Learning Critical Scenarios in Feedback Control Systems for Automated Driving

Mengjia Zhu[1], Alberto Bemporad[1], Maximilian Kneissl[2], Hasan Esen[2]

ITSC 2023

[1]IMT School for Advanced Studies Lucca

[2]Collaboration project with DENSO

September 28, 2023

IEEE ITSC 2023 BILBAO

IMT SCUOLA ALTI STUDI LUCCA

DENSO Crafting the Core

Overview

- **Project goals**
- **Problem formulation and solution strategy**
- **Case studies**
- **Open questions and discussions**

Project goals

- Detect **undesired simulation scenarios** (= **corner-cases**) for controller validation in safety-critical automated driving (AD) applications to support controller design choices

Project goals

- Detect **undesired simulation scenarios** (= **corner-cases**) for controller validation in safety-critical automated driving (AD) applications to support controller design choices

- Reduce verification and validation (V&V) effort by using **scenario-based** method with **global optimization** as the exploration method (sampler)

Terminologies

- A **scenario** is described by a vector x_{ODD} (**operational design domain**) $\subseteq \mathbb{R}^n$ of parameters.

  Definition: An ODD provides the set of conditions under which the AD system is designed to **function**.

- $x_{\mathrm{scene}} \in x_{\mathrm{ODD}}$ = set of meaningful scenario parameters to consider

  Examples: initial distance between the SV and OVs, acceleration of the OV,...

- **Critical scenario** = vector x_{scene} for which closed-loop behavior is critical

  Examples: time-to-collision is too short, excessive jerk of the SV, ...

Critical-case generation

Key ideas:

- Formulate the critical scenarios identification problem as an **optimization problem**
  - Provide a holistic problem formulation
  - Considers an ODD description
- Generate critical scenarios by minimizing an objective function $f_{system} : \mathbb{R}^n \mapsto \mathbb{R}$
  - Use global optimizer GLIS to generate critical corner-cases

Problem formulation

Optimization problem:

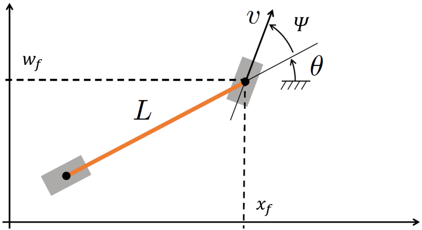
$$\begin{aligned} x^*_{\text{scene}} \in \arg\min_{x_{\text{scene}}} \quad & f_{\text{system}}(x_{\text{scene}}) \\ \text{s.t.} \quad & \ell \leq x_{\text{scene}} \leq u \\ & x_{\text{scene}} \in \chi, \end{aligned} \tag{1}$$

- $f_{\mathrm{system}} : \mathbb{R}^n \mapsto \mathbb{R}$ is the objective function to **minimize**
  - Criticality of closed-loop simulation (or experiment) determined by scenario x_{scene}
    - the smaller $f(x)$, the more critical x_{scene} is
  - Known or pre-designed
- $x_{\mathrm{scene}} \in x_{\mathrm{ODD}} \subseteq \mathbb{R}^n$ is the vector of parameters to be **optimized**
- $\ell, u \in \mathbb{R}^n$: vectors of lower and upper bounds on x_{scene}
- $\chi \in \mathbb{R}^n$: other arbitrary constraints on x_{scene} (Known)

Case study

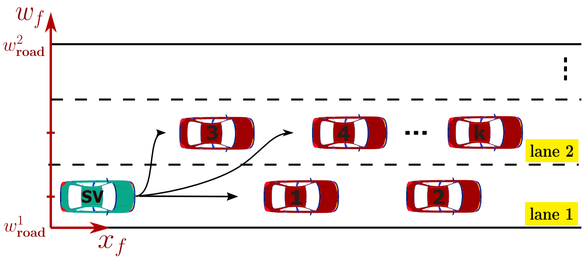
- **Problem**: find critical scenarios in automated driving w/ obstacles
- **MPC controller** for lane-keeping and obstacle-avoidance based on simple kinematic bicycle model (Zhu, Piga, and Bemporad, 2021)

$$\dot{x}_f = v\cos(\theta + \psi)$$
$$\dot{w}_f = v\sin(\theta + \psi)$$
$$\dot{\theta} = \frac{v\sin(\psi)}{L}$$
(x_f, w_f) = front-wheel position

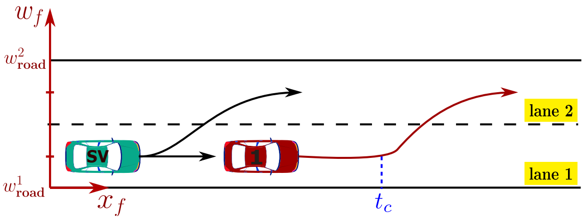


Case studies

Logical Scenario 1:



Logical Scenario 2:

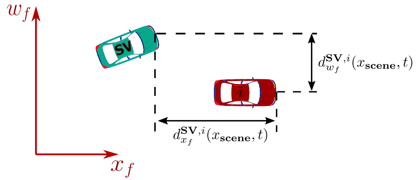


Optimization problem

Black-box optimization problem: given k obstacles, solve

$$\min_{x_{\text{scene}} \in x_{\text{ODD}}} \sum_{i=1,\ldots,k} d^{\text{SV},i}_{x_f,\text{critical}}(x_{\text{scene}}) + d^{\text{SV},i}_{w_f,\text{critical}}(x_{\text{scene}})$$

$$\text{s.t.} \quad \ell \leq x_{\text{scene}} \leq u \;\; \& \text{ other constraints}$$



$$\text{where} \quad d^{\text{SV},i}_{x_f,\text{critical}}(x_{\text{scene}}) = \begin{cases} \min\limits_{t \in T_{\text{collision}}} d^{\text{SV},i}_{x_f}(x_{\text{scene}}, t) & \mathcal{I}^i_{\text{collision}} \\ L & \sim \mathcal{I}^i_{\text{collision}} \& \mathcal{I}_{\text{collision}} \\ \sum\limits_{t \in T_{\text{sim}}} d^{\text{SV},i}_{x_f}(x_{\text{scene}}, t) & \sim \mathcal{I}_{\text{collision}} \end{cases} \quad \begin{matrix} \text{min time of collision with } \#i \\ \text{collision with other } \#j \neq \#i \\ \text{no collision} \end{matrix}$$

$$d^{\text{SV},i}_{w_f,\text{critical}}(x_{\text{scene}}) = \begin{cases} \min\limits_{t \in T_{\text{collision}}} d^{\text{SV},i}_{w_f}(x_{\text{scene}}, t) & \mathcal{I}^i_{\text{collision}} \\ w_{f,\text{safe}} & \sim \mathcal{I}^i_{\text{collision}} \& \mathcal{I}_{\text{collision}} \\ \sum\limits_{t \in T_{\text{sim}}} d^{\text{SV},i}_{w_f}(x_{\text{scene}}, t) & \sim \mathcal{I}_{\text{collision}} \end{cases} \tag{2}$$

