

MathWorks
**AUTOMOTIVE
CONFERENCE
2018**

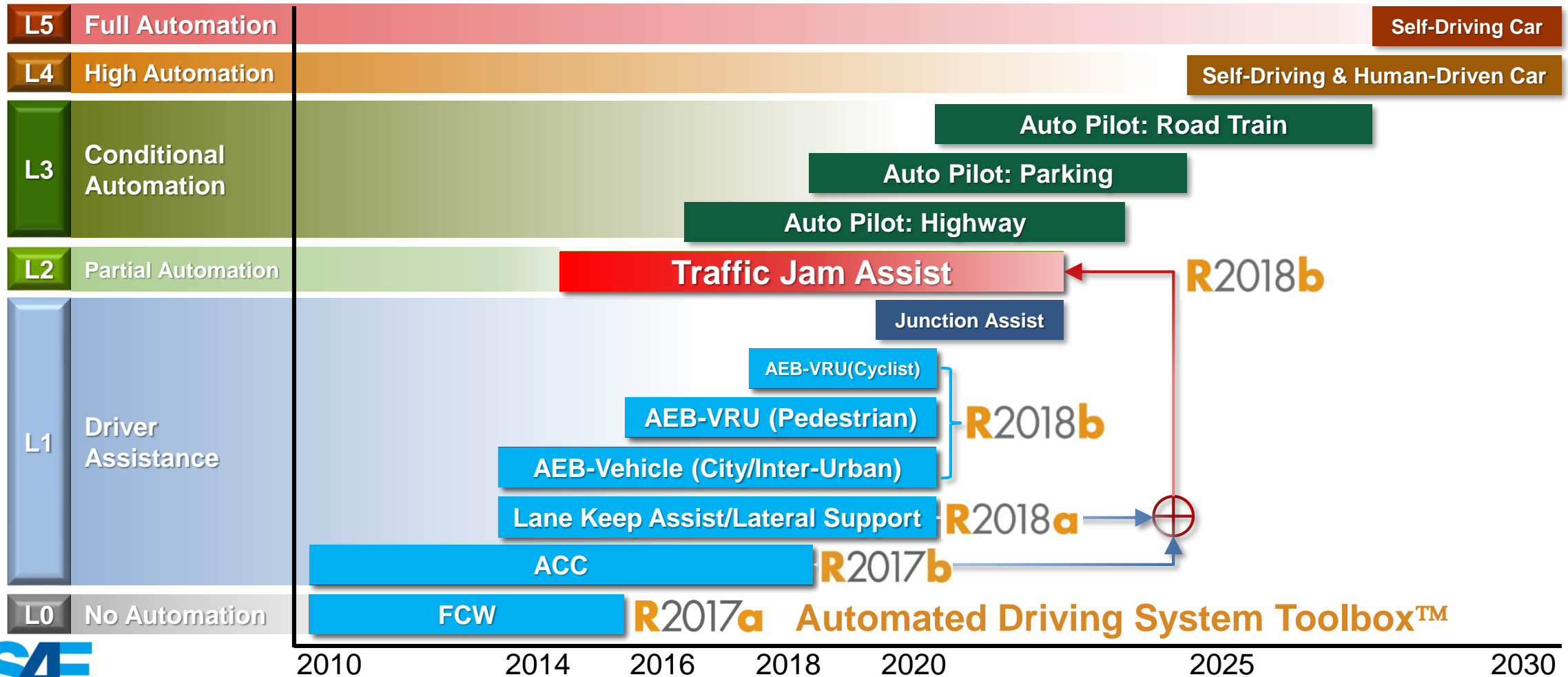
Design and Test
Traffic Jam Assist

*A Case Study
Using Automated Driving System Toolbox™*

Seo-Wook Park
Principal Application Engineer, MathWorks

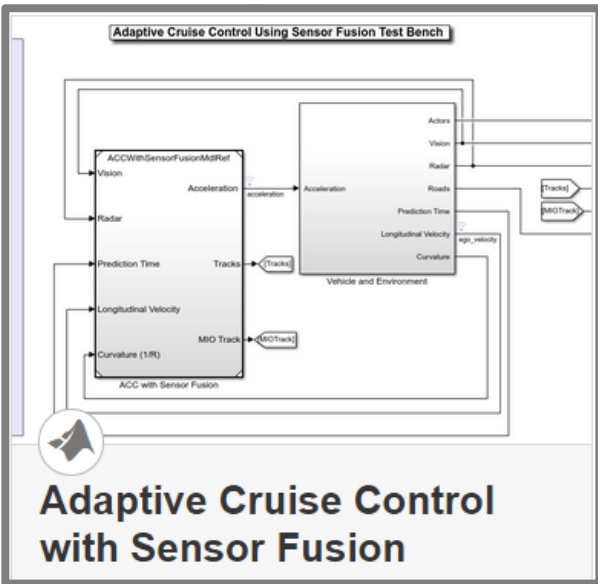


Evolution of ADAS and Autonomous Driving Car Technologies



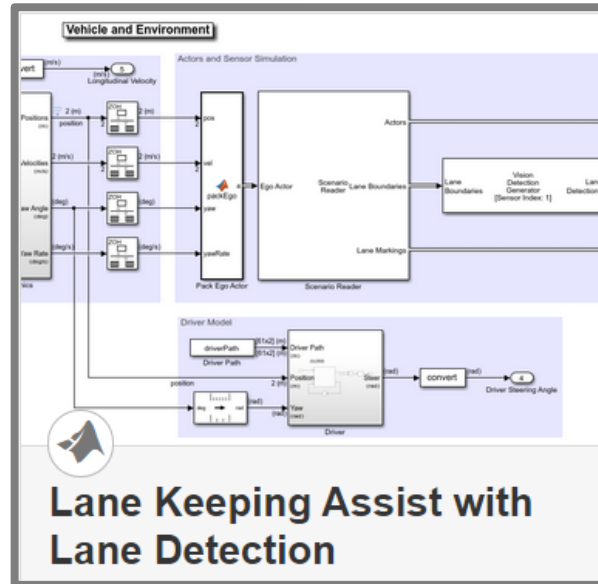
ACC and Lane Following Control for Traffic Jam Assist

Automated Driving System Toolbox™
R2017b + **R2018a**



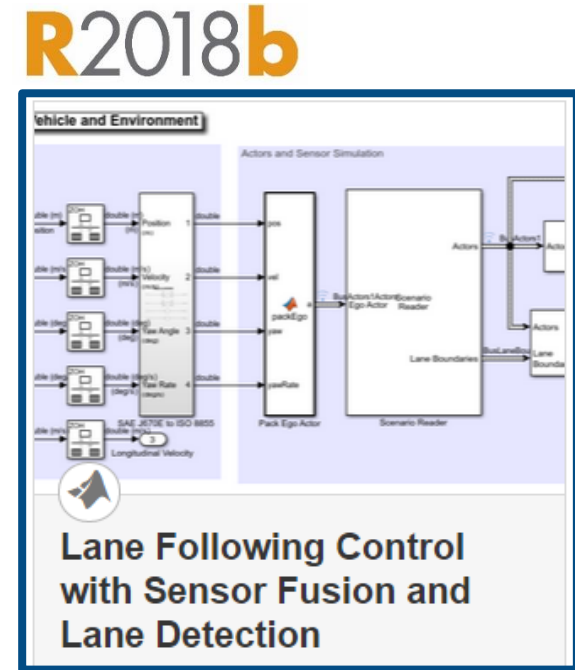
ACC
 (Longitudinal Control)

+



Lane Following
 (Lateral Control)

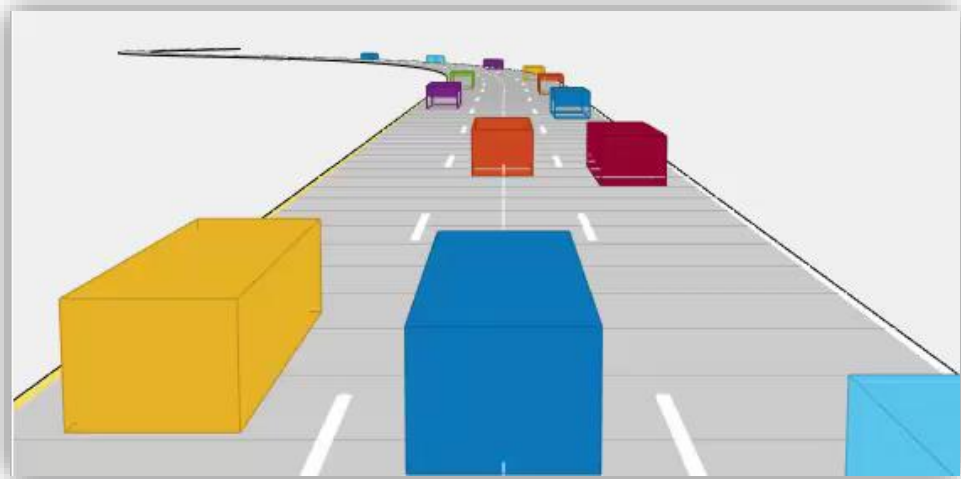
=



Traffic Jam Assist
 (Longitudinal
 + Lateral Control)

Traffic Jam Assist

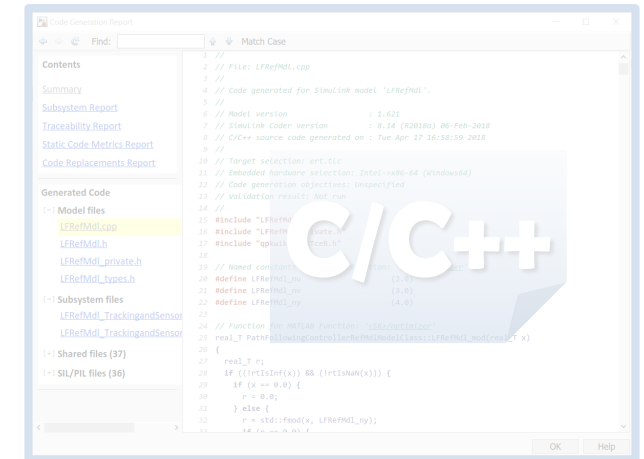
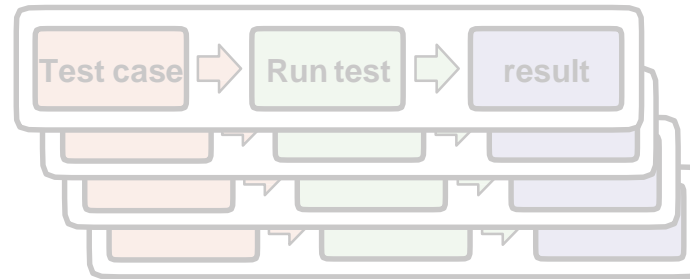
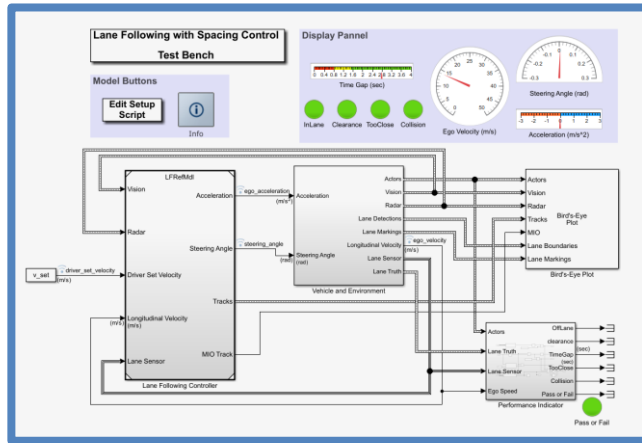
- It helps drivers to follow the preceding vehicle automatically with a predefined time interval in a dense traffic condition
 - ... while controlling steering for keeping current lane.
- } **Longitudinal control** with ACC with stop & go
- } **Lateral control** with lane following



- Partial/conditional automation at level 2/3
 - Speed limit < 60~65 km/h
 - Dense traffic condition in highway

Automated Driving System Toolbox™

Design and Test Traffic Jam Assist, A Case study



Design ACC and Lane Following Controller

- Create driving scenario
- Synthesize sensor detection
- Include Vehicle Dynamics
- Design sensor fusion algorithm
- Design controller using MPC

Automate Regression Test

- Define performance evaluation metrics
- Develop test cases
- Build test suites
- Verification and validation

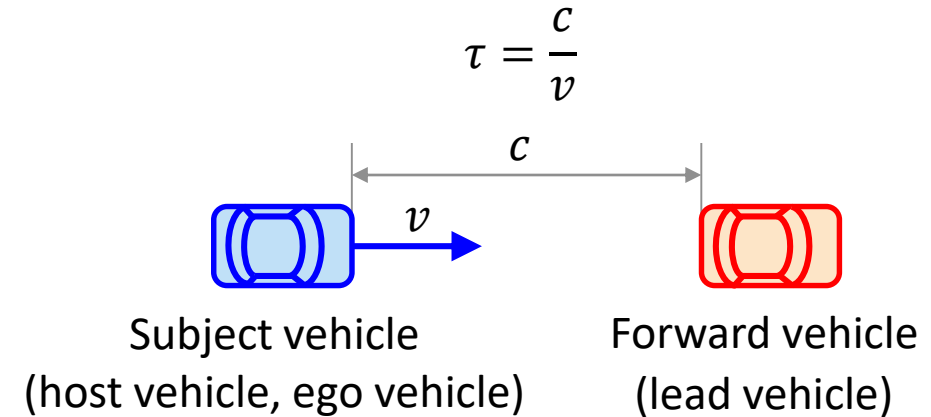
Generate and Verify Code

- SIL test
- Code generation
- Coverage test

ACC Performance Requirements

- **Ego velocity control :** $v \leq v_{set}$
where, v : ego velocity, v_{set} : set velocity

- **Time gap control:** $\tau \geq \tau_{min}$
where, $\tau = \frac{c}{v}$: time gap = 1.5 .. 2.2 sec
 τ_{min} : min time gap = 0.8 sec



- ACC operation limits
 - Minimum operational speed, $v_{min} = 5\text{m/s}$
 - Average automatic deceleration of ACC $\leq 3.5 \text{ m/s}^2$ (average over 2s)
 - Average automatic acceleration of ACC $\leq 2.0 \text{ m/s}^2$

Lane Following Performance Requirements

- Vehicle should follow the lane center with allowable lateral deviation.

$$|(d_{left} + d_{right})/2| \leq e_{max}$$

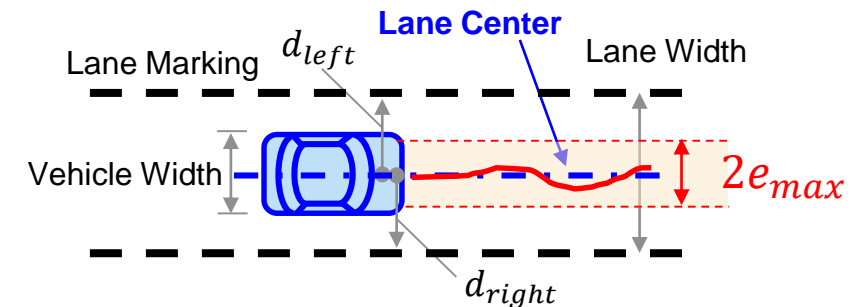
where,

d_{left} : lateral offset of left lane w.r.t. ego car

d_{right} : lateral offset of right lane w.r.t. ego car

e_{max} : allowable lateral deviation

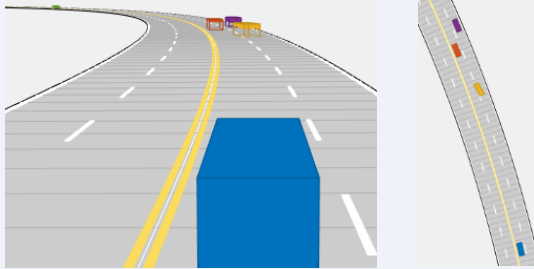
For example, $e_{max} = (LaneWidth - VehicleWidth)/2 = (3.6-1.8)/2 = 0.9$ m



