Phased Array Design and Simulation in MATLAB

Hands-on Workshop

Akash Gopisetty, **Darius** Subacius

DSP Online Conference

MathWorks

AGENDA

1. Setup and Logistics
2. Introduction
3. Hands-on exercises
4. Takeaways and Resources

THE SPEAKER

Akash Gopisetty

Product Manager, Signal Processing

Focus: wireless systems, antenna arrays, signal processing, stem advocacy

Akash is passionate about translating complex technology into stories that resonate with engineers. As a Product Marketing Manager at MathWorks, he focuses on phased arrays and wireless communication, helping customers design and simulate systems. Prior to this role, Akash worked with schools around the world at the pre-university level, promoting STEM education to the next generation of engineers and scientists. He holds a Master of Science degree in Electrical and Computer Engineering from Carnegie Mellon University (Go Tartans!).

THE SPEAKER

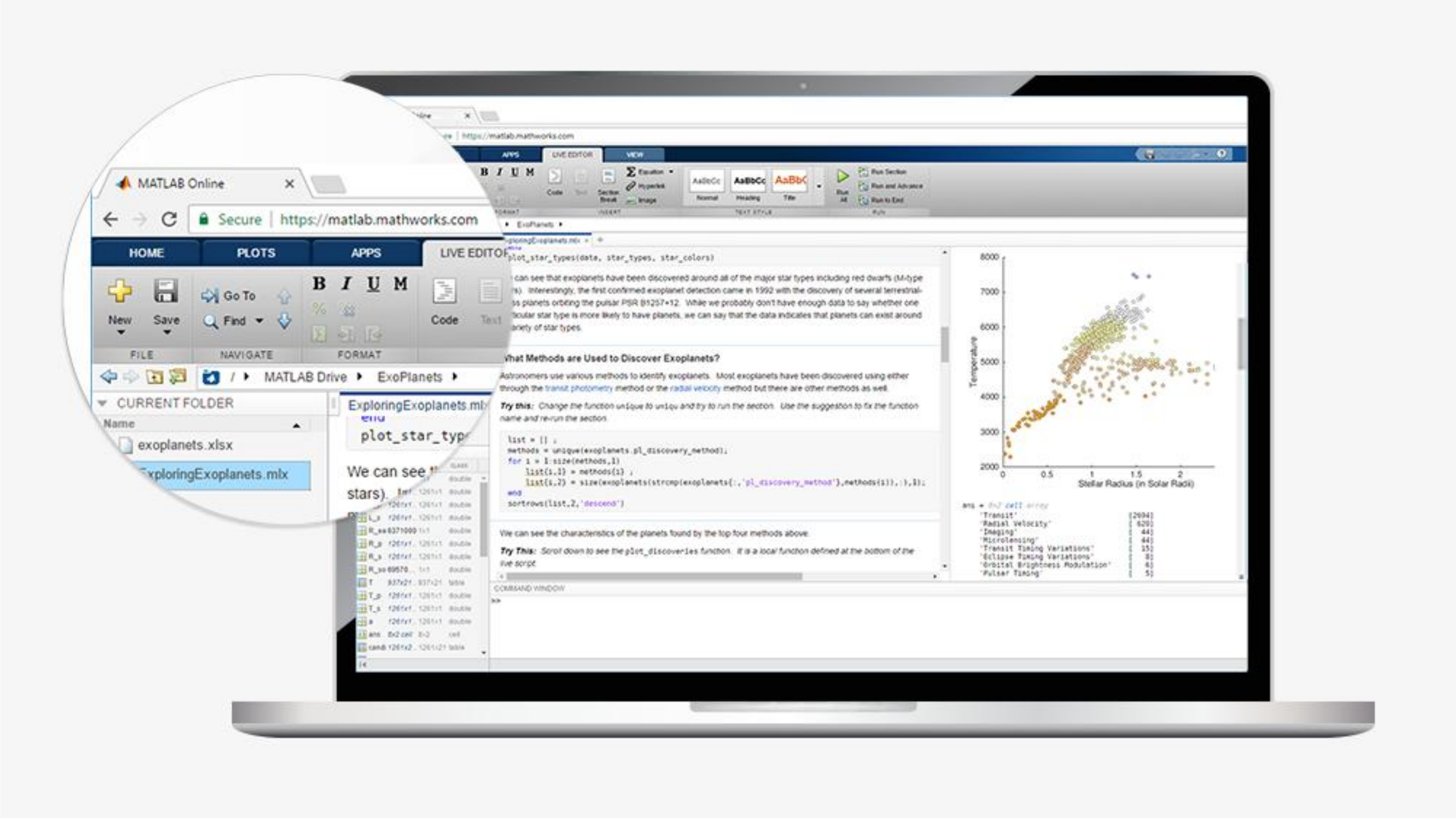
Darius Subacius

Sr. Application Engineer, Radars

Focus: radar systems, antenna arrays, EMSO, communication systems

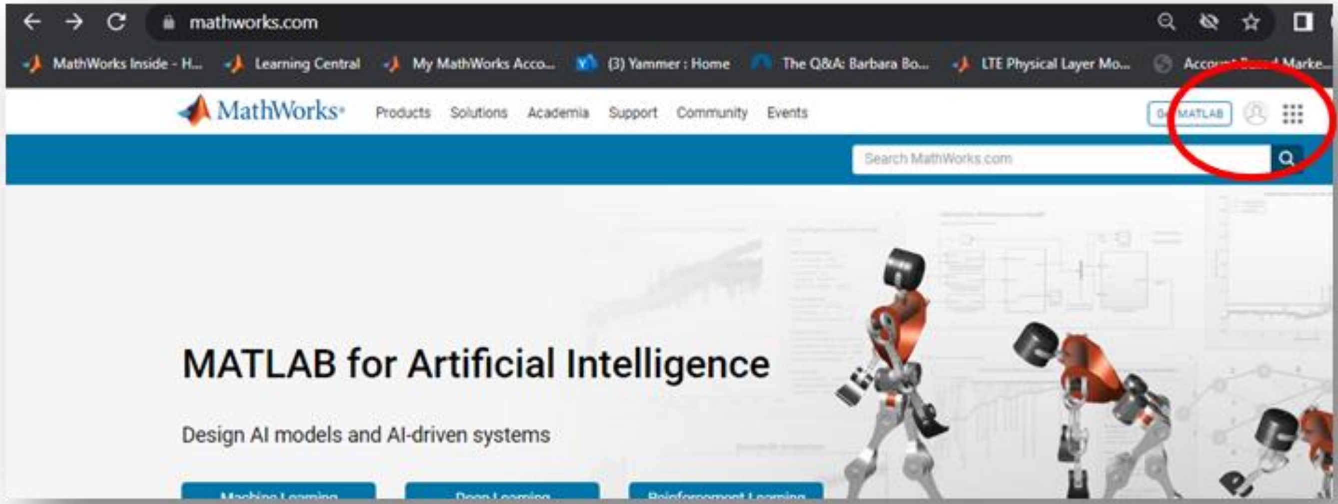
Darius holds a PhD in Solid State Physics focused on Electro-optics. With over 20 years of experience in the technology industry, Darius has a proven track record in the design of complex communication systems, radars, and signal processing. Prior to joining MathWorks, Darius worked for 11 years at Raytheon Technologies, where he was responsible for the design and development of cutting-edge Radar and RF communication systems. Following this, he spent 6 years at Viasat, where he continued to work on system engineering of advanced communication technologies.

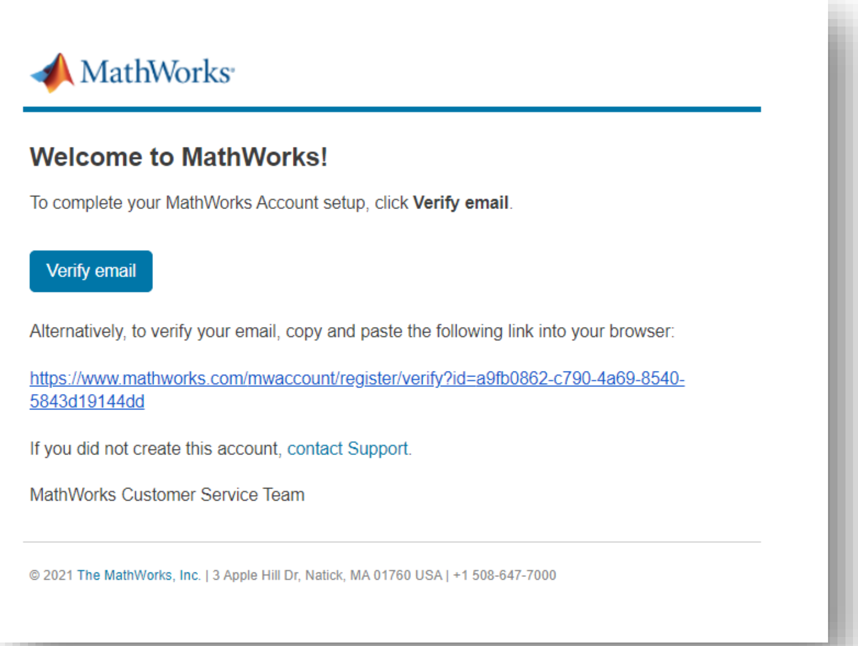
Use MATLAB Online for this workshop



If you don't already have MATLAB Online access:

- Go to www.mathworks.com and click on the profile image at the top right
- You will be prompted to create an account (please use your work address)
- Once you get your confirmation email, please verify your account





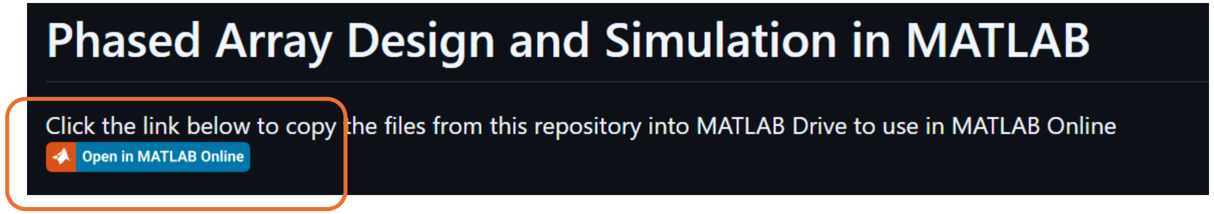
You will need two links for the workshop

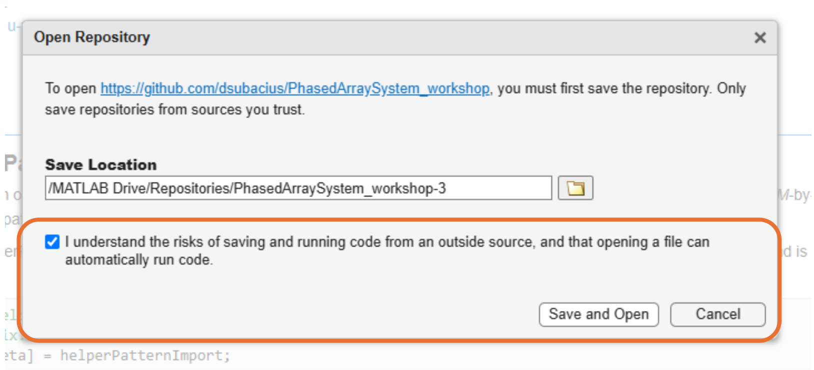
1. Access MATLAB Online until 14 Jan 2026:

   https://www.mathworks.com/licensecenter/classroom/5050250/

2. Workshop code files on GitHub:

   https://github.com/dsubacius/PhasedArraySystem_workshop/tree/main





Before we start: Quick Check In!

