Respected Company with BIC Technology

모델기반설계를 이용한 자율주행 소프트웨어 개발 적용 사례

2020. 7. 16
MANDO ADAS BU
이동한 책임

Mando

I. Challenges

- **Development Platform**
  - Linux / ROS (Robotic Operating System)
  - Easy to implement application software components with ROS

- **Simulation**
  - Unit test for each software component
  - Test scenario generation
  - Easy to utilize external resources

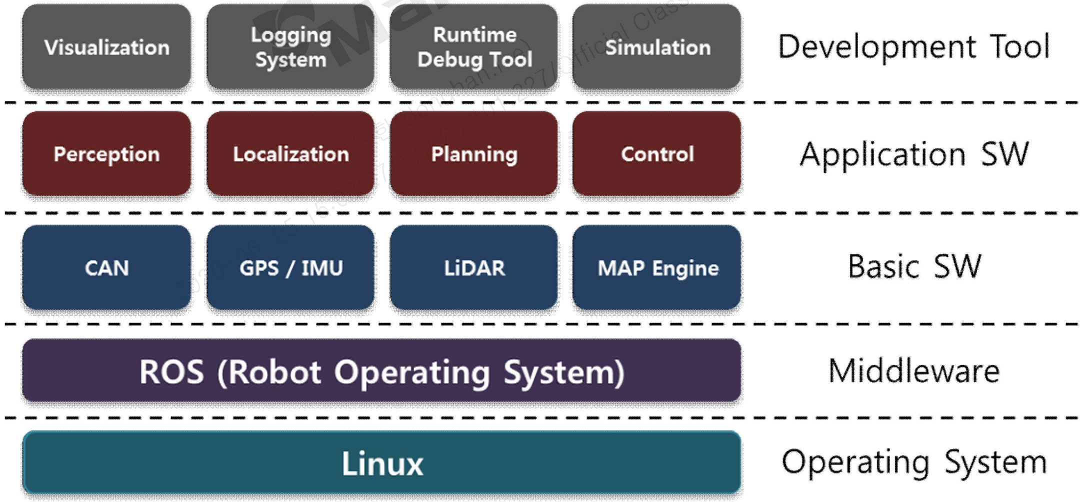
- **Model Predictive Control**
  - Optimization solver
  - Code generation

