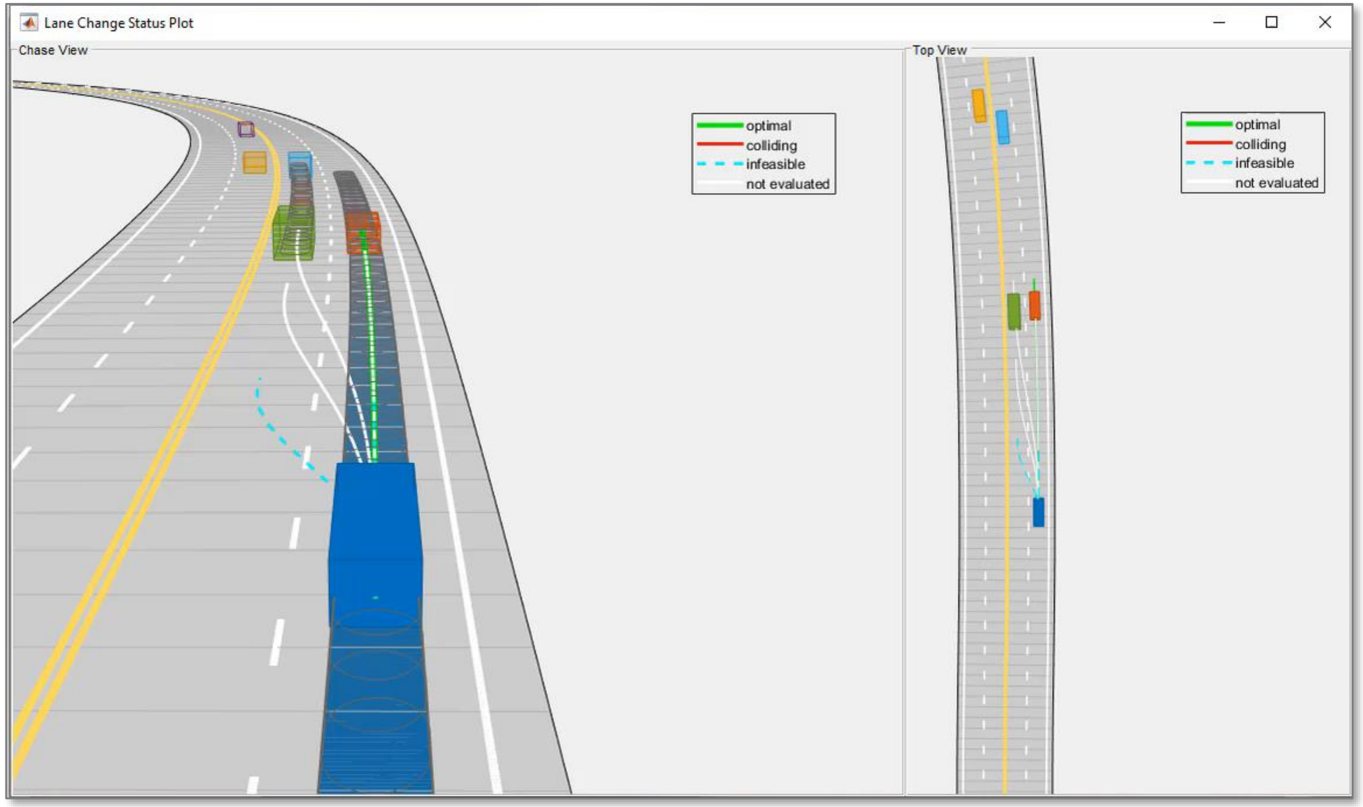
MATLAB EXPO 2021

Developing a Motion Planner for Highway Lane Change Maneuvers

Seo-Wook Park

MathWorks

Motion planner for Highway Lane Change Maneuver

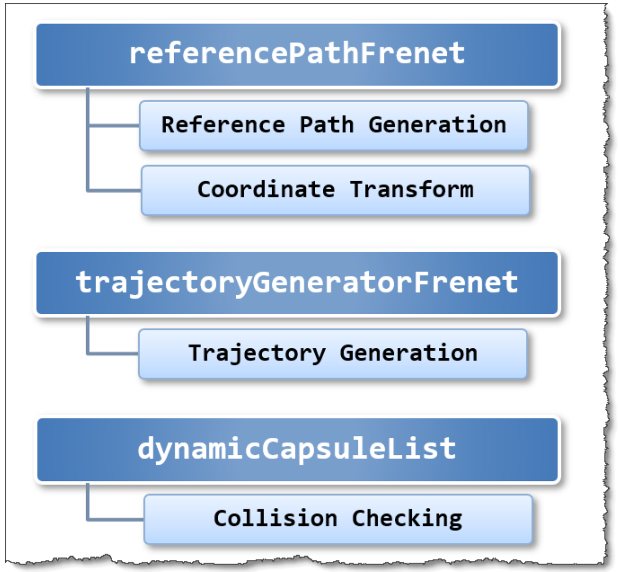


- Automated lane change maneuver (LCM) system for highway driving scenario
- Generates an optimal trajectory in Frenet space
- Implement driving maneuver behavior depending on surrounding traffic conditions
- Collision checking using dynamic capsule-based objects

Automated Driving Toolbox™
Navigation Toolbox™
Model Predictive Control Toolbox™

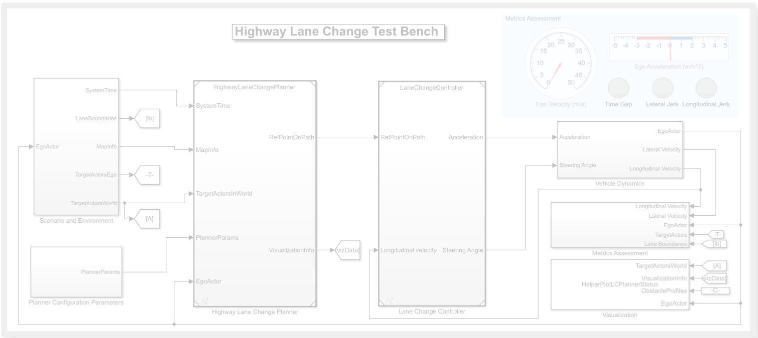
R2021a

Motion planner for Highway Lane Change Maneuver



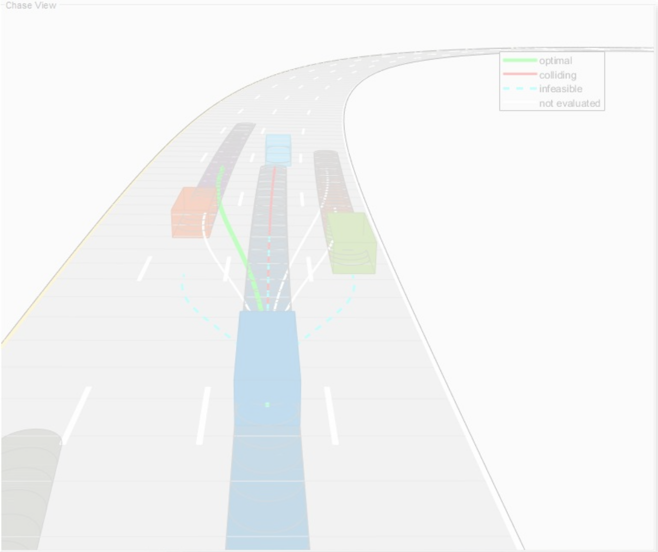
Learn fundamentals

- Motivation
- Schematic of Motion Planner
- Three MATLAB functions
  - ✓ `referencePathFrenet`
  - ✓ `trajectoryGeneratorFrenet`
  - ✓ `dynamicCapsuleList`



Implement motion planner and controller

- Learn through reference example
- Architecture of highway lane change planer
- Closed-loop controller

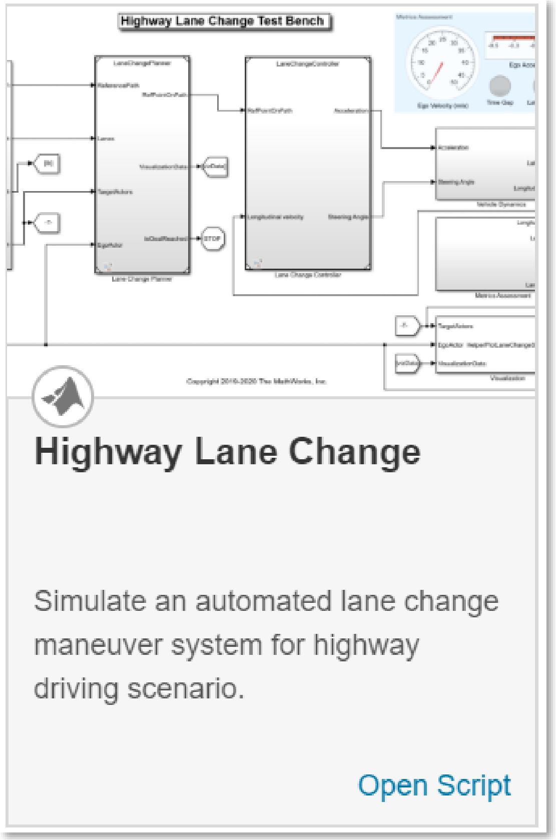


Simulate test bench

- Scenarios in straight and curved roads
- Scenarios imported from HERE HD map

Lane Change Maneuver (LCM) system – example

R2020a



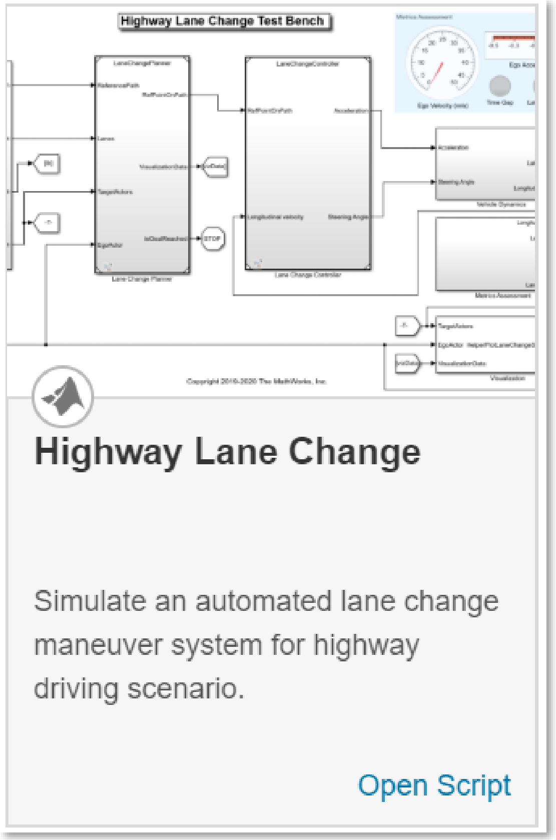
Initial version of LCM

- `trajectoryOptimalFrenet`
  - Reference Path Generation
  - Coordinate Transform
  - Trajectory Generation
  - Feasibility Checking
  - Cost Evaluation
  - Collision Checking

- Feedback
  - Lack of configurability and flexibility
  - How to customize?

Modular architecture to enable customized design

R2020a



Initial version of LCM

- **trajectoryOptimalFrenet**
  - Reference Path Generation
  - Coordinate Transform
  - Trajectory Generation
  - Feasibility Checking
  - Cost Evaluation
  - Collision Checking

- Feedback
  - Lack of configurability and flexibility
  - How to customize?

R2020b

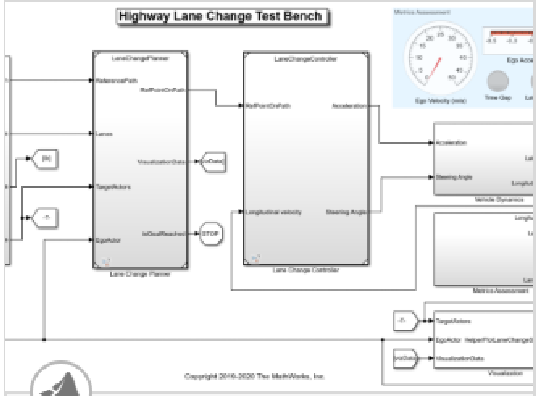
- **referencePathFrenet**
  - Reference Path Generation
  - Coordinate Transform
- **trajectoryGeneratorFrenet**
  - Trajectory Generation
- **dynamicCapsuleList**
  - Collision Checking
- Feasibility Checking
- Cost Evaluation

- Modular architecture

(Grey box means purely custom implementation)

Modular architecture to enable customized design

R2020a



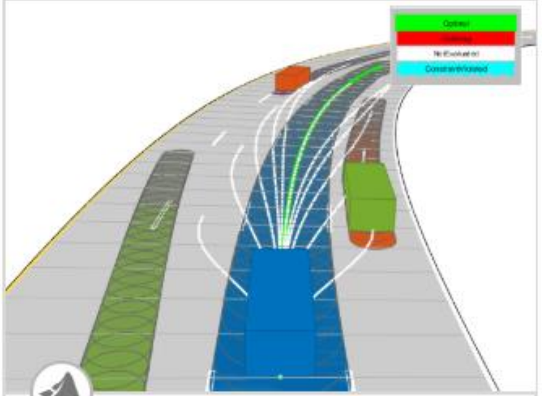
Highway Lane Change

Simulate an automated lane change maneuver system for highway driving scenario.

Open Script

Initial version of LCM

R2020b



Highway Trajectory Planning Using Frenet Reference Path

Demonstrates how to plan a local trajectory in a highway driving scenario. This example uses a reference path and dynamic list of

Open Live Script

Major upgrade for planner (MATLAB version)

R2020b

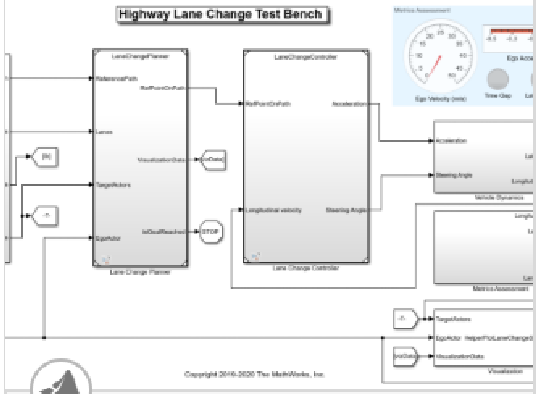
- `referencePathFrenet`
  - Reference Path Generation
  - Coordinate Transform
- `trajectoryGeneratorFrenet`
  - Trajectory Generation
- `dynamicCapsuleList`
  - Collision Checking
- Feasibility Checking
- Cost Evaluation

- Modular architecture

(Grey box means purely custom implementation)

Evolution of examples for Lane Change Maneuver (LCM) system

R2020a



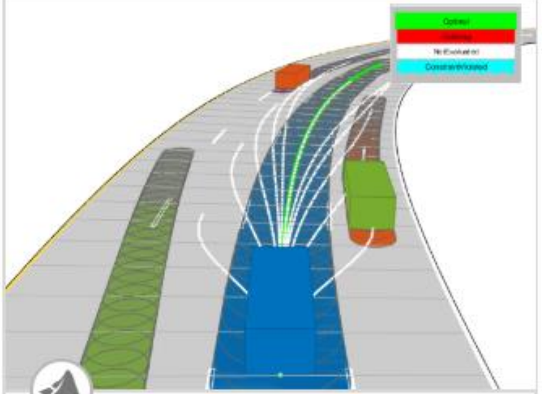
Highway Lane Change

Simulate an automated lane change maneuver system for highway driving scenario.

