Multimodal critical-scenarios search method for test of autonomous vehicles

Tianyue Feng, Lihao Liu, Xingyu Xing and Junyi Chen
Tongji University, Shanghai, China

Abstract

Purpose – The purpose of this paper is to search for the critical-scenarios of autonomous vehicles (AVs) quickly and comprehensively, which is essential for verification and validation (V&V).

Design/methodology/approach – The author adopted the index F1 to quantitative critical-scenarios' coverage of the search space and proposed the improved particle swarm optimization (IPSO) to enhance exploration ability for higher coverage. Compared with the particle swarm optimization (PSO), there were three improvements. In the initial phase, the Latin hypercube sampling method was introduced for a uniform distribution of particles. In the iteration phase, the neighborhood operator was adapted to explore more modals with the particles divided into groups. In the convergence phase, the convergence judgment and restart strategy were used to explore the search space by avoiding local convergence. Compared with the Monte Carlo method (MC) and PSO, experiments on the artificial function and critical-scenarios search were carried out to verify the efficiency and the application effect of the method.

Findings – Results show that IPSO can search for multimodal critical-scenarios comprehensively, with a stricter threshold and fewer samples in the experiment on critical-scenario search, the coverage of IPSO is 14% higher than PSO and 40% higher than MC.

Originality/value – The critical-scenarios' coverage of the search space is firstly quantified by the index F1, and the proposed method has higher search efficiency and coverage for the critical-scenarios search of AVs, which shows application potential for V&V.

Keywords Autonomous driving system, Virtual test, Scenario, Optimization algorithm

Paper type Research paper

1. Introduction

Scenario-based virtual test of autonomous vehicles (AVs) has drawn much attention for its advantages of reproducibility, high efficiency and flexible setting of test cases (Kalra and Susan, 2016; Zhu, 2019). The scenario, as the basis of the above method, can be abstracted into three levels of functional, logical and concrete scenarios, which lay a foundation for automatic virtual tests (Menzel *et al.*, 2018). Without the scenario-based test, it is not enough to ensure the safety of AVs by the functional and logical scenarios generated based knowledge (Schuldt *et al.*, 2018). Therefore, grid search (GS) and the Monte Carlo method (MC) are widely applied to generate concrete scenarios based on the logical scenario. However, searching for the critical-scenarios essential for verification and validation (V&V) using the above methods is very costly. Thus, the main problem discussed in this paper is how to search for the critical-scenarios quickly and comprehensively within a given search space of the logical scenario.

The critical-scenarios search can be transformed into an optimization problem with the system under test (SUT) regarded as a black box. Optimization algorithms have been applied to the critical-scenarios search of AVs. Beglerovic *et al.* (2017) proposed a method based on surrogate models combined with stochastic optimization to find more critical-scenarios by less real system evaluations. Tuncali *et al.* (2018) demonstrated a way of finding a false vehicle behavior by using simulated annealing to find critical-scenarios. Masuda *et al.* (2018) proposed a method of rule-based searching for collision test cases of AVs. Mullins *et al.* (2018) developed adaptive search algorithms to discover performance boundaries of AVs without estimating the coverage. Feng *et al.* (2020) designed a critical scenario searching method based on multi-start optimization and seed-fill method. Klischat *et al.* (2020) used particle swarm optimization (PSO) to increase the criticality of the simulation scenarios. Zhu *et al.* (2021) proposed an optimization searching method to explore the critical-scenarios in a huge search space faster.

In summary, to verify the safety of AVs, it is necessary to search for scenarios with high criticality as comprehensively as

The current issue and full text archive of this journal is available on Emerald Insight at: **https://www.emerald.com/insight/2399-9802.htm**

Journal of Intelligent and Connected Vehicles
5/3 (2022) 167–176
Emerald Publishing Limited [ISSN 2399-9802]
[DOI 10.1108/JICV-04-2022-0016]

This paper forms part of a special section "Intelligent Safety for Intelligent and Connected", guest edited by Jun Li.

Thanks are due to Hexagon for assistance with the experiments. This work is supported by the National Key Research and Development Program of China. (2021YFB2501205). Thanks are due to Hexagon for assistance with the experiments.

