Radar System Design Using MATLAB and Simulink

김용정 부장(James.kim@mathworks.com)
Application Engineer
MathWorks

© 2016 The MathWorks, Inc.

Outline

- Introduction
- Radar System Design and Analysis
- ADAS Radar Systems
- Increasing the fidelity of RF and Antenna models
- Summary

Radar System Design: From Antenna to Algorithms

Antenna, Antenna arrays
type of element, # elements, configuration

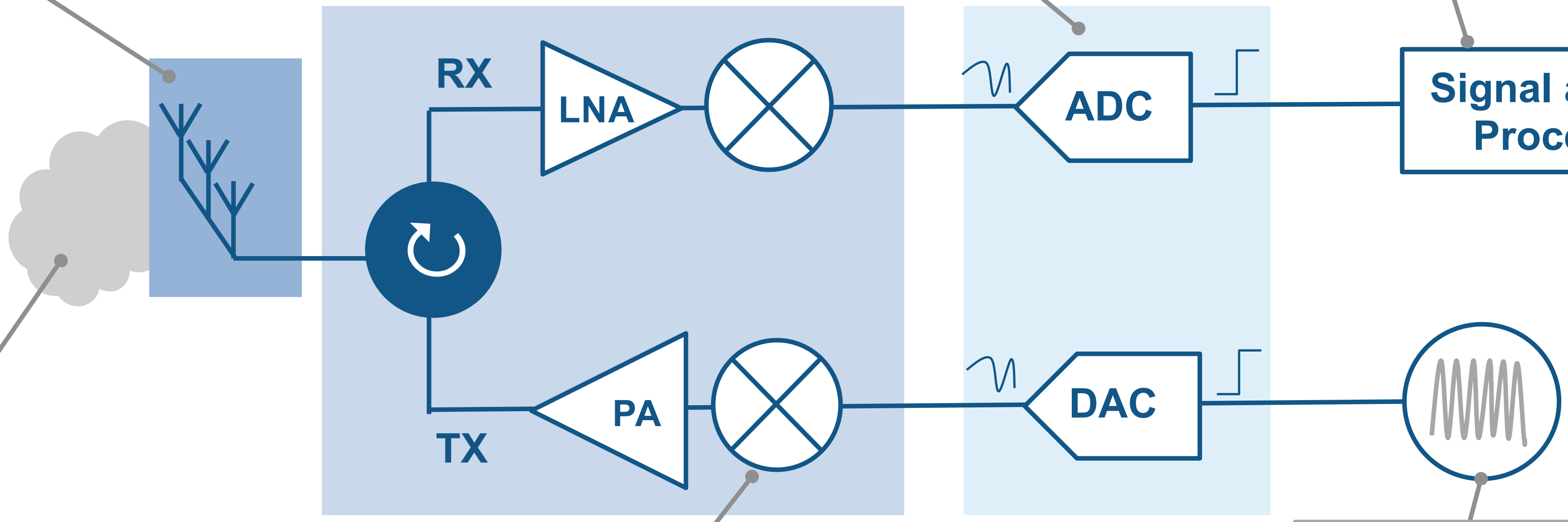
- Antenna Toolbox
- Phased Array System Toolbox

Mixed-Signal
Continuous & discrete time

- Simulink (Simscape)
- DSP System Toolbox
- Control System Toolbox

Algorithms
beamforming, beamsteering, MIMO

- Phased Array System Toolbox
- Communications System Toolbox
- DSP System Toolbox



- Communications System Toolbox
- Phased Array System Toolbox

Channel
interference, clutter, noise

- SimRF
- RF Toolbox

RF Impairments
frequency dependency, non-linearity, noise, mismatches

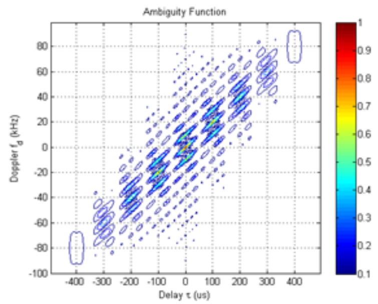
- Phased Array System Toolbox
- Signal Processing Toolbox
- Instrument Control Toolbox
- SimEvents

Waveforms and Task Scheduling

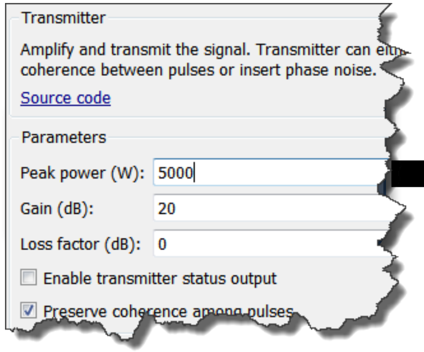
Radar System Design with Phased Array System Toolbox

- Specify radar requirements
- Research and design new algorithms
- Test algorithms with realistic data
- Simulate end-to-end radar systems

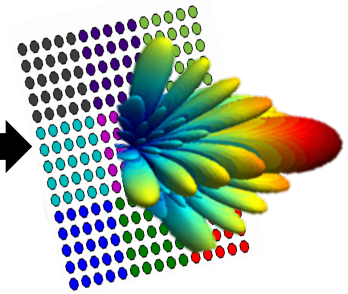
Waveforms
Pulse, LFM, FMCW, etc.

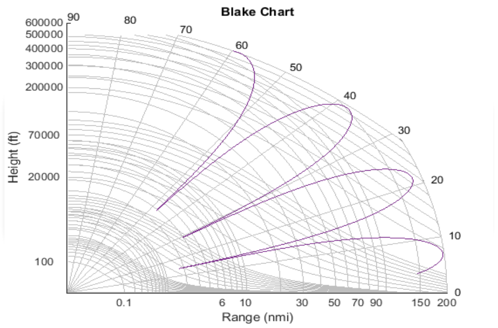


Transmitter
Monostatic and Bistatic

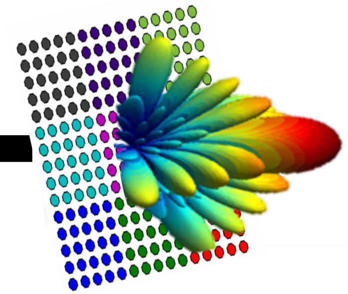


Tx Antenna Arrays
ULA, URA, UCA, etc.

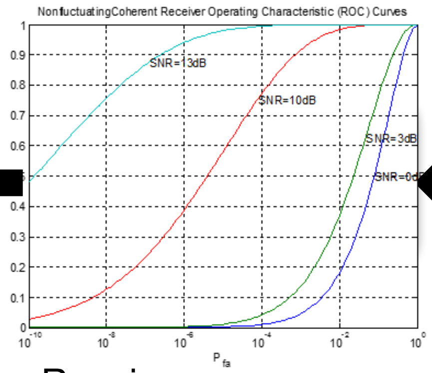




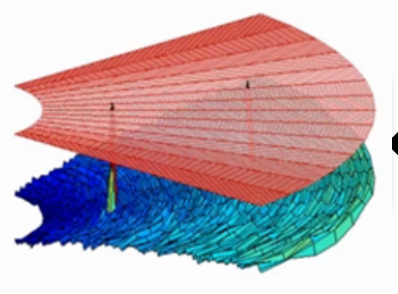
Environment effects,
impairments, interferences



Rx Antenna Arrays
ULA, URA, UCA, etc.



Receiver
Monostatic and Bistatic



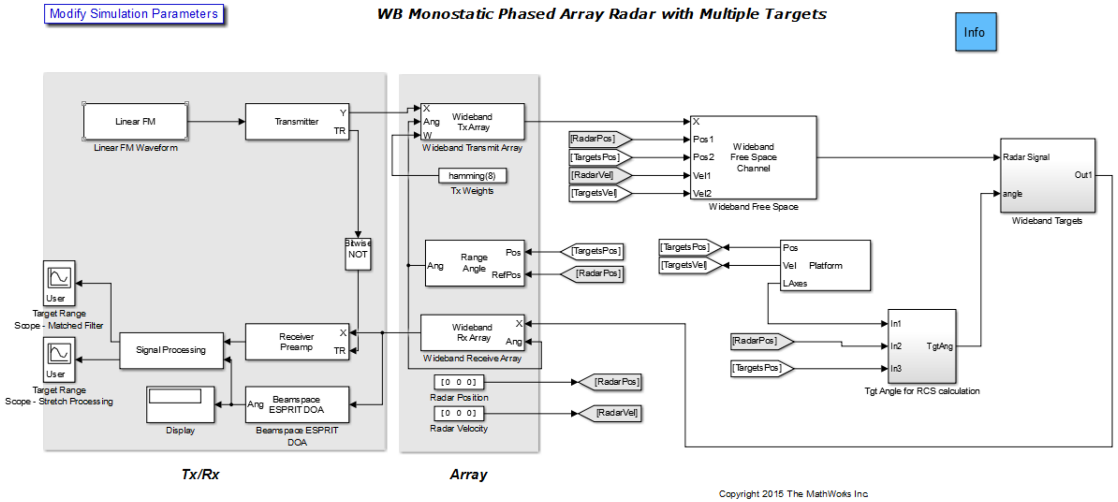
Beamforming, Matched
Filtering, Detection, CFAR,
STAP, etc.