Create Test Scenario using Driving Scenario Designer

Test Description

Lead car cut in and out in curved highway
(curvature of road = $1/500$ m)



Host car

initial velocity = 20.6m/s

HWT(Headway Time) to lead car = 4sec

HW(Headway) to lead car = ~80m

v_{set} (set velocity for ego car) = 21.5m/s

Lead Car

Initially, fast moving car (orange) at 19.4m/s

Passing car (yellow) at 19.6m/s cut in the ego path with HWT=2.3s, then cut out

Third Car

Slow moving car (purple) at 11.1m/s
in the 2nd lane

Driving Scenario Designer - PFACC_05_Curve_CutInOut.mat - Scenario Canvas

DESIGNER

FILE SCENARIO SENSORS SIMULATE VIEW EXPORT

New Open Save Add Road Add Actor Add Camera Add Radar Go to Start Step Back Run Step Forward Settings Repeat Default Layout Export

Roads Actors Scenario Canvas Ego Centric View

Road: 1
Name:
Width (m): 14.7
Bank Angle (deg): 0

Lanes
Number of lanes: [2 2]
Lane Width (m): 3.6
Marking: 1.Solid

Road Centers

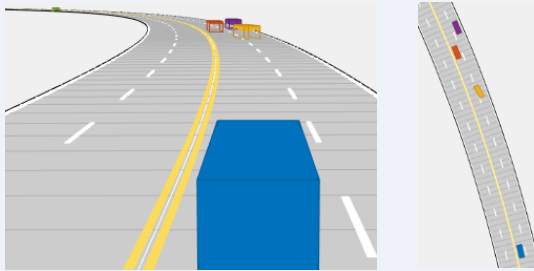
	x (m)	y (m)	z (m)
1	0	-500	
2	34.8782	-498.7820	
3	69.5866	-495.1340	
4	103.9558	-489.0738	
5	137.8187	-480.6308	
6	171.0101	-469.8463	
7	203.3683	-456.7727	
8	234.7358	-441.4738	
9	264.9596	-424.0240	
10	293.8926	-404.5085	

X (m)
Y (m)

Simulation with Simulink Model for Traffic Jam Assist

Test Description

Lead car cut in and out in curved highway
(curvature of road = 1/500 m)



Host car

initial velocity = 20.6m/s
HWT(Headway Time) to lead car = 4sec
HW(Headway) to lead car = ~80m
 v_set (set velocity for ego car) = 21.5m/s

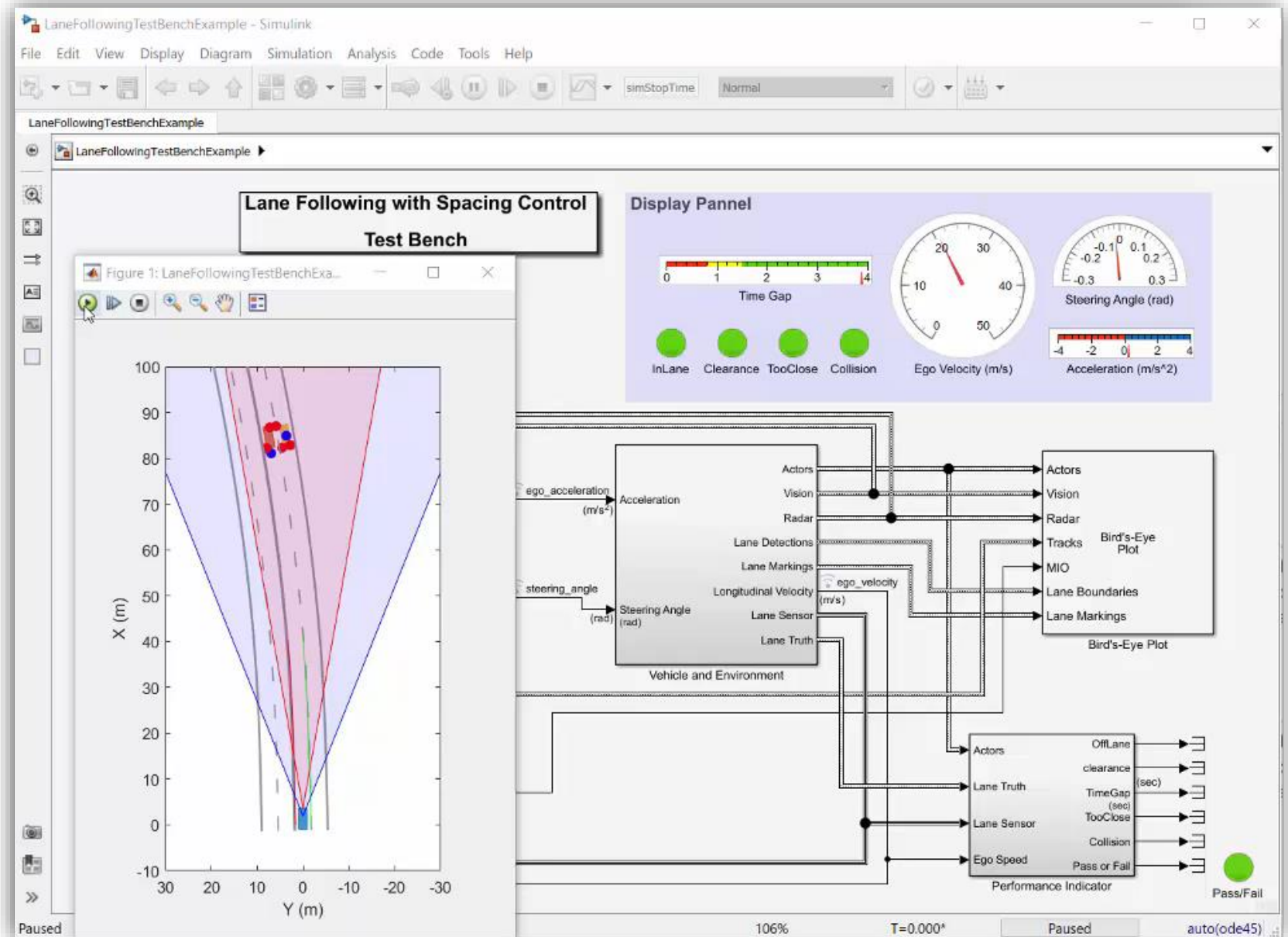
Lead Car

Initially, fast moving car (orange) at 19.4m/s

Passing car (yellow) at 19.6m/s cut in the ego path with HWT=2.3s, then cut out

Third Car

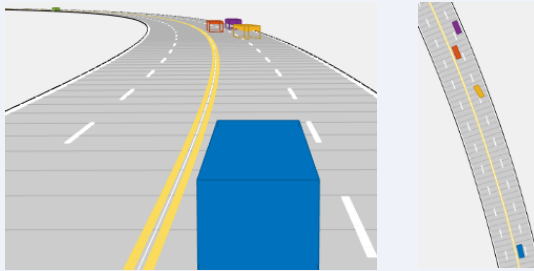
Slow moving car (purple) at 11.1m/s
in the 2nd lane



Simulation with Simulink Model for Traffic Jam Assist

Test Description

Lead car cut in and out in curved highway (curvature of road = 1/500 m)



Host car

initial velocity = 20.6m/s
 HWT(Headway Time) to lead car = 4sec
 HW(Headway) to lead car = ~80m
 v_set (set velocity for ego car) = 21.5m/s

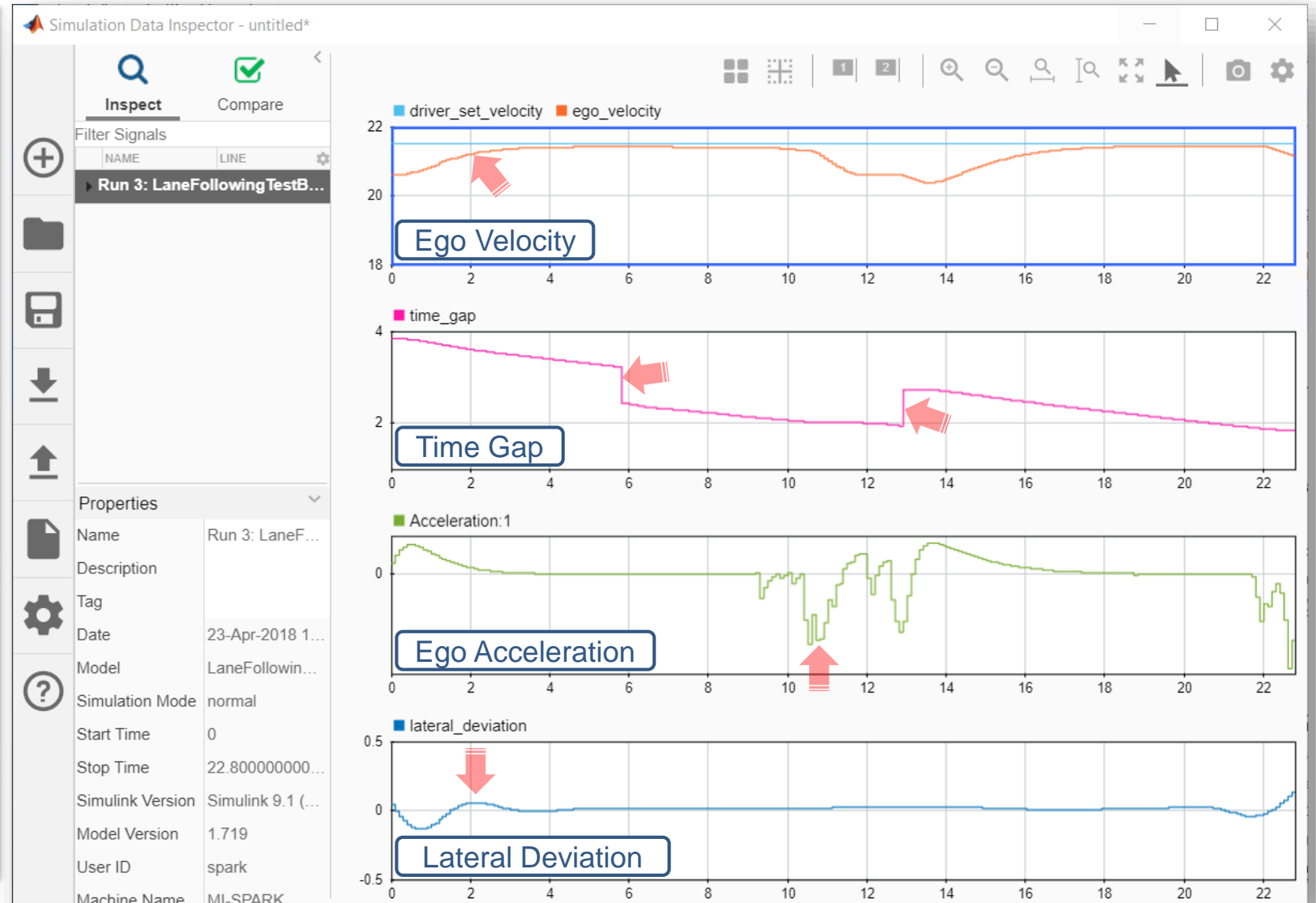
Lead Car

Initially, fast moving car (orange) at 19.4m/s

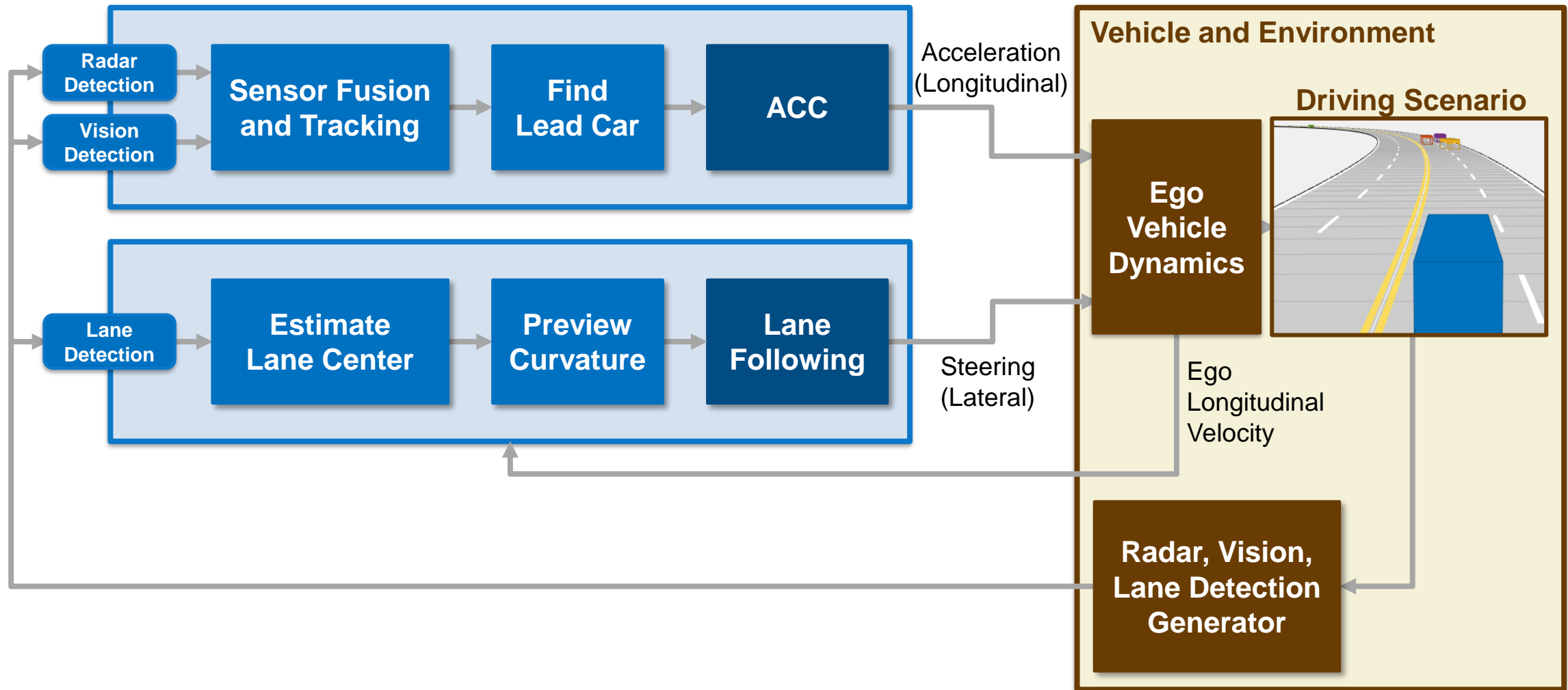
Passing car (yellow) at 19.6m/s cut in the ego path with HWT=2.3s, then cut out

