FLCA
Design Concept

Rui ZHAO

SMART TECHNOLOGY FOR SMARTER MOBILITY

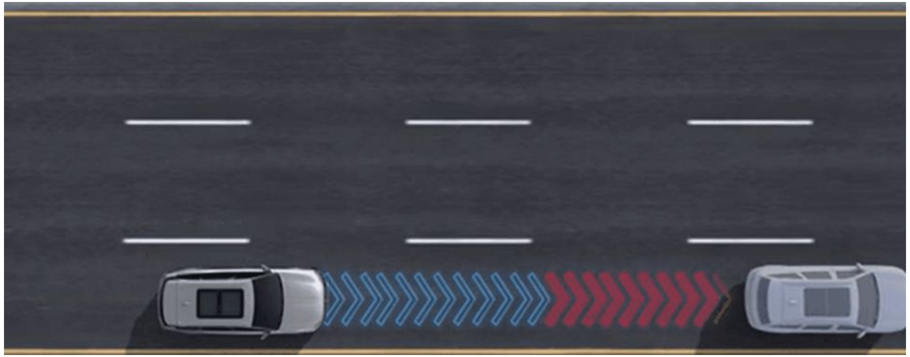
Valeo

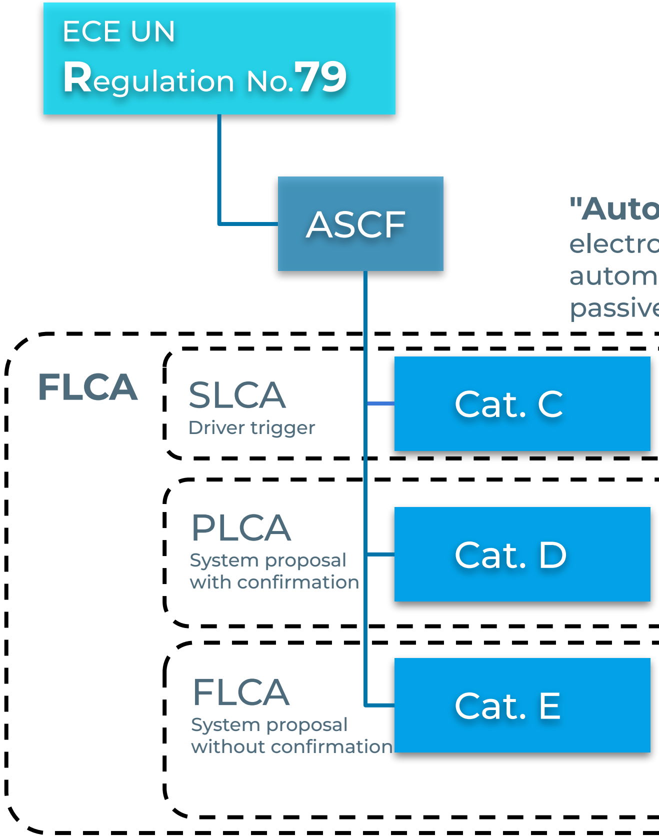
FLCA Design concept

SMART TECHNOLOGY FOR SMARTER MOBILITY

Fully-automatic Lane Change Assist General Description

Regulation Application





ECE UN
Regulation No.79

ASCF

FLCA

SLCA
Driver trigger

Cat. C

PLCA
System proposal
with confirmation

Cat. D

FLCA
System proposal
without confirmation

Cat. E

"Automatically commanded steering function " means a function within an electronic control system where actuation of the steering system can result from automatic evaluation of signals initiated on-board the vehicle, possibly in conjunction with passive infrastructure features, to generate control action in order to assist the driver.

“Category C" means, a function which is initiated/activated by the driver and which can perform a single lateral manoeuvre (e.g. lane change) ***when commanded by the driver***.

“Category D" means, a function which is initiated/activated by the driver and which can indicate the possibility of a single lateral manoeuvre (e.g. lane change) but performs that function only ***following a confirmation by the driver***

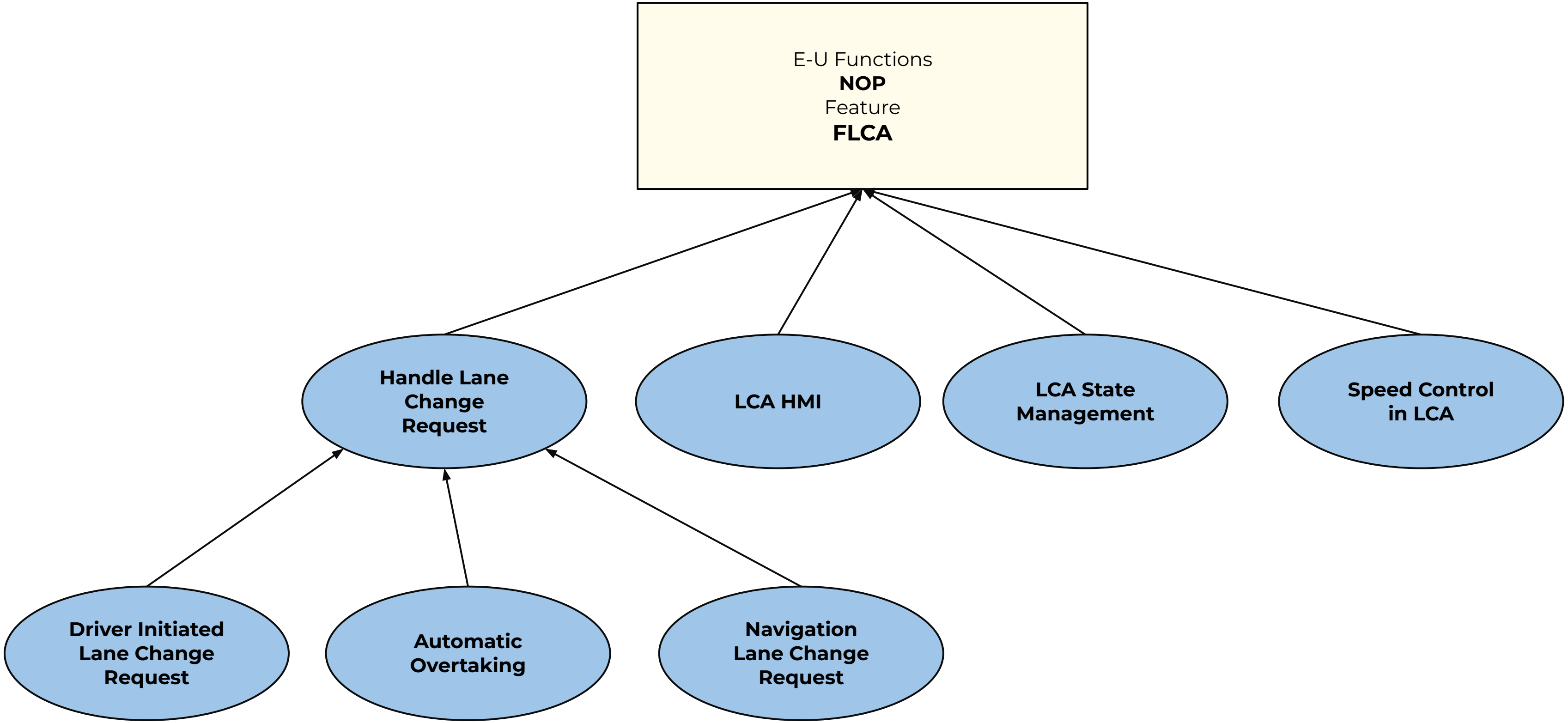
“Category E" means, a function which is initiated/activated by the driver and which can continuously determine the possibility of a manoeuvre (e.g. lane change) and complete these manoeuvres for extended periods without further driver command/confirmation.

FLCA Operational Design Condition

- Operational Design Domain
  - Highway or other closed roads.
  - Good weather that surroundings around Host Vehicle can be evaluated.
  - Map information received.
- Driver status
  - Driver in the loop: hands-on and eyes-on.
- Vehicle status
  - Speed in range of 0-150 kph.
  - During NOP is activated.

FLCA Product Design Map

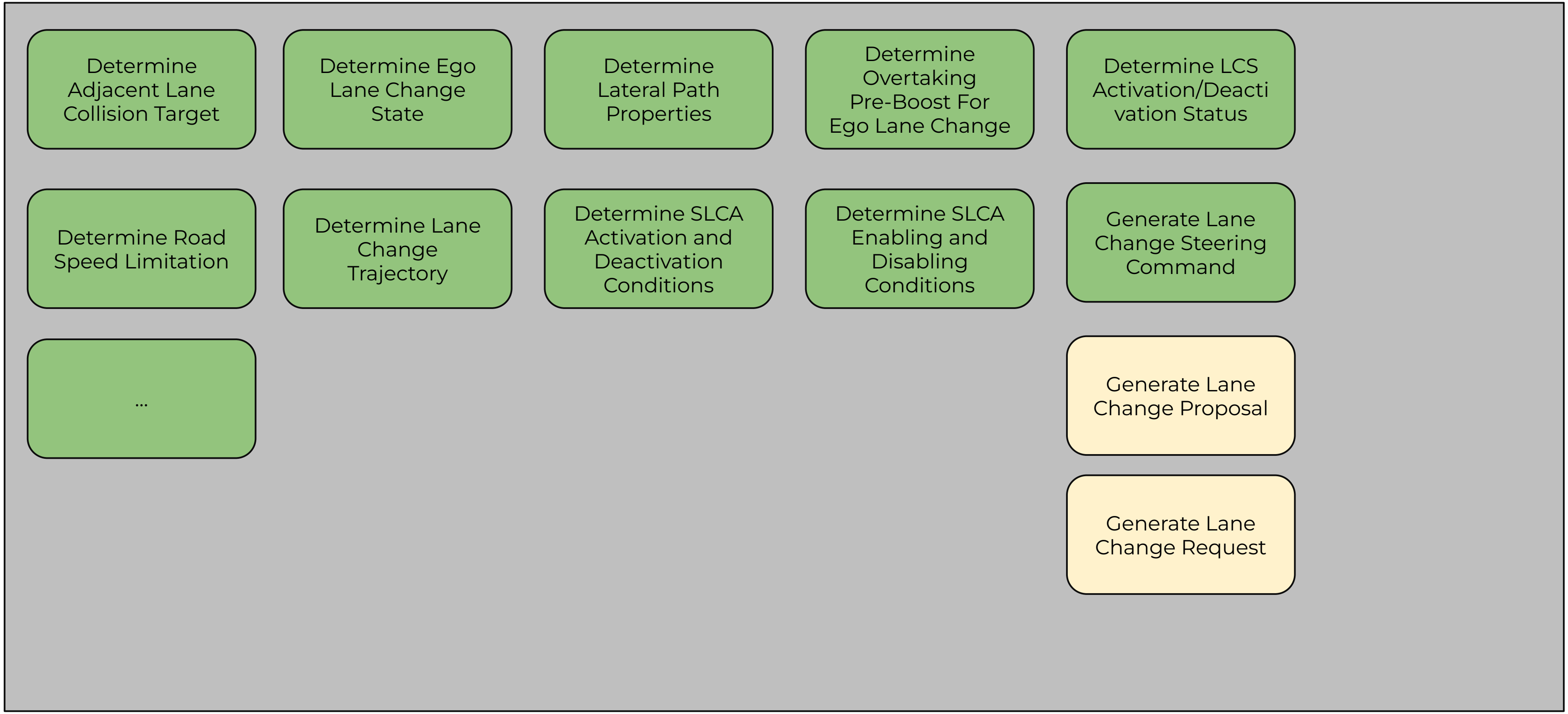
Functions and Scenario Catalogs



FLCA Functional Architecture

Finished

In review



Determine Adjacent Lane Collision Target

Determine Ego Lane Change State

Determine Lateral Path Properties

Determine Overtaking Pre-Boost For Ego Lane Change

Determine LCS Activation/Deactivation Status

Determine Road Speed Limitation

Determine Lane Change Trajectory

Determine SLCA Activation and Deactivation Conditions

Determine SLCA Enabling and Disabling Conditions

Generate Lane Change Steering Command

...