Open Script

Initial version of LCM

R2020b



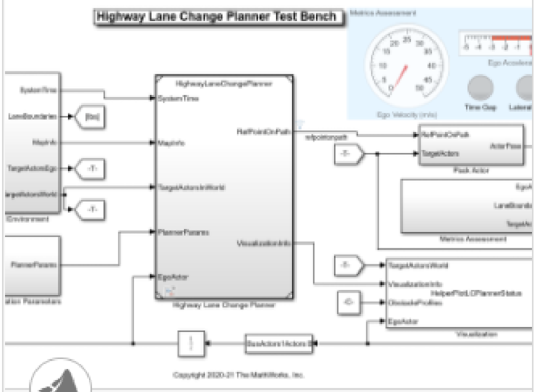
Highway Trajectory Planning Using Frenet Reference Path

Demonstrates how to plan a local trajectory in a highway driving scenario. This example uses a reference path and dynamic list of

Open Live Script

Major upgrade for planner (MATLAB version)

R2021a

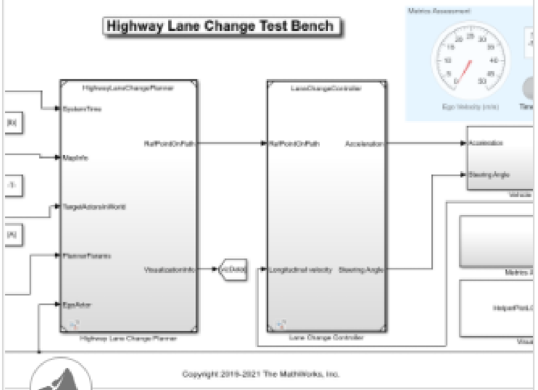


Generate Code for Highway Lane Change Planner

Design, test, and generate C++ code for a lane change planner for highway driving. This example closely follows the Highway

Motion planner (Simulink Model)

R2021a



Highway Lane Change

Design and test the planner and controller components of a lane change maneuver system designed for highway driving.

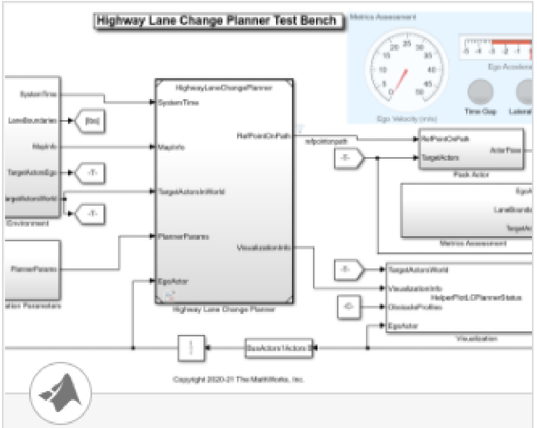
Closed-loop system (Simulink Model)

Modular architecture

Incrementally learn about lane change planning and controls

- **referencePathFrenet**
  - Reference Path Generation
  - Coordinate Transform
- **trajectoryGeneratorFrenet**
  - Trajectory Generation
- **dynamicCapsuleList**
  - Collision Checking
- Feasibility Checking
- Cost Evaluation

R2021a

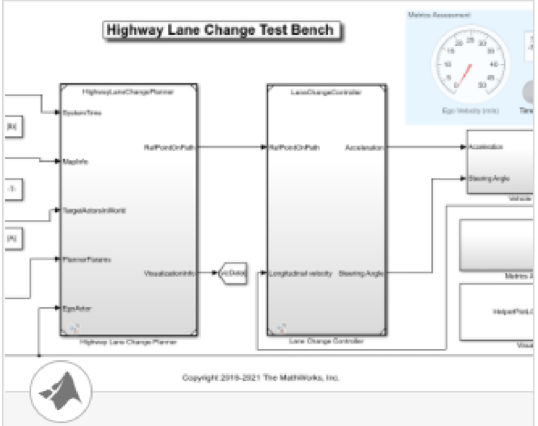


Generate Code for Highway Lane Change Planner

Design, test, and generate C++ code for a lane change planner for highway driving. This example closely follows the Highway

Motion planner
(Simulink Model)

R2021a

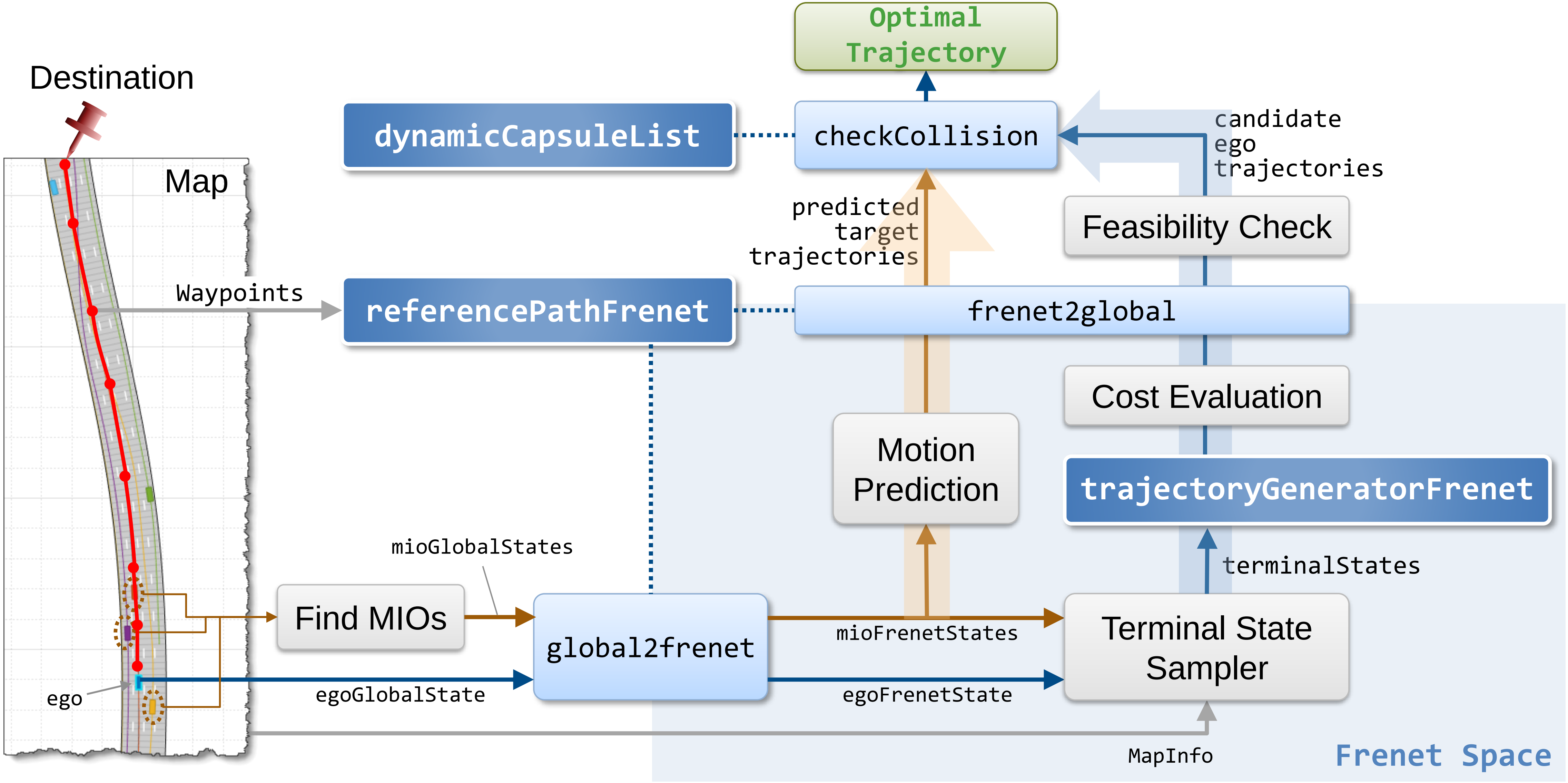


Highway Lane Change

Design and test the planner and controller components of a lane change maneuver system designed for highway driving.

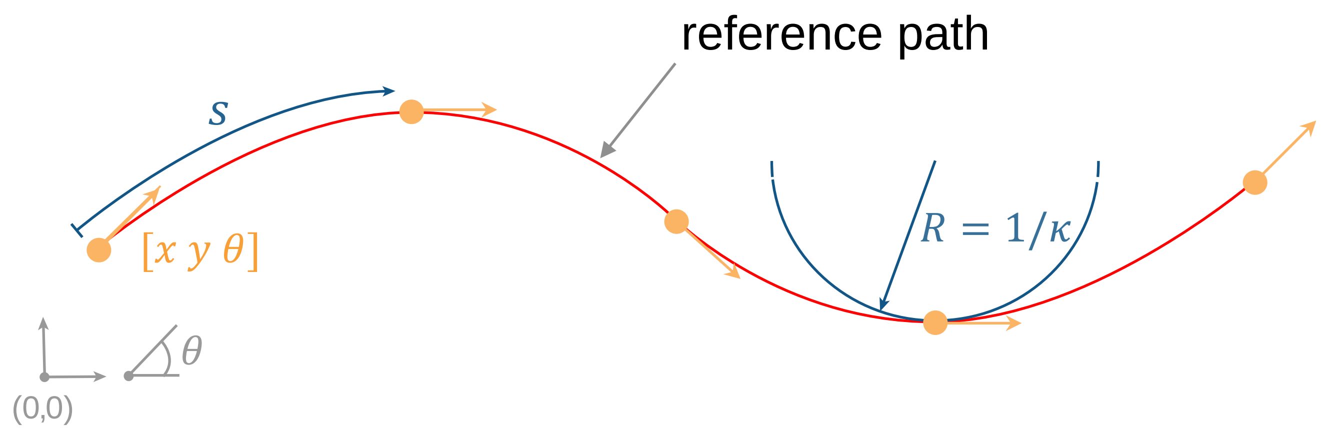
Closed-loop system
(Simulink Model)

Schematic of Motion Planner for Lane Change Maneuver



Advantage of trajectory planning in Frenet coordinate

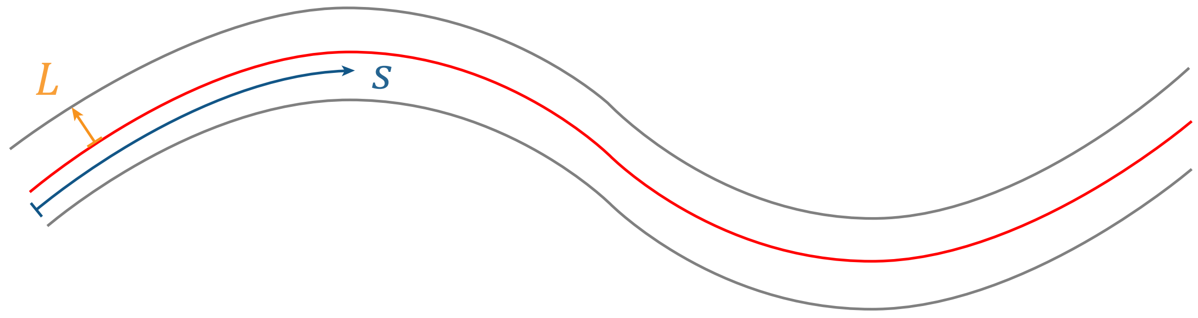
- Frenet system represents an object and its trajectory with respect to the reference path (road center or lane center).
- This approach dramatically simplifies the trajectory planning task when a car is traveling on a curved road.



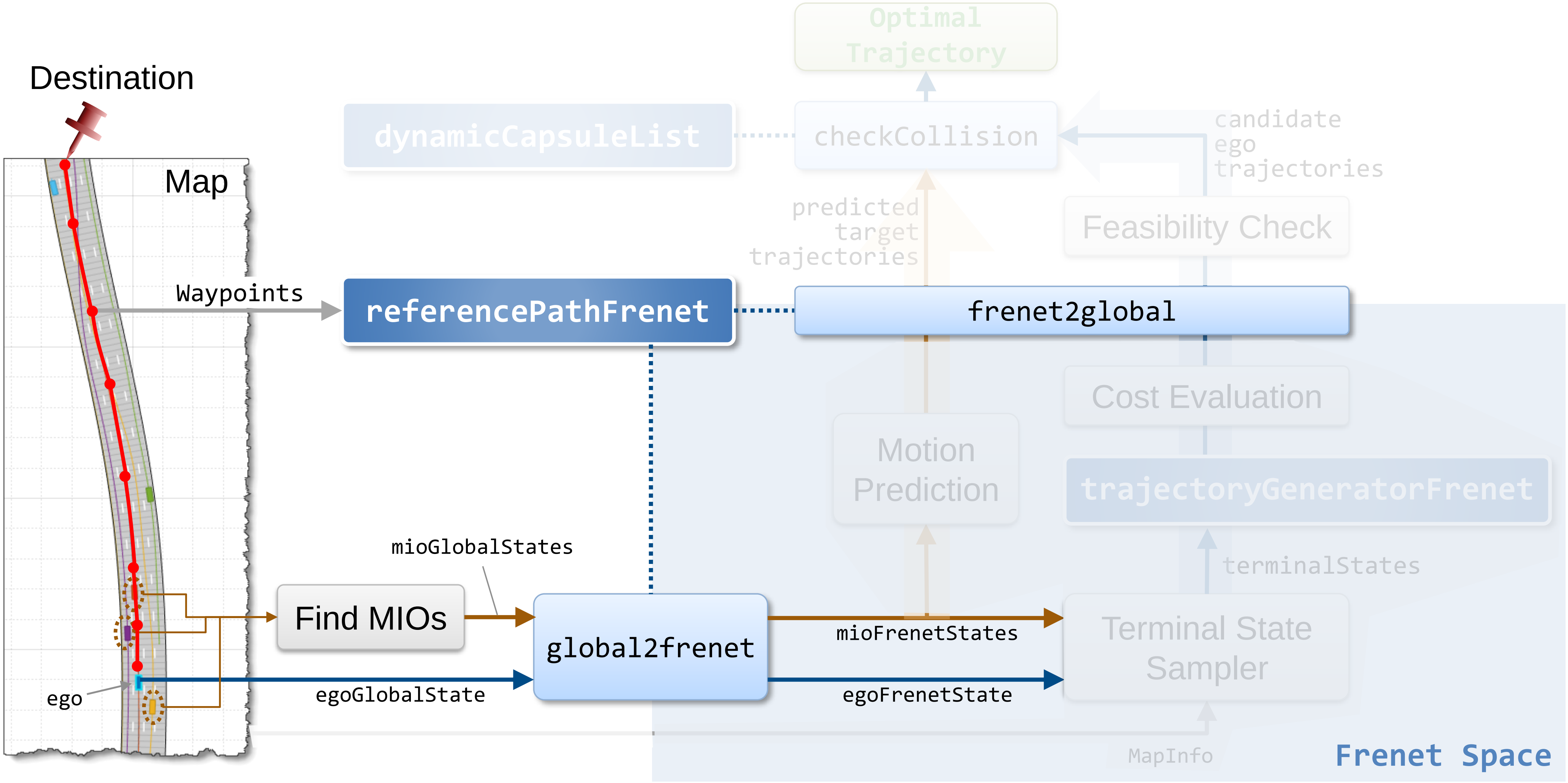
Global States: $[x\ y\ \theta\ \kappa\ v\ a]$

