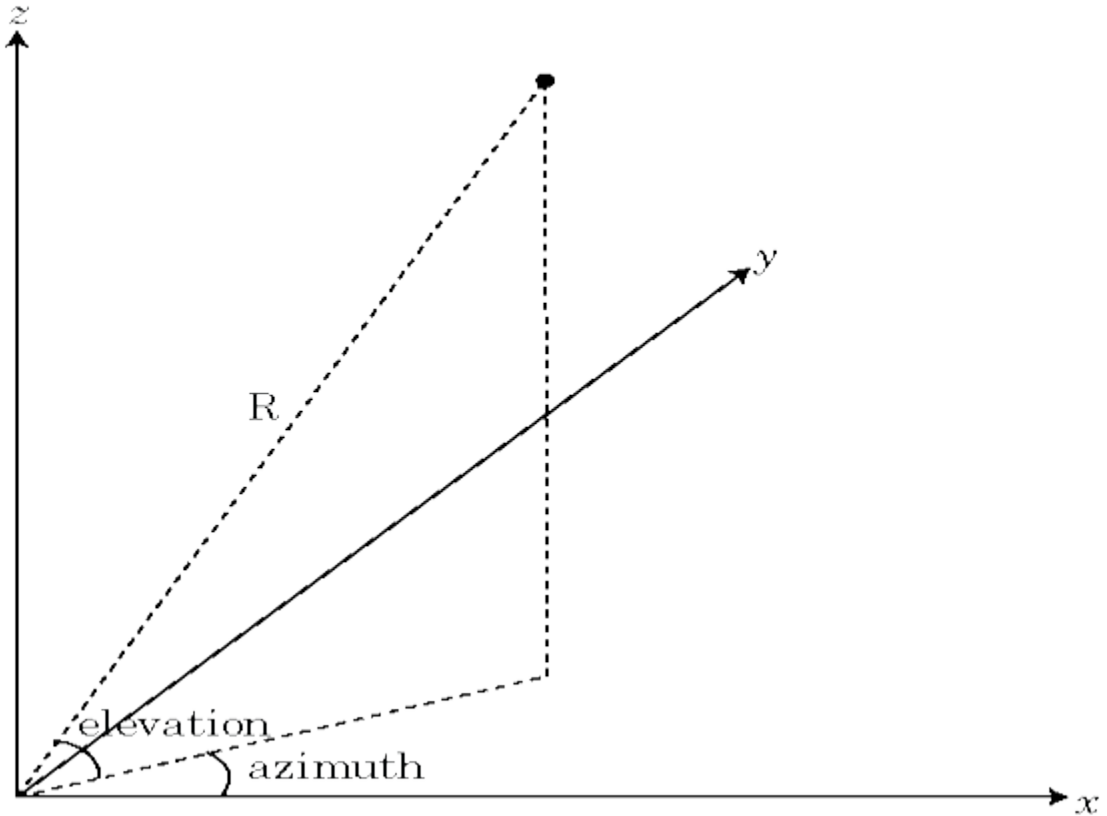
- Azimuth and elevation angles

- Phi ( $\Phi$ ) and theta ( $\theta$ ) angles
- $u$  and  $v$  coordinates

### Azimuth and Elevation Angles

The *azimuth angle* is the angle from the positive  $x$ -axis to the vector's orthogonal projection onto the  $xy$  plane, moving in the direction towards the  $y$ -axis. The azimuth angle is in the range  $-180$  and  $180$  degrees.

The *elevation angle* is the angle from the vector's orthogonal projection on the  $xy$  plane toward the positive  $z$ -axis, to the vector. The elevation angle is in the  $-90$  and  $90$  degrees.

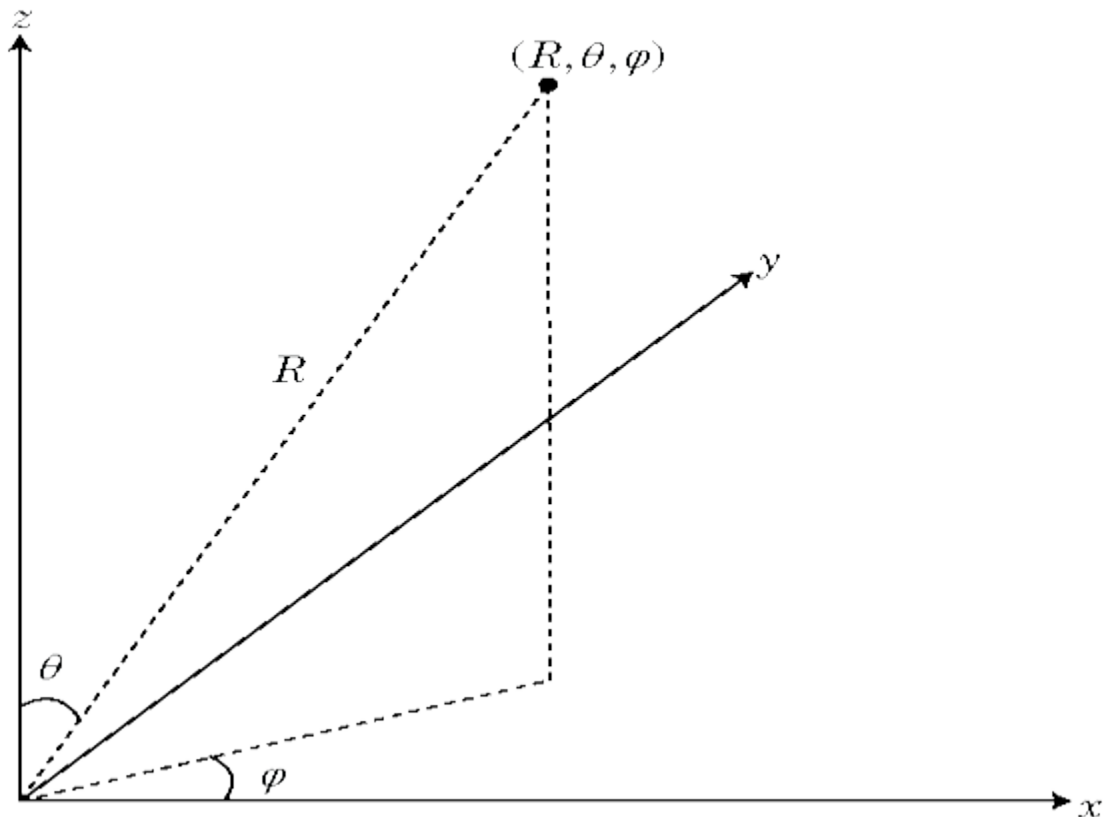


### Phi ( $\Phi$ ) and Theta ( $\theta$ ) Angles

The  $\varphi$  angle is the angle from the positive  $x$ -axis to the vector's orthogonal projection onto the  $xy$  plane, moving in the direction towards the  $y$ -axis. The azimuth angle is between  $-180$  and  $180$  degrees.

The  $\theta$  angle is the angle from the positive  $z$ -axis to the vector's orthogonal projection on the  $xy$  plane measured clockwise. The  $\theta$  angle is in the range  $0$  and  $180$  degrees.

These angles are an alternative to using azimuth and elevation angles to express the location of point in a unit sphere.



### $u$ and $v$ Coordinates

You can define  $u$  and  $v$  in terms of  $\varphi$  and  $\theta$ :

$$u = \sin \theta \cos \phi$$

$$v = \sin \theta \sin \phi$$

In terms of azimuth and elevation angles, the  $u$  and  $v$  coordinates are:

$$u = \cos el \sin az$$

$$v = \sin el$$

The values of  $u$  and  $v$  satisfy the inequalities:

$$-1 \leq u \leq 1$$

$$-1 \leq v \leq 1$$

$$u^2 + v^2 \leq 1$$

The  $\phi$  and  $\theta$  angles in terms of  $u$  and  $v$  are:

$$\tan \phi = u / v$$

$$\sin \theta = \sqrt{u^2 + v^2}$$

The azimuth and elevation angles in terms of  $u$  and  $v$  are:

$$\sin el = v$$

$$\tan az = \frac{u}{\sqrt{1 - u^2 - v^2}}$$

## Conversion Between Rectangular and Spherical Coordinates

Convert rectangular coordinates to spherical coordinates ( $az$ ,  $el$ ,  $R$ ) using:

$$R = \sqrt{x^2 + y^2 + z^2}$$

$$az = \tan^{-1}(y / x)$$

$$el = \tan^{-1}(z / \sqrt{x^2 + y^2})$$

Convert spherical coordinates ( $az$ ,  $el$ ,  $R$ ) to rectangular coordinates using:



$$x = R \cos(el) \cos(az)$$

$$y = R \cos(el) \sin(az)$$

$$z = R \sin(el)$$

where:

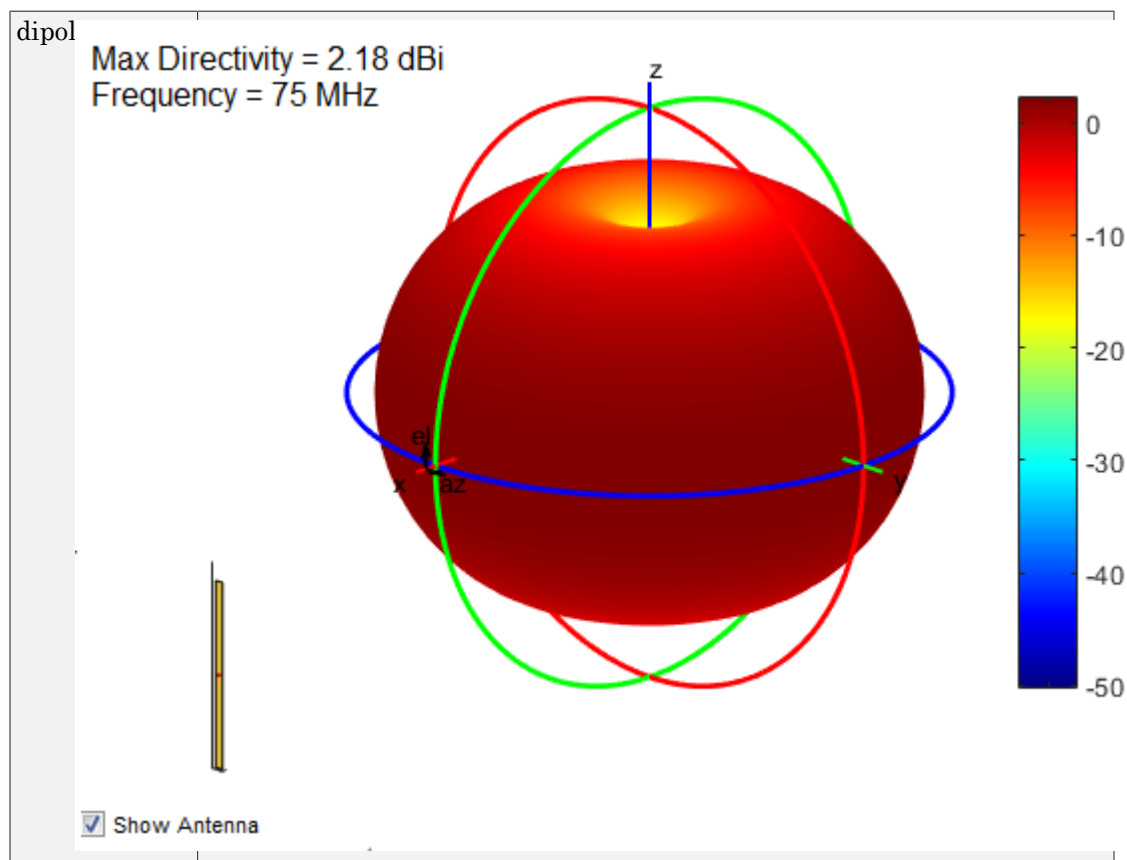
- $R$  is the distance from the antenna
- $el$  and  $az$  are the azimuth and elevation angles

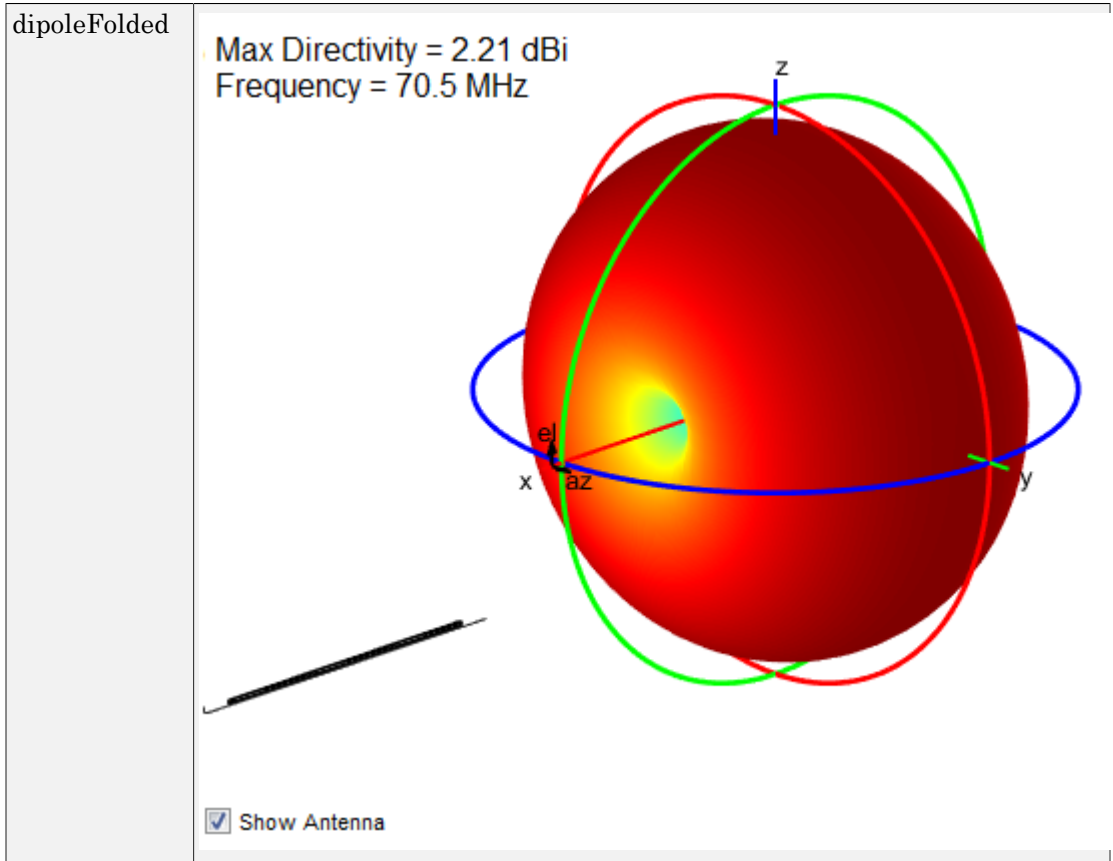
## References

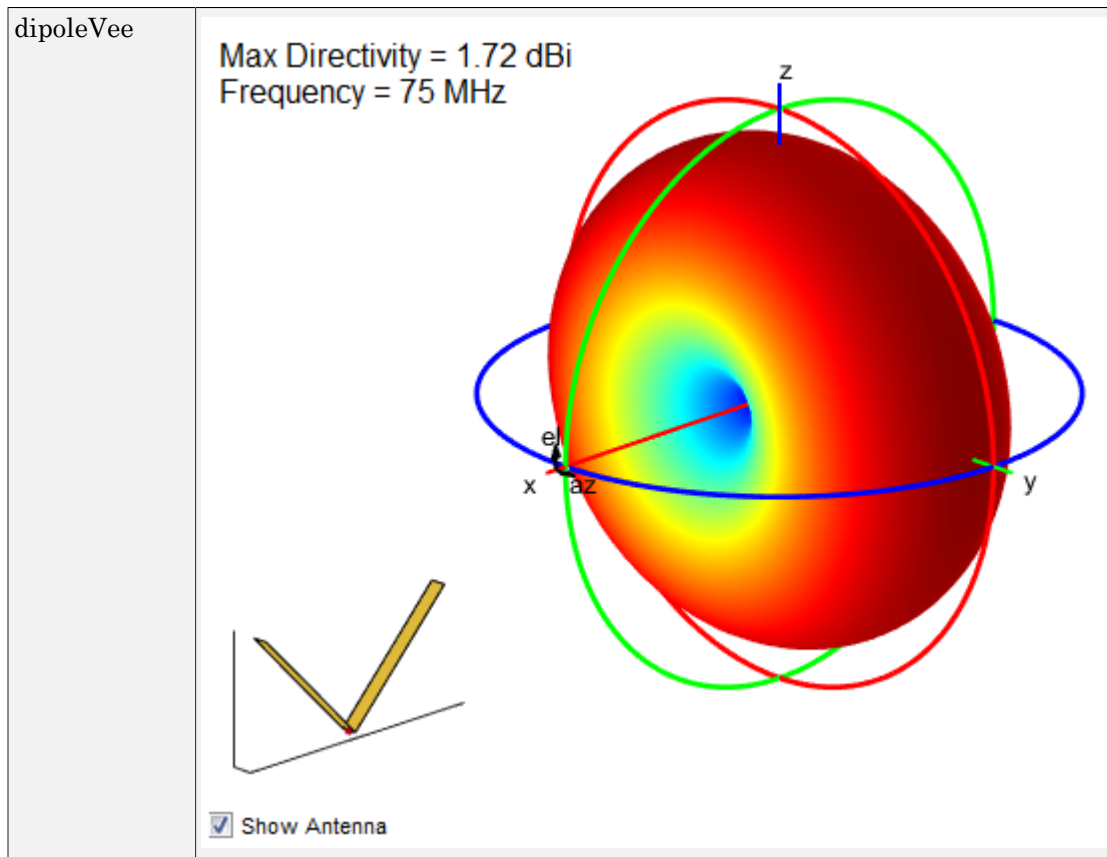
- [1] Balanis, C.A. *Antenna Theory: Analysis and Design*. 3rd Ed. New York: Wiley, 2005.

## Antenna Radiation Patterns

Antenna radiation patterns at default operating frequencies.

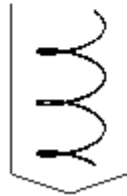
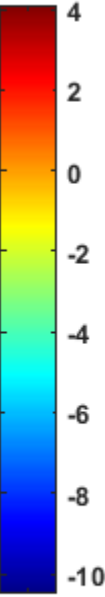
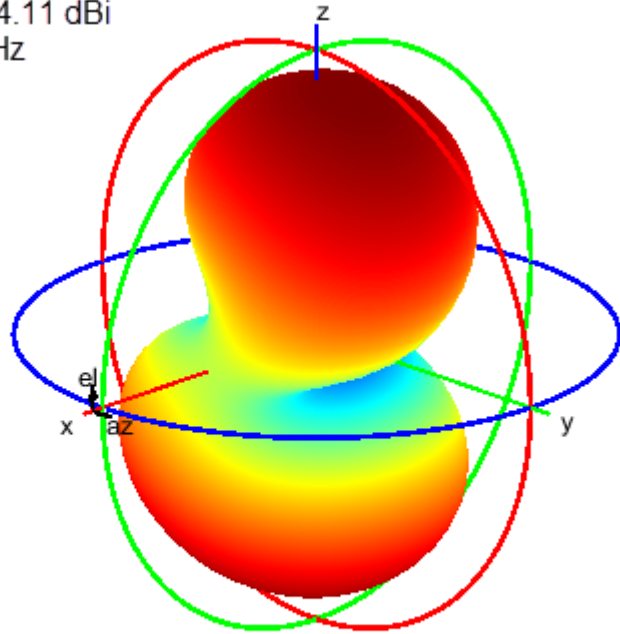




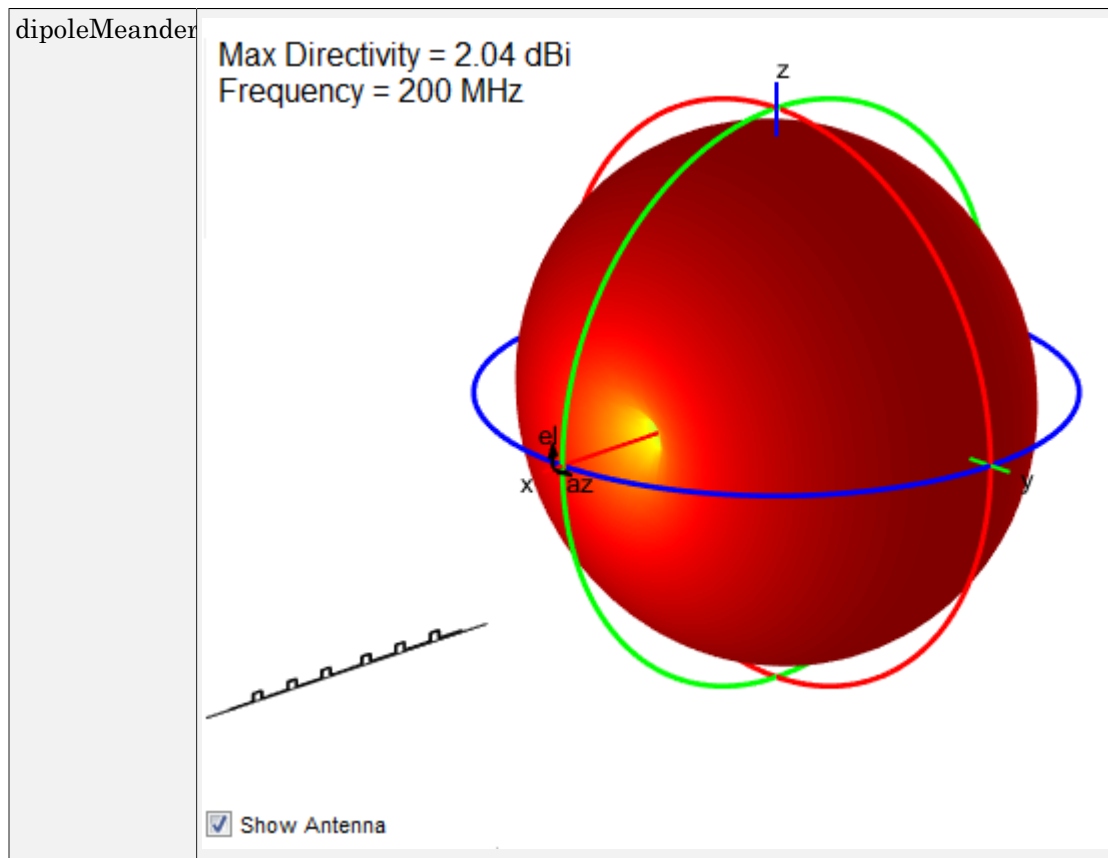


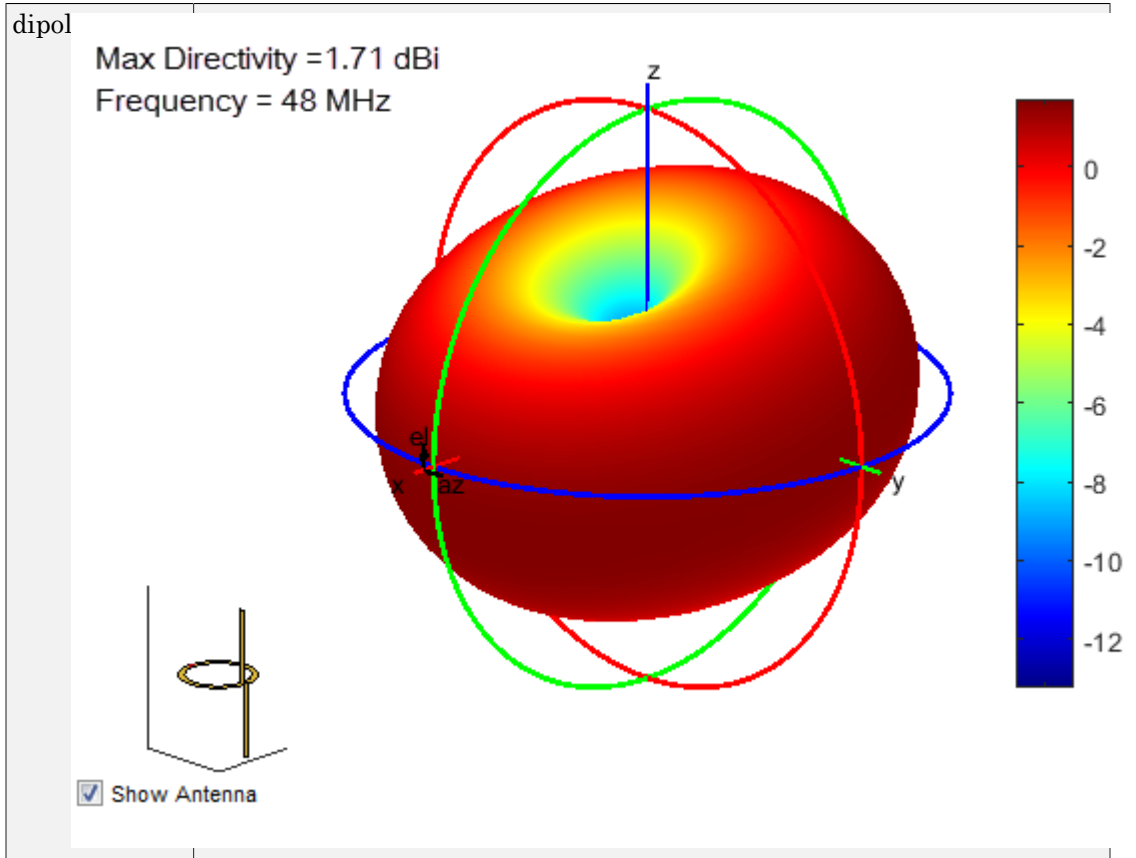
dipoleHelix

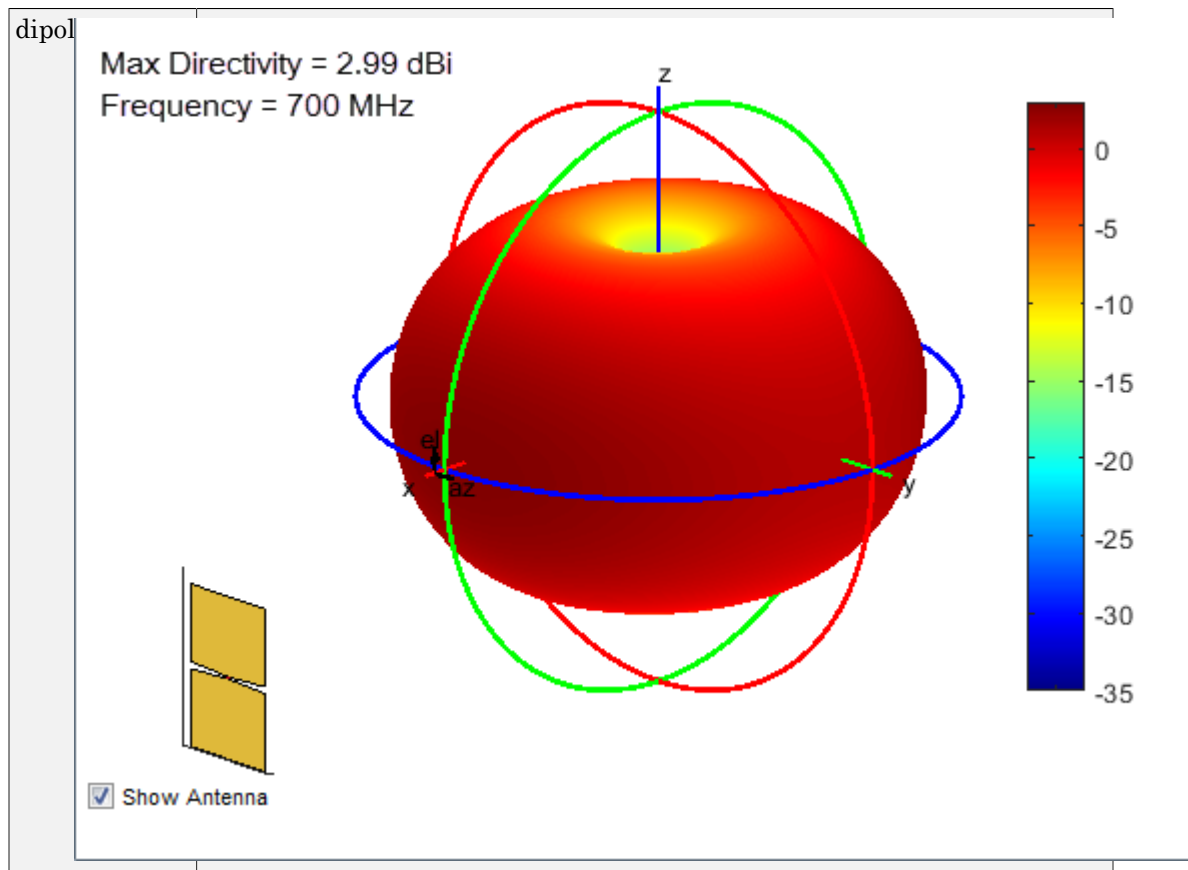
Max Directivity = 4.11 dBi  
 Frequency = 2 GHz