donghan.lee/2020-06-25 15:03:27/Official Class

II. Simulink models with ROS interface

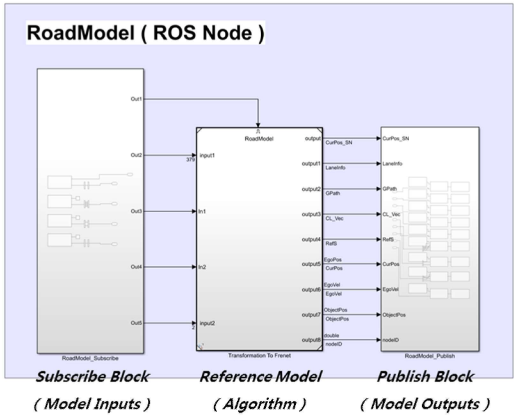
■ Development Environment

- SW Architecture based on Linux & ROS

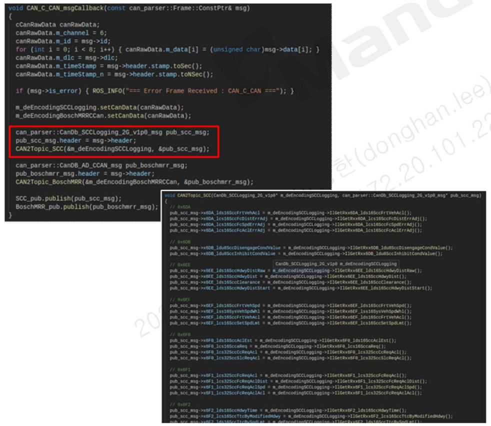


- Two ways to develop Application SW

- Planning / Control : with MATLAB



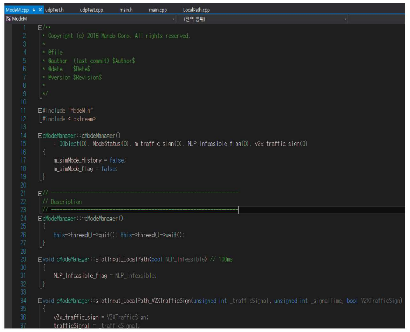
- Perception / Localization : with C++

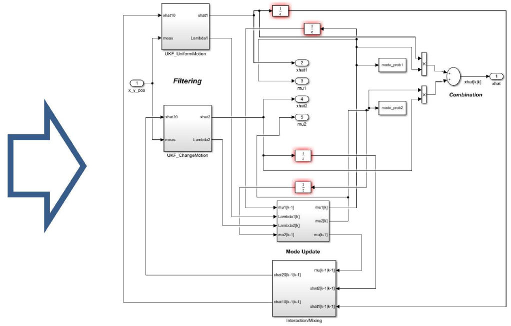


Mando Corporation

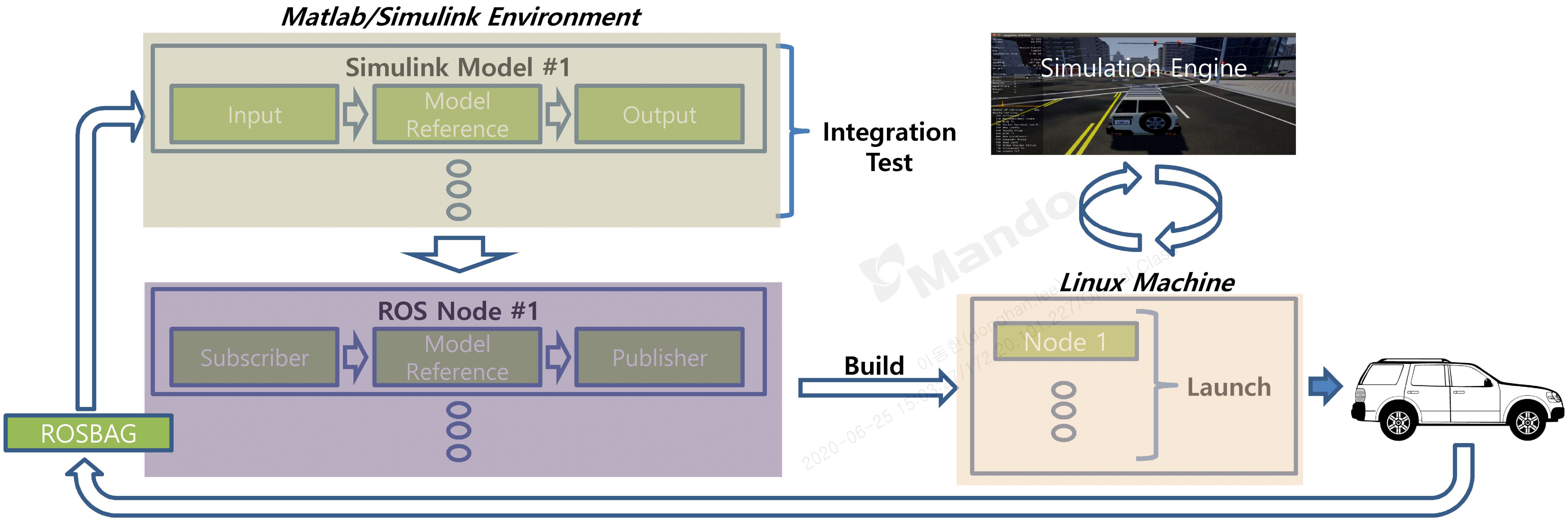
II. Simulink models with ROS interface

- **Pros**
  - Reduce the development cycle
  - Improve reusability
  - Easy to conduct unit / integration tests
- **Cons**
  - Difference from actual environment