Third Car

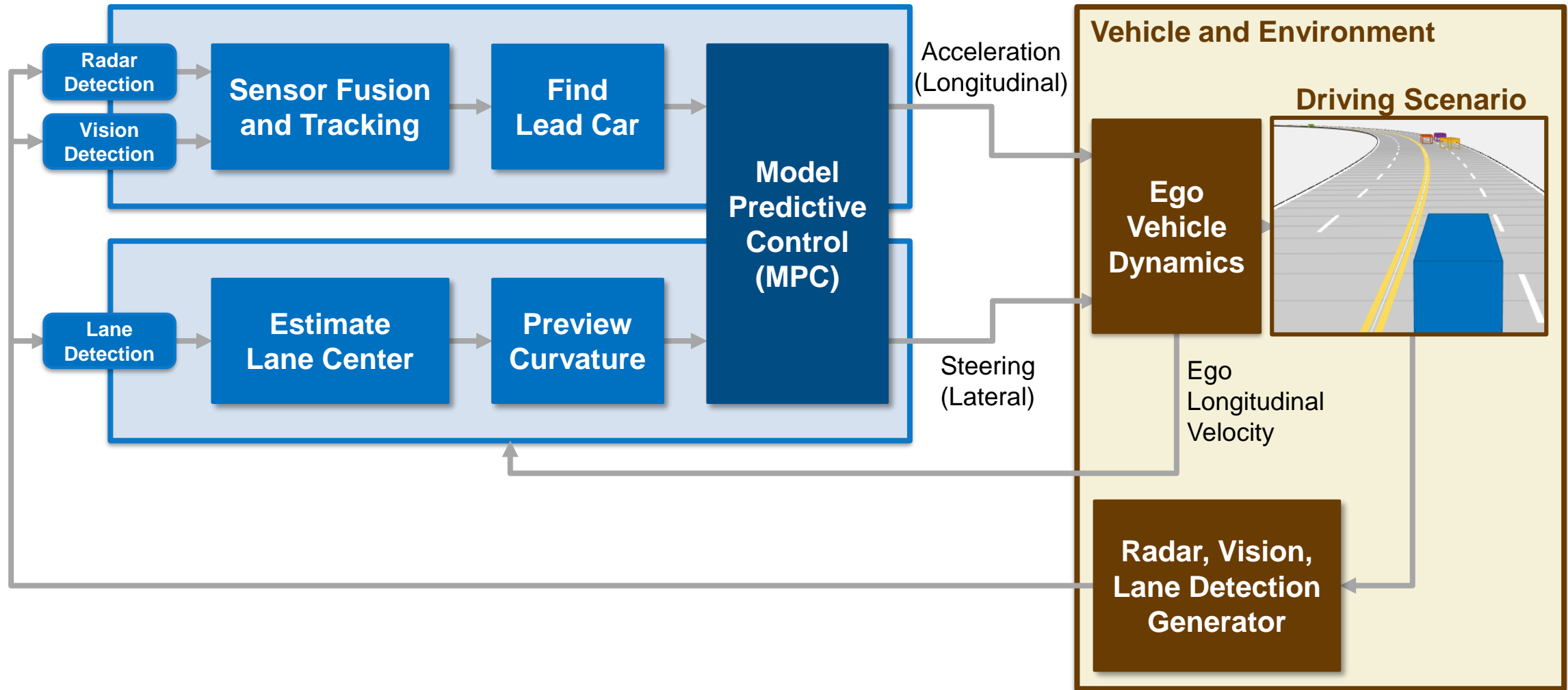
Slow moving car (purple) at 11.1m/s in the 2nd lane



Architecture for ACC and Lane Following Controller

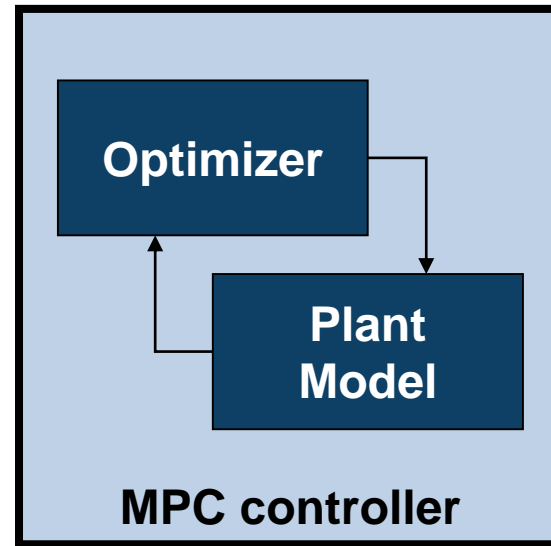


Architecture for ACC and Lane Following Controller



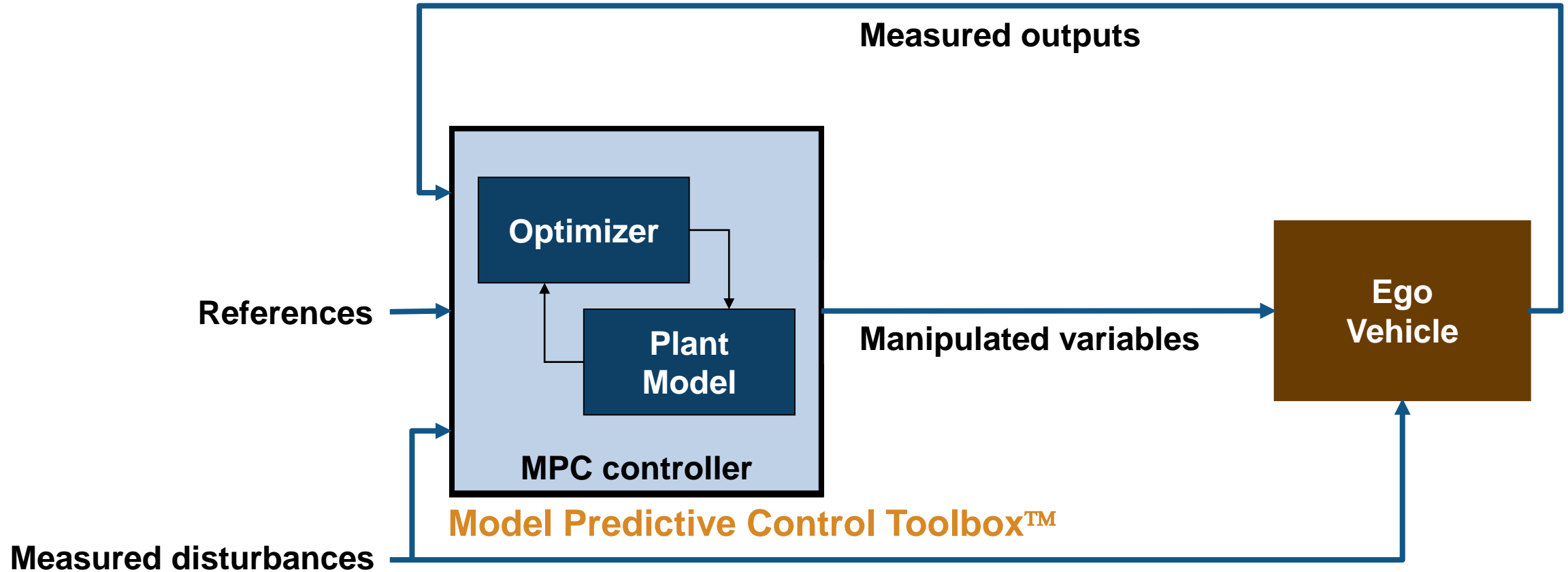
What is model predictive control (MPC)?

- **Multi-variable control** strategy leveraging an internal model to predict plant behavior in the near future
- **Optimizes** for the current timeslot while keeping future timeslots in account

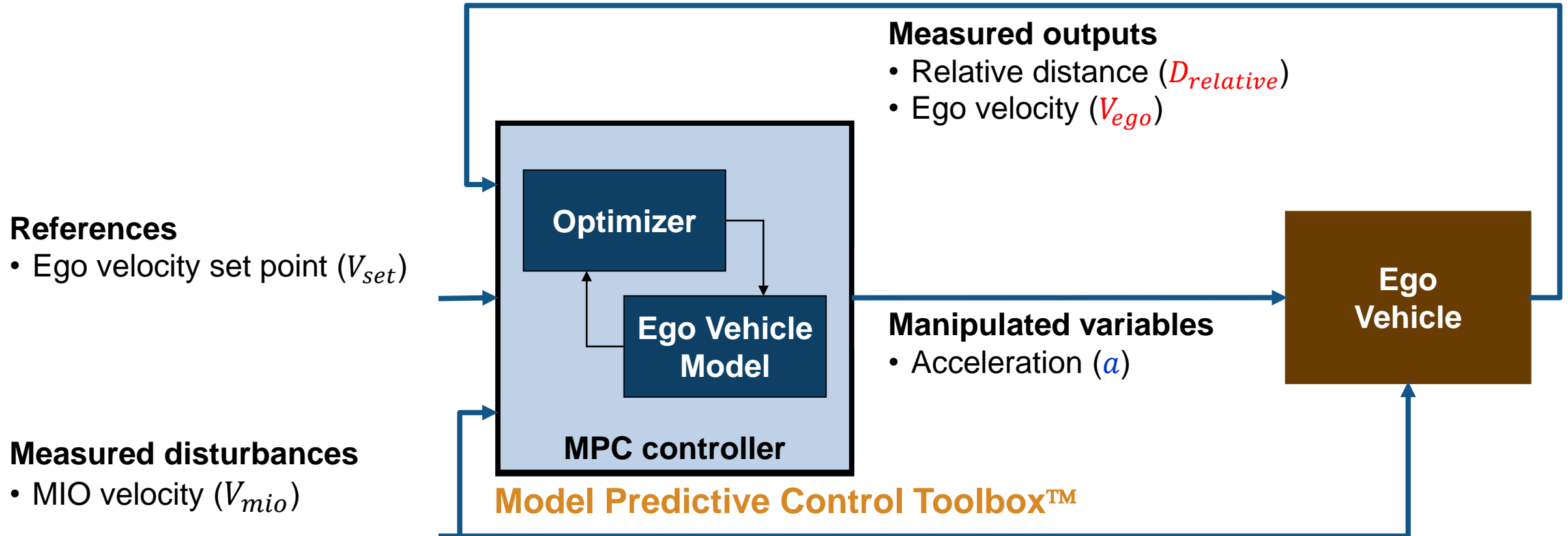


- **Mature** control solution used in industrial applications
- **Gaining popularity in automated driving** applications to improve vehicle responsiveness while maintaining passenger comfort

What is model predictive control (MPC)?



How can MPC be applied to ACC?



How can MPC be applied to ACC and lane following control?

minimize:

$$w_1 |V_{ego} - V_{set}|^2 + w_2 |E_{lateral}|^2$$

References

- Ego velocity set point (V_{set})
- Target lateral deviation (=0)

Measured disturbances

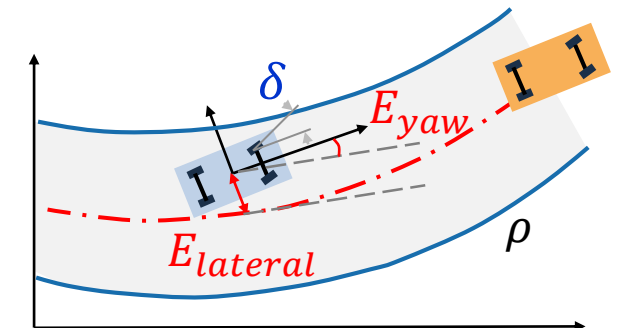
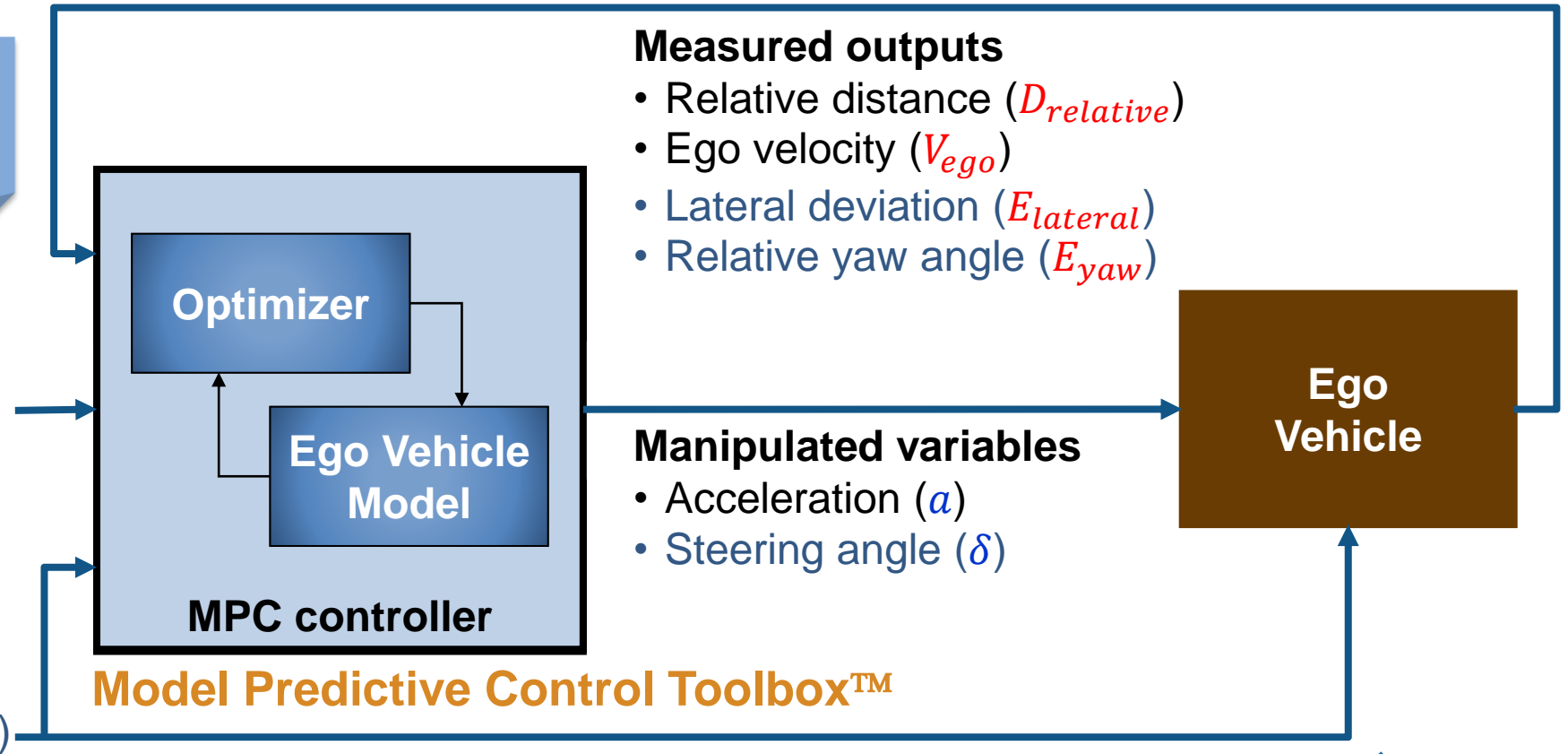
- MIO velocity (V_{mio})
- Previewed road curvature (ρ)

subject to:

$$D_{relative} \geq D_{safe}$$

$$a_{min} \leq a \leq a_{max}$$

$$\delta_{min} \leq \delta \leq \delta_{max}$$



Internal MPC model for ACC and Lane Following Controller



Longitudinal model for ACC

$$\begin{pmatrix} D_{relative} \\ V_{ego} \\ E_{lateral} \\ E_{yaw} \end{pmatrix} = sys \begin{pmatrix} a \\ V_{mio} \\ \delta \\ \rho \end{pmatrix}$$