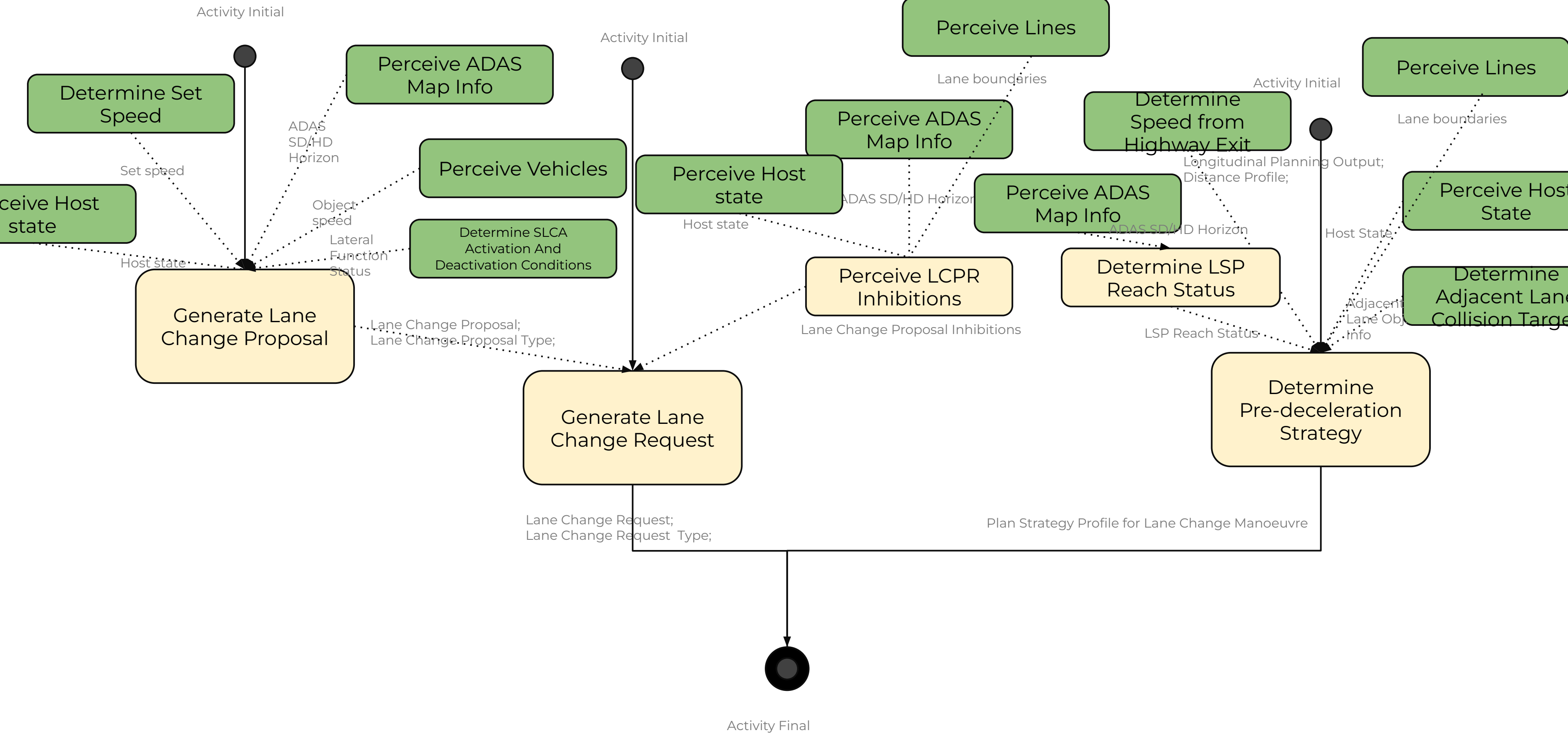
Generate Lane Change Proposal

Generate Lane Change Request

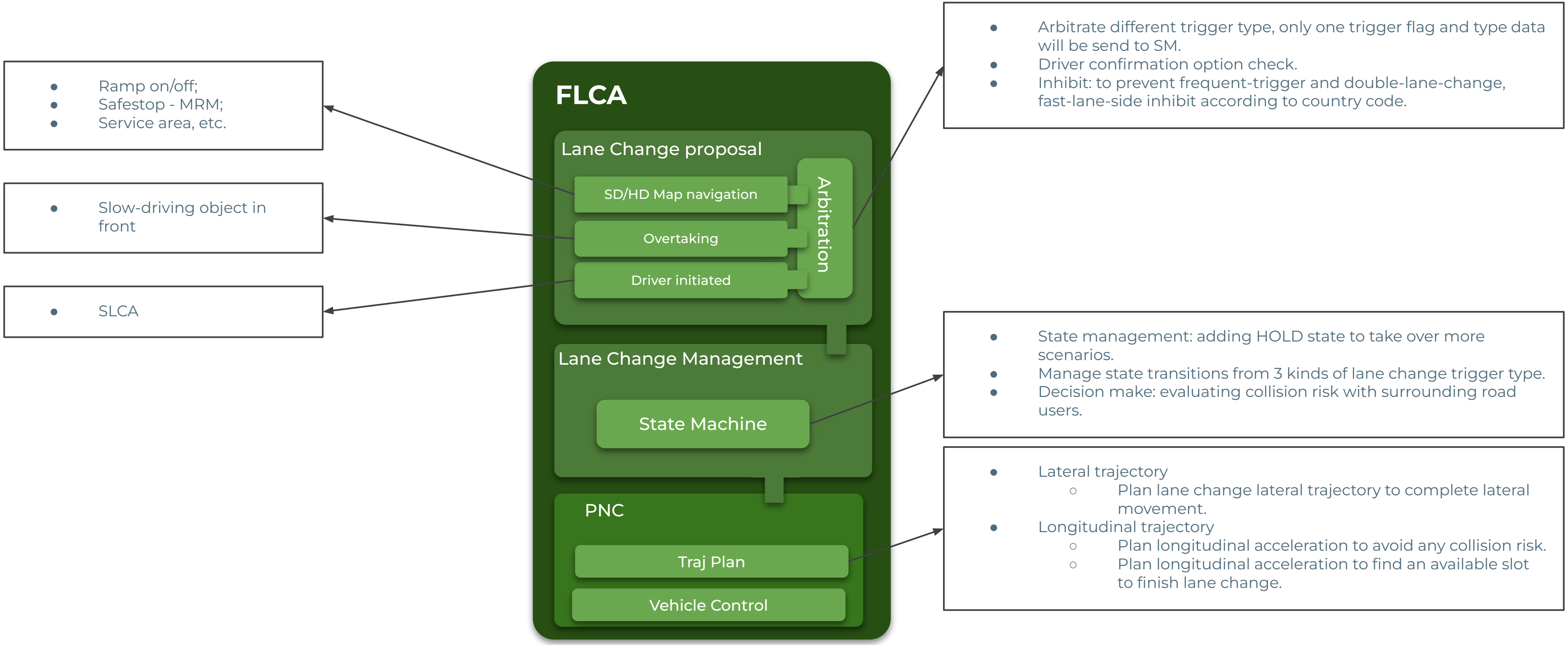
Feature Activity Diagram

🟨 In review

🟩 Finished



FLCA Functional Control Diagram



- Ramp on/off;
- Safestop - MRM;
- Service area, etc.

- Slow-driving object in front

- SLCA

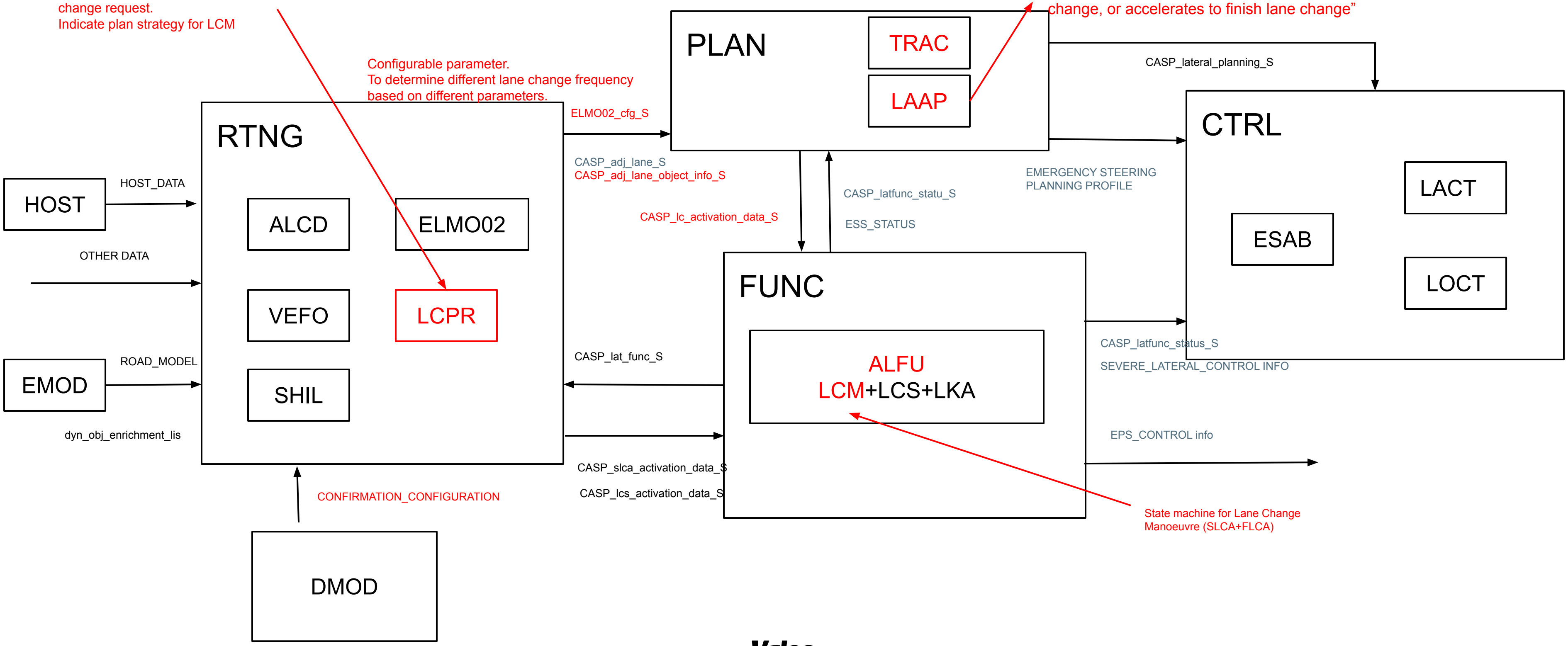
- Arbitrate different trigger type, only one trigger flag and type data will be send to SM.
- Driver confirmation option check.
- Inhibit: to prevent frequent-trigger and double-lane-change, fast-lane-side inhibit according to country code.

- State management: adding HOLD state to take over more scenarios.
- Manage state transitions from 3 kinds of lane change trigger type.
- Decision make: evaluating collision risk with surrounding road users.

- Lateral trajectory
  - Plan lane change lateral trajectory to complete lateral movement.
- Longitudinal trajectory
  - Plan longitudinal acceleration to avoid any collision risk.
  - Plan longitudinal acceleration to find an available slot to finish lane change.

