賢くソフトウェアをつつく・直す

探索・学習によるソフトウェア工学

ERATO-MMSDプロジェクト グループ3

どんな問題？

ソフトウェアが扱う領域はますます増えておりシステムの不具合検出やその修正などの様々なタスクは， 非常に難しく人手で太刀打ちできないものになりつつあります。

どんな研究？

ソフトウェア工学における様々な問題を、探索・学習の技術を活用し、賢くポイントを絞り込んだり、傾向を学習したりすることによって効果的・効率的に解くことを目指しています。

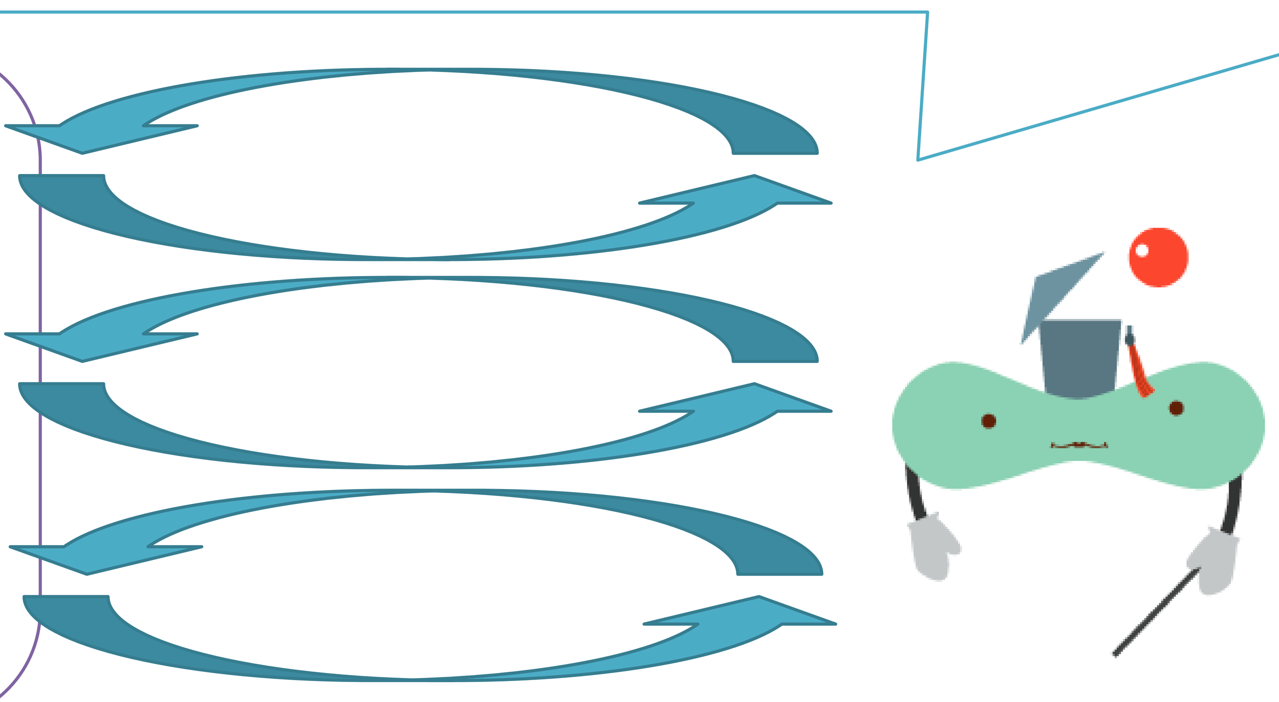
探索・学習技術

例： 自動ブレーキソフトウェアの不具合発見

設定１でやってみたら歩行者まで最小８ｍ,
設定２では３ｍだったぞ
設定２を少しいじるともっと危ないケースが出るかな？

シミュレーションによりテストするときの多数のパラメータ・設定項目

- 初期速度
- 先行車や歩行者の位置
- 路面状況・カーブや坂
- ドライバーのアクセル・ブレーキ・ハンドル操作



ポイント絞り込みや傾向の把握により「欲しいもの」にどんどん近づけていく！

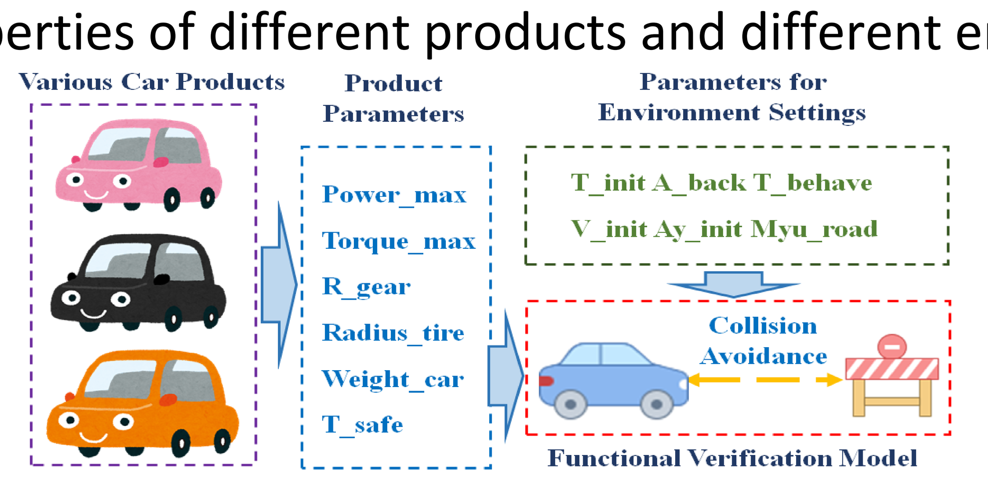
※ 遺伝子の進化を模倣した進化計算などのメタヒューリスティックに近い考え方

様々な応用

- 危険なケースなど要件に違反するテストケースを生成する
- バージョンアップで挙動が大きく変わってしまうテストケースを生成する
- 網羅性や多様性が高いテストケースリスト（テストスイート）を生成する
- 正しさの基準を満たすような設計やコードの断片や修正を生成する
- ・・・

Hazard Assessment for Automotive Systems (Xiaoyi Zhang)

- The functions of automotive products should be examined based on simulation models (e.g., simulink models) . We set different parameters to represent the properties of different products and different environments