Received 30 April 2022
Revised 26 May 2022
Accepted 10 June 2022

possible (Batsch *et al.*, 2019), especially when the search space is multimodal. However, most of the research only focuses on the criticality and search efficiency rather than the critical-scenarios' coverage in the search space, which makes the critical-scenarios insufficient to validate the safety of AVs. With the critical-scenarios search regarded as an optimization problem, the search efficiency and critical-scenarios' coverage can not be guaranteed to be optimal at the same time based on the "no free lunch" theorem (Wolpert and Macready, 1997).

In this paper, the critical-scenarios' coverage of the search space is firstly quantified by metric F1, which represents the difference between the space fitted by samples and the ground truth, and the improved particle swarm optimization (IPSO) is proposed for higher coverage. The main contributions of this paper are discussed below.

The critical-scenarios' coverage in the search space is proposed and measured by the metric F1 in the scenario-based test of AVs to evaluate the performance of optimization algorithms in critical-scenarios search. IPSO is proposed to search for the multimodal critical-scenarios more comprehensively by enhancing the exploration ability of PSO in each phase. By the experiments on the artificial function and critical-scenarios search, the efficiency and the application prospect of the method are verified.

The remainder of this paper is organized into four sections. Section 2 describes the method. Section 3 describes the experiments on the multimodal artificial function and critical-scenarios search to verify the method's efficiency and application prospect. Section 4 presents the results and discussions. Section 5 concludes this paper.

2. Method of critical-scenarios search

In this paper, the decision-making SUT is regarded as a black box, and the critical-scenarios search is transformed into an optimization problem by regarding the search space of the logical scenario as input and the safety evaluation result as output.

To quantify the critical-scenarios' coverage in more detail, the metric F1 based on the image mask operation is adopted as it compares two spaces pixel-by-pixel. For higher coverage of the search space, critical-scenarios are searched purposefully based on IPSO.

2.1 Framework of critical-scenarios search

The framework of the critical-scenarios search applied in this paper is shown in Figure 1. First, the search space of the logical scenario D is taken as the input of the optimization algorithm. Second, based on the D, the optimization algorithm searches for the parameter of the concrete scenario *d*. Then, according to *d*, the concrete scenario is generated through automatic testing (Chen *et al.*, 2020). Finally, calculated by the fitness function *f*, the evaluation result *r* of the SUT safety in the concrete scenario is taken as the output and returned to the optimization algorithm. According to the previous result, the optimization algorithm adjusts the *d* in the next iteration. By iterating and outputting the critical-scenarios based on the evaluation criteria, the critical-scenarios set is obtained.

2.2 Metric of coverage

For the critical-scenarios search, there is no method to measure the coverage of the search space as most of the research focuses on the search efficiency by counting the critical-scenarios in all samples. In this paper, the metric F1 is adopted to quantify the coverage of the search space. The method to measure the coverage is as follows.

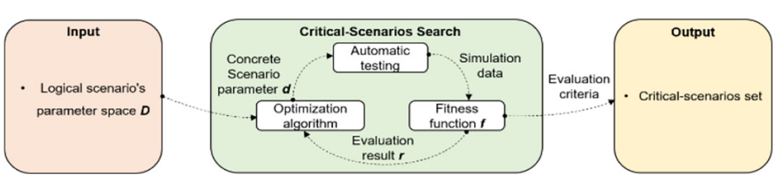
First, with the linear fitting method, the samples obtained by optimization algorithms and GS are transformed into continuous spaces. With the same high resolution, these continuous spaces are grided into the matrix, the matrix obtained from optimization algorithm is called "Fitted Matrix," and the ground truth obtained by GS is called "True Matrix.". Each grid in matrix is called pixel.

Second, the critical threshold is defined and the image mask operation is adopted. The pixel whose value is less than the threshold is defined as the critical-pixel whose flag is 1, and the others' flags are 0.