Frenet States: $[s\ \delta s\ \delta^2 s\ L\ \delta L\ \delta^2 L]$

- s : Arc length
- L : Perpendicular deviation from the direction of the reference path



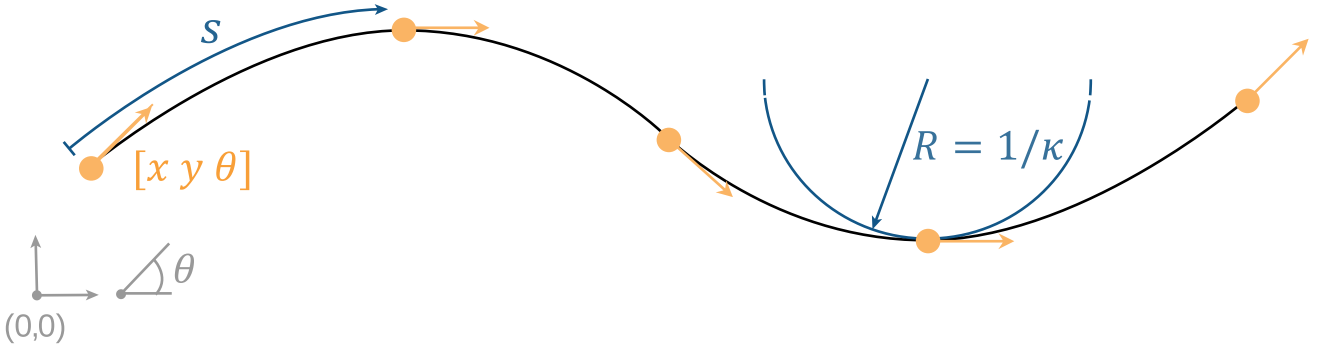
Schematic of Motion Planner for Lane Change Maneuver



referencePathFrenet

`referencePathFrenet`

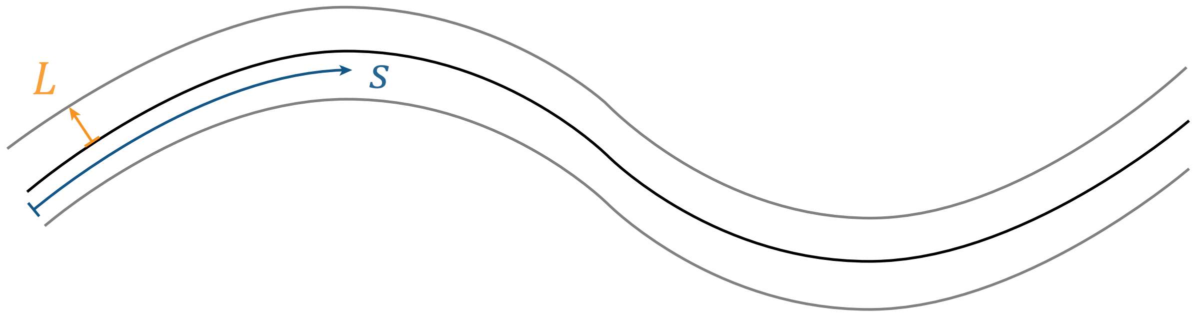
- Fits a smooth, piecewise, continuous curve to a set of waypoints given as $[x\ y]$ or $[x\ y\ \theta]$



Path Points: $[x\ y\ \theta\ \kappa\ \delta\kappa\ s]$

- κ : Curvature
- $\delta\kappa$: Derivative of curvature
- s : Arc length

- Convert trajectories between global and Frenet coordinate systems
`(frenet2global, global2frenet)`

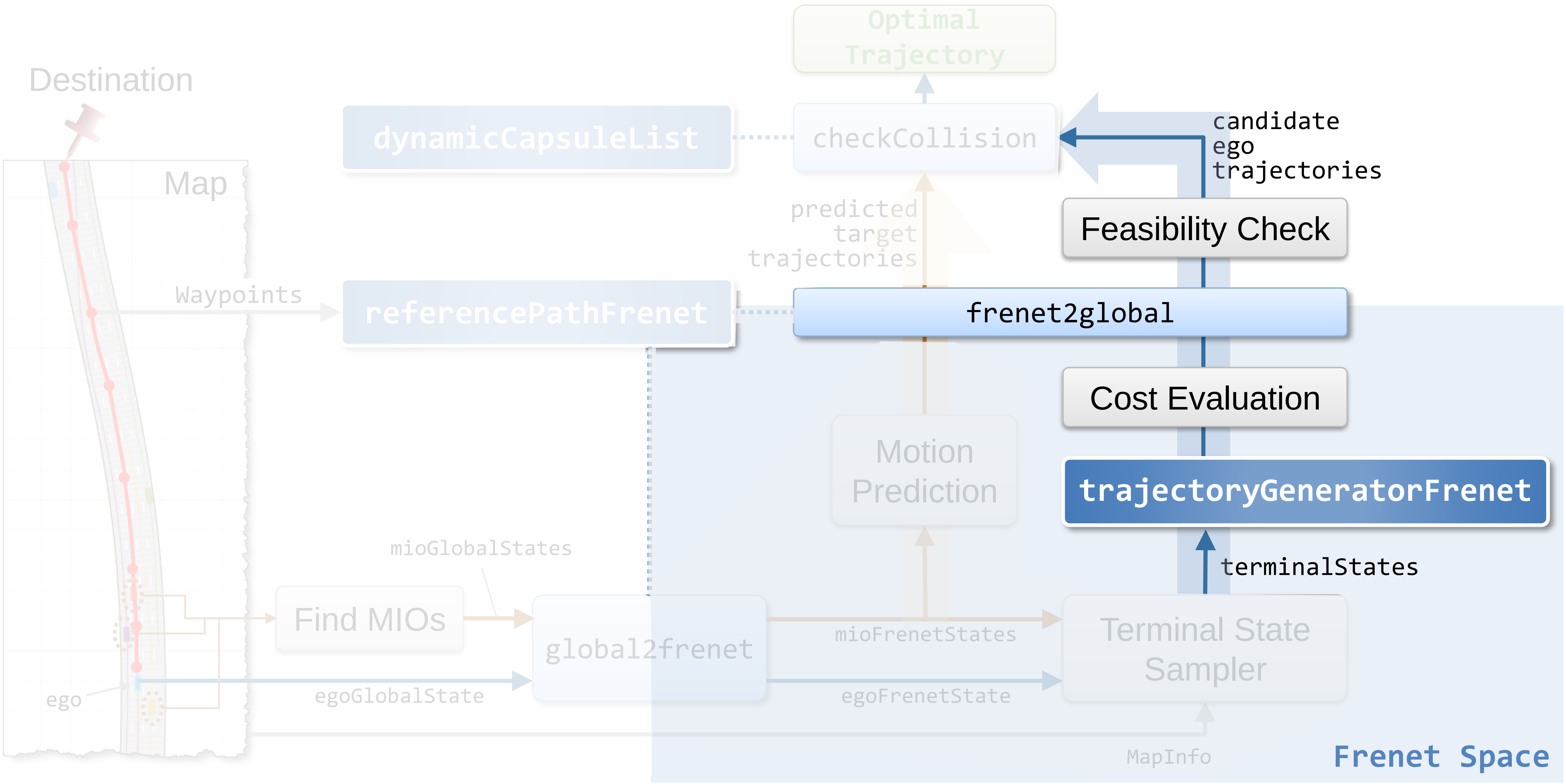


Global States: $[x\ y\ \theta\ \kappa\ v\ a]$

Frenet States: $[s\ \delta s\ \delta^2 s\ L\ \delta L\ \delta^2 L]$

- s : Arc length
- L : Perpendicular deviation from the direction of the reference path

Schematic of Motion Planner for Lane Change Maneuver

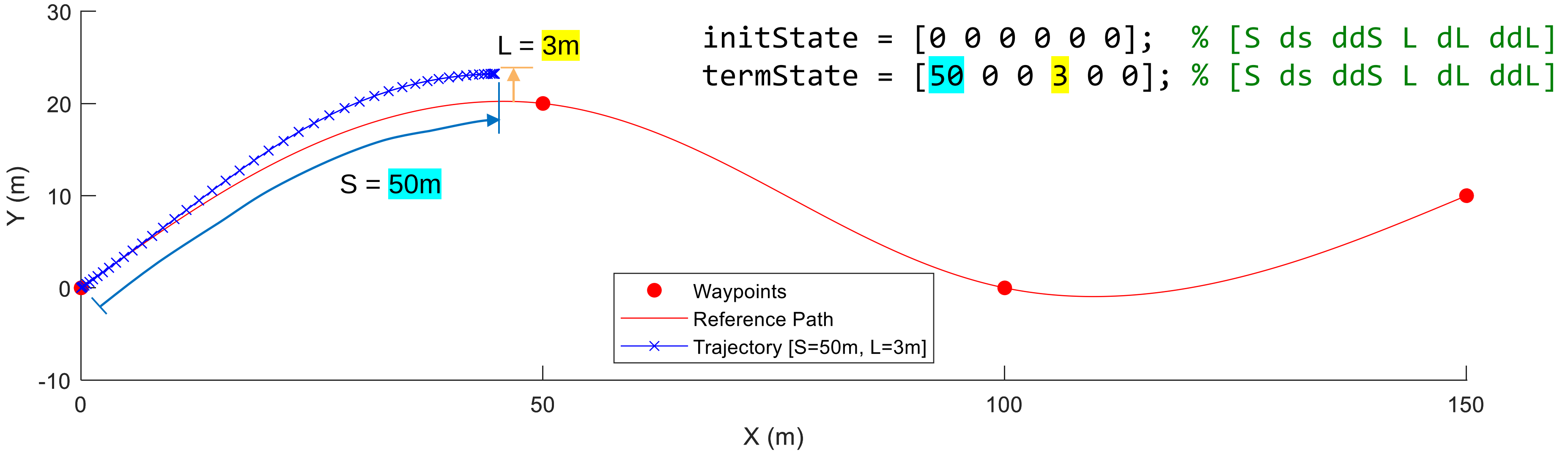


trajectoryGeneratorFrenet

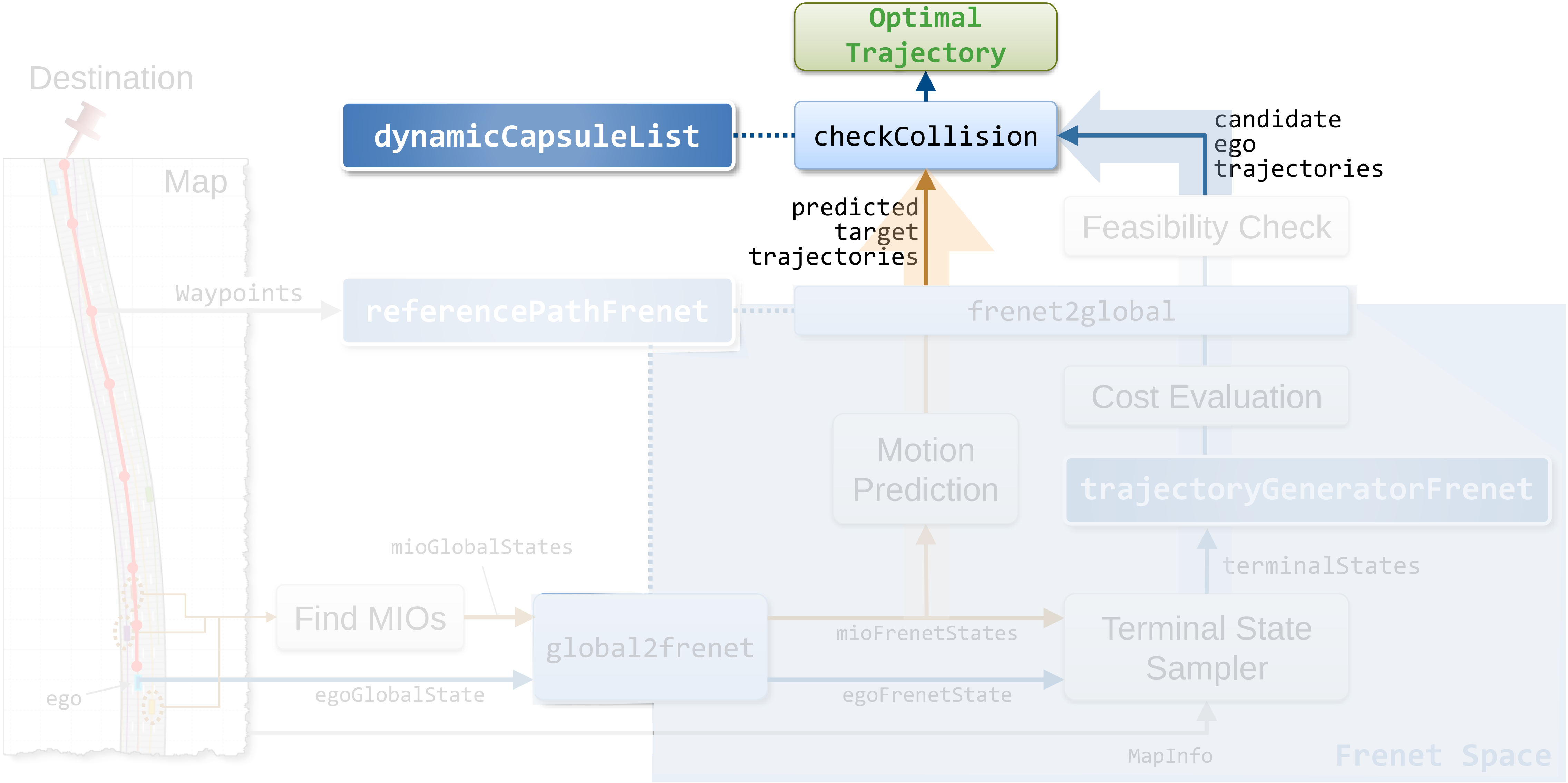
trajectoryGeneratorFrenet

- Generates trajectory by solving 4th or 5th order polynomials that satisfy boundary conditions in Frenet space relative to a given reference path

```
refPath = referencePathFrenet(waypoints);
connector = trajectoryGeneratorFrenet(refPath);

[frenetTraj,globalTraj] = ...
    connect(connector,initState,termState,timeSpan);
```