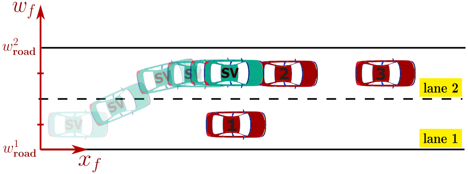
$$\mathcal{I}^i_{\text{collision}} = \text{True, if } \exists t \in T_{\text{sim}}, \text{ s.t. } \; (d^{\text{SV},i}_{x_f}(x_{\text{scene}}, t) \leq L) \;\&\; (d^{\text{SV},i}_{w_f}(x_{\text{scene}}, t) \leq W),$$

$$\mathcal{I}_{\text{collision}} = \text{True, if } \exists h \in \{1, \ldots, k\}, \text{ s.t. } \; \mathcal{I}^h_{\text{collision}} = \text{True}.$$

Results and discussions

Logical scenario 1 - Test 2: GLIS identifies 64 collision cases within 100 simulations

Iter	x_{scene}					
	x_{f1}^0	v_1^0	x_{f2}^0	v_2^0	x_{f3}^0	v_3^0
51	15.00	30.00	44.14	10.00	49.10	47.39
79	28.09	30.00	70.29	10.00	74.79	31.74
40	34.30	30.00	60.59	10.00	77.80	35.97



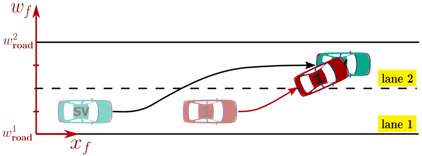
Collision triggering conditions and discussions

- 1) SV change lane to avoid OV_1; 2) SV cannot brake fast enough to avoid OV_2
- To avoid OV_2, lane change is not an option for SV (OV_1 blocks the way)
- **Critical x_{scene}**:
  - A relatively large x_{f1}^0 coupled with a relatively slow v_1^0
    - The smaller x_{f1}^0, the greater v_1^0
  - A slow v_2^0 with a large x_{f2}^0

Results and discussions

Logical scenario 2: GLIS identifies 9 collision cases within 100 simulations

Iter	x_{scene}		
	x_{f1}^0	v_1^0	t_c
28	12.57	46.94	16.75
16	17.53	47.48	23.65
88	44.54	41.26	16.02



Collision triggering conditions and discussions

- **Critical** x_{scene}:
  - a combination of a relatively large x_{f1}^0 with a relatively small v_1^0 and a $t_c < t_{\text{exp}}$
  - a larger x_{f1}^0 is coupled with either a smaller v_1^0 or a lager t_c or both
- 1) SV changes lane to avoid OV_1;
  2) SV collides with OV_1 after t_c (during lane-changing of OV_1)
- SV do not have enough response time to decelerate for the sudden lane-changing of OV_1

Conclusion

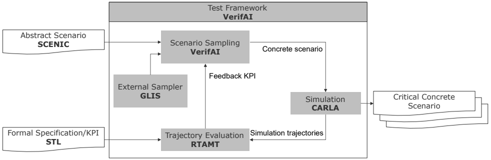
- The global opt. framework can effectively determine safety-critical test scenarios
  - based on learning a surrogate model of the criticality function
- The collision triggering conditions can be found by analyzing the identifed critical test scenarios
- The information synthesized from the critical cases can then be used to
  - refine the ODD definitions **AND/OR**
  - upgrade the design of the system

Challenges with the current approach

The design of the objective function

- It is often based on multiple criteria
- Its formulation can be hard to determine beforehand

Possible solutions: Integrate with RTAMT monitors, see (Molin et al, 2023)



Thank you!

Questions?

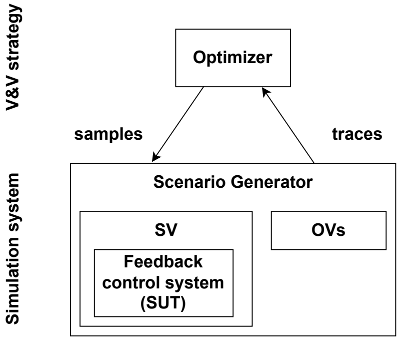
Contact info: mengjia.zhu@imtlucca.it

Complementary slides

Summary

Goal: Test the applicability of a **designed feedback control system** (System Under Test, SUT) in an AD vehicle

- Specifically, we consider a subject vehicle (SV) actuated by the given controller for
  - lane keeping & collision avoidance with obstacle vehicles (OVs)
- Reduce test efforts: use a **systematic** way to efficiently identify **test scenarios**



V&V strategy

- Search-based testing framework
- Exploration method (sampler): **learning-based optimization**

Notes on the optimization problem

Objective function f_{system}:

- A single assessing criterion **OR**
- A weighted combination of different criteria
- A closed form expression of f_{system} with x_{scene} is often **NOT** available
  - Due to the complex way the level of criticality of the system depends on the variables in x_{scene}
  - But f_{system} can be **evaluated** through real experiments or simulations

Solution strategy:

- Surrogate-based optimization methods are suitable to solve (1)
- For this project, global optimization algorithm GLIS (Bemporad, 2020) is used
  - Benefits: easy incorporation of constraints and cheap computational cost
  - Alternatives: Bayesian optimization (Brochu et al, 2010), ...