Benefits of Flexible Modeling and Simulation Framework

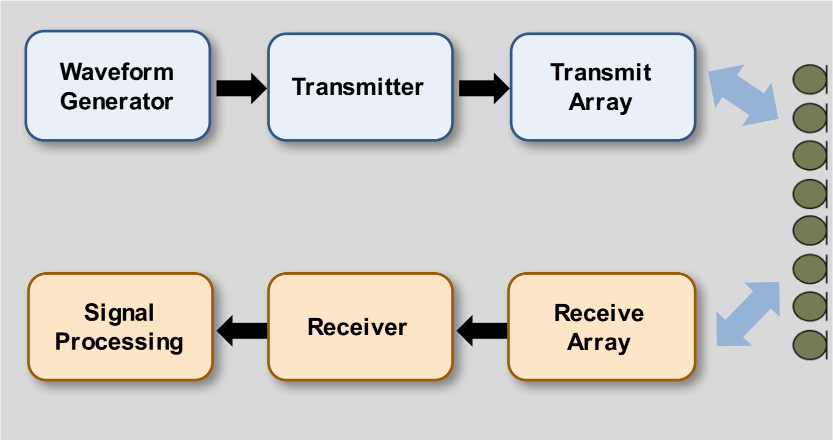
- Rapidly model and simulate phased array systems in the MATLAB and Simulink environments
  - Interactive development with algorithms and tools specifically for phased array systems
  - Explore alternative system architectures and make system level trade-offs
  - Access to MATLAB's visualization and analysis tools
  - Capture system requirements in an executable model
  - Common "language" and interface across teams and projects

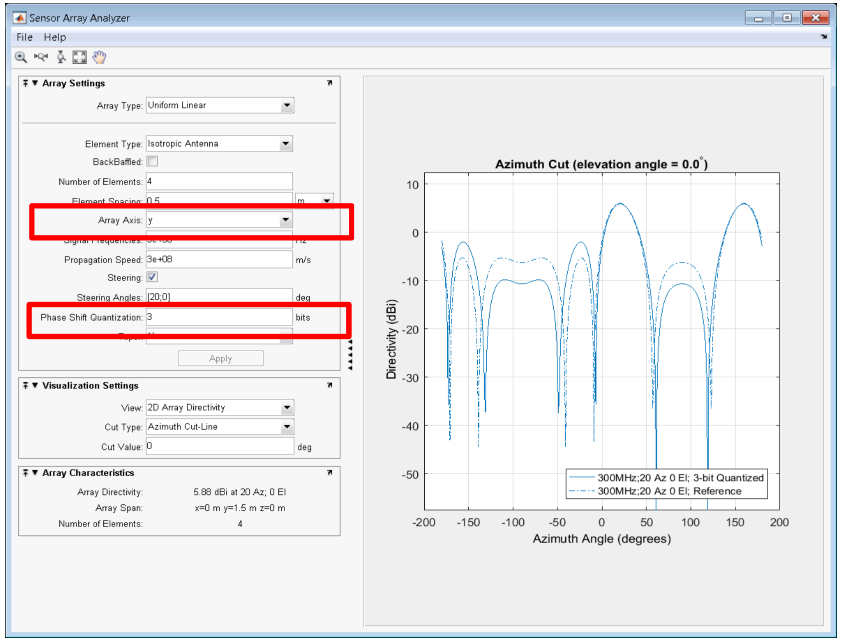
- Re-use and extend existing code and IP
  - Use existing C, MATLAB, and other code
  - Open API lets you include proprietary target models and environment models

- Process offline data in the same environment
  - Re-use the same algorithms and test benches



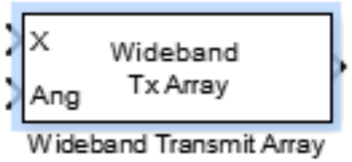
What’s New in Radar Modeling?





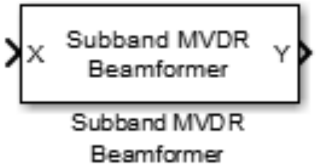
R2016a

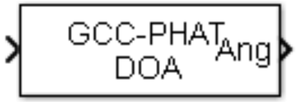
Array orientation & Phase shift quantization

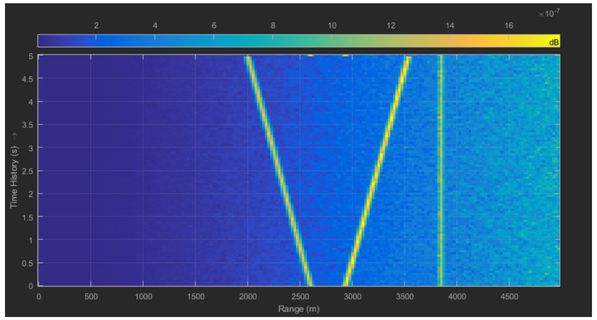


Wideband support

R2015b





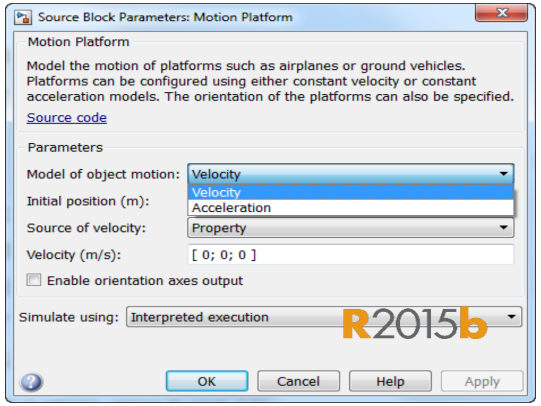


R2016a

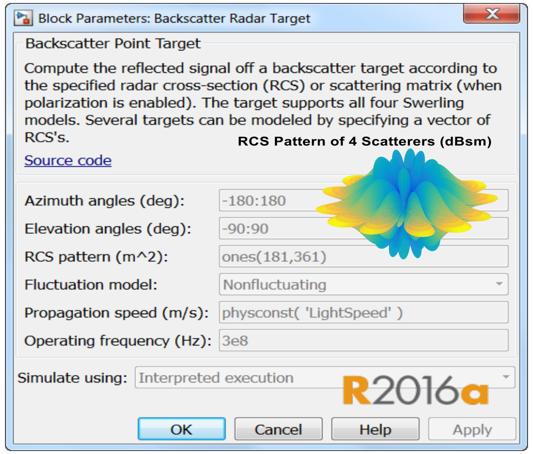
Range-time Intensity
Doppler-time Intensity

What's New in Targets, Platform and Environment?