Finally, each pixel's flag of the True Matrix and Fitted Matrix are traversed and compared to obtain the "Result Matrix." The rule of pixel comparison is shown in Figure 2. If the pixels of the Fitted Matrix and True Matrix are both 1, the pixel at the same location of the Result Matrix will be denoted as true positive (TP). If the pixel of the Fitted Matrix is 0 and the True Matrix is 1, it will be denoted as false negative (FN). If the pixel of the Fitted Matrix and True Matrix are both 0, it will be denoted as true negative (TN). If the pixel of the Fitted Matrix is 1 and the True Matrix is 0, it will be denoted as false positive (FP). The number of critical-pixel in the True Matrix is denoted as T.

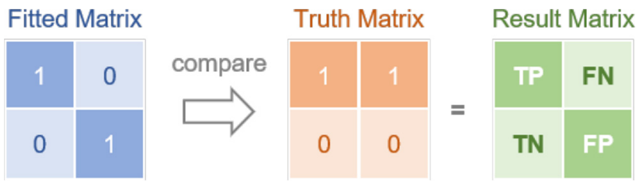
We choose the metric to reflect the critical-scenarios' coverage of the search space objectively. For the critical-scenarios search problem studied in this paper, the emphasis is the ratio of TP to T, which is the metric Recall. However, all pixels can be classified as the critical-pixels if the samples are few and critical, which keeps the Recall at 100% but is meaningless. Therefore, to ensure the accuracy of the fitting, we refer to the F1 as the metric of the coverage, which is the harmonic average of the Recall and Precision, and balance the two metrics well. Equations (1) and (2) are the definitions of the Recall and Precision, and equation (3) is defined to calculate the F1:

Figure 1 Framework of the critical-scenarios search



Journal of Intelligent and Connected Vehicles
Volume 5 · Number 3 · 2022 · 167–176

Figure 2 Rule of pixel comparison



$$\mathrm{Recall} = \frac{\mathrm{TP}}{\mathrm{T}} \tag{1}$$

$$\mathrm{Precision} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}} \tag{2}$$

$$\mathrm{F1} = \frac{2 \ast \mathrm{Recall} \ast \mathrm{Precision}}{\mathrm{Recall}+\mathrm{Precision}} \tag{3}$$

2.3 Principle of improved particle swarm optimization

To achieve better performance of the optimization algorithm for the critical-scenarios search, PSO is improved as IPSO in this paper. Because of its simple principle and fast convergency speed, PSO has been widely applied for optimization problems in many fields (Kennedy and Eberhart, 1995). However, PSO has the defect of premature convergence, which leads to an insufficient exploration of the search space (Suganthan, 1999). Therefore, IPSO is proposed for the critical-scenarios search in this paper. As IPSO will be applied to the test in the loop, which costs few seconds each round. By contrast, the time consumption of IPSO, which is about a few milliseconds, is negligible; therefore, the time complexity of the IPSO is compromised to improve the performance of IPSO. The principle of IPSO is shown in Figure 3, with the improvement in bold.

First, the position initialization mechanism of particle swarms is improved. When the randomly initialized particles are close in the space, it is easy for all particles to fall into the local optimum, which means premature convergence. Therefore, to avoid the above situation and pursue comprehensive exploration of the search space, the Latin hypercube sampling method (McKay *et al.*, 1979) is introduced into the initialization phase of the particles, which ensures the initial samples cover the search space uniformly and comprehensively. Compared with random sampling in the initial phase of PSO, the general picture of the search space globally with fewer samples can be obtained by the Latin hypercube sampling method, which improves the exploration ability.