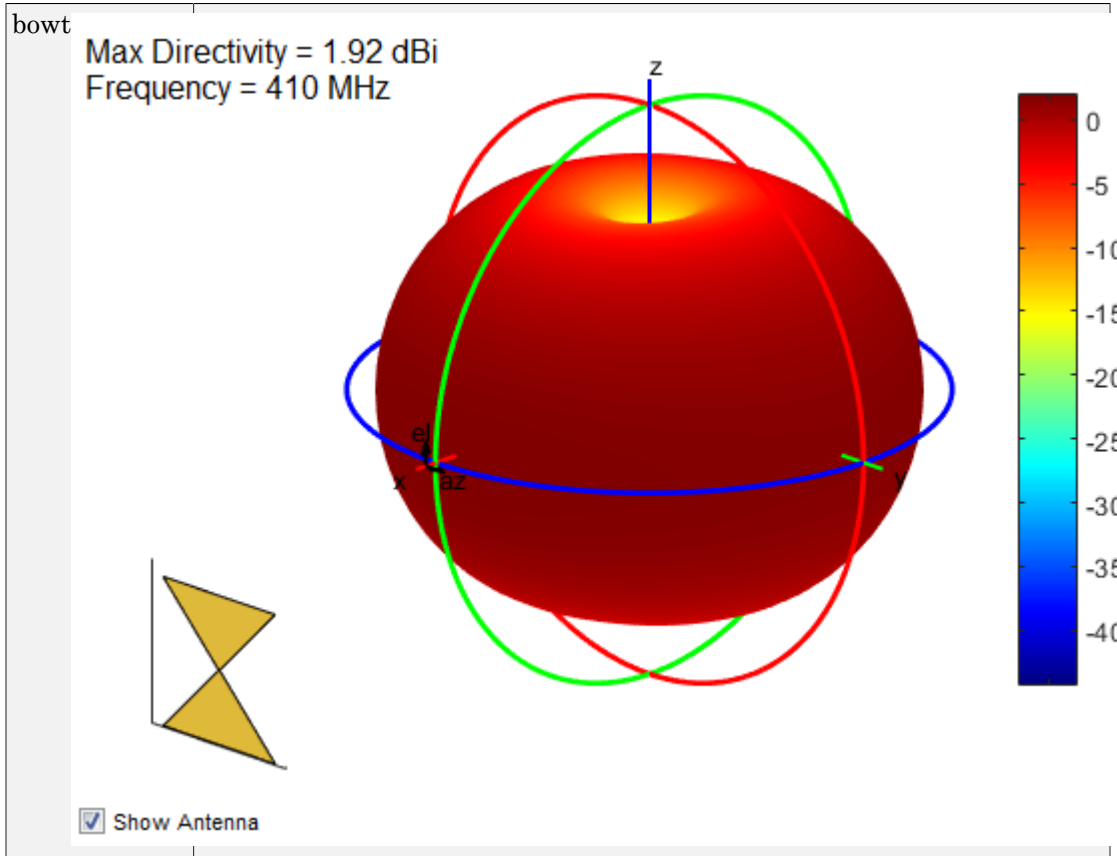
Show Antenna

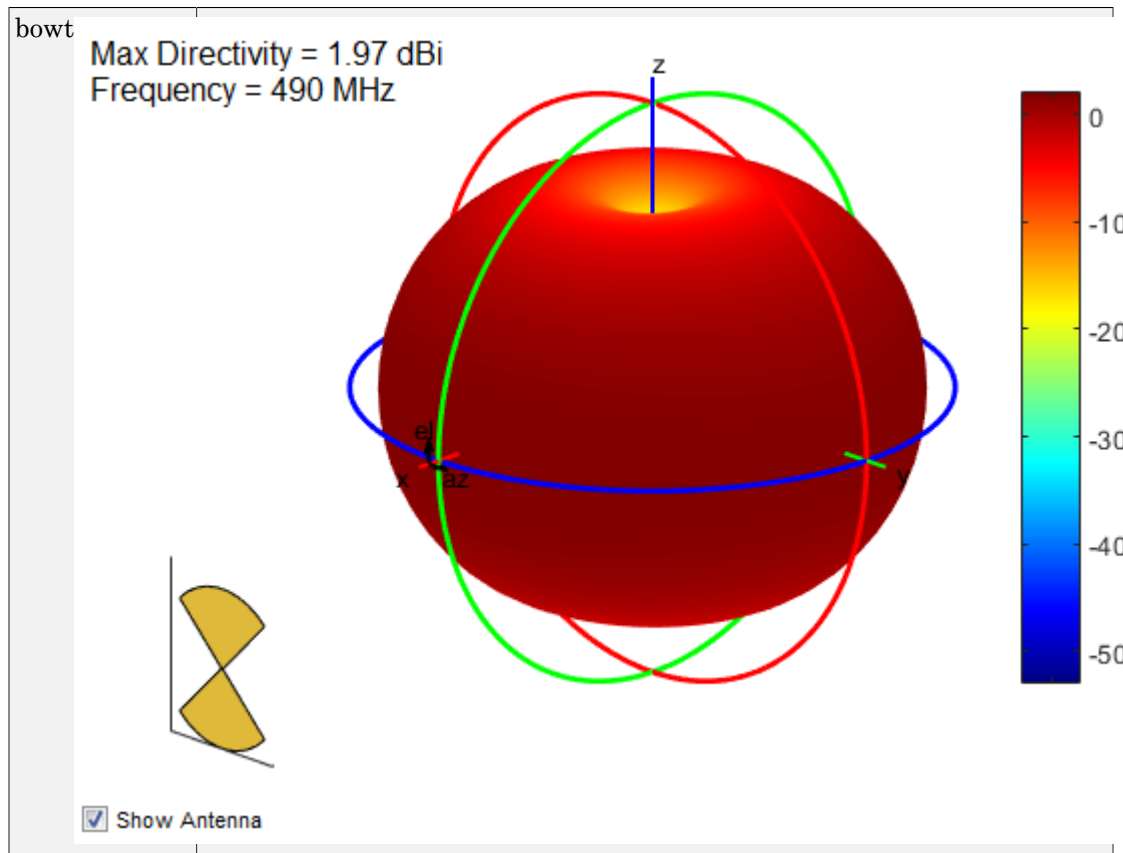


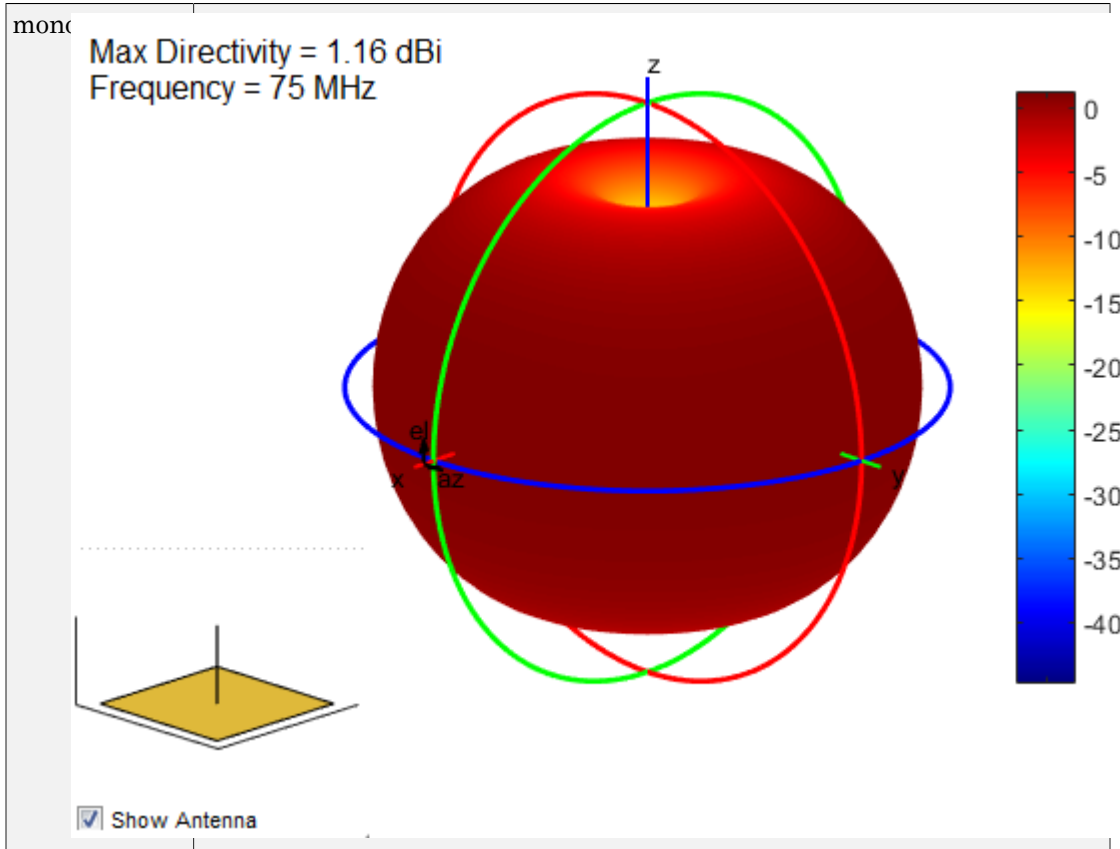


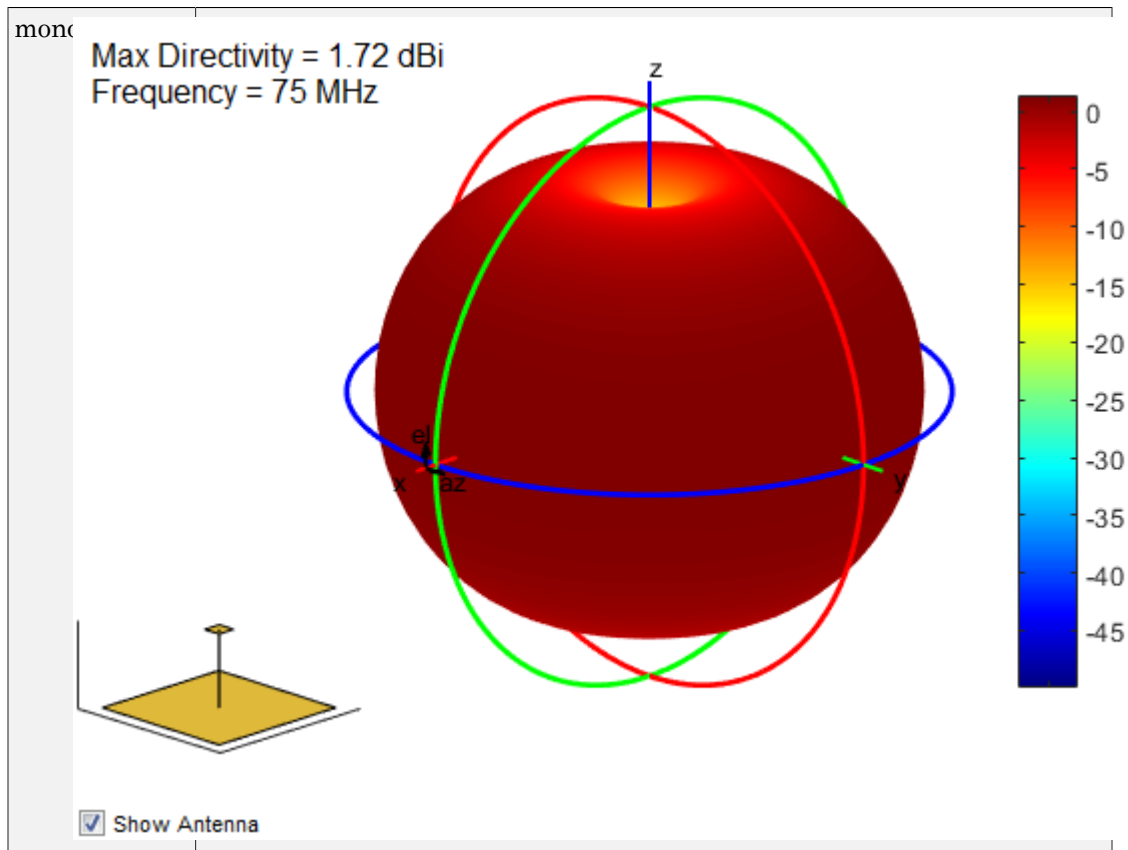


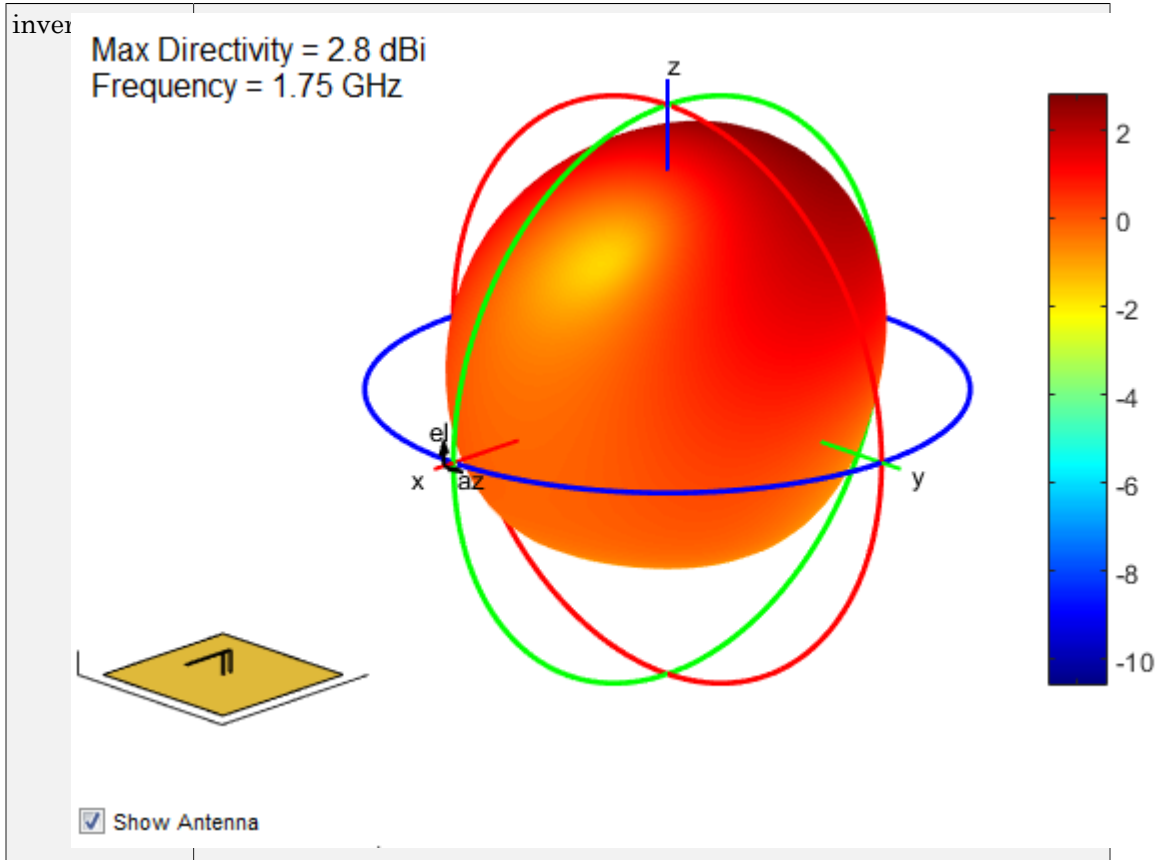


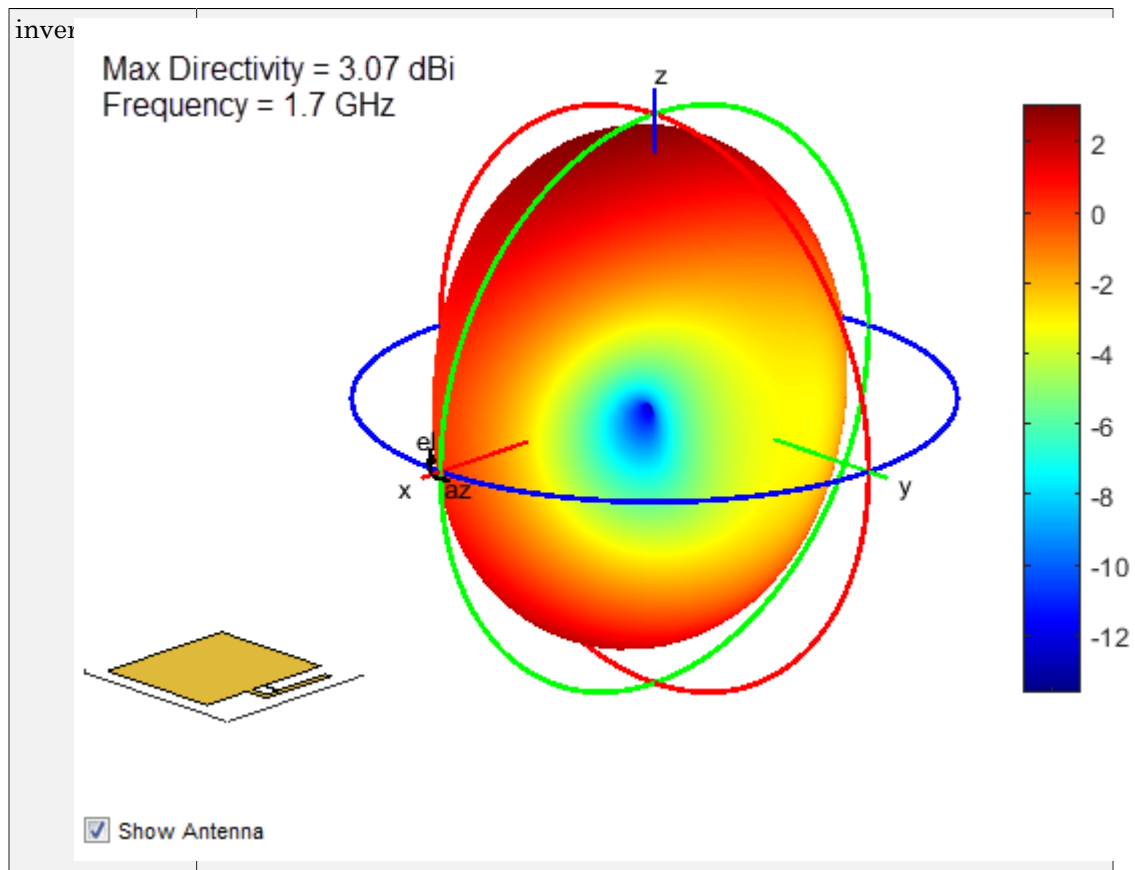


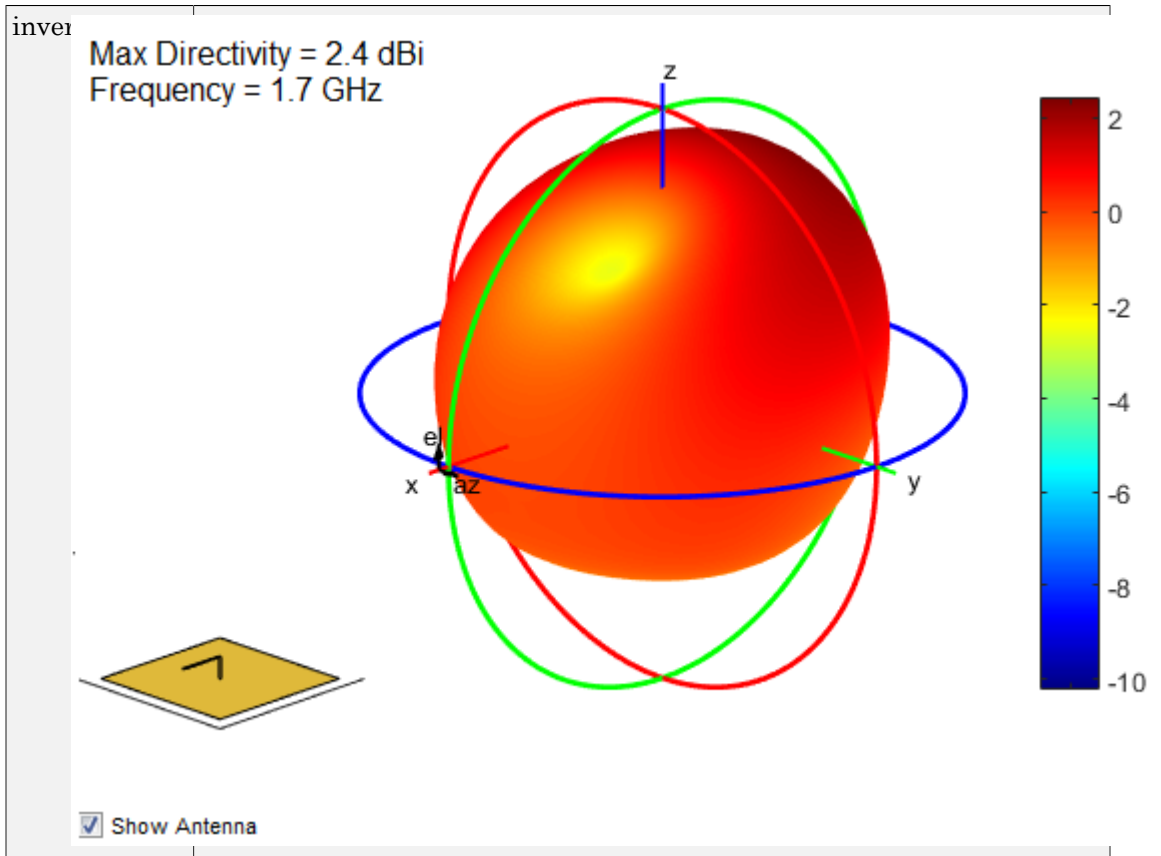


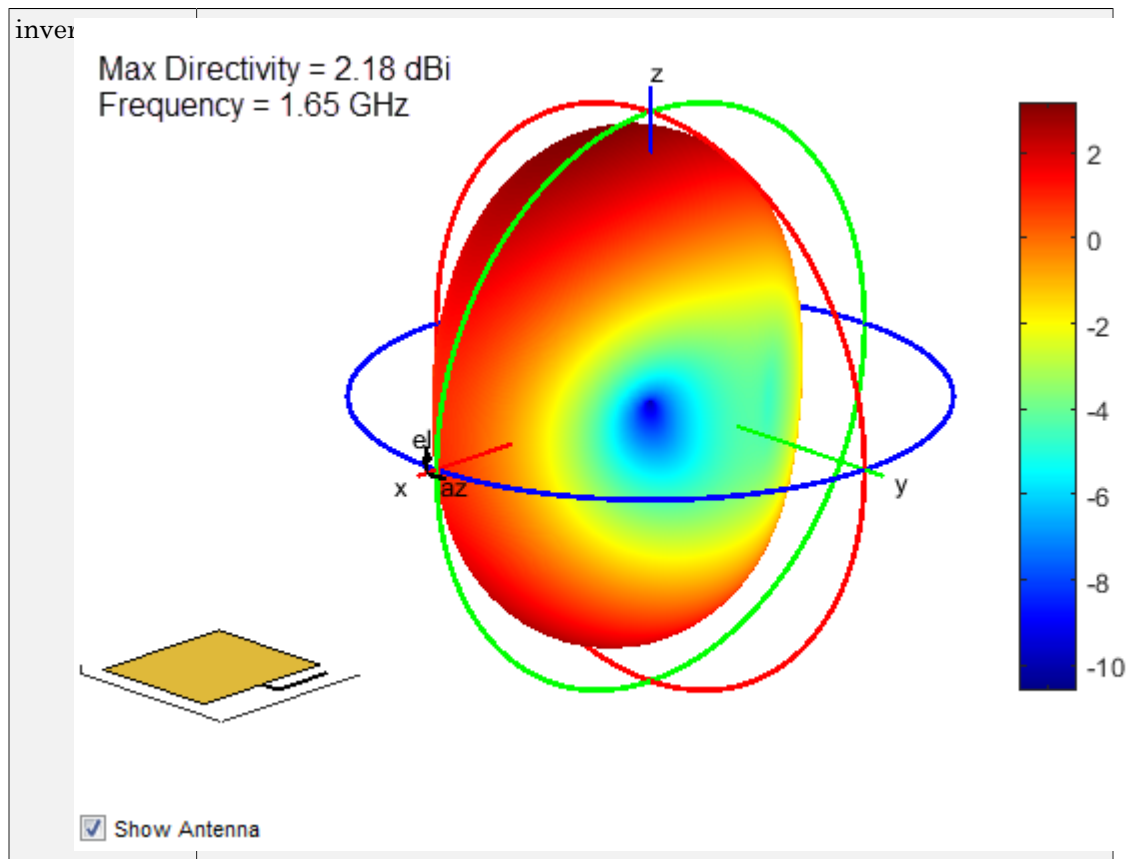




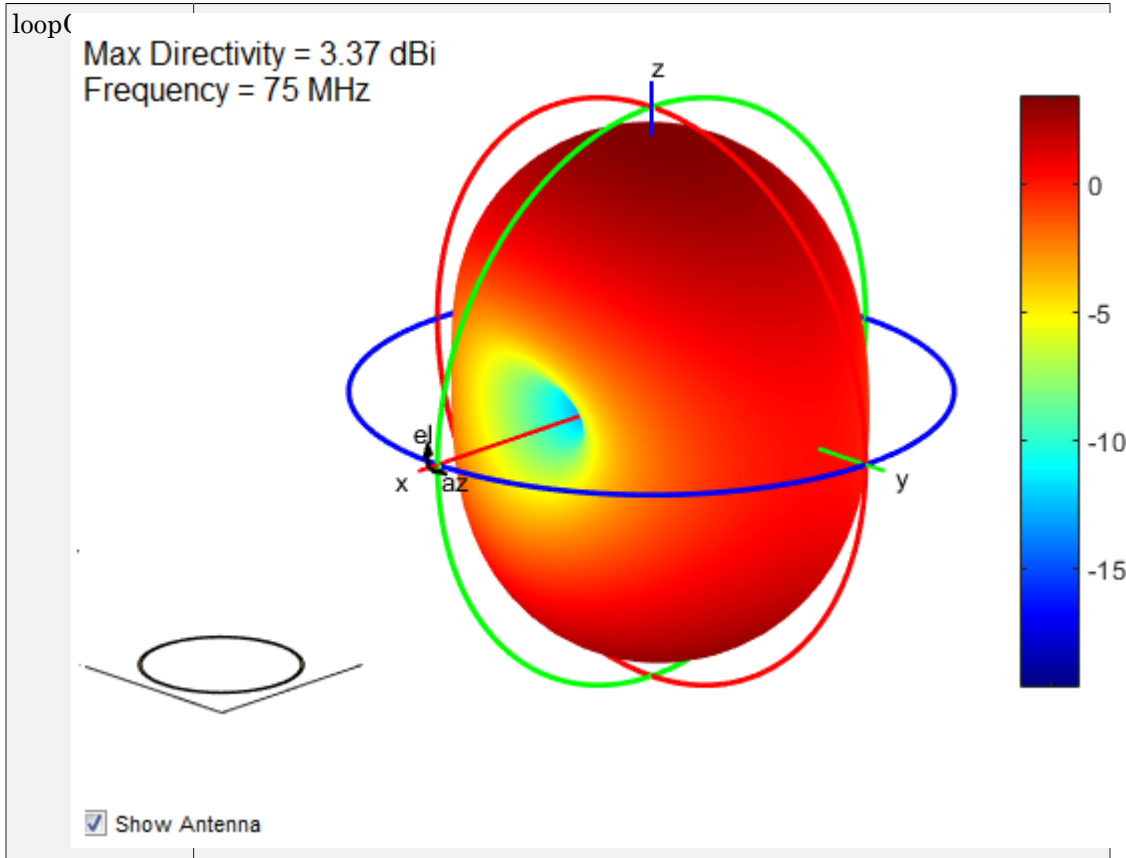


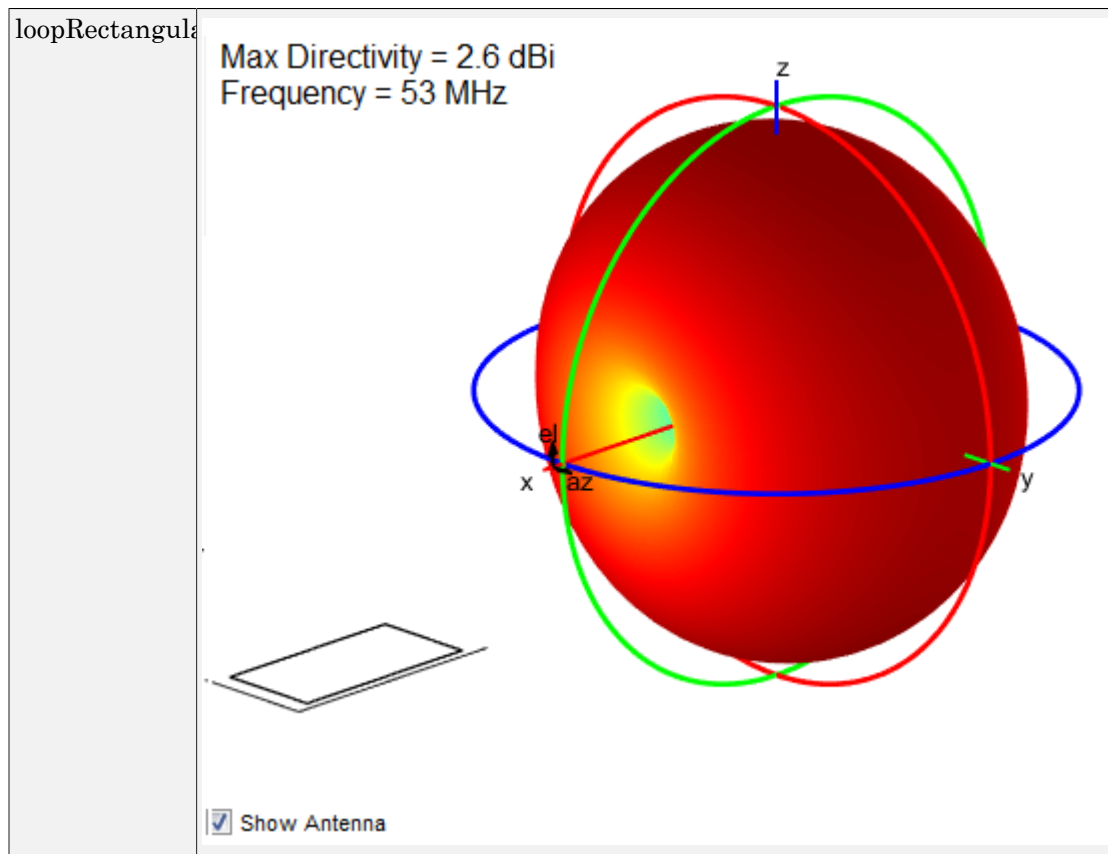


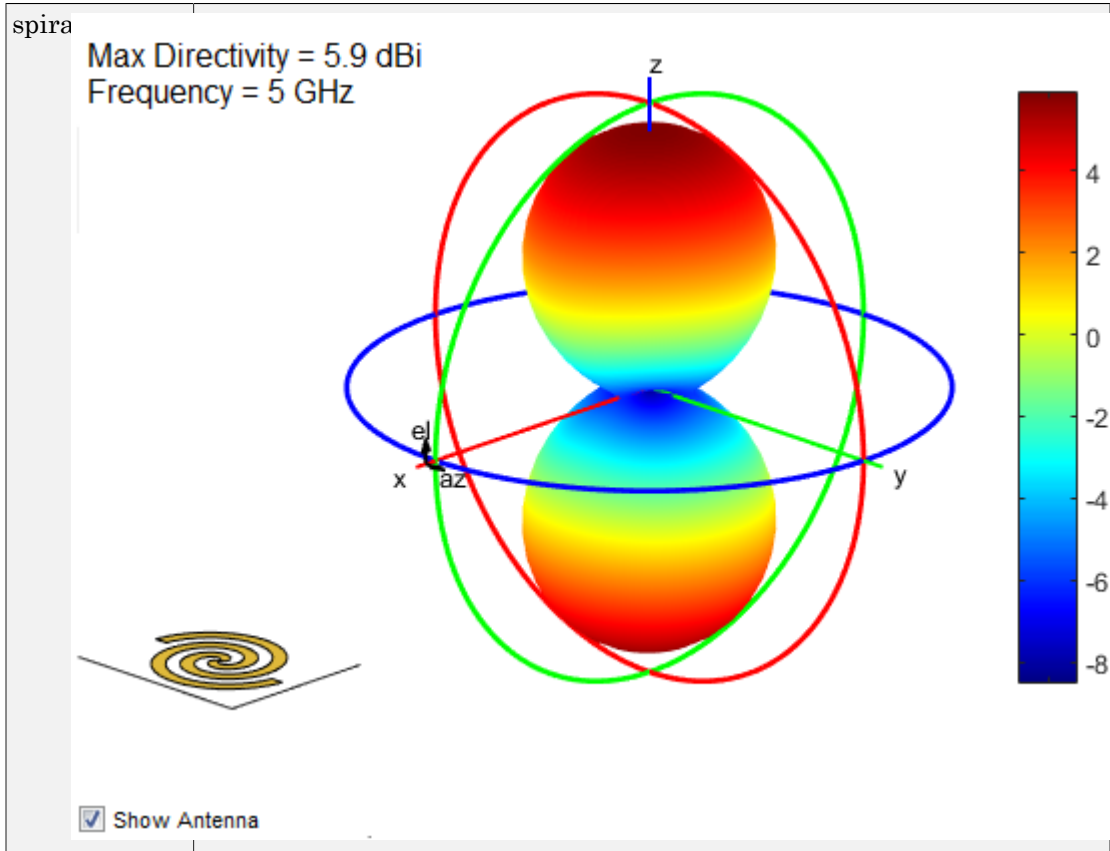


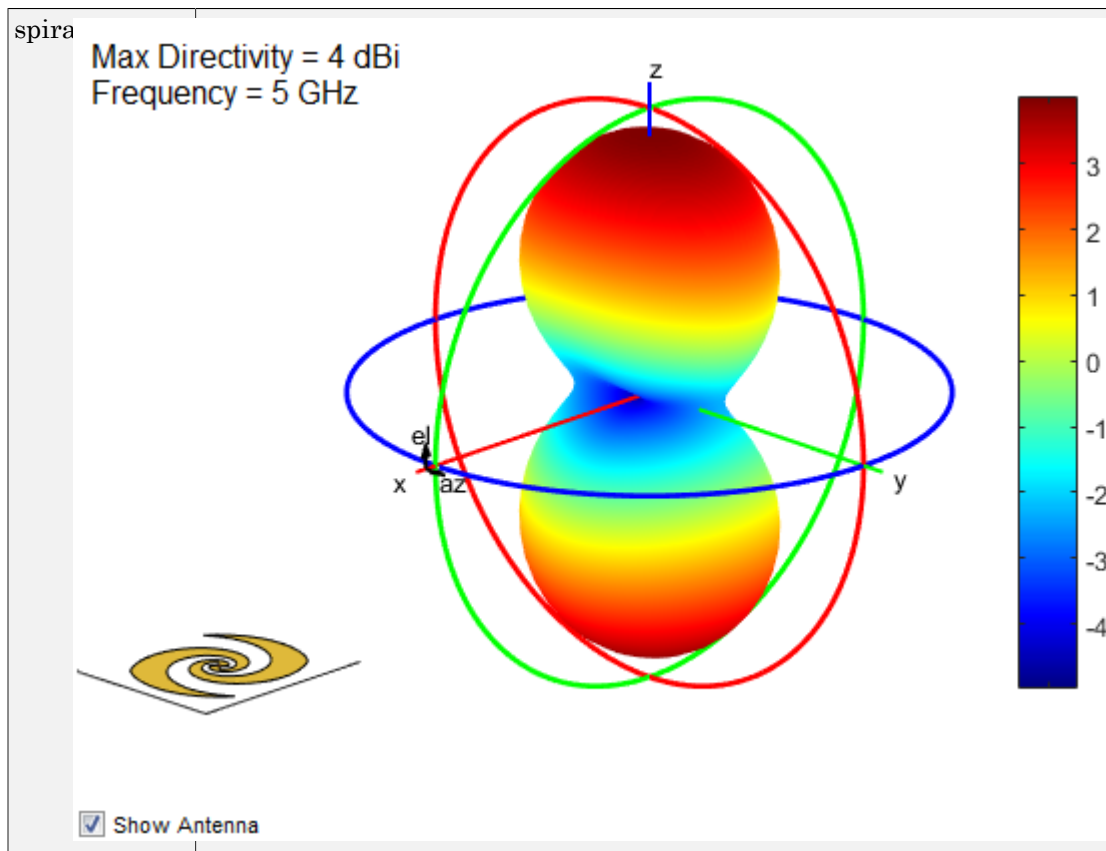


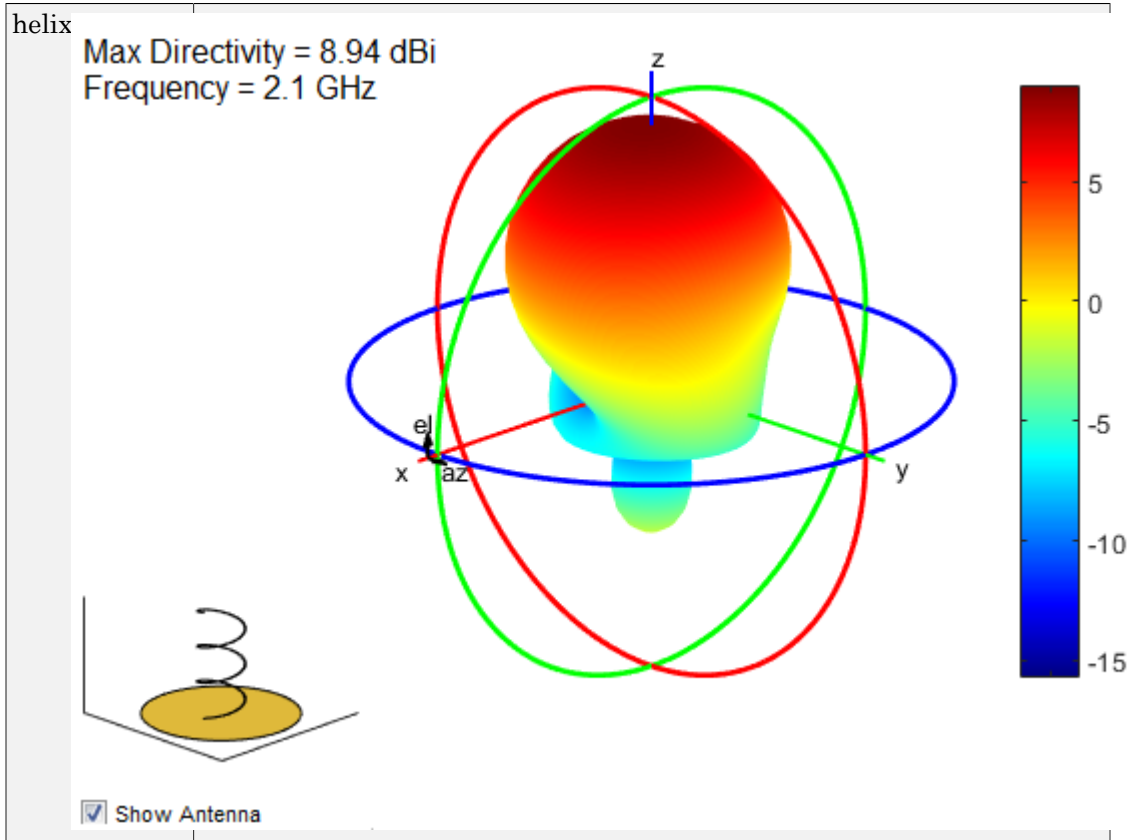


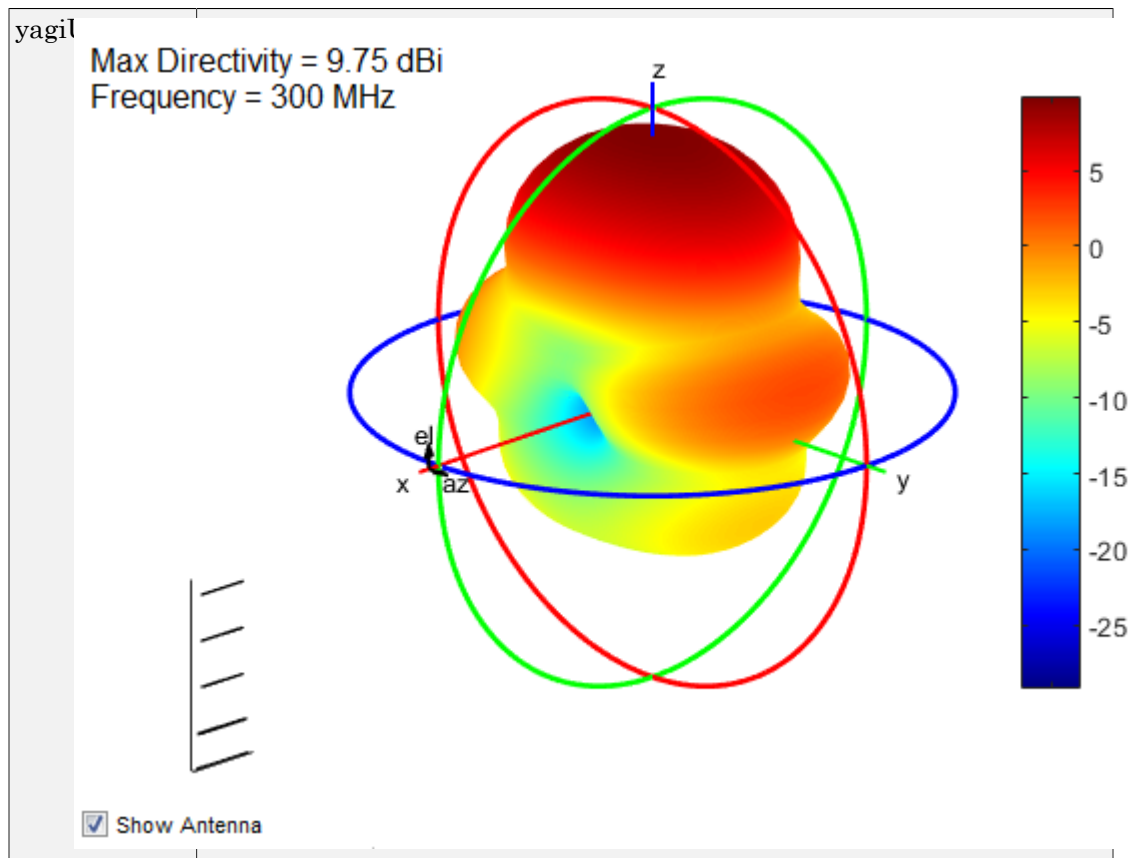


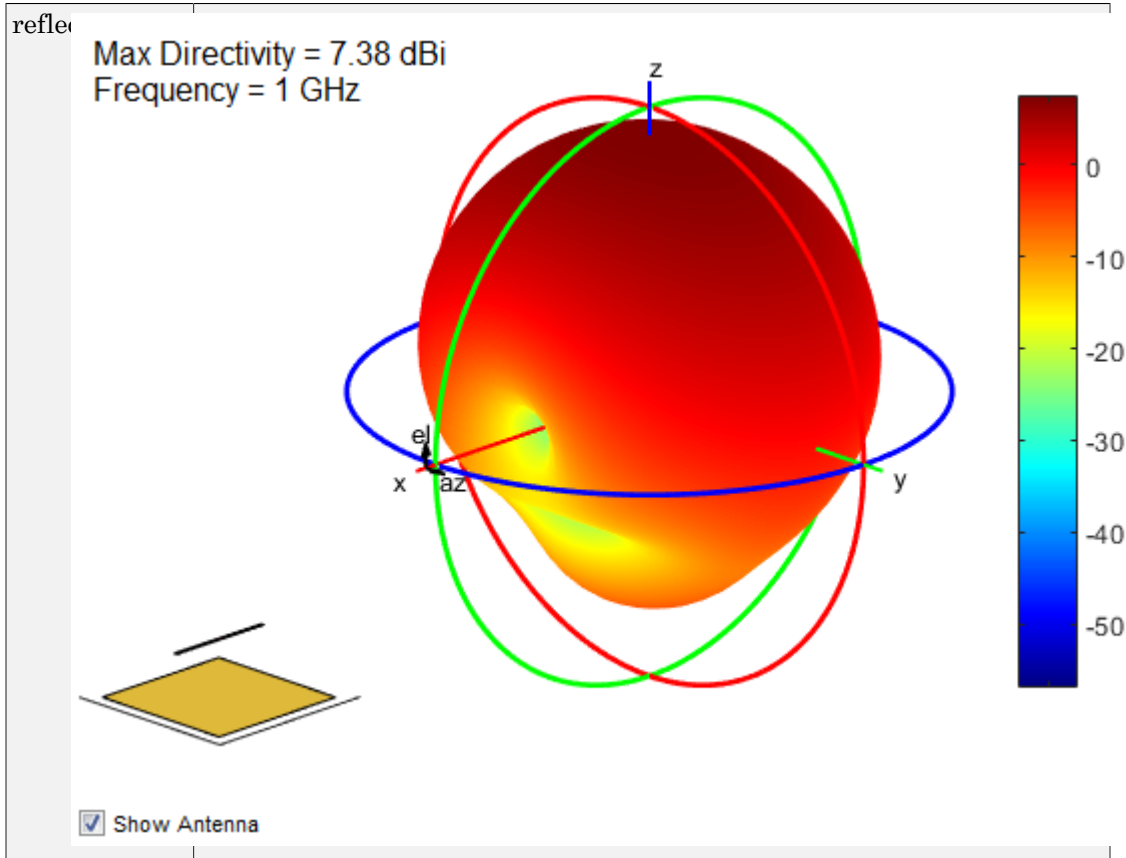


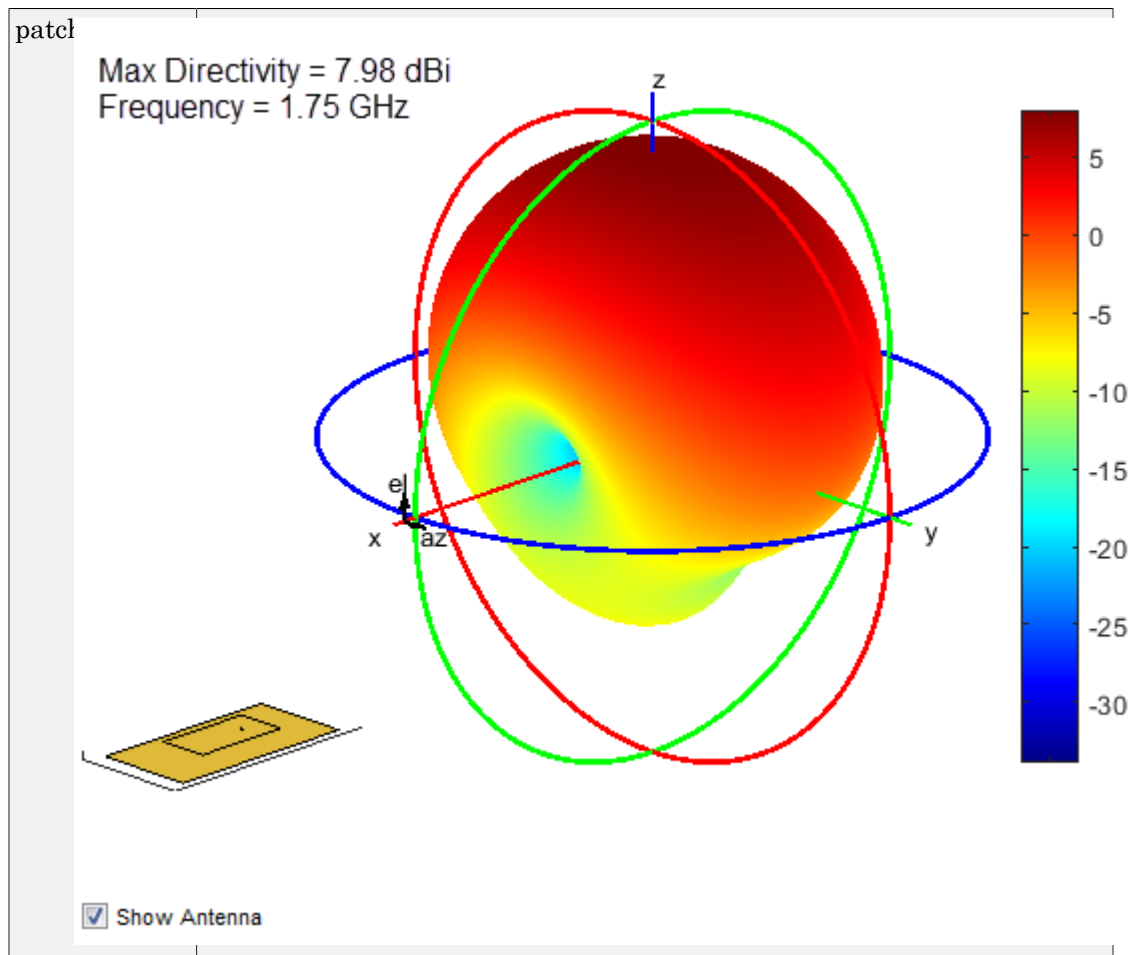




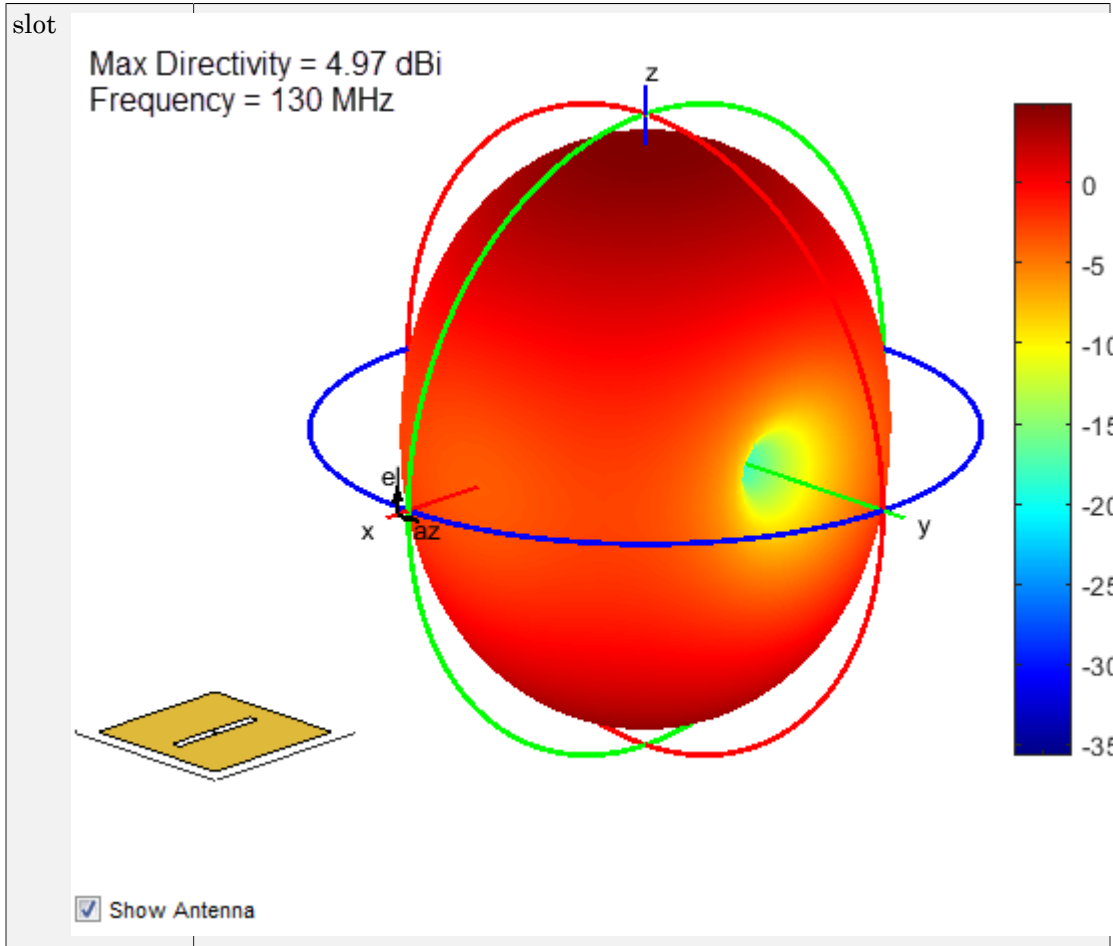


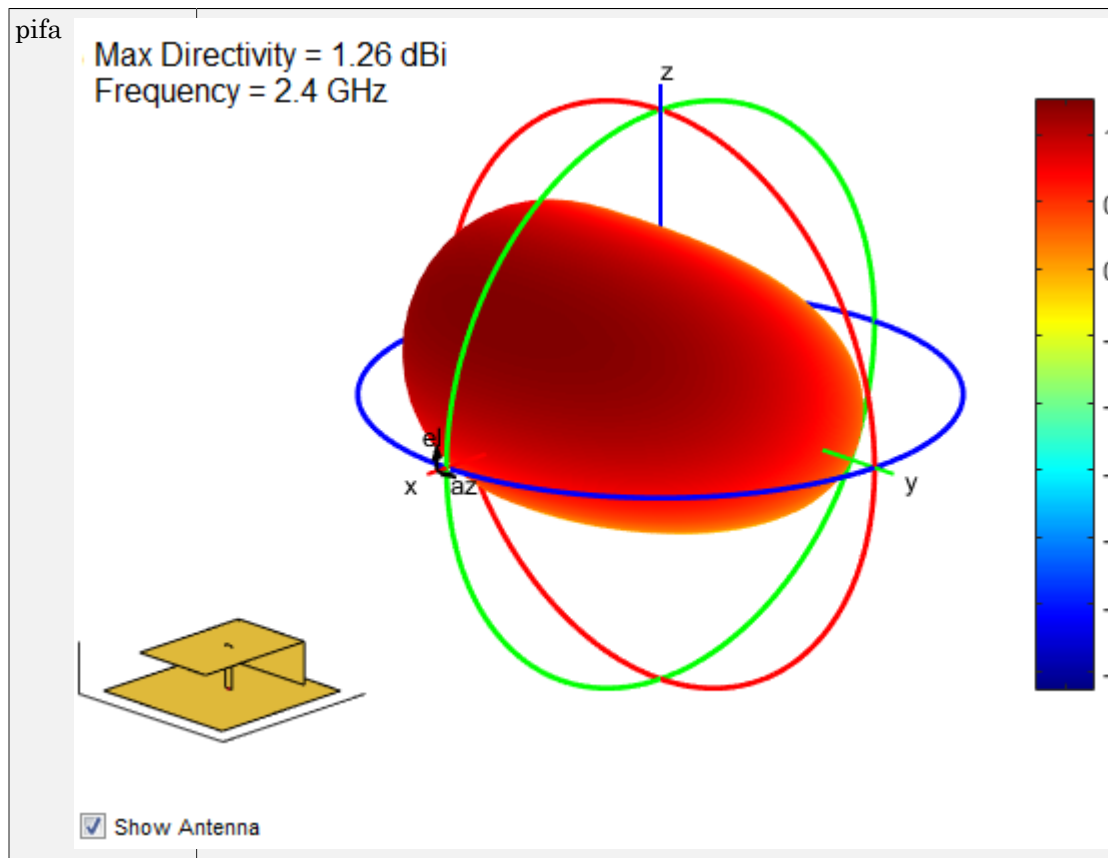


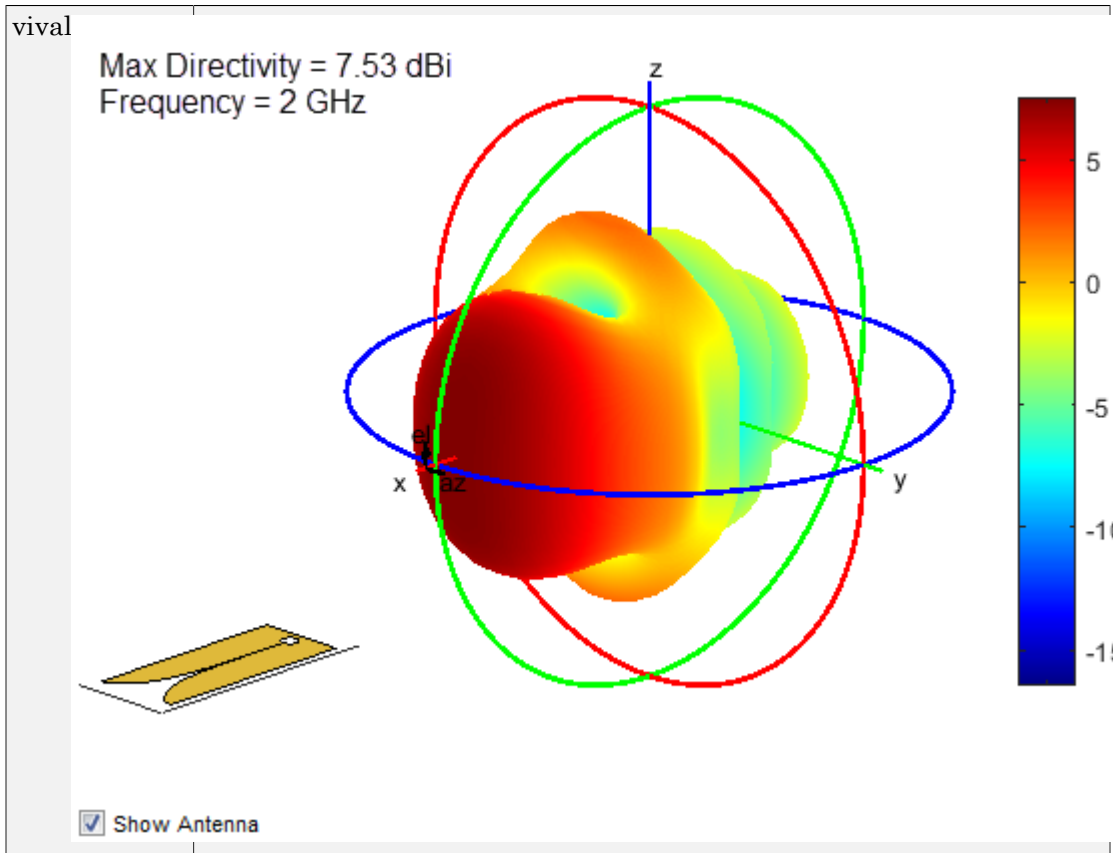


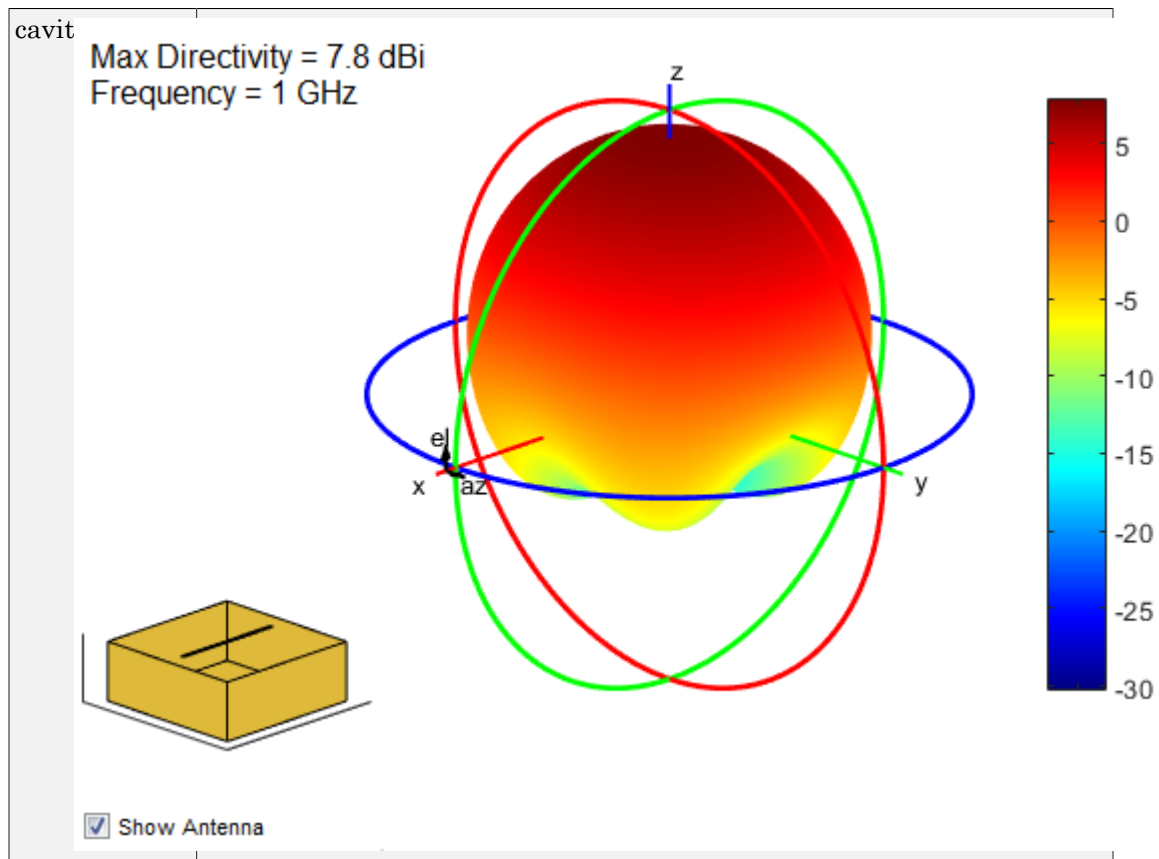


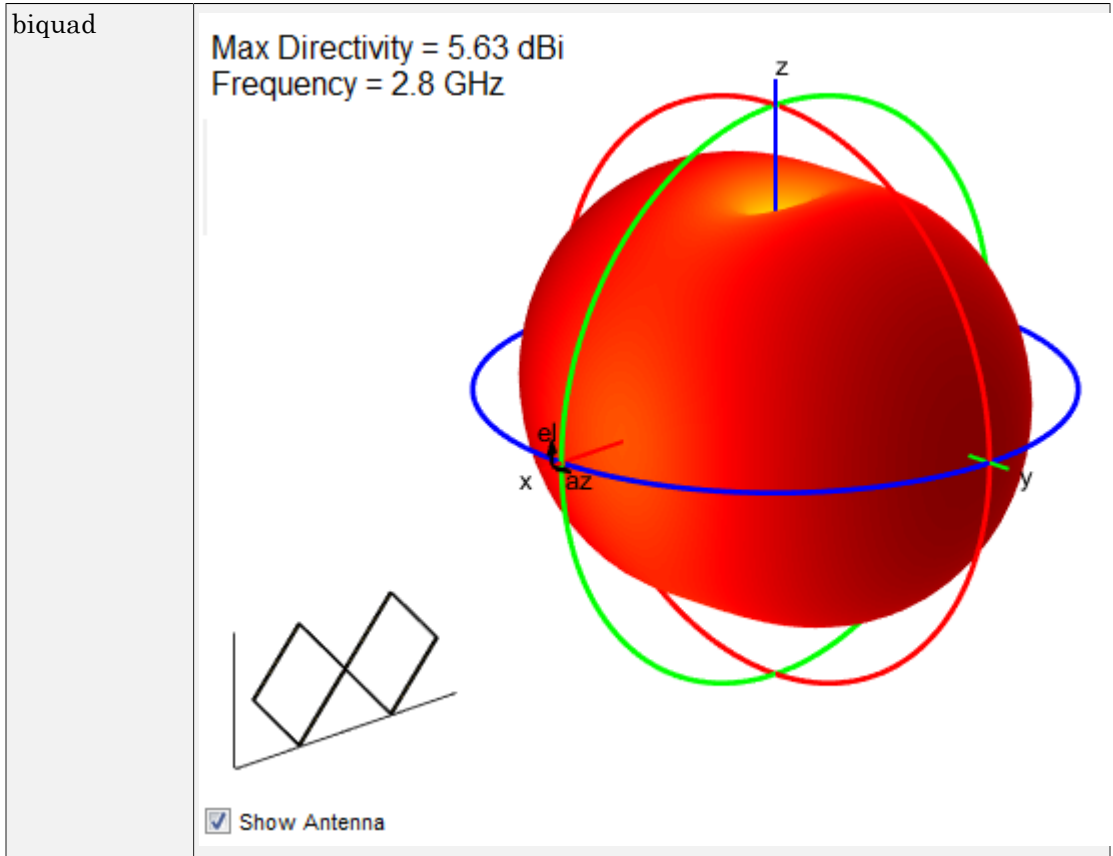


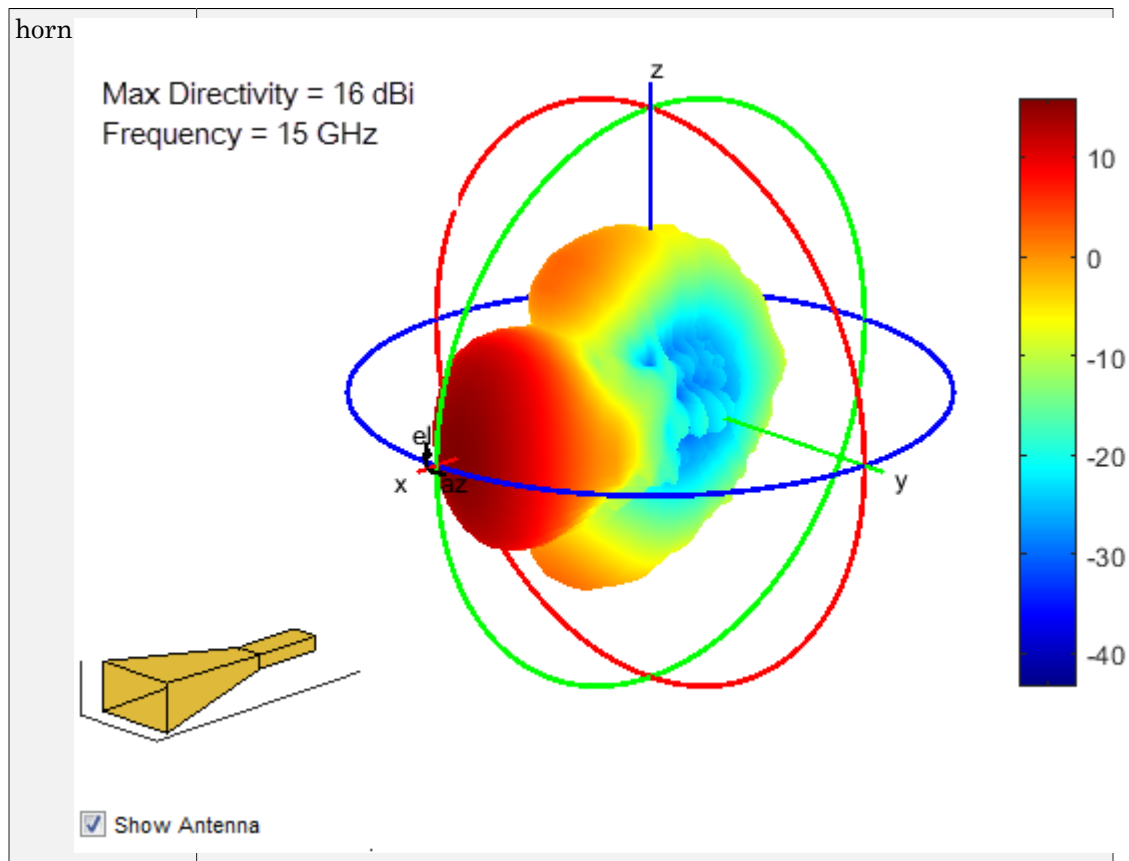


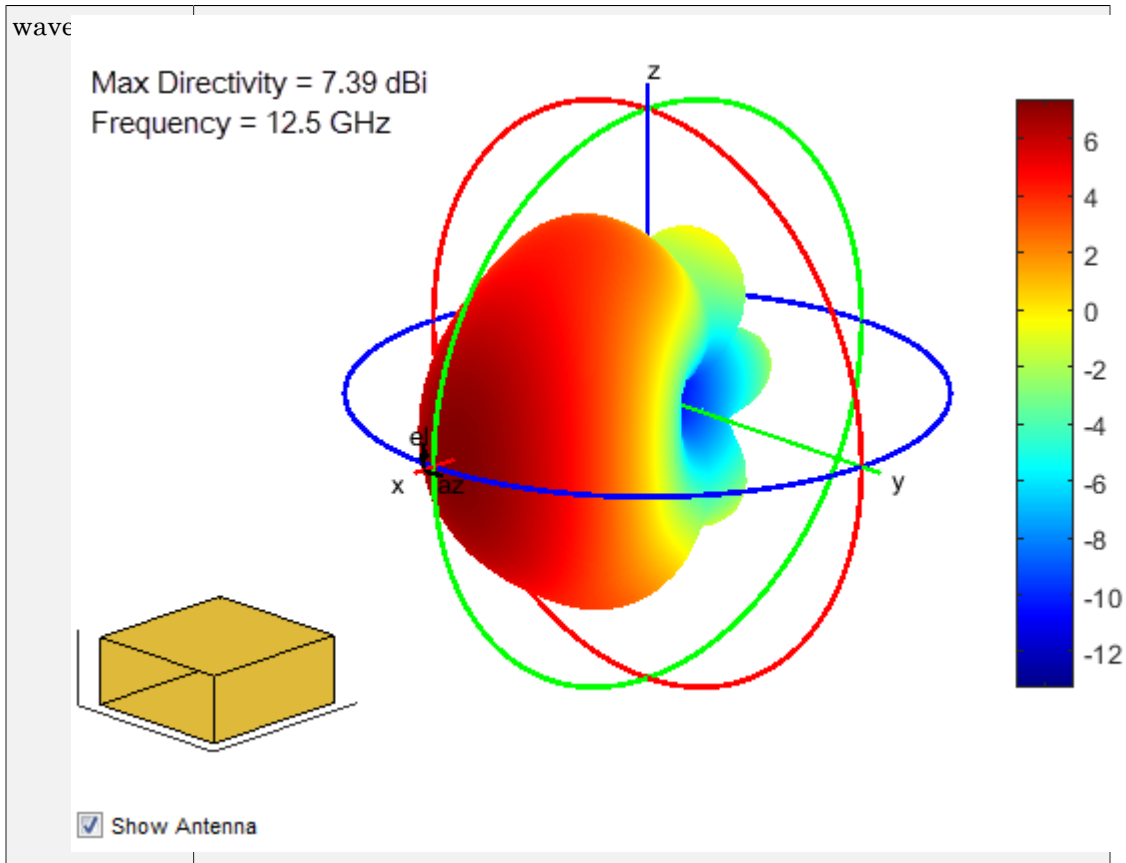












## Antenna Toolbox Limitations

The Antenna Toolbox does not support the following features.

### Antenna Library

This antenna library objects does not support:

- PIFA and inverted-F antennas with the infinite ground plane.
- Antenna analysis at frequencies less than 1 kHz or greater than 200 GHz.
- `pcbStack` cannot be used with any other antenna, array, or `planeWaveExcitation` elements in the library
- `pcbStack` using dielectrics - Only 1 and 2 layer metal with 1 layer being fully metallized is supported