Platform acceleration

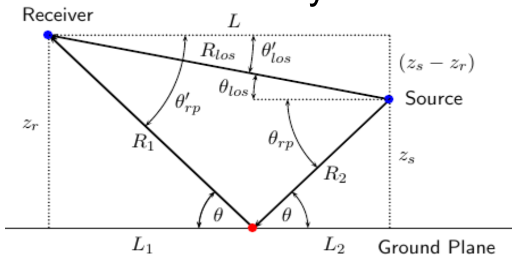


Angle dependent backscatter



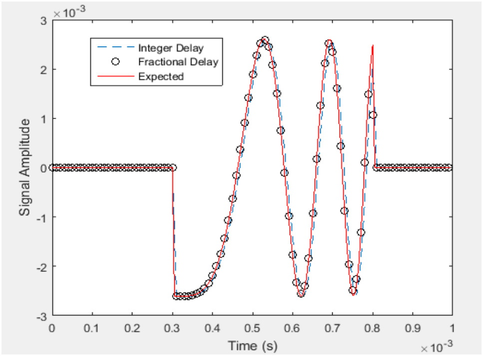
Targets & Platforms

Two-ray



R2015b

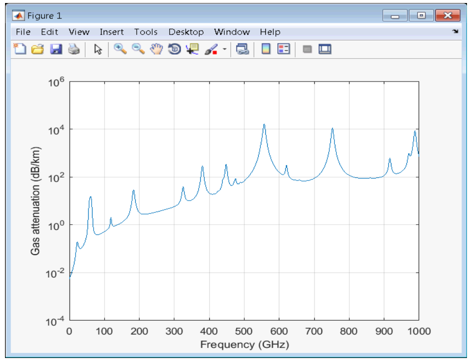
Fractional Delay



R2015b

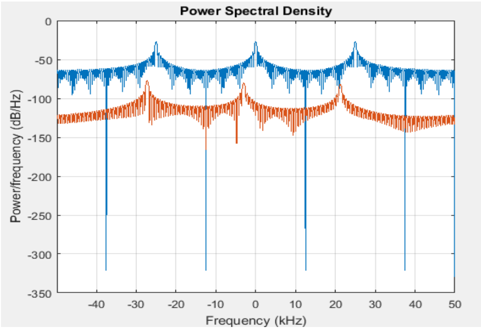
Propagation Path

Rain, fog and gas



R2016a

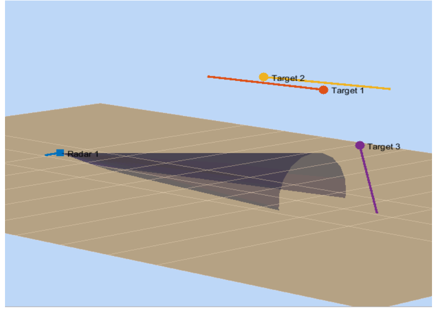
Wideband free space



R2015b

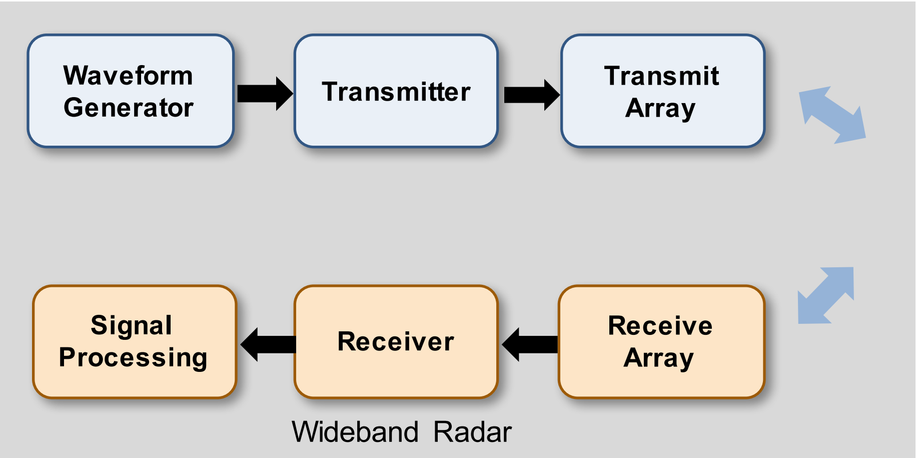
Environment

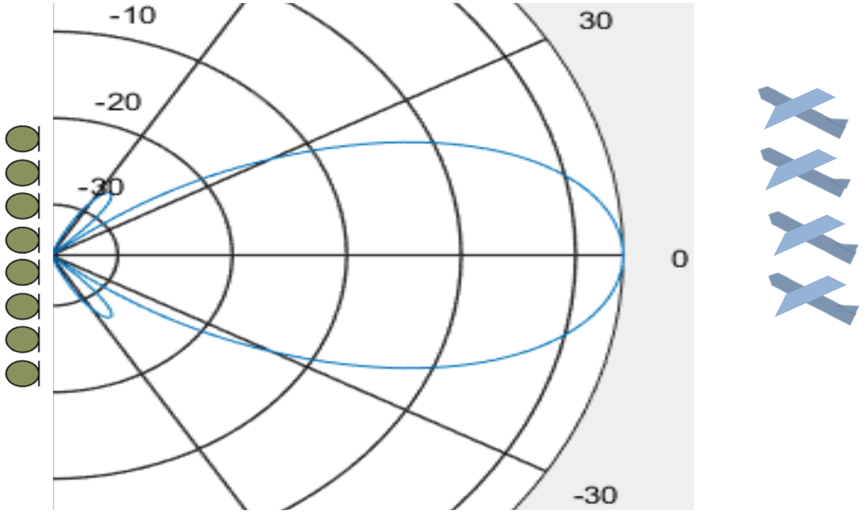
R2016a



Scenario Viewer

End-to-End Wideband Radar System – Example Summary





- Tx/Rx Array
  - Transmitter, Receiver, Antenna array
- Propagation channel
  - Path loss, environment
- Signal Processing
  - Beamformer, Matched filter, Stretch processing and Integration

Built with blocks from Phased Array System Toolbox

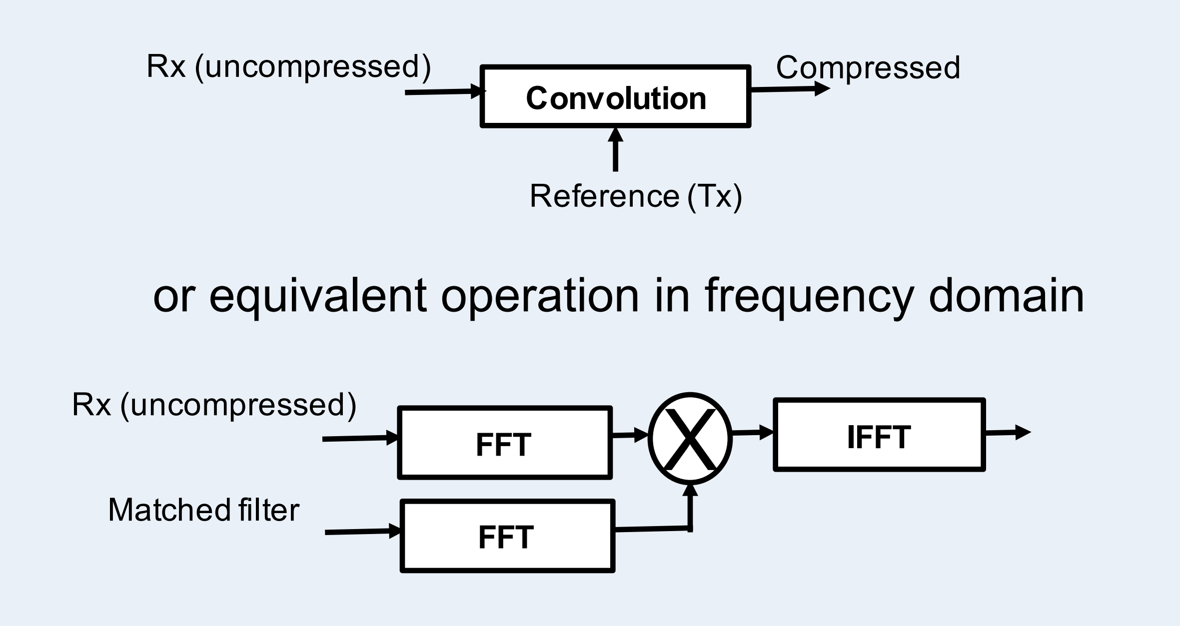
- Target model
  - Custom function with angle and frequency inputs

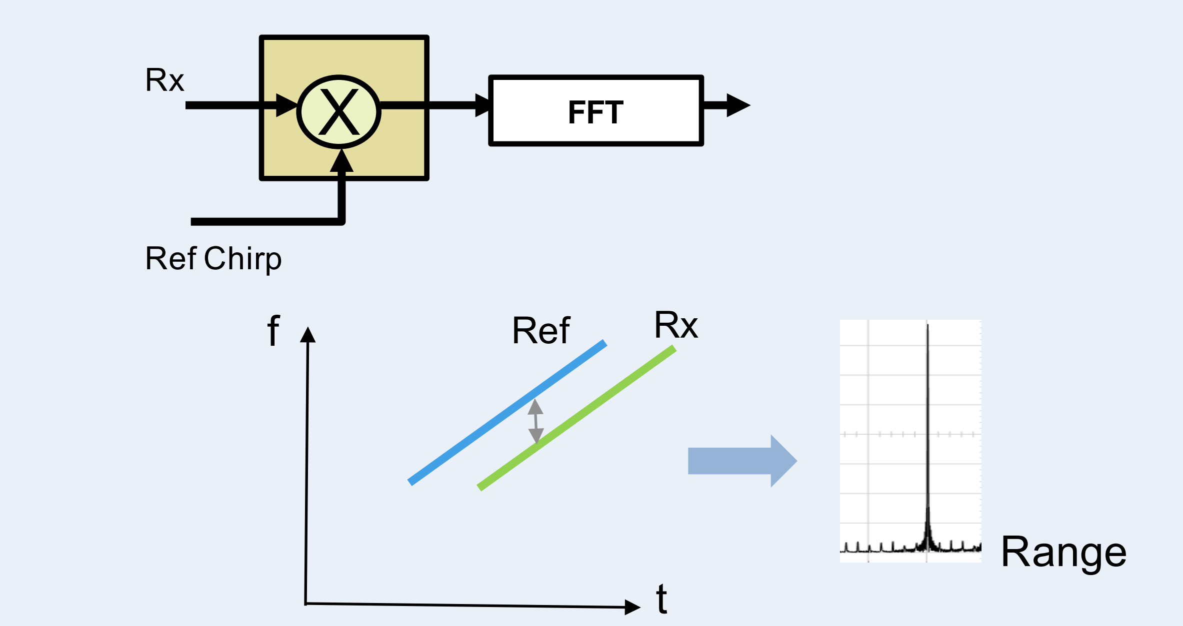
User-defined MATLAB function

Pulse Compression in the Wideband Example

- Pulse Compression
  - Maximizes peak SNR
  - Convolution in time domain with matched filter
  - Multiply in frequency domain with matched filter