1. Have you used MATLAB in the past?
   Type “MATLAB” for Yes,
   or “No MATLAB” for No

2. Do you have access to MATLAB via your browser, and
   Do you plan to follow along with the exercises?
   Type “Yes” or “No”

3. How familiar are you with phased arrays?
   Scale of 1 (novice)- 10 (expert)

Typical applications of phased arrays

Multifunction Radars

Wireless Communications

Satellite Communications

Audio / Underwater Acoustics

Design, Simulate and Analyze Phased Arrays

Array Design

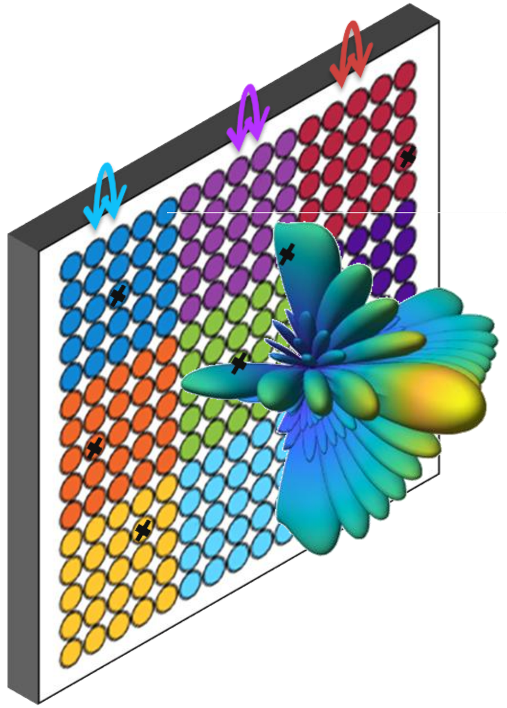
Mutual Coupling

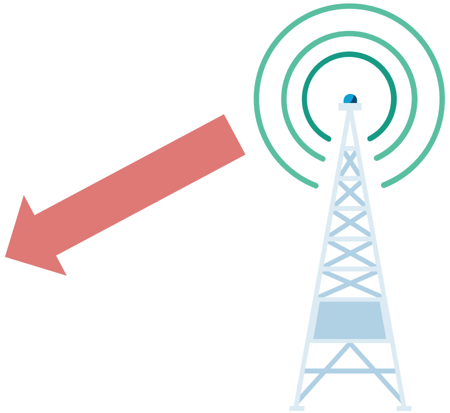
Signal Processing

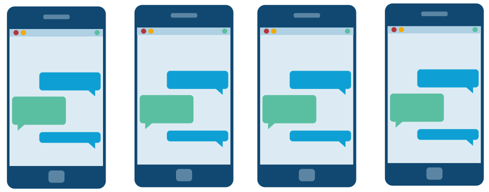
Interference Mitigation

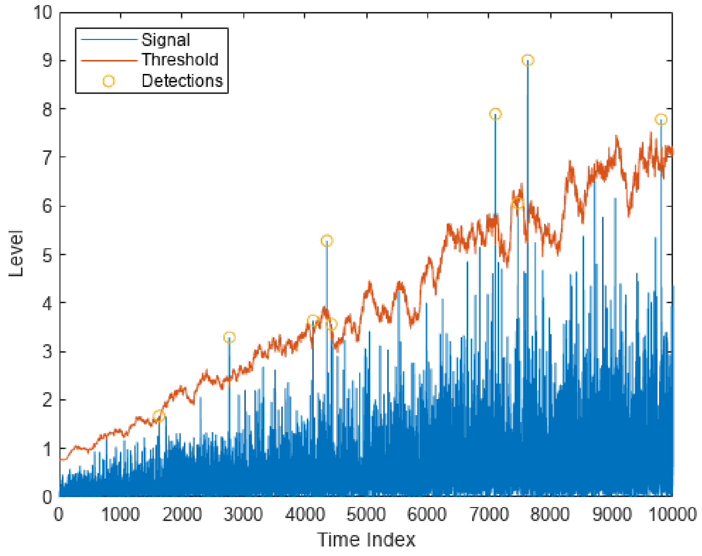
Calibration

Waveform Design









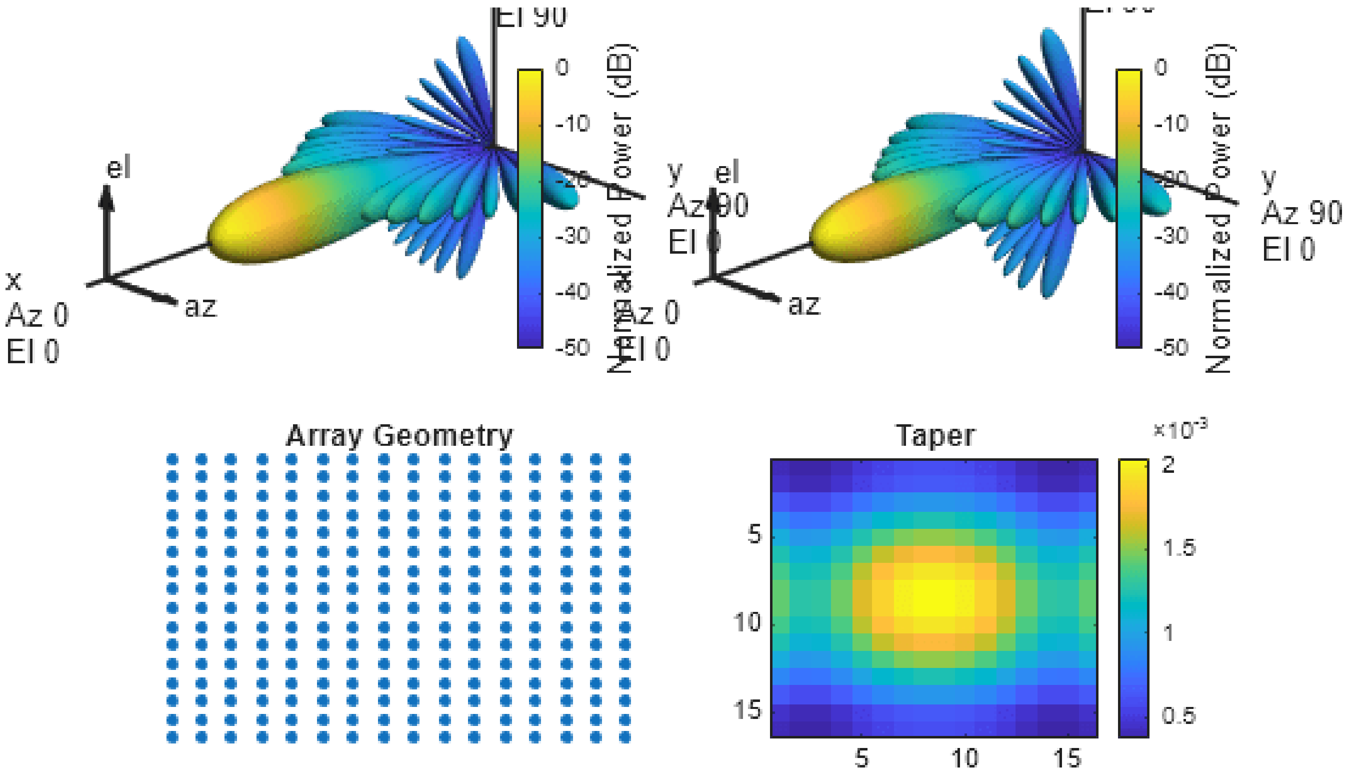
By the end of the workshop, you will be able to