### Array Library

This array library objects does not support:

- Arrays using slot, and cavity antennas.
- Reflector-based arrays using helix, slot, Vivaldi, cavity, PIFA antennas as exciters.
- Building arrays using antennas with tilted ground planes.
- Conformal arrays created using unbalanced antennas with infinite ground plane.
- Infinite arrays using dielectric materials.



## Interact with Polar Plot

This example shows how to interact with a polar plot created using `polarpattern` class.

### Create Polar Plot of Helix Antenna

Create a helix antenna that has a 28 mm radius, a 1.2 mm width and 4 turns.

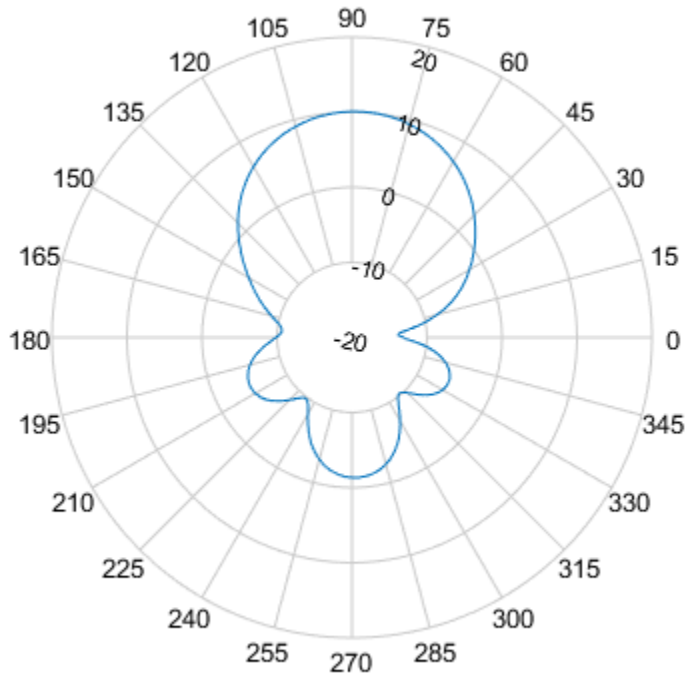
```
hx = helix('Radius',28e-3,'Width',1.2e-3,'Turns',4);
```

Calculate the directivity of the antenna at 1.8 GHz.

```
H = pattern(hx,1.8e9,0,0:1:360);
```

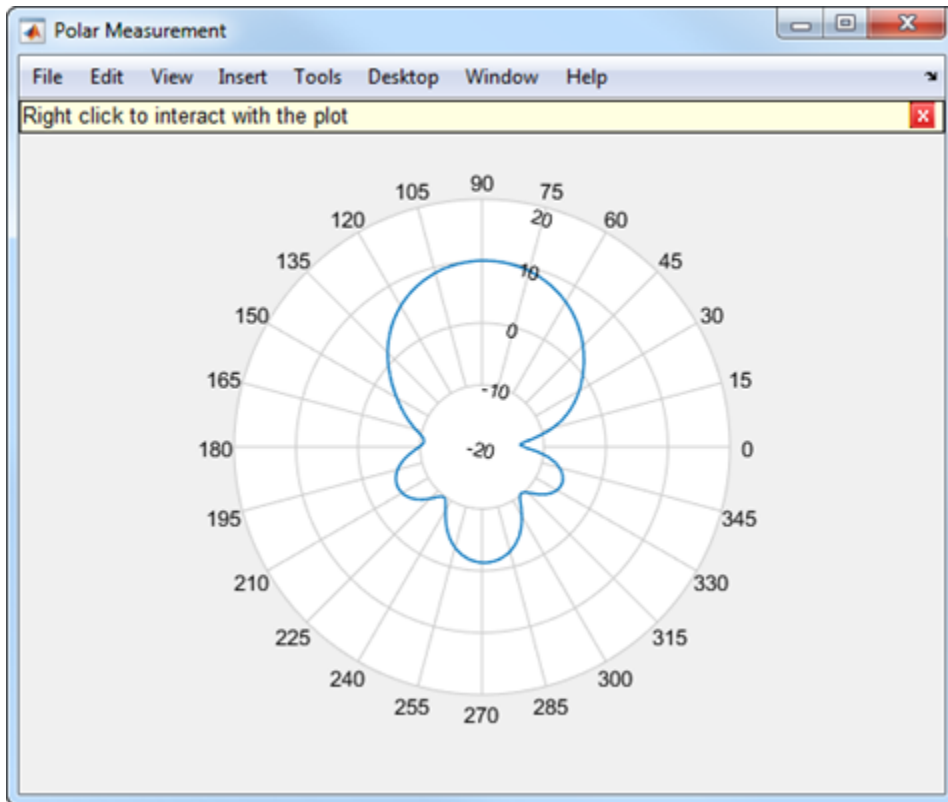
Display the polar plot of the antenna.

```
P = polarpattern(H);
```



### Interact with Polar Plot

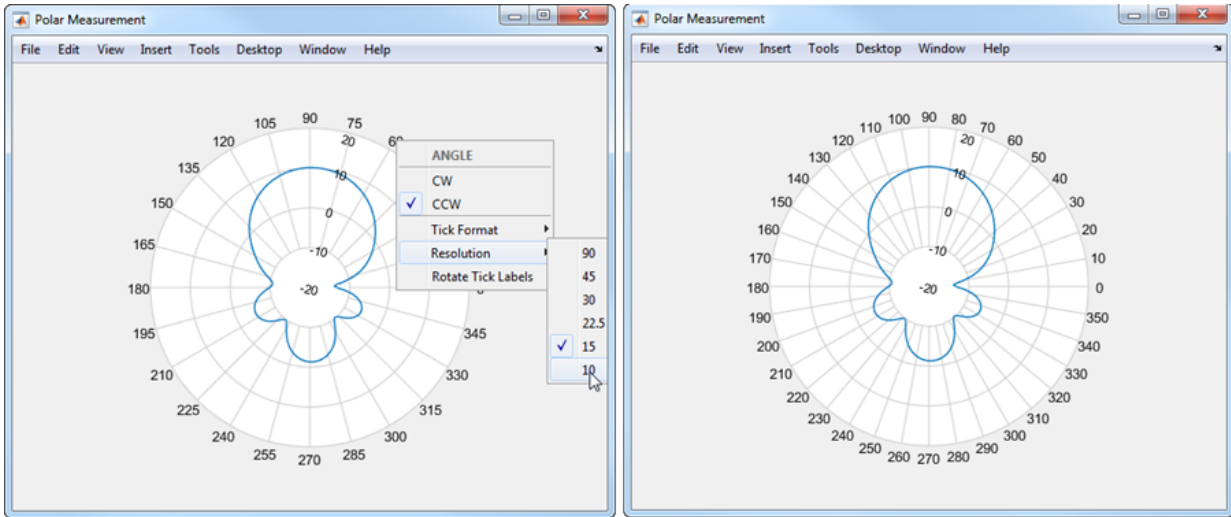
Hover over the plot. You see a message on top of the plot: **Right click to interact with the plot**. Right-click anywhere in the Polar Measurement window to display a context menu for interacting with the plot. For example, right-click outside the plot to show the **Main** context menu. Right-click inside the plot to show the **Display** context menu.



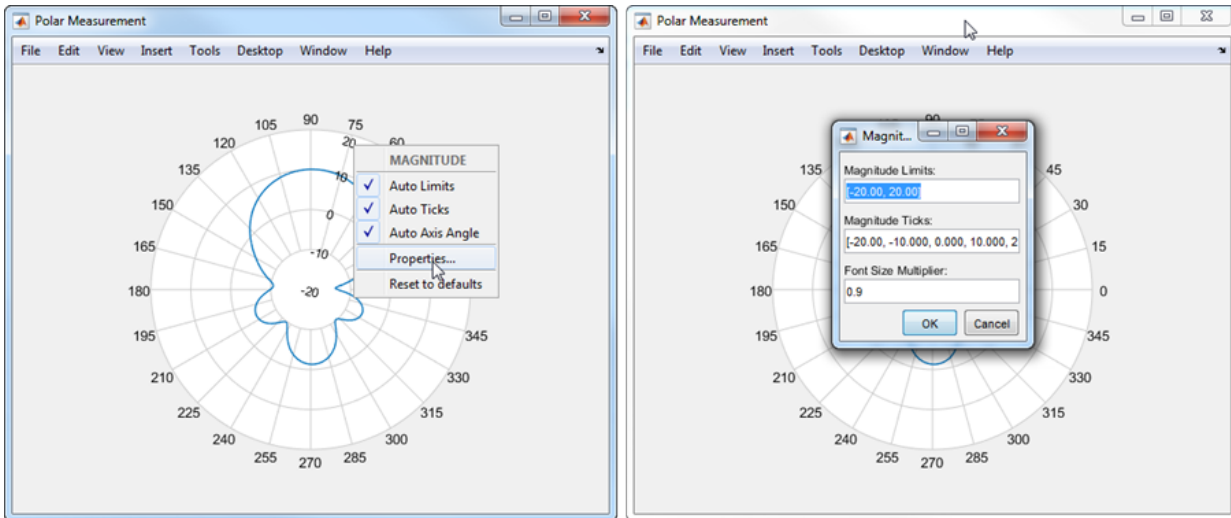
### Update Angle and Magnitude Values

The angular values are around the circumference of the polar plot. Right-click any of the angle values to open **ANGLE** context menu. By default, the angles are displayed CCW (counterclockwise).