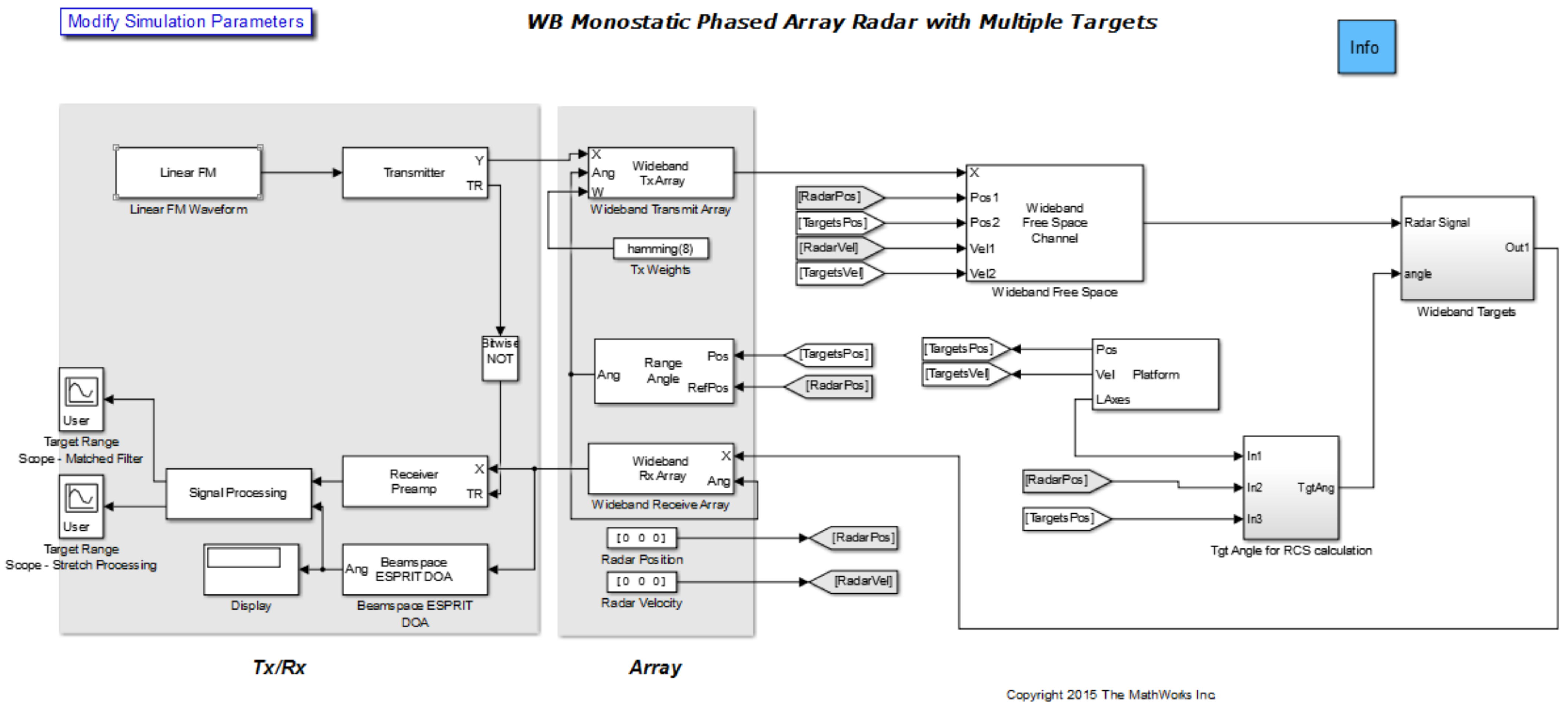
- Stretch Processing
  - Apply dechirp processing to a linear FM pulse over a given range span.
    - Resulting sinusoid translates directly to range
  - Alternative for LFM pulses with large bandwidth signals