Get started with phased arrays in MATLAB and apply them in your projects

Design and analyze various array configurations

Understand the basics of pattern synthesis

Integrate phased arrays into larger system-level models



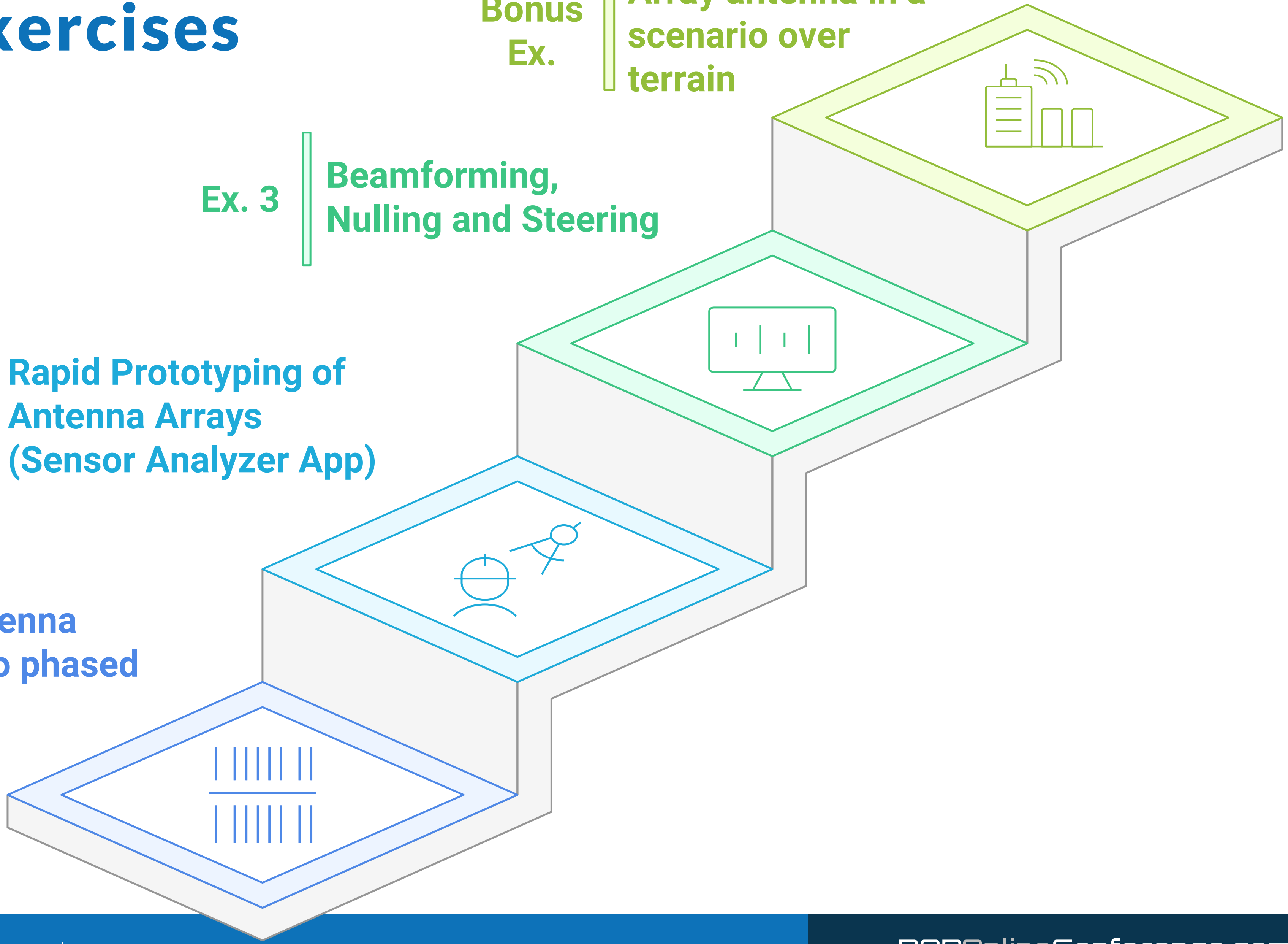
Overview of exercises

Bonus Ex. | Array antenna in a scenario over terrain

Ex. 3 | Beamforming, Nulling and Steering

Ex. 2 | Rapid Prototyping of Antenna Arrays (Sensor Analyzer App)

Ex. 1 | Integrate antenna elements into phased array



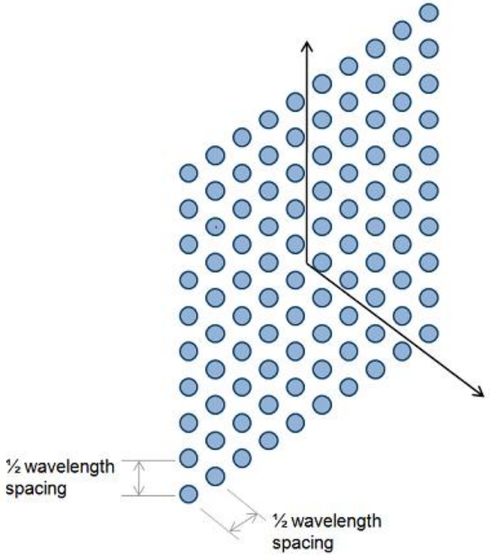
1

Integrate antenna elements into phased array

- Familiarize with Phased Array System toolbox objects using command line interface
- Learn how to import a custom antenna elements from other tools e.g., HFSS
- Construct and analyze a Uniform Rectangular Array (URA) using the defined antenna elements.
- Familiarize yourself with the MATLAB Online environment

Phased Array System Toolbox uses functions and objects:

- Open the link below and view list
- Once **defined,** system objects can be **used** as functions



Antennas

phased.CardioidAntennaElement	Cardioid antenna element *(Since R2021b)*
phased.CosineAntennaElement	Cosine antenna element
phased.CrossedDipoleAntennaElement	Crossed-dipole antenna element
phased.CustomAntennaElement	Custom antenna element
phased.GaussianAntennaElement	Gaussian antenna element *(Since R2021b)*
phased.IsotropicAntennaElement	Isotropic antenna element
phased.NRAntennaElement	5G antenna element described in 3GPP TR 38.901 specificatio
phased.ShortDipoleAntennaElement	Short-dipole antenna element
phased.SincAntennaElement	Sinc antenna element *(Since R2021b)*

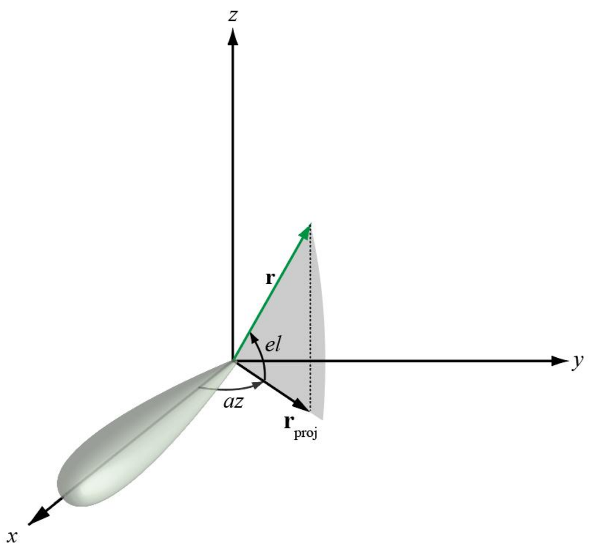
Phased Arrays

phased.ConformalArray	Conformal array
phased.HeterogeneousConformalArray	Heterogeneous conformal array
phased.HeterogeneousULA	Heterogeneous uniform linear array
phased.HeterogeneousURA	Heterogeneous uniform rectangular array
phased.NRRectangularPanelArray	5G antenna array described in 3GPP TR 38.901 specification *(Since*
phased.PartitionedArray	Partition phased array into subarrays
phased.RectangularRIS	Rectangular reconfigurable intelligent surface (RIS) *(Since R2025a)*
phased.ReplicatedSubarray	Phased array formed by replicated subarrays
phased.UCA	Uniform circular array
phased.ULA	Uniform linear array
phased.URA	Uniform rectangular array

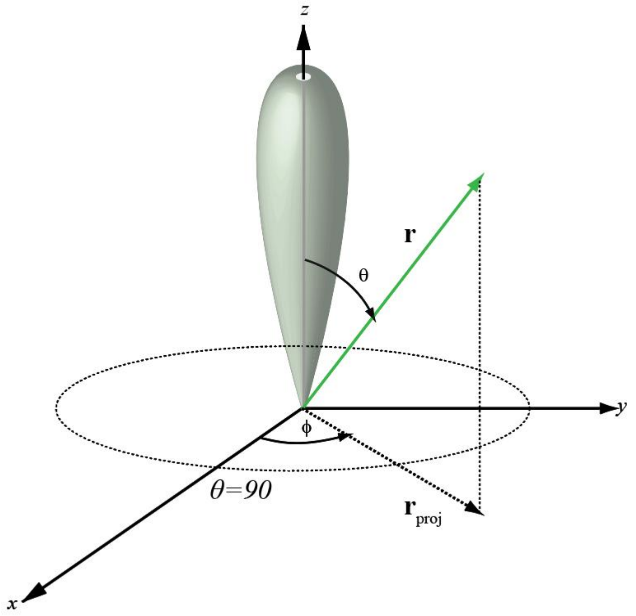
mathworks.com/help/phased/referencelist.html

Spherical Coordinate System in MATLAB

- Define a point in space with distance and two angles
- Phased Array Toolbox natively uses azimuth / elevation
- The toolbox provides **functions to convert between different spherical coordinate systems**



Az [-180:180]
El [-90:90]

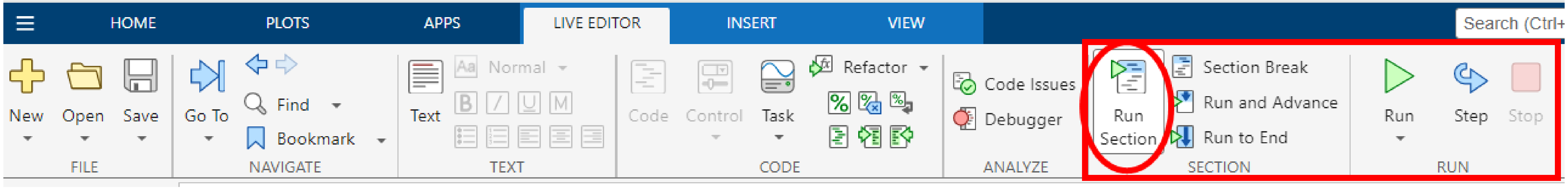


Phi [0:360]
Theta [0:180]

$u = \sin\vartheta\cos\phi$
$v = \sin\vartheta\sin\phi$

mathworks.com/help/phased/ug/spherical-coordinates.html

MATLAB Live Scripts and Coding Practices



- Value-only arguments don't need any modifiers
- Name-value pairs use property names

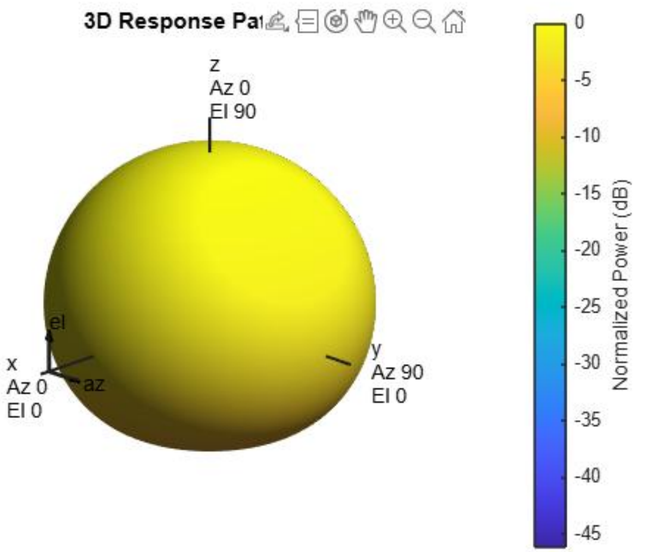
```
y = foo(arg1, ...              % Value-only argument
        Name1=value1);  % Name-value pair
```

- Name=value, and "Name",value, and 'Name',value are all equivalent when specifying name-value pairs
  - Use only one style in a given function call

- Read the code comments! They contain helpful instructions to guide your code, including the usage of value-only arguments and name-value pairs.

Exercise 1: Create an array with custom antenna elements

- Steps:
  1. Create antenna element object
     1. Create a Cosine Antenna element object
     2. Import antenna element
     3. Plot element radiation pattern
  2. Construct 8x8 URA using these elements
  3. Visualize and analyze URA radiation pattern



- Related Resources

  mathworks.com/help/phased/ug/antenna-array-analysis-with-custom-radiation-pattern.html

WORK ON EXERCISE 1
(15 minutes)

Let's hear from you!

1. **Do you design antennas specifically for phased array systems?**
   Are you focused more on element-level design, array configuration, or both?