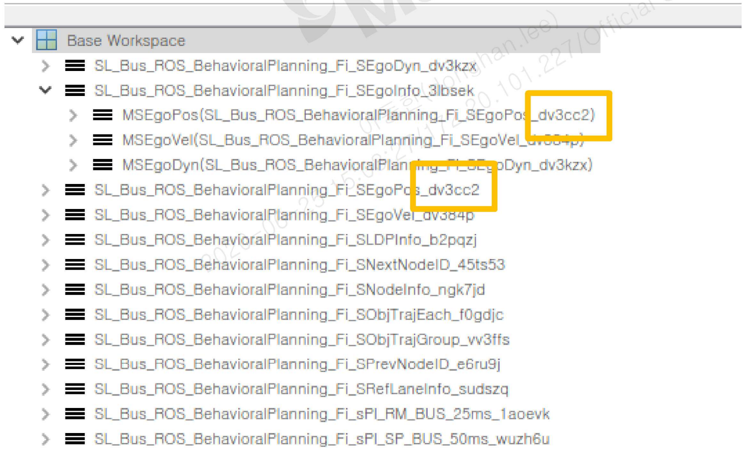
- **Model Configuration**

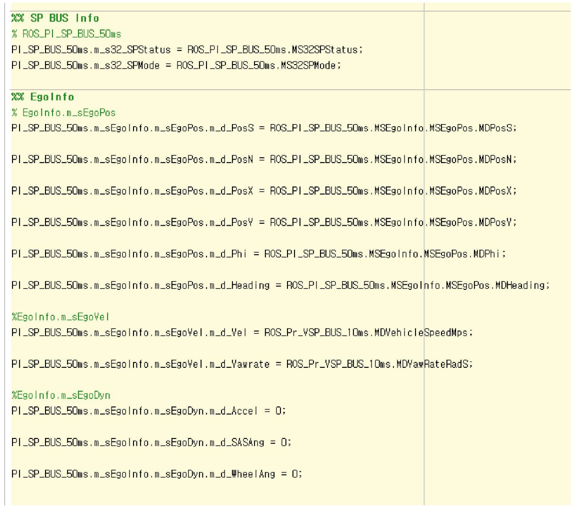


II. Simulink models with ROS interface

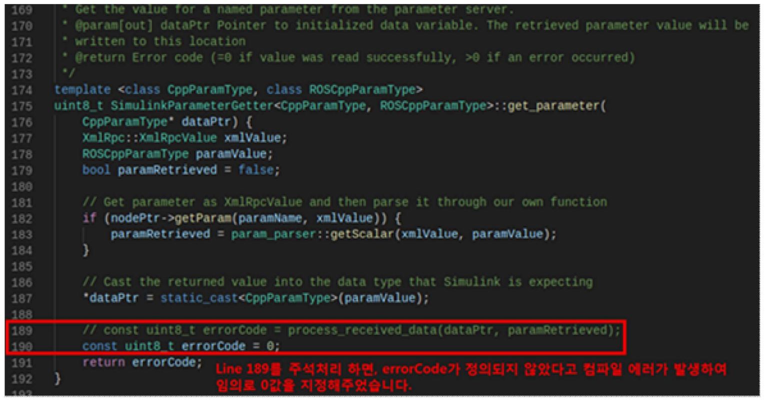
Issues

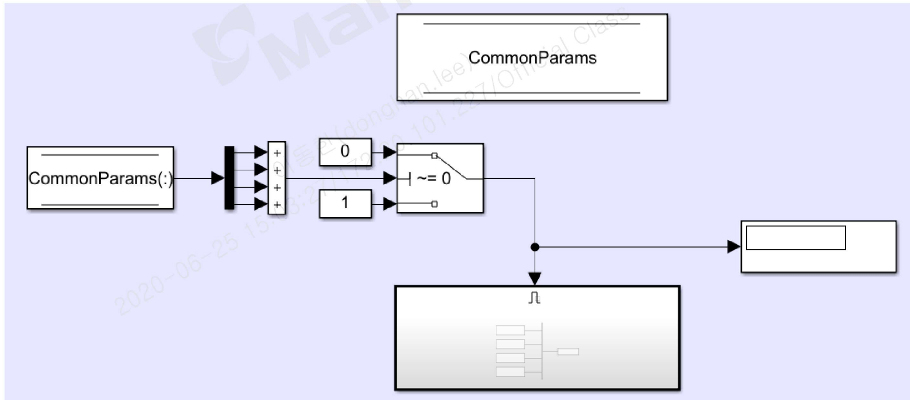
- NOT correlate automatically generated ROS message bus with pre-defined ROS message





- 'ROS Get Parameter' block makes the computational problem
  → Resolved by the initialization with 'Enable block'