LCPR FLCA Concept

FLCA Software Architecture



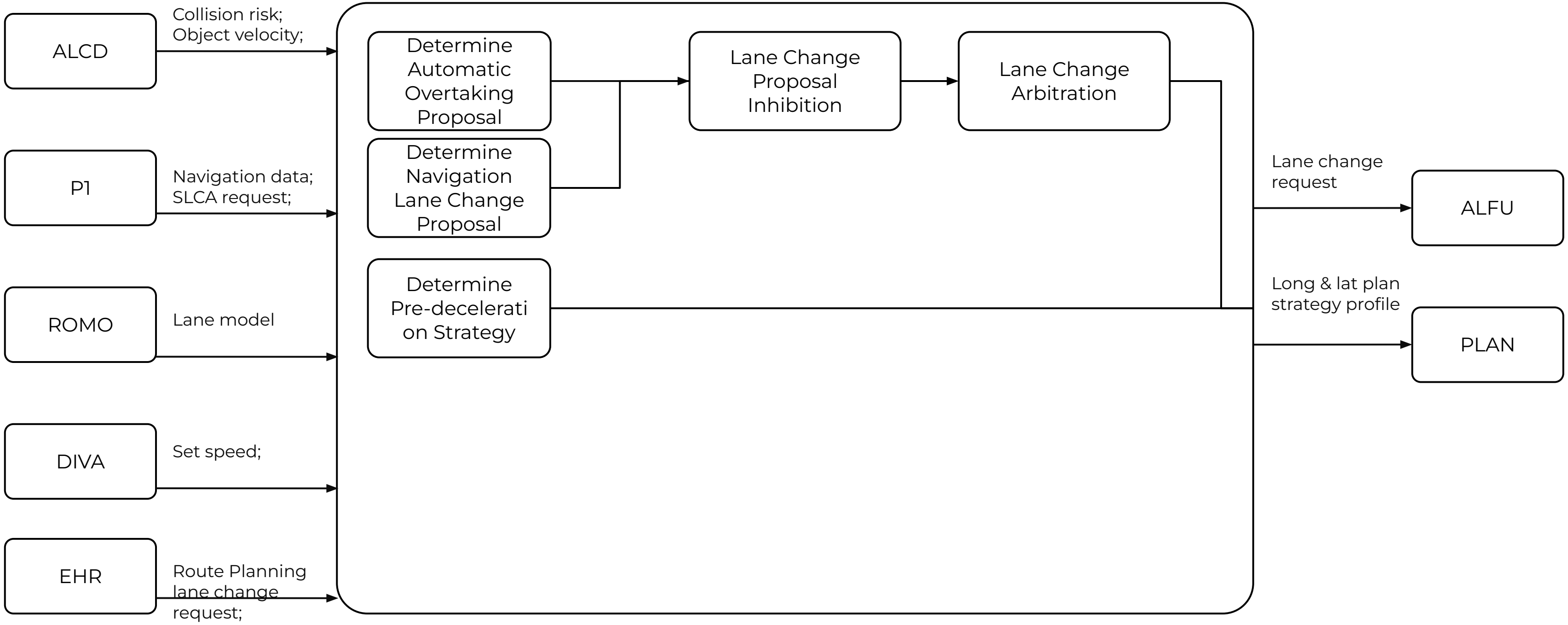
New worker to manage all lane change request.
Indicate plan strategy for LCM

Configurable parameter.
To determine different lane change frequency based on different parameters.

new longitudinal and lateral planning algorithm will be proposed in the next stage to achieve “hold lateral movement for waiting to complete lane change, or accelerates to finish lane change”

State machine for Lane Change Manoeuvre (SLCA+FLCA)

LCPR Module Structure


ALCD
Collision risk;
Object velocity;
P1
Navigation data;
SLCA request;
ROMO
Lane model
DIVA
Set speed;
EHR
Route Planning lane change request;
Determine Automatic Overtaking Proposal
Determine Navigation Lane Change Proposal
Determine Pre-decelerati on Strategy
Lane Change Proposal Inhibition
Lane Change Arbitration
Lane change request
ALFU
Long & lat plan strategy profile
PLAN

Interface definition

Determine Automatic Overtaking Proposal

- LCPR should perceive driver initiated lane change request from P1 to the LCM state machine.
- LCPR should perceive HD Map initiated lane change request.
- LCPR should generate lane change proposal for automatic overtaking and Navigation (with SD Map) lane change proposal.
- LCPR should propose lane change to the side with the highest Lane Eff.
- LCPR should define Lane Eff. for each lanes:
    - **Lane Eff. = min(ObjV, LimV)/Set_speed**
    - LimV: speed limit in according lane.
    - ObjV: Object velocity in according lane, returned by ALCD
- Efficiency Availability logic:
    - if (Eff_left >= Eff_ego)   {Eff_left_avl = available;}
    - if (Eff_right > Eff_ego)   {Eff_right_avl = available;}
    - Hysteresis - en: debounce_eff_left/right_en,   ex: debounce_eff_left/right_ex, unit: ms
- Side Determination logic:
    - if ((Eff_left_avl == available) && (Eff_right_avl == not_available)) {lc_proposal = propose_left;}
    - if ((Eff_right_avl == available) && (Eff_left_avl == not_available)) {lc_proposal = propose_right ;}
    - if ((Eff_left_avl == available)&& (Eff_right_avl == available)) {

```
   if (Eff_right > Eff_left*overtaking_side_determine_cal_factor) {
      lc_proposal = propose_right;
   else
      lc_proposal = propose_left;
   end}
else
end}
```

- Request Sending Timer: After side is determined, Request Sending Timer shall be counter down.
  Timer Period: max(0, min(t1,t2)), t2 shall be calibratable.
  After Request Sending Timer counts down, HMI reminder shall be send to remind the driver that lane change is processing, or request confirmation of driver if driver confirmation is needed.
- Once a single request or HMI reminder is send, lane change direction shall not be changed until it is completed, interrupted, or denied by the driver.

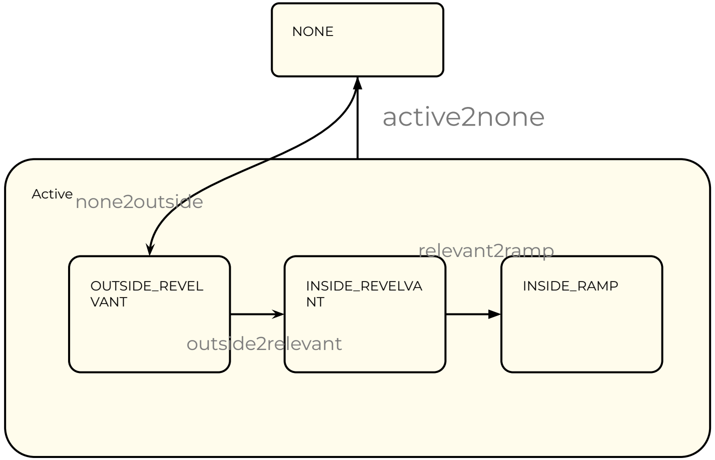
Lane Change Proposal Inhibition

- Lane change inhibition should only apply for low priority lane change proposal (3rd, 4th) in LCPR.
- Country rules: available side to overtake from will be different according to different country regulations, which shall be achieved by country code and fast lane side.
  - For EU mainland, overtaking from right shall be inhibited.
  - For CN mainland, overtaking from both sides shall NOT be inhibited.
- In case of following the same Primary Object (ObjectID), overtaking proposal shall be limited within TWICE. Ego lane index change, active indicators, function status switch, and another ignition cycle shall reset this inhibit.
- Overtaking from left and right shall be inhibited for InhibitionAlternatePeriod_L, InhibitionAlternatePeriod_R, respectively. Active indicators, function status switch, and another ignition cycle shall reset ths inhibit.
- Overtaking lane change proposal shall be inhibited on below conditions:
  - According lane marking type is solid.
  - Navigation lane change proposal is 2 kilometers ahead (for proposals of the opposite direction).
  - The ego is in ramp.
  - Road curve > OvertakingLaneChangeCurvatureThreshold
  - P1 Inhibitions
- Navigation lane change proposal shall be inhibited on below conditions:
  - P1 Inhibitions
  - RemainDistance>400 && Pre_dec_strategy == inside_relevant

Lane Change Arbitration

- Fewer number should represent higher priority of lane change proposal:
    - Driver initiated lane change proposal = 1
    - Navigation initiated lane change proposal = 3
    - Automatic overtaking lane change proposal =4

Determine Pre-deceleration Strategy

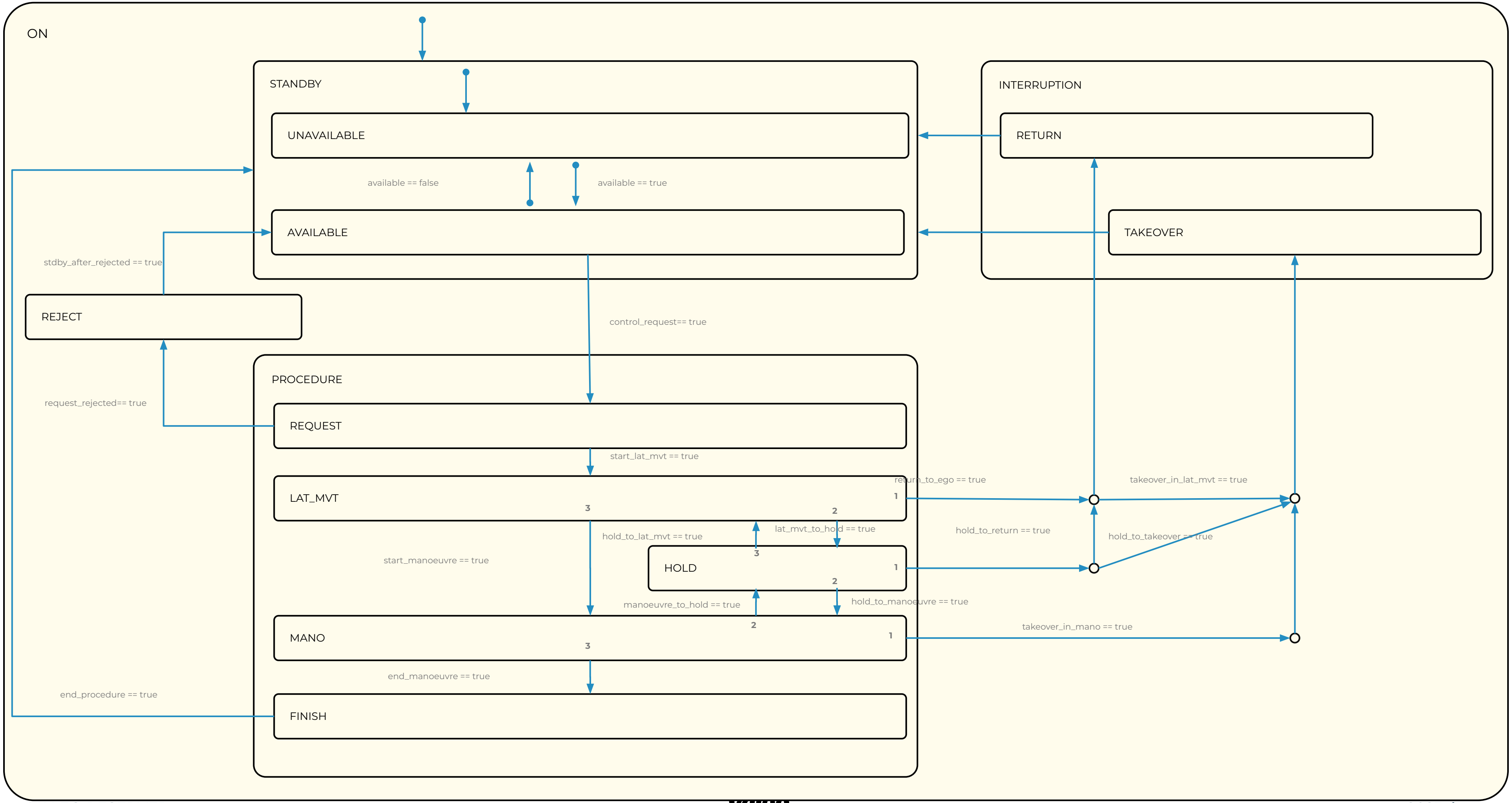


- Init State: none
- none2outside == true:
  - ||(NavNextIntersectionRemainDis < 2000m && navi_lc_direction != CASP_LCR_NO_REQUEST)
  - || speed_limit < CANCEL_DEC_STR_SPEED_CMPS
- outside2relevant == true:
  - && According lane marking type is solid
  - && AccordingLMDistanceToRoadEdge < 4m
  - && navi_lc_direction != CASP_LCR_NO_REQUEST)
  - || speed_limit < CANCEL_DEC_STR_SPEED_CMPS.
- relevant2ramp == false:
- active2none == true:
  - speed_limitation > CANCEL_DEC_STR_SPEED_CMPS && NavNextIntersectionRemainDis > 2000

Determine Navigation Lane Change Proposal

- Navigation lane change shall only be proposed when road_type is CASP_ROT_HIGHWAY or CASP_ROT_URBAN_EXPRESSWAY.
- Navigation lane change shall be proposed according to different state of pre_decel_strategy:
  - CASP_PDS_NONE: from the moment entering this state, the system shall keep requesting left change until driving over 800m.
  - CASP_PDS_OUTSIDE_RELEVANT: the system shall keep requesting the same direction of lane change with navigation message.
  - CASP_PDS_INSIDE_RELEVANT:
    - the system shall keep requesting the same direction of lane change with navigation message within the NavNextIntersectionRemainDis is smaller than 400m;
    - The system shall request left lane change if NavNextIntersectionRemainDis increases over 50m and perceived speed limit is over CANCEL_DEC_STR_SPEED_CMPS.
  - CASP_PDS_INSIDE_RAMP: the system shall keep requesting left lane change in this state.