2. **What tools or software do you typically use for antenna or array design?**
   e.g., MATLAB, CST Studio, HFSS, in-house tools?

3. **How important is electromagnetic (EM) analysis in your workflow?**
   Do you perform full-wave simulations, or rely more on array-level approximations?

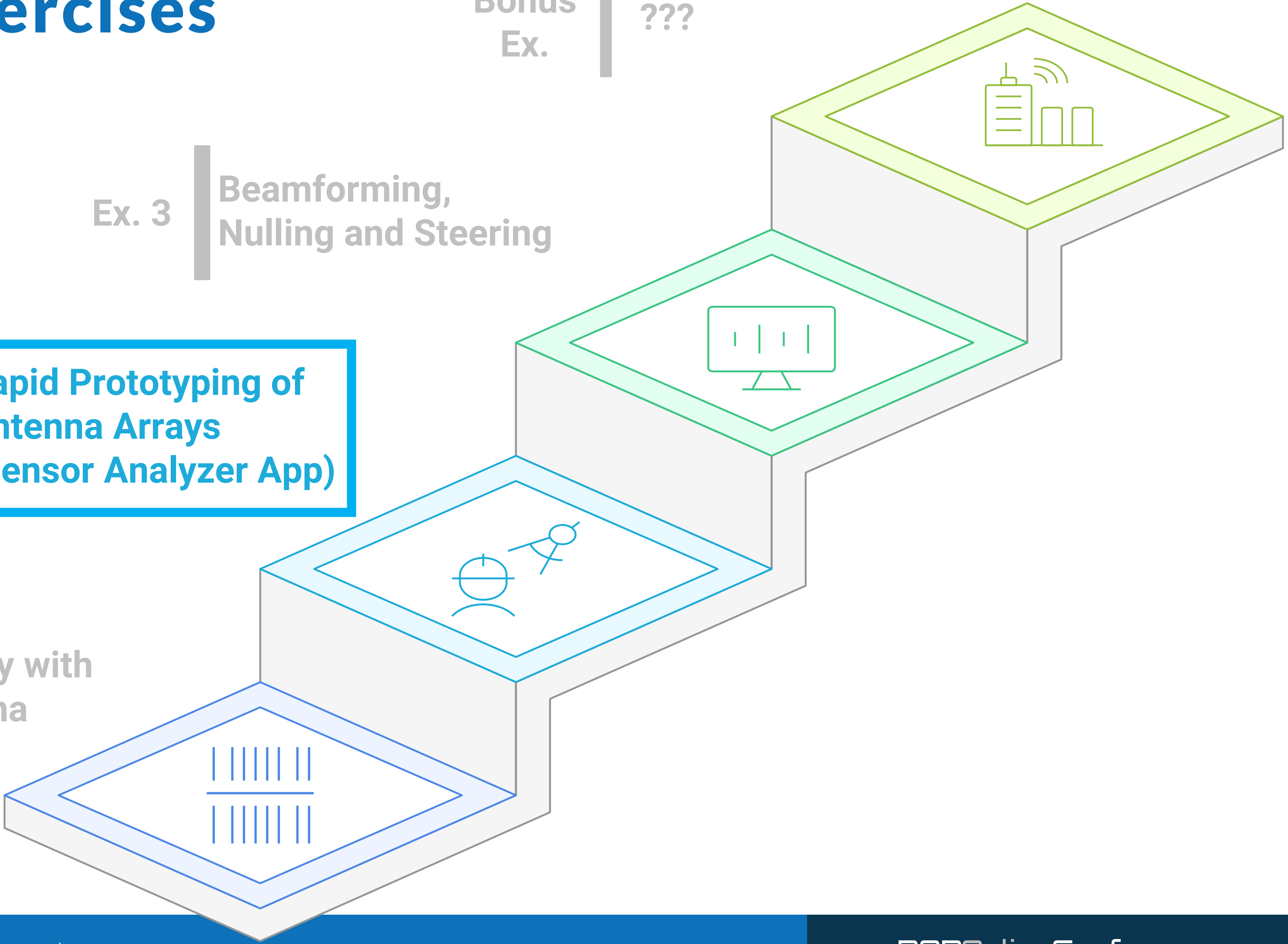
Overview of exercises

Bonus Ex. | ???

Ex. 3 | Beamforming, Nulling and Steering

Ex. 2 | Rapid Prototyping of Antenna Arrays (Sensor Analyzer App)

Ex. 1 | Create an array with custom antenna elements



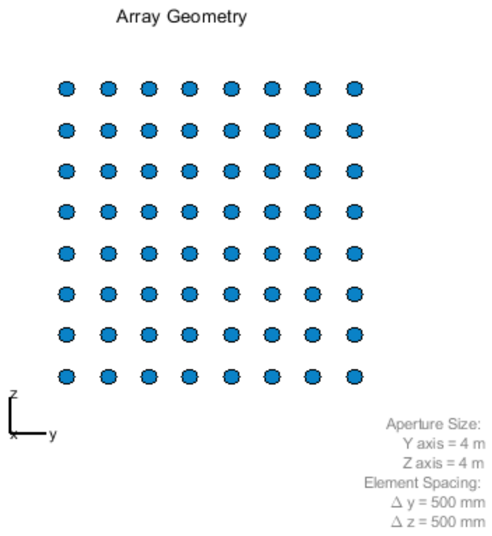
2

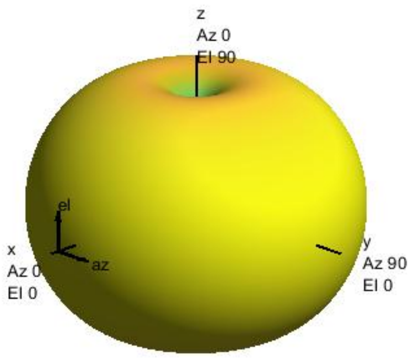
Rapid Prototyping of Antenna Arrays

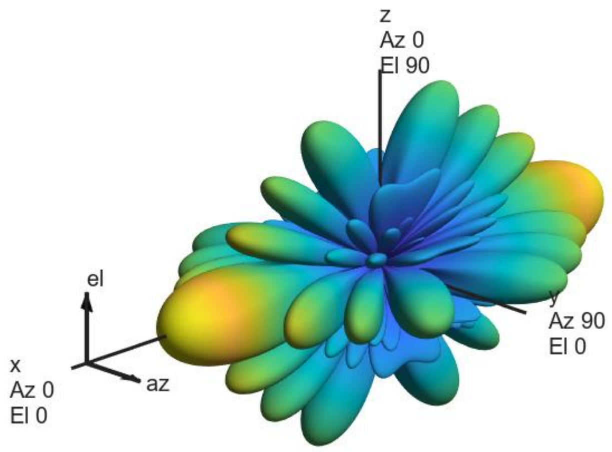
- Quick experiment with different array configurations
- View resultant beam pattern and 3D patterns
- Use the Sensor Array Analyzer App

Array Patterns are Determined by Element Pattern and Spatial Configuration

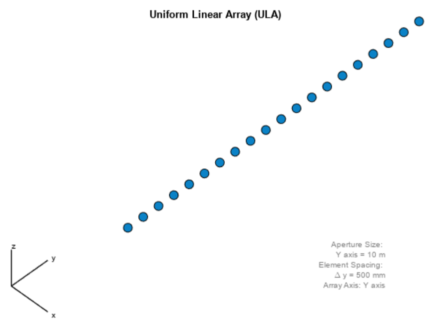
Array Factor × Element Pattern = Array Pattern