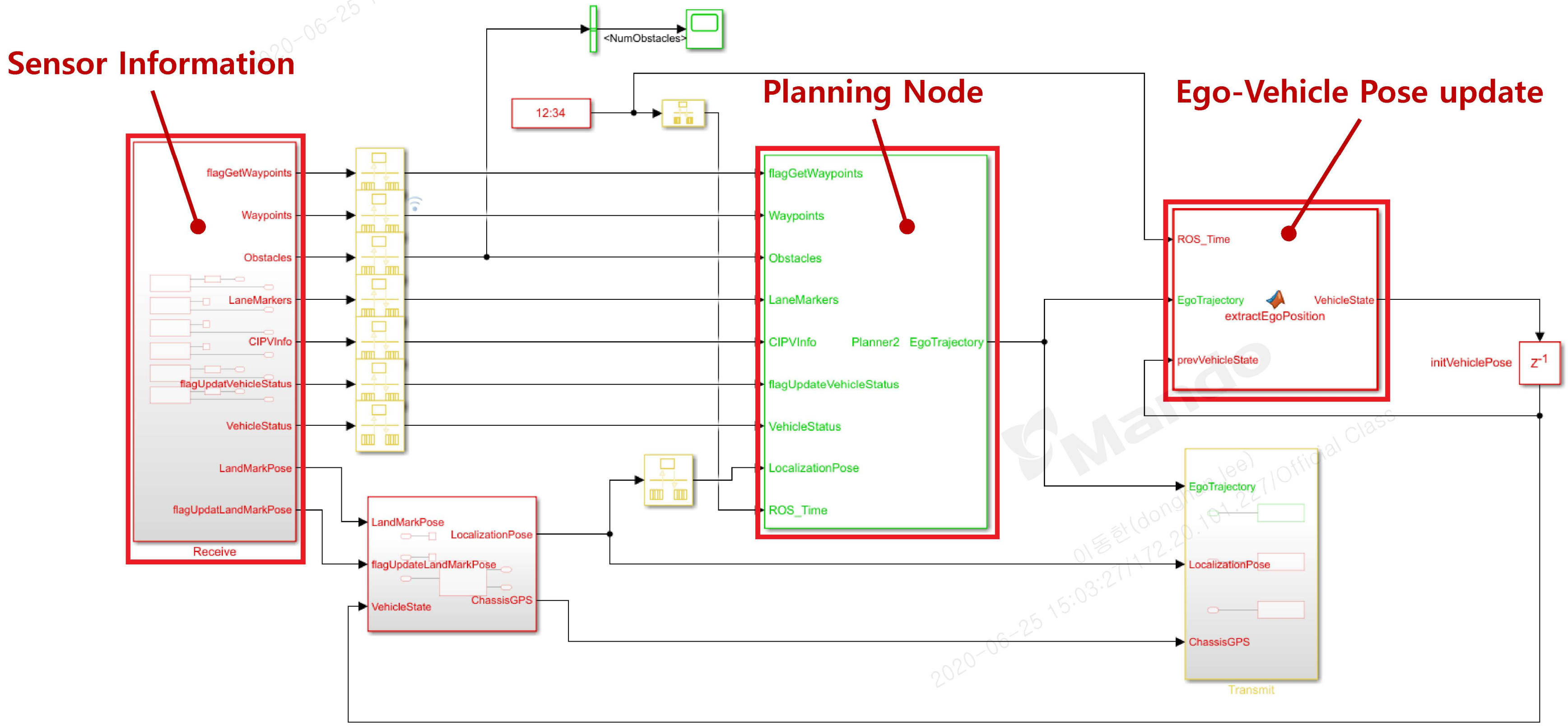


Mando Corporation

III. Simulation to evaluate a trajectory planning

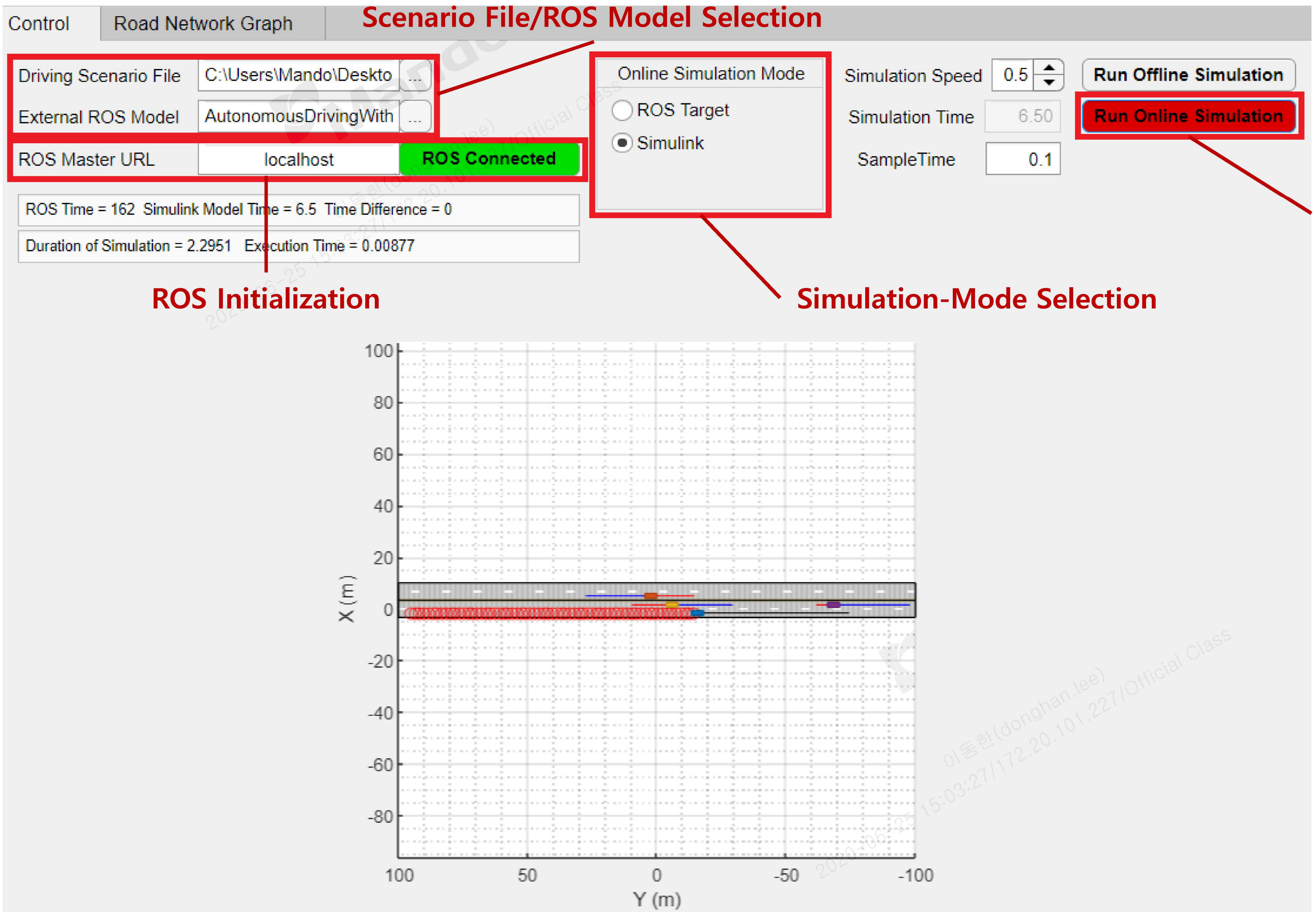
■ Simulation Model

- Goal : Planning node evaluation in diverse driving conditions
- Requirements
  - Easy to generate driving scenarios (Agents, Road attributes, etc.)
  - Work with ROS nodes
  - Take into account interaction between agents in the driving scene