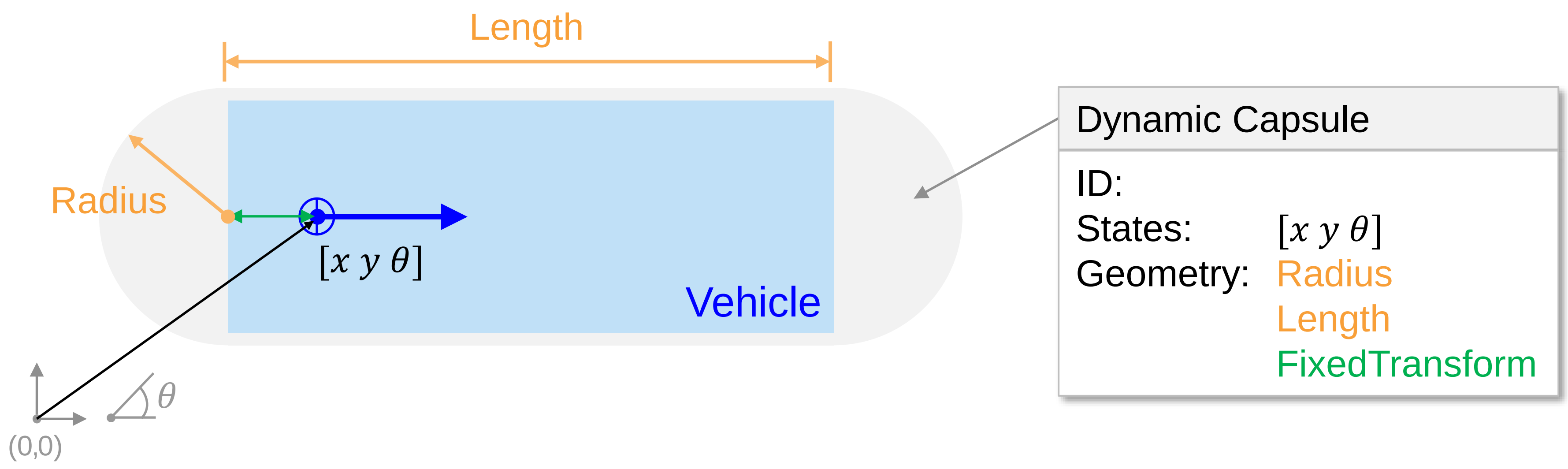
Schematic of Motion Planner for Lane Change Maneuver



dynamicCapsuleList

dynamicCapsuleList

- Manages lists of capsule-based objects for collision checking



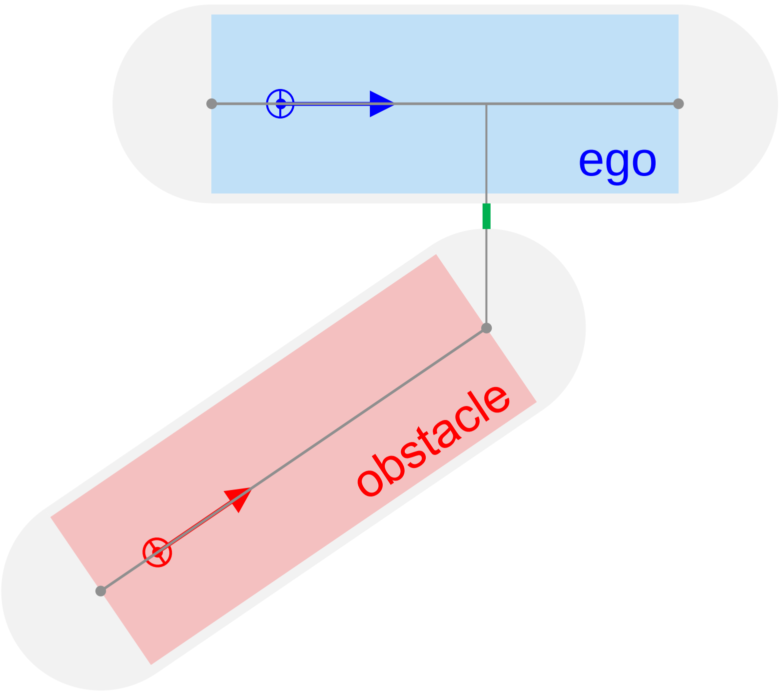
- Why Capsules?
  - Automatically build a buffer on front/rear of rectangular objects
  - Avoid inflating the sides of rectangular objects
  - Computationally efficient for collision checking

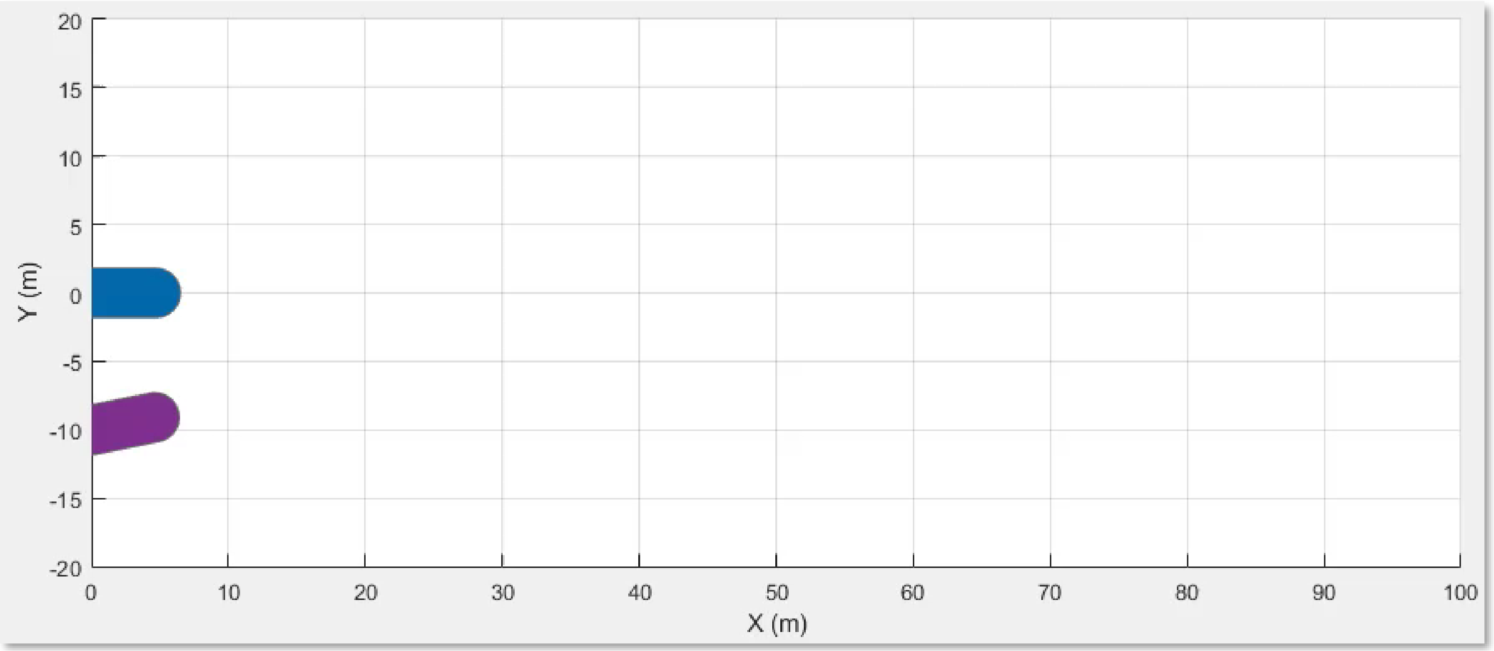
dynamicCapsuleList

dynamicCapsuleList

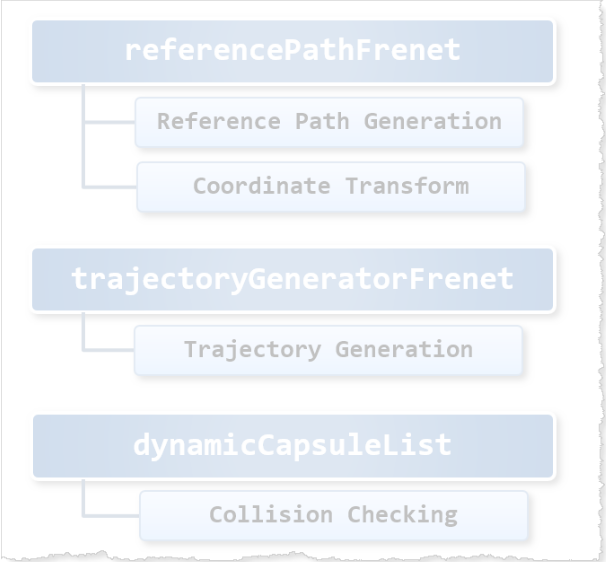
- Check for collisions between ego and obstacles using dynamicCapsuleList

```
isColliding = checkCollision(capList)
```



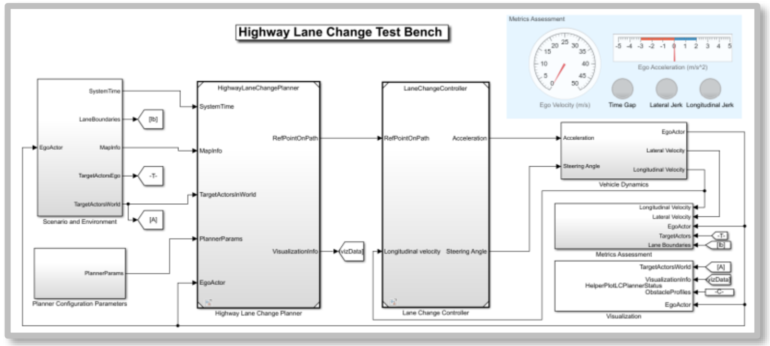


Motion planner for Highway Lane Change Maneuver



Learn fundamentals

- Motivation
- Schematic of Motion Planner
- Three MATLAB functions
  - ✓ `referencePathFrenet`
  - ✓ `trajectoryGeneratorFrenet`
  - ✓ `dynamicCapsuleList`



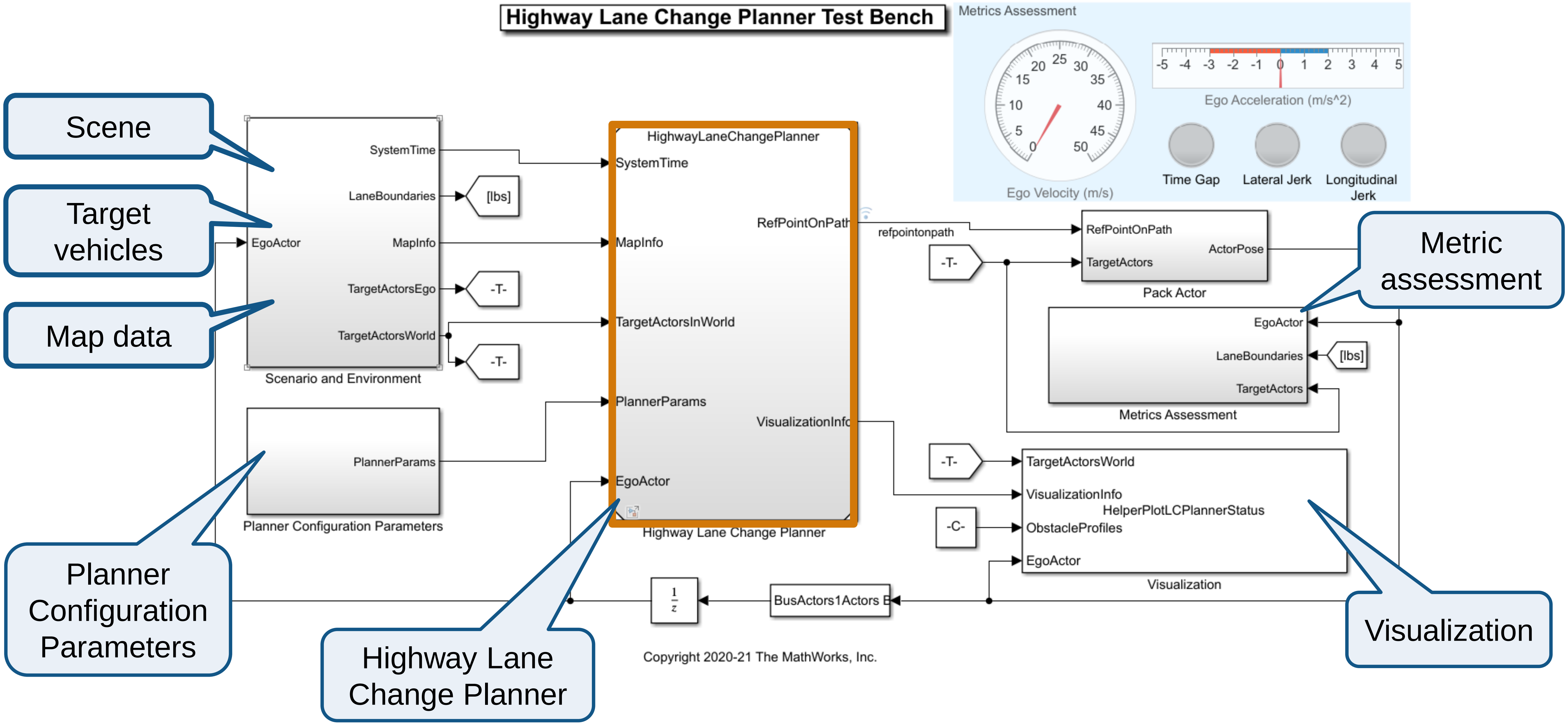
Implement motion planner and controller

- Learn through reference example
- Architecture of highway lane change planer
- Closed-loop controller