Second, the velocity updating mechanism of particle swarms is improved. The neighborhood operator (Suganthan, 1999) is adopted in the iteration phase to update the velocity. The global optimal particle position is replaced with the neighborhood optimal particle position. Each particle can explore independently as the particles are divided into groups, avoiding the fast convergence speed caused by all particles moving toward the optimal particle. The above improvement enhances the exploration ability to find multiple modals. The improved velocity updating is defined in equation (4), where v is the velocity of the particle, k is the number of iterations, i is the serial number of the particle, d is the serial number of the dimension, w is the inertia factor, which shows how well the particle remembers its velocity in the last iteration, c_1 and c_2 are learning factors, which balance exploration and exploitation of the algorithm, r_1 and r_2 are the random numbers, which randomize the algorithm, x is the position of the particle, *pbest* is the best position of the particle in its personal history, *lbest* is the best position of the particle in its neighborhood. The calculation of neighborhood is described as follows: N is the total number of particles. The neighborhood of a particle is the hypersphere formed with the current position of the particle as the center and the distance S as the diameter, where S is $1/N$ of the maximum Euclidean distance of any two particles in the search space. The particle finds the best position in its neighborhood to update velocity at each iteration. If there is no better position in its neighborhood, the velocity will update only according to the best position of the particle in its personal history. To keep the balance between exploration and exploitation, some parameters are set according to suggestions of Zhang (2005) in this paper as follows: $N = 50$, $w = 0.8$, $c_1 = 1.5$, $c_2 = 1.5$:

$$\mathrm{V}_{\mathrm{id}}^{\mathrm{k}} = \mathrm{w}\mathrm{v}_{\mathrm{id}}^{\mathrm{k}-1} + \mathrm{c}_1\mathrm{r}_1\left(\mathrm{pbest}_{\mathrm{id}} - \mathrm{x}_{\mathrm{id}}^{\mathrm{k}-1}\right) + \mathrm{c}_2\mathrm{r}_2\left(\mathrm{lbest}_{\mathrm{d}} - \mathrm{x}_{\mathrm{id}}^{\mathrm{k}-1}\right) \tag{4}$$

Finally, convergence judgment and restart strategy are introduced (Huberman *et al.*, 1997). PSO converges after all particles fall into the local optimum, which is not conducive to the later exploration of the search space. Therefore, the Euclidean distance between any two particles will be calculated after the position updating at each iteration in IPSO. If the maximum distance between particles is lower than the threshold three consecutive times, the algorithm will be judged as convergence. Then in the next iteration, the Latin hypercube sampling method will be used to reset the positions of the particles, with the velocities of the particles reassigned randomly.

3. Design of experiments

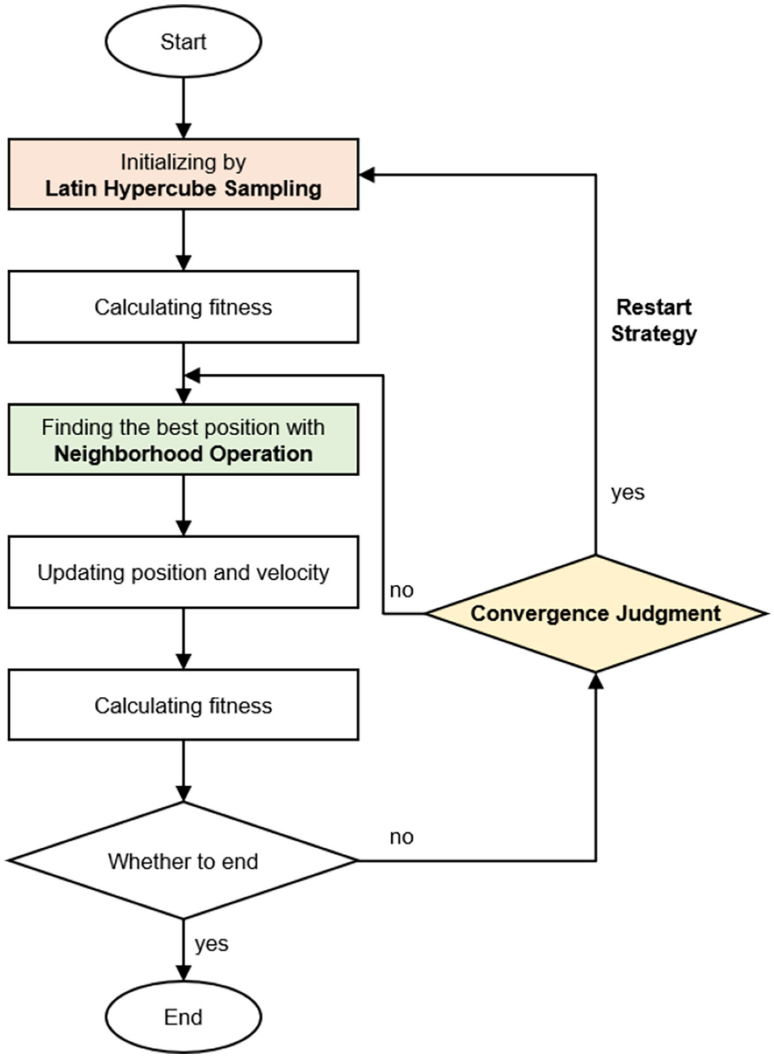
To verify the efficiency and the application effect of the method, we benchmark IPSO with PSO and MC on the multimodal artificial function and the practical issue of critical-scenarios search of AVs. To avoid contingency, the tests are repeated ten times. With the ground truth based on GS, the performance of IPSO and the other baseline algorithms is quantitatively evaluated by F1, the metric of the coverage.

