MathWorks
AUTOMOTIVE CONFERENCE 2019

MATLAB을 활용한 자동차 레이더 개발

서기환
Agenda

- Different Usage for Radar Modeling

- Radar Design Workflow
  - DSP Design and Simulation
  - RF/Antenna Modeling

- High Level Simulation with Probabilistic Model
Agenda

- Radar Modeling with Fidelity Control
  - Radar Design Workflow
    - DSP Design and Simulation
    - RF/Antenna Modeling
  - High Level Simulation with Probabilistic Model
Two Personas using Automotive Radar Sensor Models

Radar Designers

- Radar Sensor
  - World / Truth
  - Detections
  - Tracks
  - IQ Data
  - …

Radar Users

- Radar Sensor
- Detections
- Tracks
- World / Truth

- Lidar Sensor
- Camera Sensor
- Ultrasound Sensor

Antenna | RF Front-End | Signal Processing
Radar Modeling and Simulation

RF/Antenna Modeling DSP Algorithm Modeling

Antenna/Array/RF

Radar Signal Processing
  - Signal Processing
  - Waveform Generation

Scheduling and Radar Control

Target Classification

Tracking and Sensor Fusion
Radar Modeling and Simulation

Probabilistic Model

Radar Sensor

Target Classification

Tracking and Sensor Fusion
# Automotive Radar Sensor Models

<table>
<thead>
<tr>
<th></th>
<th>DSP algorithm Model</th>
<th>RF/Antenna Model</th>
<th>Probabilistic Model</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Engineers</strong></td>
<td>Radar Designers</td>
<td>Radar Designers</td>
<td>Radar Users</td>
</tr>
<tr>
<td><strong>Model usage</strong></td>
<td>Radar Algorithms</td>
<td>Analog-mixed Signal Simulation</td>
<td>Sensor Fusion, Controller</td>
</tr>
<tr>
<td><strong>Outputs</strong></td>
<td>IQ Data or Detections</td>
<td>IQ Data</td>
<td>Detections or Tracks</td>
</tr>
<tr>
<td><strong>Benefit</strong></td>
<td>-</td>
<td>Highest Fidelity</td>
<td>Simulation Speed</td>
</tr>
<tr>
<td><strong>Disadvantage</strong></td>
<td>-</td>
<td>Simulation speed</td>
<td>Can’t Access IQ Data</td>
</tr>
</tbody>
</table>
Agenda

- Radar Modeling with Fidelity Control

- Radar Design Workflow
  - DSP Design and Simulation
  - RF/Antenna Modeling

- High Level Simulation with Probabilistic Model
Radar Simulation and Modeling Architecture

- Functions for calculations and analysis
- Apps for common workflows
- Parameterized components for system modeling
- Easy path to increased fidelity for antenna and RF design
- Code generation for deployment

\[ P_r = \frac{P_t G_t G_r \lambda^2 \sigma}{(4\pi)^3 R_t^2 R_r^2 L} \]
Radar Model to Simulate High Fidelity Raw Data

Antenna, Antenna arrays
- type of element, # elements, configuration
  - Antenna Toolbox
  - Phased Array System Toolbox

Channel
- interference, clutter, noise

RX
- LNA
- PA

Mixed-Signal
- Continuous & discrete time
  - Simulink
  - DSP System Toolbox
  - Control System Toolbox

Algorithms
- beamforming, beamsteering, MIMO
  - Phased Array System Toolbox
  - Communications System Toolbox
  - DSP System Toolbox

Signal Processing
- Tracking & Sensor Fusion
  - Sensor Fusion and Tracking Toolbox

DAC
- RF Blockset
- RF Toolbox

Waveforms & Resource Scheduling
- Phased Array System Toolbox
- Signal Processing Toolbox
- Wavelet Toolbox
Path to Higher Fidelity

- Extend model fidelity over project evolution
- Simple interface to replace off-the-shelf components with custom ones

<table>
<thead>
<tr>
<th>Antenna element</th>
<th>Target model</th>
<th>Propagation model</th>
<th>RF signal chain</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ideal elements</td>
<td>Point target</td>
<td>Free space</td>
<td>Baseband</td>
</tr>
<tr>
<td>EM solver with mutual coupling</td>
<td>Synthesized backscatter (angle &amp; frequency)</td>
<td>Line of sight atmospheric effects</td>
<td>RF components</td>
</tr>
<tr>
<td>Measured pattern import</td>
<td>Measured return (angle &amp; frequency)</td>
<td>Multipath, terrain and ducting effects</td>
<td>Analog-mixed simulation</td>
</tr>
</tbody>
</table>
DSP Algorithms for Radar Systems

Beamforming

Detections

Direction of Arrival
Antenna Array Design

Design subarray with desired fidelity

Replicate to build array

Assess resulting pattern

\[ p = \text{design(patchMicrostrip,66e9)} \]
Beam Steering
Synthesizing an Array from a Specified Pattern

- Introduce optimization workflow

https://blog.naver.com/matlablove/221205030640
Hybrid Beamforming

- 4 subarrays of 8 patch antennas operating at 66GHz → 4x8 = 32 antennas
- Digital beamforming applied to the 4 subarrays (azimuth steering)
- RF beamforming (phase shifters) applied to the 8 antennas (elevation steering)
RF Front End Modelling using Circuit Envelope

- Direct conversion to IF (5GHz) and superhet up-conversion to mmWave (79GHz)
- Non-linear impairments such as IP2, IP3, P1dB.
- Power dividers (e.g. S-parameters)
- Variable phase-shifters
Radar Waveform Analyzer
Sensor Array Analyzer

Azimuth Cut (elevation angle = 0.0°)

Directivity (dBi), Broadside at 0.00°
Radar Equation Calculator, RF Budget Analyzer
Model FMCW RADARs at mmWave Frequencies
Model FMCW Radar – RF Front-End

**FMCW Radar Range, Speed, and Angle Estimation**

- Transmitter
  - Amplify and transmit the signal. Transmitter can either maintain coherence between pulses or insert phase noise.