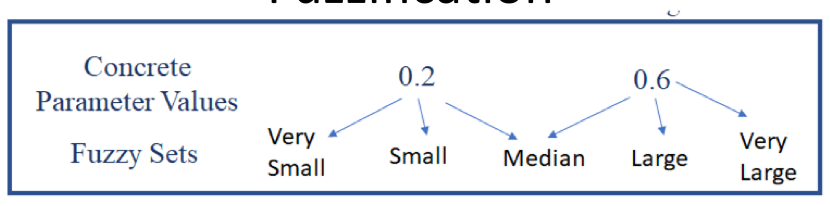


- Considering a function of automotive vehicles - what we can do?
  - To find instances that a product *fails to* complete a function → Falsification
  - **To explore the mechanism of potential hazard and improve the product line from a high level → Hazard Assessment**

	Parameter Values												Hazard Degree
Simulation 1:	4.0736	2.5116	85.0795	0.4567	3.5294	0.278	0.3228	0.3047	9.6813	98.2444	123.642	1.48E+03	1
Simulation 2:	4.7858	1.6708	112.0112	0.0709	2.687	0.9326	0.7338	0.3459	7.4181	51.7856	227.3694	1.45E+03	0.5
	...	...	...	...	...	...	...	...	...	...	...	...	...

Which Parameters are related to hazard? ⇐ **Analysis based on large amount of simulation data** ⇒ Which Types of Potential Hazards the system has?

Fuzzification

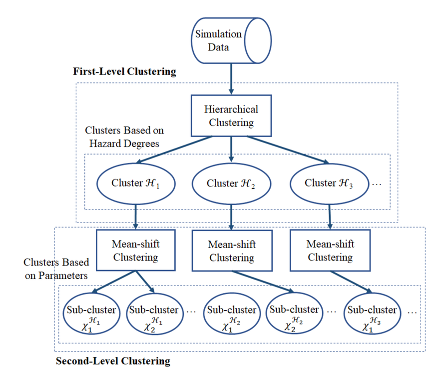


Statistic Matrics:

$$\Phi(f^{p_i}) = a_h^{f^{p_i}} / (a_h^{f^{p_i}} + a_r^{f^{p_i}})$$

Important Ranks

Para.	Deg.	Fuz.
T_init	0.1	VS
A_Back	0.05	VS
V_init	0.03	VS
T_safe	0.03	VL
T_safe	0.03	L
...	...	...