III. Simulation to evaluate a trajectory planning

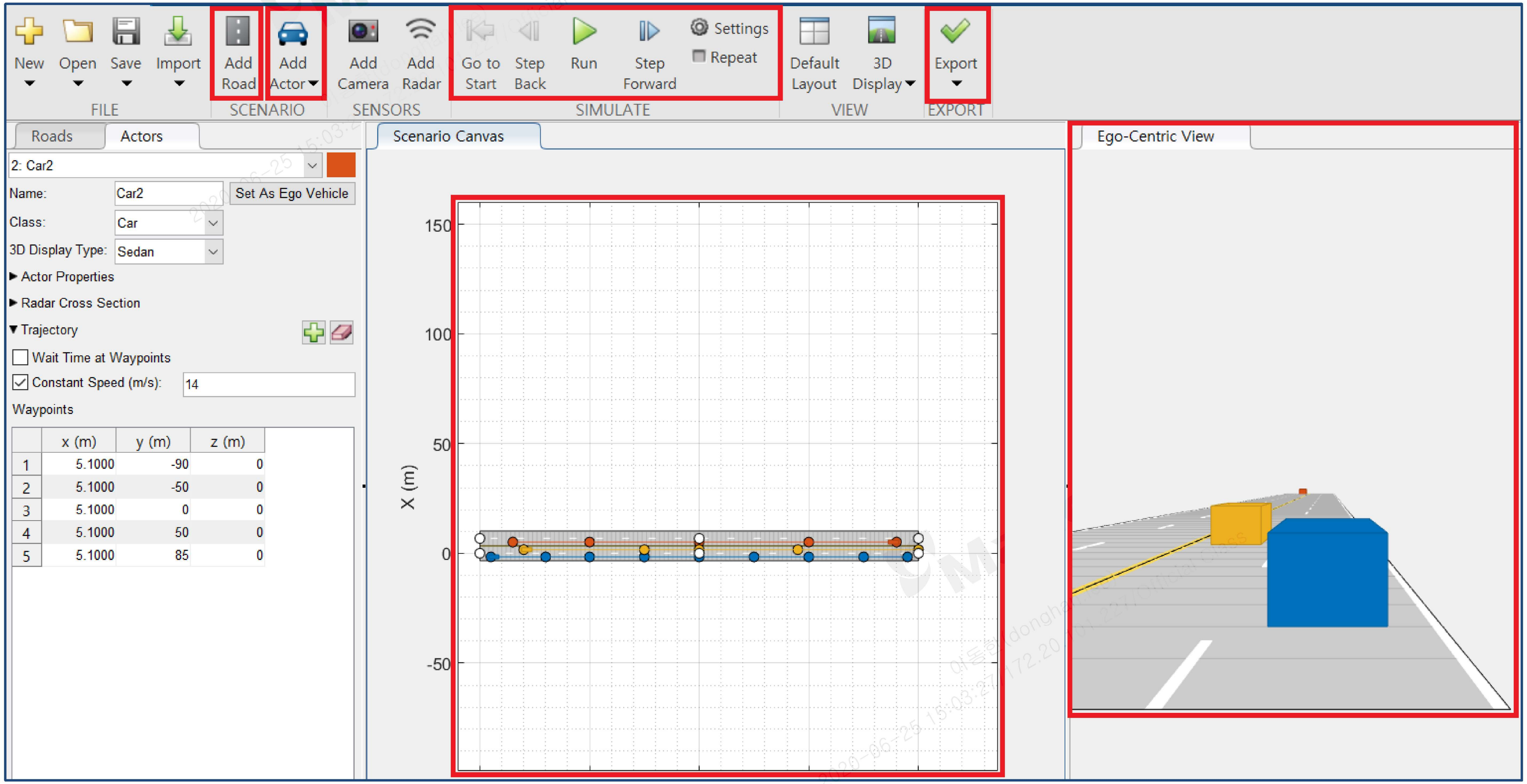
Graphic User Interface



III. Simulation to evaluate a trajectory planning

Driving Scenario Designer

- Easy to build driving scenarios
- Export information regarding road-networks and actors

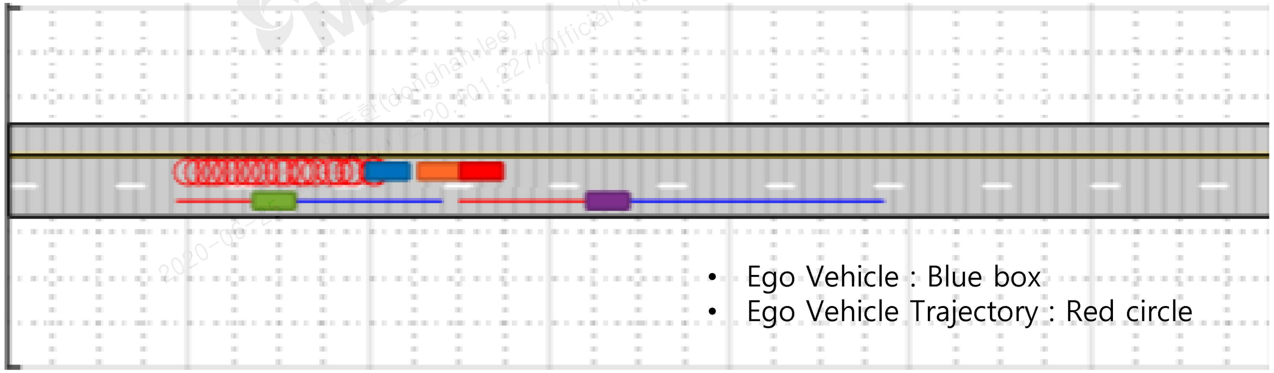


	x (m)	y (m)	z (m)
1	5.1000	-90	0
2	5.1000	-50	0
3	5.1000	0	0
4	5.1000	50	0
5	5.1000	85	0

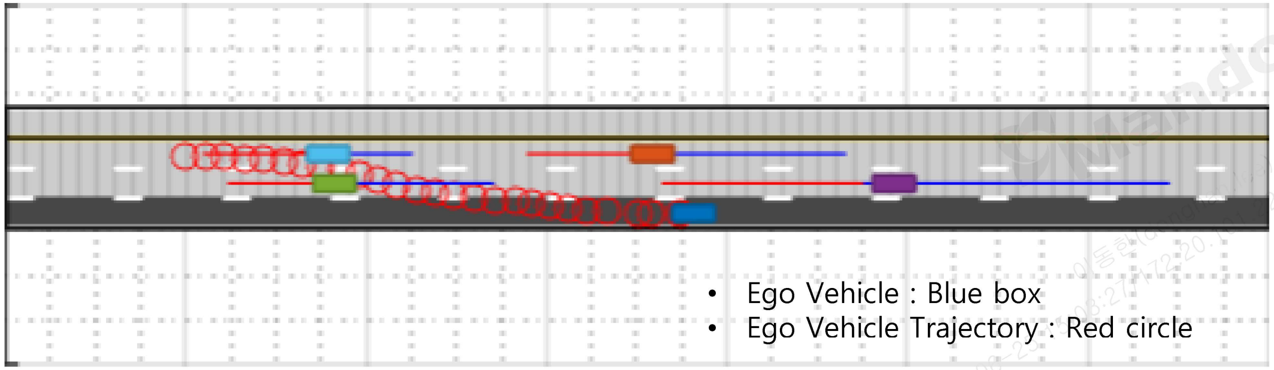
III. Simulation to evaluate a trajectory planning

■ Test case

- MRM(Minimum Risk Maneuver) / Unplanned Event
- Case 1. within the critical situation : Slow down inside the lane



- Case 2. without the critical situation : Lane change maneuvers



III. Simulation to evaluate a trajectory planning

Issues

- Repetitively simulation features on different driving conditions in a given scenario (Automated Testing)

Time data	
Start	Thu Jul 30 2015 13:58:48 GMT+0200 (CEST)
End	Thu Jul 30 2015 13:59:17 GMT+0200 (CEST)
Duration	29 sec.