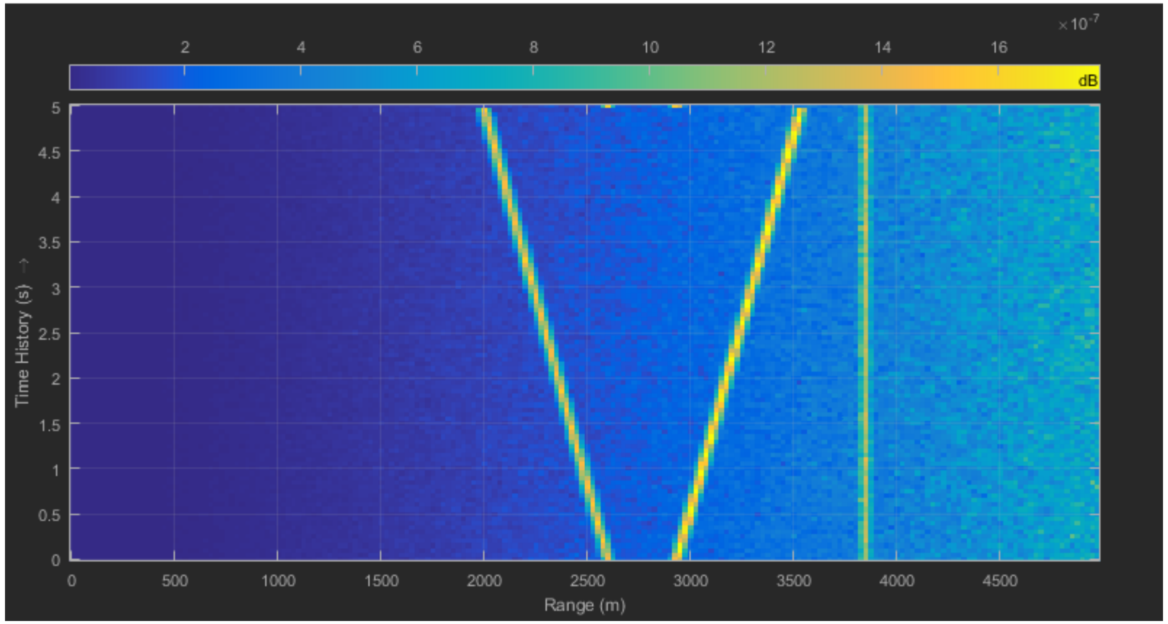
Wideband Simulink Models

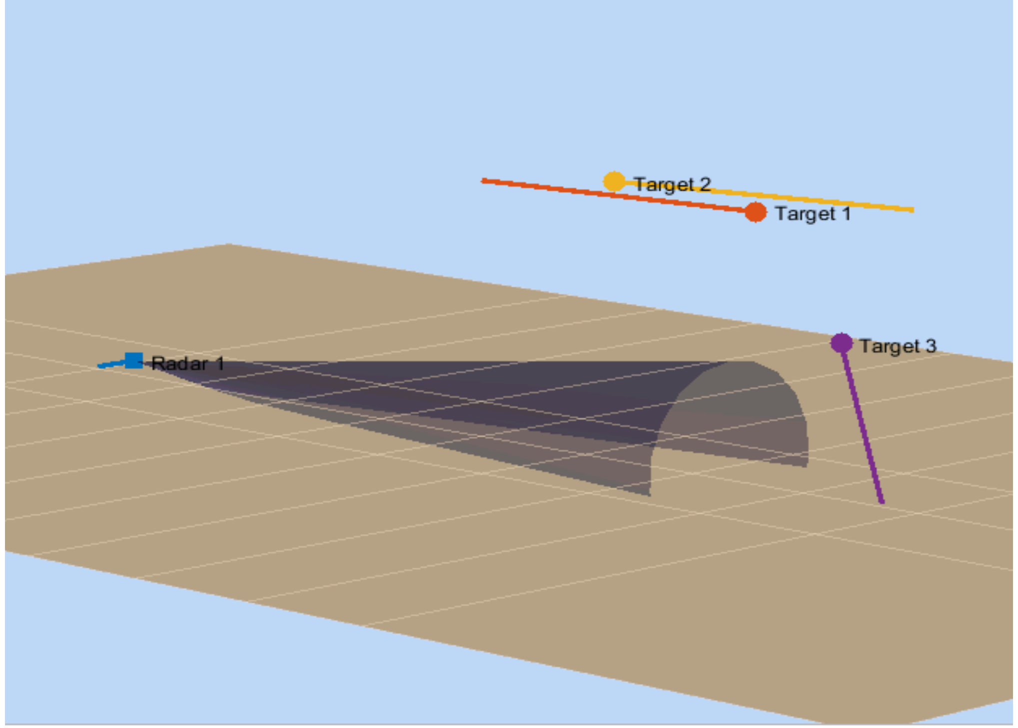
- End-to-End Wideband System Demo



Visualize Radar and Target Trajectories

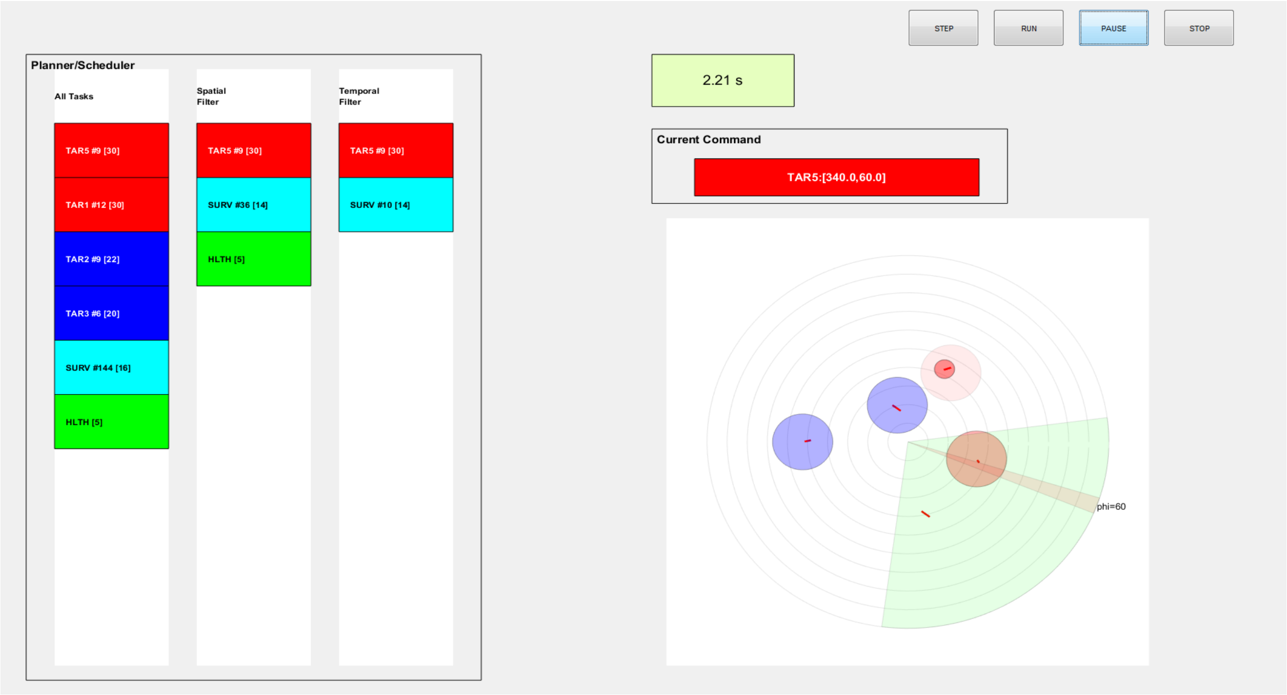
- Demo



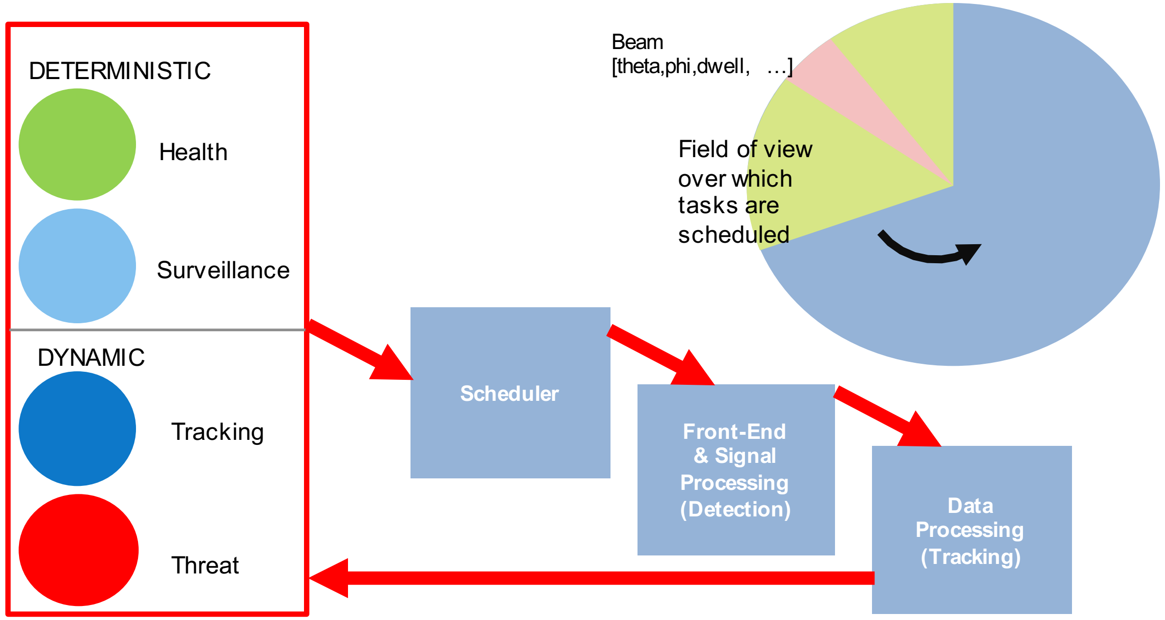


Modeling a Radar Task Scheduler

- Demo



Radar Scheduler Using SimEvents



ADAS Radar

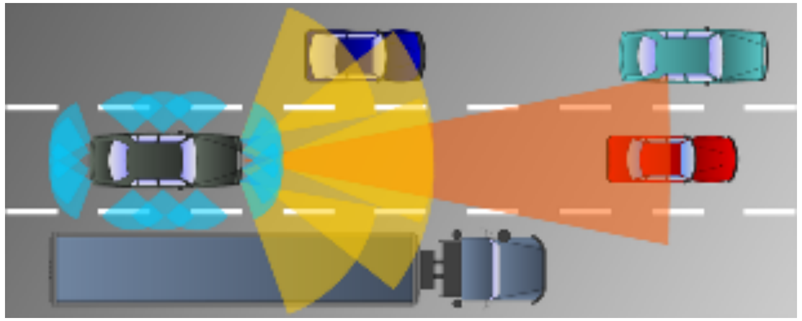
- Modeling an ADAS Radar System
- Long Range Radar Model - Adaptive Cruise Control (ACC)
- Short Range Radar Model - Blind Spot Detection

Background and Motivation

- Consumer demand and legislation drives safety improvements every model year
- Sensor accuracy requirements increase as “passive” safety evolves to “active” system control
- Momentum of autonomous vehicles accelerates ADAS technology

- Reasons for using radar in ADAS:
  - Range and velocity precision
  - Angular resolution and width of view
  - Conditions that impact camera vision do not impact radar view

- Short range radar characteristics
  - Wide bandwidth enables high range resolution at short ranges.
- Long-range radar characteristics
  - Range distance and beam width

