What’s New in Automated Driving with MATLAB and Simulink

Mark Corless
Industry Marketing
Automated Driving Segment Manager

Marco Roggero
Application Engineering
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?
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  CAN  C/C++  Python

Cross Release Third Party  CAN

MathWorks
Graphically author driving scenarios

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

**Automated Driving Toolbox™**

R2018a
Integrate driving scenarios into Simulink simulations

Test Open-Loop ADAS Algorithm Using Driving Scenario

- Edit driving scenario
- Integrate into Simulink
- Add sensor models
- Visualize results
- Pace simulation

Automated Driving Toolbox™ R2019a
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
  ▪ Specify driving scenario

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

Requirements link
Simulink Model
Define scenario ID and data initialization
Plot the results
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™**

![MATLAB interface with driving scenario simulation](image)
How can I design with virtual scenarios?

<table>
<thead>
<tr>
<th>Scenes</th>
<th>Driving Scenarios (cuboid)</th>
</tr>
</thead>
</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?

<table>
<thead>
<tr>
<th>Scenes</th>
<th>Driving Scenarios (cuboid)</th>
<th>Unreal Engine</th>
</tr>
</thead>
<tbody>
<tr>
<td>Testing</td>
<td>Controls Controls + sensor fusion</td>
<td>Controls Controls + vision</td>
</tr>
<tr>
<td>Authoring</td>
<td>Driving Scenario Designer App drivingScenario programmatic API</td>
<td>Unreal Editor</td>
</tr>
<tr>
<td>Sensing</td>
<td>Probabilistic radar detections Probabilistic vision detections Probabilistic lane detections</td>
<td>Ideal camera (viewer)</td>
</tr>
</tbody>
</table>
Simulate controls and perception systems

Lane Following Control with Sensor Fusion
Model Predictive Control Toolbox™
Automated Driving Toolbox™
Embedded Coder®

Visual Perception Using Monocular Camera
Automated Driving Toolbox™

Lane-Following Control with Monocular Camera Perception
Model Predictive Control Toolbox™
Automated Driving Toolbox™
Vehicle Dynamics Blockset™
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?
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™**

**Updated R2019a**
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*
Evaluate tracking performance

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

![Bar chart showing the number of tracks for different types of trackers.]

- Multi-object tracker
- Probability Hypothesis Density tracker
- Extended object (size and orientation) tracker
Evaluate error metrics

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
Compare relative execution times of object trackers

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

**Sensor Fusion and Tracking Toolbox™**
**Automated Driving Toolbox™**

Updated R2019a

---

![Graph showing execution times of different trackers](image)

- Blue: Multi-object tracker
- Orange: Probability Hypothesis Density tracker
- Yellow: Extended object (size and orientation) tracker
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
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™
Create region of interest labels and groups

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

**Automated Driving Toolbox™**

Updated R2019a
Create sublabels and add attributes

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

Automated Driving Toolbox™

Updated R2019a
Create pixel labels

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

Automated Driving Toolbox™
Updated R2019a
Create scene labels and groups

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

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
Add custom visualizations for multi-sensor data

Connect Lidar Display to Ground Truth Labeler
- Sync external tool to each frame change
- Control external tool through playback controls

*Automated Driving Toolbox™ R2017a*
Interoperate with neural network frameworks

Open Neural Network Exchange
Design camera, lidar, and radar perception algorithms

- **Detect vehicle with camera**
- **Detect ground with lidar**
- **Detect pedestrian with radar**

**Object Detection Using YOLO v2 Deep Learning**
*Computer Vision Toolbox™
Deep Learning Toolbox™*

**Segment Ground Points from Organized Lidar Data**
*Computer Vision Toolbox™*

**Introduction to Micro-Doppler Effects**
*Phased Array System Toolbox™*
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

Control
Planning
Perception

Third Party
Cross Release
Python
C/C++
CAN
ROS
Read road and speed 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
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
Visualize HERE HD Live Map recorded 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
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™**
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™**

R2018b
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

```c
// model step function
void step0();

// model step function
void step1();

// model terminate function
void terminate();

// Constructor
AutomatedParkingValetModelClass();

// Destructor
~AutomatedParkingValetModelClass();

// Root import: 'Root>/Costmap' set method
void setCostmap(costmapBus localArgInput);

// Root import: 'Root>/GoalPose' set method
void setGoalPose(real_T localArgInput[3]);
```
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

CAN
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?

Simulation Integration

- Perception
- Planning
- Control

MathWorks

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

**Call C++**
- Import C++ Library Functionality into MATLAB
- MATLAB®
- R2019a

**Call Python**
- Call Python from MATLAB
- MATLAB®
- R2014a

**Call OpenCV & OpenCV GPU**
- Install and Use Computer Vision Toolbox OpenCV Interface
- Computer Vision System Toolbox™
- OpenCV Interface Support Package
- Updated R2018b

```cpp
cv::Rect
cv::KeyPoint
cv::Size
cv::Mat
cv::Ptr...
```

```python
py.textwrap.TextWrapper(...
    pyargs(...
        'initial_indent', '% ', ...
        'subsequent_indent', '% ', ...
        'width', int32(30)))
```
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**

**Import Structure and Enumerated Types**

**Simulink®**

**Simulink® R2017b**

**Simulink R2017a**

**Simulink R2019a**
Connect to third party tools

152 Interfaces to 3rd 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

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

**Control**

**Planning**

**Perception**
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
Develop Automated Driving Systems with MATLAB and Simulink

Simulation Integration

Perception

Planning

Control

Discuss your application with a MathWorks field engineer to help you structure your evaluation

- Understand your goals
- Recommend tasks
- Answer questions