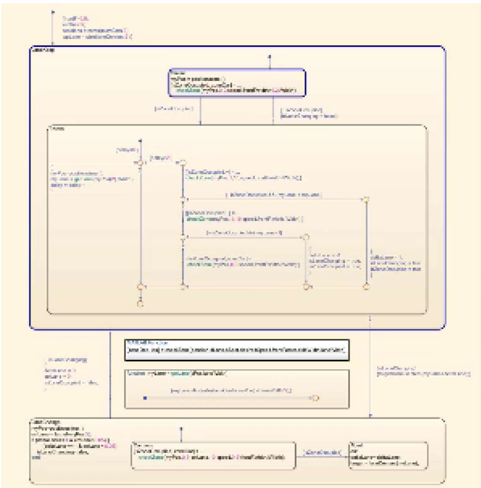
	Test Suites	Test Cases
✔ Passes	1	1
! Warnings	0	0
Failures	0	0
Skipped	0	0
Total	1	1

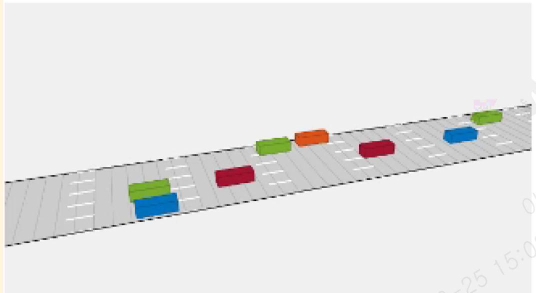
	Features	Scenarios	Steps
✔ Passes	1	4	27
! Warnings	0	0	0
Failures	0	0	0
Skipped	0	0	0
Total	1	4	27

[https://doc.froglogic.com/squish/6.0/ug-batchtesting.html]

- Interactions with other agents in a given scenario (e.g. Intersection / Merge-in / Merge-out)

  → Multi-agent simulation will be implemented based on the shipping demo. In Stateflow



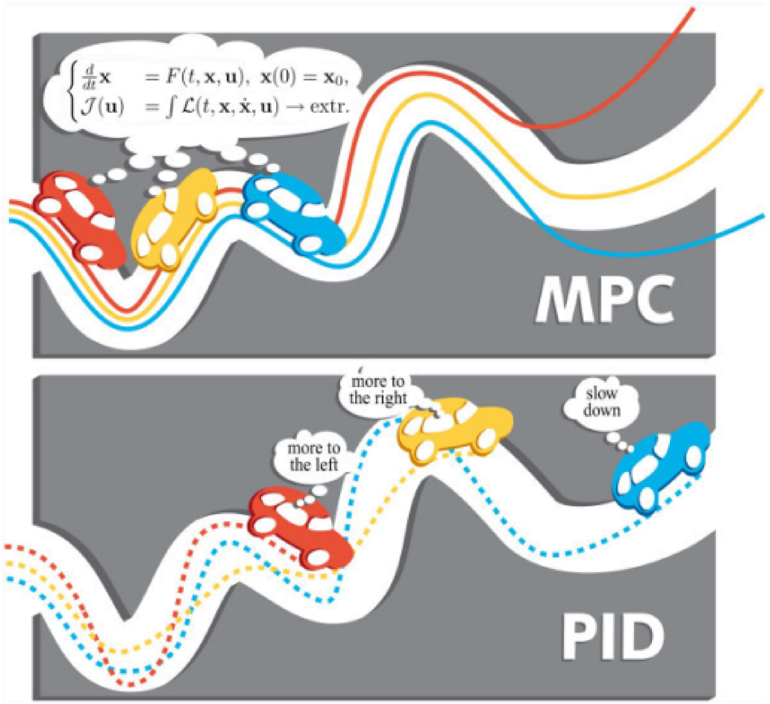


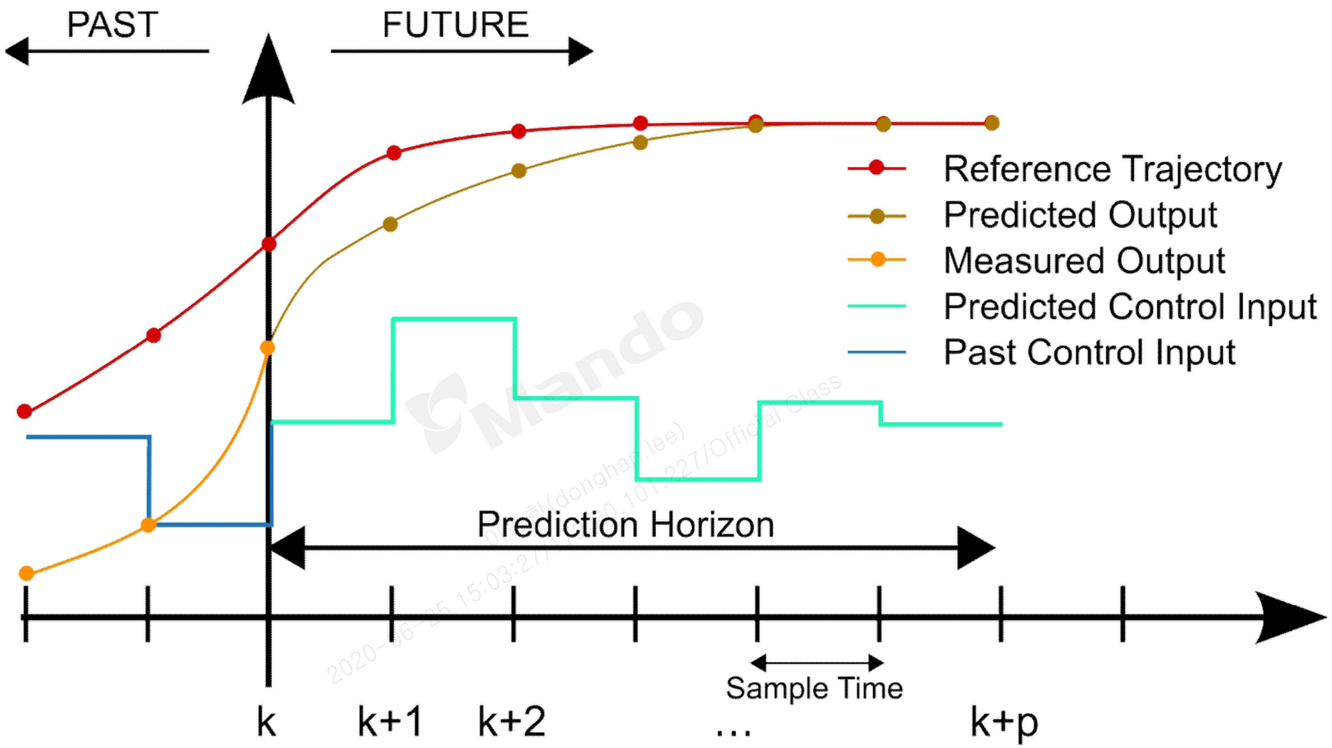
donghan.lee/2020-06-25 15:03:27/Official Class

