AmbieGenVAE at the SBFT 2024 Tool Competition - Cyber-Physical Systems Track

Dmytro Humeniuk
dmytro.humeniuk@polymtl.ca
Polytechnique Montréal
Montréal, Canada

Foutse Khomh
foutse.khomh@polymtl.ca
Polytechnique Montréal
Montréal, Canada

ABSTRACT

Testing and verification of autonomous systems is critically important. In the context of SBFT 2024 Cyber-physical systems (CPS) testing tool competition, we present our tool AmbieGenVAE for generating virtual roads to test an autonomous vehicle lane keeping assist system. AmbieGenVAE leverages optimization in a Variational Autoencoder latent space to produce challenging test scenarios. It has achieved the highest score for one of the test subjects and the second-highest final score among 2 other submitted tools.

CCS CONCEPTS

• **Software and its engineering → Software verification and validation**; • **Computing methodologies → Search methodologies**.

KEYWORDS

lane-keeping assist system, testing, genetic algorithms, latent space

ACM Reference Format:
Dmytro Humeniuk and Foutse Khomh. 2024. AmbieGenVAE at the SBFT 2024 Tool Competition - Cyber-Physical Systems Track. In *2024 ACM/IEEE International Workshop on Search-Based and Fuzz Testing (SBFT '24), April 14, 2024, Lisbon, Portugal.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3643659.3648560

1 INTRODUCTION

The architectures of modern autonomous cyber-physical systems (CPS), such as self-driving cars, are highly complex, comprising hardware, software and machine learning components that interact with each other. It is thus essential to ensure that the developed software for autonomous CPS is robust and enables these systems to withstand the numerous challenges arising in unstructured and dynamic real-world environments.

One of the common practices is to perform simulation based testing of the developed system [9]. A number of previous works proposed optimization techniques based on evolutionary algorithms to effectively generate challenging and diverse test scenarios to be executed in simulation [6], [8]. Evolutionary algorithms were originally designed to operate on elements represented as one-dimensional arrays of real numbers or graphs (genetic programming). State-of-the-art algorithms such as evolutionary strategies, CMA-ES, differential evolution (DE), or particle swarm optimization (PSO) require the solutions to be represented as an array of real numbers [5]. At the same time, the commonly used representations for test scenarios for autonomous CPS are rather complex, including multidimensional arrays of different shapes [7], combinations of discrete and continuous values as well as sequential dependence between elements [3]. This complicates the usage of the advanced search algorithms and search operators in the test generation process. To address the problem of obtaining an efficient representation, we developed a tool AmbieGenVAE that first learns a simple one-dimensional representation of the test generation problem and then performs optimization of the candidate solutions in a latent space.

In the context of SBFT 2024 CPS testing tool competition [2], we have adopted our tool to generate virtual test scenarios falsifying an autonomous vehicle (AV) lane-keeping assist system (LKAS). A test scenario for an AV lane-keeping assist system is represented as a virtual road, defined by a sequence of 2D road points. The road points are then interpolated using cubic splines to obtain the final road geometry. According to the competition rules, the road has to remain within the map borders, the road cannot be too sharp, and cannot self-intersect. In the next section, we describe the AmbeiGenVAE tool operation in more detail.

2 AMBIEGENVAE TOOL DESCRIPTION

The AmbieGenVAE approach includes three key steps: (1) collecting a dataset of test scenarios, (2) training a variational autoencoder (VAE) model, and (3) executing an evolutionary search within the latent space of the autoencoder. The first two steps are illustrated in Figure 1.

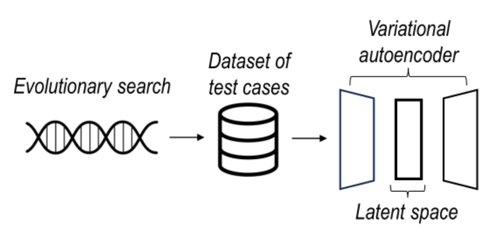

Figure 1: Collecting a dataset and training a VAE

To represent the road points we leveraged a curvature-based representation introduced by Castellano et al. [3], where a road topology is described by a sequence of N curvature values (kappa) $\kappa_0, \kappa_1, ..., \kappa_{N-1}$. To simplify the VAE training, we chose to have a

SBFT '24, April 14, 2024, Lisbon, Portugal

ACM ISBN 979-8-4007-0562-5/24/04
https://doi.org/10.1145/3643659.3648560

fixed size of a kappa vector of 17. To produce a dataset of more relevant and challenging test scenarios i.e road topologies, we first set up a search algorithm optimizing some parameters of the road topologies and ensuring their validity. We used the maximum road curvature value as the objective function. Intuitively, the bigger the curvature angle is present in the road topology, the more challenging it should be for the driving agent. When the maximum road curvature exceeded the limits, the test cases were given a low fitness value. We performed 50 runs of a genetic algorithm with a population size of 200 for 100 generations and saved the population in the last generation to obtain a dataset of 10 000 test scenarios. Each scenario was represented as 17-dimensional vector of kappa values. To ensure the diversity of the produced test scenarios, we added a duplicate removal strategy based on the cosine similarity of the vectors of kappa values that was executed after each generation of the search.

In the second step, we trained a VAE with a simple architecture [1] consisting of four fully connected layers and two ReLU activation functions. The primary objective of training the VAE was to learn a mapping function between a given road topology and its representation in a latent space. In our implementation the latent space consisted of a 17-D vector with values from the normal distribution. We opted for a latent space size identical to the input size, i.e., 17, to minimize the reconstruction loss. New test scenarios can be produced by sampling the latent space and passing the obtained vectors through the decoder module of the trained VAE.