Measured outputs (OV)

- Relative distance ($D_{relative}$)
- Ego velocity (V_{ego})
- Lateral deviation ($E_{lateral}$)
- Relative yaw angle (E_{yaw})

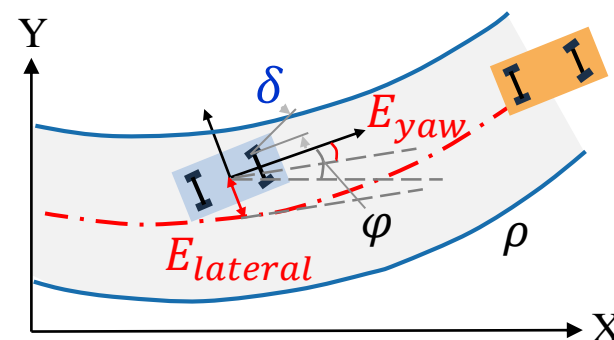
Manipulated variables (MV)

- Acceleration (a)
- Steering angle (δ)

Measured disturbance (MD)

- MIO velocity (V_{mio})
- Previewed road curvature (ρ)

Lateral model for Lane Following



Longitudinal and Lateral Model for MPC

- Longitudinal Model for ACC

$$\frac{d}{dt} \begin{bmatrix} \dot{V}_x \\ V_x \\ D_{relative} \end{bmatrix} = \begin{bmatrix} -\frac{1}{\tau} & 0 & 0 \\ 1 & 0 & 0 \\ 0 & -1 & 0 \end{bmatrix} \begin{bmatrix} \dot{V}_x \\ V_x \\ D_{relative} \end{bmatrix} + \begin{bmatrix} \frac{1}{\tau} & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} a \\ V_{mio} \end{bmatrix}$$

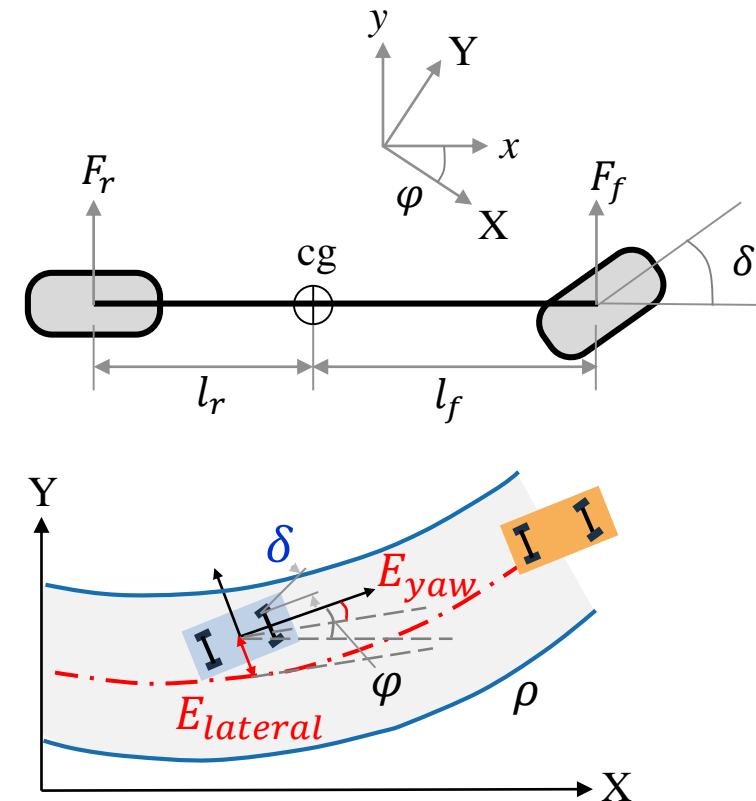
$$\begin{bmatrix} D_{relative} \\ V_x \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \dot{V}_x \\ V_x \\ D_{relative} \end{bmatrix}$$



- Lateral Model for Lane Following

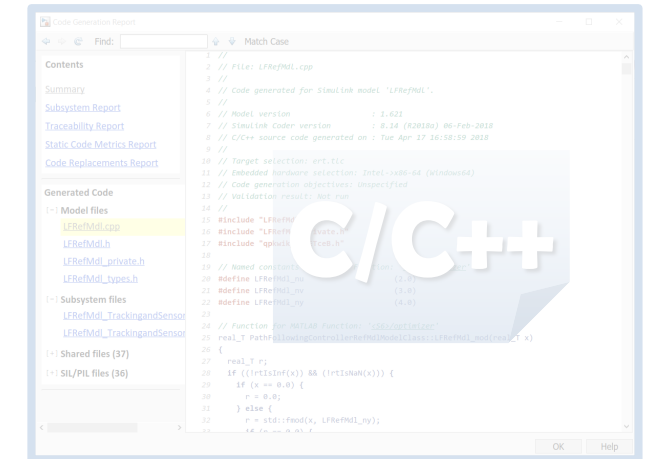
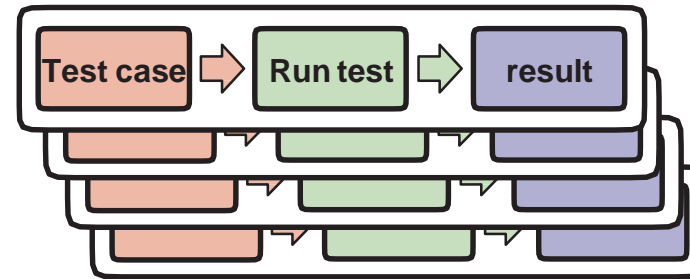
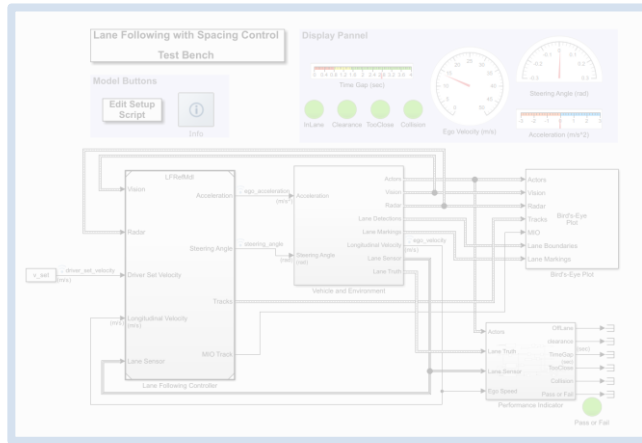
$$\frac{d}{dt} \begin{bmatrix} V_y \\ \dot{\phi} \\ E_{lateral} \\ E_{yaw} \end{bmatrix} = \begin{bmatrix} -\frac{2C_f + 2C_r}{mV_x} & -V_x - \frac{2C_f l_f - 2C_r l_r}{mV_x} & 0 & 0 \\ \frac{2C_f l_f - 2C_r l_r}{I_z V_x} & -\frac{2C_f l_f^2 + 2C_r l_r^2}{I_z V_x} & 0 & 0 \\ 1 & 0 & 0 & V_x \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} V_y \\ \dot{\phi} \\ E_{lateral} \\ E_{yaw} \end{bmatrix} + \begin{bmatrix} \frac{2C_f}{m} & 0 \\ \frac{2C_f l_f}{I_z} & 0 \\ 0 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} \delta \\ V_x \rho \end{bmatrix}$$

$$\begin{bmatrix} E_{lateral} \\ E_{yaw} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} V_y \\ \dot{\phi} \\ E_{lateral} \\ E_{yaw} \end{bmatrix}$$



Automated Driving System Toolbox™

Design and Test Traffic Jam Assist, A Case study



Design ACC and Lane Following Controller

- Create driving scenario
- Synthesize sensor detection
- Include Vehicle Dynamics
- Design sensor fusion algorithm
- Design controller using MPC

Automate Regression Test

- Define performance evaluation metrics
- Develop test cases
- Build test suites
- Verification and validation

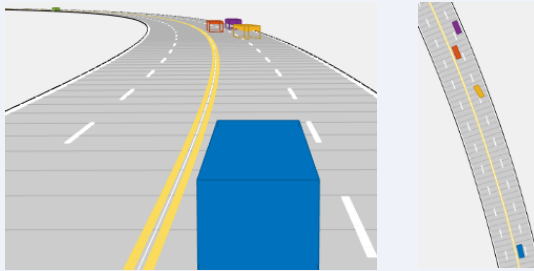
Generate and Verify Code

- SIL test
- Code generation
- Coverage test

Simulation result assessment

Test Description

Lead car cut in and out in curved highway
(curvature of road = 1/500 m)



Host car

initial velocity = 20.6m/s

HWT(Headway Time) to lead car = 4sec

HW(Headway) to lead car = ~80m

v_set(set velocity for ego car) = 21.5m/s

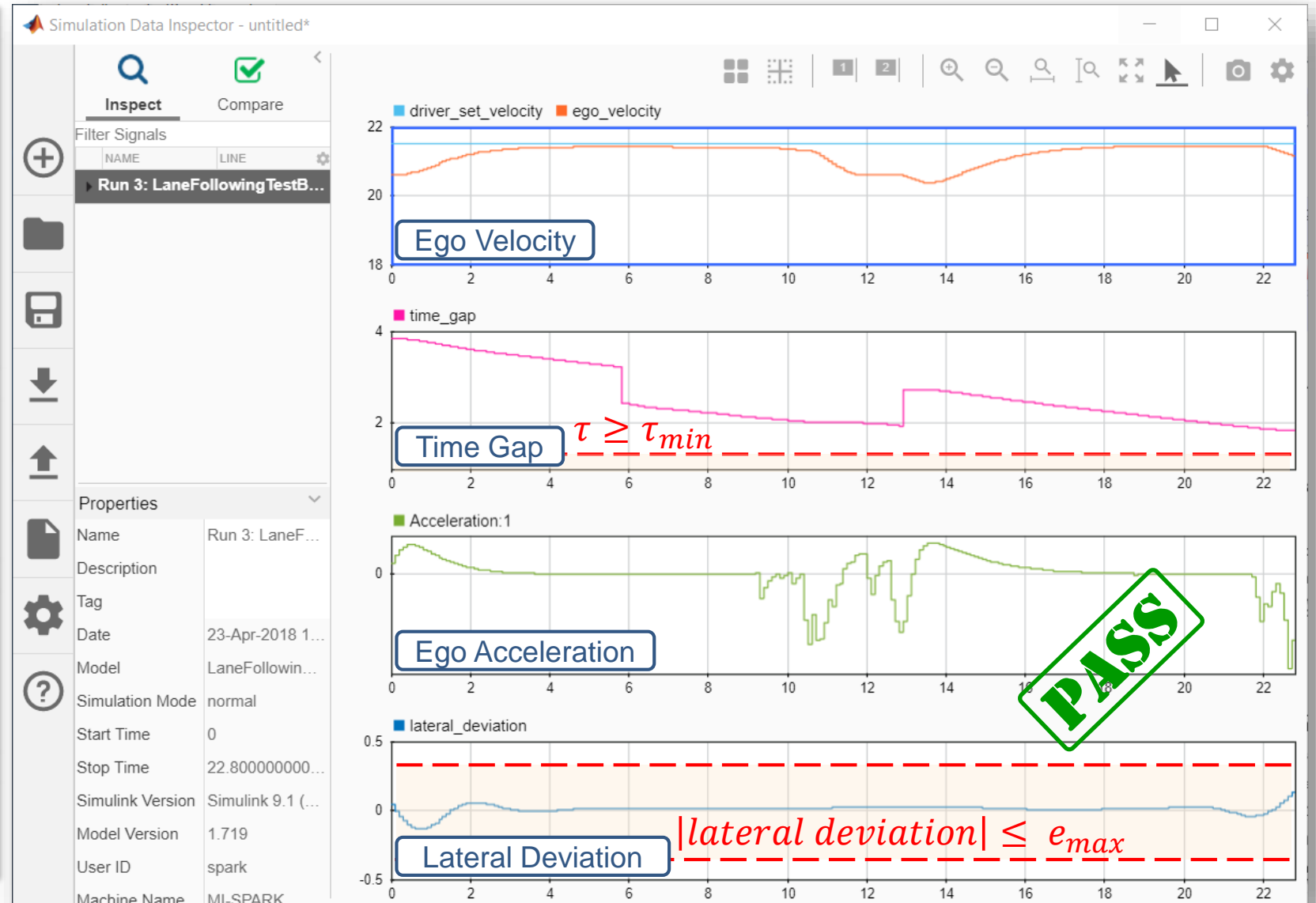
Lead Car

Initially, fast moving car (orange) at 19.4m/s

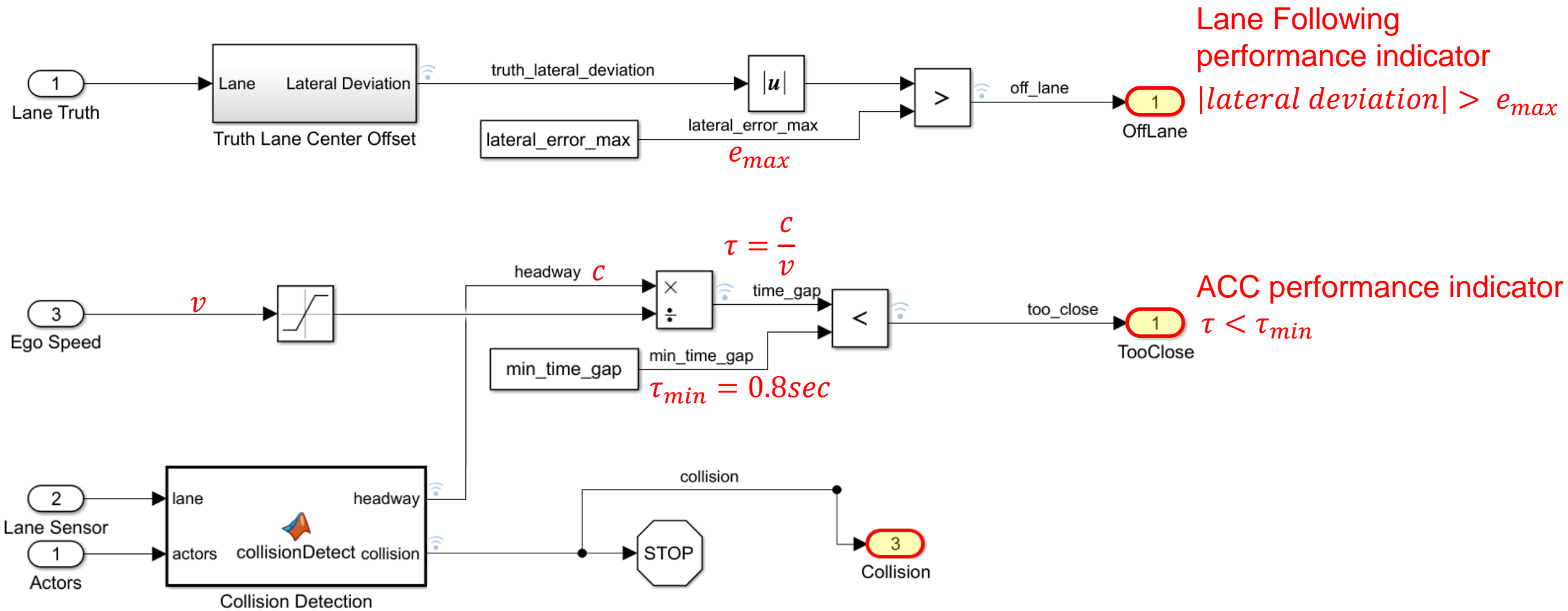
Passing car (yellow) at 19.6m/s cut in the ego path with HWT=2.3s, then cut out