IV. Model Predictive Control Toolbox

Model Predictive Control

- **Pros**
  - Any objective
  - Any model
  - Optimization with constraints
- **Cons**
  - Computationally demanding in the general case
  - Stability & Feasibility are not guaranteed





IV. Model Predictive Control Toolbox

Constrained finite time optimal control problem

$$\min_{\kappa_{0:N-1|t}(\cdot)} \sum_{k=0}^{N-1} E\left[J(x^c_{k+1|t}, x^{ref}_{k+1|t}, u^c_{k|t}, u^c_{k-1|t})\right]$$

$$\begin{aligned} \text{subject to: } & x^c_{k+1|t} = f^c(x^c_{k|t}, u^c_{k|t}, w^c_{k|t}), \\ & u^c_{k|t} = \kappa_{k|t}(x^c_{k|t}), \\ & \left[u^c_{k|t}, u^c_{k-1|t}\right] \in \mathcal{U}, \\ & g(x^c_{k|t}, x^e_{k|t}) \leq 0, \\ & (k = 0, \ldots, N-1) \\ & x^c_{0|t} = x^c_t, \quad u^c_{-1|t} = u^c_{t-1}. \end{aligned}$$

System Dynamics

Control policy

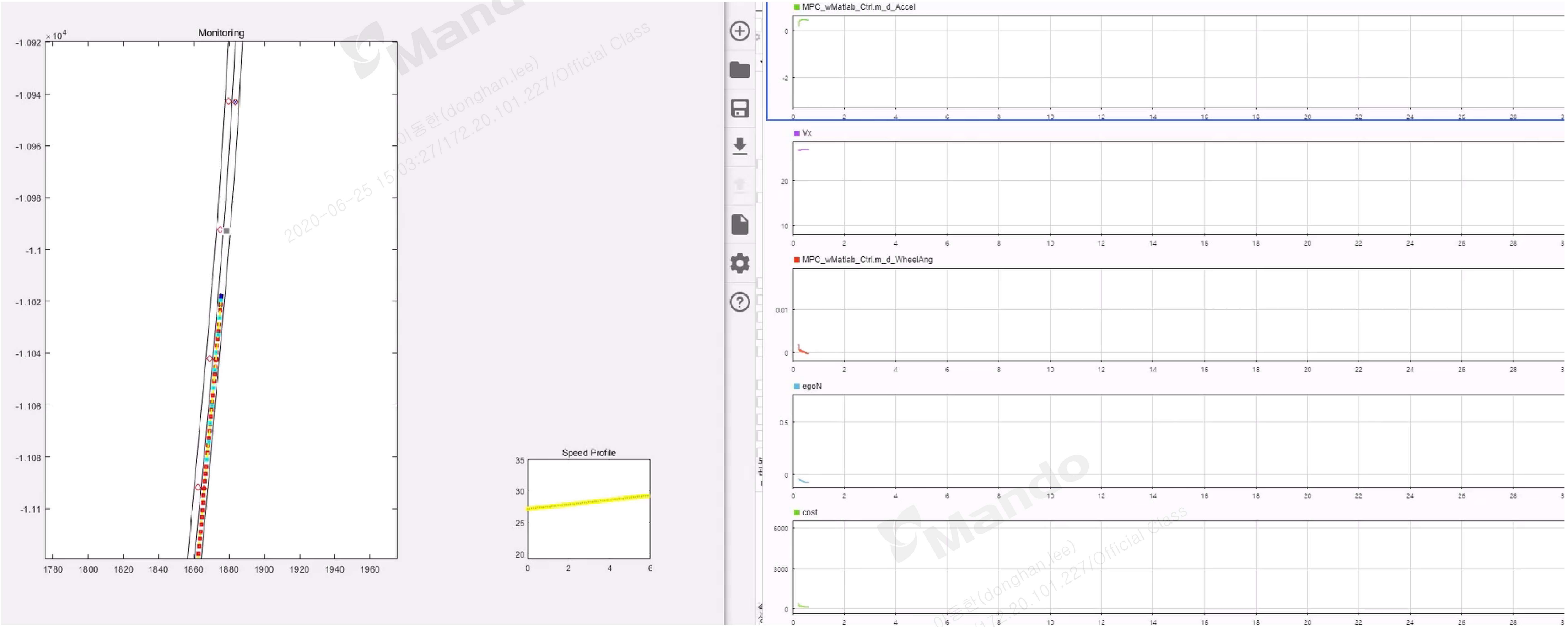
Input Constraint

State Constraint

- **Three different approaches depending on characteristics of uncertainty**
  - Nominal MPC: $g(x^c_k, x^e_k) \leq 0 \text{ with } w^*_i = \bar{w}^*_i \quad \forall i < k$
  - Stochastic MPC: $\mathbf{P}(g(x^c_k, x^e_k) \leq 0) \geq 1 - \epsilon_k \text{ with } w^*_i \sim p^*_i \quad \forall i < k$
  - Scenario-based MPC: $g(x^{c(s)}_k, x^{e(s)}_k) \leq 0 \text{ with } w^*_i = w^{*(s)}_i \quad \forall s = 1, \ldots, S, \quad \forall i < k$

IV. Model Predictive Control Toolbox

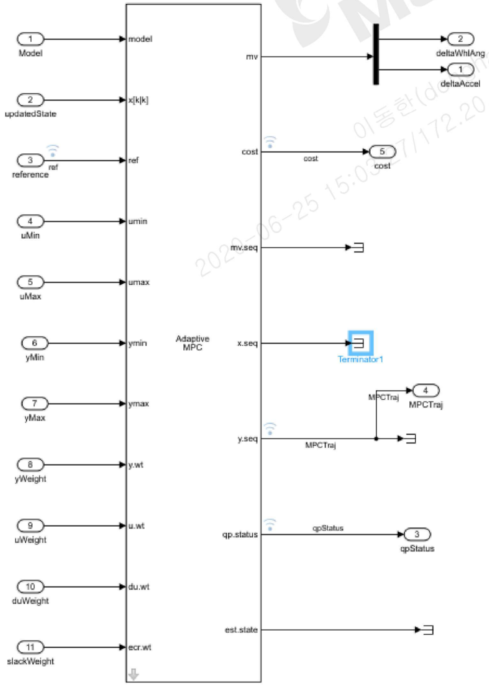
■ Simulation example

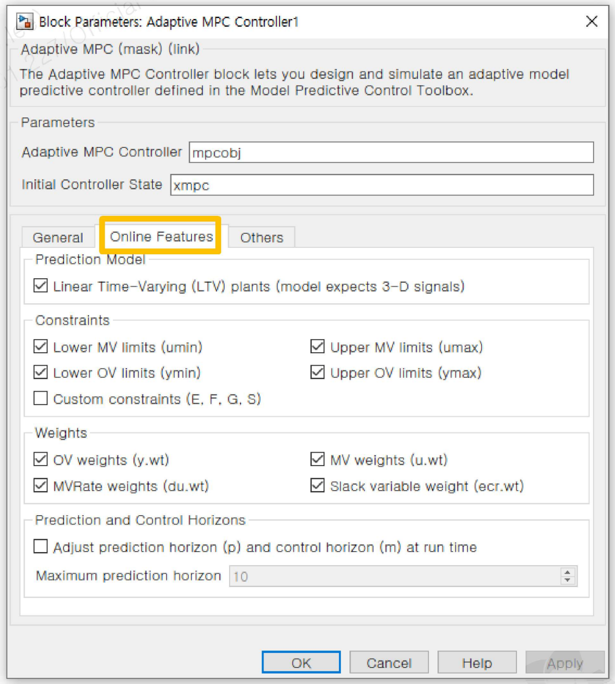


IV. Model Predictive Control Toolbox

■ Issues

- 'Adaptive MPC' has no MVRate-Constraints in Online Features





- Hard to define a problem formulation in the model
  - Script-based problem formulation option