3.1 Experiment on multimodal artificial function

We test algorithms on the Holder Table function (Bak *et al.*, 2019) for a fair and general evaluation. The Holder Table function has many local minima, with four global minima, which are used to verify each algorithm's ability to jump out of the local optimum and find modalities, and it can be seen as an extreme case of scenario search.

The Holder Table function is shown in equation (5), where x_1 is the value of the first dimension and x_2 is of the second:

Figure 3 Principle of IPSO



$$f(x) = -\left|\sin(x_1) * \cos(x_2) * e^{\left|1-\sqrt{\frac{x_1^2+x_2^2}{\pi}}\right|}\right| \tag{5}$$

To get the ground truth, we first test the Holder Table function with GS, which needs a trade-off between computation affordance and sampling accuracy. We discretize the search space of the Holder Table function and sample based on the parameters shown in Table 1.

By default, the range of parameters is set to the same of all dimensions.

Table 1 Parameters of the test on the Holder Table function

Type of parameters	Parameters of low dimensional space
Dimensionality	2
Range of parameters	−10 to 10
Grid size	100
Number of samples	10,000
Threshold	−18

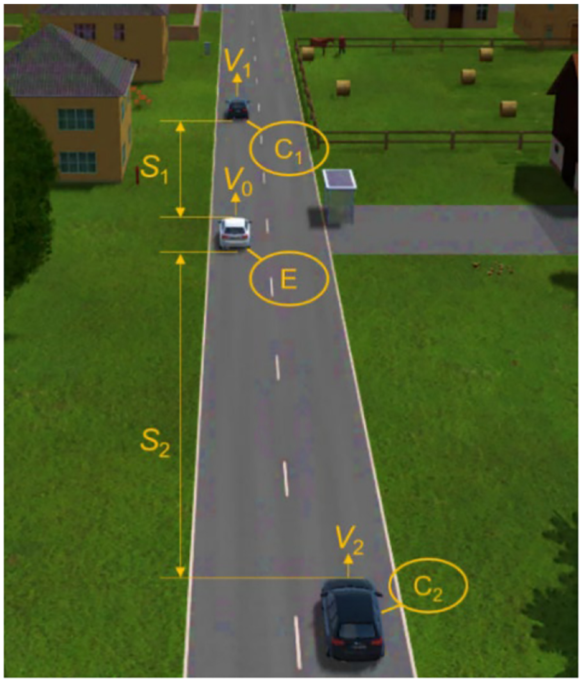
The budget for experiments with optimization algorithms is 3,000 samples for the two-dimensional search space.

3.2 Experiment on critical-scenarios search

To illustrate the application prospect, the proposed method is also applied to a practical issue of AVs' virtual test. First, the simulation scenario is constructed, and the search space is defined. Second, the fitness function is designed. Finally, virtual tests are carried out with the algorithms.

In this paper, the virtual environment is provided by the software Virtual Test Drive (Pilz *et al.*, 2019), and the autonomous driving SUT is also configured by it. The SUT has functions like emergency braking, emergency lane change and others with high intelligence. For the virtual test, a typical car-following scenario on a two-lane straight road is chosen, as shown in Figure 4. In the initial phase, the ego vehicle E is in the left lane and follows the front vehicle C_1 at a constant speed, while the vehicle C_2 in the right lane runs at a constant speed. After a while, C_1 brakes until standstill, and E responds to this emergency.