Case study - MPC controller

- Let us describe the model in general nonlinear multi-input multi-output form

$$\dot{x} = f(x, u)$$
$$y = g(x, u),$$

- **Linear time-varying (LTV)** MPC strategy, with constant sampling time T_s (Diehl, Bock, Schlöder, 2005; Gros et al, 2020):

$$\tilde{x}_{j+1} = A_j \tilde{x}_j + B_j \tilde{u}_j$$
$$\tilde{y}_j = C_j \tilde{x}_j + D_j \tilde{u}_j,$$

- At each sample t, compute the MPC action $u_{t|t}$ by solving a **quadratic problem (QP)**

$$\min_{\{u_{t+j|t}\}_{j=0}^{N_u-1}, \varepsilon} \sum_{j=0}^{N_p-1} \left\| y_{t+j|t} - y_{t+j}^{\mathrm{ref}} \right\|_{Q_y}^2 + \sum_{j=0}^{N_p-1} \left\| u_{t+j|t} - u_{t+j}^{\mathrm{ref}} \right\|_{Q_u}^2 + \sum_{j=0}^{N_p-1} \left\| \Delta u_{t+j|t} \right\|_{Q_{\Delta u}}^2$$

- Finely-tuned MPC parameters already calibrated and fixed

Case study -MPC controller

Discrete-time state-space model for the case study:

$$\tilde{s}_{j+1} = \begin{bmatrix} 1 & 0 & -\bar{v}_j \sin(\bar{\theta}_j + \bar{\psi}_j) T_s \\ 0 & 1 & \bar{v}_j \cos(\bar{\theta}_j + \bar{\psi}_j) T_s \\ 0 & 0 & 1 \end{bmatrix} \tilde{s}_j + \begin{bmatrix} \cos(\bar{\theta}_j + \bar{\psi}_j) T_s & -\bar{v}_j \sin(\bar{\theta}_j + \bar{\psi}_j) T_s \\ \sin(\bar{\theta}_j + \bar{\psi}_j) T_s & \bar{v}_j \cos(\bar{\theta}_j + \bar{\psi}_j) T_s \\ \frac{\sin(\bar{\psi}_j)}{L} T_s & \frac{\bar{v}_j \cos(\bar{\psi}_j)}{L} T_s \end{bmatrix} \tilde{u}_j$$

$$\tilde{y}_j = \tilde{s}_j,$$

- The subscript $_j$ denotes the value at time step j
- Nominal trajectory: $\bar{s}_j = [\bar{x}_{f_j}\ \bar{w}_{f_j}\ \bar{\theta}_j]'$, $\bar{u}_j = [\bar{v}_j\ \bar{\psi}_j]'$, and $\bar{y}_j = \bar{s}_j$
- $\widetilde{\text{Var}} = \text{Var} - \overline{\text{Var}}$ denotes the deviation from the nominal value

GLIS algorithm

Two stages: Initial sampling & Active learning (Bemporad, 2020)

1. Collect N_{init} initial samples
   $\{(x_{\text{scene}}^1, f_{system}^1), (x_{\text{scene}}^2, f_{system}^2), \ldots, (x_{\text{scene}}^{N_{\text{init}}}, f_{system}^{N_{\text{init}}})\}$

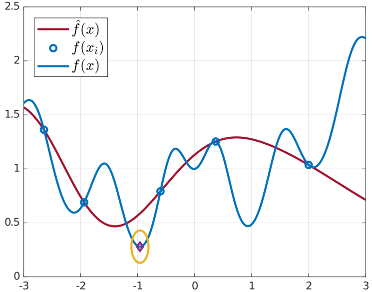
2. Build a **surrogate function**

$$\hat{f}(x_{\text{scene}}) = \sum_{i=1}^{N} \alpha_i \phi(\left\|x_{\text{scene}} - x_{\text{scene}}^i\right\|_2)$$

$\phi =$ radial basis function

Example: $\phi(d) = \frac{1}{1+(\epsilon d)^2}$
(*inverse quadratic*)

true $f(x_{\text{scene}})$
surrogate $\hat{f}(x_{\text{scene}})$



Note: just minimizing $\hat{f}(x_{\text{scene}})$ to find x_{scene}^{N+1} may easily miss the global optimum

GLIS Algorithm: exploration vs. exploitation

(Bemporad, 2020)

3. Construct the IDW **exploration function**

$$z(x_{\text{scene}}) = \frac{2}{\pi}\Delta F \tan^{-1}\left(\frac{1}{\sum_{i=1}^{N} w_i(x_{\text{scene}})}\right)$$

where $w_i(x_{\text{scene}}) = \frac{e^{-\left\|x_{\text{scene}}-x^i_{\text{scene}}\right\|^2}}{\left\|x_{\text{scene}}-x^i_{\text{scene}}\right\|^2}$

4. Optimize the **acquisition function**:

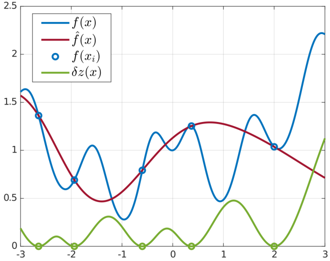
$$x_{\text{scene}N+1} = \arg\min_{\substack{x_{\text{scene}} \in x_{ODD} \\ \ell \le x_{\text{scene}} \le u; \\ x_{\text{scene}} \in \mathcal{X}}} \hat{f}(x_{\text{scene}}) - \delta z(x_{\text{scene}})$$

to get the **query point** x^{N+1}_{scene}.

5. Test the case with x^{N+1}_{scene}, measure f^{N+1}.

6. Iterate the procedure for $N+2, N+3 \ldots$.

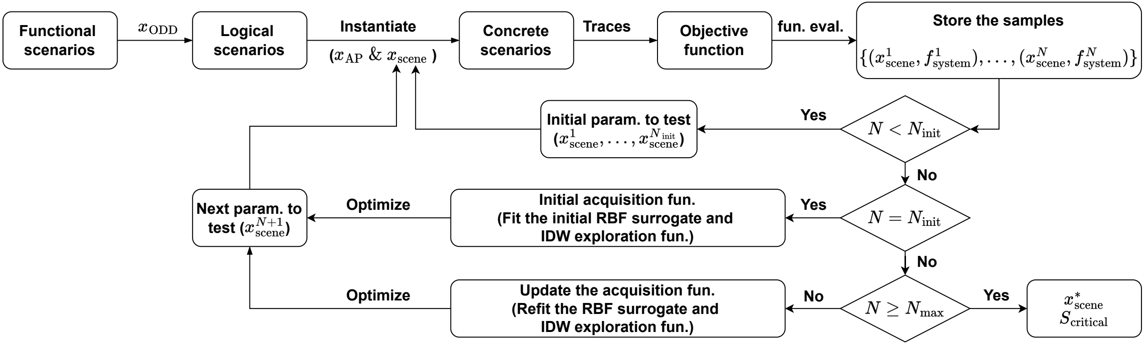
Exploration function $z(x_{\text{scene}})$