- Parameters
  - Peak power (W): paramFMCWRSA.ppow
  - Gain (dB): paramFMCWRSA.TxGain
  - Loss factor (dB): 0

- Simulator:
  - Enable transmitter status output
  - Preserve coherence among pulses

Simulate using: Interpreted execution

---

**MathWorks**

AUTOMOTIVE CONFERENCE 2019
Model FMCW Radar – Transmit Array Antenna

FMCW Radar Range, Speed, and Ang
Model FMCW Radar – Signal Processing
Visualizing Radar and Target Trajectory
Automated Driving Simulation with IQ-level Radar Signal

Without ground reflection
Simulating Micro-Doppler Signatures

Micro-Doppler for Pedestrian

Bicycle
Pedestrian Micro-Doppler with and without Parked Vehicle

Micro-Doppler for pedestrian (only)  Micro-Doppler for pedestrian and parked vehicle
Pedestrian and Bicyclist Classification Using Deep Learning
Pedestrian and Bicyclist Classification Using Deep Learning

<table>
<thead>
<tr>
<th>True Class</th>
<th>ped</th>
<th>bic</th>
<th>ped+bic</th>
<th>ped+ped</th>
<th>bic+bic</th>
</tr>
</thead>
<tbody>
<tr>
<td>ped</td>
<td>908</td>
<td>1</td>
<td>1</td>
<td>90</td>
<td></td>
</tr>
<tr>
<td>bic</td>
<td>15</td>
<td>911</td>
<td>28</td>
<td>3</td>
<td>43</td>
</tr>
<tr>
<td>ped+bic</td>
<td>12</td>
<td>78</td>
<td>835</td>
<td>39</td>
<td>36</td>
</tr>
<tr>
<td>ped+ped</td>
<td>165</td>
<td>1</td>
<td>4</td>
<td>830</td>
<td></td>
</tr>
<tr>
<td>bic+bic</td>
<td>4</td>
<td>82</td>
<td>33</td>
<td>1</td>
<td>880</td>
</tr>
</tbody>
</table>
Increasing Angular Resolution with MIMO Radars (Virtual Array)

Two options: Increase number of receive elements or perform signal processing
Agenda

- Radar Modeling with Fidelity Control

- Radar Design Workflow
  - DSP Design and Simulation
  - RF/Antenna Modeling

- High Level Simulation with Probabilistic Model
# Virtual Driving Scenarios with Radar Sensor

<table>
<thead>
<tr>
<th>Scenes</th>
<th>Cuboid</th>
<th>3D Simulation</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1" alt="Cuboid Image" /></td>
<td><img src="image2" alt="3D Simulation Image" /></td>
<td><img src="image3" alt="3D Simulation Image" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Testing</th>
<th>Controls, sensor fusion, planning</th>
<th>Controls, sensor fusion, planning, perception</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Authoring</th>
<th>Driving Scenario Designer App Programmatic API (drivingScenario)</th>
<th>Unreal Engine Editor</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Sensing</th>
<th>Probabilistic radar (detection list)</th>
<th>Probabilistic vision (detection list)</th>
<th>Probabilistic lane (detection list)</th>
<th>Probabilistic radar (detection list)</th>
<th>Monocular camera (image, labels, depth)</th>
<th>Fisheye camera (image)</th>
<th>Lidar (point cloud)</th>
</tr>
</thead>
</table>
Synthesize Radar Sensor Data

Simulate Radar Sensors in 3D Environment
- Extract the center locations
- Use center location for road creation using driving scenario
- Define multiple moving vehicles
- Export trajectories from app
- Configure multiple probabilistic radar models
- Calculate confirmed track

Automated Driving Toolbox™
Design trackers

- Multi-object tracker
- Global Nearest Neighbor (GNN) tracker
- Joint Probabilistic Data Association (JPDA) tracker
- Track-Oriented Multi-Hypothesis Tracker (TOMHT)
- Probability Hypothesis Density (PHD) tracker

- Linear, extended, and unscented Kalman filters
- Particle, Gaussian-sum, and Interacting Multiple Model (IMM) filters

Automated Driving Toolbox™
Sensor Fusion and Tracking Toolbox™
Design Multi-Object Trackers

Extended Object Tracking
- Design multi-object tracker
- Design extended object trackers
- Evaluate tracking metrics
- Evaluate error metrics
- Evaluate desktop execution time

Sensor Fusion and Tracking Toolbox™
Automated Driving Toolbox™
Design track level fusion systems
Track-Level Fusion

Track-to-Track Fusion for Automotive Safety Applications

Sensor Fusion and Tracking Toolbox™
Automated Driving Toolbox™

Parked vehicles observed by vehicle 1
Pedestrian observed by vehicle 1

Occluded vehicle fused from vehicle 1
Occluded pedestrian fused from vehicle 1

Rumor control: the fused track is dropped by vehicle 1 because vehicle 2 is coasting and there is no update by vehicle 1 sensors
Radar System Modeling for Perception

Antenna/RF

Waveforms

Tracking/ Sensor Fusion

Scenario Generation

Spatial Signal Processing

Environment

Deep Learning

Code generation

HDL
Key Takeaways

- Choose the Right Modeling Method
  - You can control the fidelity

- Start Your First Radar Design with Various Apps

- High Level Simulation with Probabilistic Model
  - Tracking
  - Control
Thank You!