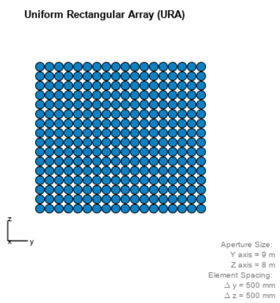




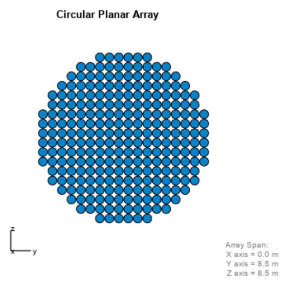
Options for Array Factor



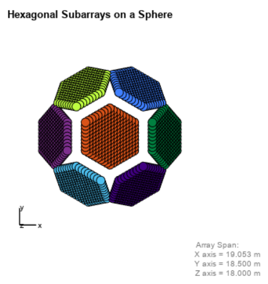
Uniform Linear Array



Uniform Rectangular Array



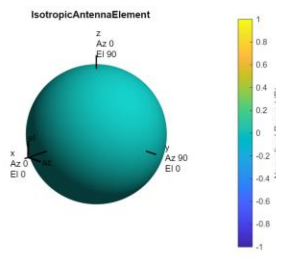
Circular Planar Array



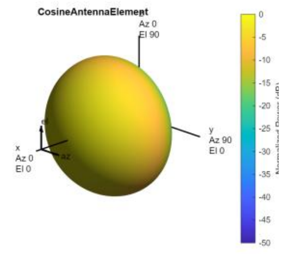
Custom Array Geometry

Array Factor × Element Pattern = Array Pattern

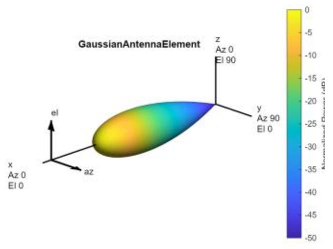
Options for element patterns



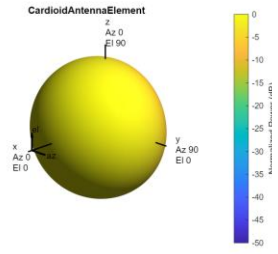
Isotropic Antenna



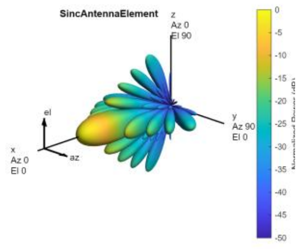
Cosine Antenna



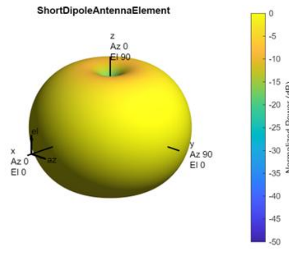
Gaussian Antenna



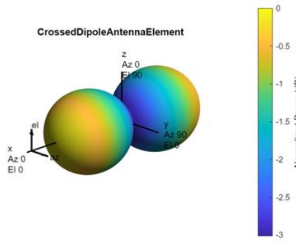
Cardioid Antenna



Sinc Antenna



Short Dipole



Crossed Dipole

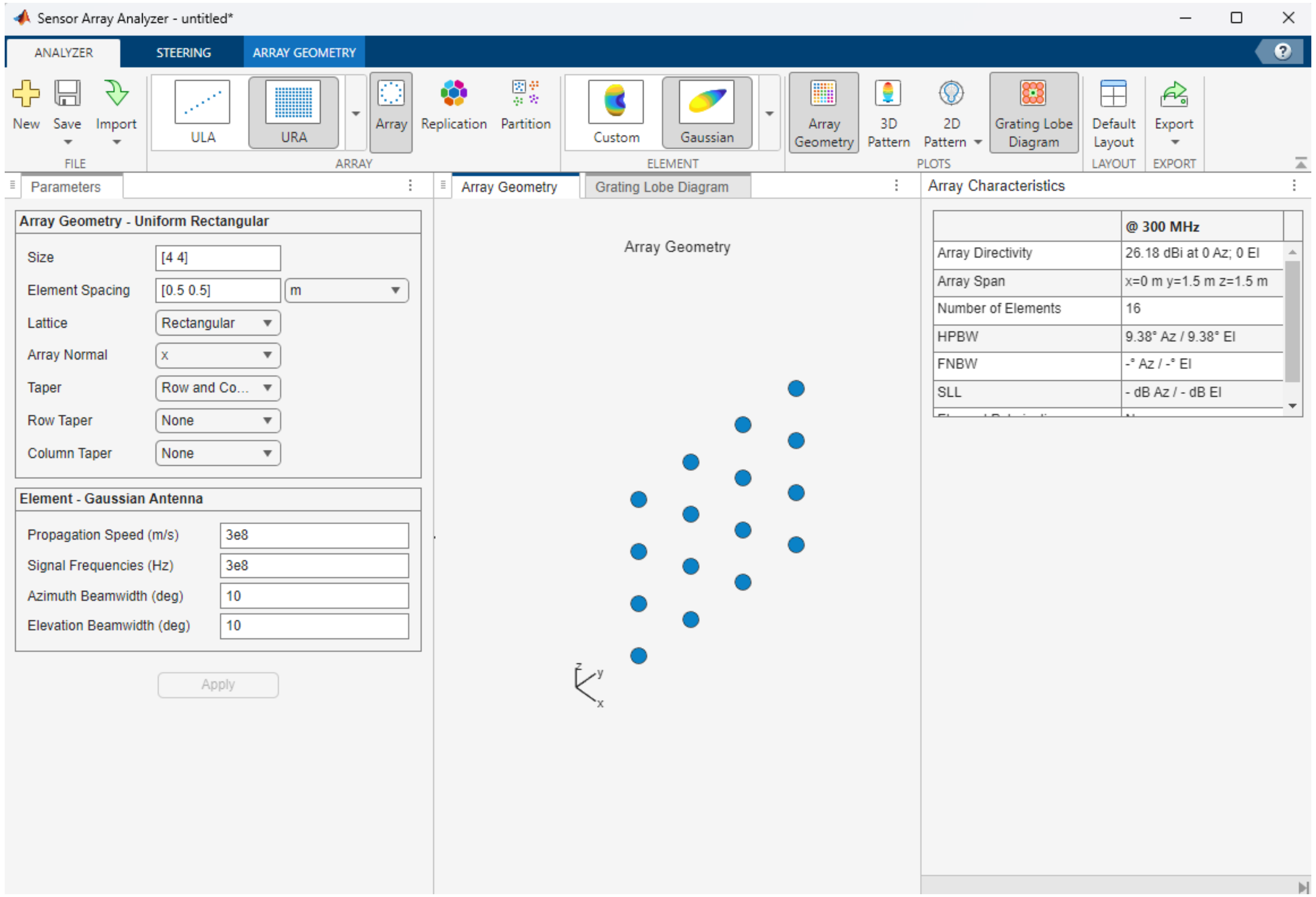
Array Factor × **Element Pattern** = Array Pattern

Exercise 2: Rapid Prototyping of Antenna Arrays

Open the Sensor Array Analyzer App

- Apps tab from MATLAB
- Command Line:

```
>> sensorArrayAnalyzer
```



Exercise 2: Rapid Prototyping of Antenna Arrays

Steps:

1. Experiment with different array factors and element patterns.
   - View 3D and 2D patterns

2. Design and export phased array with the following configuration:
   - Antenna Elements:
     - Isotropic
     - Back Baffled
   - URA 8 x 8
   - Spacing ½ λ
   - Signal Freq 2.5 GHz
   - Azimuth Steering Angle 15

3. Export to MATLAB

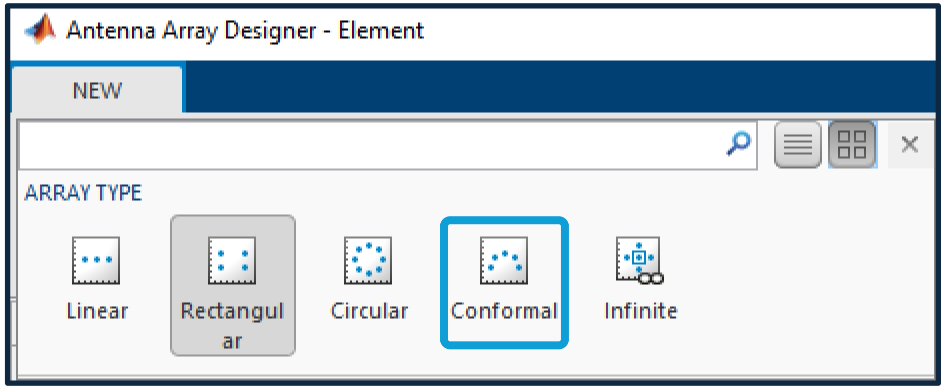
4. Extra Credit:
   - Find the effect of grating lobes and tapering
   - Partition array into subarrays

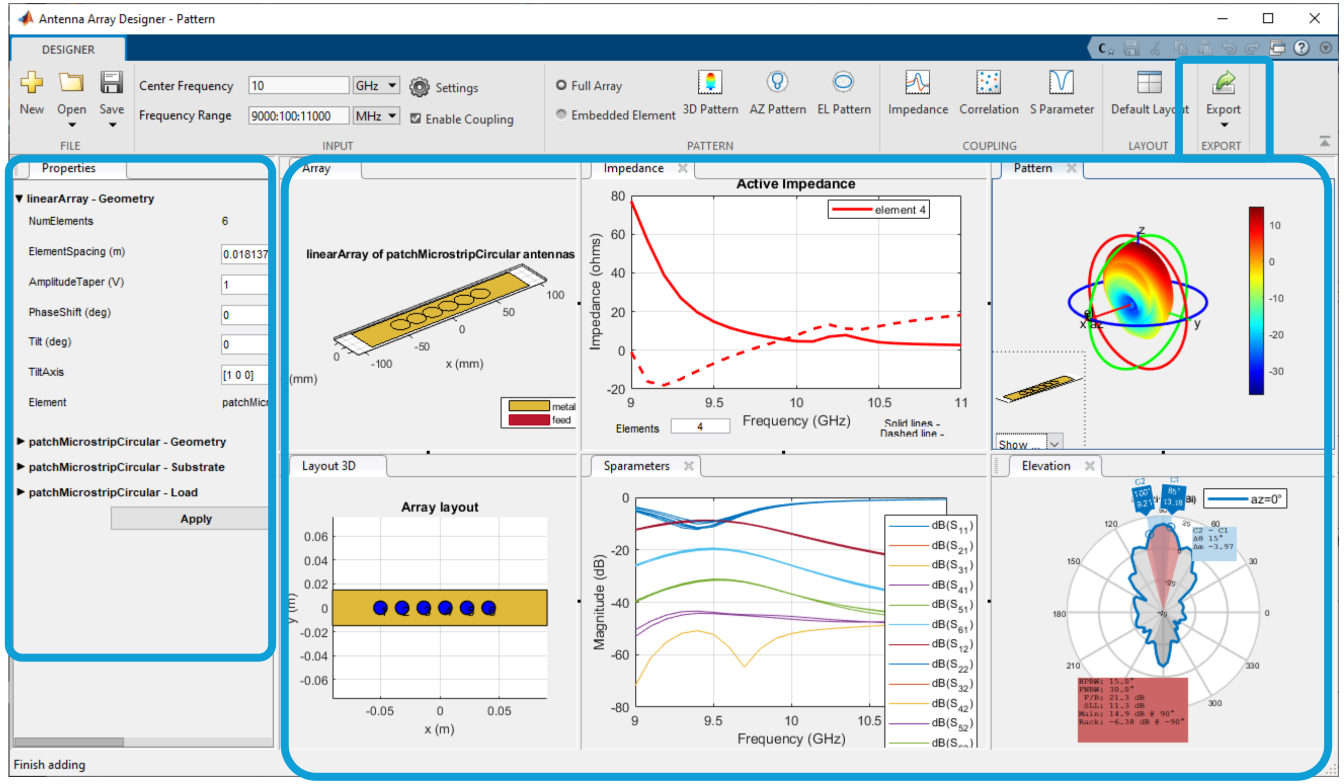
WORK ON EXERCISE 2 (15 minutes)

Related Resources

Antenna Toolbox:

- Full-wave and hybrid electromagnetic solvers for antenna elements and arrays
- Do port, surface, and field analysis, antenna placement studies, and radar cross section (RCS) calculations.
- Export designed antennas for fabrication using CAD and Gerber files