ALFU FLCA Concept



ON

STANDBY

UNAVAILABLE

available == false

available == true

AVAILABLE

INTERRUPTION

RETURN

TAKEOVER

stdby_after_rejected == true

REJECT

control_request== true

PROCEDURE

request_rejected== true

REQUEST

start_lat_mvt == true

LAT_MVT

return_to_ego == true

takeover_in_lat_mvt == true

hold_to_lat_mvt == true

lat_mvt_to_hold == true

hold_to_return == true

hold_to_takeover == true

start_manoeuvre == true

HOLD

manoeuvre_to_hold == true

hold_to_manoeuvre == true

takeover_in_mano == true

MANO

end_manoeuvre == true

end_procedure == true

FINISH

Difference Statement- Hold Strategy

To make FLCA complete single lane change as many times as possible, there are two methods to achieve this.

1. Propose every lane change cautiously;
2. Propose as much as possible and evaluate situation to perform lane change at the proper timing;

Given the factor that above methods can work under the same performance, Propose and evaluate is better since that active turning indicators can **deliver the message to other road users that ego vehicle would like to change lane**, which means **more guaranteed safety** and **management chance**.

Therefore, besides **REQUEST** will remain for extra time compared with SLCA, **HOLD** shall be designed to continue fully control to the host vehicle and manage lane change behavior (details will be provided in part of ALFU).

Difference Statement- State Management

- LCA_OFF: overtaking feature switched off by the driver (to be discussed).
- LCA_ON: overtaking feature switched on after each ignition, containing the following sub-states:
  - LCA_STANDBY: LCA activated default state, containing the following sub-states:
    - UNAVAILABLE: LCA is not allowed to propose lane change.
    - AVAILABLE: LCA is allowed to propose lane change.
  - LCA_PROCEDURE: This group state contains the nominal lane change procedure after LCA lane change proposal. It contains the following sub-states:
    - REQUEST: LCA is proposing a lane change. Turning indicator is controlled by the function and lateral movement has not started yet. Compared with SLCA, LCA REQUEST will stay in this state longer to waiting for a proper timing to start lateral movement.
    - LAT_MVT: Lateral movement has started but the vehicle has not reached the point of no return.
    - MANO: The vehicle has crossed the point of no return, but the last wheel has not crossed target lane marking.
    - HOLD: Lane change will be held in this state that vehicle heading angle is parallelized with lane markings. This state will remain limited period and transit to LAT_MVT only.

      NOTICE: function shall entry HOLD once **in a single procedure**. The second time triggering HOLD will bring into **INTERRUPTION**.
    - FINISH: The last wheel has crossed target lane marking, but the vehicle has not reached center of target lane.
  - LCA_INTERRUPTION: This group states stand for the vehicle behavior that it will abort lane change.
    - RETURN: This state stands for cases that the function considers returning to the ego lane is the best option and is capable of doing so.
    - HAND_OVER: This state requires a take-over of driver in some critical scenarios.
  - REJECT: LCA consider canceling lane change is the best option. Turning indicator control will be released.
- FAIL: The function has a failure.

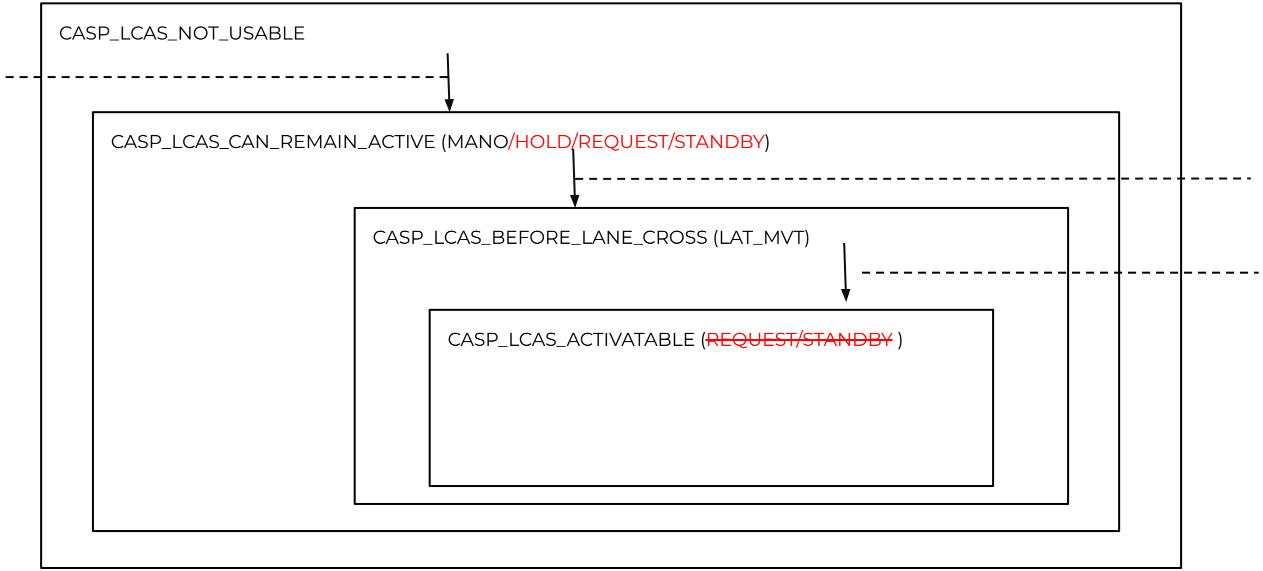
Difference Statement - Transitions

Transitions shall be managed in a general principal. It shall be divided into several types as below:

- Basic condition:
  - Lane usability is not CASP_LCAS_NOT_USABLE.
  - Speed is in range as designed.
  - No LCS inhibit condition.
  - No ACC inhibit condition.
  - P1 conditions, defined according to customer requirements.
- Collision condition:
  - Collision risk shall be defined in ALCD as 3 levels, regarding to 3 strategies in lane change process.
- Lane condition:
  - Lane marking type is crossable.

Difference Statement- Lane Usability Definition

- Left-left and left line is detected or virtual
- Ego and adjacent line is >10m
- Ego and adjacent line is overlapped with HV.



CASP_LCAS_NOT_USABLE

CASP_LCAS_CAN_REMAIN_ACTIVE (MANO/HOLD/REQUEST/STANDBY)

CASP_LCAS_BEFORE_LANE_CROSS (LAT_MVT)

CASP_LCAS_ACTIVATABLE (~~REQUEST/STANDBY~~)

- ***
- **Ego line is dashed.**

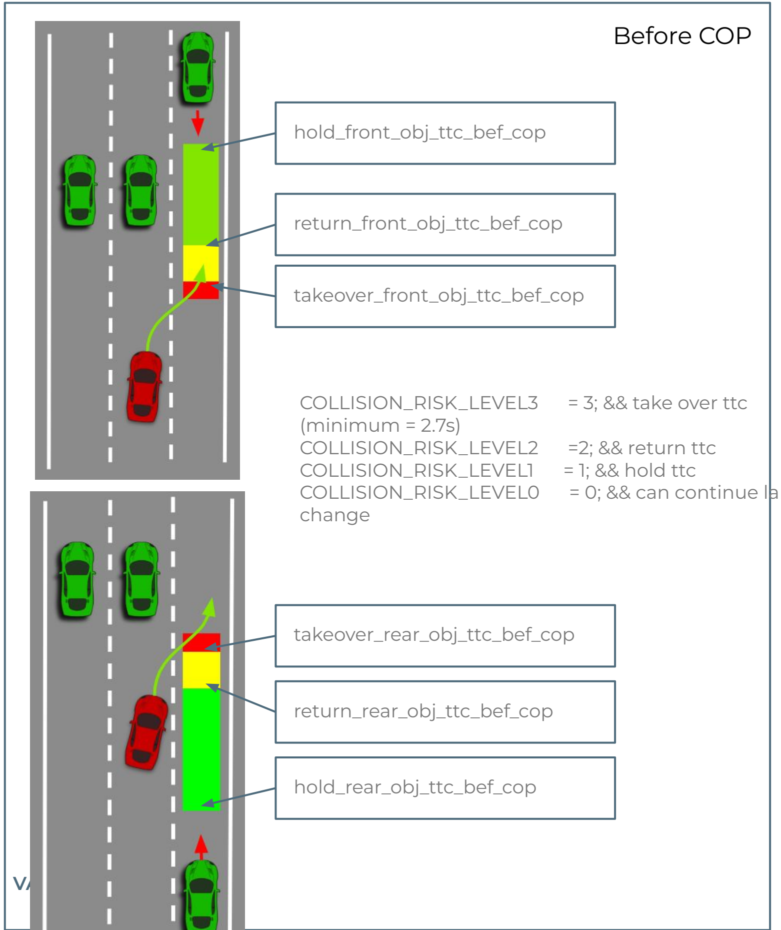
- ***
- **Left-left line NOT none.**
- **Right line NOT none.**
- **Left-left and left line is stably detected.**
- **Distance>LookAhead_dist**
- **LaneWidth> 83m**

For left lane changing cases

Driver Confirmation Check

- Driver confirmation check:
    - If driver chooses “No need confirmation”, lane change proposal shall propose lane change request to ALFU after Timer counting down;
    - If driver chooses “Need confirmation”, lane change proposal shall activate a HMI reminder to request for driver confirmation after Timer counting down.
        - If driver confirmed lane change, request will be send to ALFU
        - If driver denied lane change, reminder shall be canceled.
    - Reminder lasts for ReminderPeriod s.
    - During Reminder lasts, driver confirmation shall propose lane change request; driver denial shall cancel reminder and reset Timer.
    - Reminder cancel:
        - || Reminder lasts for over threshold;
        - || driver denial;

Collision Risk TTC Definition



Before COP

hold_front_obj_ttc_bef_cop

return_front_obj_ttc_bef_cop

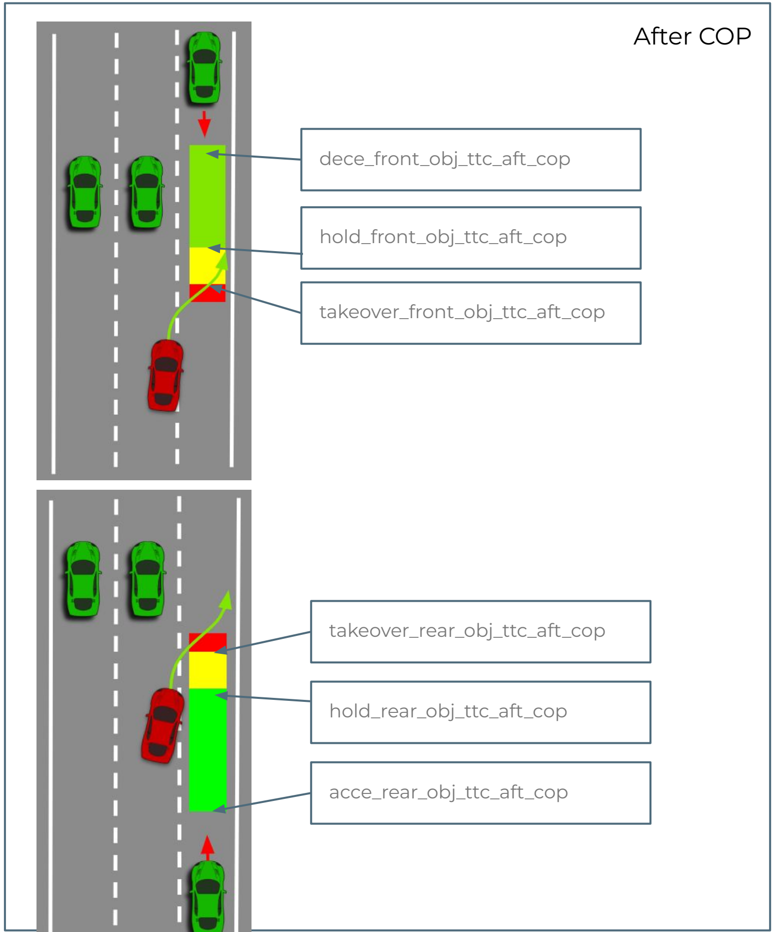
takeover_front_obj_ttc_bef_cop

takeover_rear_obj_ttc_bef_cop

return_rear_obj_ttc_bef_cop

hold_rear_obj_ttc_bef_cop

COLLISION_RISK_LEVEL3 = 3; && take over ttc (minimum = 2.7s)
COLLISION_RISK_LEVEL2 =2; && return ttc
COLLISION_RISK_LEVEL1 = 1; && hold ttc
COLLISION_RISK_LEVEL0 = 0; && can continue lane change



After COP

dece_front_obj_ttc_aft_cop

hold_front_obj_ttc_aft_cop

takeover_front_obj_ttc_aft_cop

takeover_rear_obj_ttc_aft_cop

hold_rear_obj_ttc_aft_cop

acce_rear_obj_ttc_aft_cop

available

&& Basic condition is met.

control_request

&& target side is available

&& lane change request is received.

request_rejected

|| request time is over MAX_REQUEST_DURATION_MS

|| Basic condition not met.

start_lat_mvt

&& Basic condition met.

