MATLAB EXPO 2019

Automated Driving with MATLAB and Simulink

Fulvio Martinelli
Some common questions from automated driving engineers

How can I synthesize scenarios to test my designs?

How can I discover and design in multiple domains?

How can I integrate with other environments?

Perception

Planning

Control

Simulation Integration

ROS

CAN

C/C++

Python

Cross Release

Third Party
Some common questions from automated driving engineers

How can I synthesize scenarios to test my designs?

How can I discover and design in multiple domains?

How can I integrate with other environments?

Control
Perception
Planning
Simulation Integration

ROS
C/C++
Python
Cross Release
Third Party
CAN
Graphically author driving scenarios

**Driving Scenario Designer**
- Create roads and lane markings
- Add actors and trajectories
- Specify actor size and radar cross-section (RCS)
- Add sensor models
- Explore pre-built scenarios
- Import OpenDRIVE roads

**Automated Driving Toolbox™**

R2018a
Simulate driving scenarios into closed loop simulations

**Automatic Emergency Braking (AEB) with Sensor Fusion**
- Specify driving scenario
- Design AEB logic
- Integrate sensor fusion
- Simulate system
- Generate C/C++ code
- Test with software in the loop (SIL) simulation

**Automated Driving Toolbox™**

**Stateflow®**

**Embedded Coder®**
Automate testing against driving scenarios

**Testing a Lane Following Controller with Simulink Test**
- Define scenarios as test cases
- Customize tests using callbacks
- Link test cases to requirements
- Manage test cases
- Run tests
- Automatically generate reports

**Simulink Test™**
**Automated Driving Toolbox™**
**Model Predictive Control Toolbox™**

R2018b
Synthesize driving scenarios from recorded data

Scenario Generation from Recorded Vehicle Data
- Visualize video
- Import OpenDRIVE roads
- Import GPS
- Import object lists

Automated Driving Toolbox™

R2019a
Read lane attributes from HERE HD Live Map data

Use HERE HD Live Map Data to Verify Lane Configurations
- Load camera and GPS data
- Retrieve speed limit
- Retrieve lane configurations
- Visualize composite data

Automated Driving Toolbox™

R2019a
How can I design with virtual scenarios?

<table>
<thead>
<tr>
<th>Scenes</th>
<th><strong>Driving Scenarios (cuboid)</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><img src="image" alt="Diagram" /></td>
</tr>
</tbody>
</table>

| Testing | Controls
Controls + sensor fusion |
|---------|--------------------------|
| Authoring | Driving Scenario Designer App
drivingScenario programmatic API |
| Sensing | Probabilistic radar detections
Probabilistic vision detections
Probabilistic lane detections |
### How can I design with virtual scenarios?

#### Scenes

<table>
<thead>
<tr>
<th>Testing</th>
<th>Controls</th>
<th>Controls + sensor fusion</th>
</tr>
</thead>
<tbody>
<tr>
<td>Authoring</td>
<td>Driving Scenario Designer App</td>
<td>Unreal Editor</td>
</tr>
<tr>
<td>Sensing</td>
<td>Probabilistic radar detections</td>
<td>Ideal camera (viewer)</td>
</tr>
<tr>
<td></td>
<td>Probabilistic vision detections</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Probabilistic lane detections</td>
<td></td>
</tr>
</tbody>
</table>

#### Driving Scenarios (cuboid)

![Driving Scenarios (cuboid)](image)

#### Unreal Engine

![Unreal Engine](image)
Simulate lane controls with vision based perception

**Lane-Following Control with Monocular Camera Perception**

- Integrate Simulink controller
  - Lane follower
  - Spacing control
- Integrate MATLAB perception
  - Lane boundary detector
  - Vehicle detector
- Synthesize ideal camera image from Unreal Engine

*Model Predictive Control Toolbox™*

*Automated Driving Toolbox™*

*Vehicle Dynamics Blockset™*
Some common questions from automated driving engineers

How can I synthesize scenarios to test my designs?

How can I discover and design in multiple domains?

How can I integrate with other environments?
Create region of interest labels and groups

Get Started with the Ground Truth Labeler

- Label rectangles
- Create label groups

Automated Driving Toolbox™

Updated R2019a
Create sublabels and add attributes

Get Started with the Ground Truth Labeler
- Label rectangles
- Create label groups
- Create sublabels
- Add label attributes

Automated Driving Toolbox™

Updated R2019a
Create polyline labels and add attributes

**Get Started with the Ground Truth Labeler**
- Label rectangles
- Create label groups
- Create sublabels
- Add label attributes
- Label lane markings

*Automated Driving Toolbox™*
Updated R2019a
Create pixel labels

Get Started with the Ground Truth Labeler

- Label rectangles
- Create label groups
- Create sublabels
- Add label attributes
- Label lane markings
- Label pixels

Automated Driving Toolbox™

Updated R2019a
Create scene labels and groups

Get Started with the Ground Truth Labeler
- Label rectangles
- Create label groups
- Create sublabels
- Add label attributes
- Label lane markings
- Label pixels
- Label scenes

Automated Driving Toolbox™
Updated R2019a
Import custom automation algorithms

**Automate Attributes of Labeled Objects**
- Import automation algorithm into Ground Truth Labeling app
- Detect vehicles from monocular camera
- Estimate distance to detected vehicles
- Run automation algorithm and interactively validate labels

*Automated Driving Toolbox™ R2018b*
Design detector for lidar point cloud data

Track Vehicles Using Lidar: From Point Cloud to Track List

- Design 3-D bounding box detector
- Design tracker (target state and measurement models)
- Generate C/C++ code for detector and tracker