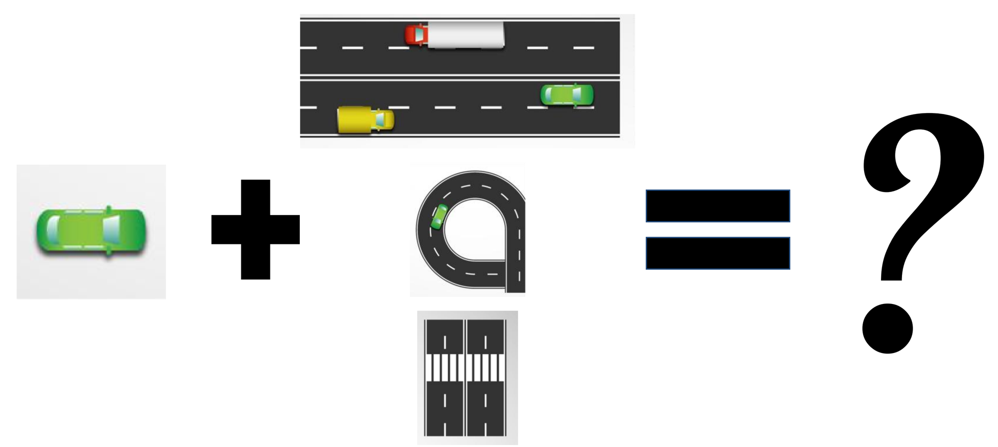
Typical Hazard-Related Parameters Settings

h	**T_safe**	*T_behav*	**V_init**	Ay_init	**T_init**	Myu_road	**A_back**	Radius_tire	*R_gear*	Power_max	Torque_max	Weight_car
	L, 5	SL, 16	S, 7	SL, 29	S, 11	SS, 43	VS, 2	M, 34	SL, 18	L, 13	SL, 21	S, 15

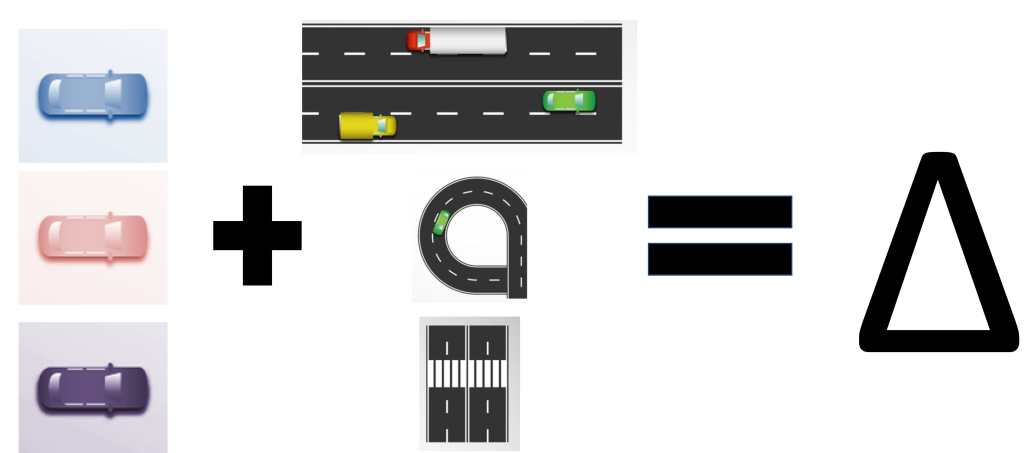
Mechanism: Which Parameters Lead to Which Types of Hazards (and How?)

Mutation based driving scenario assessment (Thomas Laurent)

- We test autonomous driving systems by simulating different scenarios and checking that the system's behaviour is adequate (e.g. no crash, safety distances respected, …)



- How to know that the scenarios we use are enough to stress the system and thoroughly test it? We apply the principle of mutation analysis:
  - We create "mutant" systems by changing the original decision taking method
  - We run these mutants on the same scenarios we used on the system

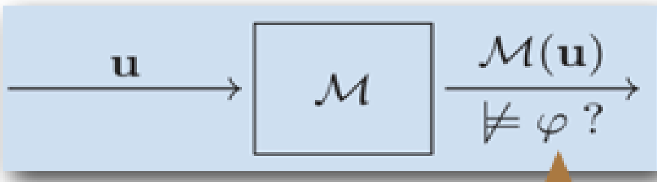


- We look at the difference Δ between the original system's performance and the mutants.
  - If the mutants behave very differently, this shows the strength of our scenarios. We are reassured about the quality of the system
  - If the mutants behave similarly, our scenarios are not stressing some aspect of the system. The mutants point us to some missing scenarios