The third step of the approach is demonstrated in Figure 2. It

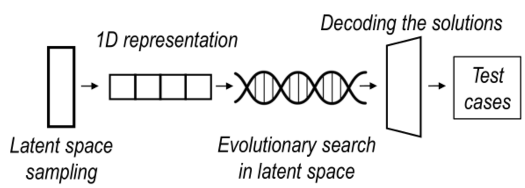

Figure 2: Performing the search in the VAE latent space

begins by sampling the latent space from the trained VAE and obtaining one-dimensional vectors of real numbers from the normal distribution. To optimize the test cases, we ran a genetic algorithm with a population size of 40 and a given time budget, which was of 3 hours in the SBFT 2024 competition. The optimization was performed in the VAE latent space. As the search operators, we used simulated binary crossover and polynomial mutation [4]. Prior to evaluation, each latent vector was transformed to an actual test case by passing it through the decoder part of the VAE. The search was guided by the performance of the driving agent in a given test scenario in the simulation. The goal was to maximize the percentage of the vehicle going out of the road lane bounds. To ensure the diversity of the test scenarios, we added a duplicate removal step after each generation based on cosine similarity of vectors representing the test scenarios.

3 EVALUATION RESULTS

The evaluation results based on 6 runs of 3 hours for two driving agents, BeamNG and Dave2, are shown in Figure 3. BeamNG drives at the maximum speed of 70 km/h and the failure is detected when more than 85 % of the vehicle goes of the bounds. The DAVE-2 agent drives slower with a maximum speed of 25 km/h, but a lower tolerance of 10 % is used to trigger failures. AmbieGenVAE revealed on average 49 failures for the BeamNG agent, achieving the highest feature space coverage score of 0.13 for this agent among the two other submitted tools. For the Dave-2 driving agent it revealed 2 failures on average, achieving the feature space coverage of 0.02 and the second-best total score.

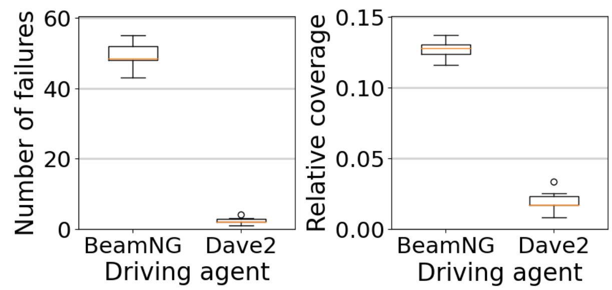

Figure 3: The evaluation results in terms of revealed failures and feature space coverage

4 CONCLUSIONS

In this paper, we introduce AmbieGenVAE, a tool for generating test scenarios for an autonomous vehicle LKAS system. The tool leverages evolutionary search in a latent space of a pre-trained VAE. The VAE is trained on a dataset of challenging, diverse and valid test scenarios. AmbieGenVAE has proven to be effective at testing the LKAS system of the BeamNG driving agent. In the future, we plan to improve the performance of the tool for the Dave-2 agent.

REFERENCES

[1] Peter J Bentley, Soo Ling Lim, Adam Gaier, and Linh Tran. 2022. COIL: Constrained optimization in learned latent space: learning representations for valid solutions. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion*. 1870–1877.

[2] Matteo Biagiola and Stefan Klikovits. [n. d.]. SBFT Tool Competition 2024 - Cyber-Physical Systems Track. In *17th IEEE/ACM International Workshop on Search-Based and Fuzz Testing, SBFT 2024, Lisbon, Portugal, April 14, 2024*.

[3] Ezequiel Castellano, Ahmet Cetinkaya, and Paolo Arcaini. 2021. Analysis of Road Representations in Search-Based Testing of Autonomous Driving Systems. In *2021 IEEE 21st International Conference on Software Quality, Reliability and Security (QRS)*. IEEE, 167–178.

[4] Kalyanmoy Deb, Karthik Sindhya, and Tatsuya Okabe. 2007. Self-adaptive simulated binary crossover for real-parameter optimization. In *Proceedings of the 9th annual conference on genetic and evolutionary computation*. 1187–1194.

[5] Agoston E Eiben and James E Smith. 2015. *Introduction to evolutionary computing*. Springer.

[6] Alessio Gambi, Gunel Jahangirova, Vincenzo Riccio, and Fiorella Zampetti. 2022. SBST tool competition 2022. In *2022 IEEE/ACM 15th International Workshop on Search-Based Software Testing (SBST)*. IEEE, 25–32.

[7] Yuqi Huai, Yuntianyi Chen, Sumaya Almanee, Tuan Ngo, Xiang Liao, Ziwen Wan, Qi Alfred Chen, and Joshua Garcia. 2023. Doppelgänger Test Generation for Revealing Bugs in Autonomous Driving Software. In *2023 IEEE/ACM 45th International Conference on Software Engineering (ICSE)*. IEEE, 2591–2603.

[8] Dmytro Humeniuk, Foutse Khomh, and Giuliano Antoniol. 2022. A search-based framework for automatic generation of testing environments for cyber-physical systems. *Information and Software Technology* 149 (2022), 106936.

[9] Cumhur Erkan Tuncali, Georgios Fainekos, Hisahiro Ito, and James Kapinski. 2018. Simulation-based adversarial test generation for autonomous vehicles with machine learning components. In *2018 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 1555–1562.