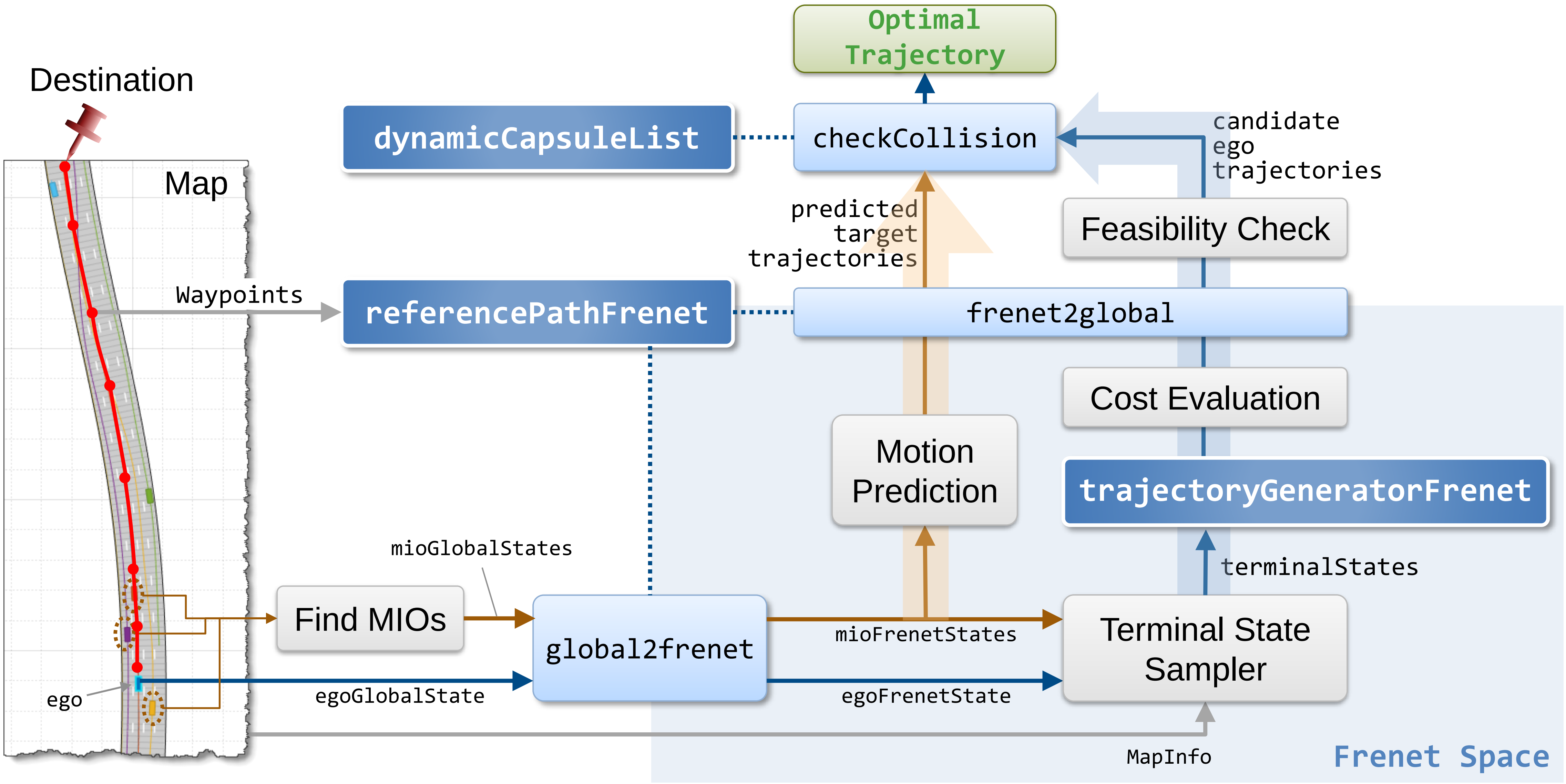
Simulate test bench

- Scenarios in straight and curved roads
- Scenarios imported from HERE HD map

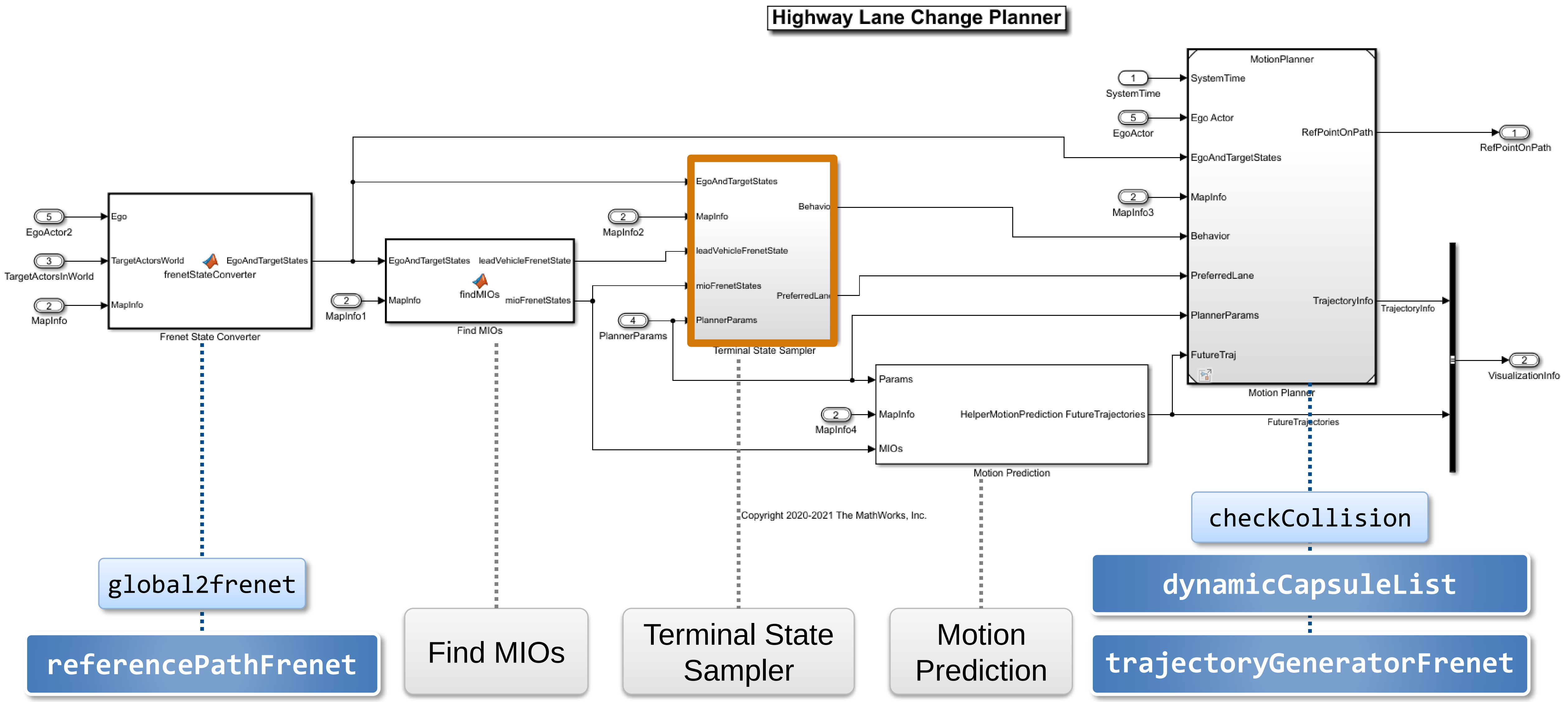
Highway lane change planner test bench



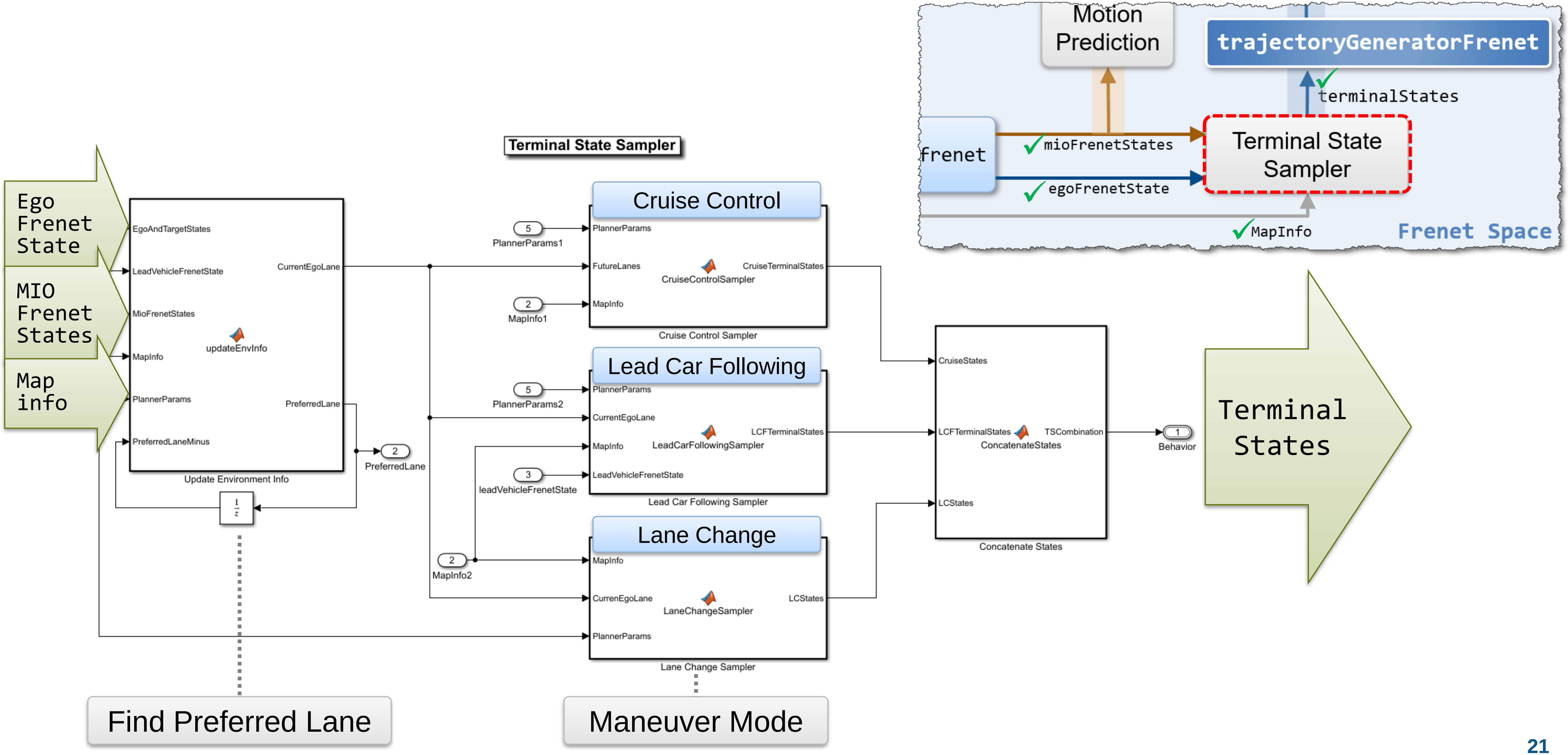
Schematic of Motion Planner for Lane Change Maneuver



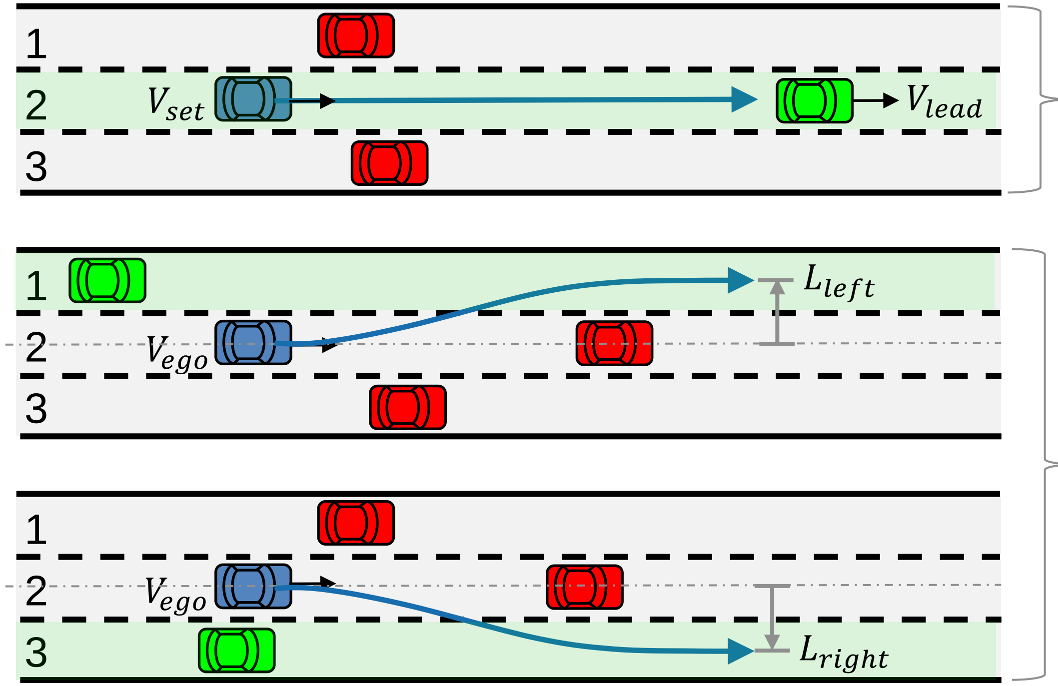
Highway Lane Change Planner



Highway Lane Change Planner : Terminal State Sampler



Preferred lane and maneuver mode



ego

unsafe target $(TTC_{ego-mio} < TTC_{safe})$

safe target $(TTC_{ego-mio} \geq TTC_{safe})$

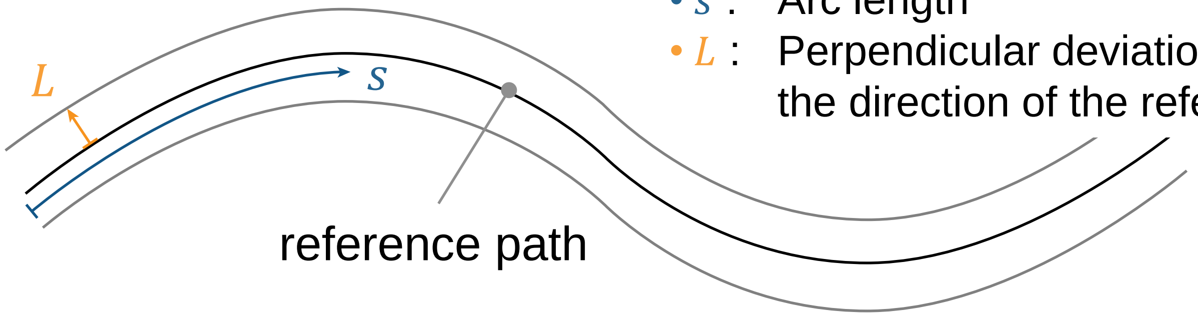
where TTC = Time-to-collision

Terminal Frenet States	s	δs	$\delta^2 s$	L	δL	$\delta^2 L$
Cruise Control Mode	NaN	V_{set}	0	L_{ego}	0	0
Lead Car Following Mode	NaN	V_{lead}	0	L_{ego}	0	0
Lane Change Mode	NaN	V_{ego}	0	L_{left}	0	0
	NaN	V_{ego}	0	L_{right}	0	0

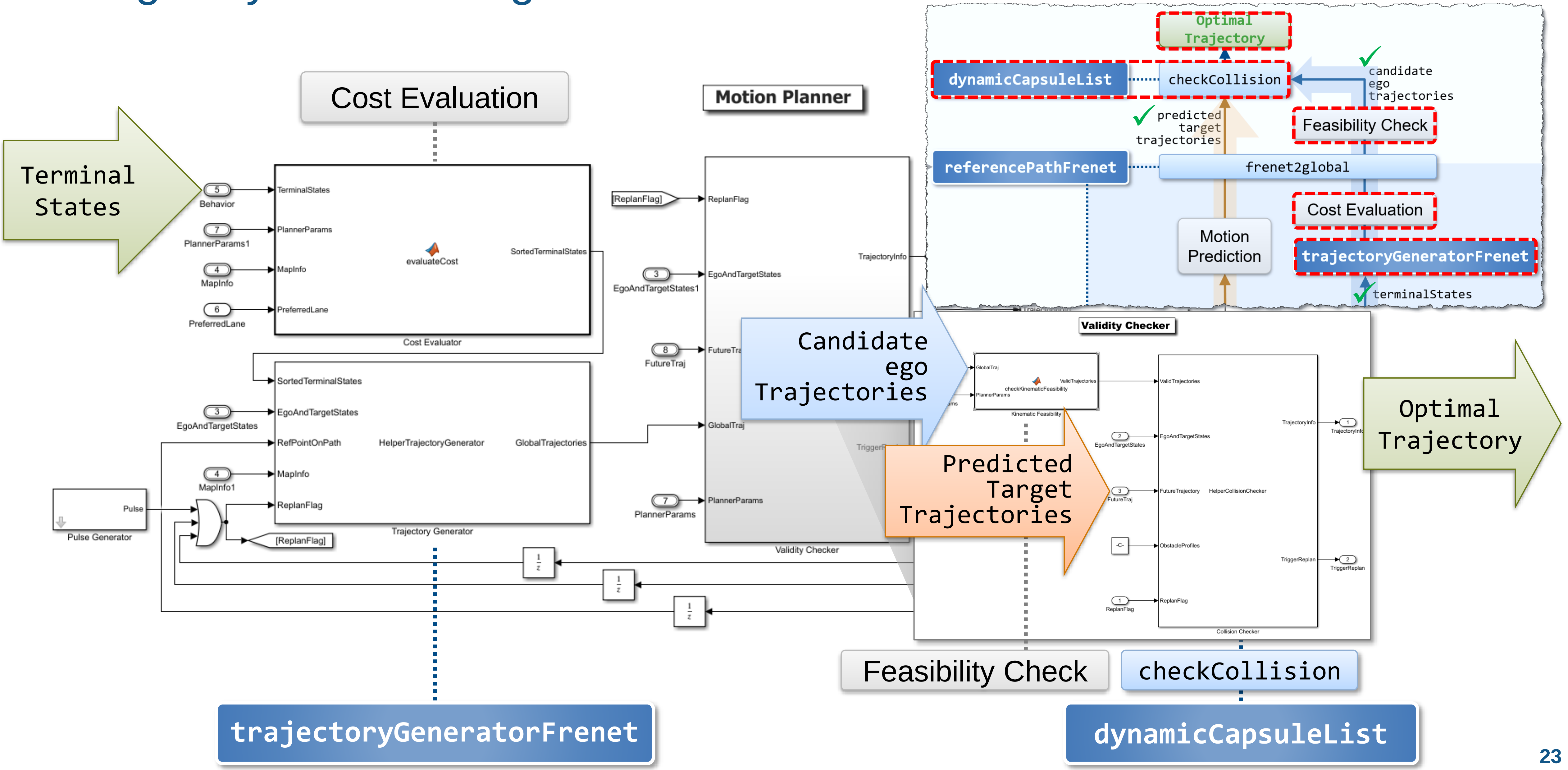
Unrestricted longitudinal position
→ use 4th-order polynomial