&& No Collision condition.

&& Lane condition met, target side lane usability is CASP_LCAS_ACTIVATABLE

In this concept slides, cases are explained with left lane change for an example.

lat_mvt_to_hold

&& EnterHoldCount == 0

|| Collision condition is COLLISION_RISK_LEVEL1

hold_to_lat_mvt

hold2latmvt = FALSE

hold_to_return

|| Collision condition is COLLISION_RISK_LEVEL2;

|| The system spends more than MAX_HOLD_TO_RETURN_TIME_MS && The ego vehicle has not passed the cross-over point. && No collision risk in initial lane.

|| lane condition not met, lane marking is not crossable.

|| basic condition not met.

return_to_ego

|| Lane condition not met, lane marking is not crossable.

|| Basic condition not met.

|| Collision condition is COLLISION_RISK_LEVEL2.

takeover_before_pnr

|| basic condition not met.

|| Collision condition is COLLISION_RISK_LEVEL3 && the host vehicle has reached the point of no return.

|| The system spends more time than max_time_to_cross_line_ms

start_manoeuvre

&& Basic condition met.

&& Lane condition is dashed, lane usability is CASP_LCAS_ACTIVATABLE or CASP_LCAS_BEFORE_LANE_CROSS

&& All wheels in lane not met.

&& collision condition is CASP_LCAS_BEFORE_LANE_CROSS

manoeuvre_to_hold

&& EnterHoldCount == 0

|| Collision condition is COLLISION_RISK_LEVEL1

hold_to_manoeuvre

&& no collision condition.

&& basic condition met.

&& lane condition met, lane usability is CASP_LCAS_BEFORE_LANE_CROSS or CASP_LCAS_ACTIVATABLE.

takeover_after_pnr

|| Basic condition not met.

|| Collision condition is COLLISION_RISK_LEVEL3.

|| the system spend more time than max_manoeuvre_dur_ms

hold_to_takeover

|| Collision condition is COLLISION_RISK_LEVEL3 && cross over point passed.

|| Collision condition is COLLISION_RISK_LEVEL2 && cross over point passed.

|| Basic condition not met.

|| The system spends more than MAX_HOLD_DURATION_MSseconds && The ego vehicle has passed the cross over point.

end_manoeuvre

&& common_lat_act_data.CASP_lateral_deprt_status_E == CASP_LDS_CENTER_LANE

&& all wheels inside ego lane.

end_procedure

&& maneuver is completed.

&& distance to the center of the ego lane is smaller than max_pos_error_cm

stdby_after_return

|| maneuver is completed.

|| The function remains in LCA_ON:SLCA_INTERRUPTION:RETURN for more than max_return_dur_ms

stdby_after_takeover

|| P1 condition, driver overrule.

|| The function remains in SLCA_ON:SLCA_INTERRUPTION:TAKEOVER_REQUEST for more than overrule_ind_ms.

stdby_after_rejected

|| the system spend more than min_reject_dur_ms

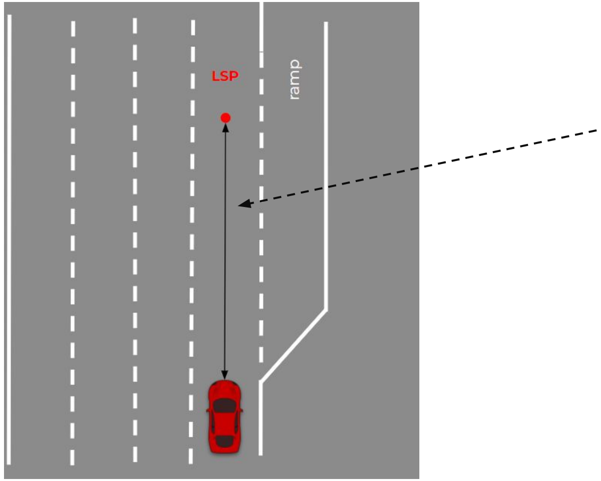
LAAP FLCA Concept

LAAP变更

1. 进入hold时拉平车头
2. 从hold 进mano时重新规划曲线
3. 从hold进interruption时拉平车头
4. 从hold进return时规划返回曲线
5. 规划返回从压线点变成了坐标系返回点
6. 进弯道单侧invalid时向右偏移
7.

TRAC FLCA Concept

Ramp Deceleration Determination



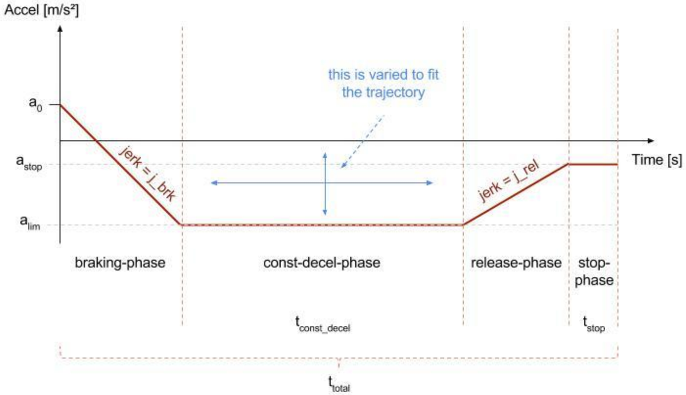
Based on LSP profile, perform Trapezoid Control.
Known conditions:

- v_0 = initial velocity;
- x_0 = initial position;
- a_0 = initial acceleration;
- v_end = ramp speed limitation;
- x_end = LSP distance;
- a_end = 0;

Arbitration:
Actual acceleration = min(follow_deceleration, trpz_in_curve)

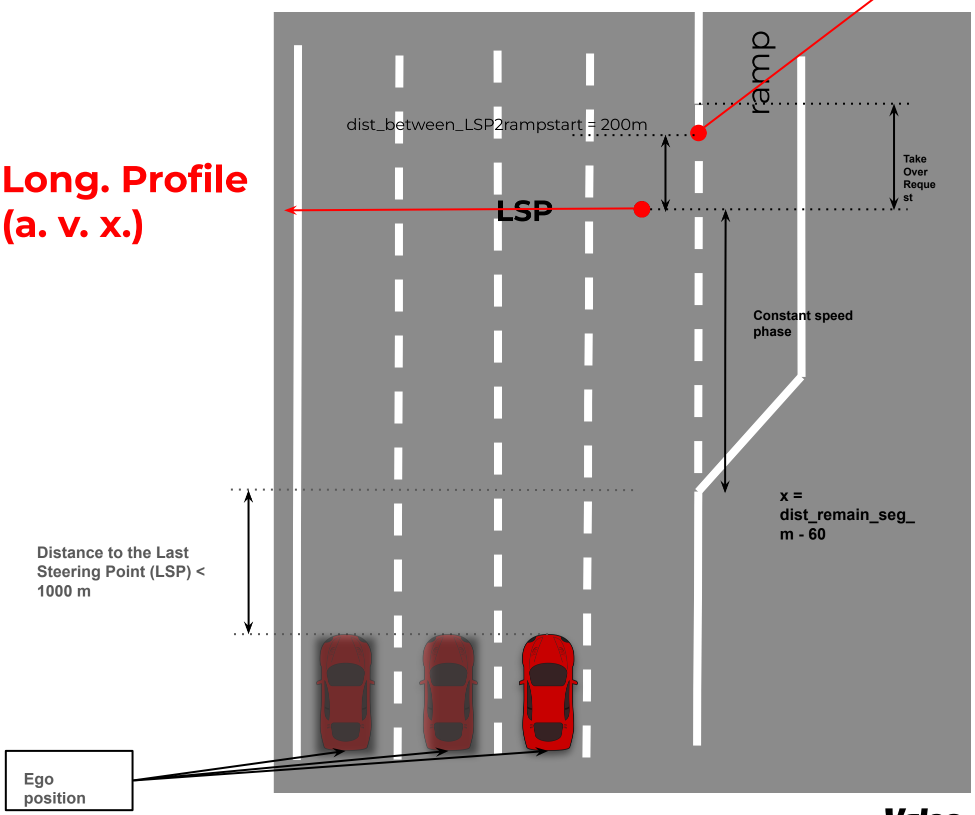
The system shall request a take-over of driver if:

- Bypassed LSP;
- Host vehicle is localized on target path;
- Host vehicle is not on lane change status;



Determine Longitudinal Strategy for Lane Change Manoeuvre

CASP_lc_proposal_S lc_pre_dec_strategy = CASP_PDS_OUTSIDE_RELEVANT



LSP & Long. Profile attributes:
Acceleration: 0
Velocity: major_speed_limitation_low
Distance: dist_remain_seg_m + 200

Pre deceleration strategy:
Ego_target_velocity =
Max(0.5 * spd_limitation_current*(1+dec_factor),40kph)

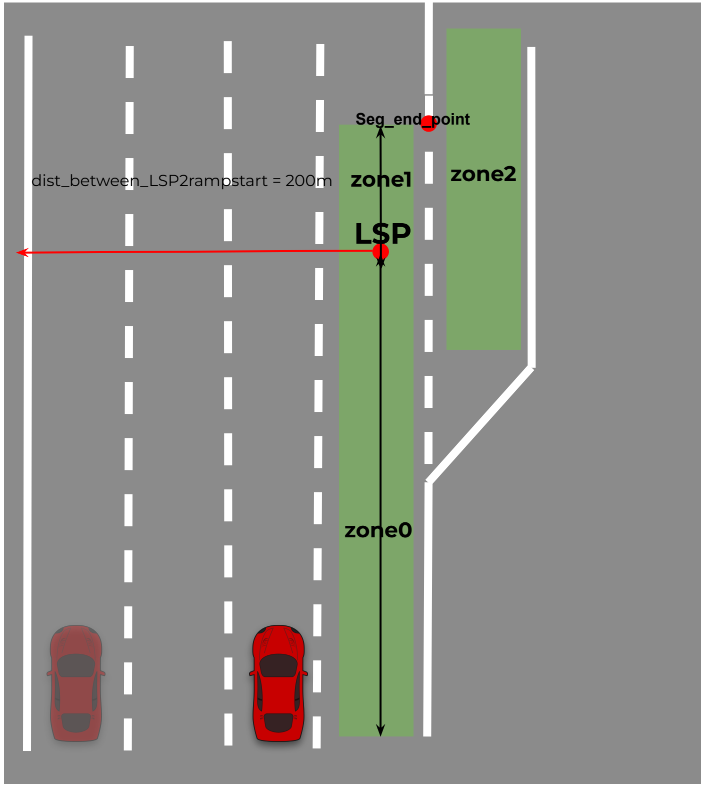
Dec_factor = distance_to_LSP/ 1000m

Determine Longitudinal Strategy for Lane Change Manoeuvre

CASP_lc_proposal_S lc_pre_dec_strategy = CASP_PDS_INSIDE_RELEVANT

The two green lanes are relevant lane

Seg_end_point = Ramp_start_point :dist_remain_seg_m = 0



LSP & Long. Profile attributes:
Acceleration: 0
Velocity: major_speed_limitation_low
Distance: dist_remain_seg_m + 200