Change the resolution from 15 degrees to 10 degrees.



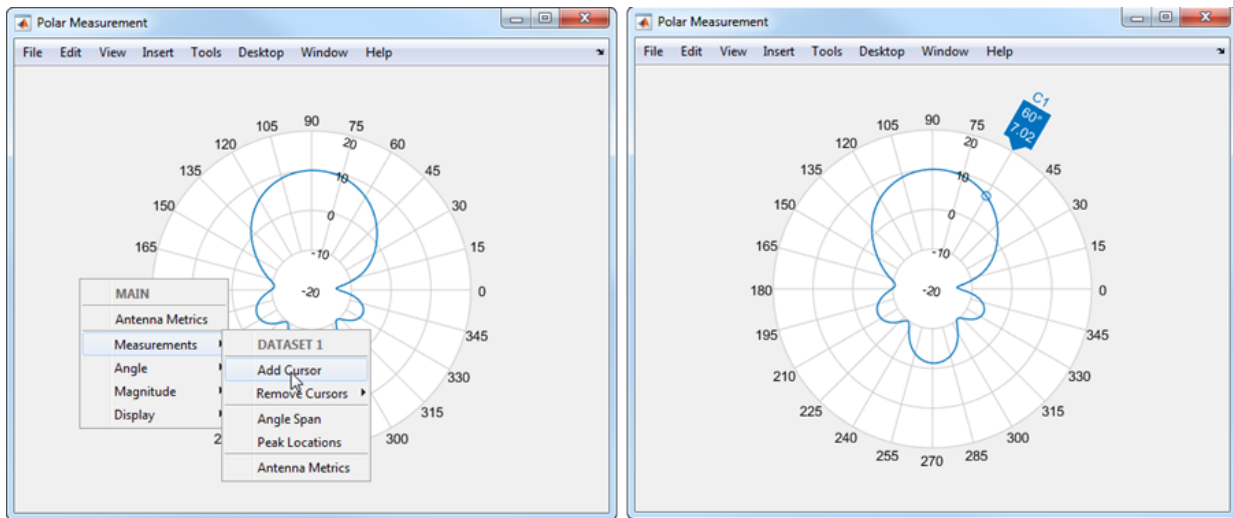
The magnitude values are on the radial lines of the plot. Right-click any of the magnitude values to open the **MAGNITUDE** context menu. Choose properties from the context menu to change the magnitude limits, magnitude ticks, or font size.



## Add Cursors and Calculate Angle Span

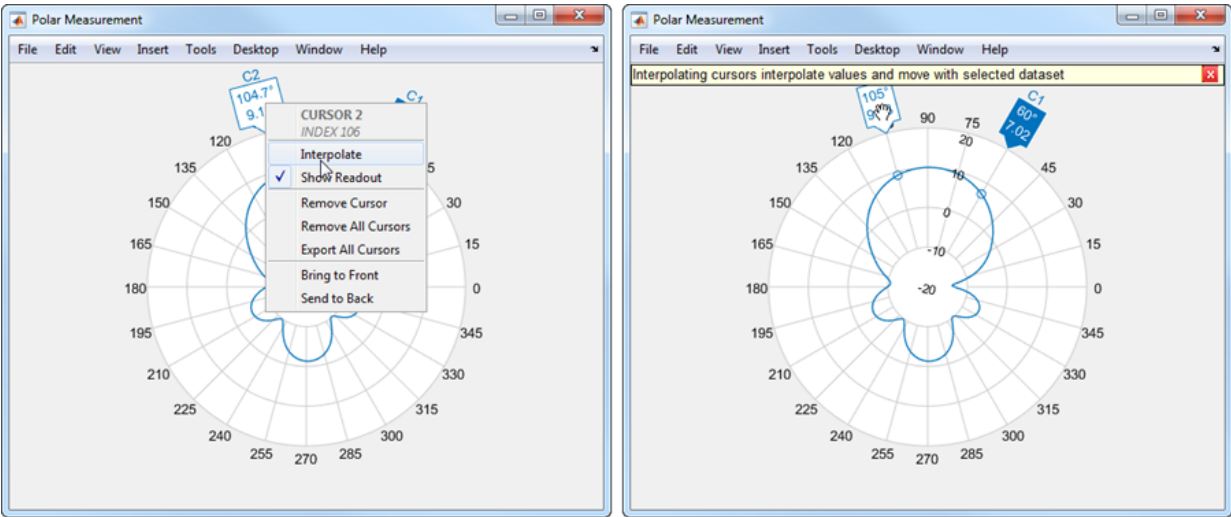
Add cursors at 60 degrees and 105 degrees.

To add a cursor from the **MAIN** or **DISPLAY** context menus, select **Measurements > Add Cursor**. After adding the cursor, place the mouse pointer on the cursor and drag it to 60 degrees.

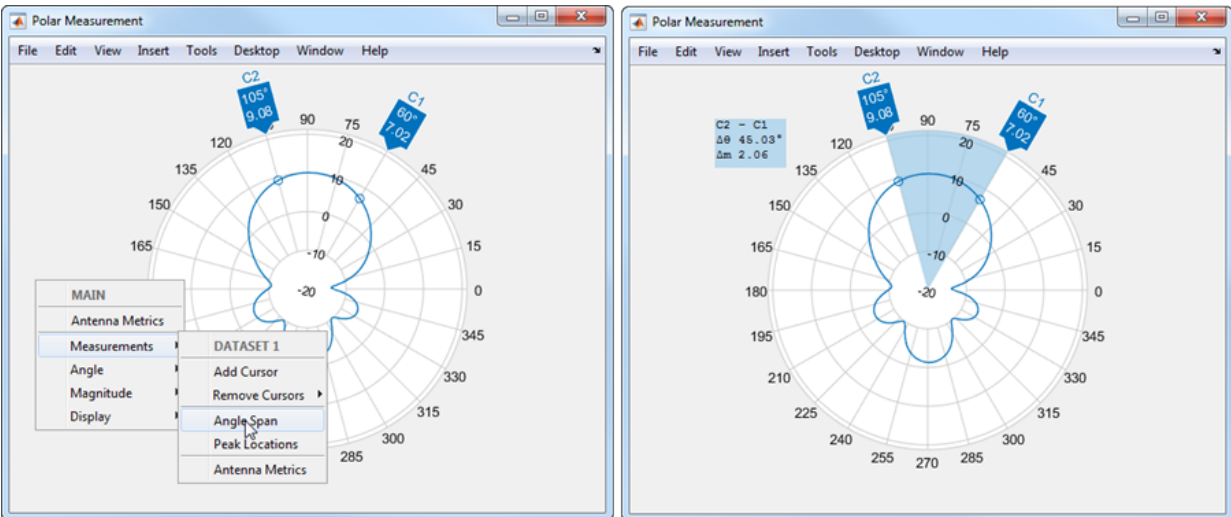


You can also add a cursor by double-clicking on the angle values. Double-click 105 to add a cursor. Right-click the newly added cursor and move the cursor to exact value of 105 degrees.

You can also interpolate the two angle values to 60 degrees and 150 degrees. Right-click on each cursor and choose **Interpolate** from the **CURSOR** context menu. To set the angle span, from the **MAIN** or **DISPLAY** context menu, select **Measurements > Angle Span**.

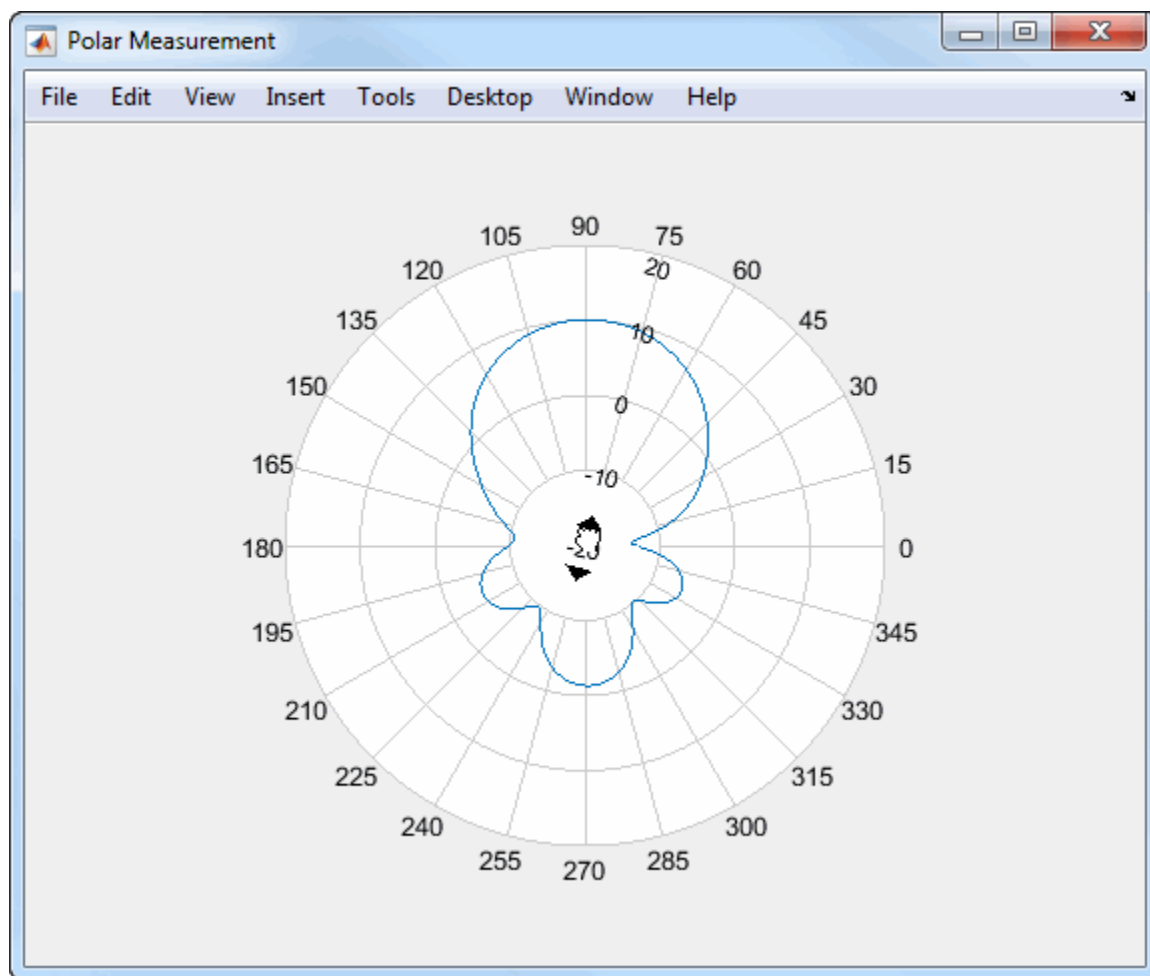


Calculate the counterclockwise angle span between 60 degrees and 105 degrees.



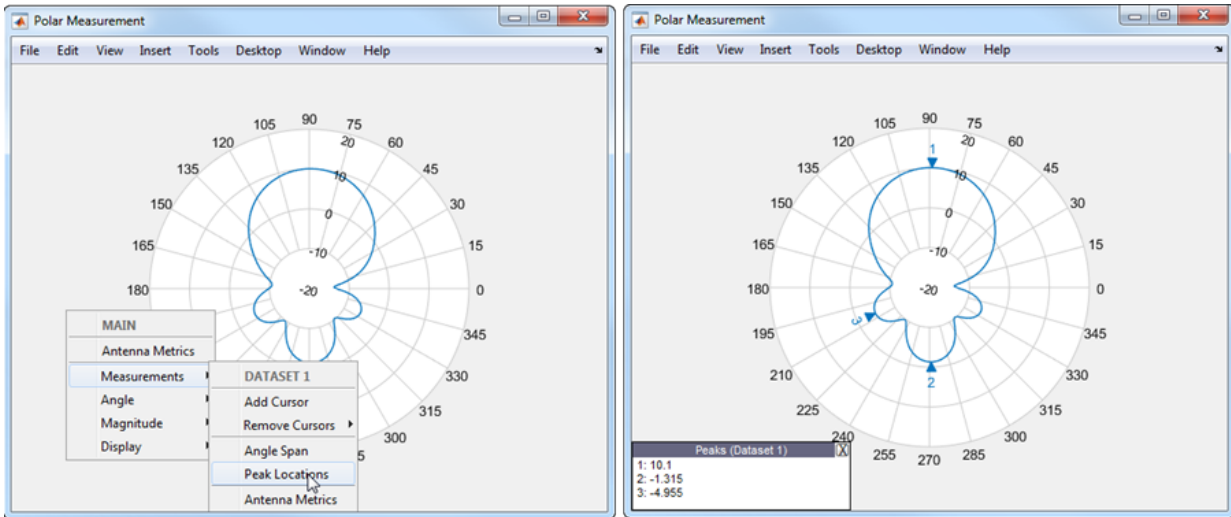
### Zoom In and Zoom Out

You can use the mouse pointer to zoom in and zoom out of the plot. Place the pointer at the center of the plot and drag radially outward.