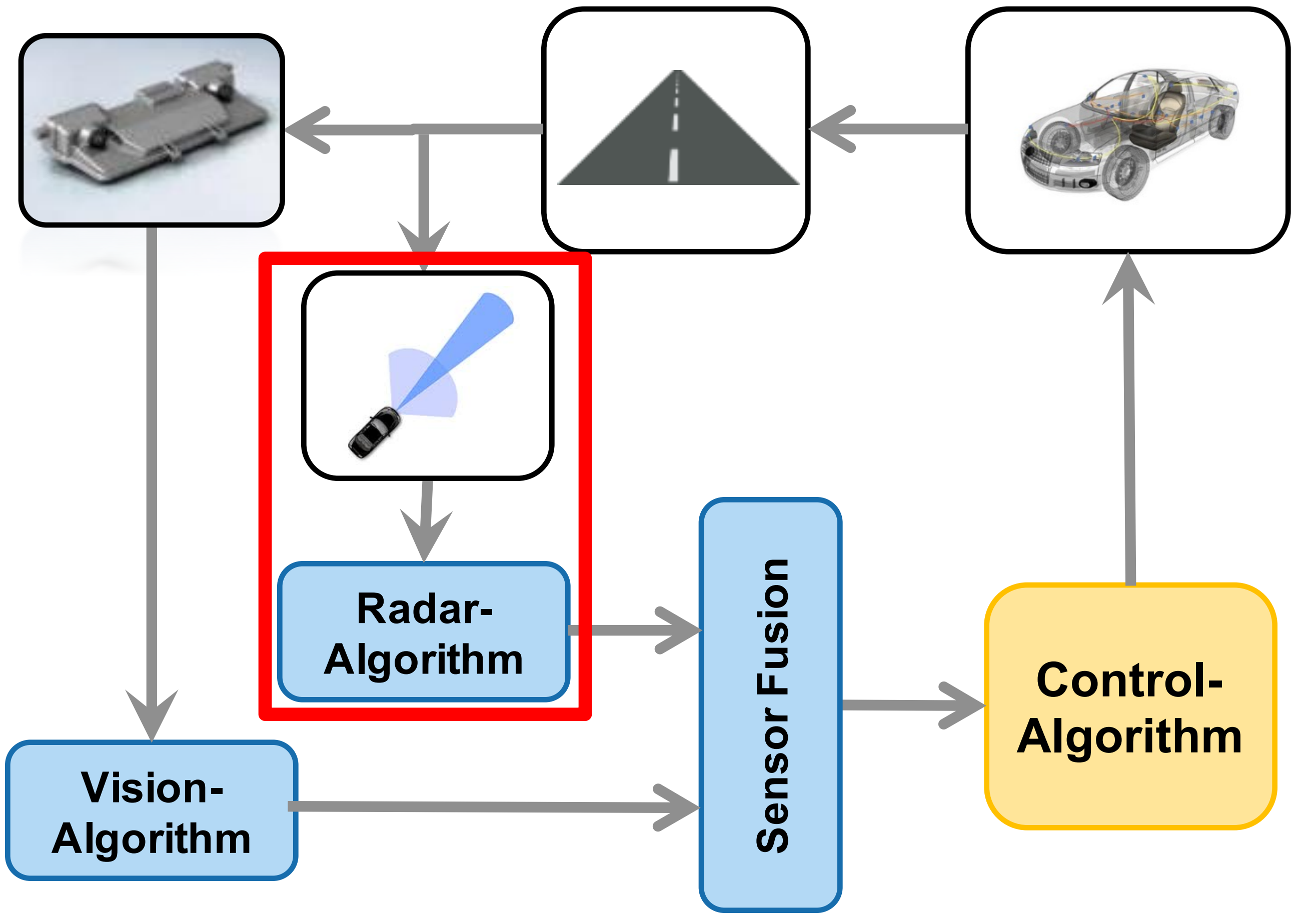


Radar-based ADAS Applications

Short/Medium Range	Long Range
Blind Spot Detection	Automatic Cruise Control
Cross Traffic Detection	Collision Avoidance
Rear Collision Warning	
Emergency Braking	

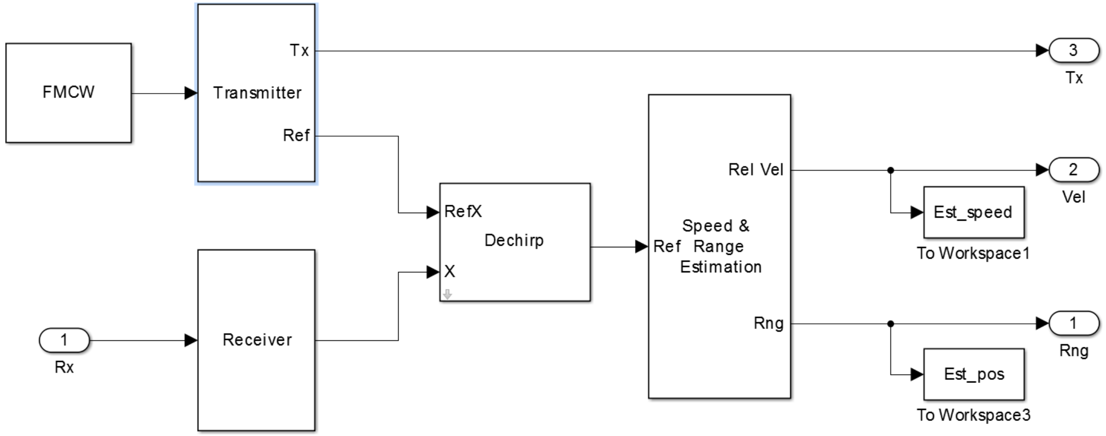
Modeling ADAS Features in Simulink

Integrate sensing (radar, vision, etc…) and control algorithms



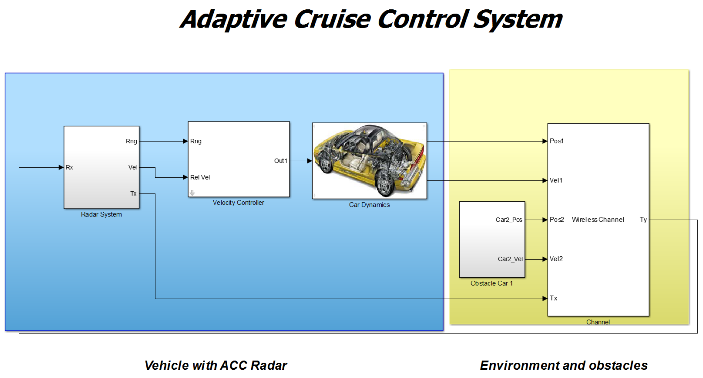
Designing Radar Systems with Simulink

- <u>Design</u> a radar component or system
  - Mix of models with different levels of fidelity
  - Multipath propagation
  - Multiple objects
  - Develop optimal detection algorithms



Radar Model

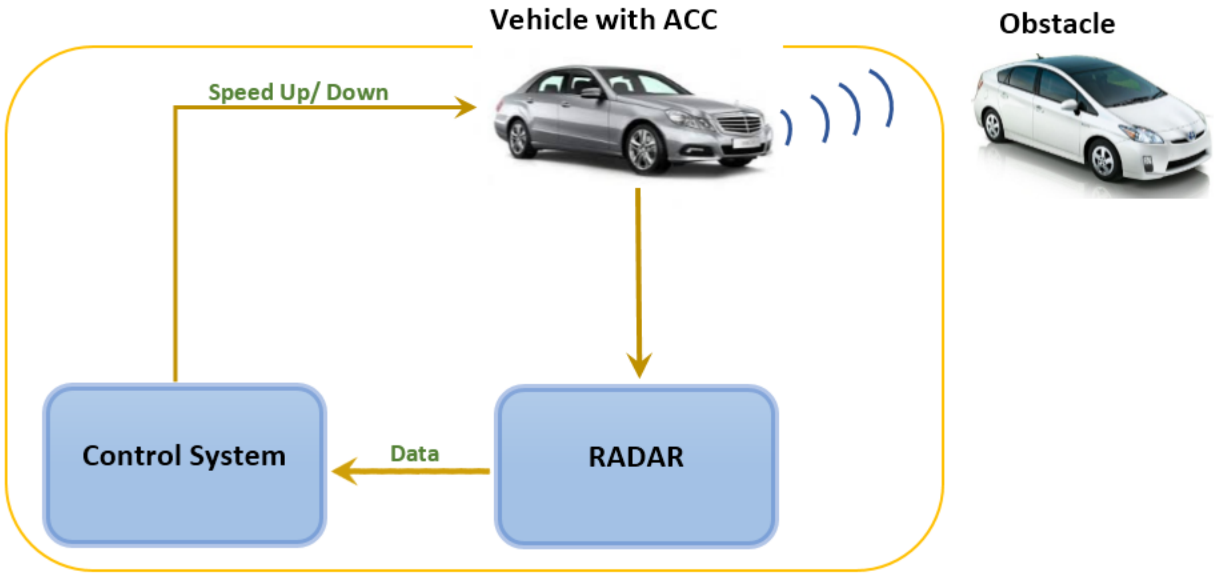
- <u>Integrate</u> a radar component or system
  - Validate radar performance and examine what-if scenarios
  - Simplify system level debug for anomalous data from road-test
  - Evaluate the use of off-the-shelf components to reduce system cost
  - Integrate the radar in a larger system and facilitate sensor integration



