Application of Simulation-Based Methods on Autonomous Vehicle Control with Deep Neural Network:Work-in-Progress

1st Yuji Date
Frontier Research Center
Toyota Motor Corporation
1200 Mishuku, Susono,
Shizuoka, Japan
yuji_date@mail.toyota.co.jp

2nd Takeshi Baba
Frontier Research Center
Toyota Motor Corporation
1200 Mishuku, Susono,
Shizuoka, Japan
l6033@mosk.tytlabs.co.jp

3rd Bardh Hoxha
Toyota Research Institute-North America
1555 Woodridge Ave,
Ann Arbor, Michigan, U.S.
bardh.hoxha@toyota.com

4th Tomoya Yamaguchi
Toyota Research Institute-North America
1555 Woodridge Ave,
Ann Arbor, Michigan, U.S.
tomoya.yamaguchi@toyota.com

5th Danil Prokhorov
Toyota Research Institute-North America
1555 Woodridge Ave,
Ann Arbor, Michigan, U.S.
danil.prokhorov@toyota.com

***Abstract*—Recent developments in simulation-based testing methods for automotive systems with machine learning components have shown promise. This work in progress paper presents our efforts in applying these methods in the evaluation and development of control and perception systems. Experimental results demonstrate a significant improvement in system performance.**

***Index Terms*—Cyber-physical systems, Image recognition, Adversarial machine learning, Automatic test pattern generation, Autonomous vehicles**

I. Introduction

The verification of safety-critical systems is particularly challenging due to the complex interactions of the system with the physical environment. These systems are typically referred to as Cyber-Physical Systems (CPS). Furthermore, most implementations utilize deep neural networks (DNNs) [1] for various tasks such as perception and planning. These networks typically have tens of millions of parameters and have been trained with millions of instances of training data. Thus it has been an open issue to ensure a quality of a control system including DNNs with many parameters.

Simulation-based methods have shown promise in testing and verification of autonomous vehicles [2], [3]. Due to scale of the problem, physical testing alone is not sufficient and often dangerous.

In this work, we utilize existing simulation-based methods to generate adversarial instances for retraining of machine learning components [4]. We demonstrate our approach on an autonomous braking scenario and show how falsified test cases can be utilized for retraining of a DNN controller.

II. Approach

The proposed approach is to utilize falsification methods to find test cases which do not satisfy system specification. System specification are defined in Signal Temporal Logic (STL) (see [5] for an overview). The notion of robustness of STL formulas is then utilized to pose the falsification problem as an optimization problem. Then we utilize various stochastic optimization algorithms to generate test cases. For this, we utilize tools S-TaLiRo [6] and Sim-ATAV [7].

III. Experimental Results

The scenario under consideration is a highway emergency braking scenario where a static obstacle or pedestrian appears in front of an autonomous ego vehicle (Fig. 1). The requirement states that if a pedestrian appears in front of the vehicle and the distance and time to collision are less than some fixed values, then the vehicle should brake and ego vehicle shall not collide with the obstacle. Formally, we define this with the STL formula:

$$\varphi = \Box\neg(V_{ego} > \epsilon_{mov} \wedge dist(ego, ped) < \epsilon_{min})$$

Here, V_{ego} is the speed of the ego vehicle and ϵ_{mov} is a safe vehicle speed. The formula states that always, it should not be the case that the vehicle is in motion with speed greater than ϵ_{mov} and the distance is shorter than ϵ_{min}.

The emergency brake controller consists of a DNN and a simple braking task. We adopted SqueezDet [8] as a perception system to detect and classify objects in front of the ego vehicle. The network is trained through supervised learning by utilizing 7500 images from KITTI [9] and 7800 images from Webots in order to recognize vehicles and pedestrians, etc. Braking is done immediately if DNN finds pedestrians. We chosen this simple controller system because we focus on the falsification of the DNN.

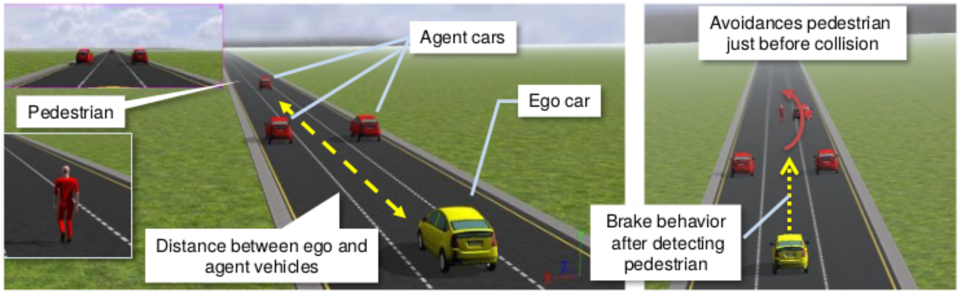
Pedestrian
Agent cars
Ego car
Distance between ego and agent vehicles
Avoidances pedestrian just before collision
Brake behavior after detecting pedestrian

Fig. 1. Emergency Braking Scenario. Left: The yellow (ego) and the red (agent) vehicles are travelling on the highway at speeds approx. 100 km/h. In front, obscured by one of the agent vehicles, a pedestrian is walking on the center lane. Right: After the agent vehicle avoids the pedestrian just before collision, the ego vehicle should brake after detecting the pedestrian.