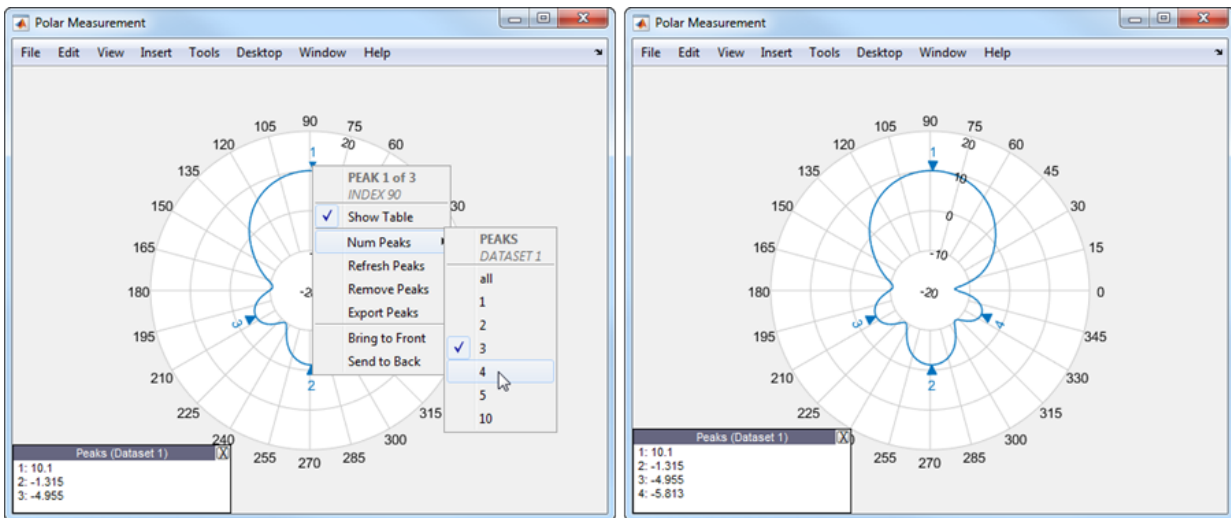


### Display Peak Locations

To display the peak locations from the **MAIN** or **DISPLAY** context menus, select **Measurements > Peak Location**.



Right-click any of the peak triangles and choose **NumPeaks**. Increase the number of peaks to 4.

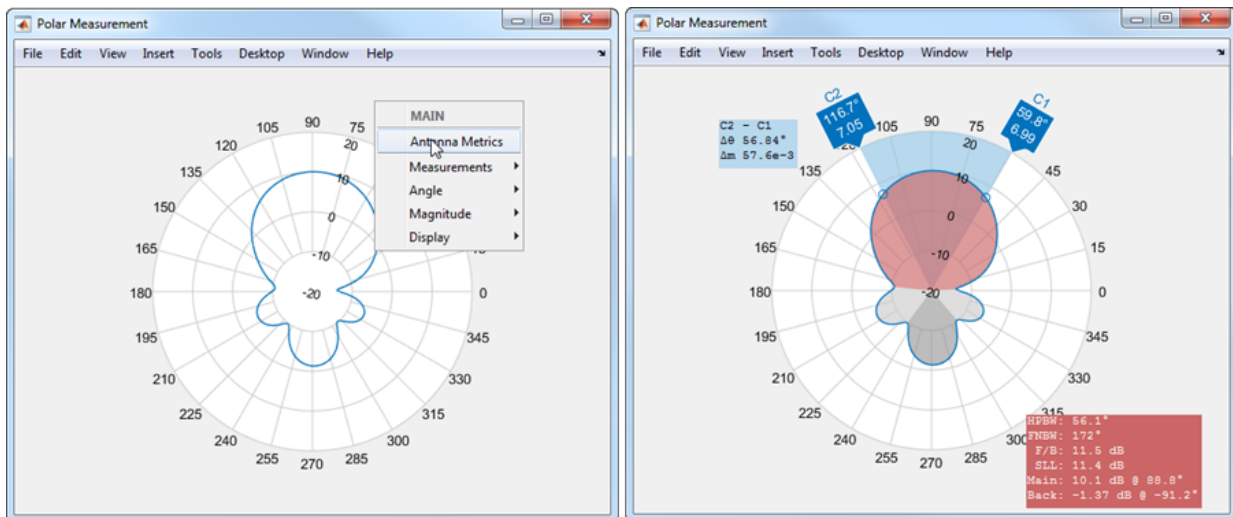




## View Antenna Metrics

Antenna metrics in the polar plot display the main, back, and side lobes of the antenna. There are two ways to turn display antenna metrics on the plot:

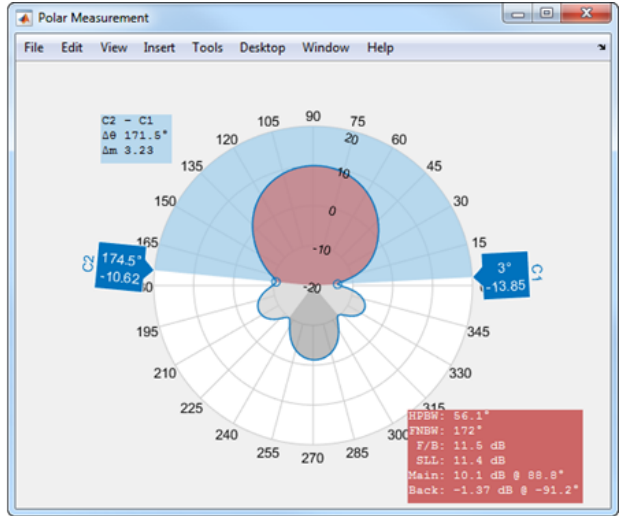
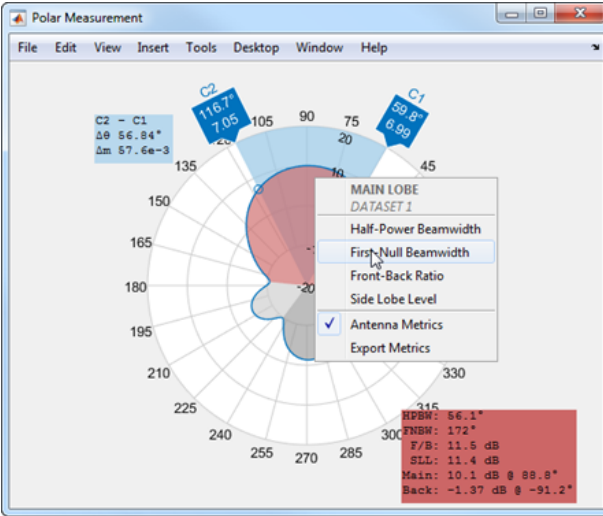
1. Right-click within the polar figure to open the **Main** context menu. Choose **Antenna Metrics**.
2. Right-click inside the polar plot to open the **Display** context menu. Inside the **Measurement** menu, choose Antenna Metrics.



By default, the plot shows the HPBW (half-power beamwidth) of the antenna. The antenna measurements text box displays:

- HPBW (half-power beamwidth)
- FNBW (first-null beamwidth)
- F/B (front-to-back ratio)
- SLL (side lobe level)
- Main (main lobe peak value and corresponding angle)
- Back (back lobe peak value and corresponding angle)

To view the FNBW, right-click inside the red or gray polar plot region to open the **MAIN LOBE** or the **BACK LOBE** context menu and then choose **First-Null Beamwidth**.



## Antenna Design and Analysis Using Antenna Designer App

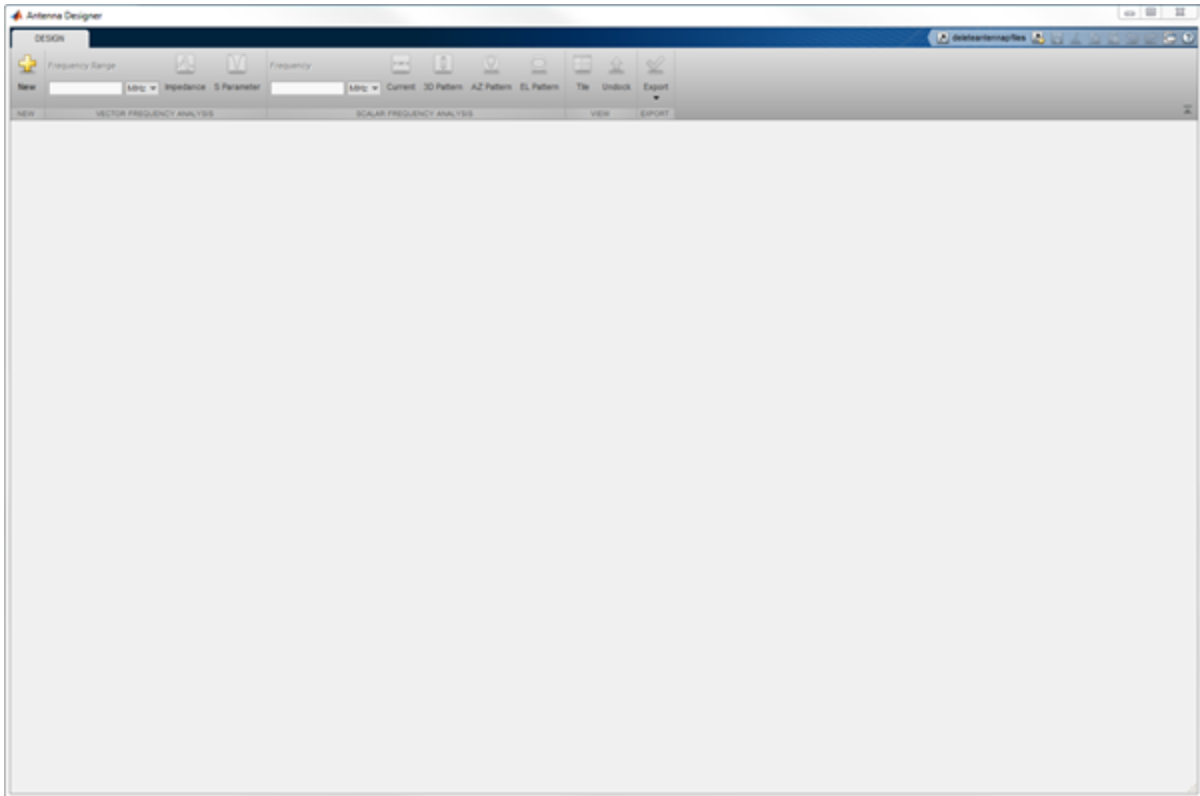
This example shows how to construct, visualize, and analyze antenna elements using the **Antenna Designer** App.

### Open Antenna Designer App

To open the app, at the MATLAB command prompt, enter: command line .

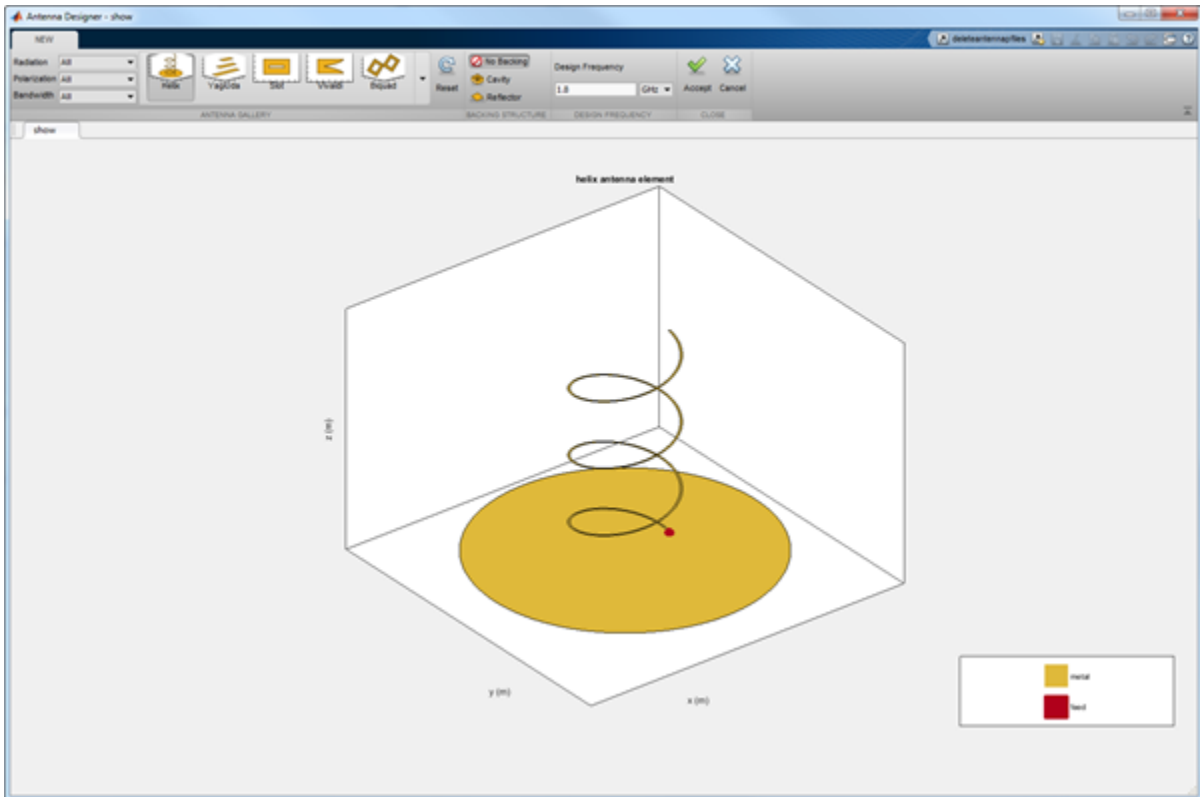
```
antennaDesigner
```

This command opens a blank canvas.



## Design Helix Antenna

In the blank canvas, click **NEW**. In the **ANTENNA GALLERY**, under **OTHERS**, select a helix antenna. Set the **Design Frequency** to 1.8 GHz.



To analyze the helix antenna, click **Accept**.

### Change Helix Antenna Properties

In the **Antenna Properties** tab, change the following:

- **Radius** = 0.0280
- **Width** = 0.0012
- **Turns** = 4

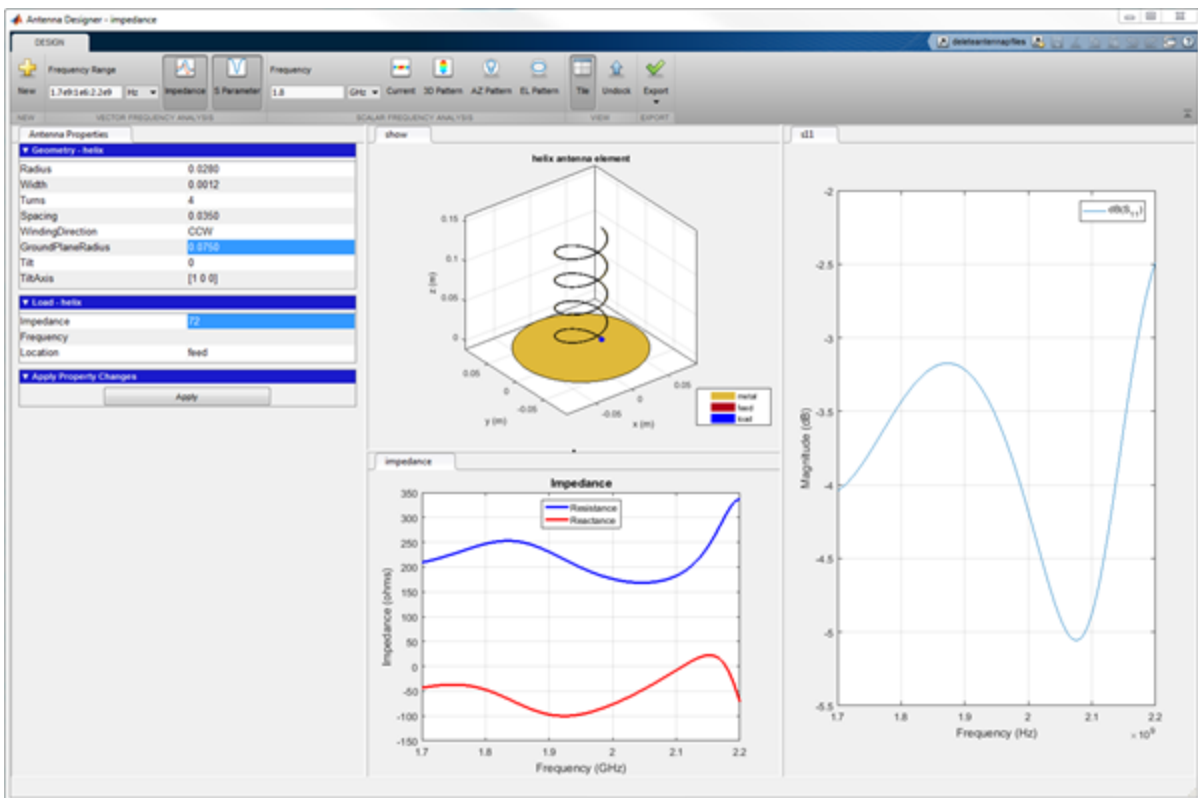
- **Spacing** = 0.0350
- **GroundPlaneRadius** = 0.0750

Click **Apply** to see the change in the helix antenna structure.

### Plot Impedance and S-Parameters

Open the **Load-helix** section and change the **Impedance** of the antenna to 72 ohms. Click **Apply**. In the toolbar, under **VECTOR FREQUENCY ANALYSIS** tab, change the **Frequency Range** to  $1.7e9:1e6:2.2e9$  Hz .

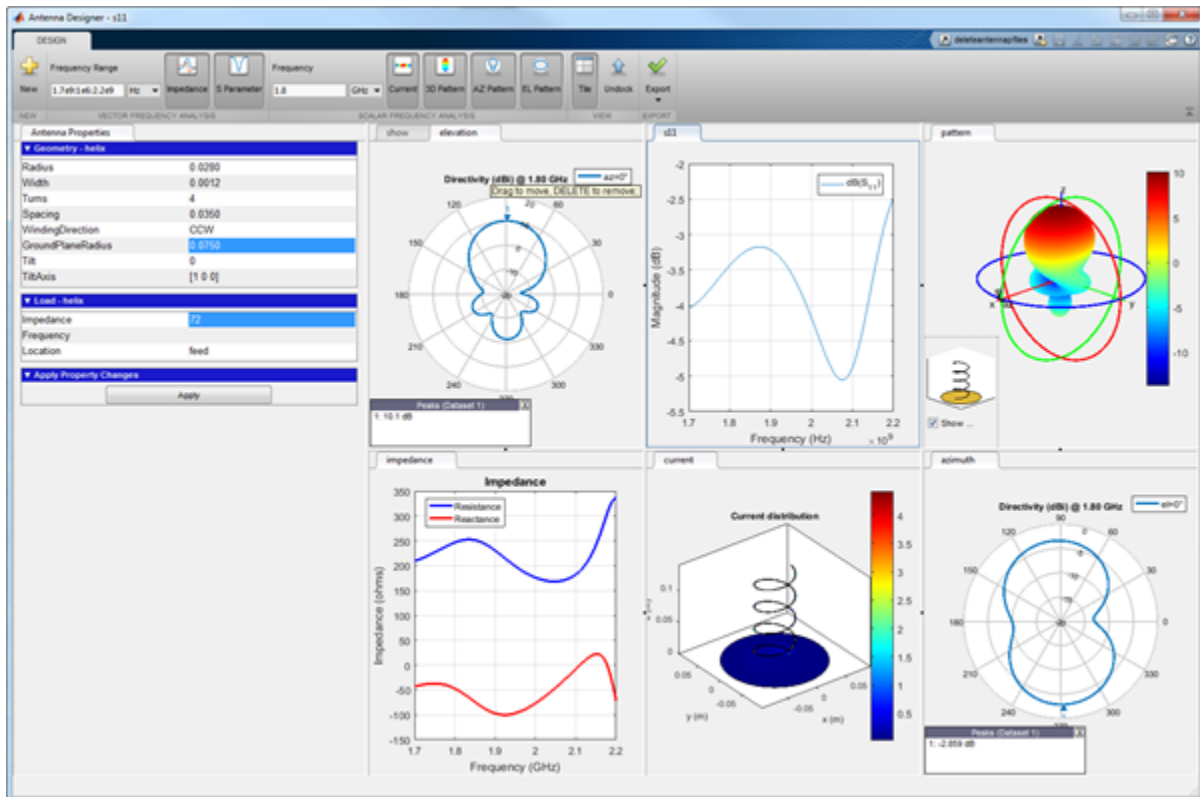
Click **Impedance** to plot the impedance of the helix antenna. Click **S Parameter** to plot the S11 value of the helix antenna. Click **Tile** to view the plots together.



## Plot Current Distribution, 3-D, Azimuth, and Elevation Patterns

In the **SCALAR FREQUENCY ANALYSIS** section of the toolstrip, click **Current** to view the current distribution of the helix at 1.8 GHz.

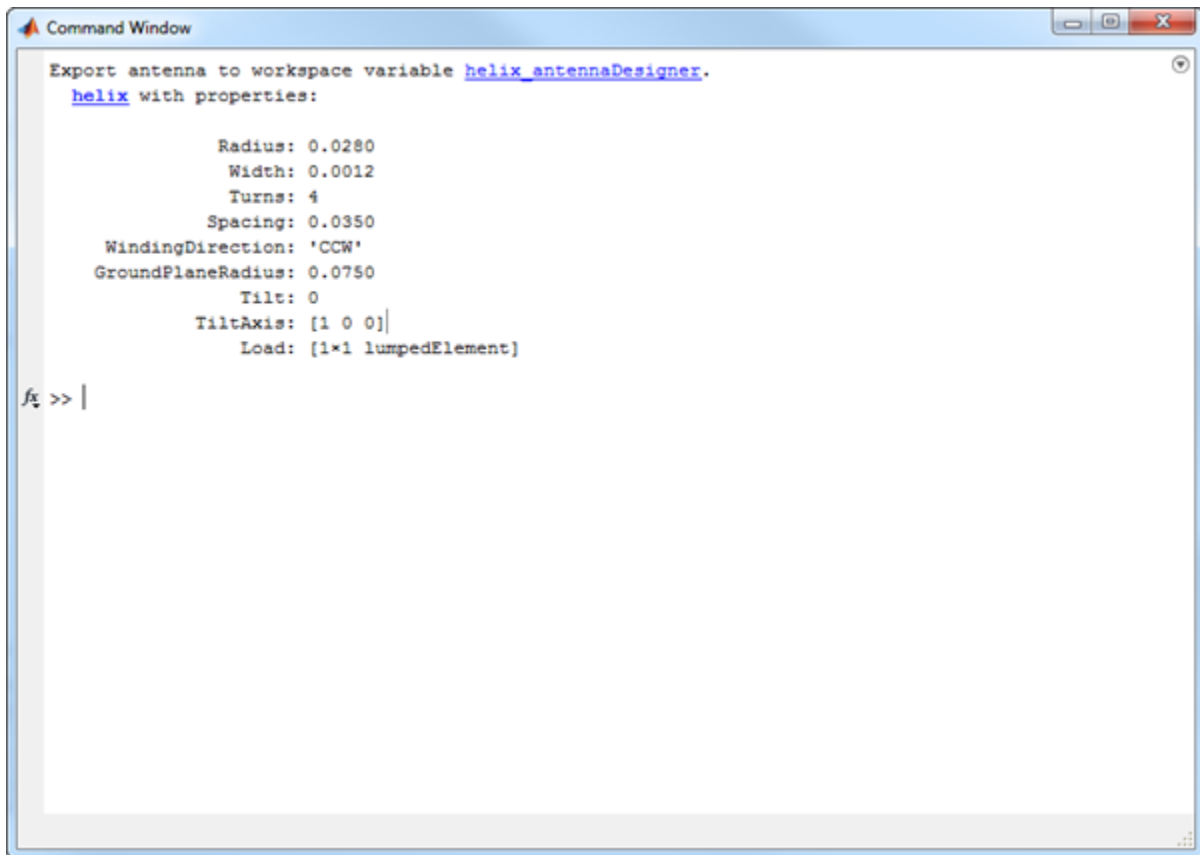
Click **3D Pattern**, **AZ Pattern**, and **EL Pattern** to view the radiation, azimuth, and elevation patterns of the helix antenna, respectively. Click **Tile** again to view all the plots together.



You can compare the results of this tutorial with the results of Antenna Modeling and Analysis tutorial: “Antenna Modeling and Analysis”

### Export to MATLAB Workspace

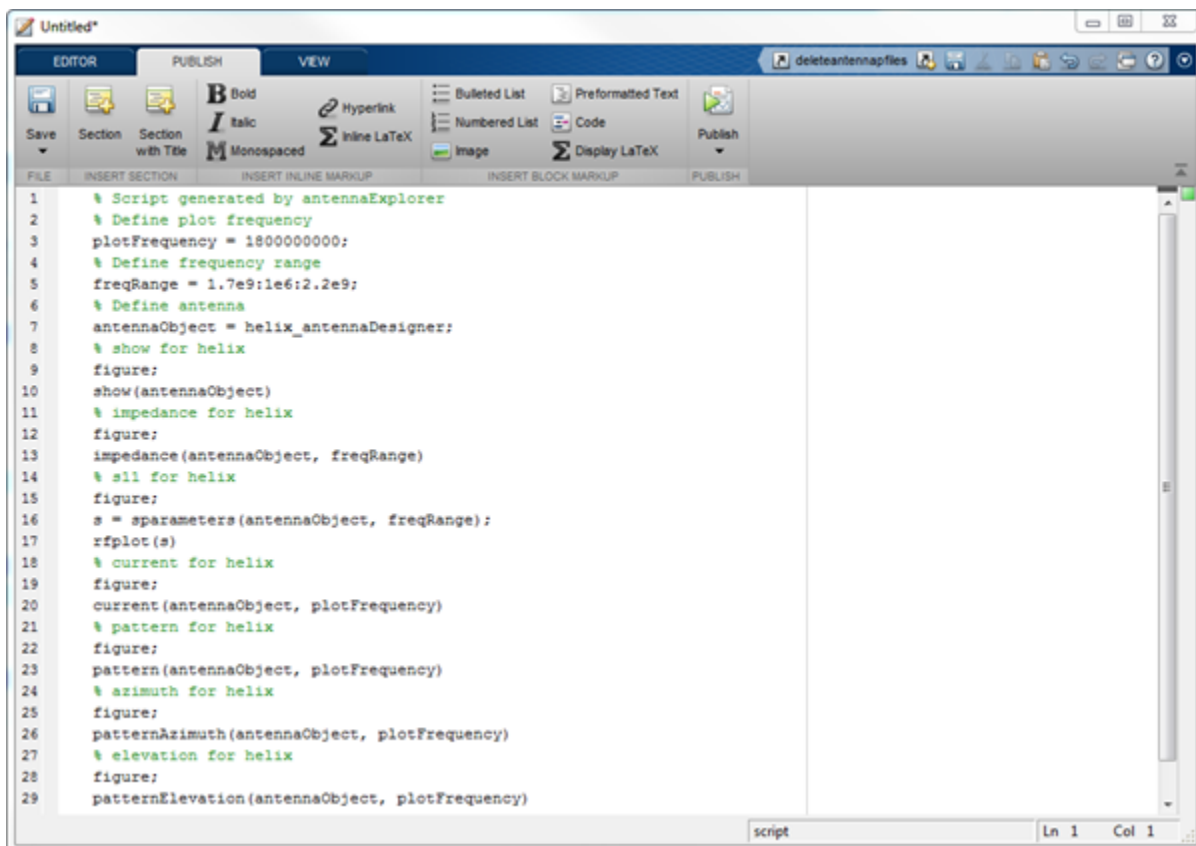
Click the **Export** button arrow and then click **Export to workspace** to view the helix antenna as a variable in the MATLAB command line. The variable name is `helix_antennaDesigner`. Click on the variable in the workspace to view the properties of the helix antenna.



```
Command Window
Export antenna to workspace variable helix_antennaDesigner.
helix with properties:
    Radius: 0.0280
    Width: 0.0012
    Turns: 4
    Spacing: 0.0350
    WindingDirection: 'CCW'
    GroundPlaneRadius: 0.0750
    Tilt: 0
    TiltAxis: [1 0 0]
    Load: [1x1 lumpedElement]
fx >> |
```

### Export to MATLAB Script

Click the **Export** button arrow again and then click **Export to script** to view the helix antenna and analysis in MATLAB script format.





## Design Variations On Microstrip Patch Antenna Using PCB Stack

Use the `pcbstack` to design basic, parasitic, direct-coupled, and CP patch antennas.