Sensor Fusion and Tracking Toolbox™
Computer Vision Toolbox™

R2019a
Design tracker for lidar point cloud data

**Track Vehicles Using Lidar: From Point Cloud to Track List**
- Design 3-D bounding box detector
- Design tracker (target state and measurement models)
- Generate C/C++ code for detector and tracker

*Sensor Fusion and Tracking Toolbox™*
*Computer Vision Toolbox™*
Generate C/C++ code for lidar detector and tracker

**Track Vehicles Using Lidar: From Point Cloud to Track List**

- Design 3-D bounding box detector
- Design tracker (target state and measurement models)
- Generate C/C++ code for detector and tracker

**Sensor Fusion and Tracking Toolbox™**

**Computer Vision Toolbox™**

*R2019a*
Design trackers

Multi-Object Tracker

Association & Track Management

Tracking Filter

Detections

Tracks

From various sensors at various update rates

- 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, IMM filters

Automated Driving Toolbox™
Sensor Fusion and Tracking Toolbox™
Design extended 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™

Updated R2019a
Some common questions from automated driving engineers

How can I synthesize scenarios to test my designs?

How can I discover and design in multiple domains?

How can I integrate with other environments?
Design path planner

**Automated Parking Valet**
- Create cost map of environment
- Inflate cost map for collision checking
- Specify goal poses
- Plan path using rapidly exploring random tree (RRT*)

**Automated Driving Toolbox™**

R2018a
Design path planner and controller

**Automated Parking Valet with Simulink**
- Integrate path planner
- Design lateral controller (based on vehicle kinematics)
- Design longitudinal controller (PID)
- Simulate closed loop with vehicle dynamics

**Automated Driving Toolbox™**
Generate C/C++ code for path planner and controller

**Code Generation for Path Planning and Vehicle Control**
- Simulate system
- Configure for code generation
- Generate C/C++ code
- Test using Software-In-the-Loop
- Measure execution time of generated code

**Automated Driving Toolbox™**
Embedded Coder

![Code Snippet]

---

(R2019a)
Some common questions from automated driving engineers

- How can I synthesize scenarios to test my designs?
- How can I discover and design in multiple domains?
- How can I integrate with other environments?

Control
Planning
Perception

Simulation Integration

ROS
C/C++
Python
CAN
Cross Release
Third Party
Design lateral and longitudinal Model Predictive Controllers

Longitudinal Control

Adaptive Cruise Control with Sensor Fusion
Automated Driving Toolbox™
Model Predictive Control Toolbox™
Embedded Coder®

Lateral Control

Lane Keeping Assist with Lane Detection
Automated Driving Toolbox™
Model Predictive Control Toolbox™
Embedded Coder®

Longitudinal + Lateral

Lane Following Control with Sensor Fusion and Lane Detection
Automated Driving Toolbox™
Model Predictive Control Toolbox™
Embedded Coder®
Train reinforcement learning networks for ADAS controllers

Train Deep Deterministic Policy Gradient (DDPG) Agent for Adaptive Cruise Control
- Create environment interface
- Create agent
- Train agent
- Simulate trained agent

Reinforcement Learning Toolbox™ R2019a
Some common questions from automated driving engineers

How can I synthesize scenarios to test my designs?

How can I discover and design in new domains?

How can I integrate with other environments?
Integrate with ROS

Replay logged ROS data

Connect to live ROS data

Generate standalone ROS node

Work with rosbag Logfiles
Robotic System Toolbox™

Exchange Data with ROS Publishers and Subscribers
Robotic System Toolbox™

Generate a Standalone ROS Node from Simulink
Robotic System Toolbox™ Simulink Coder™
Call C++, Python, and OpenCV from MATLAB

Import C++ Library Functionality into MATLAB
MATLAB®
R2019a

Call Python from MATLAB
MATLAB®
R2014a

Install and Use Computer Vision Toolbox OpenCV Interface
Computer Vision System Toolbox™
OpenCV Interface Support Package

Updated R2018b
Call C code from Simulink

- Call C code
  - src mean_filter dst
  - C Caller

- Create buses from C structs
  - typedef struct {
    double coeff;
    double init;
    fault_T fault;
  } params_T;

- Test and verify C code
  - Custom C Code Verification with Simulink Test
    - Simulink Test™
    - Simulink Coverage™

Bring Custom Image Filter Algorithms as Reusable Blocks in Simulink

Simulink®

R2017b

Import Structure and Enumerated Types

Simulink®

R2017a

R2019a
Connect to third party tools

152 Interfaces to 3\textsuperscript{rd} Party Modeling and Simulation Tools
(as of March 2019)
Cross-release simulation through code generation

Integrate Generated Code by Using Cross-Release Workflow

- Generate code from previous release (R2010a or later)
- Import generated code as a block in current release
- Tune parameters
- Access internal signals

Embedded Coder

**R2016a**
Some common questions from automated driving engineers

Synthesize scenarios to test my designs

Discover and design in multiple domains

Integrate with other environments

Perception
Planning
Control

Simulation Integration
ROS
CAN
C/C++
Python
Cross Release
Third Party

MathWorks
MathWorks can help you customize MATLAB and Simulink for your automated driving application

Voyage develops MPC controller and integrates with ROS
- 2018 MathWorks Automotive Conference

Autoliv labels ground truth lidar data
- Joint presentation with Autoliv
- SAE Paper 2018-01-0043
- 2018 MathWorks Automotive Conference

Ford tests algorithms with synthetic Lidar data from Unreal Engine
- Joint paper with Ford
- SAE Paper 2017-01-0107
MathWorks supports you in the development of your application

- Advance your skills with MathWorks Training
- Achieve results faster with MathWorks Consulting