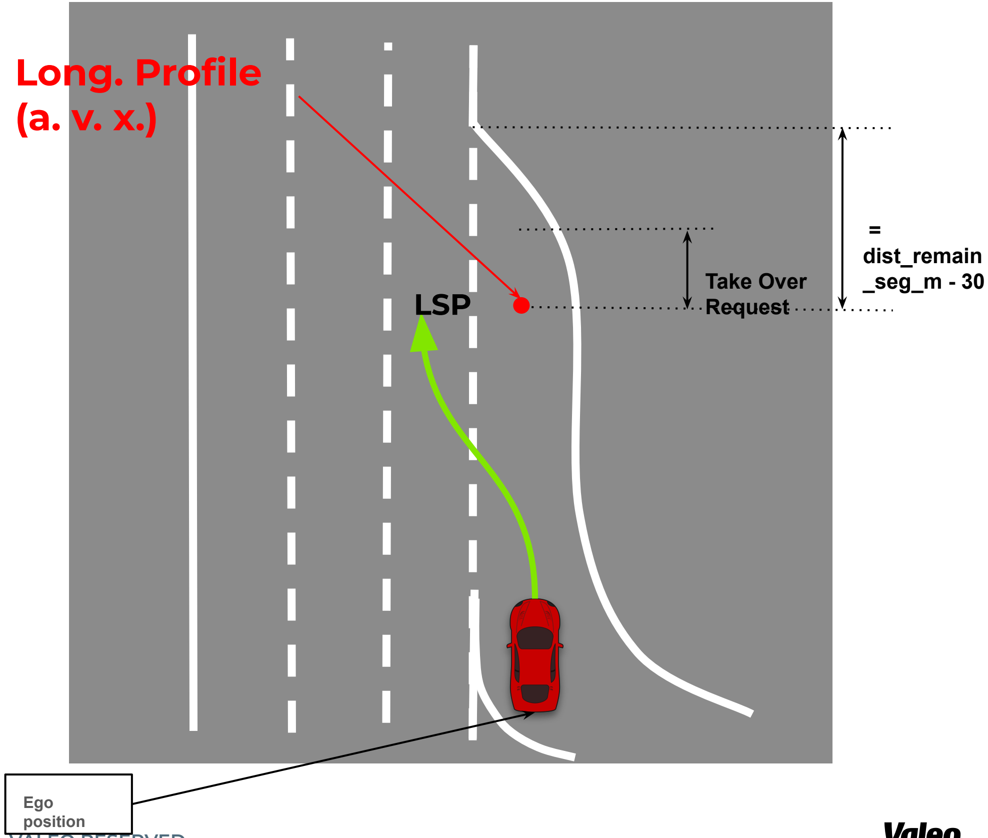
Pre deceleration strategy == CASP_PDS_INSIDE_RELEVANT **:**
zone0:
Ego_target_velocity =
Max(0.5 * spd_limitation_current*(1+dec_factor),40kph)

Dec_factor = distance_to_LSP/ 1000m
zone1:
Ego_target_velocity =
MIN(0.5 * spd_limitation_current*(1+dec_factor),40kph)
Dec_factor = distance_to_LSP/ 1000m
Take over request = TRUE

zone2:
Ego_target_velocity =
MIN(spd_limitation,40kph)

Determine Longitudinal Strategy for Lane Change Manoeuvre

CASP_lc_proposal_S lc_pre_dec_strategy = CASP_PDS_INSIDE_RAMP



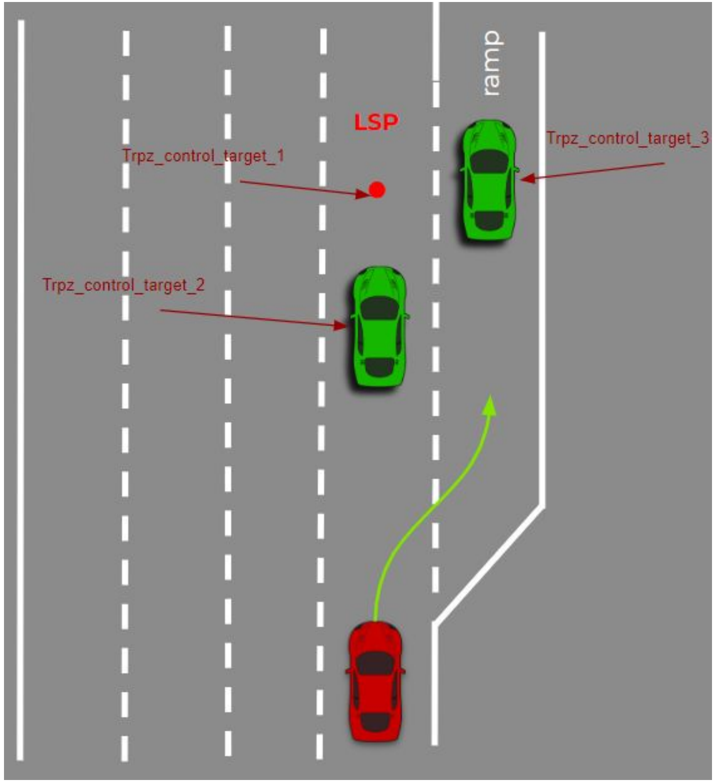
LSP attributes:
Acceleration: 0
Velocity: 0
Distance: dist_remain_seg_m - 69*

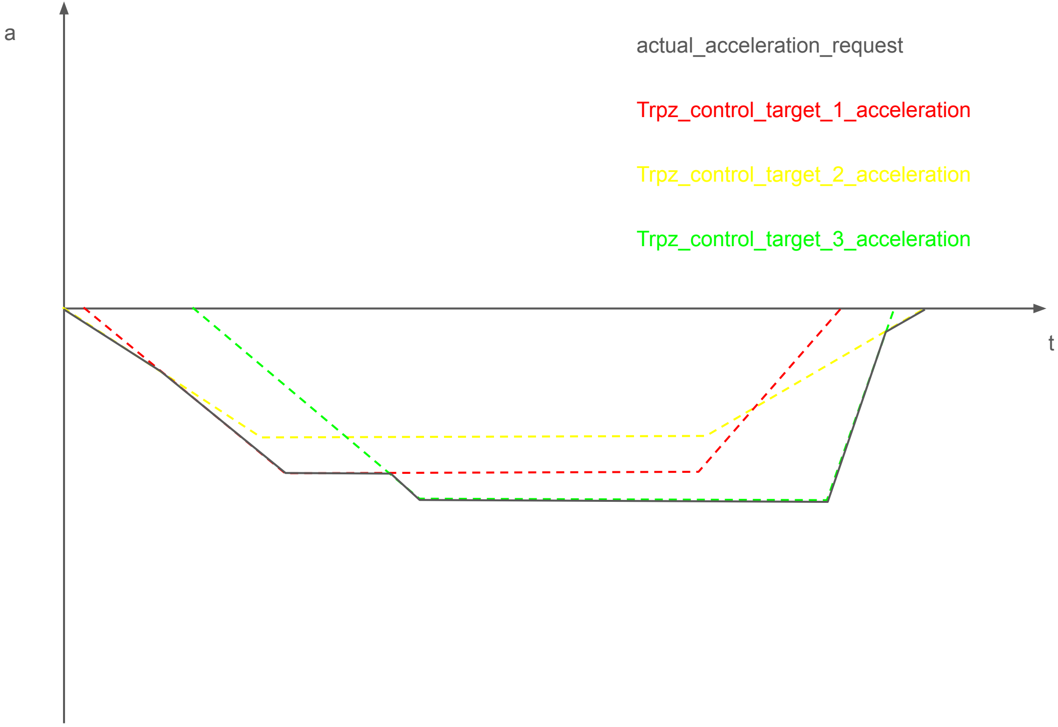
Trpz: V_init = 60kph = 16.67m/s
V_end = 0 m/s
a_lim = -2 m/s^2 (for comfort purpose)
Therefore, dist = 69m

Long. Profile:
Acceleration: 0
Velocity: highway_speed_limitation
Distance: dist_remain_seg_m - 69*