Stochastic Optimization based Hybrid System Falsification (Zhenya Zhang)

Optimization-based Falsification

Problem:



Find one counterexample to refute the specification

$\mathbf{u}$: input

$\mathcal{M}$: model

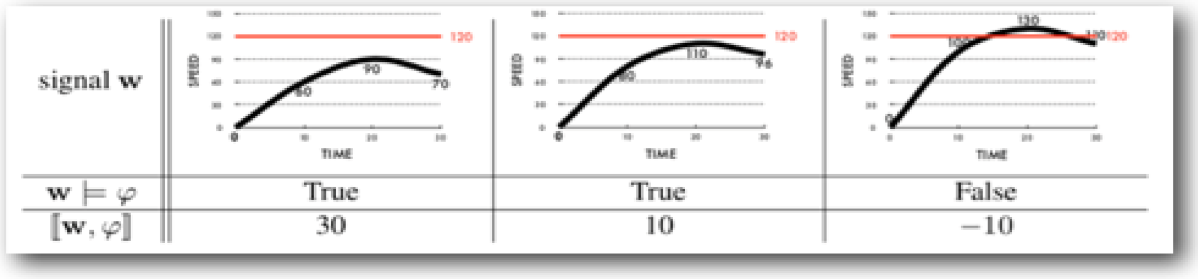
$\mathcal{M}(\mathbf{u})$: output

φ : specification in STL

Robustness: $[\![\mathcal{M}(\mathbf{u}), \varphi]\!]$

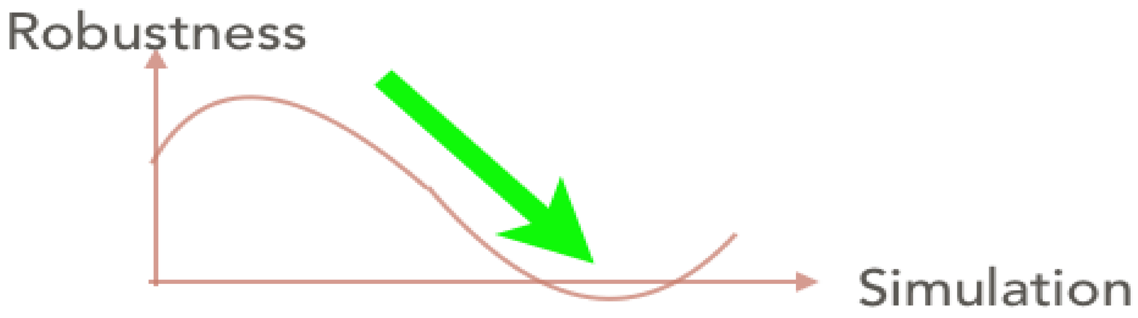
- How robustly $\mathcal{M}(\mathbf{u})$ satisfies φ

E.g. $\Box(speed < 120)$



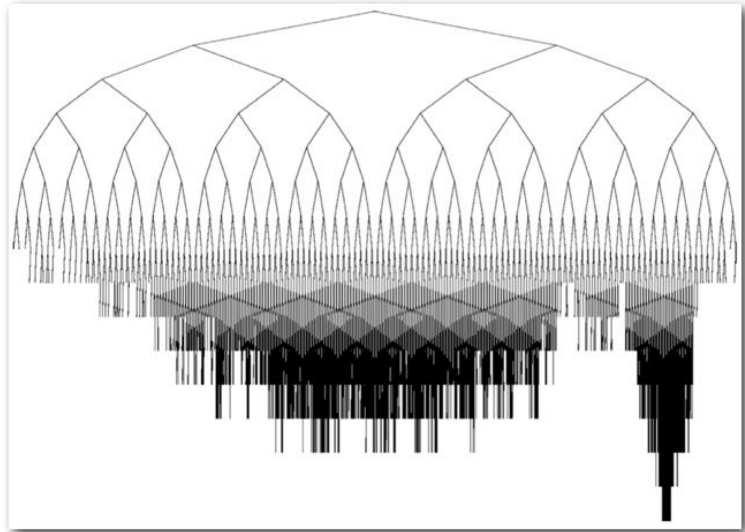
Solution:

- goal: $\min[\![\mathcal{M}(\mathbf{u}), \varphi]\!]$
- technique: hill-climbing optimization



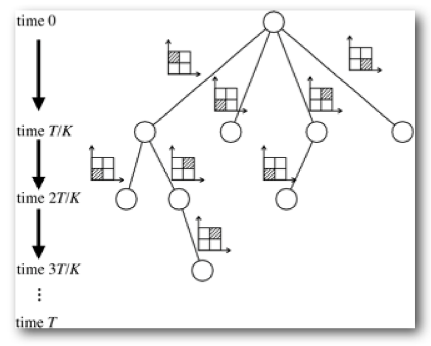
		AT model										AFC model				FFR model	
		S1		S2		S3		S4		S5		Sbasic		Sstable		Strap	
Algorithm		succ.	time	succ.	time	succ.	time	succ.	time	succ.	time	succ.	time	succ.	time	succ.	time
Random		10/10	108.9	10/10	289.1	1/10	301.1	0/10	-	0/10	-	6/10	278.7	10/10	242.6	4/10	409.3
CMA-ES	Breach	10/10	21.9	6/10	30.3	**10/10**	**193.9**	4/10	208.8	3/10	75.5	**10/10**	**111.7**	3/10	256.3	**10/10**	**119.8**
	Basic	10/10	15.8	10/10	108.5	10/10	697.1	7/10	786.8	9/10	384.4	10/10	182.0	7/10	336.9	10/10	338.0
	P.W.	**10/10**	**10.8**	**10/10**	**65.7**	10/10	728.6	**7/10**	**767.8**	**10/10**	**648.1**	10/10	177.1	**8/10**	**272.9**	10/10	473.9
GNM	Breach	**10/10**	**5.4**	**10/10**	**151.4**	0/10	-	0/10	-	0/10	-	**10/10**	**171.4**	0/10	-	0/10	-
	Basic	10/10	12.4	10/10	162.3	**10/10**	**185.6**	7/10	261.9	7/10	163.7	10/10	227.1	2/10	378.5	**10/10**	**162.2**
	P.W.	10/10	60.8	9/10	110.7	8/10	211.2	**8/10**	**313.0**	**10/10**	**178.7**	10/10	252.0	**6/10**	**153.2**	6/10	197.4
SA	Breach	**10/10**	**160.1**	0/10	-	3/10	383.7	0/10	-	3/10	80.4	0/10	-	6/10	307.0	3/10	92.8
	Basic	10/10	264.8	9/10	236.1	**8/10**	**385.6**	**8/10**	**505.3**	7/10	341.2	5/10	391.3	**8/10**	**273.8**	**10/10**	**273.2**
	P.W.	10/10	208.7	**10/10**	**377.6**	8/10	666.0	7/10	795.4	**10/10**	**624.2**	**8/10**	**665.7**	6/10	293.7	10/10	390.9

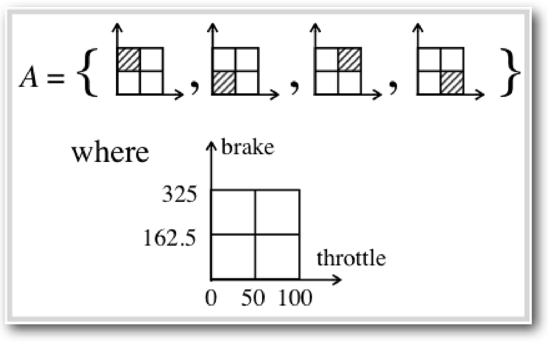
Falsification Using MCTS



- Search-based problem
- Huge search space
- Biased search
- Global optimum
- Low "regret"

The figure is from [A Survey of Monte Carlo Tree Search Methods, C Browne et al. 2012]





- Input space is discretized and structured as a tree
- Use reward to evaluate each sub-region
- Apply MCTS to address the most promising branch