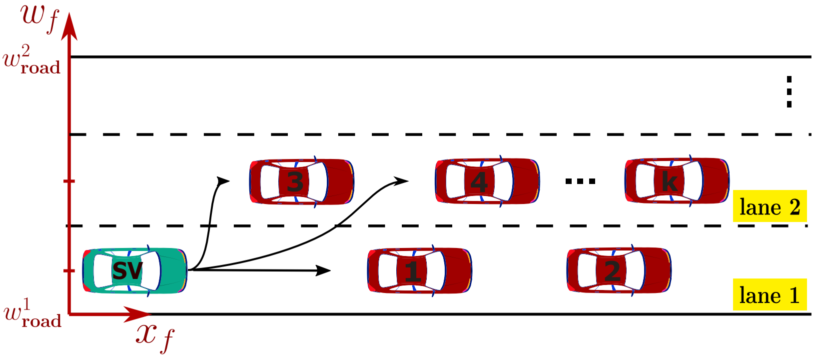
δ = **exploitation vs. exploration** trade-off parameter

GLIS Algorithm - Summary

GLIS: **active** sampler to find x_{scene} that leads to critical behaviors of the closed-loop system

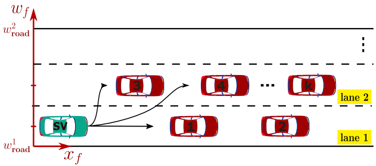


Case studies - Logical Scenario 1



- ODD description
- Optimization problem
- Numerical tests
- Results and discussions

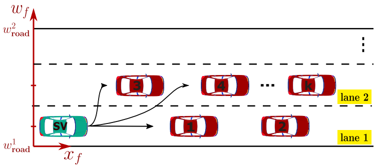
ODD description[1]:



- Two or more vehicles on a one-way horizontal road with two or more lanes
  - **AP**:# of lanes, road width, vehicle dimensions, experiment duration
- The obstacle vehicles (OVs) (1, 2, 3,...,k): on any lane, ahead or behind subject vehicle, move forward horizontally with a constant speed (**NO** collision among them)
  - **AP**: # of OVs, their initial lateral position and constant yaw angle
  - **PoI**: their initial longitudinal position (x^0_{fi}) and initial velocity (v^0_i)
- The subject vehicle (SV): commanded by a MPC controller to avoid collision (when within safety distance with any OV, change lane, decelerate or accelerate depend on the relative position and conditions (discussed in the following slides))
  - **AP**: its initial longitudinal & lateral position, reference velocity and reference yaw angle; safety distances (longitudinal & lateral)
- MPC controller: command the SV, the controller under testing
  - **AP**: MPC parameters; **Note**: constraints are adaptive to **PoI**

[1] AP: Assumed Parameters; PoI: Parameter of Interest

Case study - Logical Scenario 1



Dimensions and Exp. duration:

- Road width: 6 m total, 2 lanes (3 m/lane)
- Vehicle dim (SV & OVs): $L = 4.5$ m, $W = 1.8$ m
- Experiment duration: $t_{\text{exp}} = 30$ s

Safety distance:

- longitudinal ($x_{f,\text{safe}}$): 10 m, lateral ($w_{f,\text{safe}}$): 3 m

Initial conditions:

- SV: (0, 0) m, 50 km/h, $\theta^{\text{SV},0} = 0°$
- OV: $x_{\text{scene}} = [x^0_{f1}, v^0_1, ...x^0_{fk}, v^0_{fk}]$, k: # of obstacles (AP), $\theta^0_i = 0°$, for $i = 1, \dots, k$

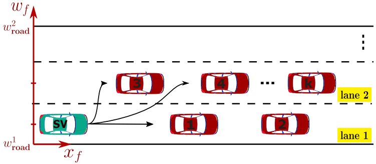
MPC parameters:

- T_s = 0.085 s, N_u = 3, N_p = 23; Q_y = diag(0, 10, 1), Q_u = diag(1, 1), $Q_{\Delta u}$ = diag(1, 0.5)

Constraints and references (fixed):

- $v^{\text{SV}} \in$ [1, 90] km/h, $\dot{v}^{\text{SV}} \in$ [-4, 4] m/s^2, with v^{SV} = 50 km/h
- $\psi^{\text{SV}} \in [-45, 45]°$, $\dot{\psi}^{\text{SV}} \in$ [-60, 60]°/s
- $w_f^{\text{SV}} \in$ [-0.6, 3.6] m, $x_f^{\text{SV}} \in [-\infty, \infty]$ m

Case study - Logical Scenario 1



Constraints and references (adaptive):

FOR $i = 1, \dots, k$, **IF** SV and OV$_i$ are on the same lane and within safety distances (both longitudinal and lateral) **THEN**

IF (OV$_i$ is ahead of SV) && (no collision between SV and OV$_i$ will happen in the next step with the current velocity) && (OV$_j$, $\forall j \neq i, i, j = 1, \dots, k$ are out of safety longitudinal and lateral distances) **THEN**:

Decision: Change lane;
Update:

$\min w_f^{SV} = w_{fi} + w_{f,\text{safe}}$ **IF** change from lower lane to higher lane; **OR**

$\max w_f^{SV} = w_{fi} - w_{f,\text{safe}}$ **IF** change from higher lane to lower lane;

(Note: 'lower' and 'higher' here refer to the relative lateral position of SV w.r.t OV$_i$)

ELSE

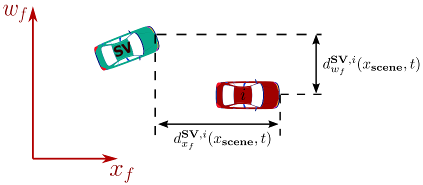
Decision: Decelerate or Accelerate;
Update:

$\min x_f^{SV} = x_{fi} + 1.1L$ **IF** OV$_i$ is behind of SV; **OR**

$\max x_f^{SV} = x_{fi} - 1.1L$ **IF** OV$_i$ is ahead of SV;

Optimization problem

Discussion:



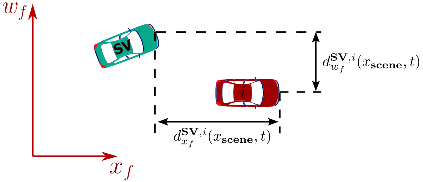
$d_{x_f,\text{critical}}^{\text{SV},i}(x_{\text{scene}}) = L$ **IF** $\sim \mathcal{I}_{\text{collision}}^{i} \,\&\, \mathcal{I}_{\text{collision}}$

$d_{w_f,\text{critical}}^{\text{SV},i}(x_{\text{scene}}) = w_{f,\text{safe}}$ **IF** $\sim \mathcal{I}_{\text{collision}}^{i} \,\&\, \mathcal{I}_{\text{collision}}$

- **Constant** values are assigned to the critical longitudinal and lateral distances of OV_i, when collision happen between SV and OV_j, where $j \neq i$
  - *i.e.*, $\mathcal{I}_{\text{collision}} = 1 \;\&\&\; \mathcal{I}_{\text{collision}}^{i} = 0$
- **Reasoning:** under this condition, the magnitude of the corresponding distance is **irrelevant** w.r.t **criticality** (collision occurence in this case).

Optimization problem

Discussion:



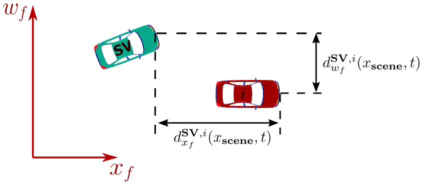
$$d_{x_f,\text{critical}}^{\text{SV},i}(x_{\text{scene}}) = \sum_{t \in T_{\text{sim}}} d_{x_f}^{\text{SV},i}(x_{\text{scene}}, t) \quad \textbf{IF} \quad \sim \mathcal{I}_{\text{collision}}$$

$$d_{w_f,\text{critical}}^{\text{SV},i}(x_{\text{scene}}) = \sum_{t \in T_{\text{sim}}} d_{w_f}^{\text{SV},i}(x_{\text{scene}}, t) \quad \textbf{IF} \quad \sim \mathcal{I}_{\text{collision}}$$

- **Sum** of its longitudinal and lateral distances at every time step are assigned to the critical longitudinal and lateral distances, when collision **DOES NOT** happen between **ANY** SV and OV$_i$, for $i = 1, \ldots, k$
  - *i.e.*, $\mathcal{I}_{\text{collision}} = 0$
- **Reasoning:** under this condition, minimizing the distances between SV and each OV$_i$ throughout the experiments **increases** the chance of collision occurrence.

Optimization problem

Discussion:



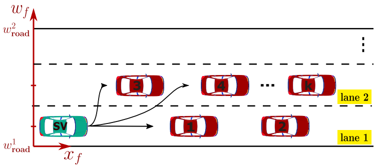
$$\min_{x_{\text{scene}}} \sum_{i=1,\dots,k} d_{x_f,\text{critical}}^{\text{SV},i}(x_{\text{scene}}) + d_{w_f,\text{critical}}^{\text{SV},i}(x_{\text{scene}})$$

- Depending on the **criticality interested**, one can
  - blend the critical distances differently
  - use an alternative function f_{system} to guide the search in the optimization process

Numerical tests

Test 1:

- # of obstacles (k): 1, $w_{f1} = 0$ [m]
- $x_{scene} = [x_{f1}^0, v_1^0]'$ [m, km/h]
- $\ell = [5, 30]'$, $u = [50, 80]'$



Test 2:

- # of obstacles (k): 3, $w_f = [0, 3, 3]$
- $x_{scene} = [x_{f1}^0, v_1^0, x_{f2}^0, v_2^0, x_{f3}^0, v_3^0]'$
- $\ell = [15, 30, 0, 10, 10, 30]'$, $u = [50, 80, 100, 80, 100, 80]'$
- $x_{f3}^0 - x_{f2}^0 > L$, $v_3^0 > v_2^0$

Test 3:

- # of obstacles (k): 5, $w_f = [0, 0, 3, 3, 3]$
- $x_{scene} = [x_{f1}^0, v_1^0, x_{f2}^0, v_2^0, x_{f3}^0, v_3^0, x_{f4}^0, v_4^0, x_{f5}^0, v_5^0]'$
- $\ell = [15, 30, 0, 10, 0, 10, 10, 10, 20, 10]'$, $u = [50, 80, 100, 80, 100, 80, 100, 80, 100, 80]'$
- $x_{f2}^0 - x_{f1}^0 > L$, $v_1^0 < v_2^0$, $x_{f4}^0 - x_{f3}^0 > L$, $v_4^0 > v_3^0$, $x_{f5}^0 - x_{f4}^0 > L$, $v_5^0 > v_4^0$

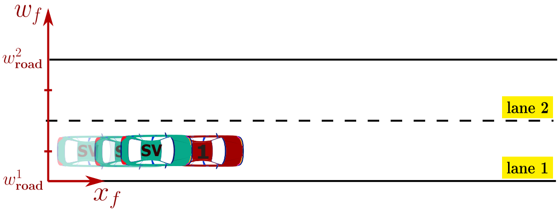
Results and discussions - Test 1

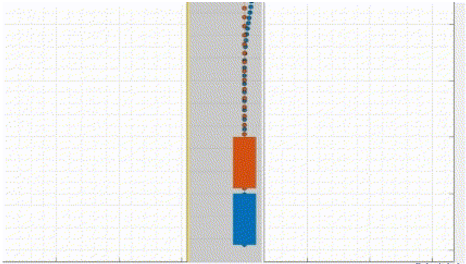
GLIS: $N_{\max} = 50$, $N_{\text{init}} = 13$

Iter	x_{scene}	
	x_{f1}^0	v_1^0
18	5	41.72
19	5	36.62
21	5	30.89

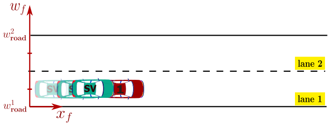
- GLIS identifies 4 collision cases within 50 simulation experiments
- 3 sample iter. with x_{scene} that can lead to **collision** are shown on the table
- The one highlighted is the 'best'/most critical one identified by the optimizer among these collision cases