  - Better for the design flexibility

Yalmip example code

```
Plant = ss(tf(1,[1 0 0]));
A = Plant.A;
B = Plant.B;
C = Plant.C;
D = Plant.D;
[nx,nu] = size(B);
Ts = 0.1;
Gd = c2d(Plant,Ts);
Ad = Gd.A;
Bd = Gd.B;

% Define data for MPC controller
N = 10;
Q = 10;
R = 0.1;

% Avoid explosion of internally defined variables in YALMIP
yalmip('clear')

% Setup the optimization problem
u = sdpvar(repmat(nu,1,N),repmat(1,1,N));
x = sdpvar(repmat(nx,1,N+1),repmat(1,1,N+1));

% Define simple standard MPC controller
% Current state is known so we replace this
x{1} = currentx;
constraints = [];
objective = 0;
for k = 1:N
    objective = objective + (r-C*x{k})'*Q*(r-C*x{k})+u{k}'*R*u{k};
    constraints = [constraints, x{k+1} == Ad*x{k}+Bd*u{k}];
    constraints = [constraints, -5 <= u{k}<= 5];
end

% Solve!
sol = optimize(constraints,objective);

% ...and return the optimal input
uout = value(u{1});
```

V. Summary & Future work

Summary

- Model-based design with Mathworks solutions works well in L4 AD application
  - Development cycle time reduction
  - Support flexible platform
  - Easy to evaluate each SW-C
  - Provide fast and robust optimization solvers

- Case study in the model-based design to develop L4 AD

Challenges	Our Solution	Mathworks Support
Development Platform	Simulink models with ROS	Robotics System Toolbox
Simulation	Design a simulator for unit test	Automated Driving Toolbox Driving Scenario Designer
Model Predictive Control	Adaptive MPC model	Model Predictive Control Toolbox

Future work

- Keep improving model-based design environment

- Adopt data-driven approaches in order to improve the performance and the reliability of the L4 AD system