S1 $\Box_{[0,30]}\ (speed < 120)$

S2 $\Box_{[0,30]}\ (gear = 3 \rightarrow speed \geq 20)$

S3 $\Diamond_{[10,30]}\ (speed \leq 53 \vee speed \geq 57)$

S4 $\Box_{[0,29]}\ (speed < 100) \vee \Box_{[29,30]}(speed > 65)$

S5 $\Box_{[0,30]}(rpm < 4770 \vee \Box_{[0,1]}(rpm > 600))$

Sbasic $\Box_{[11,30]}(\neg(|AF - AF_{\text{ref}}| > 0.05 * 14.7))$

Sstable $\neg(\Diamond_{[6,26]}\Box_{[0,4]}(AF - AF_{\text{ref}} > 0.01 * 14.7))$

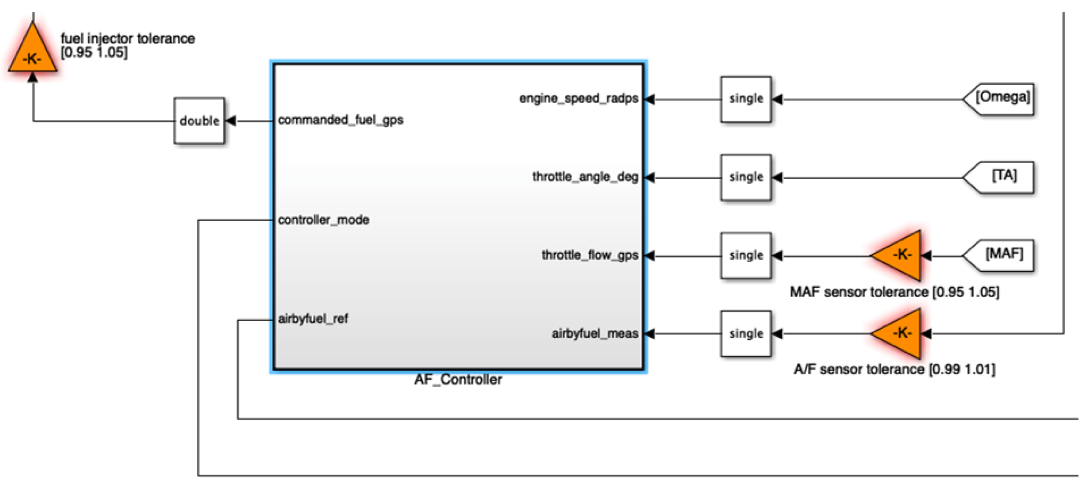
Strap $\neg\Diamond_{[0,5]}(x, y \in [3.9, 4.1] \wedge \dot{x}, \dot{y} \in [-1, 1])$

Reference:
Zhenya Zhang, Gidon Ernst, Sean Sedwards, Paolo Arcaini, Ichiro Hasuo. **Two-Layered Falsification of Hybrid Systems Guided by Monte Carlo Tree Search.** *IEEE Trans. on CAD of Integrated Circuits and Systems 37(11): 2894-2905 (2018)*

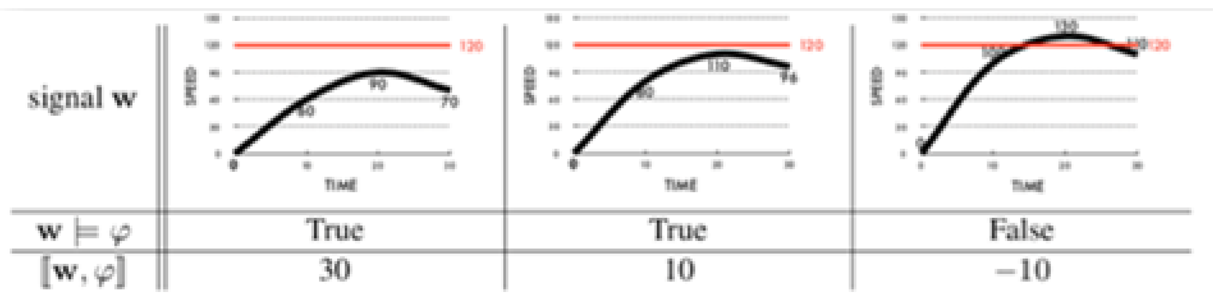
Stability Analysis for Safety of Automotive Multi-Product Lines (Paolo Arcaini)

(1) Automotive Multi-Product Lines (Multi-PL)

- We consider Simulink models of automotive systems
- Such models contains different sources of *variability*
- In [1], we view the automotive product as a *Multi-Product line* containing different *parameters*
- We identify three types of parameters:
  - *production parameters* (e.g., type of sensors or tires)
  - *environmental parameters* (e.g., type of road, weather conditions)
  - *behavioral parameters* (e.g., acceleration, initial velocity)
- A *configuration* is one particular instantiation