Figure 4 Initial phase of the car-following scenario



According to the natural traffic conditions and test requirements, the search space of the car-following scenario is set, as shown in Table 2. To facilitate the visualization of the results, the initial velocity of C_2 denoted as V_2, and the initial longitudinal distance between C_1 and E denoted as S_1 are used as variables for the test. With these two parameters evenly discretized into 101 parts as samples, the concrete scenarios for test are generated combined with other parameters.

The budget for experiments with optimization algorithms is 1,200 samples for the search space.

3.3 Fitness function for critical-scenarios search

The fitness function of the optimization problem needs to reflect the goal of the optimization objectively. In the issue of critical-scenarios search, the fitness function should meet the test requirements to represent the criticality of the scenarios accurately, which is beneficial for V&V of AVs.

The fitness function is designed based on the safety evaluation metric as this paper focuses on the safety of the decision-making system. The ideal critical-scenarios in the virtual test are the dangerous scenarios in which the ego car emergency brake, so time-to-collision (TTC) (Vogel, 2003) of the ego car is selected as the fitness function. TTC is the ratio of the relative distance and relative speed of two vehicles, which is used to represent the potential collision danger in the car-following scenario. The smaller the TTC produced by the SUT, the more dangerous the scenario is. Therefore, we take the smallest TTC value in each scenario of virtual tests and output it to the optimization algorithm for iteration. In addition, there are situations where V_0 is smaller than V_1 all the time before E changes lanes, in which TTC is increasing. So, the value of the fitness function is set as 20 s at the beginning of the scenario. If the TTC is always higher than 20 s in the test, we claim that the scenario is safe and output the fitness value the same as the initial value.

Based on the pre-safe brake system of Daimler (Schöneburg *et al.*, 2019), it is dangerous for the vehicle if TTC is smaller than 0.6 s. To calculate the coverage, the threshold of TTC is set as 0.6 s for the experiment. For a more dangerous scenario, we also set the threshold as 0.3 s to see the change of the coverage.

4. Results and discussion

4.1 Results of multimodal artificial function

The True Matrix of the Holder Table function is shown in Figure 5. The deeper the red region is, the smaller the function value is, which means the sample is more critical. Sample distribution of all algorithms after 3,000 iterations is shown in Figure 6. It shows that IPSO searches for four modalities more comprehensively than MC and PSO.

The result of coverage is shown in Figure 7, and the analysis is as follows:

IPSO keeps the highest coverage during iterating, which means the search efficiency of IPSO is higher than PSO and MC. For example, when the coverage is 40%, which means at least one modality has been searched, IPSO takes about 750 samples, while PSO and MC take about 3,000 samples, meaning the search efficiency of IPSO is 83% higher than PSO and 158% than MC.

With 3,000 samples, the coverage of IPSO is about 84%, which is 40% higher than PSO and MC. It shows the potential of IPSO in the multimodal critical-scenarios search.

4.2 Results of critical-scenarios search

Results of the virtual test with GS are shown in Figure 8. The white array points in Figure 8 are 2,601 in total, representing the scenarios for the virtual test. Based on the test results of the scenarios, the linear fitting method is adopted to calculate the

Table 2 Search space of the car-following scenario

Definition of parameters	Symbol of parameters	Range of parameters	Unit
Initial speed of E	V_0	30	m/s
Initial speed of C_1	V_1	20	m/s
Initial speed of C_2	V_2	10–30	m/s
Initial longitudinal distance between E and C_1	S_1	10–110	m
Initial longitudinal distance between E and C_2	S_2	50	m
The time C_1 starts to brake	T_B	4	s
The duration of C_1 breaking	T	3	s
Duration of a scenario	T_L	20	s

Figure 5 True Matrix of the Holder Table function (10,000 samples)