System Model

Long Range Radar Model - ACC

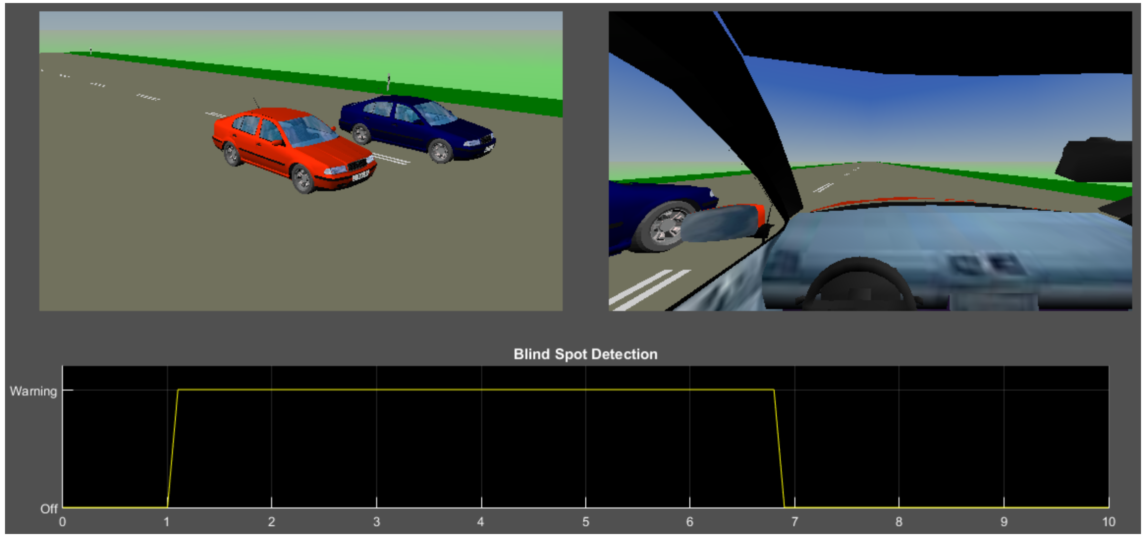
- FMCW radar model
  - Waveform generation
  - Waveform processing
  - Transmitter and receiver
- Environment model
  - Propagation channel
  - Obstacles
- Vehicle model
  - ACC speed control
  - Vehicle dynamics
- 3D visualization



Blind Spot Detection Model

- Built using Long Range Radar example as a starting point

- Change orientation of the radar
- Modify positioning and velocity of the vehicles
- Update detection algorithm



- Lower transmit power for shorter distance
- Higher range resolution for distance accuracy

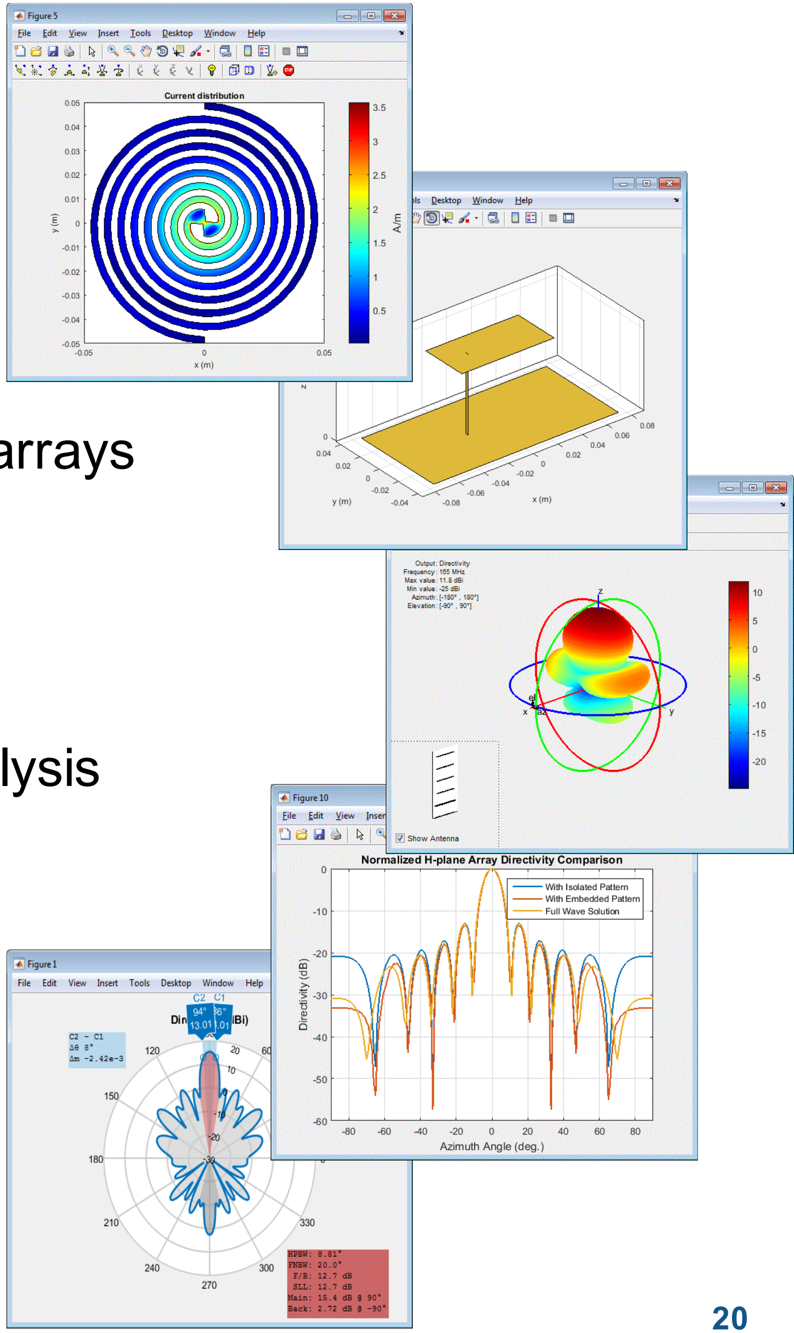
Increasing the Fidelity of the RF and Antenna Models

Antenna Toolbox

- Easy design
  - Library of parameterized antenna elements
  - Functionality for the design of linear and rectangular antenna arrays
  - No need for full CAD design

- Rapid simulation setup
  - Method of Moments field solver for port, field, and surface analysis
  - No need to be an EM expert

- Seamless integration
  - Model the antenna together with signal processing algorithms
  - Rapid iteration of different antenna scenarios for radar and communication systems design



Integrating an Antenna Array in a Radar System Model

- You can integrate your antenna into the radar model built in Phased Array System Toolbox

```
% Import antenna element in Phased Array
>> myantenna = dipole;
>> myURA = phased.URA;
>> myURA.Element = myantenna;
```

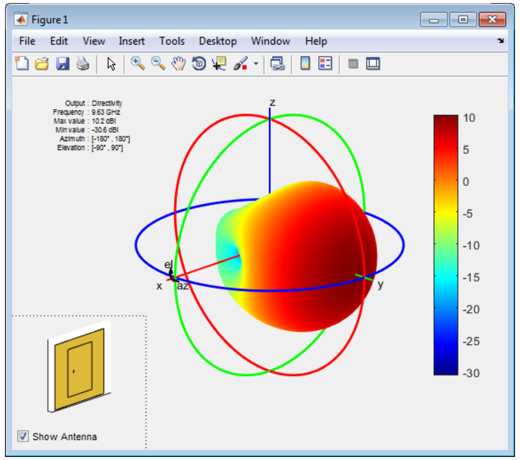
Antenna element

Phased Array System Toolbox array

Antenna Array, Phased Array, and Radiation Pattern

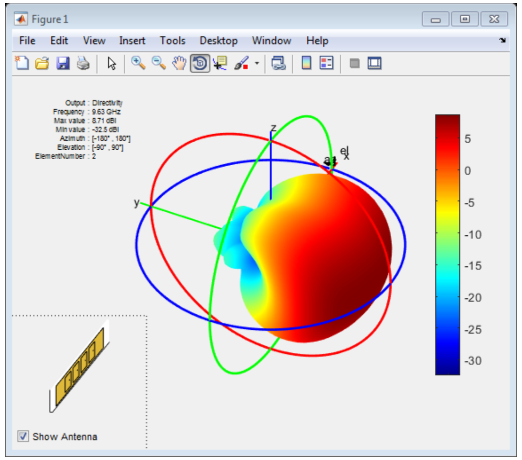
- Phased Array System Toolbox arrays use pattern superposition
  - ULA, URA, UCA and conformal arrays use the same pattern for all elements
  - Heterogeneous arrays have different patterns for different elements
- Antenna Toolbox arrays perform full wave EM analysis
  - Isolated element vs embedded element vs full array