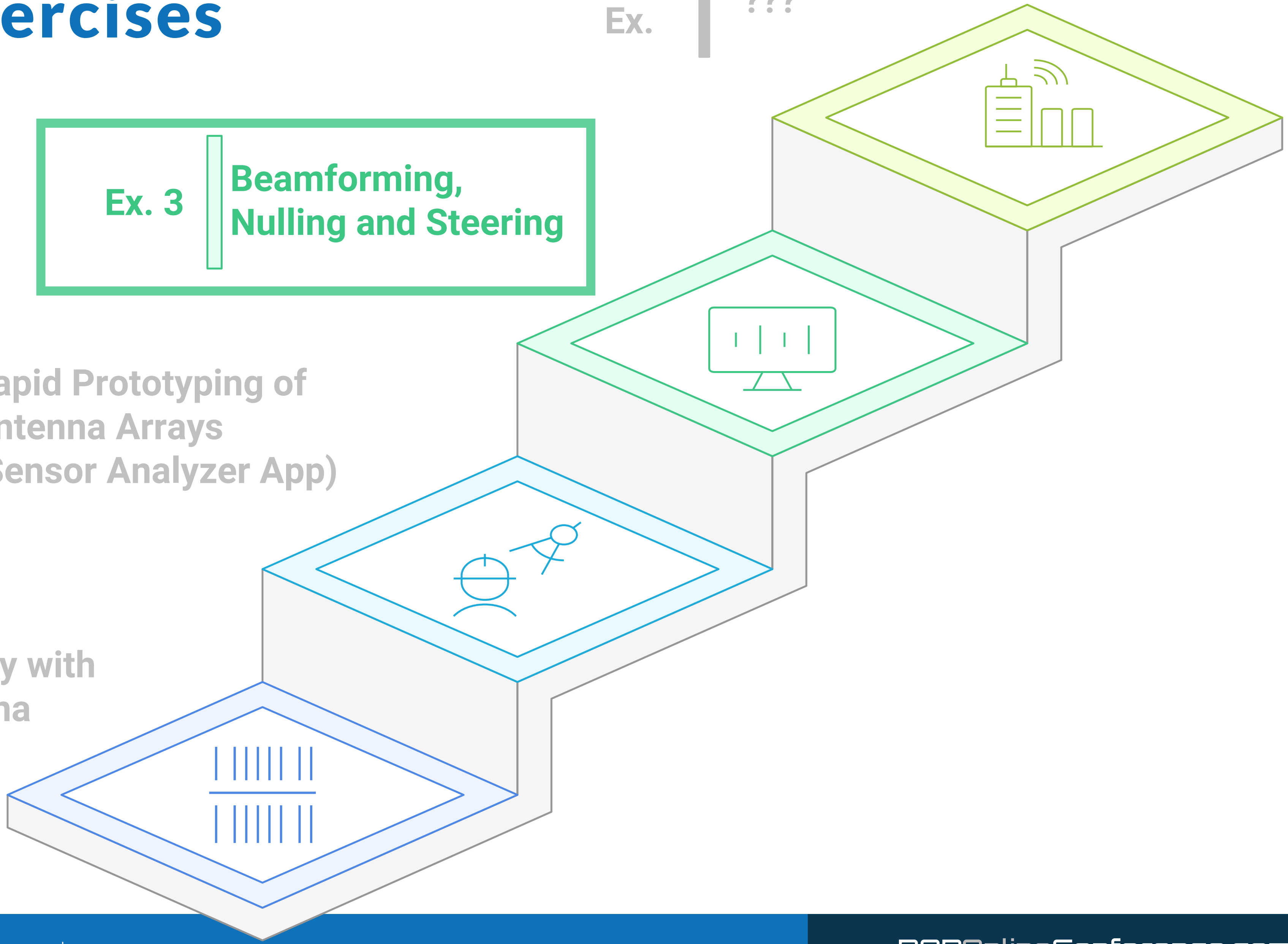
Overview of exercises

Bonus Ex. | ???

Ex. 3 | Beamforming, Nulling and Steering

Ex. 2 | Rapid Prototyping of Antenna Arrays (Sensor Analyzer App)

Ex. 1 | Create an array with custom antenna elements



3

Beamforming, Nulling and Steering

- Calculate weights for a ULA
- Meet given requirements for array performance
- Get familiar with minimum variance optimization
- Perform beamforming, nulling and steering

Phased Array Antenna Performance Specifications

Parameter	
Operating Frequency Bands and Bandwidth	e.g. Ku-band (12-18 GHz) Ka-band (27-40GHz)
Antenna Architecture and Element Design	Antenna element, polarization, number of elements
Beamforming and Steering Performance	FOV, Beamwidth, Side Lobe Levels
Gain	Gain and directivity

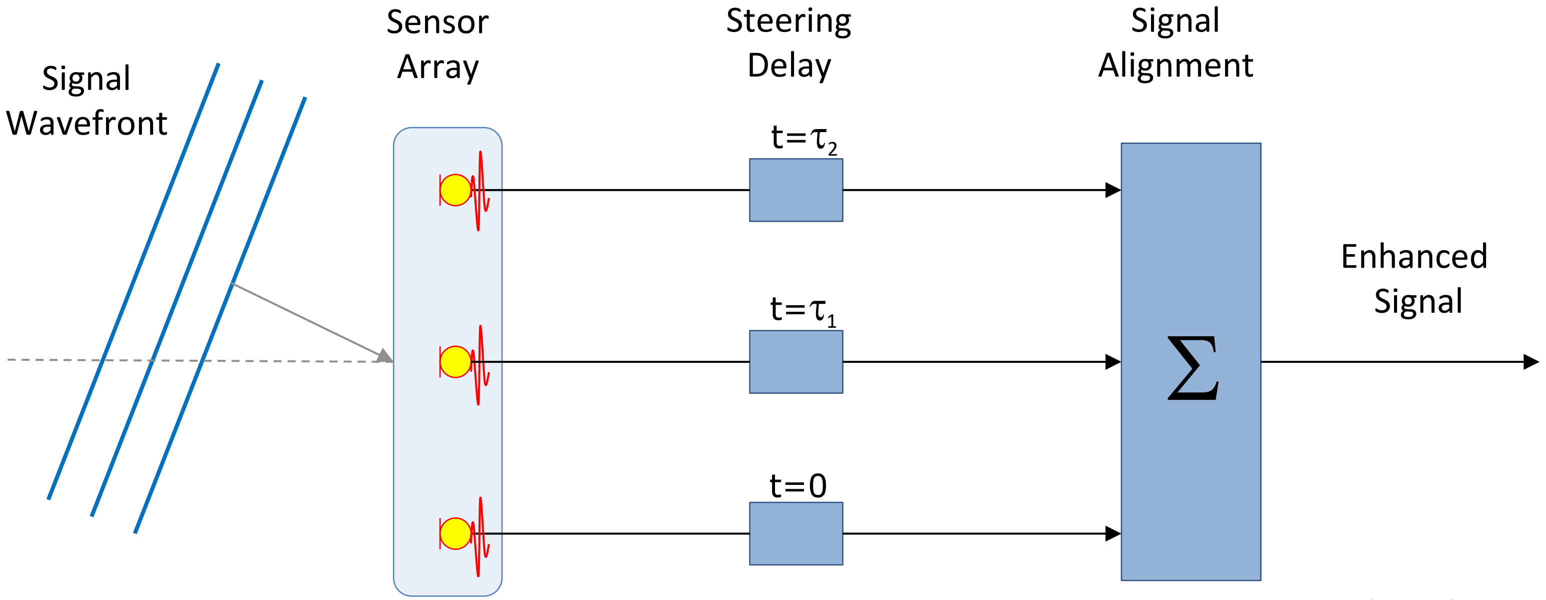
Starlink User Terminal Phased Array Antenna

Specification	Starlink High Performance
Dimensions (L x W x H)	22" × 20" × 1.6" (575 x 511 x 41 mm) [18]
Weight	16 lbs (7.2 kg)
Average Power Consumption (Active)	110-150W
Idle Power Consumption	45W
Field of View	140°
Orientation	Fixed
Primary Frequency Band	Ku-band (12-18 GHz)
Estimated Element Count	~1000-1200

https://www.starlink.com/us/specifications?spec=2

What is Beamforming and Nulling?

- Adjust weights of array elements to enhance signals from:
  - desired directions (steering)
  - suppress interference/noise from others (nulling)
- Conventional (fixed weights) vs Adaptive (respond to environment)
- Narrowband (constant phase shifts) vs Wideband (time delays)



www.mathworks.com/help/phased/beamforming.html

Supported beamformers in MATLAB

Conventional

Adaptive

Beamformer Name	Bandwidth	Processing Domain
phased.PhaseShiftBeamformer	Narrowband	Time domain
phased.TimeDelayBeamformer	Wideband	Time domain
phased.SubbandPhaseShiftBeamformer	Wideband	Frequency domain
phased.LCMVBeamformer	Narrowband	Frequency domain
phased.MVDRBeamformer	Narrowband	Frequency domain
phased.FrostBeamformer	Wideband	Time domain
phased.GSCBeamformer	Wideband	Time domain
phased.TimeDelayLCMVBeamformer	Wideband	Time domain
phased.SubbandMVDRBeamformer	Wideband	Frequency domain

mathworks.com/help/phased/ug/beamforming-concepts.html

Minimum Variance Beamforming

- Computes array weights that minimize output noise/interference while preserving gain in desired direction
- Mathematically it is an optimization problem
- Narrower main lobe, deeper nulls compared to conventional beamforming
- Adaptive: responds to interference environment
- When sidelobes are suppressed, the main lobe widens

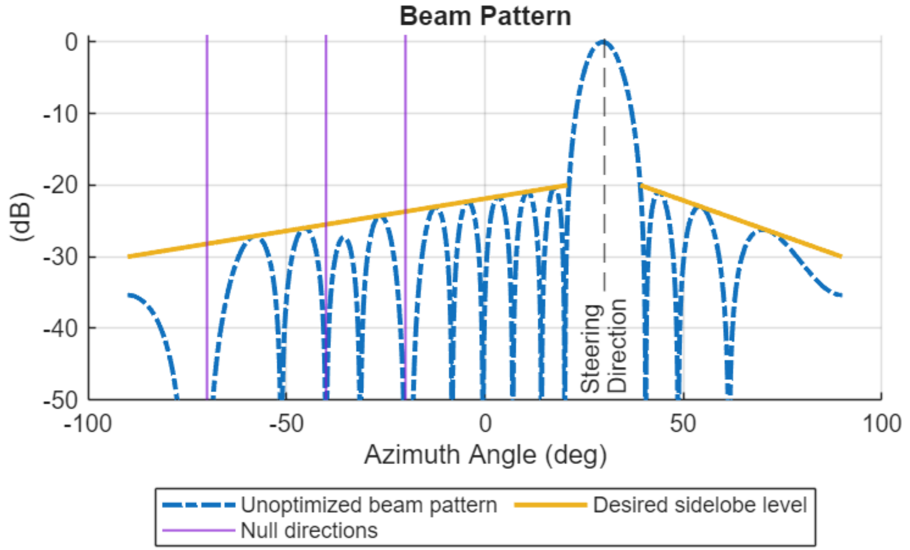
In MATLAB:

```
optimizedWeights = minvarweights(elementPositions, steeringAngle,...
    MaskAngle=maskAngles, MaskSidelobeLevel = maskValues, NullAngle=nullDirections);
```

Exercise 3: Beamforming, Nulling and Steering

- Steps:
  - Use minimum-variance algorithm to calculate array weights to meet these requirements:

Specifications	Details
Configuration and Size	Uniform Linear Array with 16 elements
Operating frequency	3 GHz
Main beam angle / Broad side (Az)	30°
Null angles	-70°, -40°, -20°
Side Lobe Level (SLL) Suppression	Linear from -20dB to -30dB (symmetric)



  - Plot and view the resultant beam pattern

WORK ON EXERCISE 3 (15 minutes)