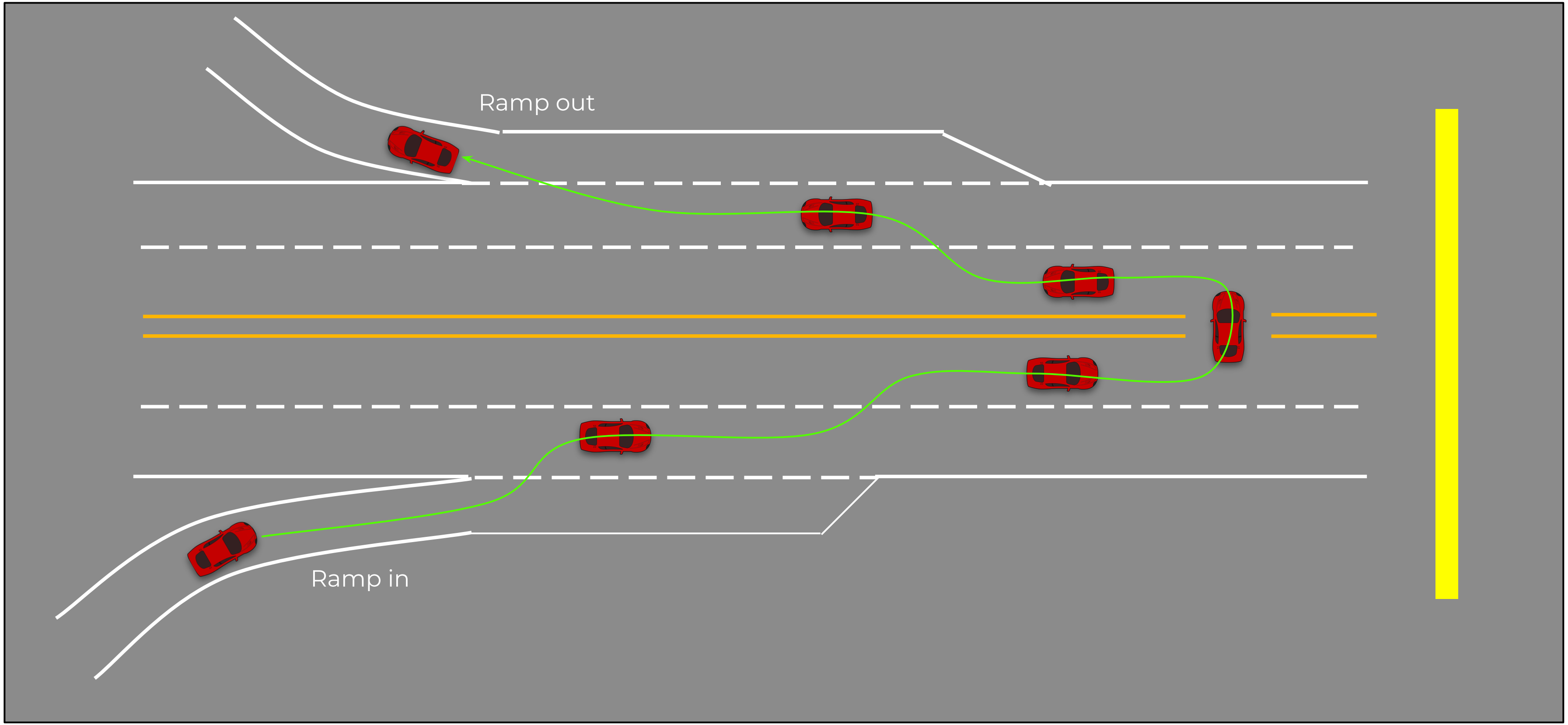
FLCA Deceleration Arbitration

CASP_lc_proposal_S lc_pre_dec_strategy != 4

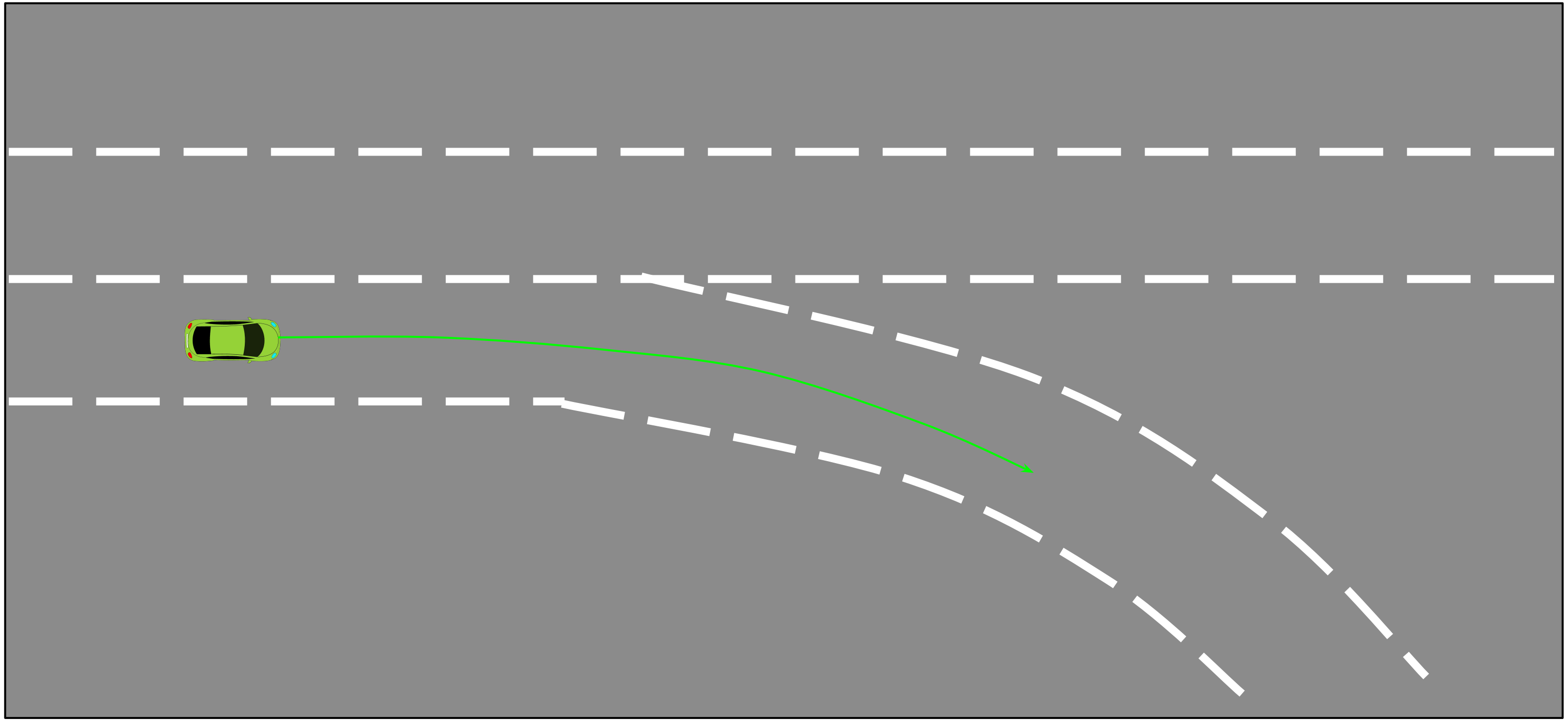




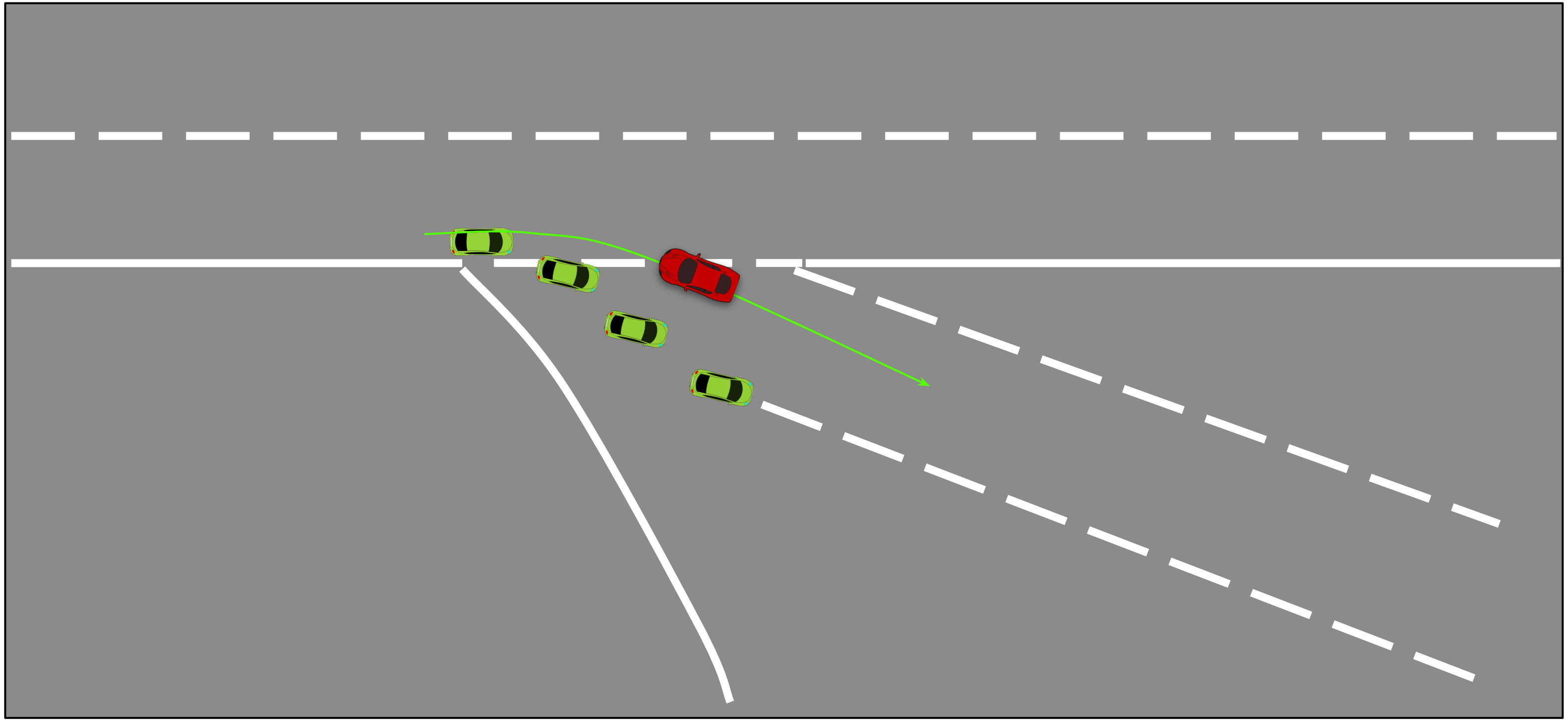
FLCA Deceleration Arbitration



FLCA Deceleration Arbitration



FLCA Deceleration Arbitration



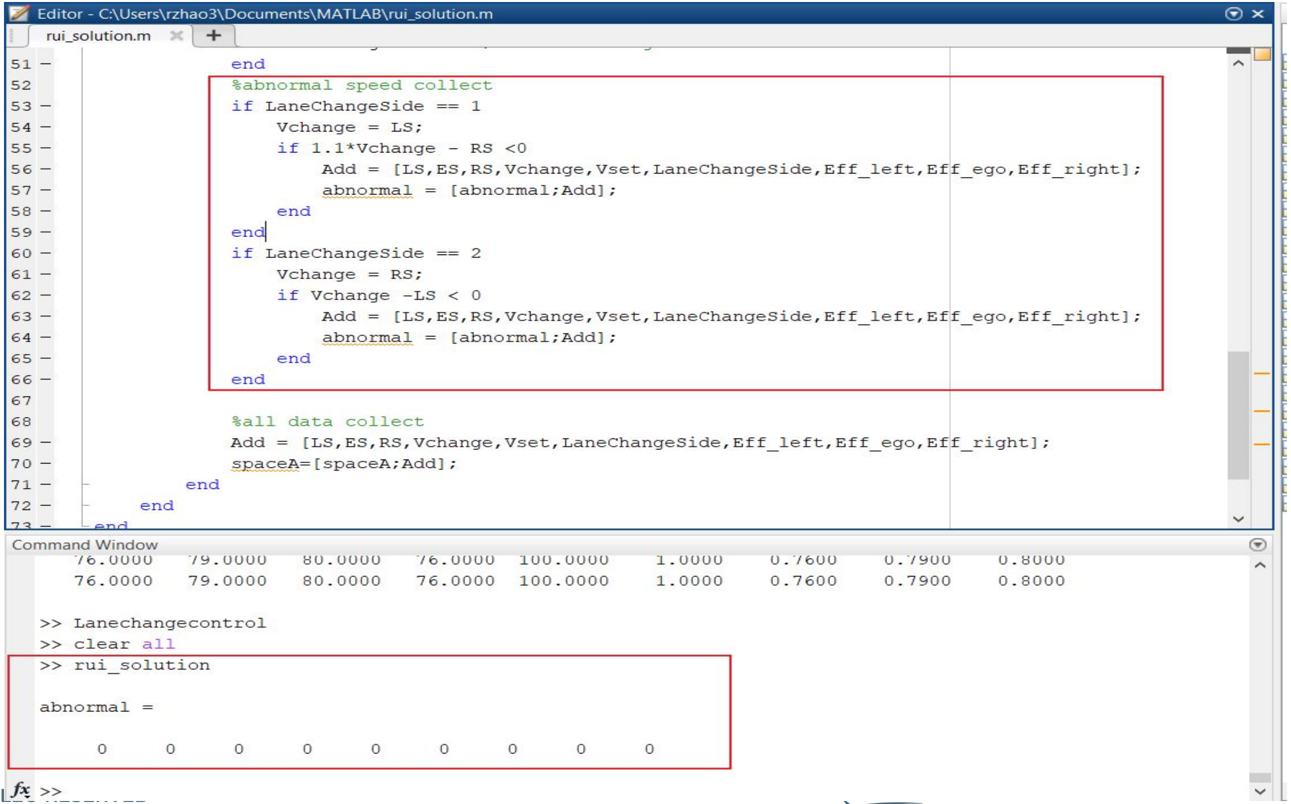
ALCD FLCA Concept

Perceive Vehicles

- ALCD shall compute the value of ObjV as an input of LCPR.
- Object select for ObjV:
  - The object is defined as the closest object in front of the ego vehicle.
  - The object should be considered if the longitudinal distance between the object and the ego is larger than *allowed_dis_cm*.
  - *allowed_dis_cm* should be a look-up table based on the ego velocity to avoid mis-selection of long distance object.
  - The object selection should be compensated with detected road curvature to avoid mis-selection of object in curve.

Concept Verification

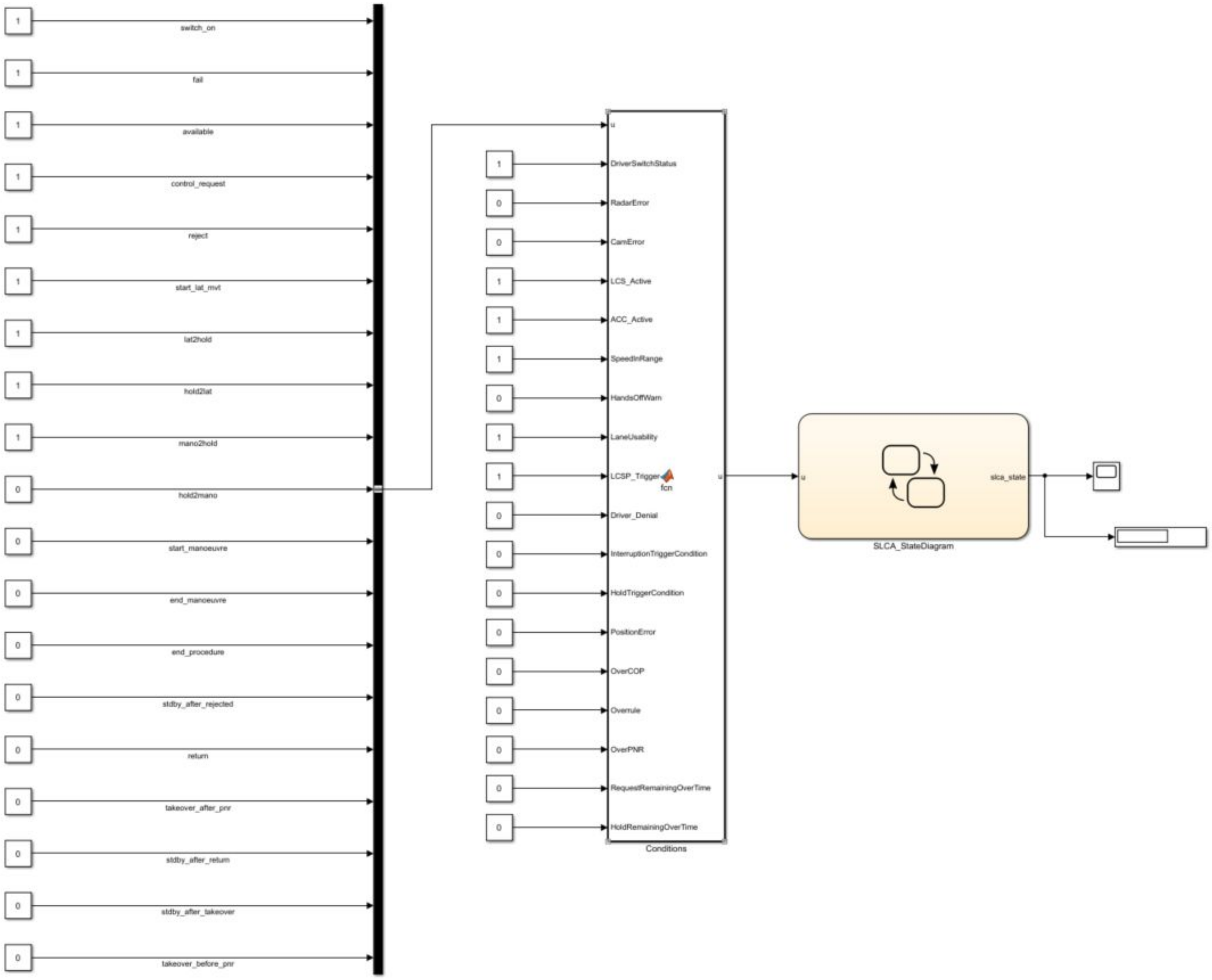
Concept Verification



```
            end
            %abnormal speed collect
            if LaneChangeSide == 1
                Vchange = LS;
                if 1.1*Vchange - RS <0
                    Add = [LS,ES,RS,Vchange,Vset,LaneChangeSide,Eff_left,Eff_ego,Eff_right];
                    abnormal = [abnormal;Add];
                end
            end
            if LaneChangeSide == 2
                Vchange = RS;
                if Vchange -LS < 0
                    Add = [LS,ES,RS,Vchange,Vset,LaneChangeSide,Eff_left,Eff_ego,Eff_right];
                    abnormal = [abnormal;Add];
                end
            end

            %all data collect
            Add = [LS,ES,RS,Vchange,Vset,LaneChangeSide,Eff_left,Eff_ego,Eff_right];
            spaceA=[spaceA;Add];
        end
    end
end
```

```
   76.0000   79.0000   80.0000   76.0000  100.0000    1.0000    0.7600    0.7900    0.8000
   76.0000   79.0000   80.0000   76.0000  100.0000    1.0000    0.7600    0.7900    0.8000

>> Lanechangecontrol
>> clear all
>> rui_solution

abnormal =

     0     0     0     0     0     0     0     0     0

>>
```