Collision illustration:





Results and discussions - Test 1



Iter	x_{scene}	
	x^0_{f1}	v^0_1
18	5	41.72
19	5	36.62
21	5	30.89

Collision triggering condition

- Initial position between the SV and OV_1 is too close
- The SV is not able to brake fast enough

Discussion

- In general, the results reveal the group of scenarios that would lead to a critical one, based on which we can refine the ODD definition
  - **Critical ones**: Small x^0_1 and slow v^0_1
  - **ODD defn refinement**: update the lower bounds on x^0_1 or v^0_1 or both

Note: Criticality can also be assessed based on predefined criteria after optimization (e.g., relative velocity at collision)

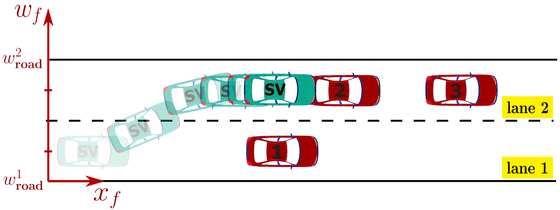
Results and discussions - Test 2

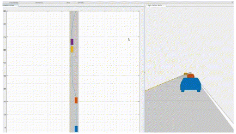
GLIS: $N_{\max} = 100$, $N_{\text{init}} = 25$

Iter	x_{scene}					
	x_{f1}^0	v_1^0	x_{f2}^0	v_2^0	x_{f3}^0	v_3^0
51	15.00	30.00	44.14	10.00	49.10	47.39
79	28.09	30.00	70.29	10.00	74.79	31.74
40	34.30	30.00	60.59	10.00	77.80	35.97

Note:

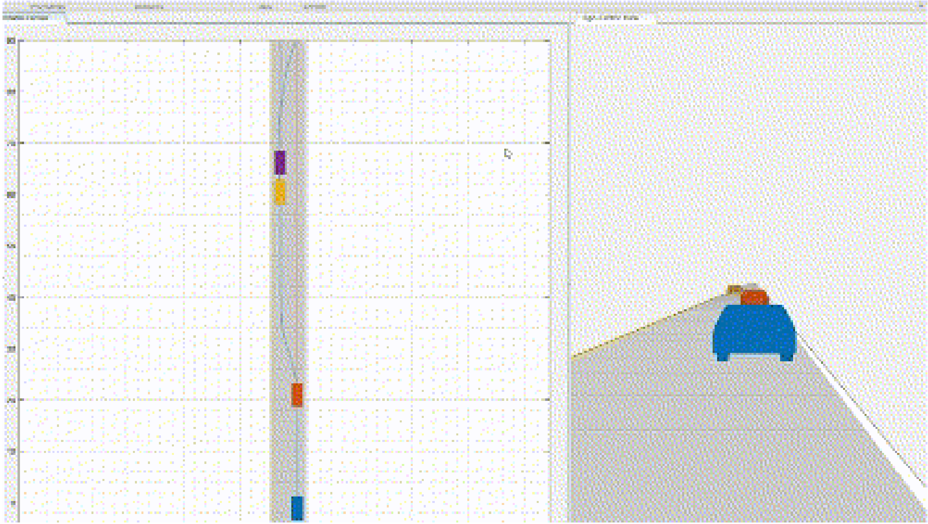
- GLIS identifies 64 collision cases within 100 simulation experiments





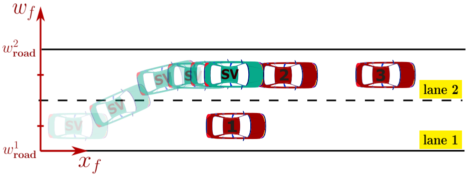
Video
(next slide)

Video



Results and discussions - Test 2

Iter	x_{scene}					
	x_{f1}^0	v_1^0	x_{f2}^0	v_2^0	x_{f3}^0	v_3^0
51	15.00	30.00	44.14	10.00	49.10	47.39
79	28.09	30.00	70.29	10.00	74.79	31.74
40	34.30	30.00	60.59	10.00	77.80	35.97



Collision triggering conditions and discussions

- 1) SV change lane to avoid OV_1; 2) SV cannot brake fast enough to avoid OV_2
- To avoid OV_2, lane change is not an option for SV (OV_1 blocks the way)
- **Critical** x_{scene}:
  - A relatively large x_{f1}^0 coupled with a relatively slow v_1^0
    - The smaller x_{f1}^0, the greater v_1^0
  - A slow v_2^0 with a large x_{f2}^0

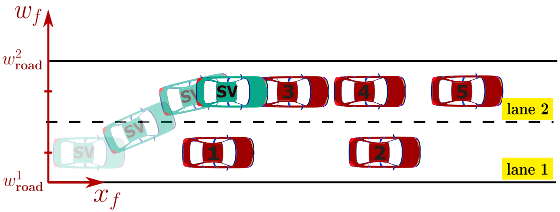
GLIS: $N_{\max} = 100$, $N_{\text{init}} = 25$

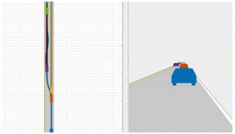
Iter	x_{scene}									
	x_1^0	v_1^0	x_2^0	v_2^0	x_3^0	v_3^0	x_4^0	v_4^0	x_5^0	v_5^0
75	15.00	30.00	19.50	30.01	48.54	10.00	60.32	10.00	86.32	51.26
97	22.89	30.00	57.34	30.00	56.06	10.00	68.76	24.45	73.26	41.54
76	29.46	30.00	62.40	36.42	42.87	16.84	65.56	31.00	76.14	42.29

Note:

- GLIS identifies 73 collision cases within 100 simulation experiments

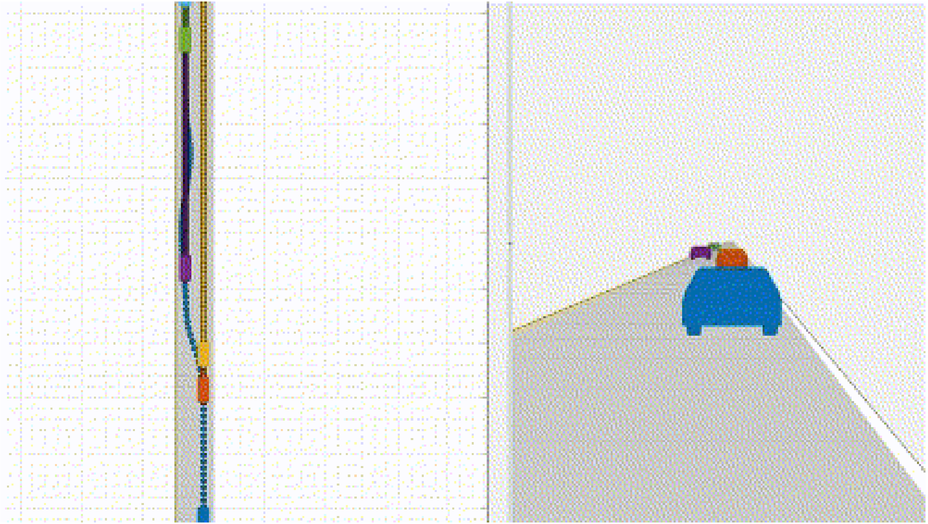
Collision illustration:





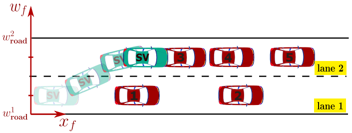
Video
(next slide)

Video



Results and discussions - Test 3

Iter	x_{scene}									
	x_1^0	v_1^0	x_2^0	v_2^0	x_3^0	v_3^0	x_4^0	v_4^0	x_5^0	v_5^0
75	15.00	30.00	19.50	30.01	48.54	10.00	60.32	10.00	86.32	51.26
97	22.89	30.00	57.34	30.00	56.06	10.00	68.76	24.45	73.26	41.54
76	29.46	30.00	62.40	36.42	42.87	16.84	65.56	31.00	76.14	42.29



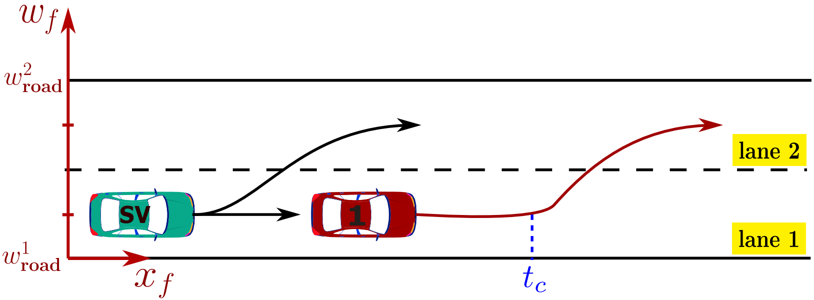
Collision triggering conditions and discussions

- 1) SV change lane to avoid OV_1; 2) SV cannot brake fast enough to avoid OV_3
- To avoid OV_3, lane change is not an option for SV (OV_1 or OV_2 or both blocks the way, depending on the initial conditions)
- **Critical** x_{scene}:
  - Similar to the ones identified in Test 2
  - A relatively large x_{f1}^0 coupled with a relatively slow v_1^0
    - The smaller x_{f1}^0, the greater v_1^0
  - A slow v_3^0 with a large x_{f3}^0

Logical scenario 1 - Discussion

- Identified **critical scenarios**:
  - SV not able to decelerate fast enough
  - OVs block the way for lane change
- The critical scenarios can be eliminated by updating the **ODD definition**
  - In this case, update the bounds of X_{scene}
  - (Or update controller designs)
- For this relatively simple setup, adding more obstacle vehicles **DOES NOT** provide more insight for potential critical scenarios
  - The SV only interact with the surrounding OVs
  - Obstacle avoidance mechanism of SV is same for every OV
  - **BUT** demonstrate the ability of GLIS to handle relatively high dimension problems

Case studies - Logical Scenario 2

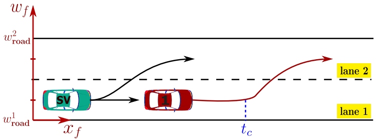


- ODD description
- Numerical tests
- Results and discussions

Case study - Logical Scenario 2

ODD description[1]:

- Two vehicles on a one-way horizontal road with two lanes
  - **AP**: road width, vehicle dimensions, experiment duration



- The OV: initially placed ahead of the SV on Lane 1, moves forward horizontally with a constant speed until time t_c, starting from t_c, **commanded by a MPC controller** to change lanes
  - **AP**: its initial lateral position and initial yaw angle, reference velocity and reference yaw angle
  - **PoI**: its initial longitudinal position (x^0_{f1}) and initial velocity (v^0_1), switch time (t_c)
- The SV: commanded by a MPC controller to avoid collision (when within safety distance with obstacle vehicles, change lane, decelerate or accelerate depend on the relative position and conditions
  - **AP**: its initial longitudinal & lateral position, reference velocity and reference yaw angle; safety distance

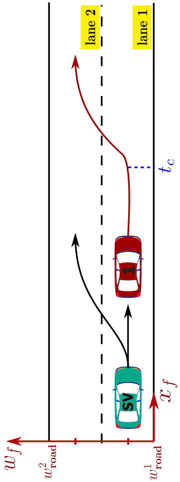
[1] AP: Assumed Parameters; PoI: Parameter of Interest

Case study - Logical Scenario 2

ODD description[1]:

- MPC controller - SV: command the subject vehicle for obstacle avoidance, the controller under testing
  - **AP**: MPC parameters
  - **Note**: constraints are adaptive to **PoI**

- MPC controller - OV: command the obstacle vehicle to change lane
  - **AP**: MPC parameters
  - **Note**: constraints are adaptive to **PoI**



[1] AP: Assumed Parameters; PoI: Parameter of Interest

Case study - Logical Scenario 2

Dimensions and Simulation time:

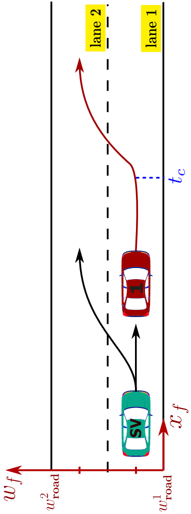
- Road width: 6 m total, 3 m/lane;
- Vehicle dim (SV & OV): $L = 4.5$ m, $W = 1.8$ m
- Experiment duration: $t_{\text{exp}} = 30$ s