Frenet States: $[s\ \delta s\ \delta^2 s\ L\ \delta L\ \delta^2 L]$

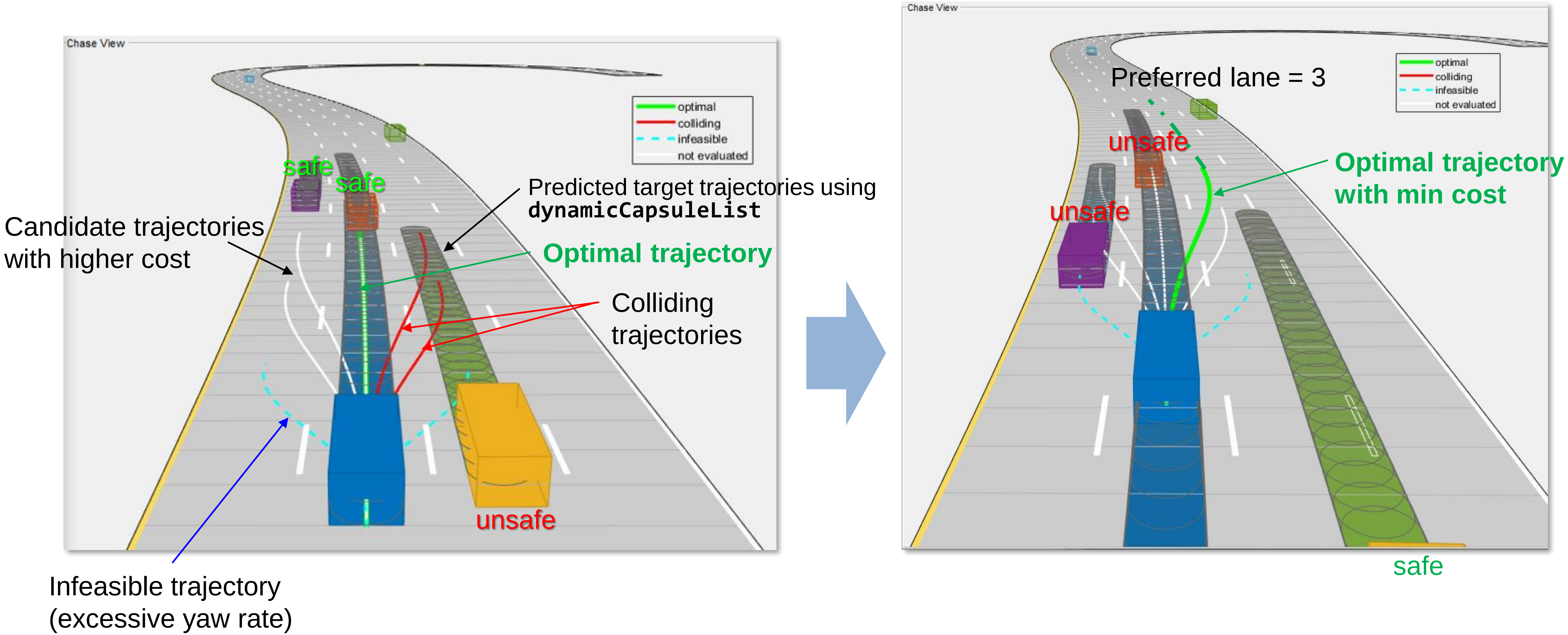
- s : Arc length
- L : Perpendicular deviation from the direction of the reference path



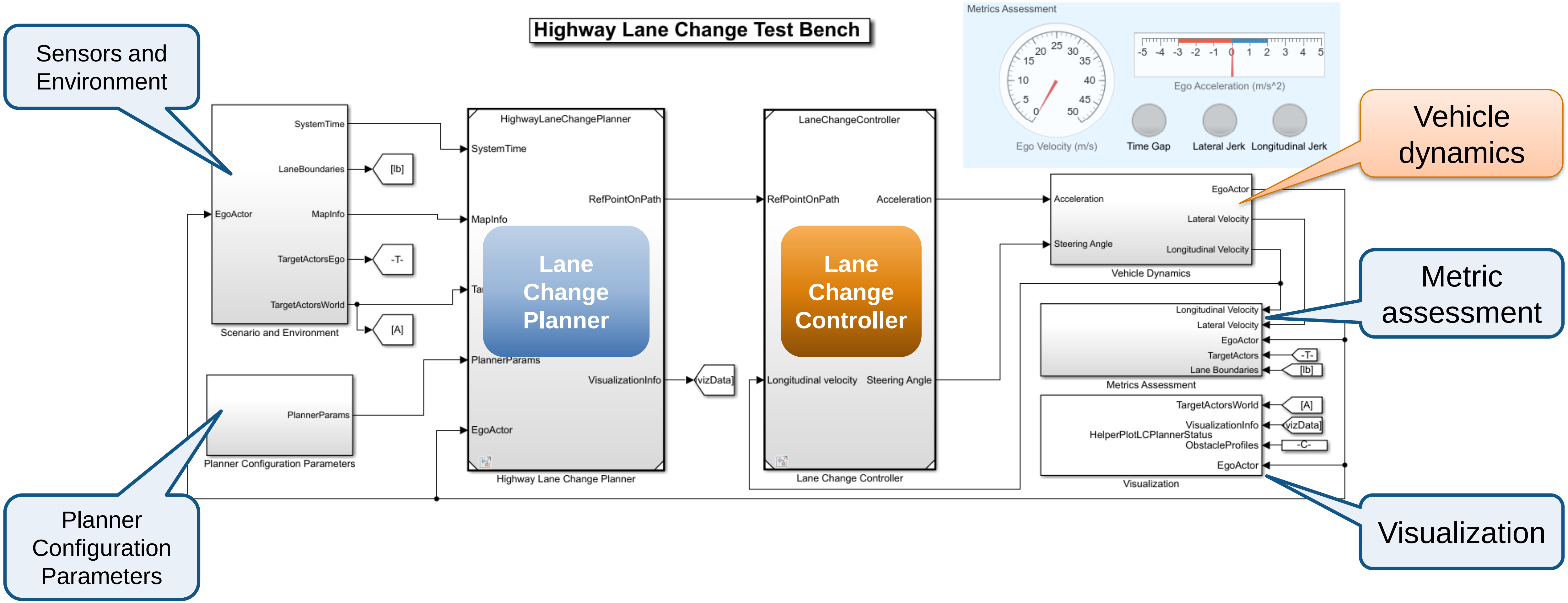
Highway Lane Change Planner : Motion Planner



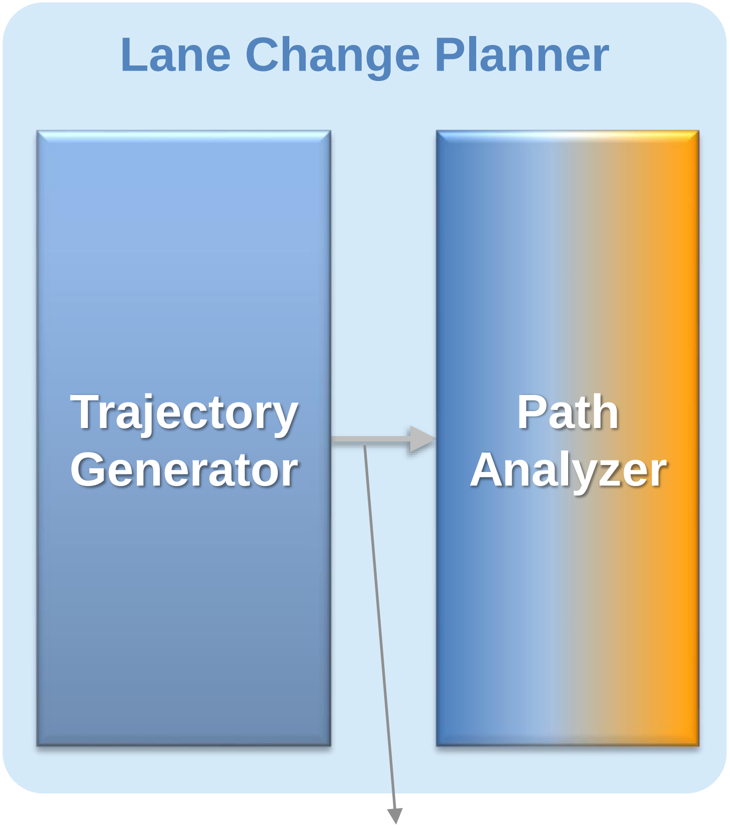
Examples for finding optimal trajectory



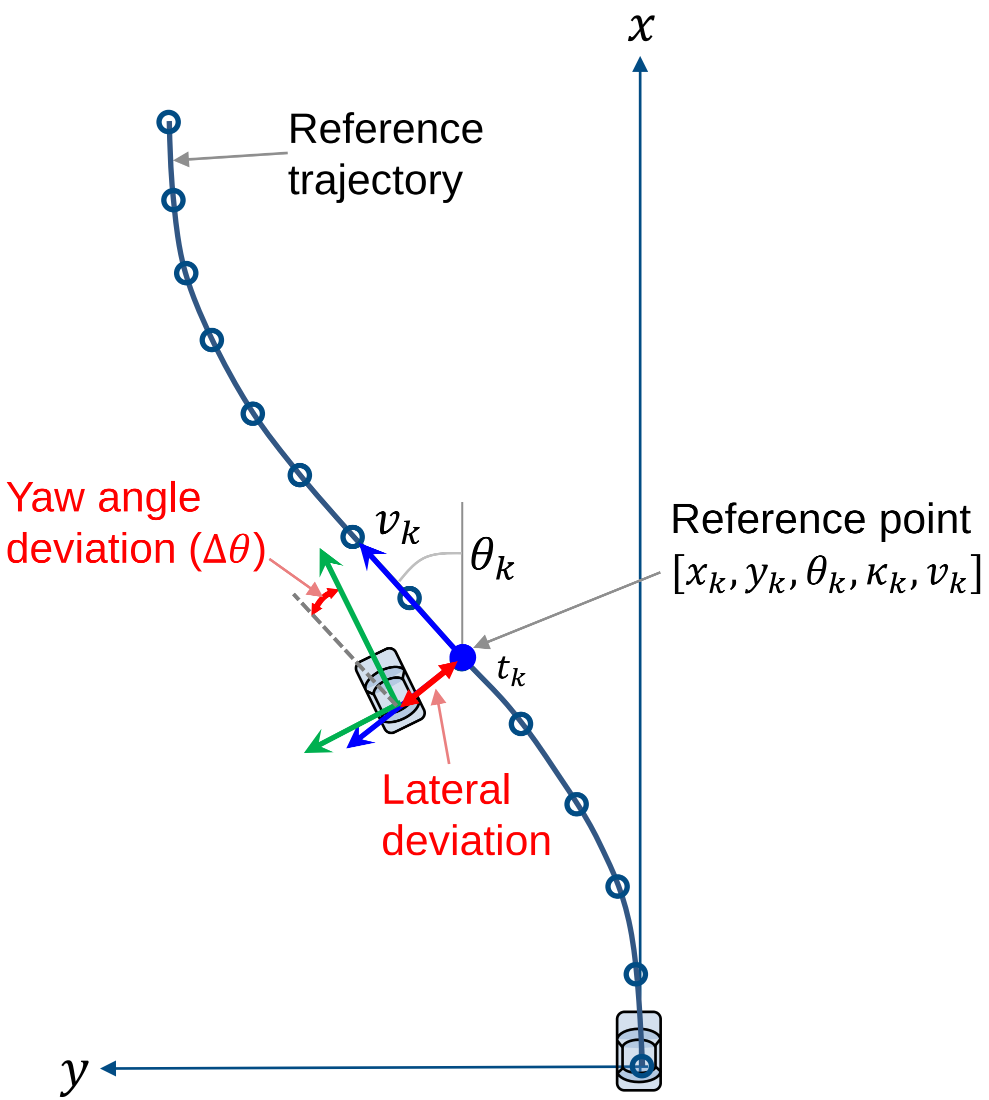
Highway lane change : closed-loop system



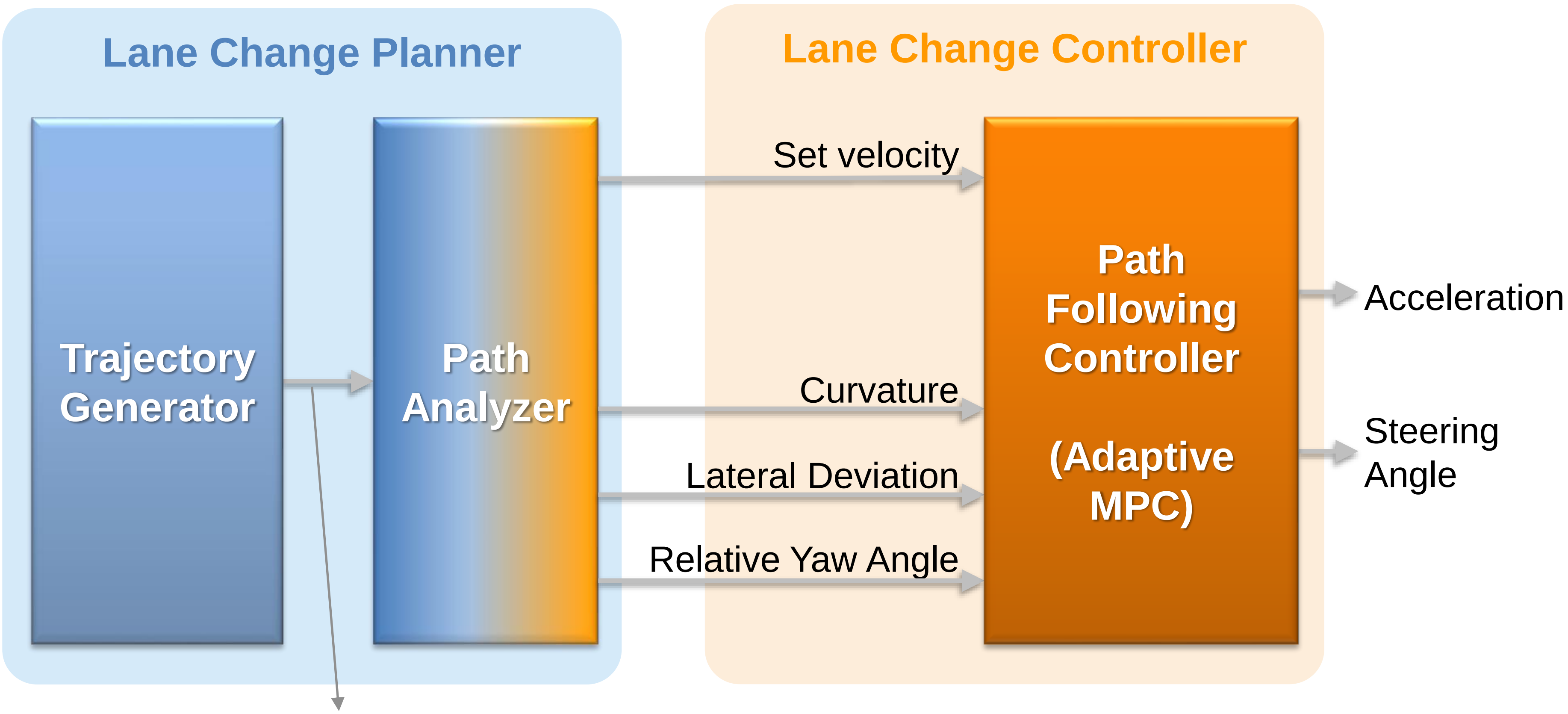
Lane Change Planner



Optimal trajectory
→ Reference trajectory
(virtual lane center)