Validation Script


1
switch_on
1
fail
1
available
1
control_request
1
reject
1
start_lat_mvt
1
lat2hold
1
hold2lat
1
mano2hold
0
hold2mano
0
start_manoeuvre
0
end_manoeuvre
0
end_procedure
0
stdby_after_rejected
0
return
0
takeover_after_pnr
0
stdby_after_return
0
stdby_after_takeover
0
takeover_before_pnr
u
1
DriverSwitchStatus
0
RadarError
0
CamError
1
LCS_Active
1
ACC_Active
1
SpeedInRange
0
HandsOffWarn
1
LaneUsability
1
LCSP_Trigger
fcn
u
0
Driver_Denial
0
InterruptionTriggerCondition
0
HoldTriggerCondition
0
PositionError
0
OverCOP
0
Overrule
0
OverPNR
0
RequestRemainingOverTime
0
HoldRemainingOverTime
Conditions
u
slca_state
SLCA_StateDiagram

FAIL = 0xB

HOLD = 0xA

Finish = 0x9

TAKEOVER = 0x8

RETURN = 0x7

REJECTED = 0x6

MANO = 0x5

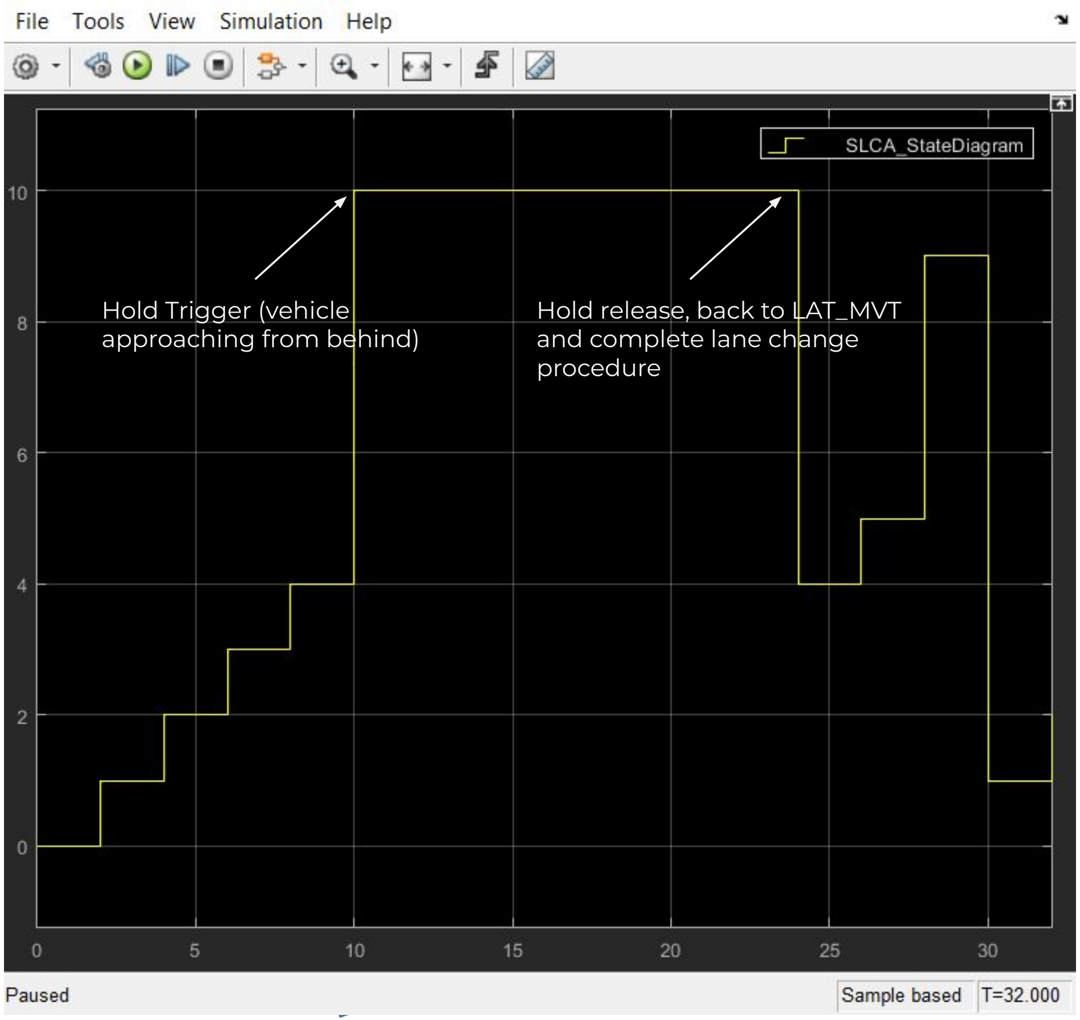
LAT_MVT = 0x4

REQUEST = 0x3

Standby/available = 0x2

Standby/Unavailable = 0x1

OFF= 0x0



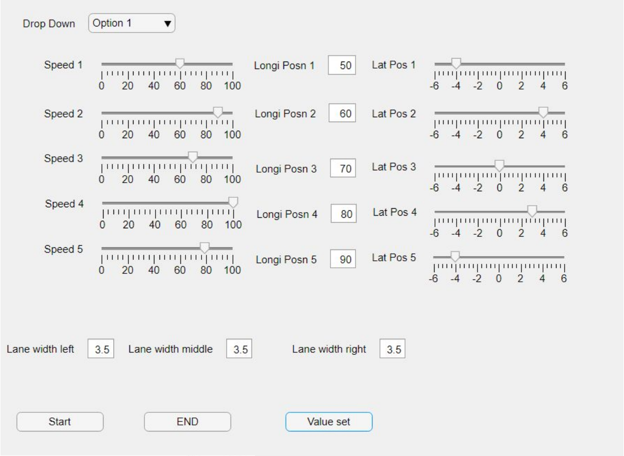
Hold Trigger (vehicle approaching from behind)

Hold release, back to LAT_MVT and complete lane change procedure

Concept Simulation

Simulation animation, developed on MATLAB
This GIF indicates:

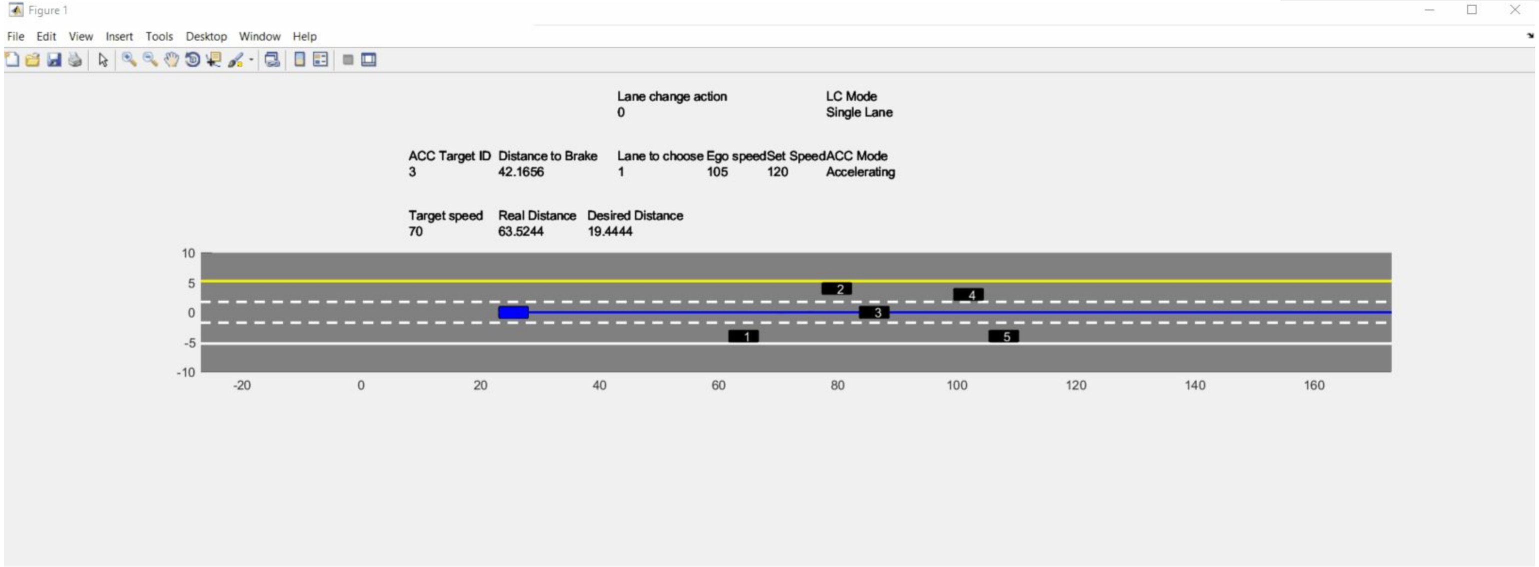
- lane change direction (1 for left, 2 for right),
- cruise mode(lane centering or lane change),
- available side to change lane (1 for left, 2 for right),
- object speed, set speed and current speed



Drop Down: Option 1

Speed 1 | Longi Posn 1: 50 | Lat Pos 1
Speed 2 | Longi Posn 2: 60 | Lat Pos 2
Speed 3 | Longi Posn 3: 70 | Lat Pos 3
Speed 4 | Longi Posn 4: 80 | Lat Pos 4
Speed 5 | Longi Posn 5: 90 | Lat Pos 5

Lane width left: 3.5 | Lane width middle: 3.5 | Lane width right: 3.5



Lane change action	LC Mode
0	Single Lane

ACC Target ID	Distance to Brake	Lane to choose	Ego speed	Set Speed	ACC Mode
3	42.1656	1	105	120	Accelerating

Target speed	Real Distance	Desired Distance
70	63.5244	19.4444

Config Panel
Adjust object speed, set speed

FLCA-overtaking on embedded ECU

Link

- FLCA function development history
- FLCA Sensor set
- FLCA Stakeholder Req. Spec.
- FLCA Stkhd req template
- FLCA Sys. Req. Spec.
- FLCA Cal & Cfg Parameter
- FLCA ALFU PTF change
- FLCA hold standazation review
- FLCA Use Case
- FLCA Decision and Planning Design