Isolated element



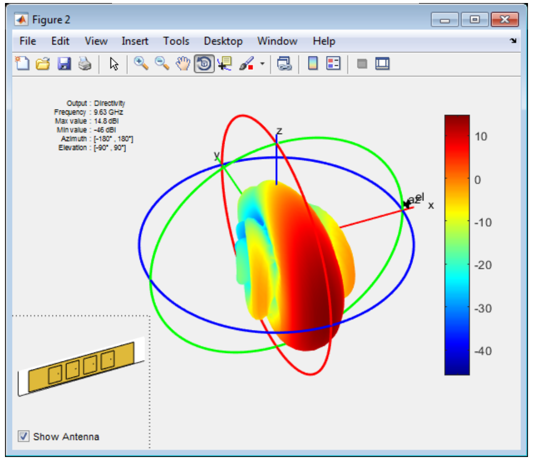

```
pattern(p, 10e9);
```

Embedded element




```
pattern(l, 10e9, ...
'ElementNumber',2);
```

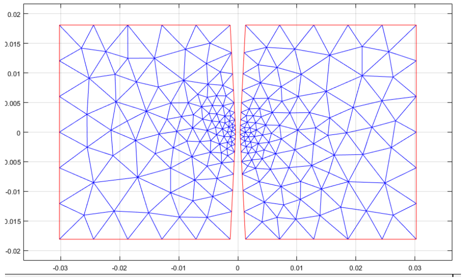
Full array

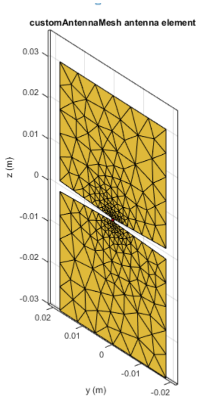


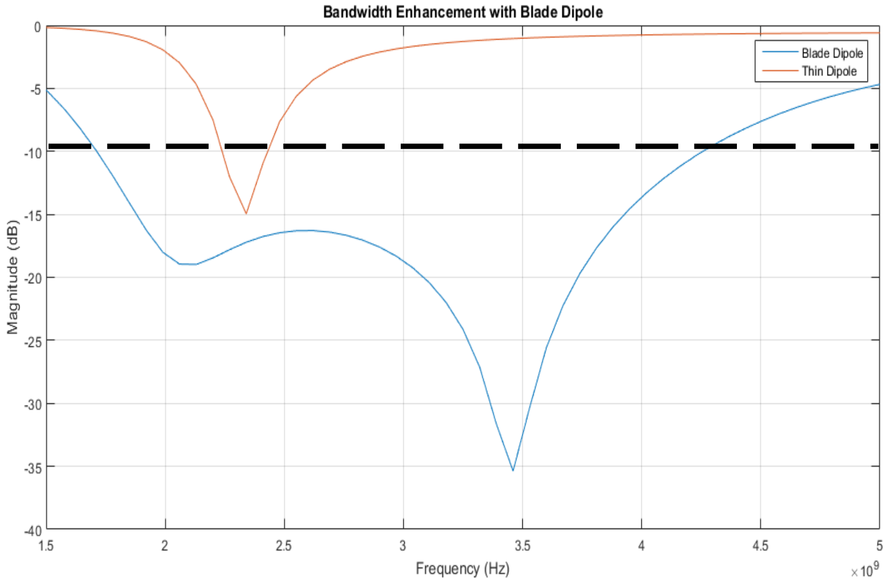

```
pattern(l, 10e9);
```

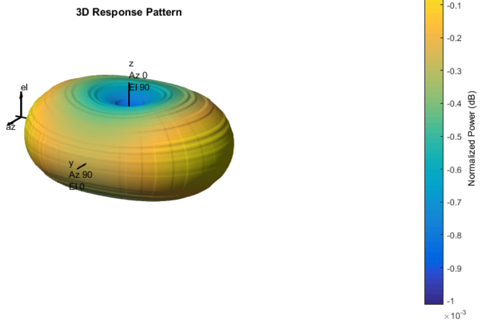
Wideband Antenna Integration Example

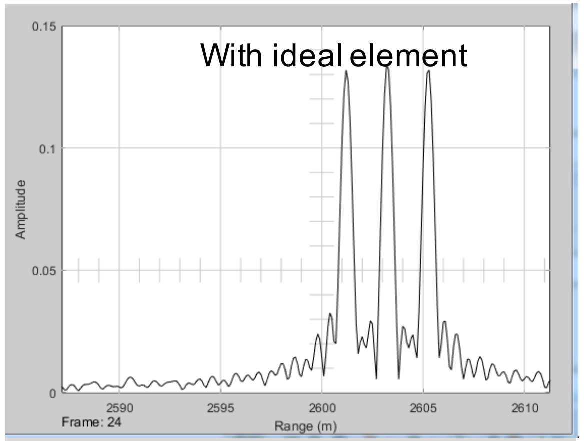
- Demo

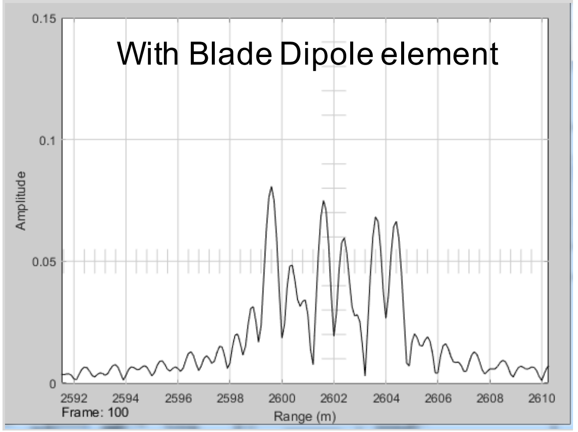




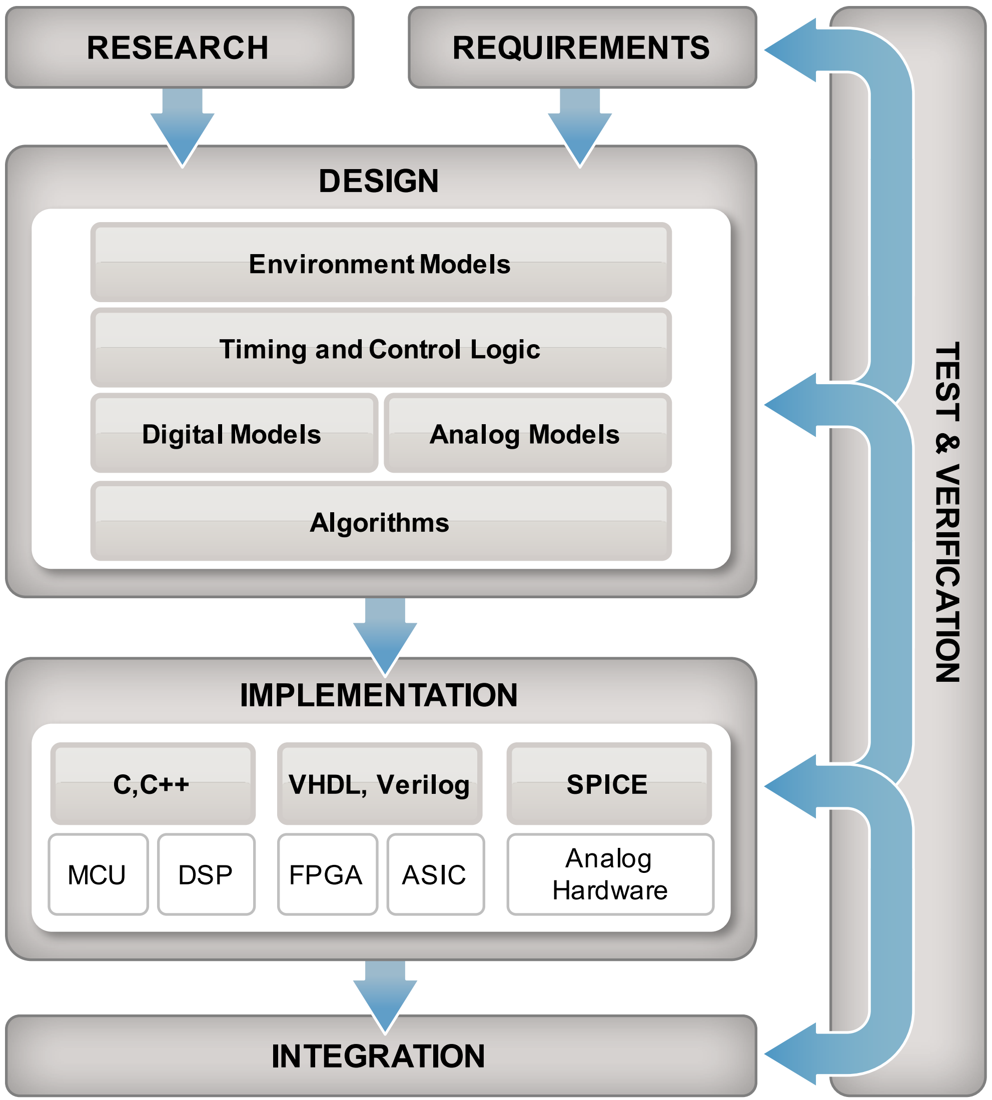








Model-Based Design for Radar Systems



- Complete system model
- Requirements traceability
- System behavior exploration
- Cross team collaboration

- Deploy on desktop
- Generate code and HDL
- Integrate into larger systems
- Explore design tradeoffs

- Configuration management
- Automated regression testing
- Report generation
- V&V and security analysis for code
- Support for certification & standards

Summary

- MATLAB, Simulink, Phased Array System Toolbox, Antenna Toolbox and SimRF provide flexible platform for radar system design and simulation
  - Large number of examples to get started with

- MathWorks products also enable design and implementation of radar systems across the workflow

- Thank you very much for you time today