Setup parameters.

```
vp = physconst('lightspeed');  
f = 850e6;  
lambda = vp./f;
```

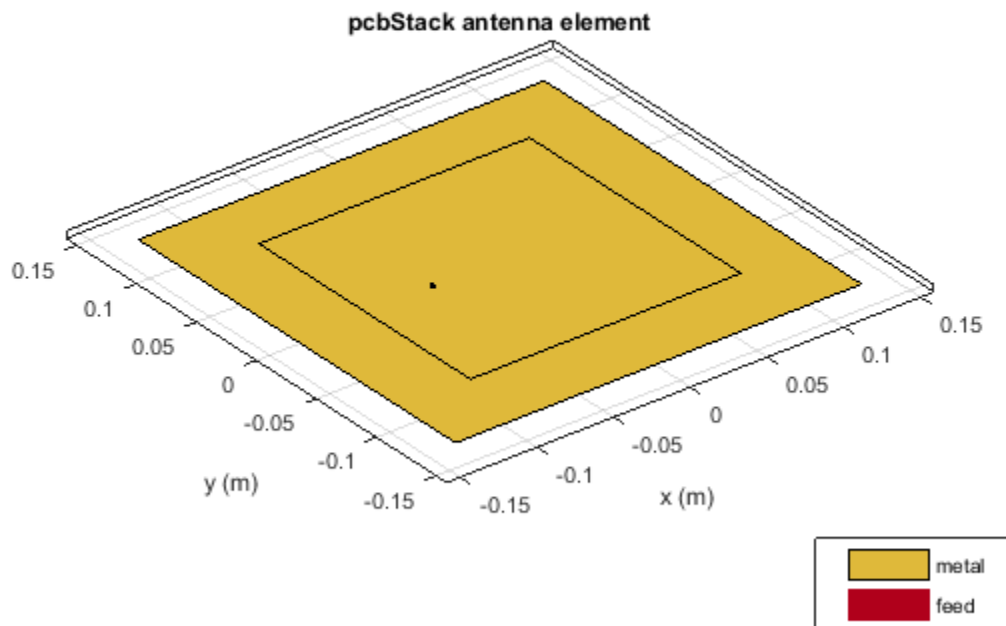
### Design Basic Patch Antenna

Set the length and width of the patch and the groundplane.

```
Lp = lambda(1)/2;  
Wp = lambda(1)/2;  
Lgp = 0.75.*lambda(1);  
Wgp = 0.75.*lambda(1);  
h = 2.e-3;  
p1 = antenna.Rectangle('Length',Lp,'Width',Wp,'NumPoints',20);  
p2 = antenna.Rectangle('Length',Lgp,'Width',Wgp);  
d1 = dielectric('Air');
```

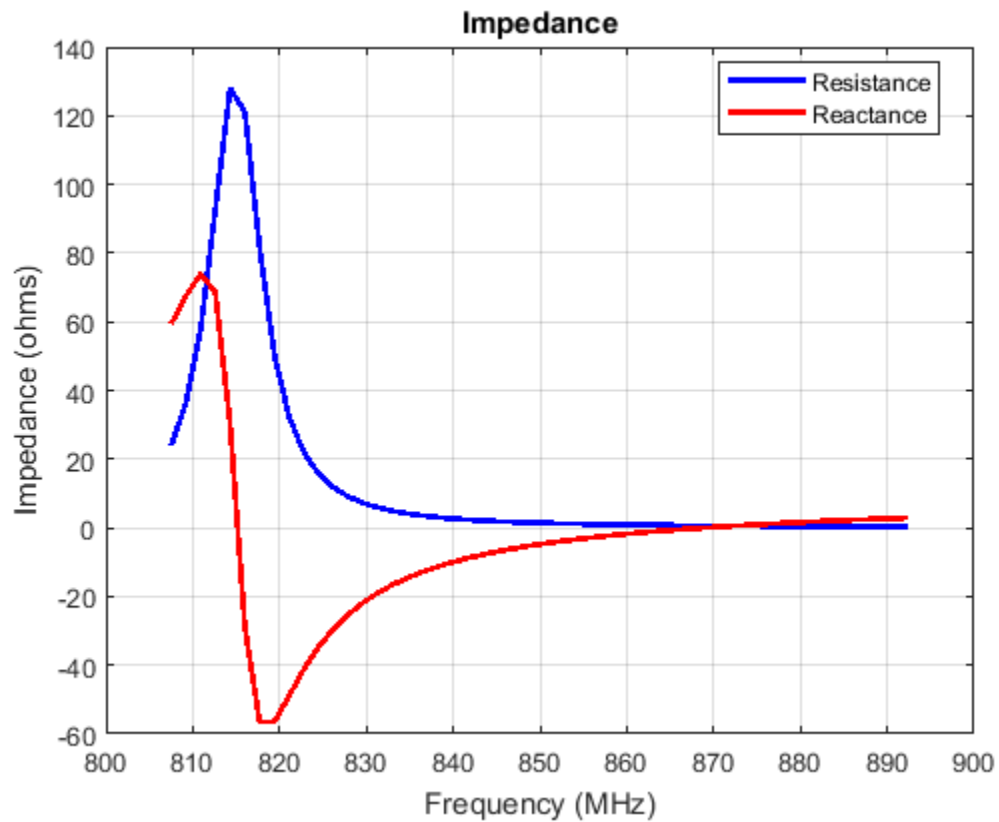
Define the properties of the PCB stack.

```
basicPatch = pcbStack;  
basicPatch.Name = 'Basic Patch';  
basicPatch.BoardThickness = h;  
basicPatch.BoardShape = p2;  
basicPatch.Layers = {p1,d1,p2};  
basicPatch.FeedLocations = [-lambda(1)/8 0 1 3];  
figure  
show(basicPatch)
```



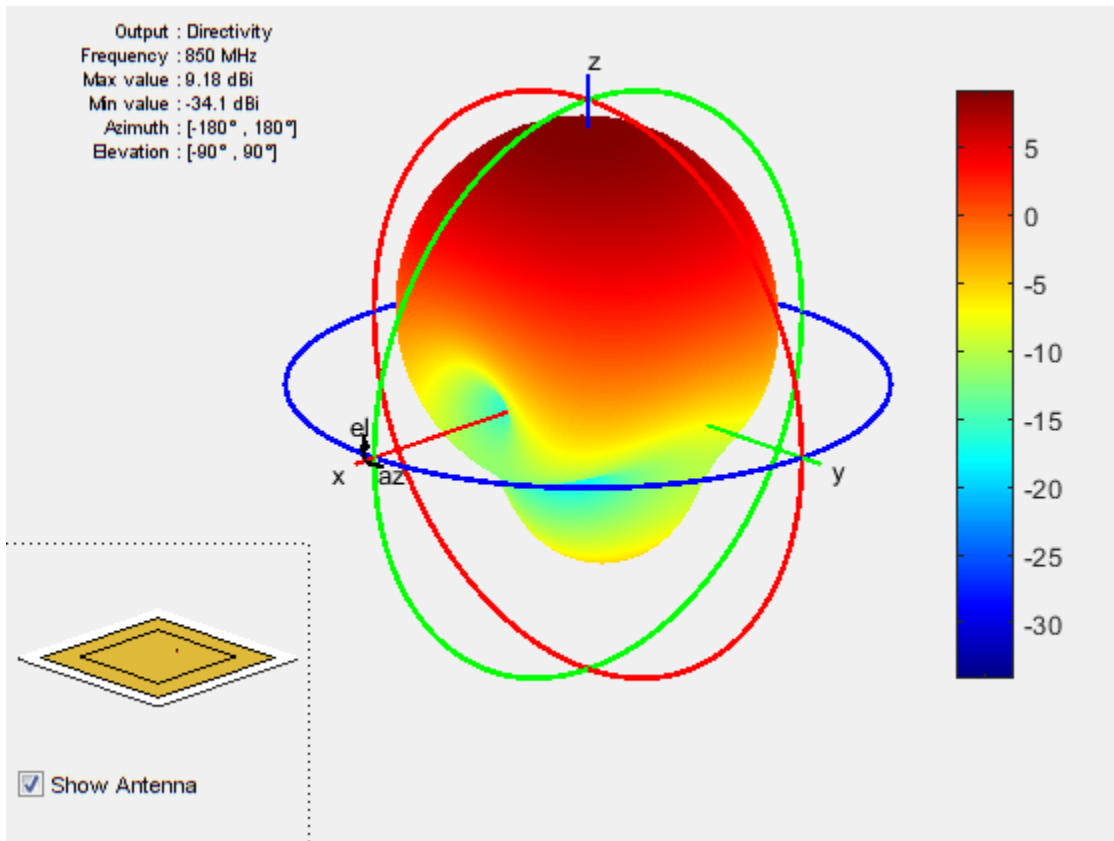
Plot the impedance of the basic patch antenna.

```
freq1 = linspace(f(1)-0.05*f(1),f(1) + 0.05*f(1),51);  
figure  
impedance(basicPatch,freq1)
```



Plot the radiation pattern of the basic patch antenna.

```
figure  
pattern(basicPatch,f(1))
```

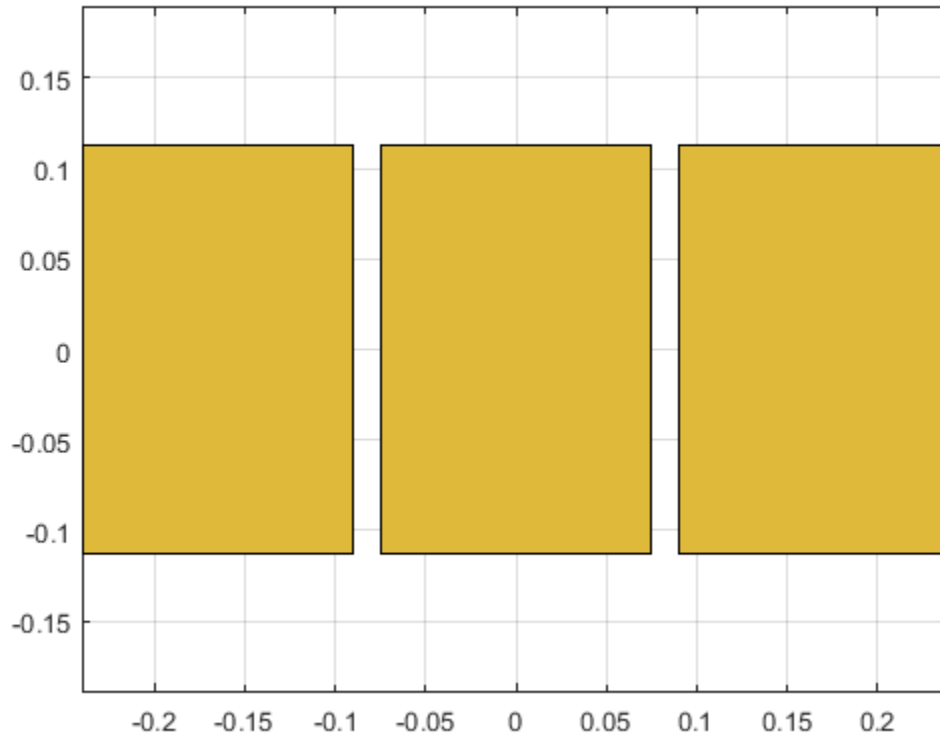


### Design Parasitic Patch Antenna

Set the dimensions for the patch.

```
L = 0.15;
W = 1.5*L;
stripL = L;
gapx = .015;
gapy = .01;
r1 = antenna.Rectangle('Center',[0,0],'Length',L,'Width',W);
r2 = antenna.Rectangle('Center',[L/2+stripL/2+gapx,0],'Length',stripL,'Width',W,'NumPorts',1);
r3 = antenna.Rectangle('Center',[-L/2-stripL/2-gapx,0],'Length',stripL,'Width',W,'NumPorts',1);
r = r1+r2+r3;
```

```
figure  
show(r)
```



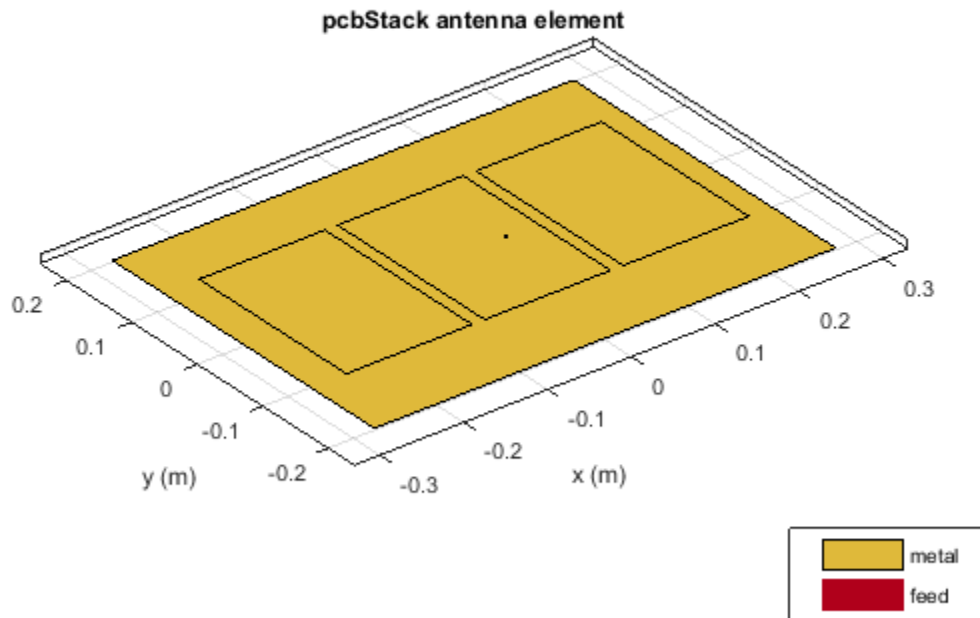
Set the dimensions of the groundplane.

```
Lgp = 0.55;  
Wgp = 0.4;  
g1 = antenna.Rectangle('Center',[0,0], 'Length',Lgp, 'Width',Wgp);
```

Define the properties of the PCB stack. Create a pcb stack by driving the center radiator.

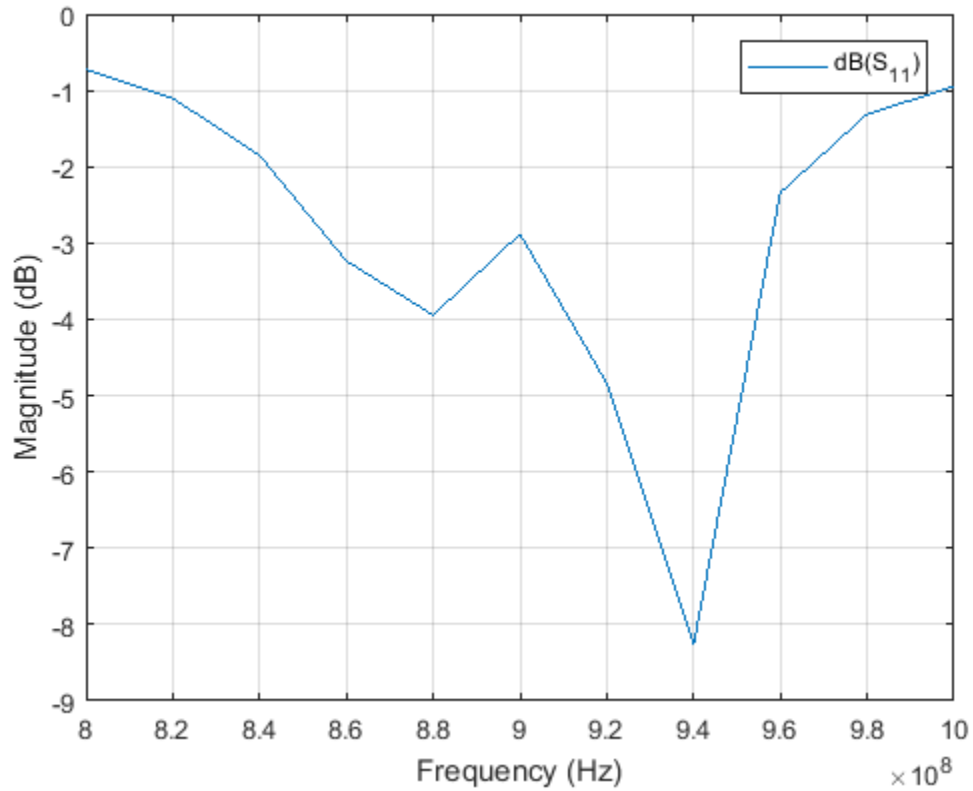
```
parasitic_patch = pcbStack;  
parasitic_patch.BoardShape = g1;
```

```
parasitic_patch.BoardThickness = .007;  
parasitic_patch.Layers = {r,g1};  
parasitic_patch.FeedLocations = [(L)/4 0 1 2];  
figure  
show(parasitic_patch)
```



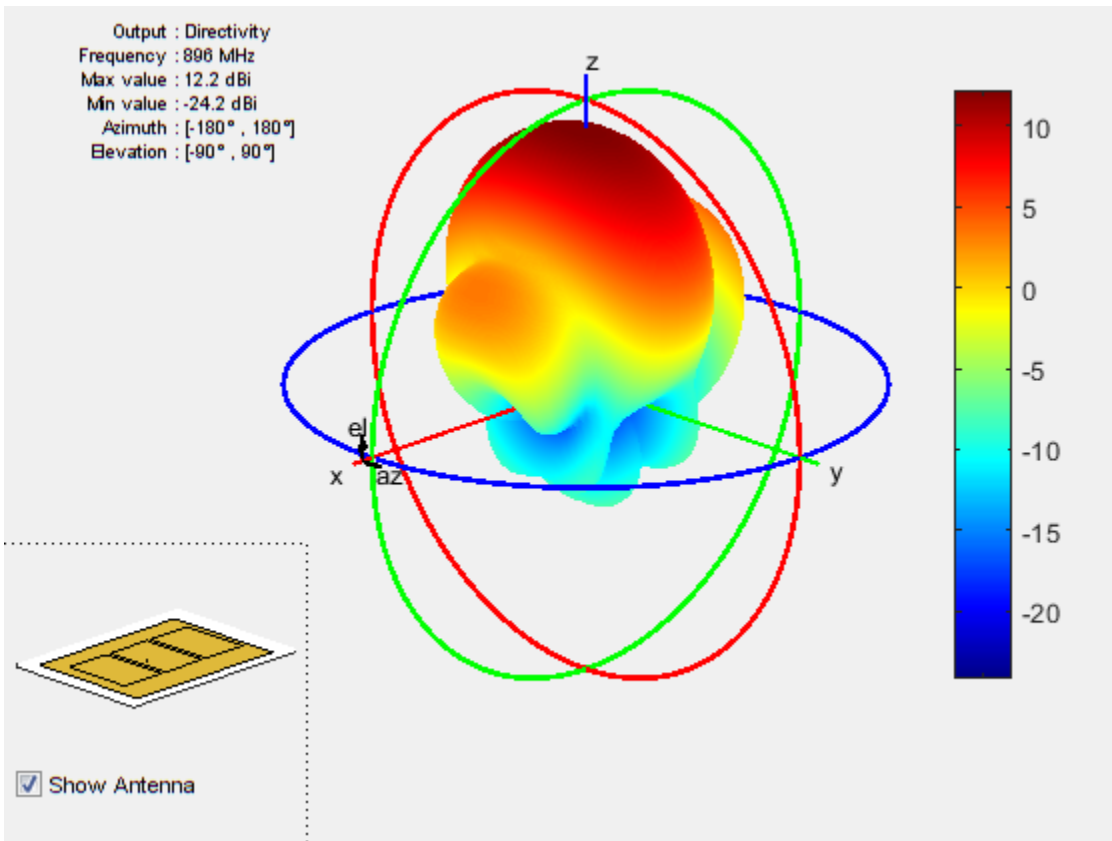
Plot the S-parameters of the parasitic patch antenna.

```
s = sparameters(parasitic_patch,linspace(0.8e9,1e9,11));  
figure  
rfplot(s)
```



Plot the radiation pattern of the parasitic patch antenna.

```
figure  
pattern(parasitic_patch,0.896e9)
```

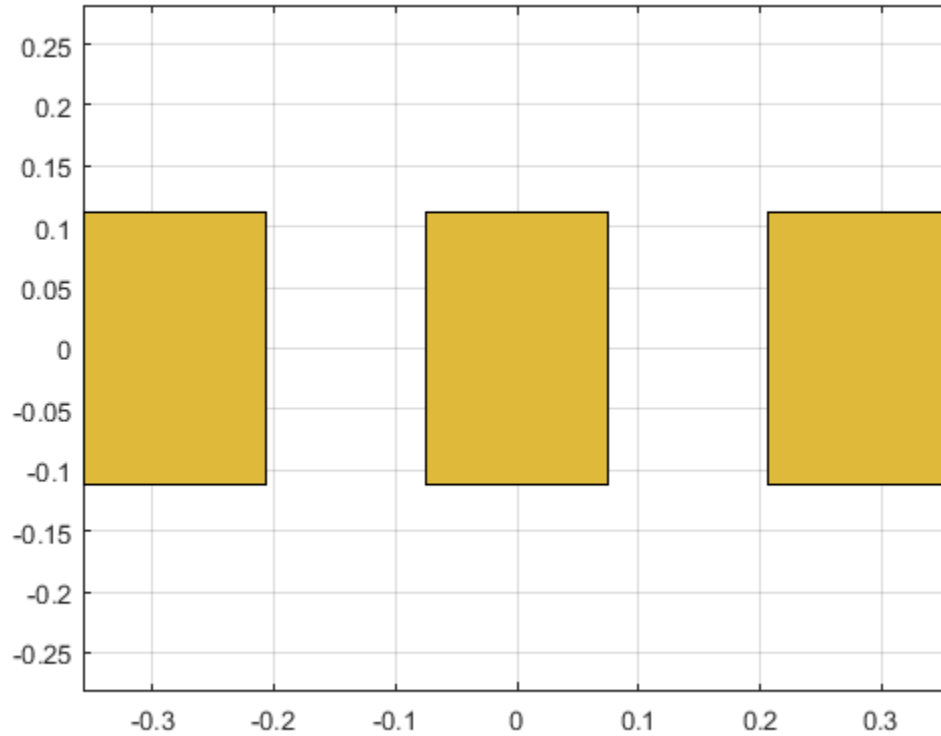


### Design Direct-coupled Patch Antenna

```

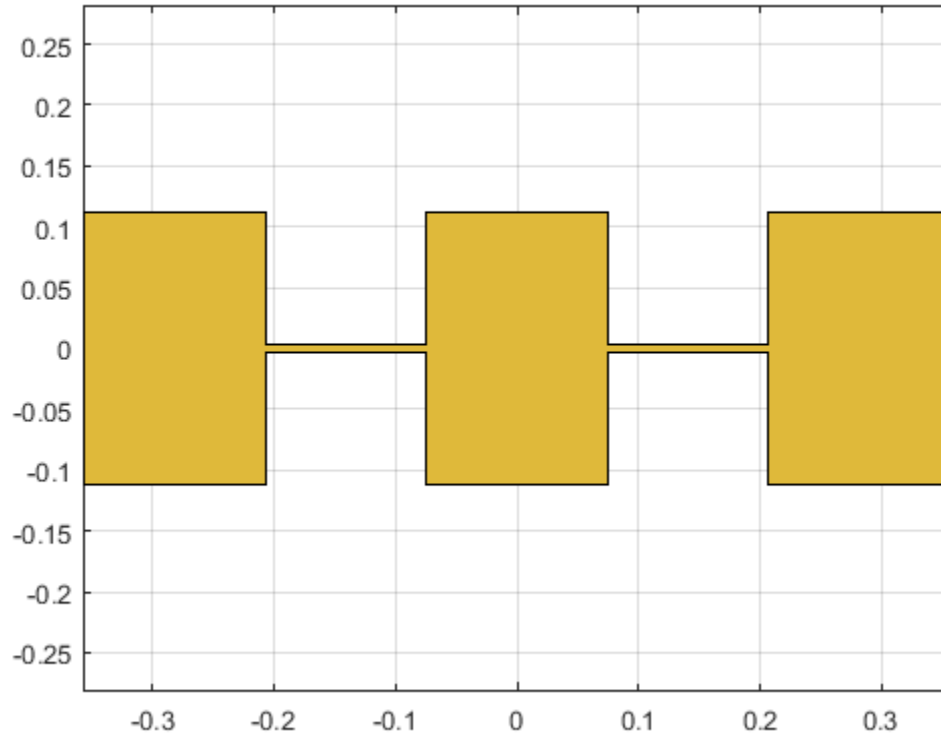
r2 = copy(r1);
r2.Center = [lambda/1.25,0];
r3 = copy(r1);
r3.Center = [-lambda/1.25,0];
r = r1+r2+r3;
figure
show(r)
    
```





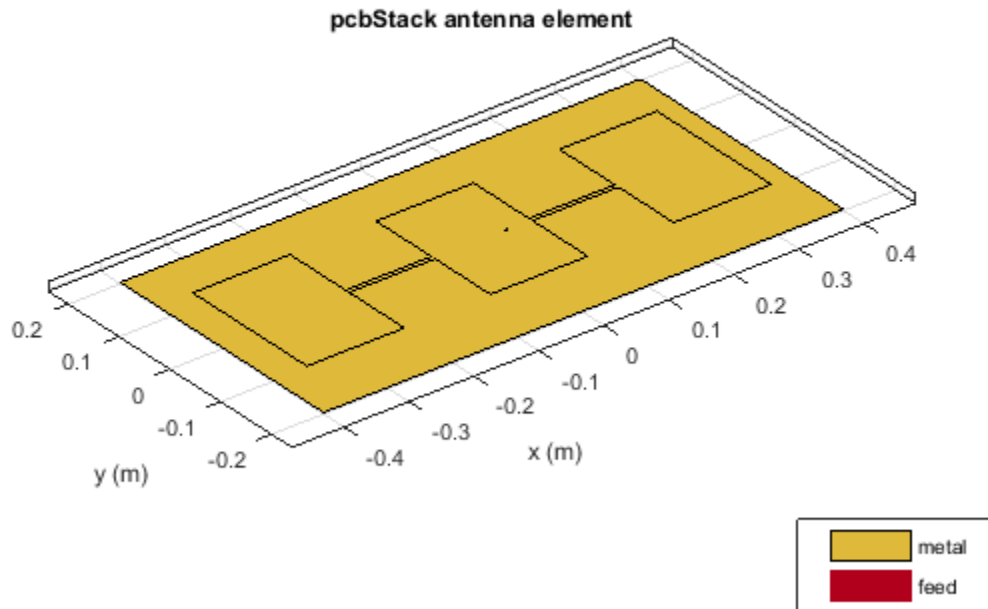
Strip join the sections.

```
r4 = antenna.Rectangle('Length',0.65*lambda,'Width',0.02*lambda,'Center',[lambda/2,0],
r5 = copy(r4);
r5.Center = [-lambda/2,0];
s = r + r4 + r5;
figure
show(s)
```



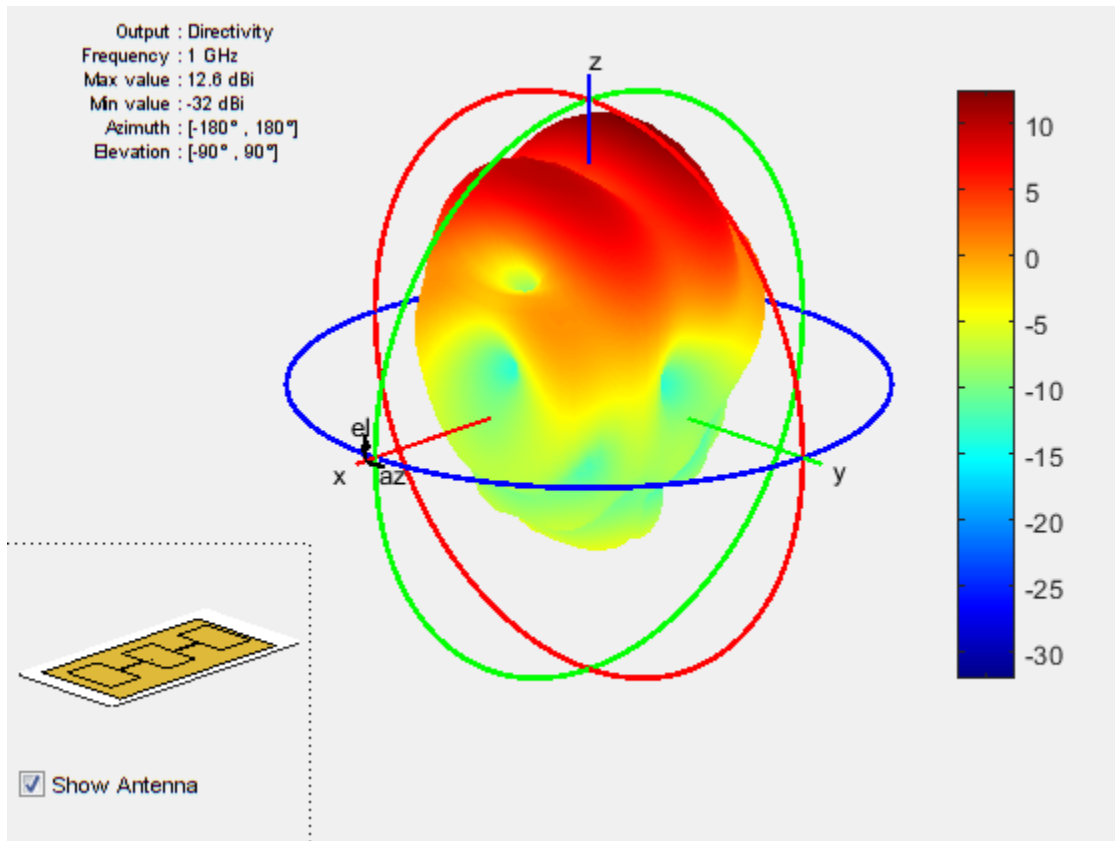
Define the properties of the PCB stack.

```
g1.Length = 0.8;  
series_patch = pcbStack;  
series_patch.BoardShape = g1;  
series_patch.Layers = {s,g1};  
series_patch.FeedLocations = [L/4 0 1 2];  
figure  
show(series_patch)
```



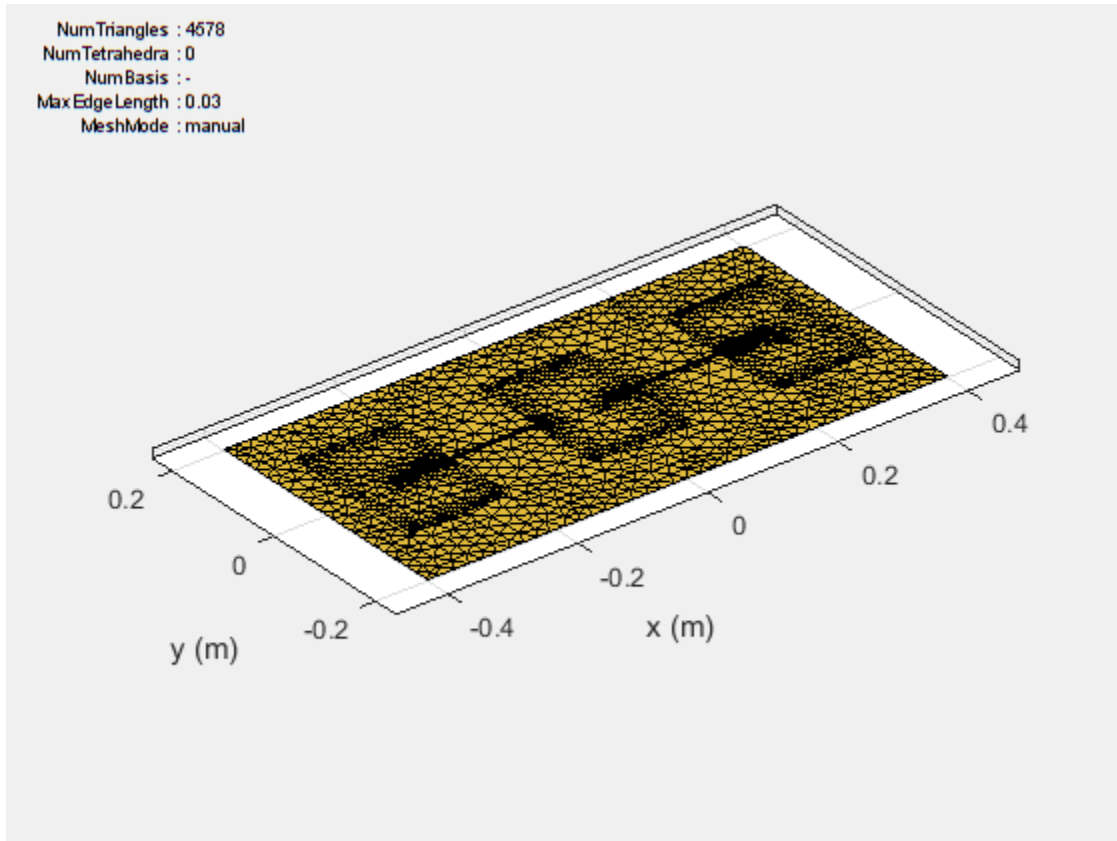
Plot the radiation pattern at 1 GHz for the direct-coupled patch antenna.