Third Car

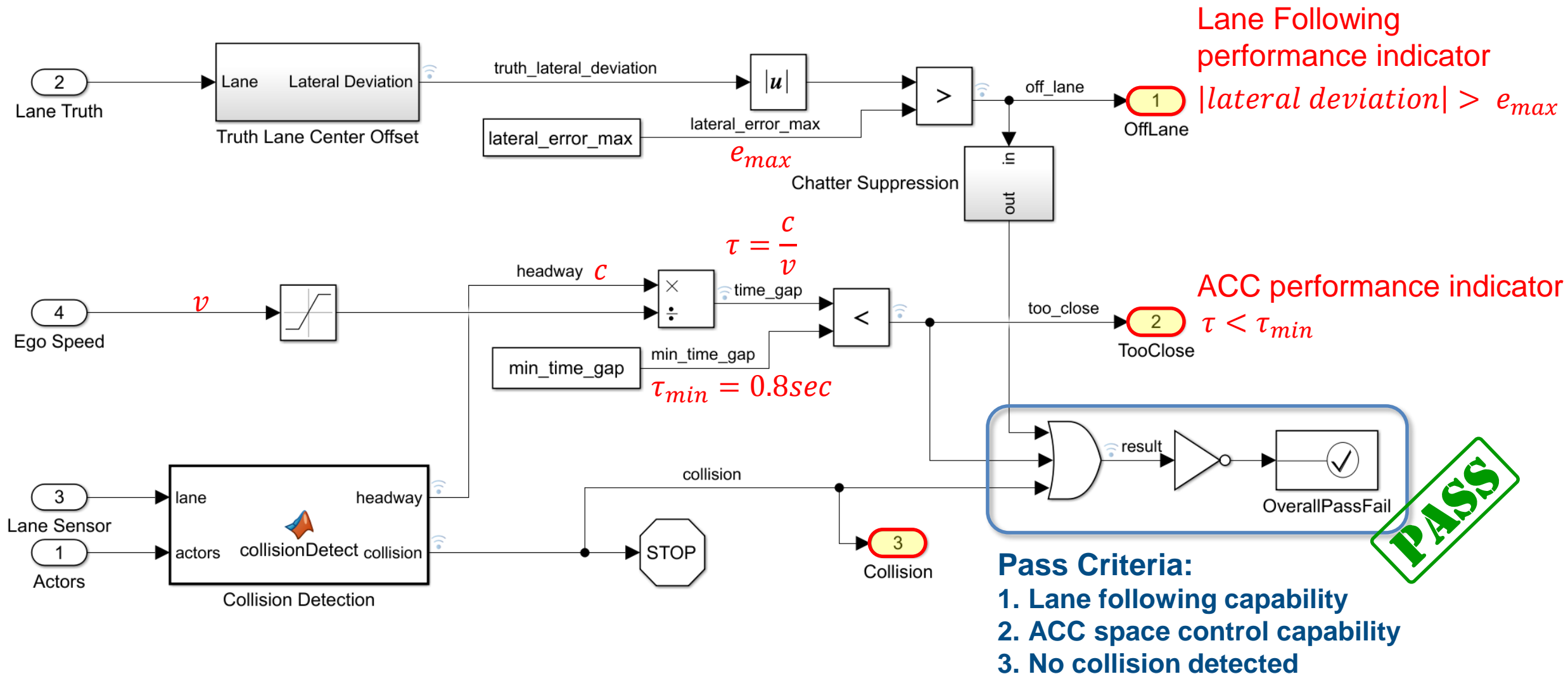
Slow moving car (purple) at 11.1m/s in the 2nd lane



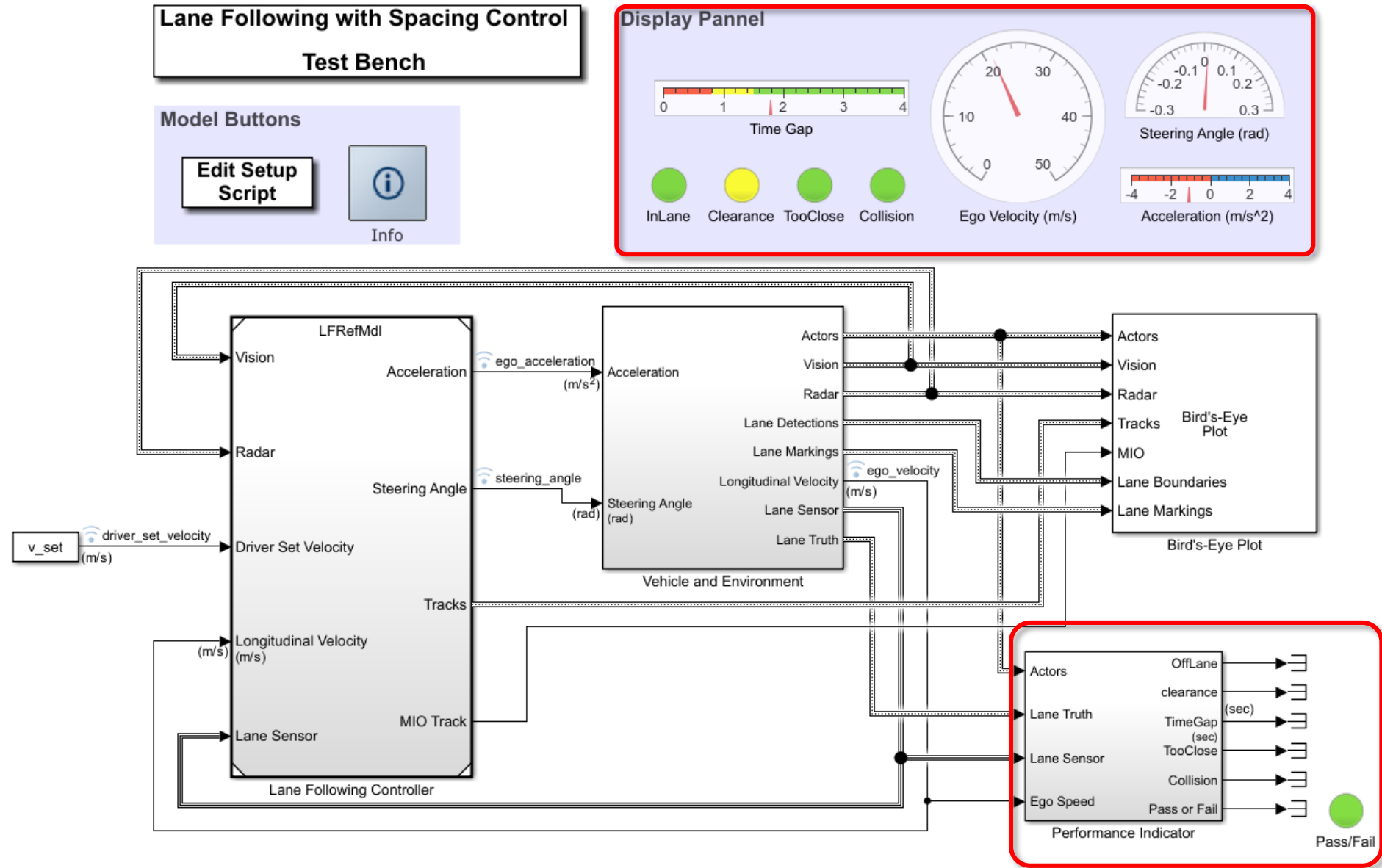
Performance Indicator



Performance Indicator

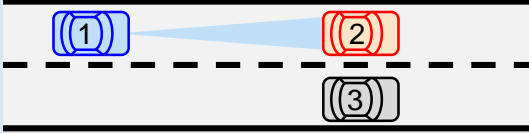
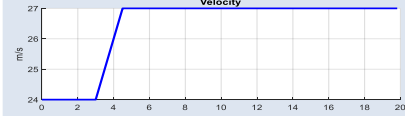
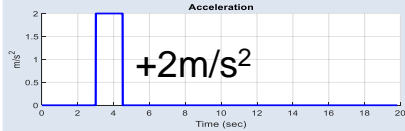
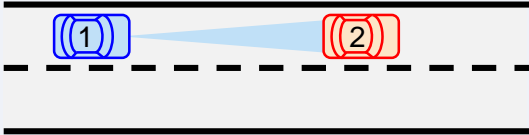


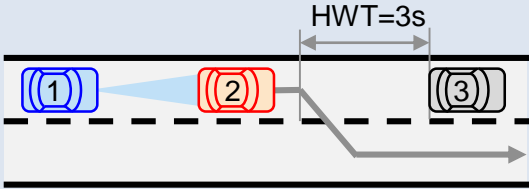


Performance indicator and dashboard in Simulink model



Test scenarios (1/4)

HW : Headway
 HWT : Headway time
 v_set : set velocity for ego car

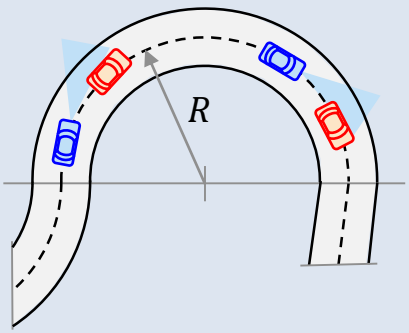
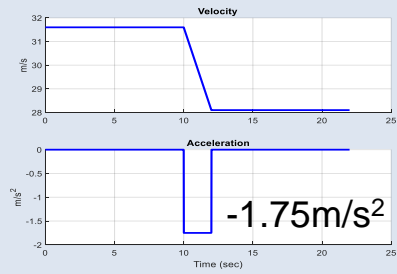
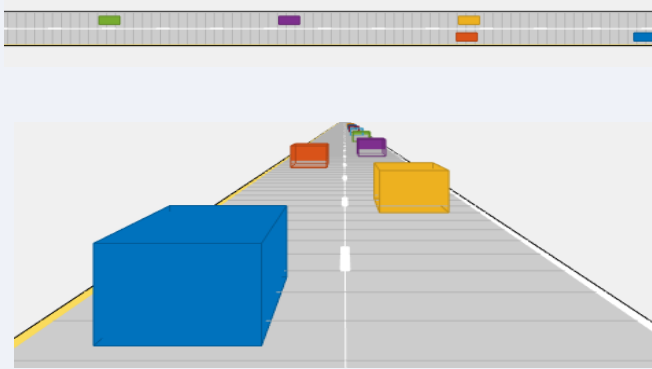
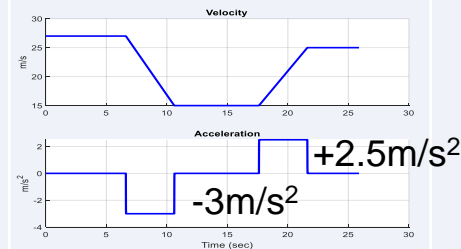
No	Test Name	Test Description	Host car	Lead car	Third car	Spec
1	ACC_01_ISO _TargetDiscriminationTest	Target Discrimination Test 	initial velocity = 30m/s HWT = 2.2sec (HW = 66m) v_set = 30m/s	constant accel 24m/s → 27m/s @ 2m/s ² V _{end} = 27m/s (97.2kph)  	24m/s	ISO 15622 ISO 22178
2	ACC_02_ISO _AutoDecelTest	Automatic Deceleration Test 	initial velocity = 15m/s HWT = 2.2sec (HW = 33m) v_set = 15m/s	initial velocity = 13.9m/s decelerates to full stop with 2.5m/s ²  	none	ISO 22178
3	ACC_03_ISO _AutoRetargetTest	Automatic Retargeting Capability Test 	initial velocity = 15m/s HWT = 2.2sec (HW = 33m) v_set = 15m/s	initial velocity = 13.9m/s Lead car changes lane @ HWT=3s to overtake slow moving car	constant speed = 2.1m/s	ISO 22178

Test scenarios (2/4)

HW : Headway

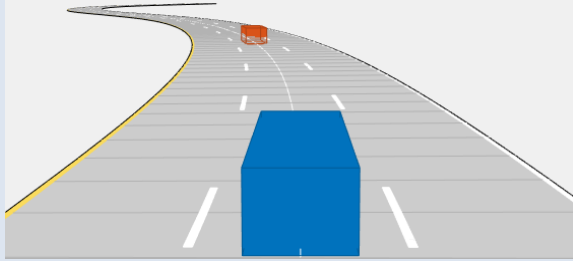
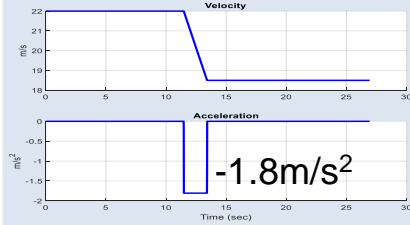
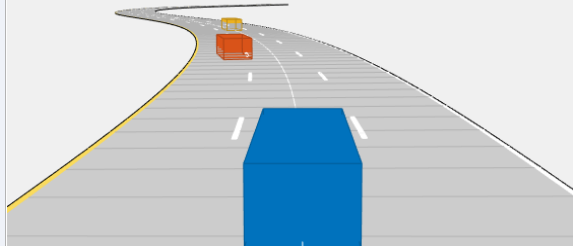
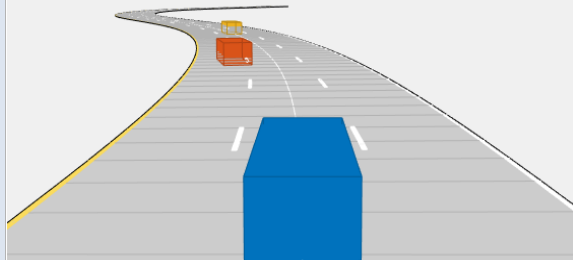
HWT : Headway time

v_set : set velocity for ego car

No	Test Name	Test Description	Host car	Lead car	Third car	Spec
4	ACC_04_ISO_CurveTest	<p>Curve Capability Test (curvature of test track = 1/500 m)</p> 	<p>initial velocity = 31.6m/s</p> <p>HWT = 1.5sec (HW = 47.4m)</p> <p>v_set = 31.6m/s</p>	<p>initial velocity = 31.6m/s</p> <p>Drive at a constant speed for 10s, decrease speed by 3.5m/s in 2s, and keep it constant.</p> 	none	<p>ISO 15622</p> <p>ISO 22178</p>
5	ACC_05_StopnGo	<p>Stop and Go in highway</p> 	<p>initial velocity = 27m/s</p> <p>HWT = 1.5sec (HW = 40.5m)</p> <p>v_set = 27m/s</p>	<p>initial velocity = 27m/s</p> <p>Lead car slows down to 15m/s at -3m/s² and stay constant for 7s, then speed up to 25m/s at 2.5m/s²</p> 	<p>8 slow moving cars at 12m/s in the second lane</p> <p>Real-world scenario</p>	