(2) Robustness



- We assess the satisfaction/violation of a safety requirement in a *quantitative* way
- *Robustness* (rob) gives the degree of satisfaction/violation of a requirement given an output signal *w*

(3) Instability of Automotive Multi-PLs

- If a pair of *similar* configurations exhibits quite *different* behaviors in terms of safety (i.e., large difference in robustness), this pair of configurations is a witness of the *instability*
- Defining *similarities* and *differences* requires domain knowledge (that we don't have)
- Therefore, we *search* for pairs of configurations that could be unstable
  - These will be inspected by the engineers

(4) Multi-objective problem

- We state the problem of detecting instability as a *multi-objective* problem
- Let ∑ be the set of feasible configurations, *c*,*d* be two configurations, and *D*(c,d) be the configuration difference between *c* and *d*. The multi-objective optimization problem is:

$$\max_{c,d\in\Sigma} |rob(c) - rob(d)| \qquad \min_{c,d\in\Sigma} D(c,d)$$

- We use search-based algorithms to solve it
- We obtain a *Pareto front* of solutions that can be inspected by engineers

(5) Case study

Scenario

- A car drives on a straight lane and spots an obstacle in front of it
- When the driver pushes the brake pedal, an active safety feature is also triggered
- The active safety feature decides whether and when to shut down the engine (if it is necessary to avoid an accident)

Safety requirement: "The car should not crash into the obstacle; if a collision is unavoidable, the car should collide at the lowest possible speed."

Robustness: $rob(c) = \begin{cases} distance(c), & \text{if distance(c)} > 0, \\ -V\_collision(c), & \text{otherwise.} \end{cases}$

Production parameters	Environmental parameters
• T_safe: engine shut down time • Radius_tire: tires radius • Power_max: maximum horsepower • Torque_max: maximum torque • Weight_car: weight of the car	• T_init: initial inter-vehicle time **Behavioural parameters** • A_back: evasive backward acceleration • T_behav: reaction time • V_init: initial car speed • R_gear: wheel RPM

(6) Case study - Results

Config.	rob	T_safe	Power_max	Torque_max	T_init	A_back	T_behav	V_init	...
$c_1^{T\_safe}$	0.189	3.132	128.041	217.062	3.063	0.881	4.076	54.775	
$c_2^{T\_safe}$	-13.873	3.255	128.041	217.062	3.063	0.881	4.076	54.775	

$c_1^{T\_safe}$ and $c_2^{T\_safe}$ show that shutting down the engine 0.123 seconds late could result in an accident

Config.	rob	T_safe	Power_max	Torque_max	T_init	A_back	T_behav	V_init	...
$c_1^{T\_init}$	0.333	4.482	94.084	306.560	2.119	0.825	4.263	68.544	
$c_2^{T\_init}$	-20.188	4.482	94.084	306.560	1.999	0.825	4.263	68.544	

$c_1^{T\_init}$ and $c_2^{T\_init}$ show that a difference of 0.12 seconds in the inter-vehicle time (a difference of 2.285 meters in their initial distances) makes a big difference

- $c_1^{T\_init}$ satisfies the safety requirement, stopping in front of the obstacle by 0.333 meters
- $c_2^{T\_init}$ crashes into the obstacle at a speed of 20.188 km/h

Config.	rob	T_safe	Power_max	Torque_max	T_init	A_back	T_behav	V_init	...
$c_1^{A\_back}$	0.539	3.366	120.473	447.615	4.547	0.112	1.064	64.748	
$c_2^{A\_back}$	-10.619	3.366	120.473	447.615	4.547	0.106	1.064	64.748	

$c_1^{A\_back}$ and $c_2^{A\_back}$ show that providing a backward acceleration 0.006 G less could result in an accident

References

[1] S. Ali, P. Arcaini, I. Hasuo, F. Ishikawa, N. Lee. Towards a Framework for the Analysis of Multi-Product Lines in the Automotive Domain. In Proceedings of the 13th International Workshop on Variability Modelling of Software-Intensive Systems (VAMOS 2019)
[2] N. Lee, P. Arcaini, S. Ali, F. Ishikawa. Stability Analysis for Safety of Automotive Multi-Product Lines: A Search-Based Approach. In Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2019)