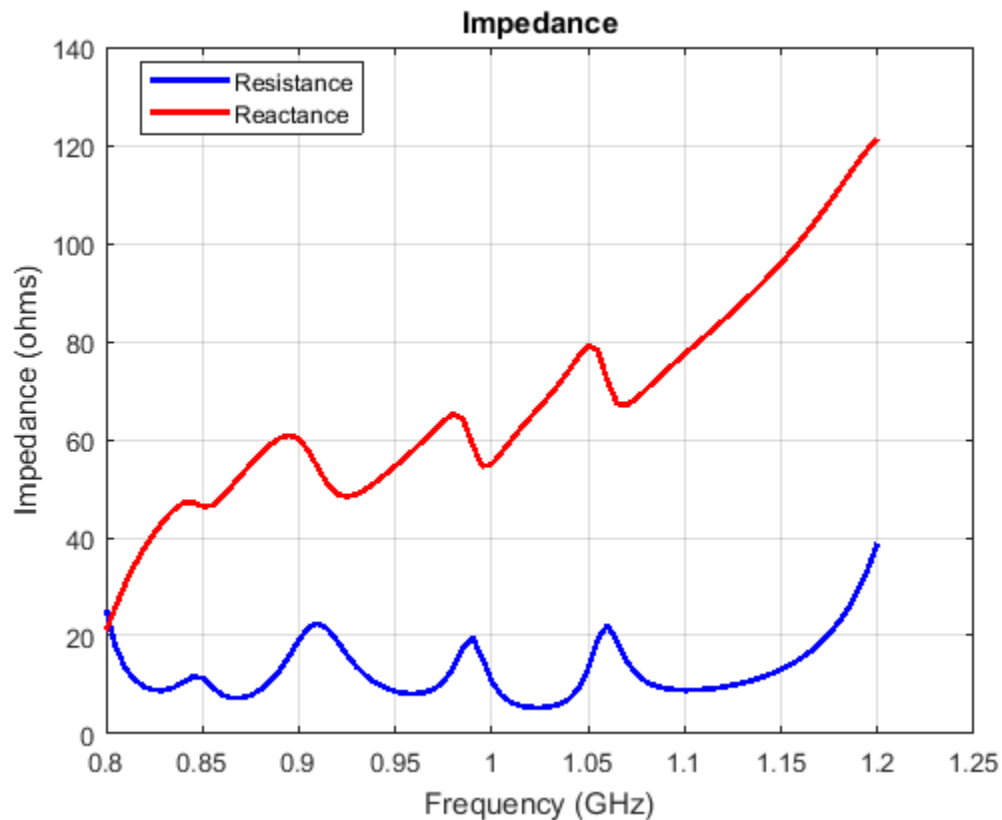
```
figure  
pattern(series_patch,1e9)
```



Mesh the antenna using maximum edge length of 0.03 m. Plot the impedance of the direct-coupled patch antenna for the frequency range, 0.8 GHz to 1.2 GHz.

```
mesh(series_patch, 'MaxEdgeLength', 0.03)
figure
impedance(series_patch, linspace(0.8e9, 1.2e9, 81))
```





### Design Circular Polarized Patch - Truncated Corners

Set the length and width of the patch and the groundplane.

```
Lp = lambda(1)/2;
Wp = lambda(1)/2;
Lgp = 0.75.*lambda(1);
Wgp = 0.75.*lambda(1);
h = 2.e-3;
```

Create the base shape for the patch.

```
p1 = antenna.Rectangle('Length',Lp,'Width',Wp,'NumPoints',20);
```

Truncate the corners of the rectangle.

```
Lcorner = 0.25*Lp;
Wcorner = 0.25*Wp;
cornerCenter1 = [-Lp/2,Wp/2,0];
cornerCenter2 = [Lp/2,-Wp/2,0];
pcorner1 = antenna.Rectangle('Length',Lcorner,'Width',Wcorner);
pcorner1 = rotateZ(pcorner1,45);
pcorner1 = translate(pcorner1,cornerCenter1');
pcorner2 = antenna.Rectangle('Length',Lcorner,'Width',Wcorner);
pcorner2 = rotateZ(pcorner2,45);
pcorner2 = translate(pcorner2,cornerCenter2');
pradiator = p1 -pcorner1-pcorner2;
```

Create the groundplane shape.

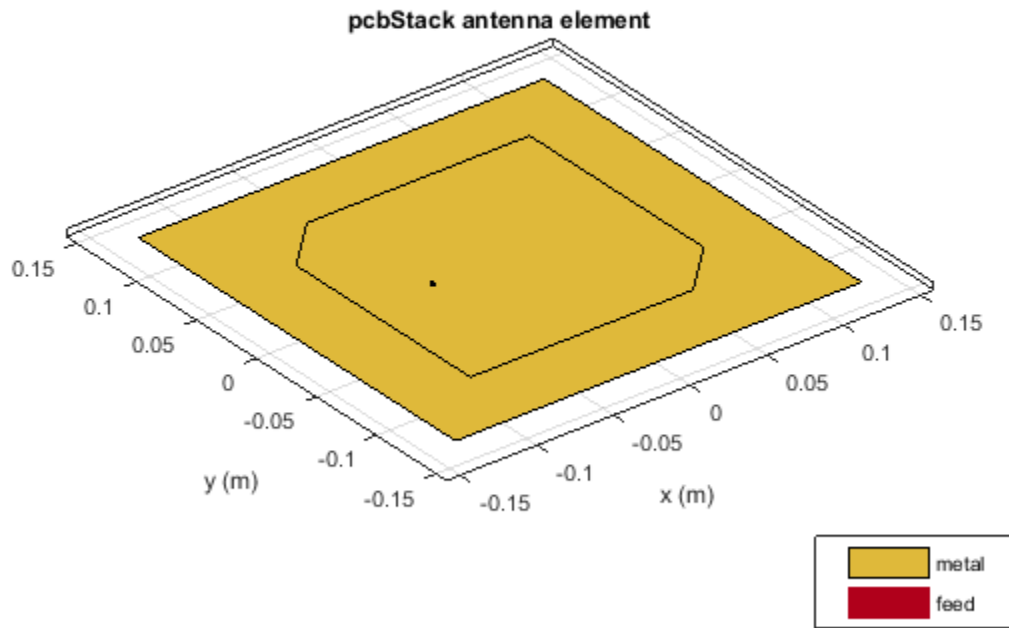
```
p2 = antenna.Rectangle('Length',Lgp,'Width',Wgp);
```

Define the dielectric Layer.

```
d1 = dielectric('Air');
```

Define the properties of the PCB stack for the circularly polarized patch.

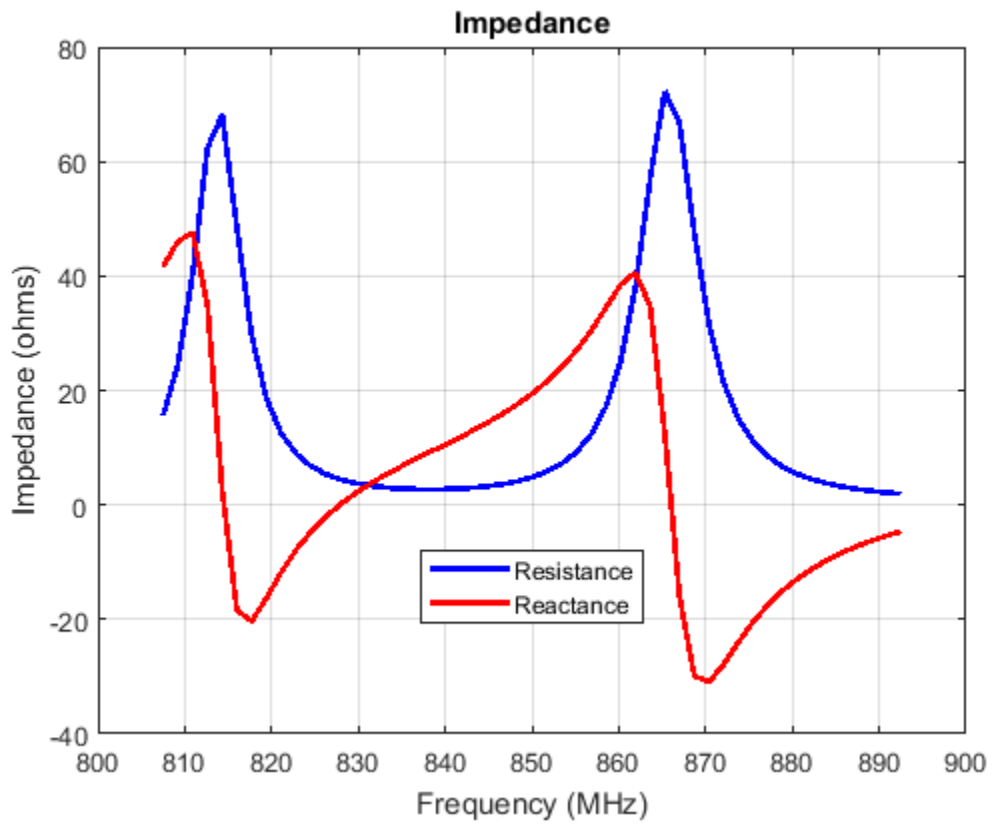
```
truncatedCornerPatch = pcbStack;
truncatedCornerPatch.Name = 'Basic Patch';
truncatedCornerPatch.BoardThickness = h;
truncatedCornerPatch.BoardShape = p2;
truncatedCornerPatch.Layers = {pradiator,d1,p2};
truncatedCornerPatch.FeedLocations = [-lambda(1)/8 0 1 3];
figure
show(truncatedCornerPatch)
```



Plot impedance of the circularly polarized antenna.

```
figure  
impedance(truncatedCornerPatch, freq1)
```







# Introduction to Arrays

---

# Array Modeling and Analysis

This example shows how to construct, visualize, and analyze an antenna array from the Antenna Toolbox.

### Create Antenna Array Using Antenna Elements

Create a default rectangular antenna array using the `rectangularArray` element in the array library. By default, this array uses the dipole antenna as element.

```
ra = rectangularArray
```

```
ra =
```

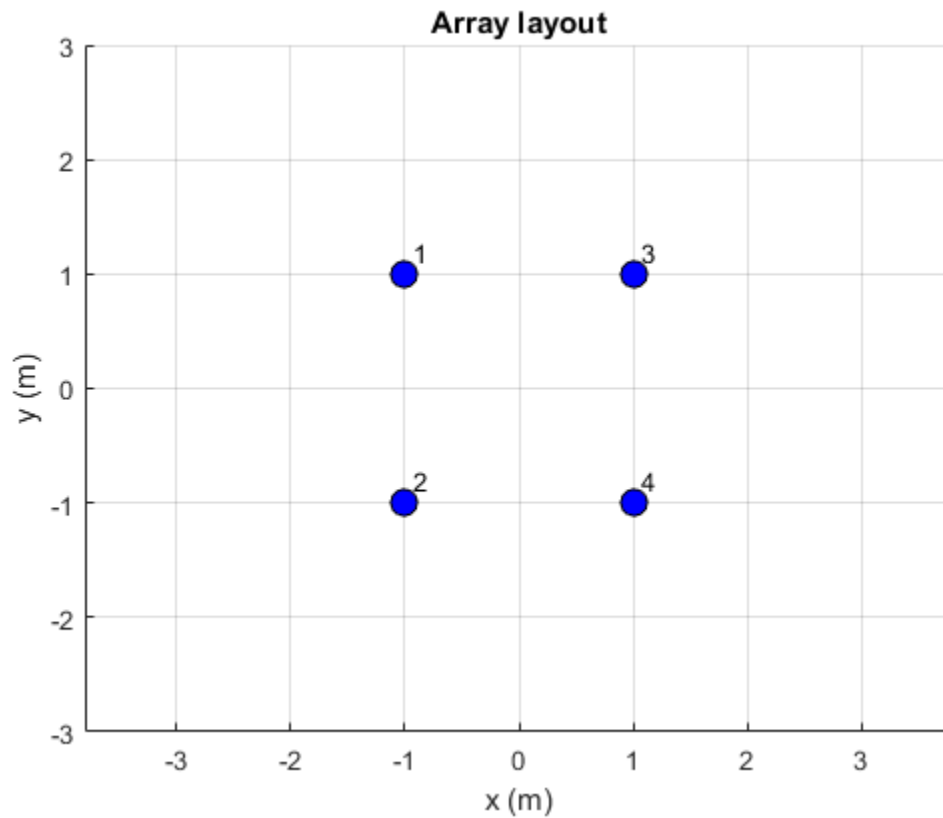
```
    rectangularArray with properties:
```

```
        Element: [1×1 dipole]
           Size: [2 2]
      RowSpacing: 2
   ColumnSpacing: 2
         Lattice: 'Rectangular'
  AmplitudeTaper: 1
      PhaseShift: 0
           Tilt: 0
      TiltAxis: [1 0 0]
```

### Visualize Layout of Array

Use the `layout` function to plot the position of array elements in the x-y plane. By default, the rectangular array is a 4-element dipole array in a 2x2 rectangular lattice.

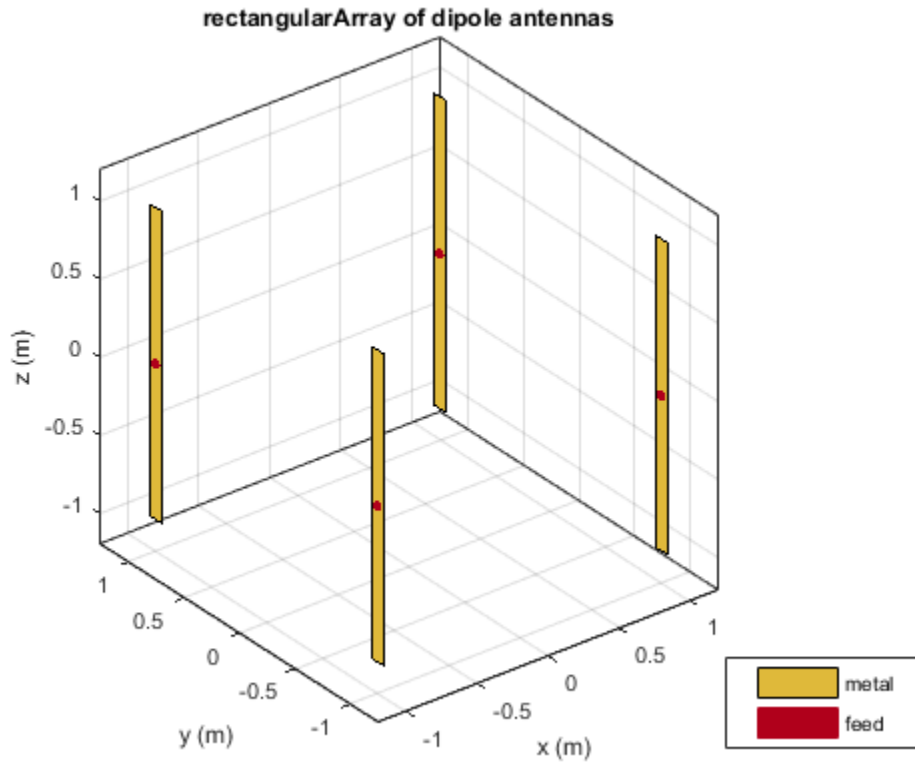
```
layout(ra)
```



### Visualize Geometry of Array

Use the show function to view the structure of the rectangular antenna array.

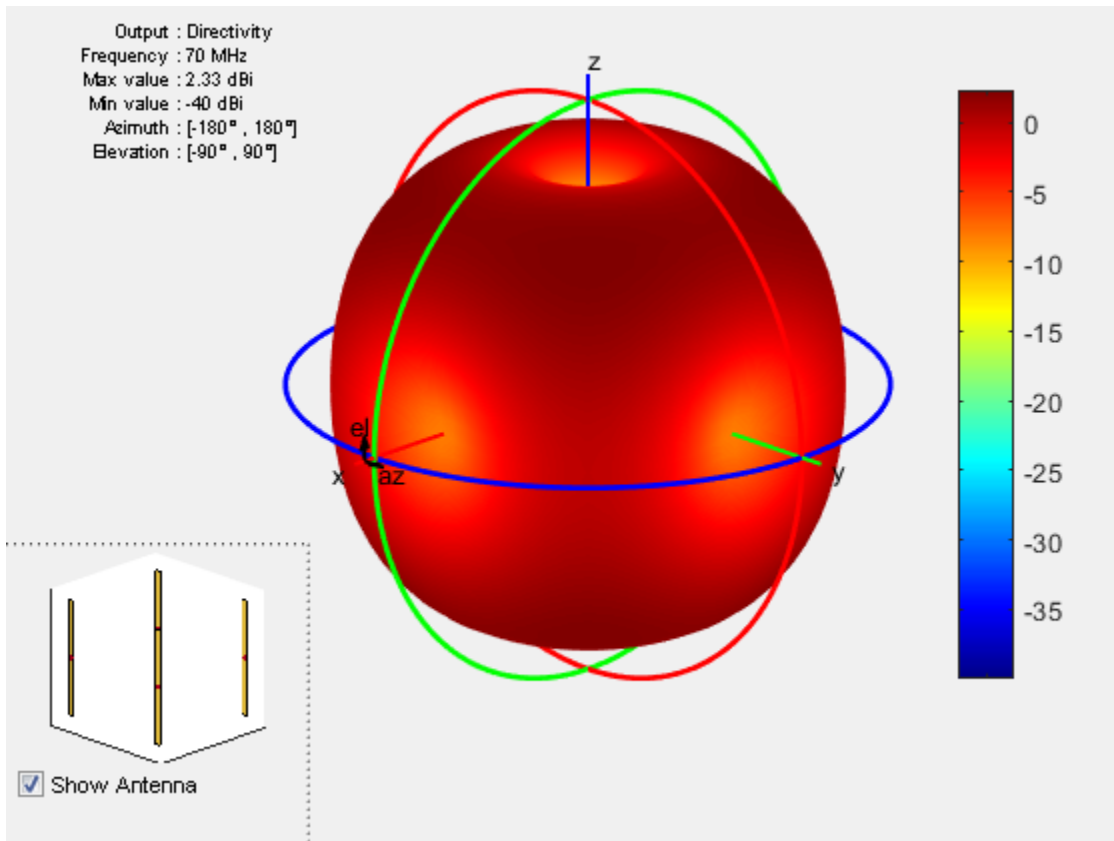
```
show(ra)
```



### Plot Radiation Pattern of Array

Use the pattern function to plot the radiation pattern of the rectangular array. The radiation pattern is the spatial distribution of the power of an array. The pattern displays the directivity or gain of the array. By default, the pattern function plots the directivity of the array.

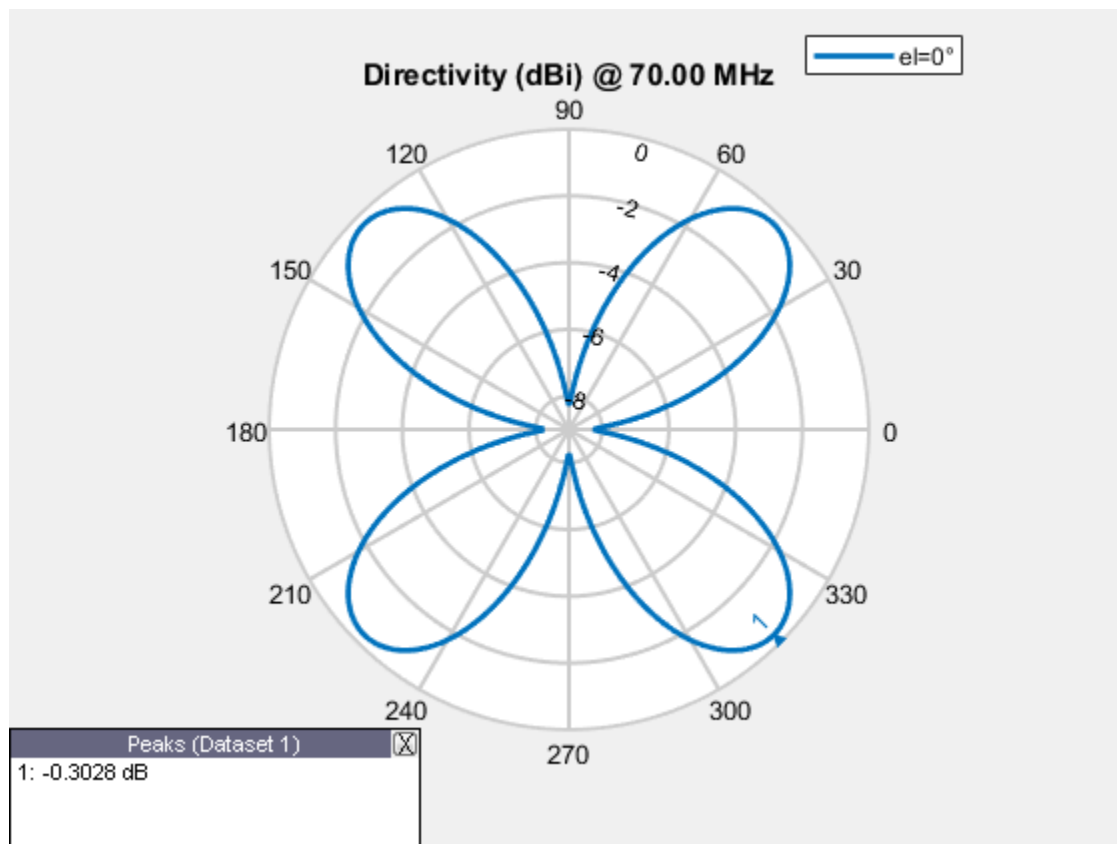
```
pattern(ra,70e6)
```



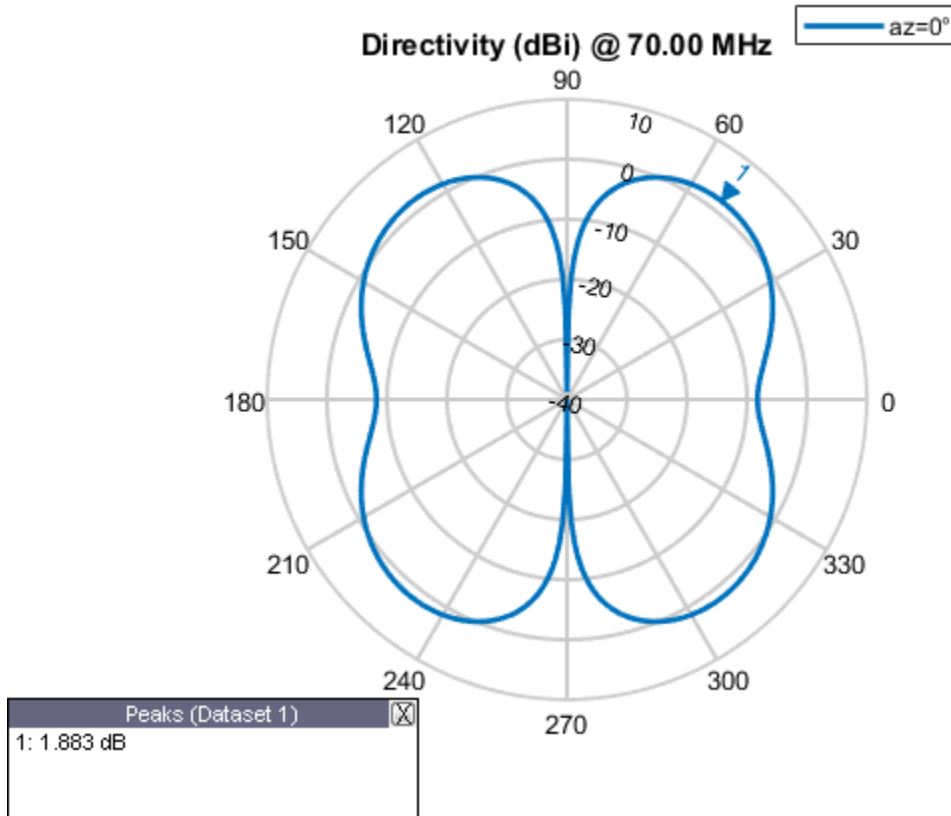
### Plot Azimuth and Elevation Pattern of Array

Use `patternAzimuth` and `patternElevation` functions to plot the azimuth and elevation pattern of the rectangular array. These two patterns are the 2D radiation pattern of the array at a specified frequency.

```
patternAzimuth(ra,70e6)
figure
patternElevation(ra,70e6)
```







### Calculate the Directivity of Array

Directivity is the ability of an array to radiate power in a particular direction. It can be defined as the ratio of the maximum radiation intensity in the desired direction to the average radiation intensity in all other directions. Use pattern function to calculate the directivity of the rectangular array.

```
[Directivity] = pattern(ra,70e6,0,90)
```

```
Directivity =
```

```
-38.7462
```

### Calculate EH Fields of Array

Use the `EHfields` function to calculate the EH fields of the rectangular array. EH fields are the x, y, and z components of the electric and magnetic fields of an array. These components are measured at a specific frequency and at specified points in space.

```
[E,H] = EHfields(ra,70e6,[0;0;1])
```

```
E =
```

```
0.0000 + 0.0000i  
-0.0023 + 0.0017i  
-1.3896 - 0.0428i
```

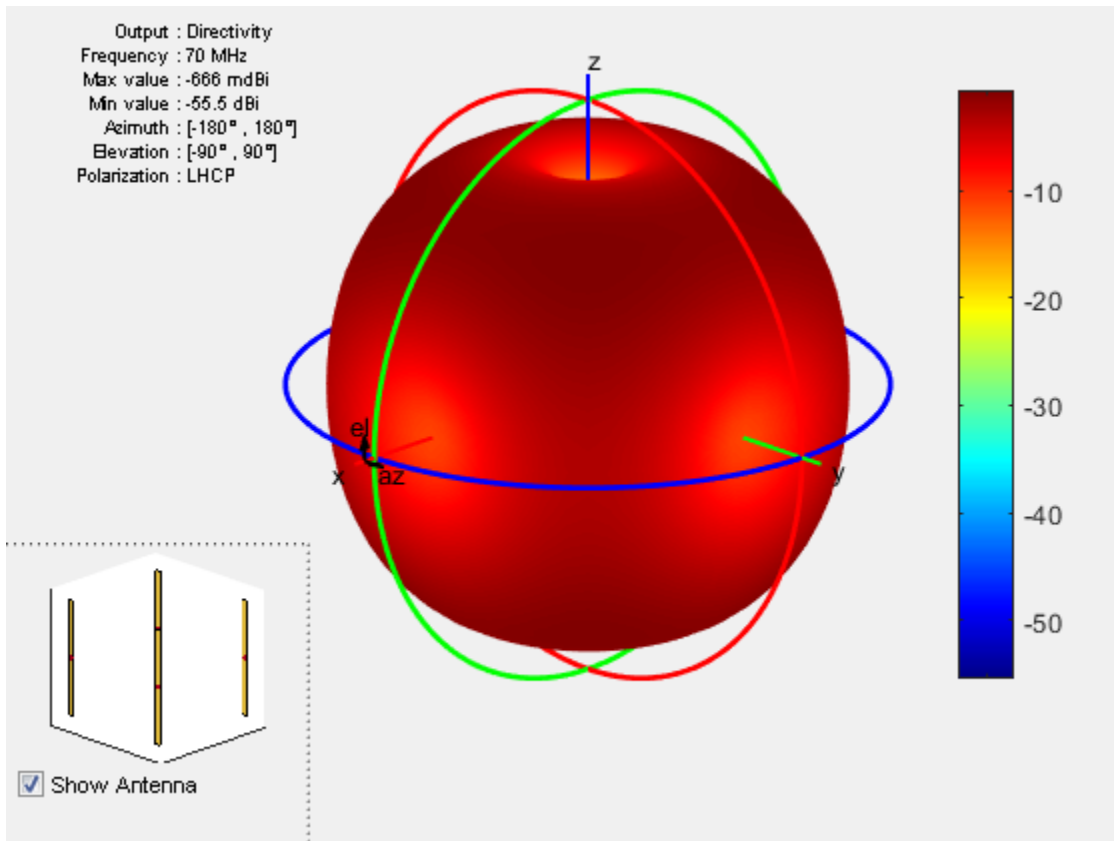
```
H =
```

```
1.0e-05 *  
  
0.1601 - 0.7689i  
0.0000 + 0.0000i  
-0.0000 - 0.0000i
```

### Plot Different Polarizations of Array

Use the Polarization name-value pair in the `pattern` function to plot the different polarization patterns of the rectangular array. Polarization is the orientation of the electric field, or E-field, of an array. Polarization is classified as elliptical, linear, or circular. This example shows the left-hand circularly polarized(LHCP) radiation pattern of the rectangular array.

```
pattern(ra,70e6,'Polarization','LHCP')
```