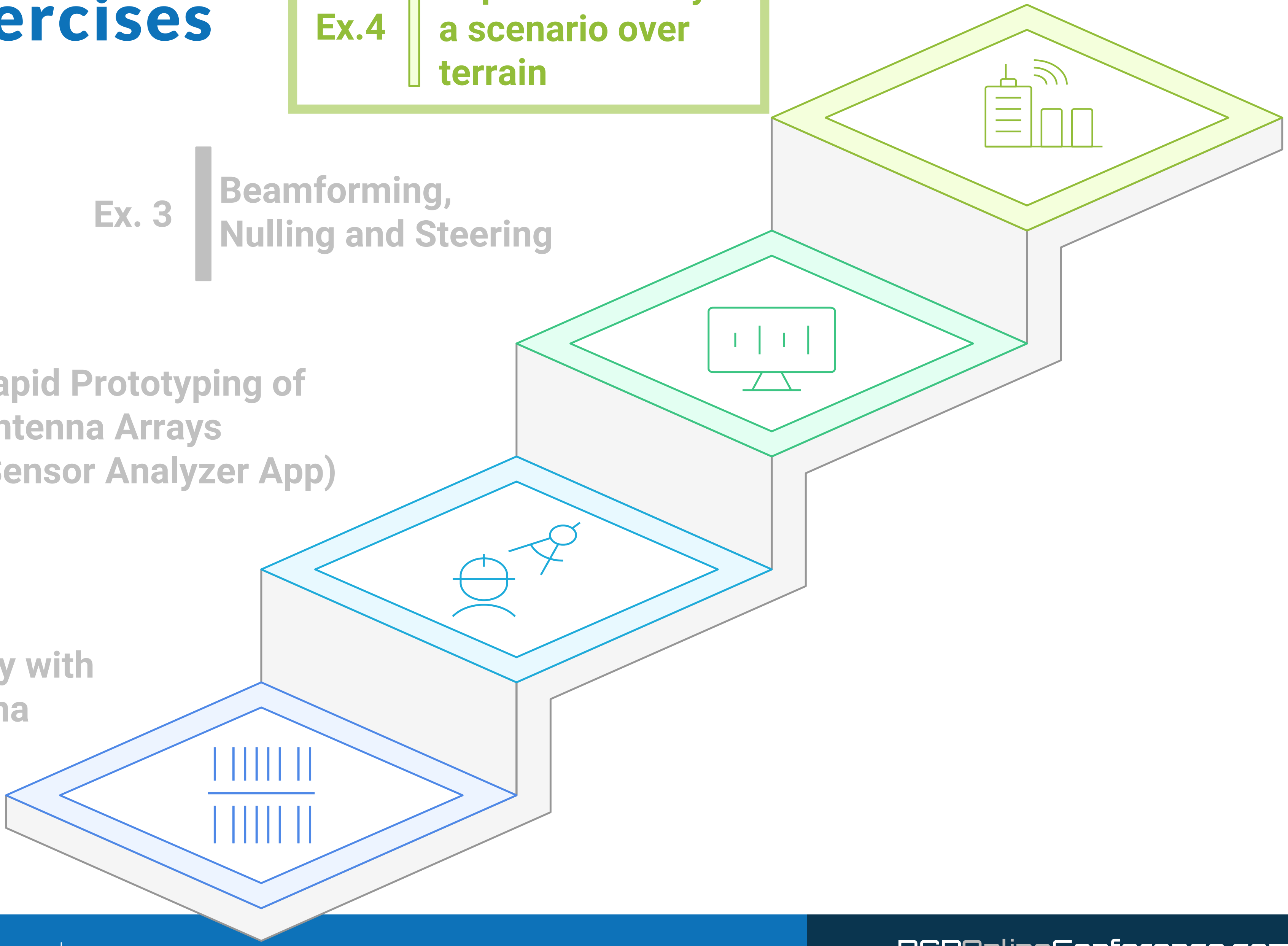
Overview of exercises

Ex.4 | Implement array in a scenario over terrain

Ex. 3 | Beamforming, Nulling and Steering

Ex. 2 | Rapid Prototyping of Antenna Arrays (Sensor Analyzer App)

Ex. 1 | Create an array with custom antenna elements

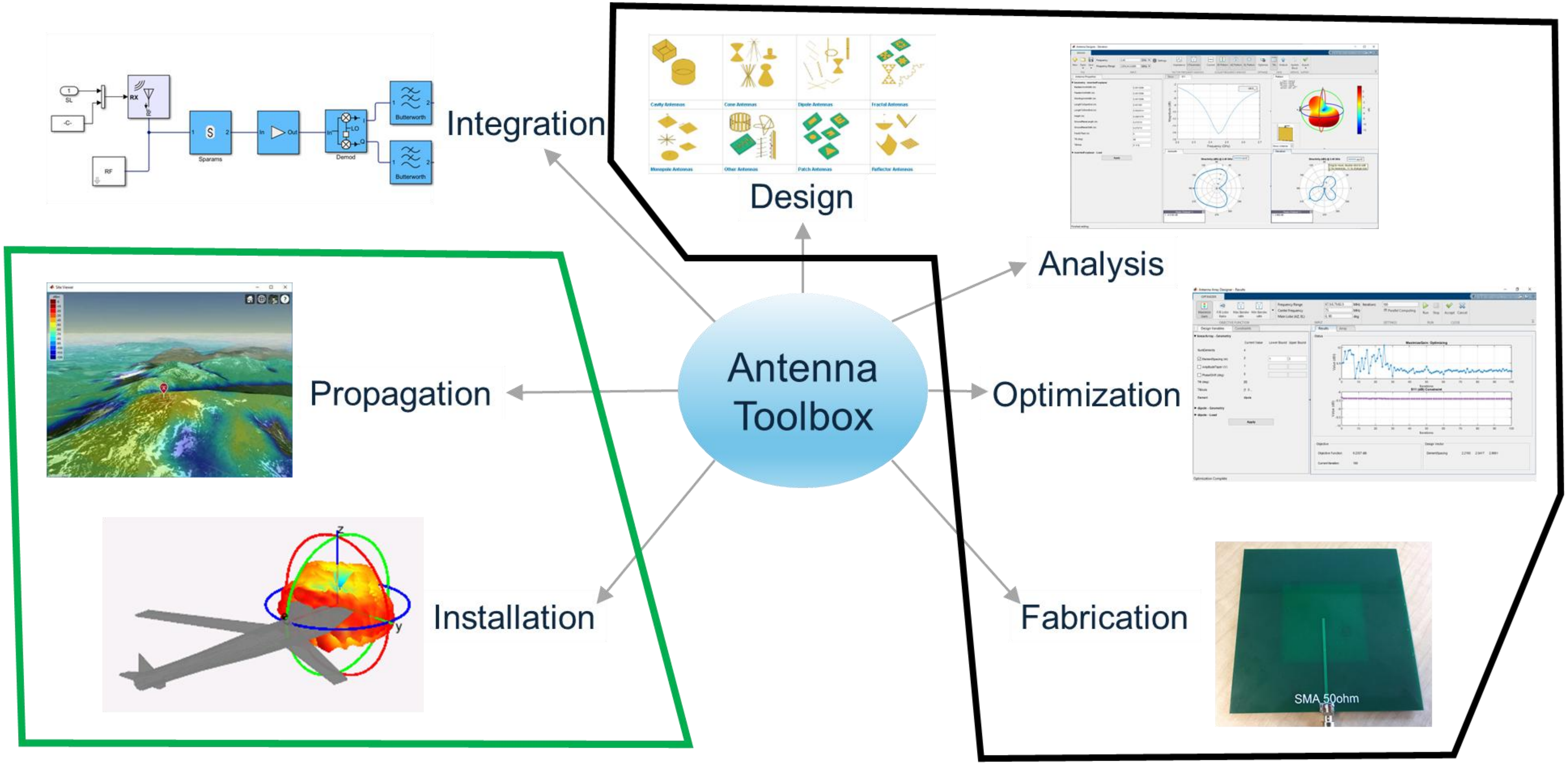


4

Implement array antenna in a scenario over terrain

- Deploy antenna array in real world scenario over a terrain
- Create RF Scenario using Antenna Toolbox Siteviewer (transmit and receive sites)
- Verify Line-of-Sight Link Visibility and measure received signal and interference power levels
- Implement Interference Nulling Workflow using MVDR Beamformer

What can I do with Antenna Toolbox?



RF Propagation Models in Antenna Toolbox

Atmospheric Models:

- **Free Space:** Ideal line-of-sight (LOS) propagation.
- **Rain, Gas, Fog:** Account for atmospheric attenuation due to weather conditions.

Empirical Models:

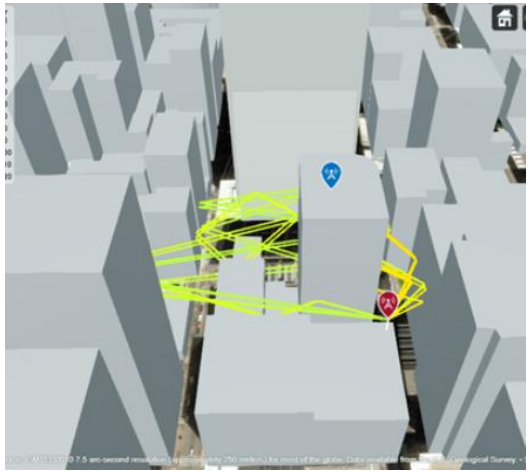
- **Close-In:** Statistical path loss for urban macro-cell scenarios, including non-line-of-sight (NLOS).

Terrain Models:

- **Longley-Rice (ITM) & TIREM:** Point-to-point path loss over irregular terrain, considering diffraction and ground reflection.

Ray Tracing Models (Advanced):

- **Shooting and Bouncing Rays (SBR):** Simulates reflections and diffractions (approximate path count, exact geometry).
- **Image Method:** Exact propagation paths with precise geometric accuracy for reflections.
- **Key Features:** Multipath analysis, material property considerations, polarization, path loss, phase shift, and GPU acceleration.