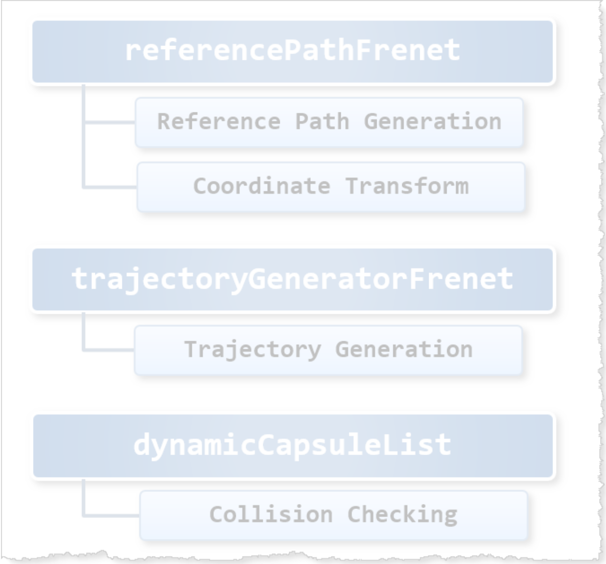


Lane Change Planner + Controller



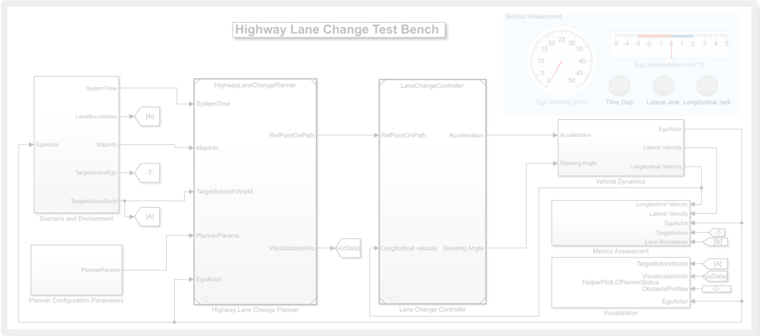
Optimal trajectory
→ Reference trajectory
(virtual lane center)

Motion planner for Highway Lane Change Maneuver



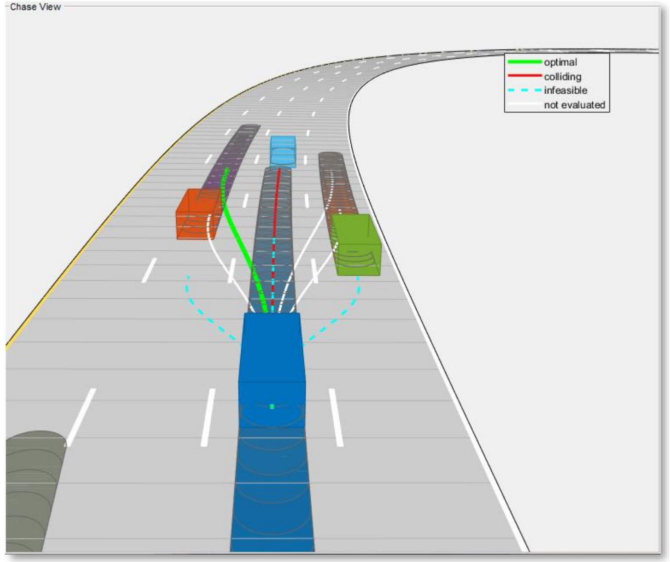
Learn fundamentals

- Motivation
- Schematic of Motion Planner
- Three MATLAB functions
  - ✓ `referencePathFrenet`
  - ✓ `trajectoryGeneratorFrenet`
  - ✓ `dynamicCapsuleList`



Implement motion planner and controller

- Learn through reference example
- Architecture of highway lane change planer
- Closed-loop controller

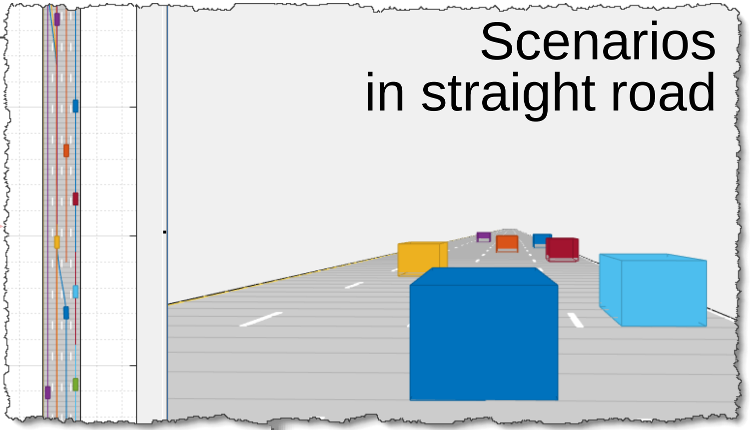


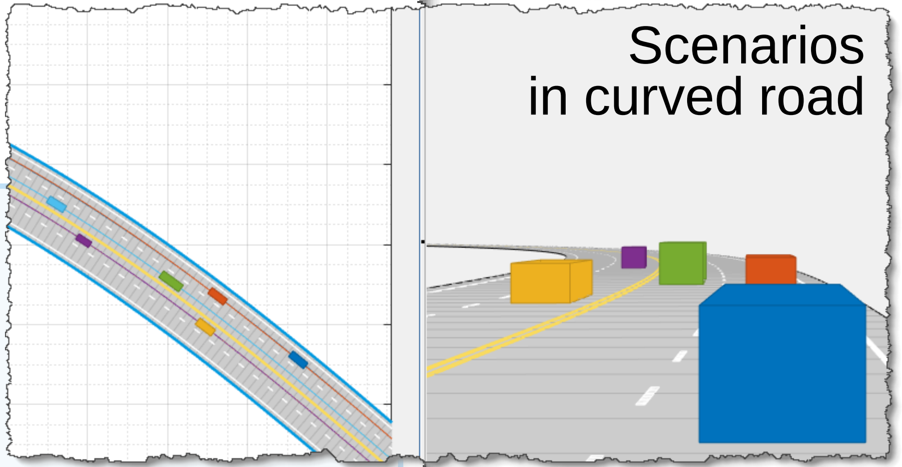
Simulate test bench

- Scenarios in straight and curved roads
- Scenarios imported from HD map

Simulate highway lane change planner with test scenarios

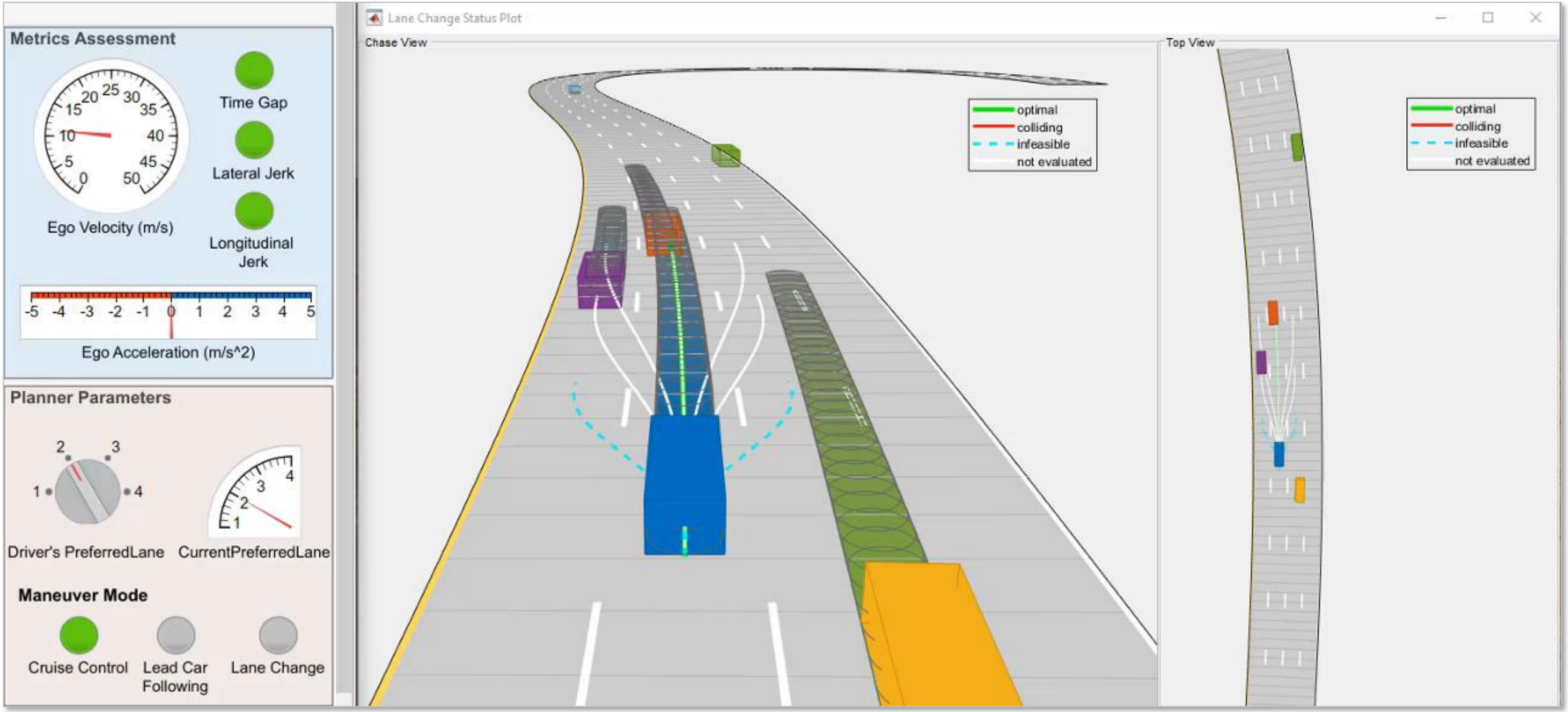
```
function helperSLHighwayLaneChangeSetup(nvp)

arguments
    nvp.scenarioFcnName {mustBeMember(nvp.scenarioFcnName,...
        ["scenario_LC_01_SlowMoving";...
        "scenario_LC_02_SlowMovingWithPassingCar";...
        "scenario_LC_03_DisabledCar";...
        "scenario_LC_04_CutInWithBrake";...
        "scenario_LC_05_SingleLaneChange";...
        "scenario_LC_06_DoubleLaneChange";...
        "scenario_LC_07_RightLaneChange";...
        "scenario_LC_08_SlowmovingCar_Curved";...
        "scenario_LC_09_CutInWithBrake_Curved";...
        "scenario_LC_10_SingleLaneChange_Curved";...
        "scenario_LC_11_MergingCar_HighwayEntry";...
        "scenario_LC_12_CutInCar_HighwayEntry";...
        "scenario_LC_13_DisabledCar_Ushape";...
        "scenario_LC_14_DoubleLaneChange_Ushape";...
        "scenario_LC_15_StopnGo_Curved"])} = "scenario_LC_15_StopnGo_Curved";
end
```