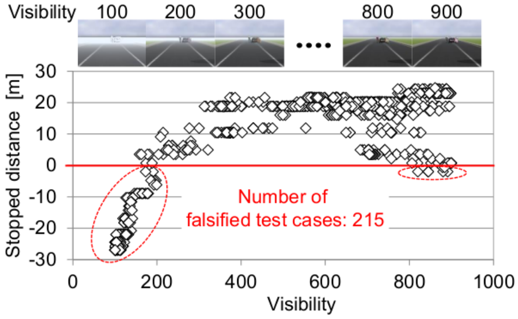

Fig. 2. Falsification results from 1,000 simulations across a range of visibility. The stopped distance is distance between ego and the pedestrian when the ego finishes the braking completely.

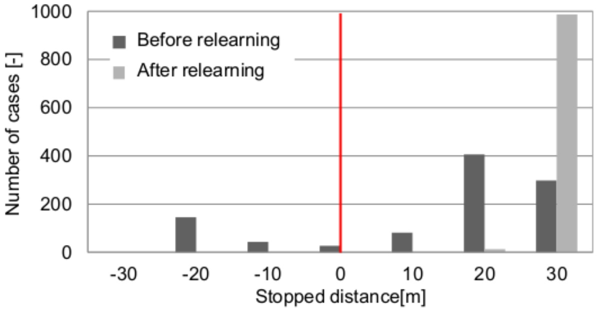

Fig. 3. Comparison of falsification results between before and after the relearning from the counter examples from the falsification. Zero or Negative values of stopped distance correspond to falsified cases.

We use the falsification algorithm in Sim-ATAV to generate test cases where the specification is not satisfied. The falsification is posed as an optimization through the theory of robustness of temporal logic specifications. Namely, for each test case, a numerical value is generated that indicates by how much the system has satisfied or failed (i.e., falsified) the specification. The search space for the problem consists of the agent vehicle color, pedestrian shirt and pants color, and visibility level, which corresponds to fog density. Parameters of color are described as the RGB parameters in the range of [0, 0, 0]-[255, 255, 255]. Parameter range of the visibility level is 100 to 900, which corresponds to low and high visibility, respectively. New test case is generated by updating these parameters to minimize the robustness value. The simulated annealing is applied for the parameter optimization.

In Fig. 2, falsification results are presented with respect to the visibility level. It can be observed that the specification is falsified at both ends of the visibility spectrum. In other words, the controller did not perform well. For the left side of the falsified cases, we can easily understand the reason, that is, dense fog inhibited the perception of pedestrian. For the right side cases, perception failed in spite of high visibility. This might be caused by insufficient learning of the DNN.

To improve the DNN model, retraining with the images of falsified cases was expected to improve the closed-loop performance. The new controller was again tested in a falsification framework. The results are presented in Fig. 3. No falsified test cases were found for the controller after the retraining process.

IV. Conclusions and Future Work

We showed the retraining of the DNN with counter examples found from our falsification algorithm in Sim-ATAV gives us an improvement of the closed-loop performance. The innovative feature of our approach is scenario based perturbations are studied, in contrast with more typical vision based adversarial testing such a pixel or Gaussian noise [10], [11]. However, more research is needed to analyze under what conditions our approach is most effective. As a future work, development of more realistic scenarios with a high fidelity simulator, such as [12], CARLA [13] is to be explored.

References

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[2] C. E. Tuncali, G. Fainekos, D. Prokhorov, H. Ito, and J. Kapinski, "Requirements-driven test generation for autonomous vehicles with machine learning components," *IEEE Transactions on Intelligent Vehicles*, 2019.

[3] T. Dreossi, D. J. Fremont, S. Ghosh, E. Kim, H. Ravanbakhsh, M. Vazquez-Chanlatte, and S. A. Seshia, "Verifai: A toolkit for the formal design and analysis of artificial intelligence-based systems," in *International Conference on Computer Aided Verification*. Springer, 2019, pp. 432–442.

[4] T. Dreossi, A. Donzé, and S. A. Seshia, "Compositional falsification of cyber-physical systems with machine learning components," *Journal of Automated Reasoning*, vol. 63, no. 4, pp. 1031–1053, 2019.

[5] O. Maler and D. Nickovic, "Monitoring temporal properties of continuous signals," in *Formal Techniques, Modelling and Analysis of Timed and Fault-Tolerant Systems*. Springer, 2004, pp. 152–166.

[6] S-TaLiRo Tools, "https://sites.google.com/a/asu.edu/s-taliro/."

[7] C. E. Tuncali, G. Fainekos, H. Ito, and J. Kapinski, "Sim-atav: Simulation-based adversarial testing framework for autonomous vehicles," in *Proceedings of the 21st International Conference on Hybrid Systems: Computation and Control (part of CPS Week)*, 2018, pp. 283–284.

[8] B. Wu, F. Iandola, P. H. Jin, and K. Keutzer, "Squeezedet: Unified, small, low power fully convolutional neural networks for real-time object detection for autonomous driving," *arXiv preprint arXiv:1612.01051*, 2016.

[9] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012.

[10] J. Lu, H. Sibai, E. Fabry, and D. Forsyth, "Standard detectors aren't (currently) fooled by physical adversarial stop signs," *arXiv preprint arXiv:1710.03337*, 2017.

[11] K. Pei, Y. Cao, J. Yang, and S. Jana, "Deepxplore: Automated whitebox testing of deep learning systems," in *proceedings of the 26th Symposium on Operating Systems Principles*, 2017, pp. 1–18.

[12] G. Rong, B. H. Shin, H. Tabatabaee, Q. Lu, S. Lemke, M. Možeiko, E. Boise, G. Uhm, M. Gerow, S. Mehta *et al.*, "Lgsvl simulator: A high fidelity simulator for autonomous driving," *arXiv preprint arXiv:2005.03778*, 2020.

[13] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "Carla: An open urban driving simulator," *arXiv preprint arXiv:1711.03938*, 2017.