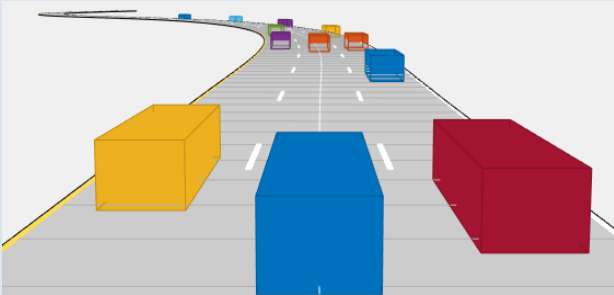
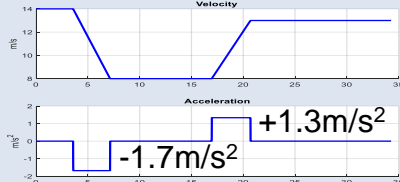
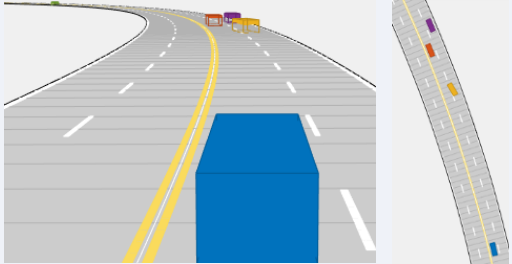
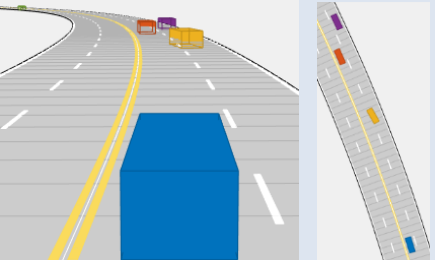
Test scenarios (3/4)

HW : Headway
 HWT : Headway time
 v_set : set velocity for ego car

No	Test Name	Test Description	Host car	Lead car	Third car	Spec
6	LFACC_01_DoubleCurve _DecelTarget (Similar with ACC_04_ISO _CurveTest)	Automatic Deceleration Test 	initial velocity = 22m/s HWT = 2sec (HW = 44m) v_set = 22m/s	initial velocity = 22m/s Drive at a constant speed for about 11s, decrease speed by 3.5m/s in 2s (deceleration: -1.8 m/s ²) and keep it const. 	none	Real-world scenario
7	LFACC_02_DoubleCurve _AutoRetarget_TooSlow (Similar with ACC_03_ISO _AutoRetargetTest)	Automatic Retargeting Capability Test 	initial velocity = 15m/s HWT = 2.8sec (HW = 43m) v_set = 15m/s	initial velocity = 13.9m/s Lead car changes lane @ HWT=3s to overtake slow moving car	Slow moving car at constant speed = 2.1m/s	~ISO 22178
8	LFACC_03_DoubleCurve _AutoRetarget (Similar with ACC_03_ISO _AutoRetargetTest)	Automatic Retargeting Capability Test 	initial velocity = 15m/s HWT = 2.8sec (HW = 43m) v_set = 15m/s	initial velocity = 13.9m/s Lead car changes lane @ HWT=3s to overtake slow moving car	Slow moving car at constant speed = 10m/s	~ISO 22178

Test scenarios (4/4)

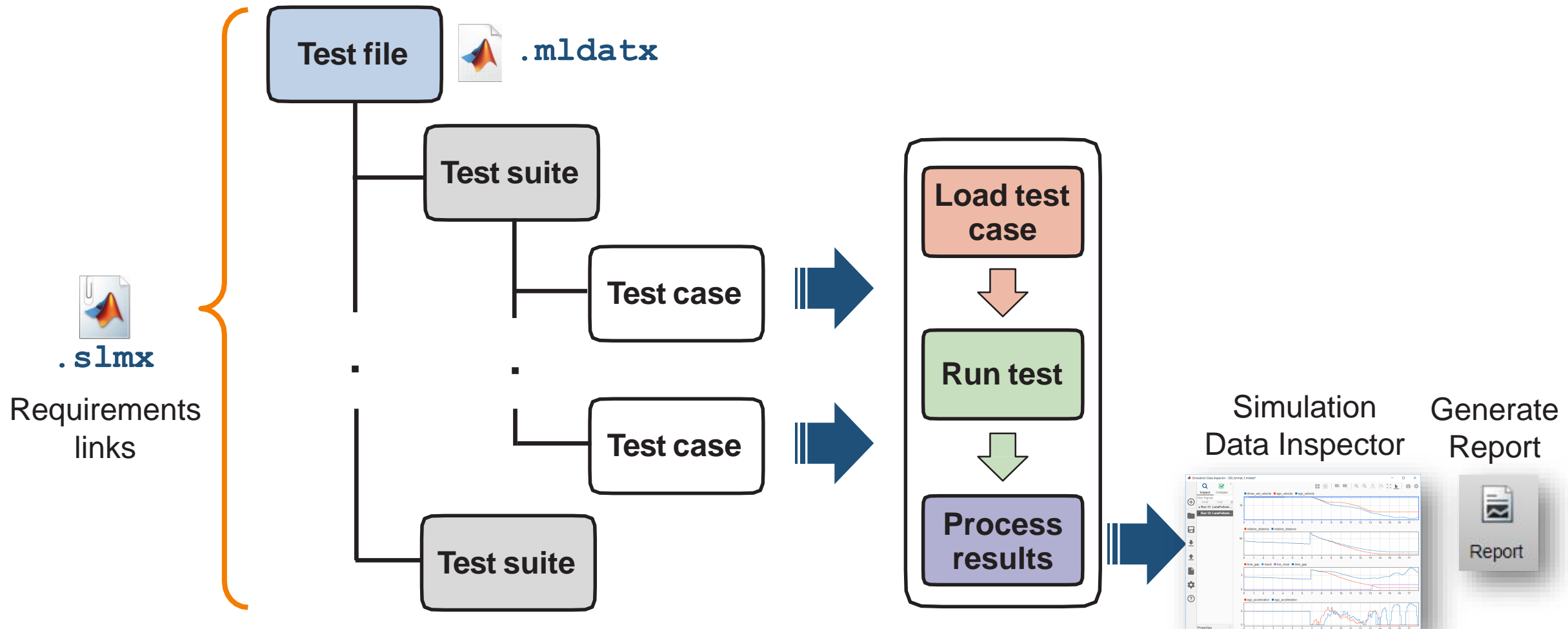
HW : Headway
 HWT : Headway time
 v_set : set velocity for ego car

No	Test Name	Test Description	Host car	Lead car	Third car	Spec
9	LFACC_04_DoubleCurve_StopnGo (Similar with ACC_05_StopnGo)	Stop and Go in curved highway 	initial velocity = 14m/s HWT = 3.6sec (HW = 50m) v_set = 14m/s	initial velocity = 14m/s Lead car slows down to 8m/s at 8m/s in the 3 rd lane at -1.7m/s ² and stay constant for 10s, then speed up to 13m/s at 1.3m/s ² 	10 slow moving cars in the 1 st lane	Real-world scenario
10	LFACC_05_Curve_CutInOut	Lead car cut in and out in curved highway (curvature of road = 1/500 m) 	initial velocity = 20.6m/s HWT = 4sec (HW = ~80m) v_set = 21.5m/s	Initially, fast moving car (orange) at 19.4m/s Passing car (yellow) at 19.6m/s cut in the ego path with HWT=2.3s, then cut out	Slow moving car (purple) at 11.1m/s in the 2 nd lane	Real-world scenario
11	LFACC_06_Curve_CutInOut_TooClose	Lead car cut in and out in curved highway (curvature of road = 1/500 m) 	initial velocity = 20.6m/s HWT = 4sec (HW = ~80m) v_set = 21.5m/s	Initially, fast moving car (orange) at 19.4m/s Passing car (yellow) at 19.6m/s cut in the ego path with HWT=1.5s, then cut out	Slow moving car (purple) at 11.1m/s in the 2 nd lane	Real-world scenario

Representative test scenario

Test Manager in Simulink® Test™

- Automate Simulink model testing using test cases with pass-fail criteria



Test Report with baseline parameter set for 11 test cases

Report Generated by Test Manager

Title: ACCAndLaneFollowing (baseline)
Author: Seo-Wook Park
Date: 21-Apr-2018 16:01:50

Test Environment

Platform: PCWIN64
MATLAB: (R2018a)

Note) Baseline parameter set was tuned based on a single test scenario.



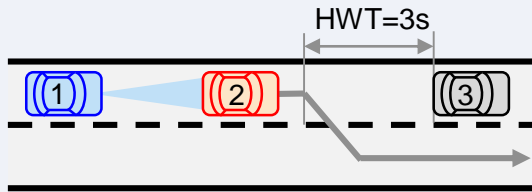
Summary

Name	Outcome	Duration (Seconds)
TestScenarios Baseline	8 ✓ 3 ✗	565
ACCTest	3 ✓ 2 ✗	210
ACC 01 ISO TargetDiscriminationTest	✓	35
ACC 02 ISO AutoDecelTest	✗	22
ACC 03 ISO AutoRetargetTest	✗	32
ACC 04 ISO CurveTest	✓	43
ACC 05 StopnGo	✓	73
LFACCTest	5 ✓ 1 ✗	354
LFACC 01 DoubleCurve DecelTarget	✓	43
LFACC 02 DoubleCurve AutoRetarget TooS low	✗	36
LFACC 03 DoubleCurve AutoRetarget	✓	65
LFACC 04 DoubleCurve StopnGo	✓	111
LFACC 05 Curve CutInOut	✓	48
LFACC 06 Curve CutInOut TooClose	✓	49

Fine-tune control parameters (1/3)

Test Description

Automatic Retargeting Capability Test



Host car

initial velocity = 15m/s

HWT = 2.2sec (HW = 33m)

v_set = 15m/s

Lead Car

initial velocity = 13.9m/s

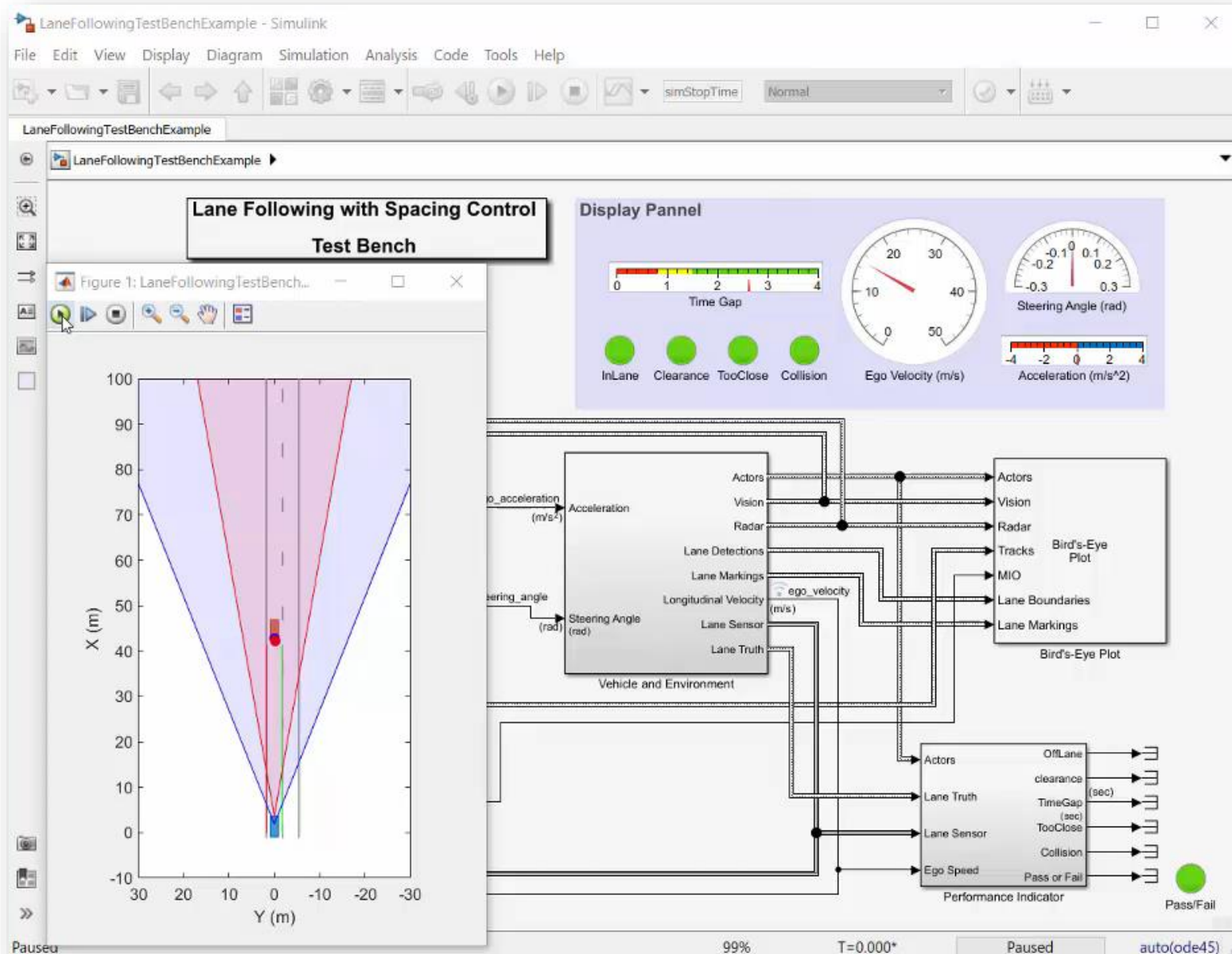
Lead car changes lane @ HWT=3s to overtake slow moving car

Third Car

constant speed = 2.1m/s

Spec

ISO 22178



Fine-tune control parameters (1/3)

Test Description
Automatic Retargeting Capability Test

Host car
 initial velocity = 15m/s
 HWT = 2.2sec (HW = 33m)
 v_set = 15m/s

Lead Car
 initial velocity = 13.9m/s

Lead car changes lane @ HWT=3s to overtake slow moving car

Third Car
 constant speed = 2.1m/s

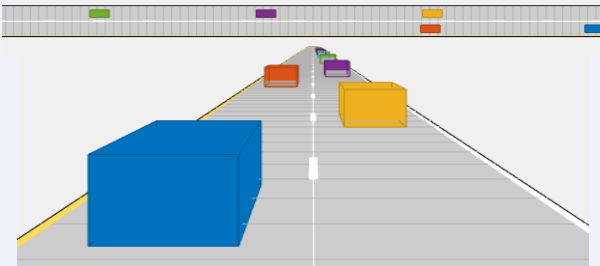
Spec
 ISO 22178



Fine-tune control parameters (2/3)