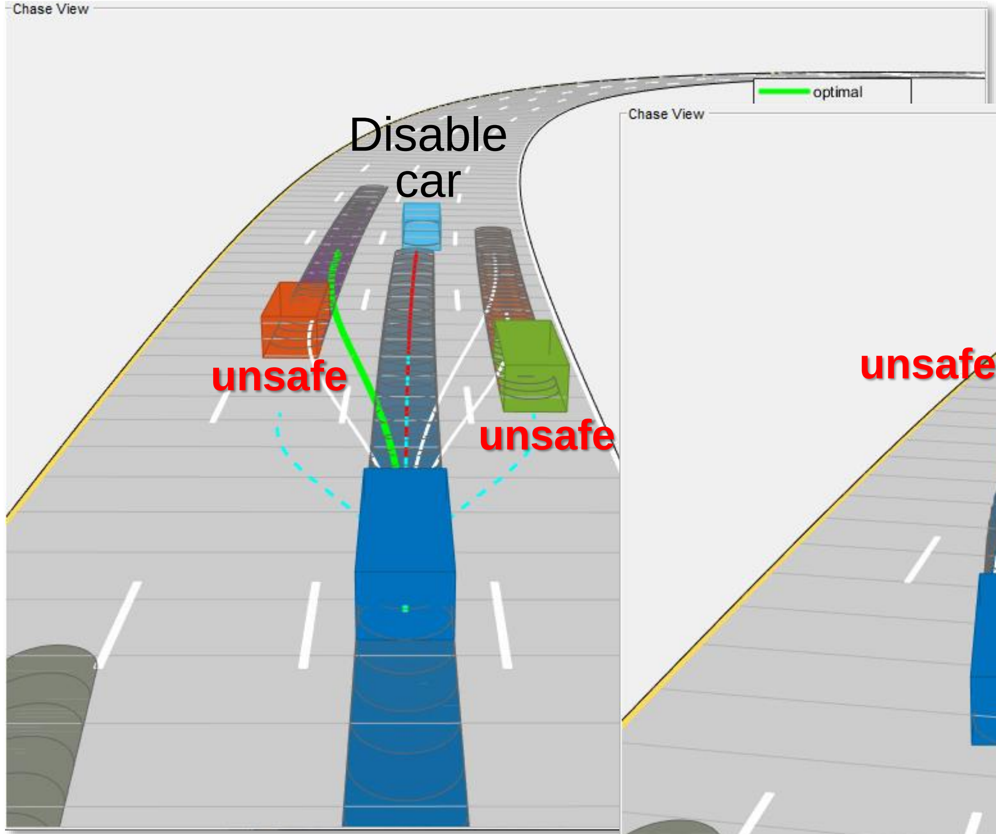


Scenarios
imported from HD map

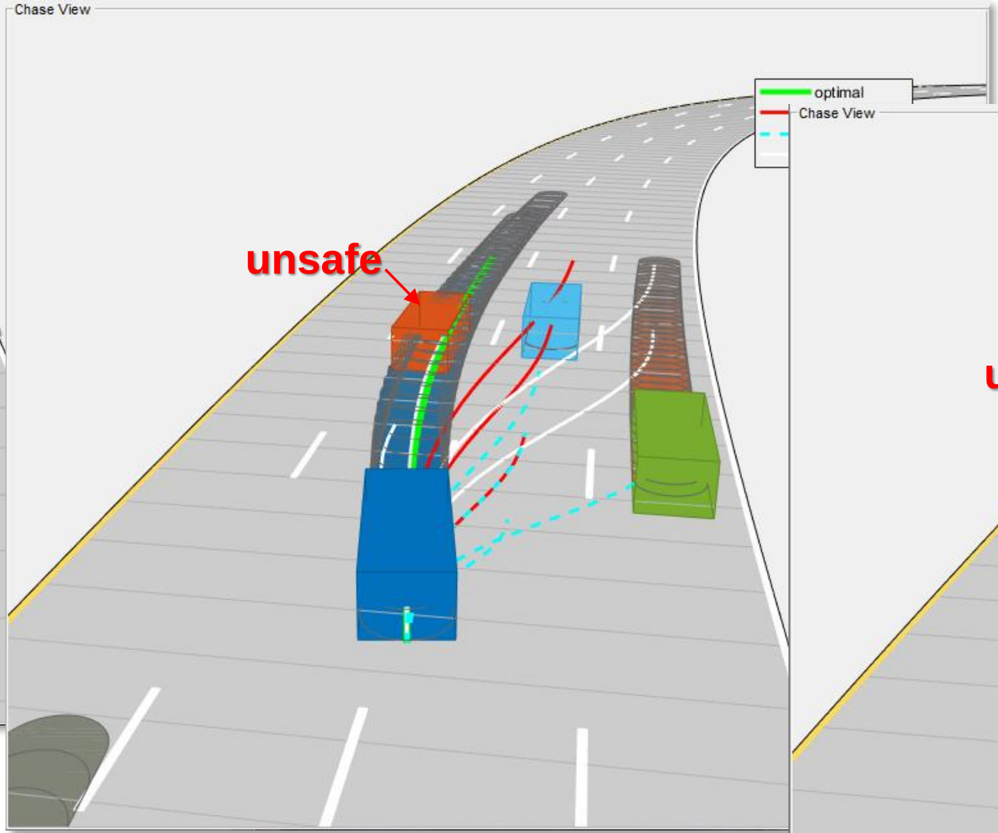
scenario_LC_14_DoubleLaneChange_Ushape



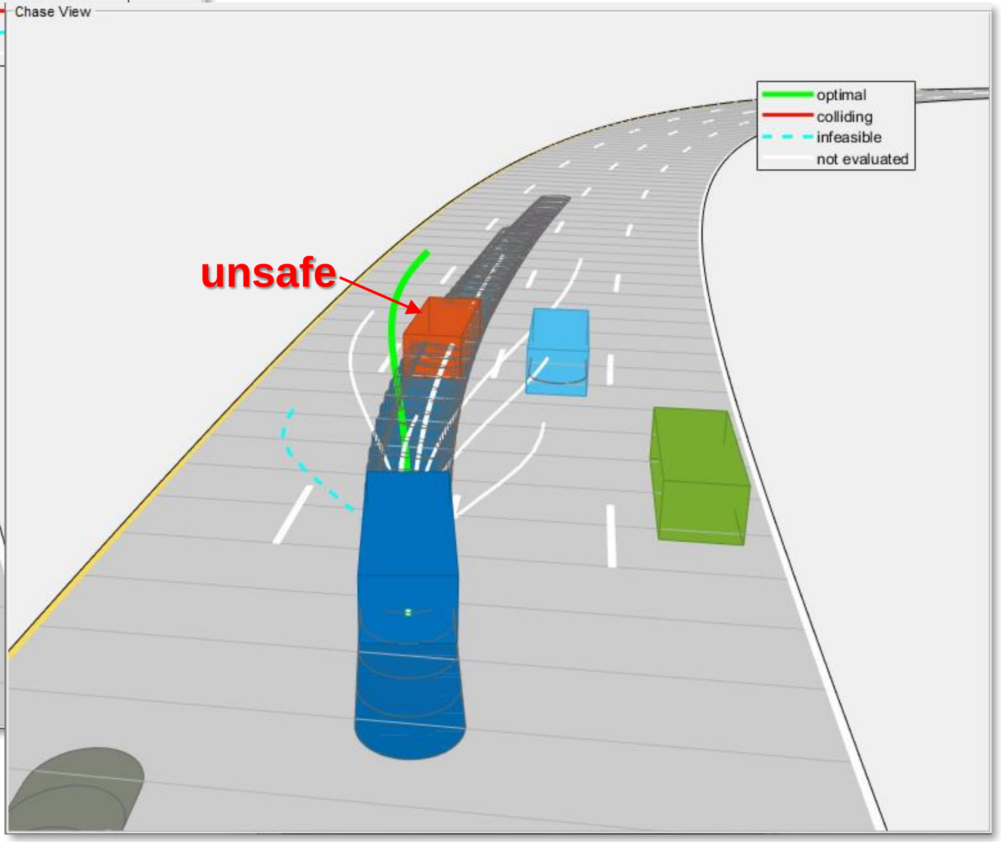
Collision avoidance against a disabled car



- Disabled car detected
- Trajectory for lane change still not collide against the unsafe target, but it’s marginal

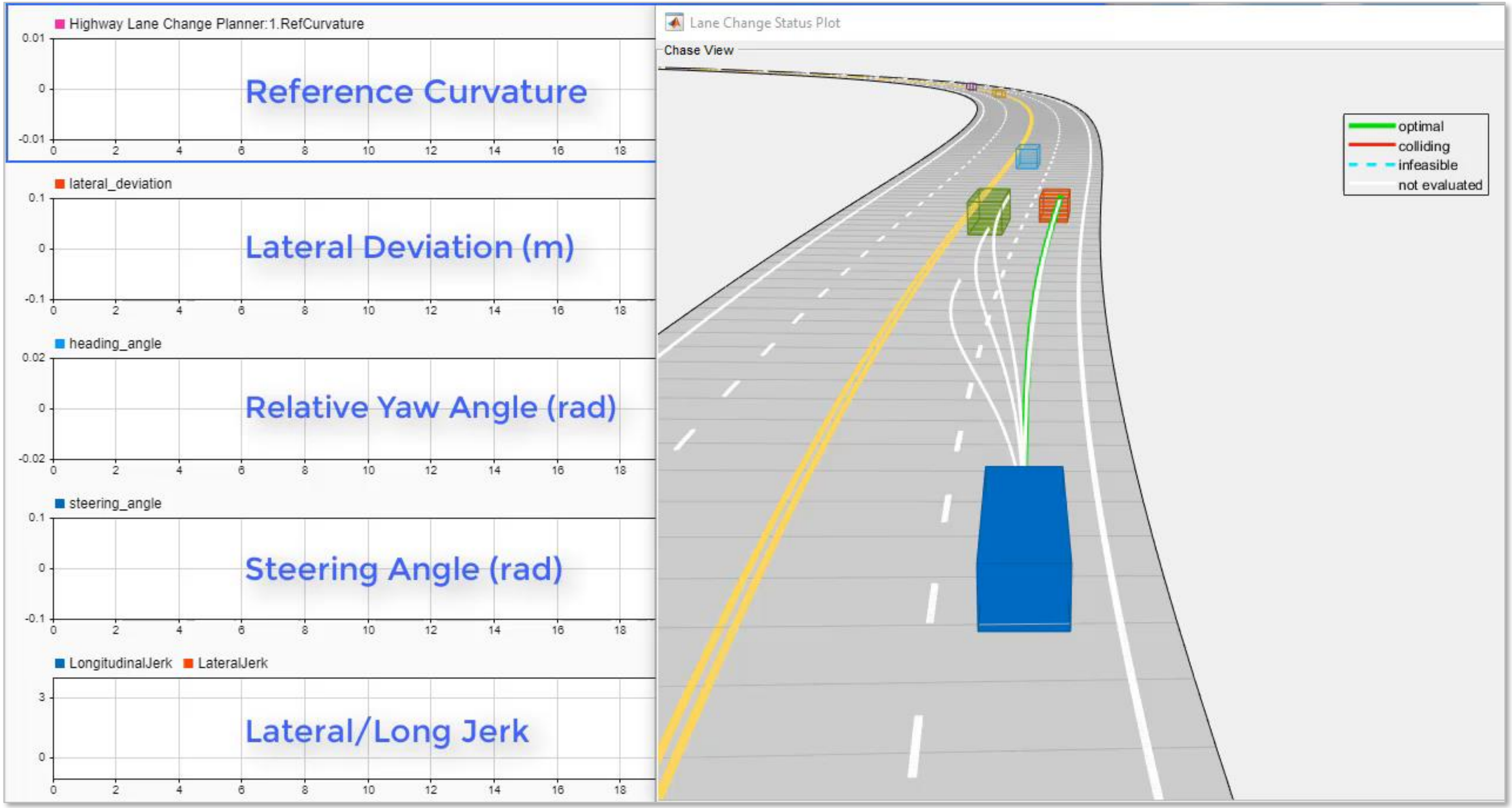


- Unsafe target still exists during lane change

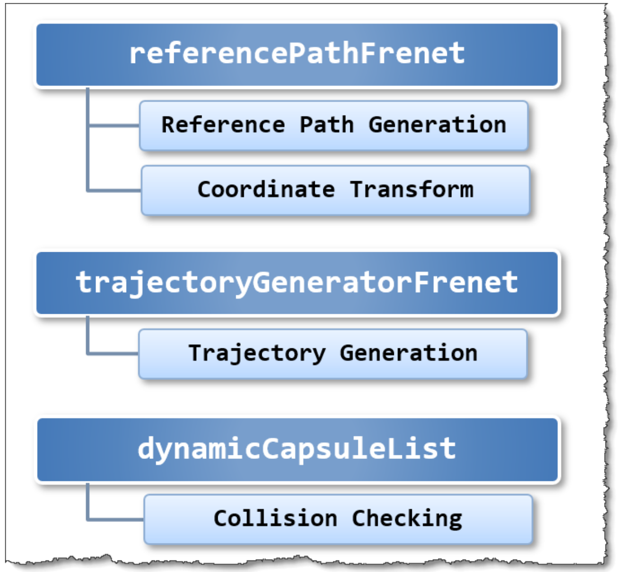


- Lane change again immediately after reaching next lane to avoid collision

Closed-loop system simulation: scenario_LC_15_StopnGo_Curved

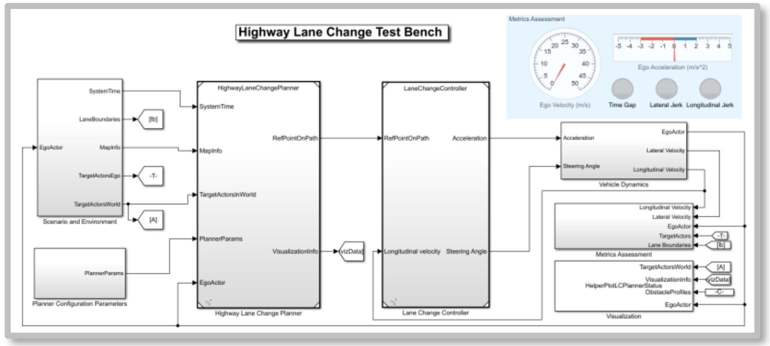


Recap: Motion planner for Highway Lane Change Maneuver



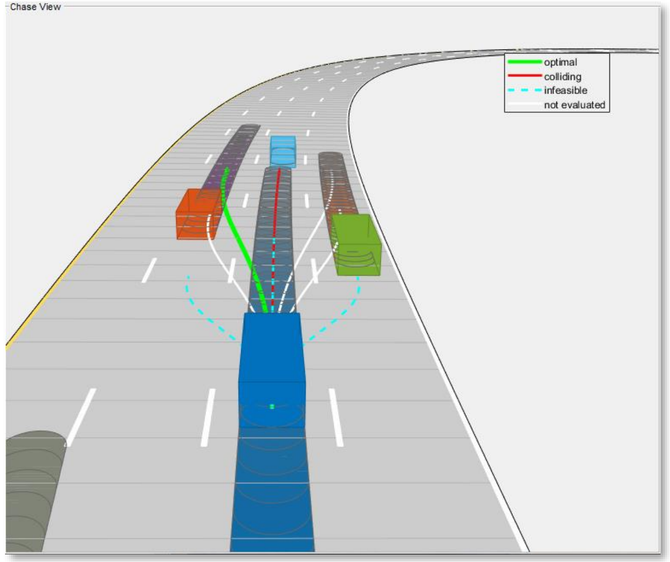
Learn fundamentals

- Motivation
- Schematic of Motion Planner
- Three MATLAB functions
  - ✓ `referencePathFrenet`
  - ✓ `trajectoryGeneratorFrenet`
  - ✓ `dynamicCapsuleList`



Implement motion planner and controller

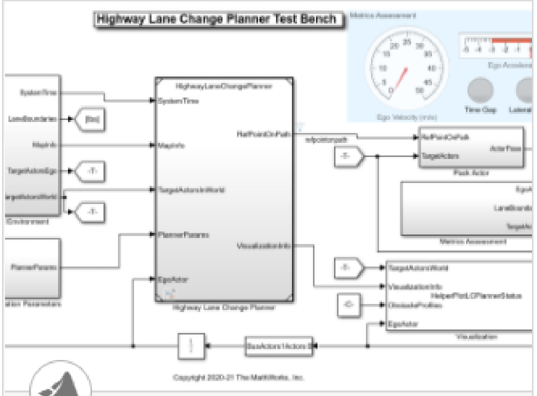
- Learn through reference example
- Architecture of highway lane change planer
- Closed-loop controller



Simulate test bench

- Scenarios in straight and curved roads
- Scenarios imported from HERE HD map

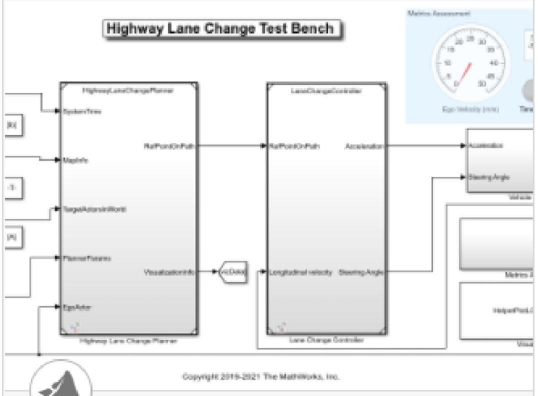
Key takeaways



Generate Code for Highway Lane Change Planner

Design, test, and generate C++ code for a lane change planner for highway driving. This example closely follows the Highway

Highway Lane Change planner



Highway Lane Change

Design and test the planner and controller components of a lane change maneuver system designed for highway driving.

Closed-loop system
+ MPC controller
+ vehicle dynamics

- Demonstrated how to design and simulate an automated lane change maneuver (LCM) system for highway driving.
- These reference examples can be used as a good framework for developing a custom LCM system.
- Navigation and Automated Driving Toolbox provide necessary components for the LCM system.

`referencePathFrenet`

`trajectoryGeneratorFrenet`

`dynamicCapsuleList`

R2021a *Automated Driving Toolbox™*
Navigation Toolbox™
Model Predictive Control Toolbox™

MATLAB EXPO 2021

Thank you

MathWorks®