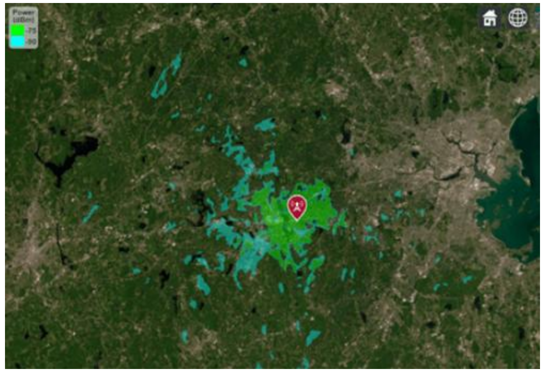
Visualization and Analysis with Antenna Toolbox

Interactive 3D Site Viewer:

- **Transmitter & Receiver Placement:** Easily define and place sites on a 3D globe or custom scene
- **Coverage Maps:** Visualize received signal strength (e.g., dBm), SINR, or link status
- **Propagation Path Visualization:** Plot individual ray paths showing reflections and diffractions
- **Terrain & Basemap Integration:** Incorporate real-world terrain and basemaps for realistic context

Analysis & Metrics:

- **Path Loss:** Calculate signal attenuation
- **Signal Strength & SINR:** Determine received signal quality
- **Link Status:** Evaluate communication link success
- **LOS/NLOS Analysis:** Determine visibility between sites



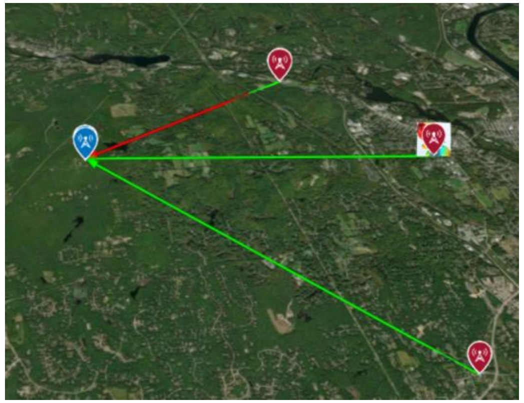
Integration & Customization:

- Define transmitter/receiver properties (txsite, rxsite).
- Configure propagation models (propagationModel).
- Programmatic control for automation and scripting.
- Customize building properties from OpenStreetMap files

Exercise 4: Deploy antenna array in real world scenario

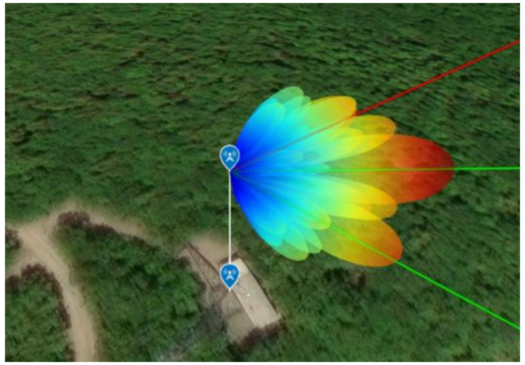
Create RF Scenario including the terrain (transmit and receive sites)

- Create RF receive site
- Create Tx Sites: 1 RF communication link and 2 interfering sites in 2.5 GHz Band



Verify Line-of-Sight Link Visibility

- Specify Receive and Transmit site antennas
- Use / Import 8-by-8 Receive Station Antenna Array
- Create 4-by-4 Tx Site Antenna Array



Calculate Received Signal Strength From Signal link and interference without Beamforming

Interference Nulling Workflow using MVDR Beamformer (next slide)

Exercise 4 steps : adaptive beamforming workflow

Define Antenna Array & Signal Scenario

Create Phased Array received signal

- Generate Tx signal waveform
- Add noise and interference signals
- Simulate received plane waves at array (using collectPlaneWave)

Create MVDR Beamformer object

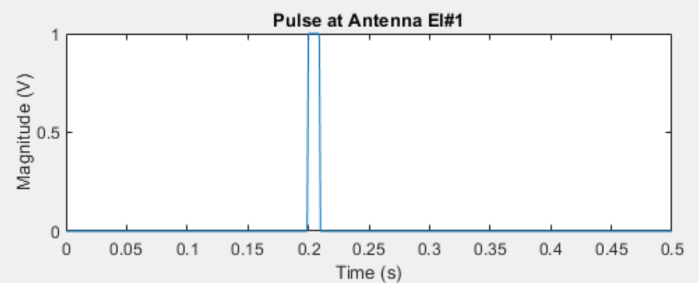
- Specify URA object, input angle

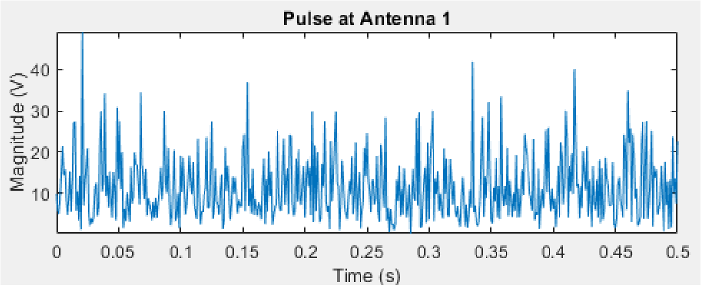
Apply MVDR Beamformer for received signal

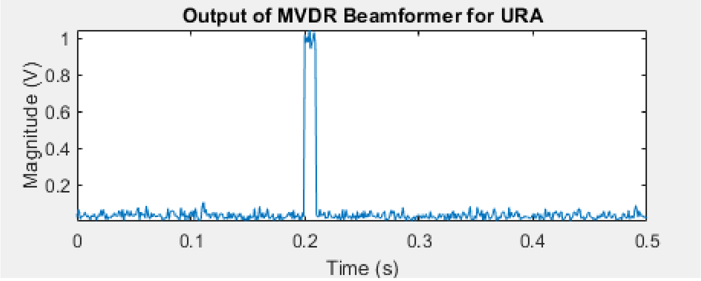
- Define Steering Vector for Desired Signal

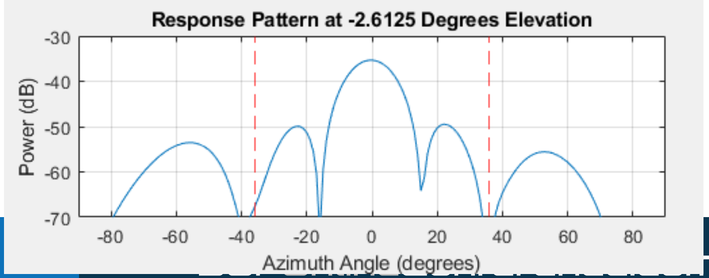
Plot Array Radiation pattern after beamforming

- Visualize the array pattern to observe nulls in interference directions
- compute received signal strength after beamforming









Exercise 4: functions used in exercise

Siteviewer

- Display transmitter sites, receiver sites, and RF propagation visualizations by using a siteviewer object.
  - Default mode: 3-D globe — Display sites that are referenced to geographic coordinates. You can customize the globe using custom terrain, high-zoom-level or custom basemaps, and buildings.

- txsite, rxsite
  - Display transmitter and receiver sites on a 3-D globe, calculate the distance and angles between the sites, and analyze the signal strength of the transmitter at the receiver site.

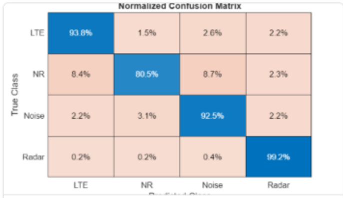
- los
  - Display or compute line-of-sight (LOS) visibility status

- sigstrength (rx,tx,"freespace")

https://www.mathworks.com/help/releases/R2024b/antenna/ref/siteviewer.html

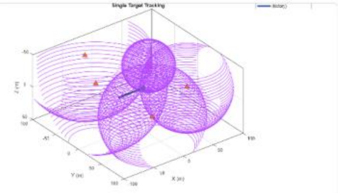
Other Key Topics (Not Covered Today)

- [Direction of Arrival Estimation and STAP]()
- [Detection, Range and Doppler Estimation]()
- [Waveform Design and Signal Synthesis]()
- [Radar and Wireless Coexistence]()
- [Localization]()



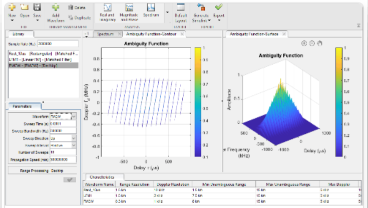
Spectrum Sensing with Deep Learning for Radar and Wireless Communications

Train a deep learning neural network using deep learning to identify radar and wireless communication signals in the air.



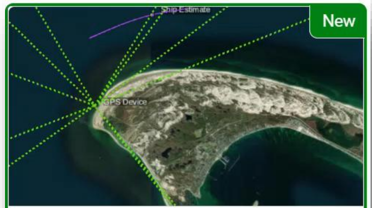
802.11az Three-Dimensional Tracking Using Time of Arrival Estimation

Use an IEEE 802.11az Wi-Fi network to track Wi-Fi devices in a three-dimensional space using time of arrival (TOA) estimation.



Compare Ambiguity Functions for Different Wave Modulation Schemes

Visualize and interpret different waveform processing schemes and their tradeoffs in the Pulse Waveform Analyzer app.



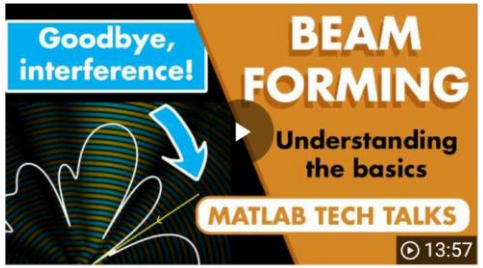
Multistatic Localization of a Ship Using GPS Illuminations

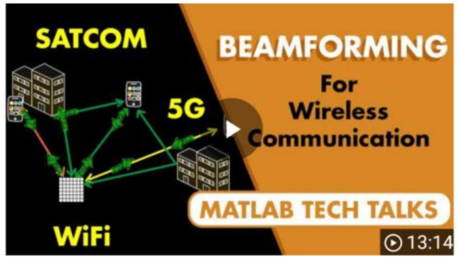
You locate a ship using received GPS signals reflecting off the ship.

Since R2025a

Related Resources:

- [Get Started with the Phased Array System Toolbox]
- [MATLAB Tech Talks Series on Phased Arrays and Beamforming]





- [2-Day Course: Modeling Radar Systems with MATLAB]

Modeling Radar Systems with MATLAB

Develop and optimize radar models with MATLAB using Radar Toolbox.

ADVANCED

Wireless Communications Systems Design with MATLAB and USRP...

Design single- and multicarrier digital communication systems, create multi-antenna and turbo-coded communication systems,...

INTERMEDIATE

Summary

Skills You Gained Today:

- Built foundational skills in modeling and simulating phased array systems using MATLAB
- Designed and visualized custom array configurations
- Leveraged interactive tools like the Sensor Array Analyzer for rapid design evaluation
- Applied optimization techniques to steer beams and enhance array performance
- Explored a streamlined workflow from antenna import to performance tuning

Next Steps in Your Learning Journey:

- Explore documentation and example workflows to deepen your understanding
- Take the next step with online training modules and self-paced courses on array systems, radar, and wireless
- Practice and prototype on your own. Explore bonus exercise!

THANK YOU

DSP
Online
Conference

www.dsponlineconference.com

DSP
Online
Conference

w w w . d s p o n l i n e c o n f e r e n c e . c o m