Safety distance:

- longitudinal ($x_{f,\text{safe}}$): 10 m
- lateral ($w_{f,\text{safe}}$): 3 m

Initial conditions:

- SV: (0, 0) m, 50 km/h, $\theta^{\text{SV},0} = 0°$
- OV: (x^0_{f1}, 0) m, v^0_1 km/h, $\theta^0_1 = 0°$



Case study - Logical Scenario 2

OV - MPC parameters:

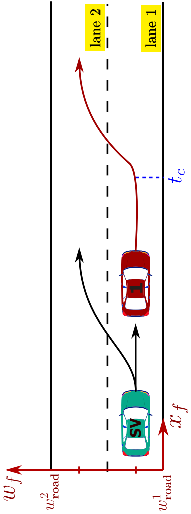
- T_s = 0.085 s, N_u = 3, N_p = 23,
- Q_y = diag(0, 10, 1), Q_u = diag(1, 1), $Q_{\Delta u}$ = diag(1, 0.5)

OV - Constraints and references (fixed):

- $v_1 = v_1^0$ km/h, $\dot{v}_1 = 0$ m/s^2, with $v_{1,\text{ref}} = v_1^0$ km/h
- $\psi_1 \in [-45, 45]^\circ$, $\dot{\psi}_1 \in$ [-60, 60]°/s
- $w_{f1} \in$ [-0.6, 3.6] m, $x_{f1} \in [x_1^0, \infty]$ m, $\theta_1 \in$ [-90, 90]°

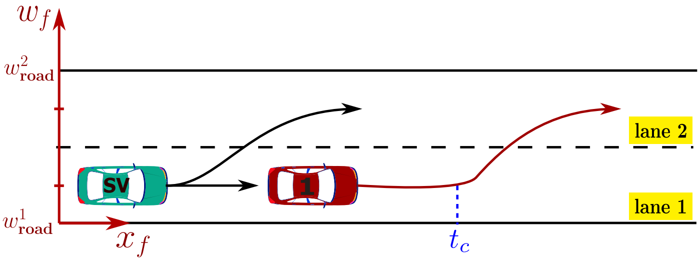
SV: the controller under testing

- The same MPC controller as in Logical Scenario 1

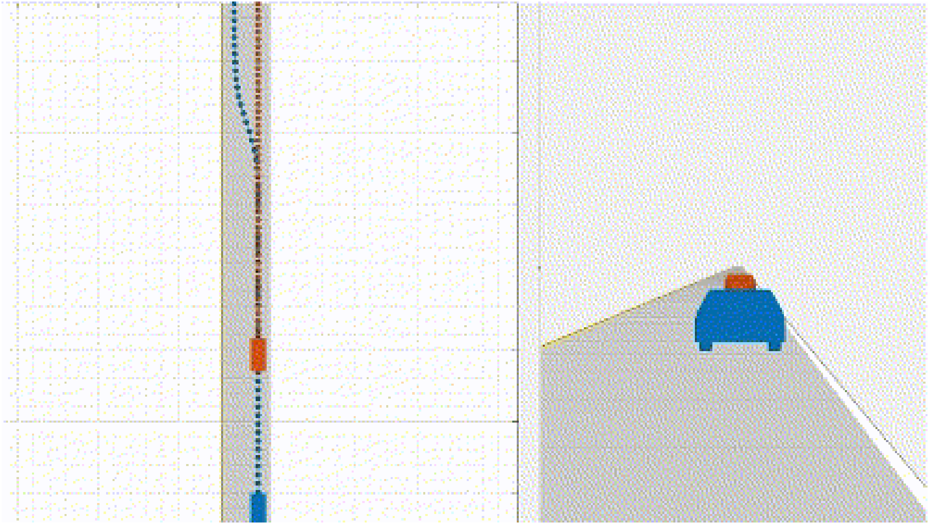


Numerical tests

- $x_{scene} = [x_{f1}^0, v_1^0, t_c]'$ [m, km/h, s]
- $\ell = [11, 30, 0]', \quad u = [50, 80, 40]'$



Video



Operational Design Domain (ODD)

ODD: The set of conditions under which a given system is designed to function (ORAD committee, 2021).

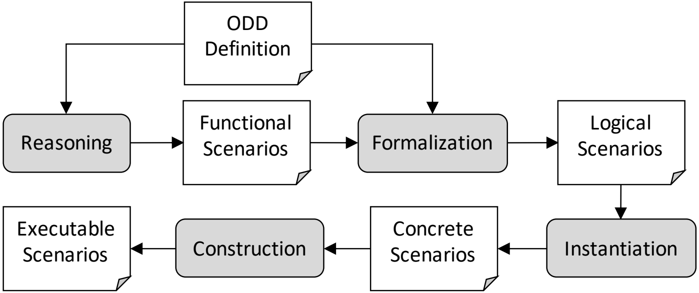


Figure 4 from (Zhang et al 2021): Relationships between scenario description at different levels of abstraction.

Scenario

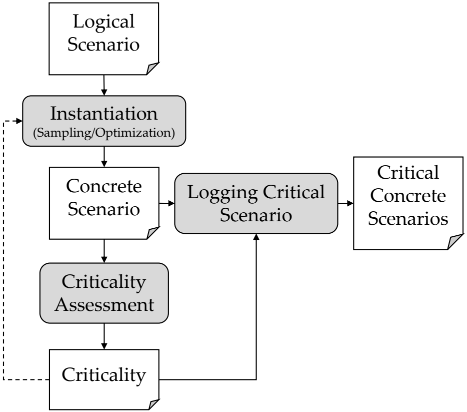


Figure 16 from (Zhang et al 2021): Critical concrete scenario identification process.

Scene: A scene describes a snapshot of the environment (Ulbrich et al, 2015).

Scenario: A scenario describes the temporal development between several scenes in a **sequence** of scenes (Ulbrich et al, 2015).

Critical scenario/edge or corner case: A relevant scenario for system design, safety analysis, verification or validation that may lead to harm (Zhang et al 2021). ('test cases' within an ODD)