### Calculate Beamwidth of Array

Use the beamwidth function to calculate the beamwidth of the rectangular array. The beamwidth of an array is the angular measure of the array pattern coverage. The beamwidth angle is measured in plane that contains the direction of main lobe of the array.

```
[bw, angles] = beamwidth(ra, 70e6, 0, 1:1:360)
```

```
bw =
```

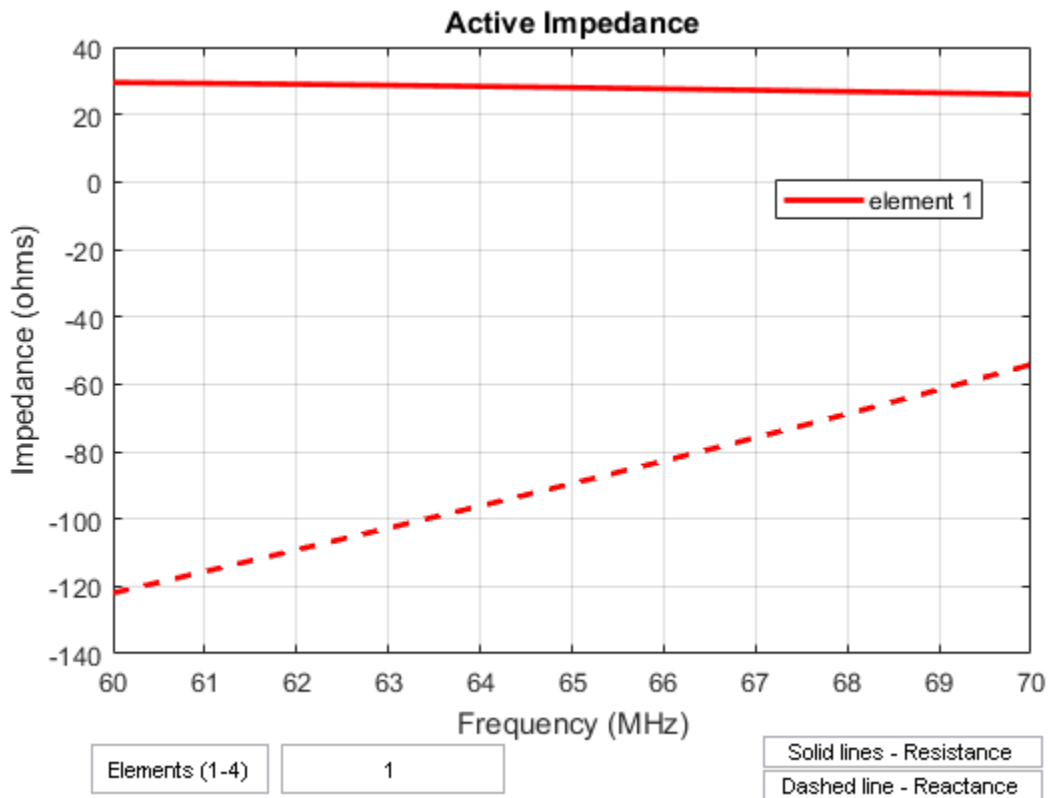
```
44.0000  
44.0000
```

```
angles =
    28    72
   108   152
```

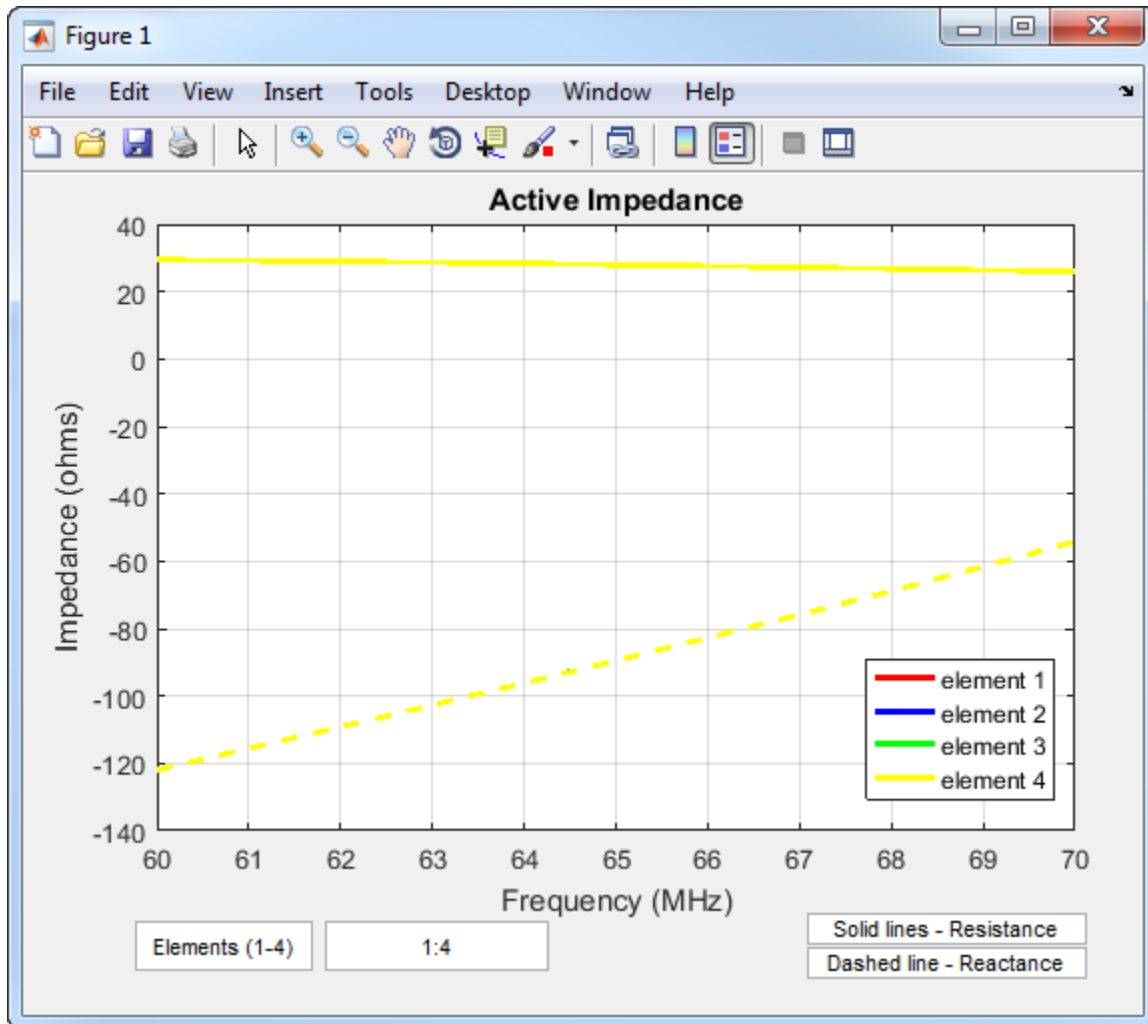
### Calculate Scan Impedance of Array

Use the impedance function to calculate and plot the input impedance of rectangular array. Active impedance, or scan impedance, is the input impedance of each antenna element in an array, when all elements are excited.

```
impedance(ra,60e6:1e6:70e6)
```



You can also view the impedance of all four elements by changing the number of elements on the plot from 1 to 1:4. See figure.



### Calculate Reflection Coefficient of Array

Use the `sparameters` function to calculate the S11 value of the rectangular array. S11 value gives the reflection coefficient of the array.

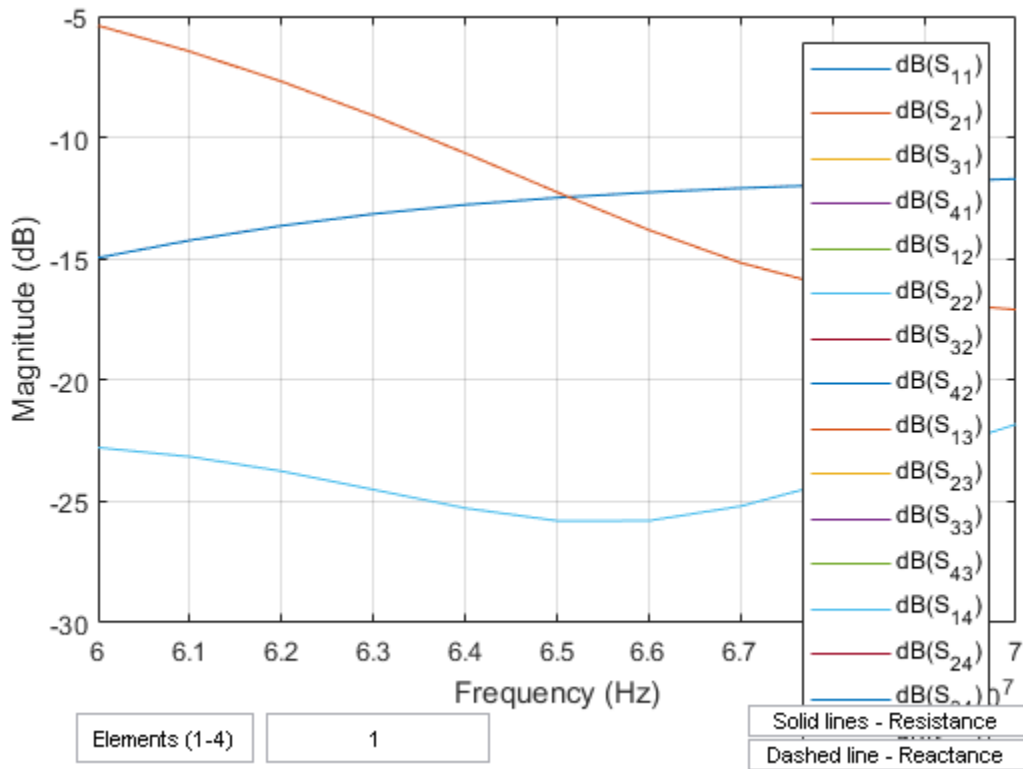
```
S = sparameters(ra,60e6:1e6:70e6,72)
rfplot(S)
```

```
S =
```

```
  sparameters: S-parameters object
```

```
    NumPorts: 4
  Frequencies: [11×1 double]
  Parameters: [4×4×11 double]
    Impedance: 72
```

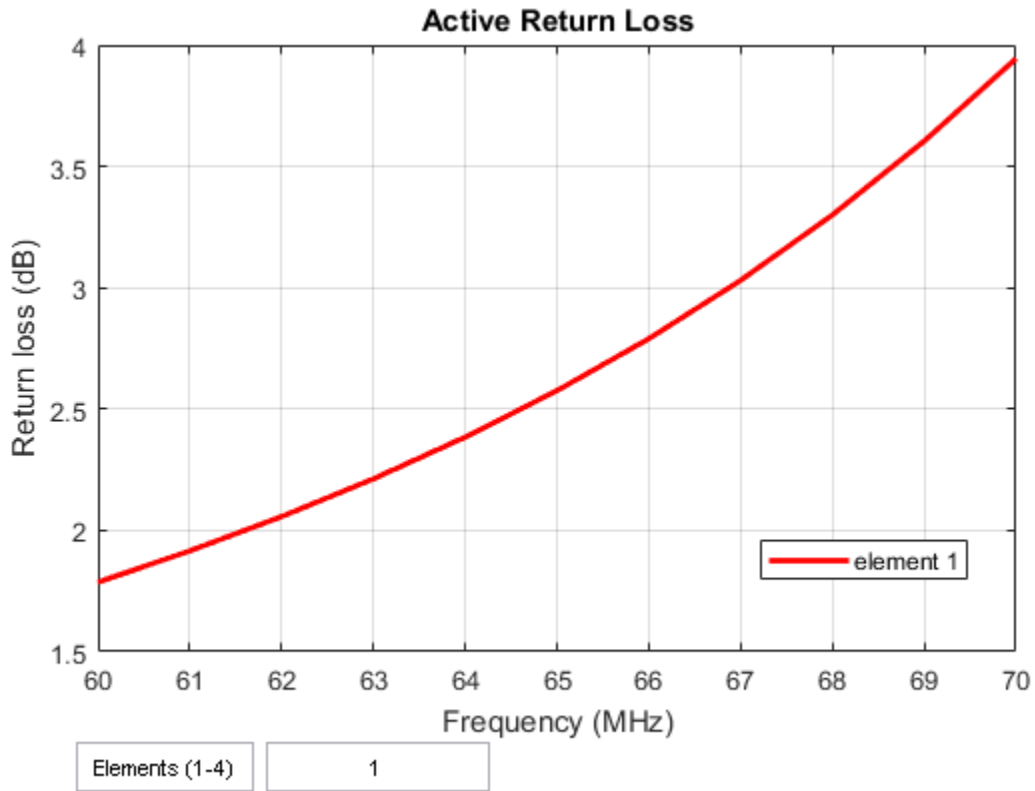
```
rfparam(obj,i,j) returns S-parameter Sij
```



### Calculate Return Loss of Array

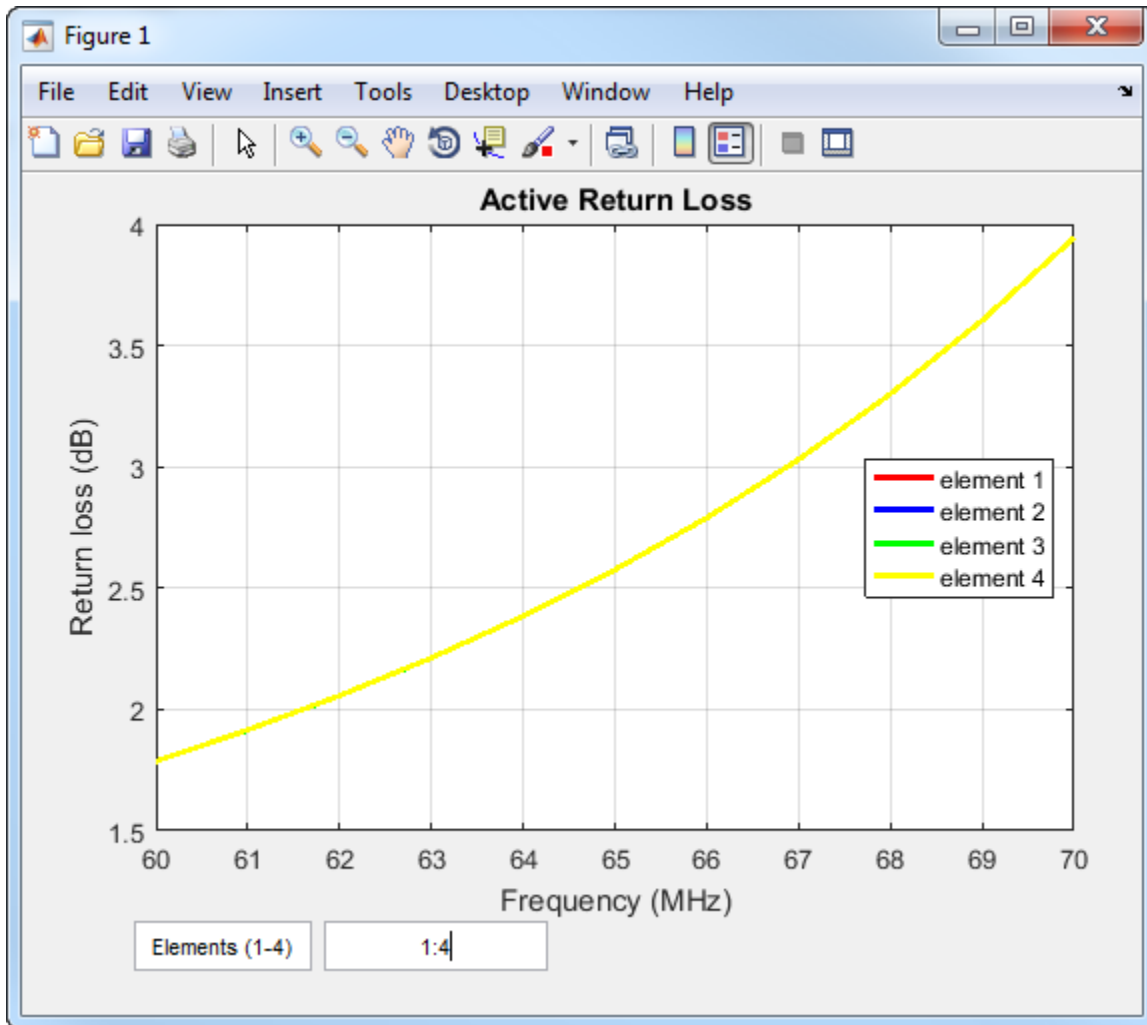
Use the returnLoss function to calculate and plot the return loss of the rectangular array.

```
returnLoss(ra, 60e6:1e6:70e6, 72)
```



You can also view the return loss of all four elements by changing the number of elements on the plot from 1 to 1:4. See figure.



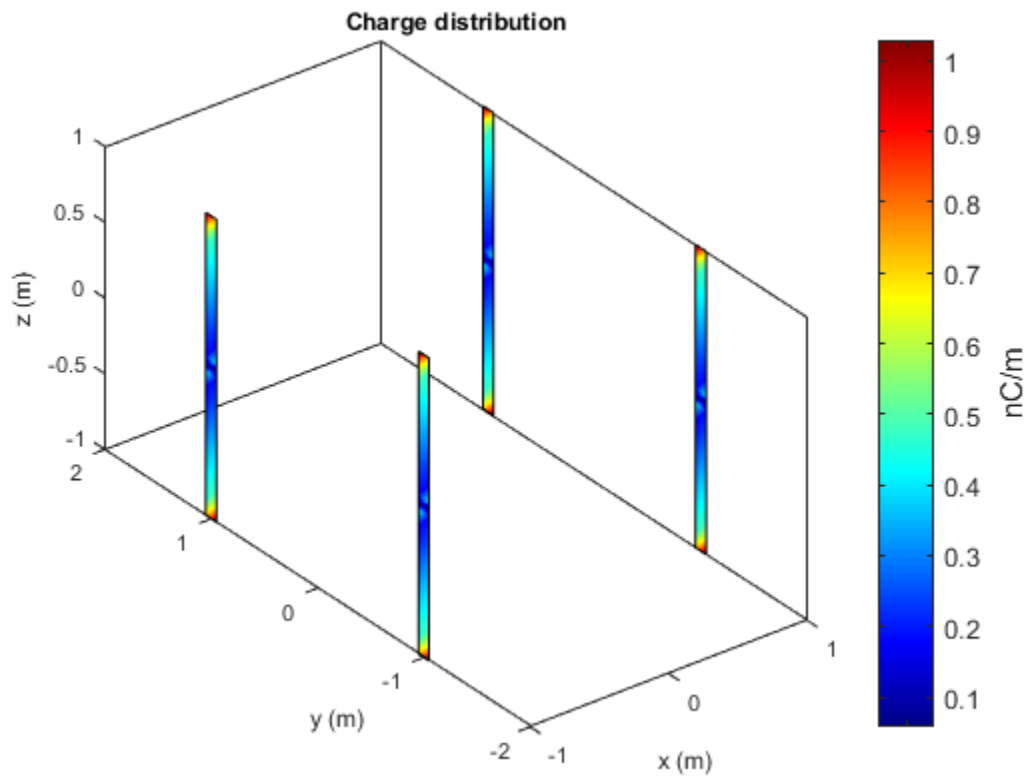


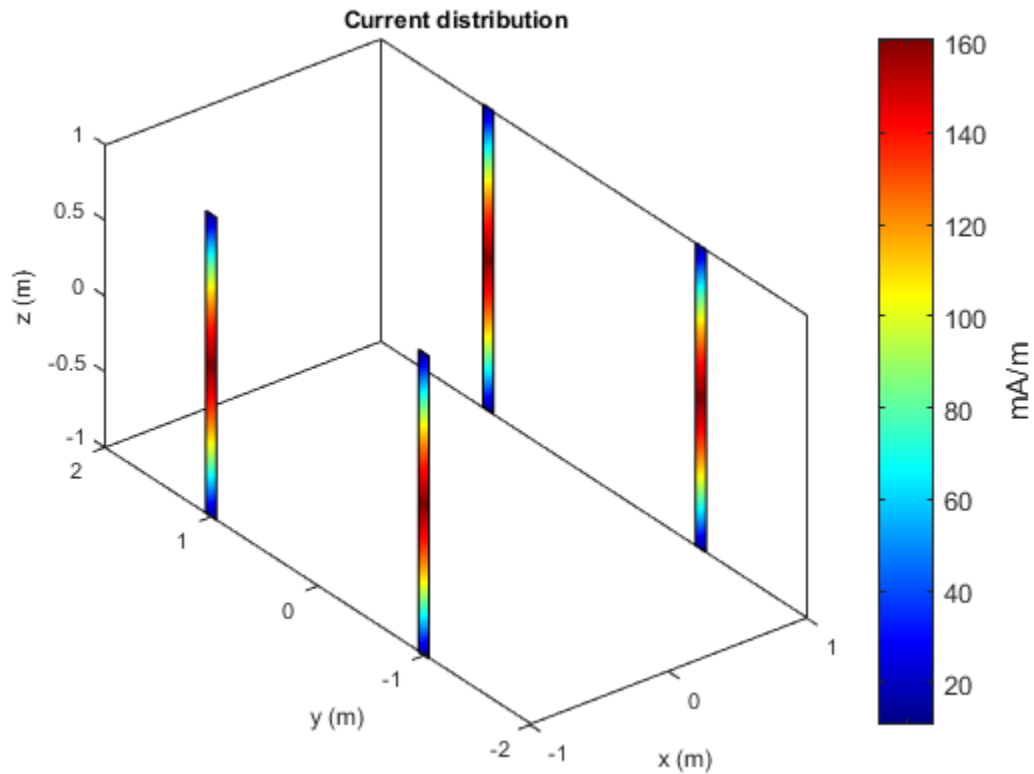
### Calculate Charge and Current Distribution Of Array

Use the charge and current functions to calculate the charge and current distribution on the rectangular array surface.

```
charge(ra, 70e6)
figure
```

current (ra, 70e6)

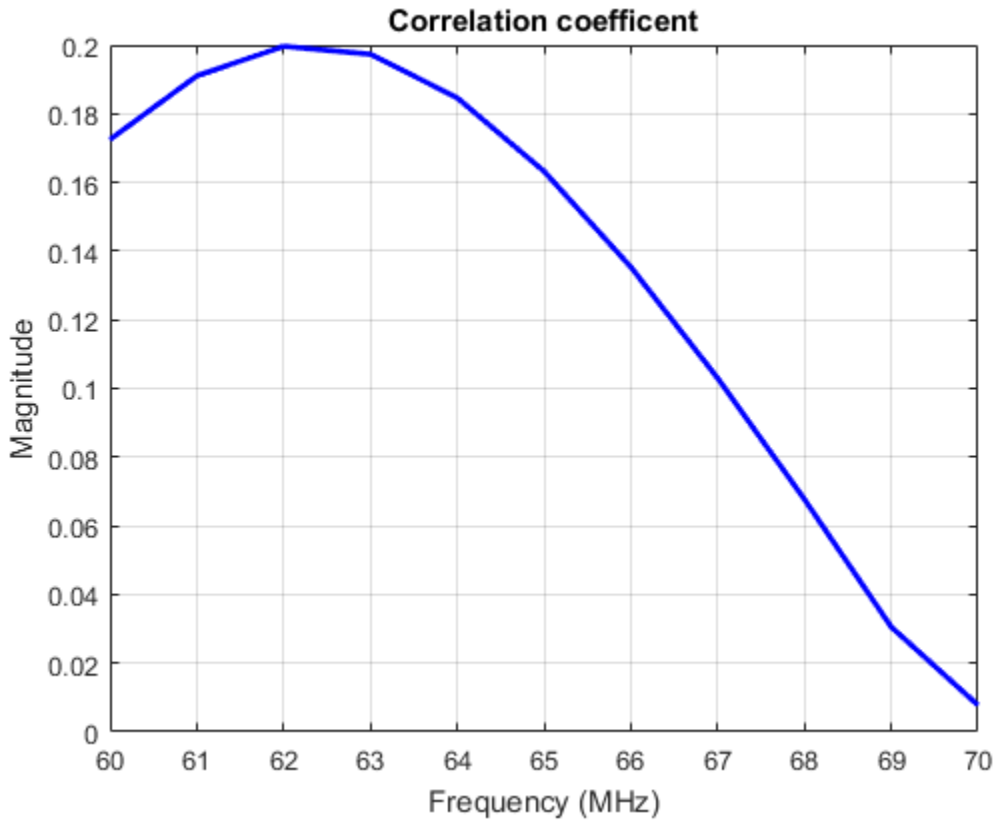




### Calculate Correlation Coefficient of Array

Use the correlation to calculate the correlation coefficient of the rectangular array. The correlation coefficient is the relationship between the incoming signals at the antenna ports in an array.

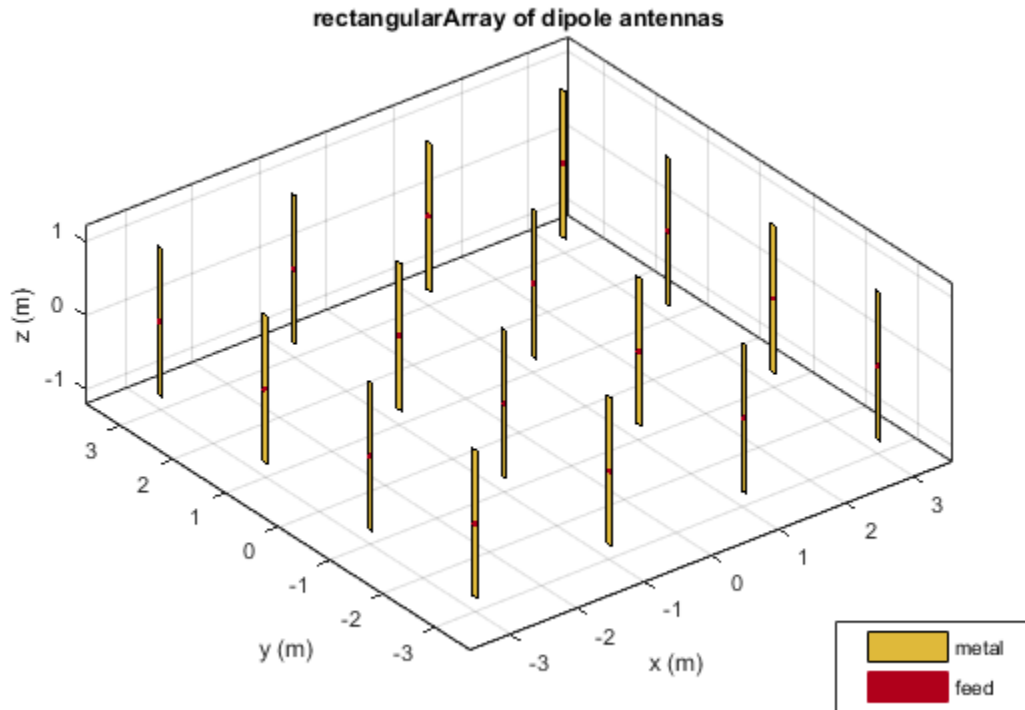
```
correlation(ra,60e6:1e6:70e6,1,2)
```



### Change Size of Array and Visualize Layout

Use the 'Size' property of the rectangular array to change it to a 16-element dipole array.

```
ra.Size = [4 4];  
show(ra)
```



### Change Array Elements Spacing To Nonuniform

Use the 'RowSpacing' and 'ColumnSpacing' properties of rectangular array to change the spacing between the antenna elements.

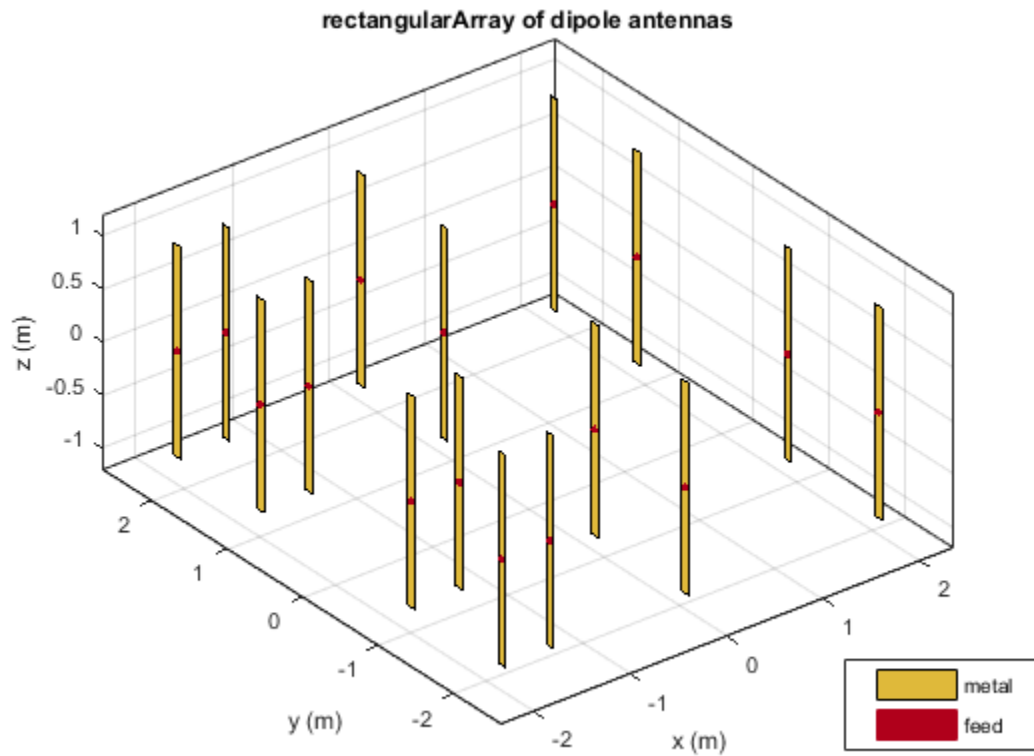
```
ra.RowSpacing = [ 1.1 2 1.2];
ra.ColumnSpacing =[0.5 1.4 2]
show(ra)
```

ra =

rectangularArray with properties:

Element: [1×1 dipole]

```
Size: [4 4]
RowSpacing: [1.1000 2 1.2000]
ColumnSpacing: [0.5000 1.4000 2]
Lattice: 'Rectangular'
AmplitudeTaper: 1
PhaseShift: 0
Tilt: 0
TiltAxis: [1 0 0]
```



### References

[1] Balanis, C.A. "Antenna Theory. Analysis and Design", p. 514, Wiley, New York, 3rd Edition, 2005.