Test Description

Stop and Go in highway



Host car

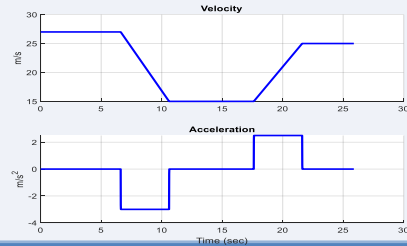
initial velocity = 27m/s

HWT = 1.5sec (HW = 40.5m)

v_set = 27m/s

Lead Car

initial velocity = 27m/s



Third Car

8 slow moving cars at 12m/s

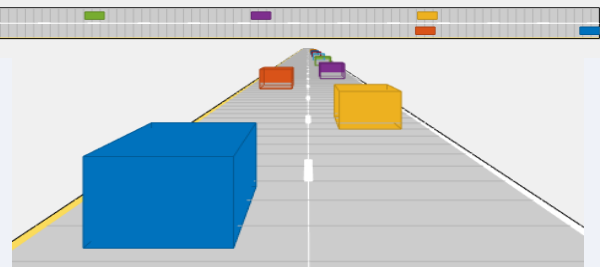
in the second lane

Spec

Real-world scenario

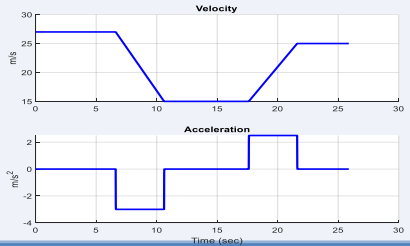
Fine-tune control parameters (2/3)

Test Description
Stop and Go in highway

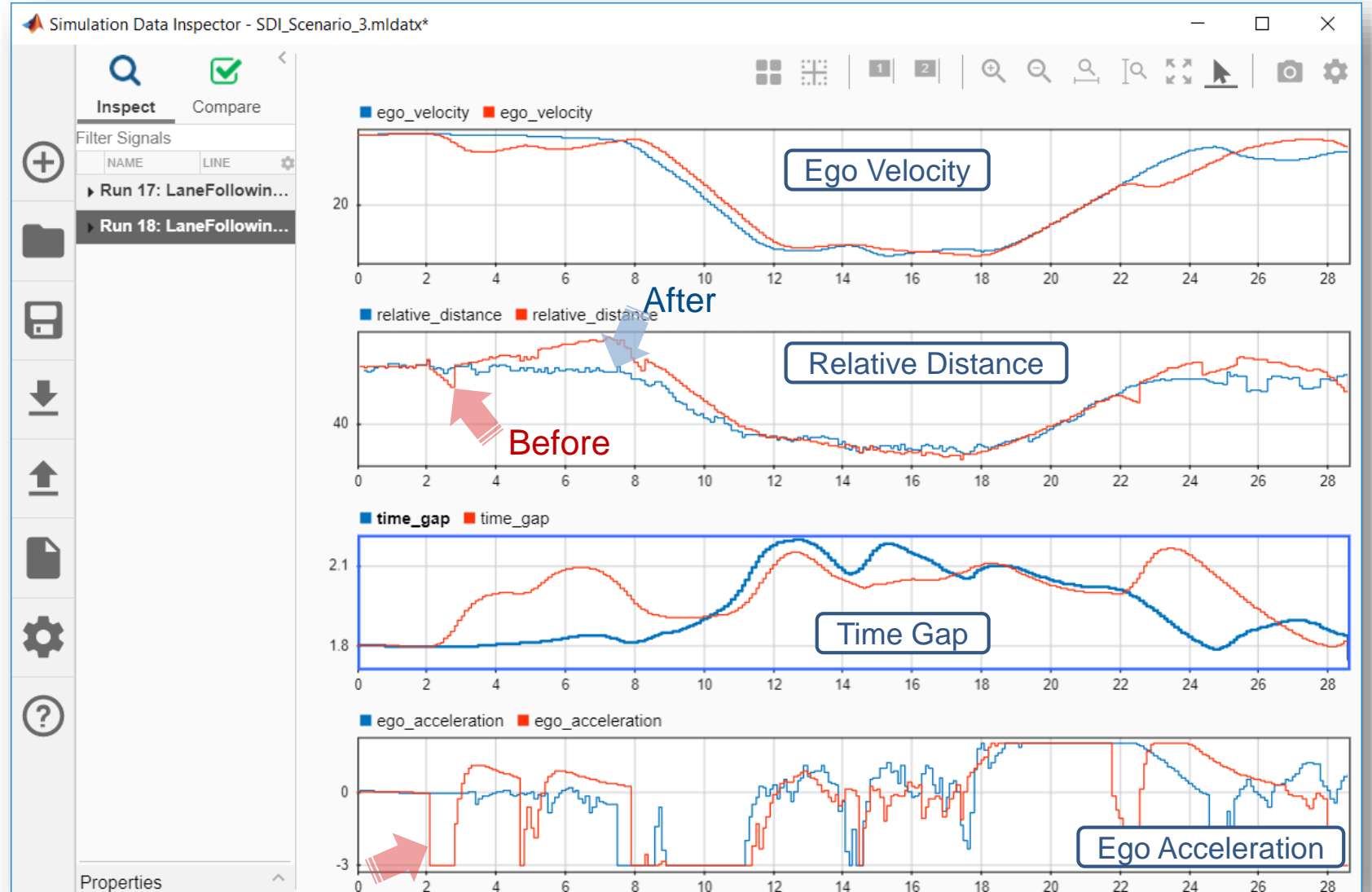


Host car
 initial velocity = 27m/s
 HWT = 1.5sec (HW = 40.5m)
 v_set = 27m/s

Lead Car
 initial velocity = 27m/s



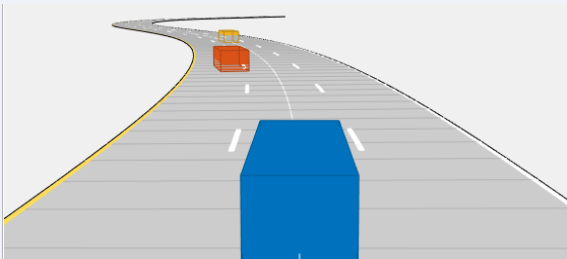
Third Car
 8 slow moving cars at 12m/s
 in the second lane
Spec
 Real-world scenario



Fine-tune control parameters (3/3)

Test Description

Automatic Retargeting Capability Test



Host car

initial velocity = 15m/s

HWT = 2.8sec (HW = 43m)

v_set = 15m/s

Lead Car

initial velocity = 13.9m/s

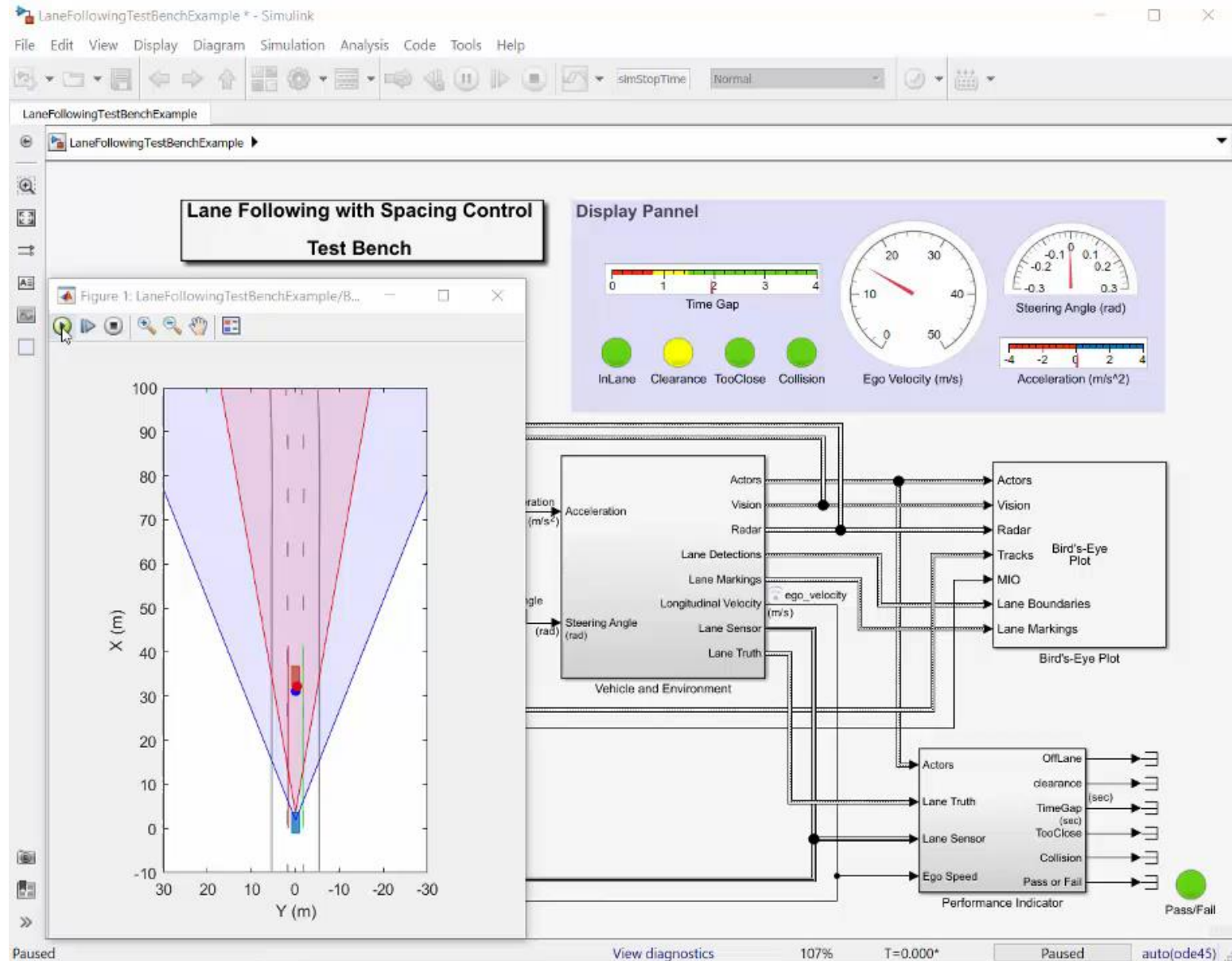
Lead car changes lane @ HWT=3s to overtake slow moving car

Third Car

Slow moving car at constant speed, 2.1m/s

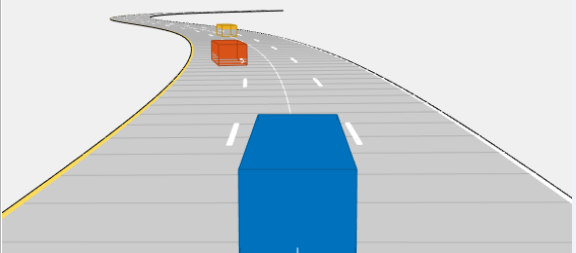
Spec

~ISO 22178



Fine-tune control parameters (3/3)

Test Description
Automatic Retargeting Capability Test



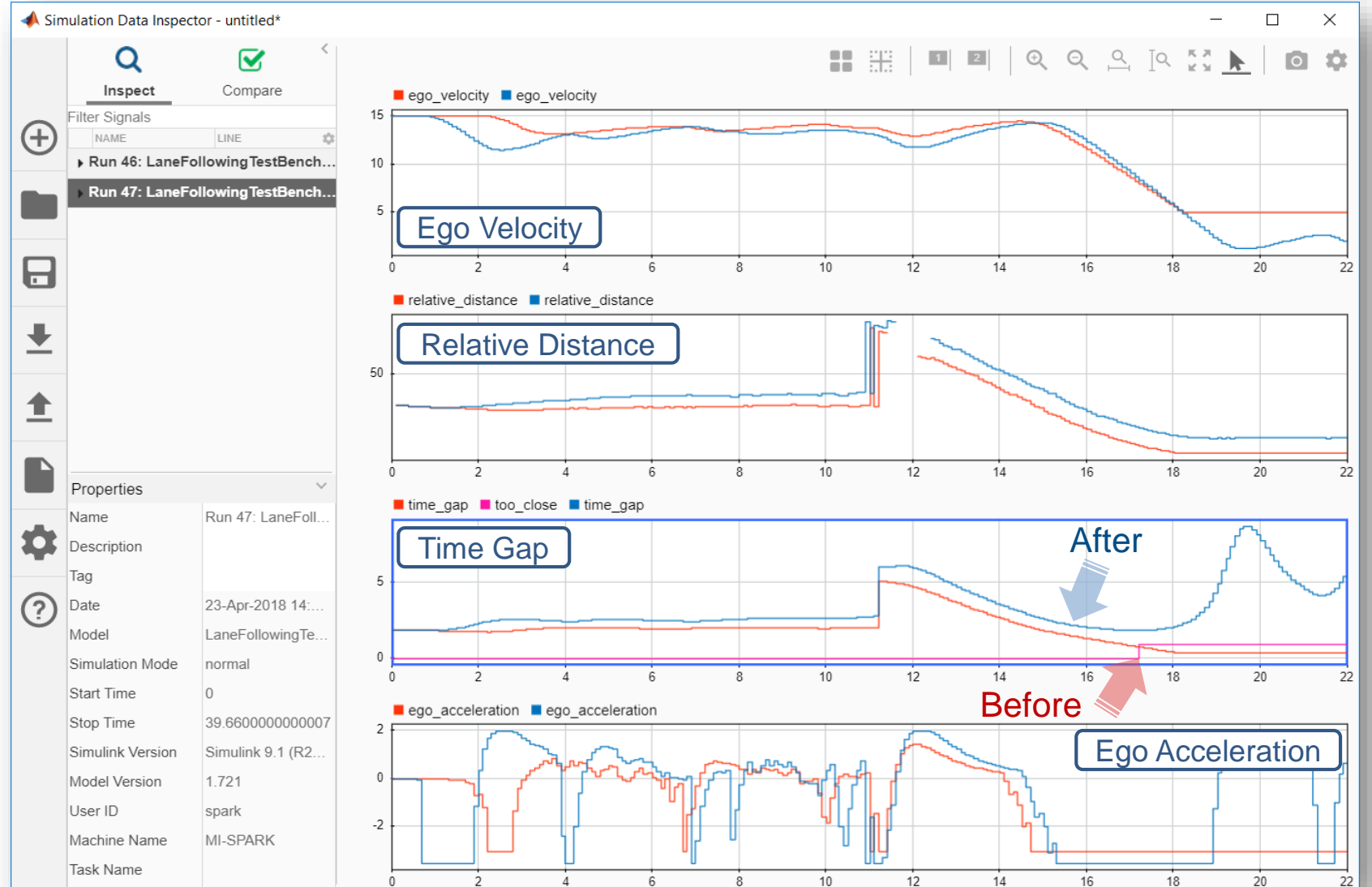
Host car
initial velocity = 15m/s
HWT = 2.8sec (HW = 43m)
v_set = 15m/s

Lead Car
initial velocity = 13.9m/s

Lead car changes lane @ HWT=3s to overtake slow moving car

Third Car
Slow moving car at constant speed, 2.1m/s

Spec
~ISO 22178



Baseline vs. Fine-tuned parameters

Parameter Name	Description	Baseline	Fine-tuned
<code>assigThresh</code>	Detection assignment threshold for <code>multiObjectTracker</code>	50	20
<code>time_gap</code>	ACC time gap (sec)	1.5	2.0
<code>default_spacing</code>	ACC safe distance margin (m)	0	10
<code>min_ac</code>	Minimum acceleration (m/s^2)	-3.0	-3.5

Test Report with fine-tuned parameter set for 11 test cases

Report Generated by Test Manager

Title: ACCAndLaneFollowing **Fine-tuned**
Author: Seo-Wook Park
Date: 26-Apr-2018 13:53:39

Test Environment

Platform: PCWIN64
MATLAB: (R2018a)

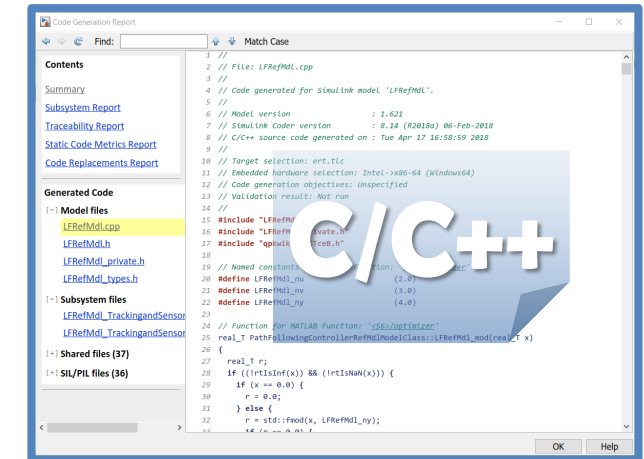
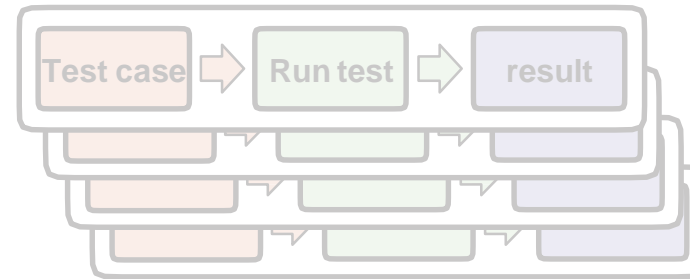
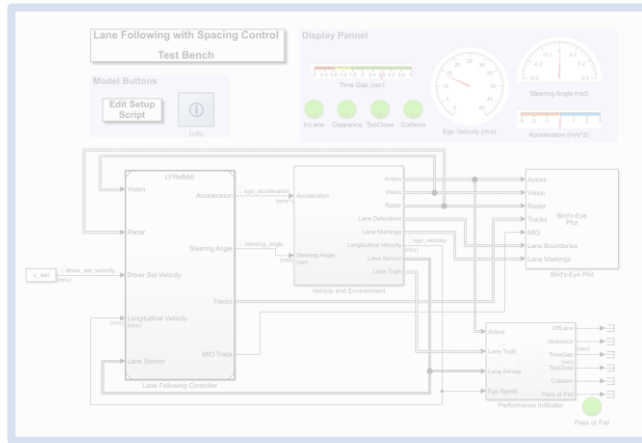
Summary

Name	Outcome	Duration (Seconds)
TestScenarios FineTuned	11 ✓	3541
ACCTest	5 ✓	1521
ACC 01 ISO TargetDiscriminationTest	✓	245
ACC 02 ISO AutoDecelTest	✓	323
ACC 03 ISO AutoRetargetTest	✓	262
ACC 04 ISO CurveTest	✓	331
ACC 05 StopnGo	✓	360
LFACCTest	6 ✓	2015
LFACC 01 DoubleCurve DecelTarget	✓	333
LFACC 02 DoubleCurve AutoRetarget TooS low	✓	380
LFACC 03 DoubleCurve AutoRetarget	✓	291
LFACC 04 DoubleCurve StopnGo	✓	398
LFACC 05 Curve CutInOut	✓	335
LFACC 06 Curve CutInOut TooClose	✓	278



Automated Driving System Toolbox™

Design and Test Traffic Jam Assist, A Case study



Design ACC and Lane Following Controller

- Create driving scenario
- Synthesize sensor detection
- Include Vehicle Dynamics
- Design sensor fusion algorithm
- Design controller using MPC

Automate Regression Test

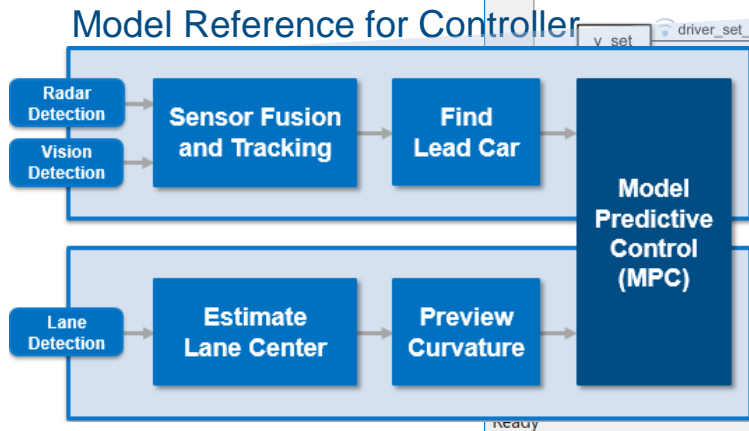
- Define performance evaluation metrics
- Develop test cases
- Build test suites
- Verification and validation

Generate and Verify Code

- SIL test
- Code generation
- Coverage test

Simulation with SIL mode

The screenshot shows the Simulink environment for a lane following test bench. The main window displays a 'Lane Following with Spacing Control Test Bench' with a 'Display Panel' showing gauges for acceleration and steering angle. A 'Block Parameters: Lane Following Controller' dialog box is open, showing 'Simulation mode' set to 'Software-in-the-loop (SIL)'. A detailed block diagram of the 'Lane Following Controller' is shown, including 'Sensor Fusion and Tracking', 'Find Lead Car', 'Model Predictive Control (MPC)', 'Estimate Lane Center', and 'Preview Curvature' blocks.



Code Generation Report

Embedded Coder™

Code Generation Report

Find: Match Case

Contents

- [Summary](#)
- [Subsystem Report](#)
- [Traceability Report](#)
- [Static Code Metrics Report](#)
- [Code Replacements Report](#)

Generated Code

- [-] Model files**
 - [LFRfMdl.cpp](#)
 - [LFRfMdl.h](#)
 - [LFRfMdl_private.h](#)
 - [LFRfMdl_types.h](#)
- [-] Subsystem files**
 - [LFRfMdl_TrackingandSensor](#)
 - [LFRfMdl_TrackingandSensor](#)
- [+] Shared files (37)**
- [+] SIL/PIL files (36)**

```

1 //
2 // File: LFRfMdl.cpp
3 //
4 // Code generated for Simulink model 'LFRfMdl'.
5 //
6 // Model version           : 1.621
7 // Simulink Coder version   : 8.14 (R2018a) 06-Feb-2018
8 // C/C++ source code generated on : Tue Apr 17 16:58:59 2018
9 //
10 // Target selection: ert.tlc
11 // Embedded hardware selection: Intel->x86-64 (Windows64)
12 // Code generation objectives: Unspecified
13 // Validation result: Not run
14 //
15 #include "LFRfMdl.h"
16 #include "LFRfMdl_private.h"
17 #include "qpkwik_YRMETceB.h"
18
19 // Named constants for MATLAB Function: '<S6>/optimizer'
20 #define LFRfMdl_nu           (2.0)
21 #define LFRfMdl_nv           (3.0)
22 #define LFRfMdl_ny           (4.0)
23
24 // Function for MATLAB Function: '<S6>/optimizer'
25 real_T PathFollowingControllerRefMdlModelClass::LFRfMdl_mod(real_T x)
26 {
27     real_T r;
28     if ((!rtIsInf(x)) && (!rtIsNaN(x))) {
29         if (x == 0.0) {
30             r = 0.0;
31         } else {
32             r = std::fmod(x, LFRfMdl_ny);
33         }
34     }
35 }

```

OK Help

Aggregated Code Coverage Report



Report Generated

Title: ACCAndLane
Author: Seo-Wook Pa
Date: 26-Apr-2018

Test Environment

Platform: PCWIN64
 MATLAB: (R2018a)

Summary

TOTAL COVERAGE

1. [LRefMdl.cpp](#)
2. [LRefMdl_TrackingandS](#)
3. [rtGetInf.cpp](#)
4. [rtGetNaN.cpp](#)
5. [rt_nonfinite.cpp](#)

Summary By Model

Model Object

1. [LRefMdl](#)
2. [Controller](#)
3. [MPC Controller](#)
4. [MPC](#)
5. [optimizer](#)
6. [Safe distance](#)
7. [Estimate Lane Center](#)
8. [Center from Left](#)
9. [Center from Left and](#)
10. [Center from Right](#)
11. [MATLAB Function](#)
12. [Preview curvature](#)
13. [Tracking and Sensor Fus](#)
14. [Clock](#)
15. [Counter Limited](#)
16. [Find Lead Car](#)

```

1778 static boolean_T LRefMdl_objectTrack_checkPromotion(const
1779   driving_internal_objectTrack_LRefMdl_T *track)
1780 {
1781   boolean_T toPromote;
1782   real_T history;
1783   int32_T b;
1784   boolean_T track_data[50];
1785   int32_T track_size[2];
1786   if (track->ObjectClassID != 0.0) {
1787     toPromote = true;
1788   } else {
1789     if ((track->pUsedHistoryLength < track->ConfirmationParameters[1]) ||

```

Decisions analyzed:

(track->pUsedHistoryLength < track->ConfirmationParameters[1]) rtIsNaN(track->ConfirmationParameters[1])	50%
false	13038/13038
true	0/13038

Conditions analyzed:

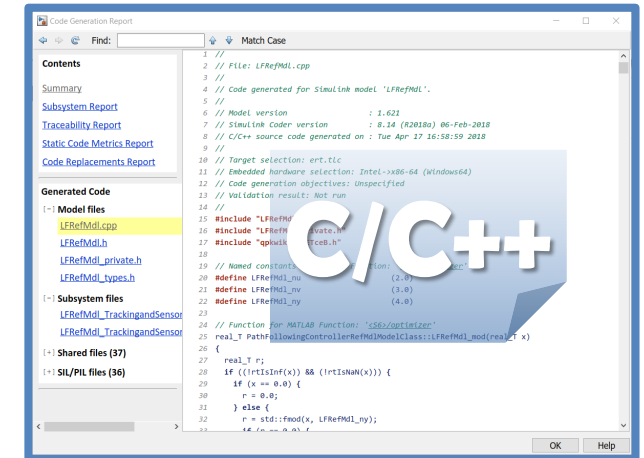
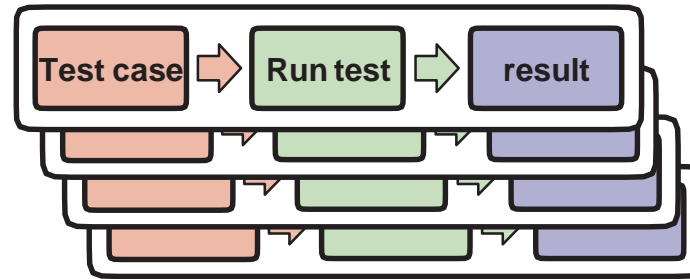
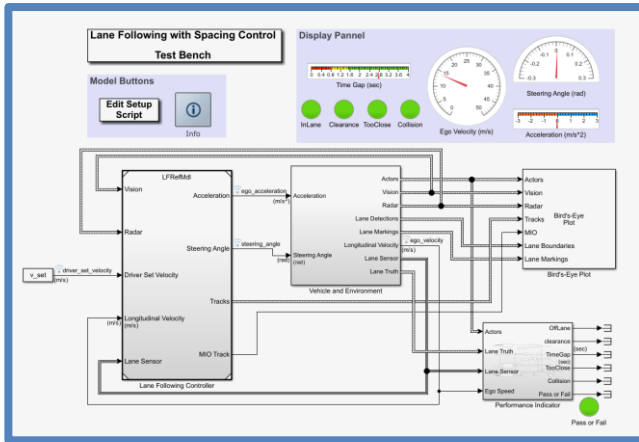
Description:	True	False
track->pUsedHistoryLength < track->ConfirmationParameters[1]	0	13038
rtIsNaN(track->ConfirmationParameters[1])	0	13038

MC/DC analysis (combinations in parentheses did not occur)

decision outcomes:	True Out	False Out
Conditions:		
track->pUsedHistoryLength < track->ConfirmationParameters[1]	(Tx)	FF
rtIsNaN(track->ConfirmationParameters[1])	(FT)	FF

Design and Test Traffic Jam Assist

A Case study Using • Automated Driving System Toolbox™ • MPC Toolbox™ • VDBS™
 • Simulink™ • Simulink Test™ • Simulink Control Design™ • Embedded Coder™



Design ACC and Lane Following Controller

- Create driving scenario
- Synthesize sensor detection
- Include Vehicle Dynamics
- Design sensor fusion algorithm
- Design controller using MPC

Automate Regression Test

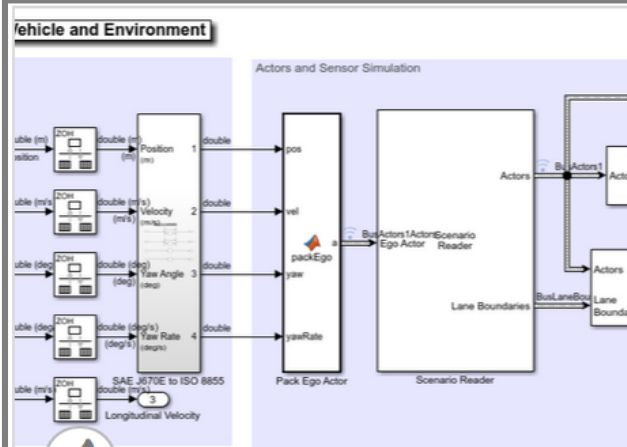
- Define performance evaluation metrics
- Develop test cases
- Build test suites
- Verification and validation

Generate and Verify Code

- SIL test
- Code generation
- Coverage test

Learn more about Traffic Jam Assist (Lane Following Control) by exploring examples in R2018b

Automated Driving System Toolbox™



Vehicle and Environment

Actors and Sensor Simulation

This example uses:
 Automated Driving System Toolbox
 Embedded Coder
 Model Predictive Control Toolbox
 Simulink Control Design
 Simulink

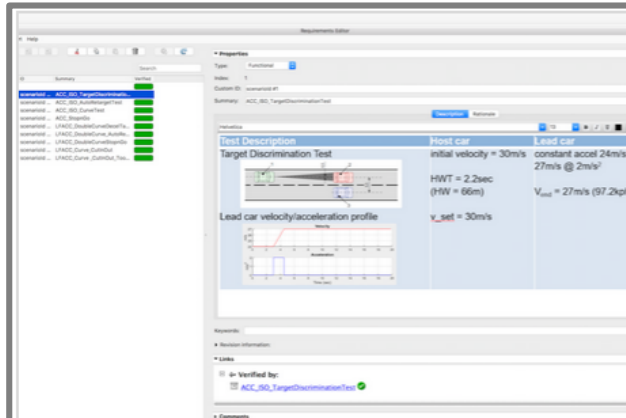
Open Model

Lane Following Control with Sensor Fusion and Lane Detection

Simulate and generate code for an automotive lane-following controller.

Open Model

Simulink Test™



Target Discrimination Test

Host car	Lead car
initial velocity = 30m/s	constant accel 24m/s ²
HWT = 2.2sec (HW = 66m)	27m/s @ 2m/s ²
v_set = 30m/s	V_max = 27m/s (97.2kph)

Open Script

Testing a Lane-Following Controller with Simulink Test

Perform requirements-based testing for an automotive lane-following system.

Open Script

Thank you for your attention !!

Email: seo-wook.park@mathworks.com