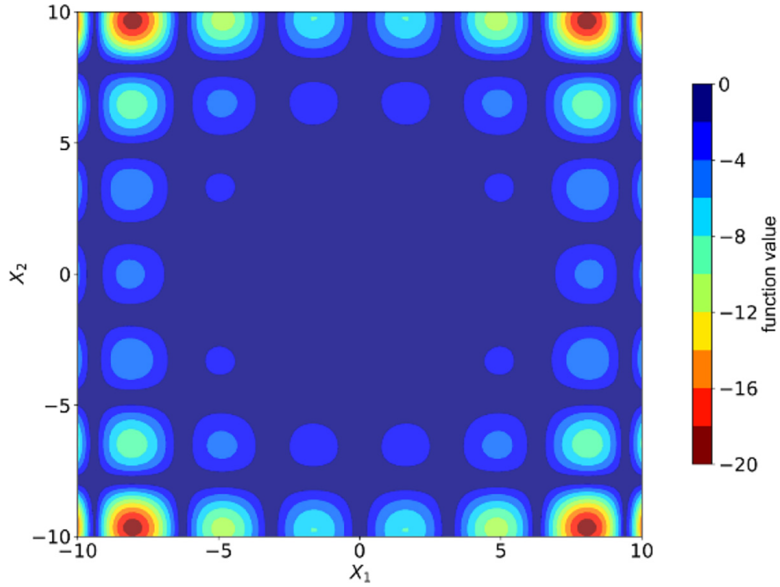
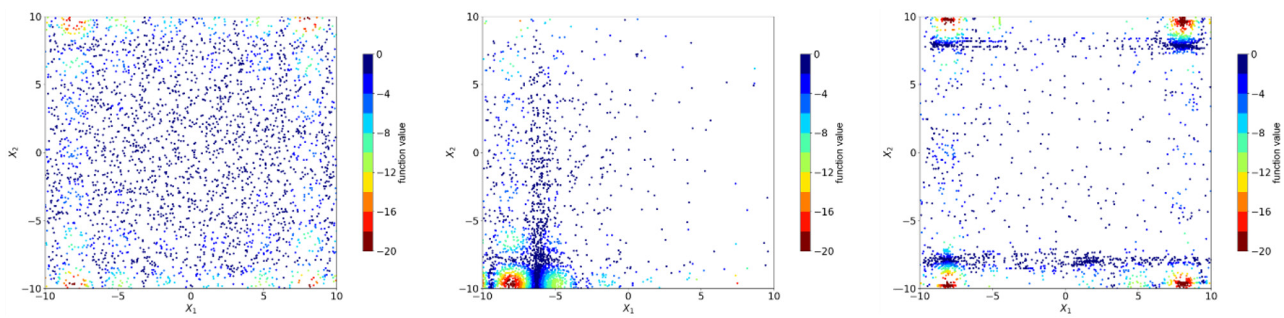


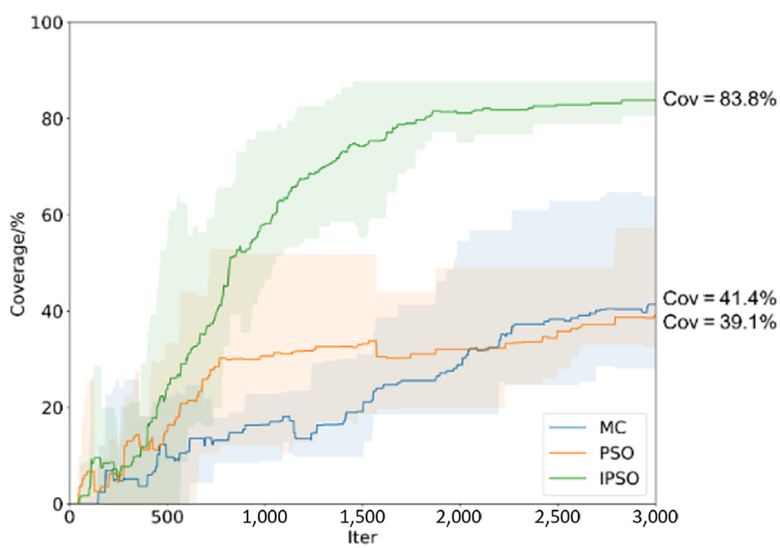
Figure 6 Samples distribution of algorithms (Holder Table, 3,000 samples)



(a) (b) (c)

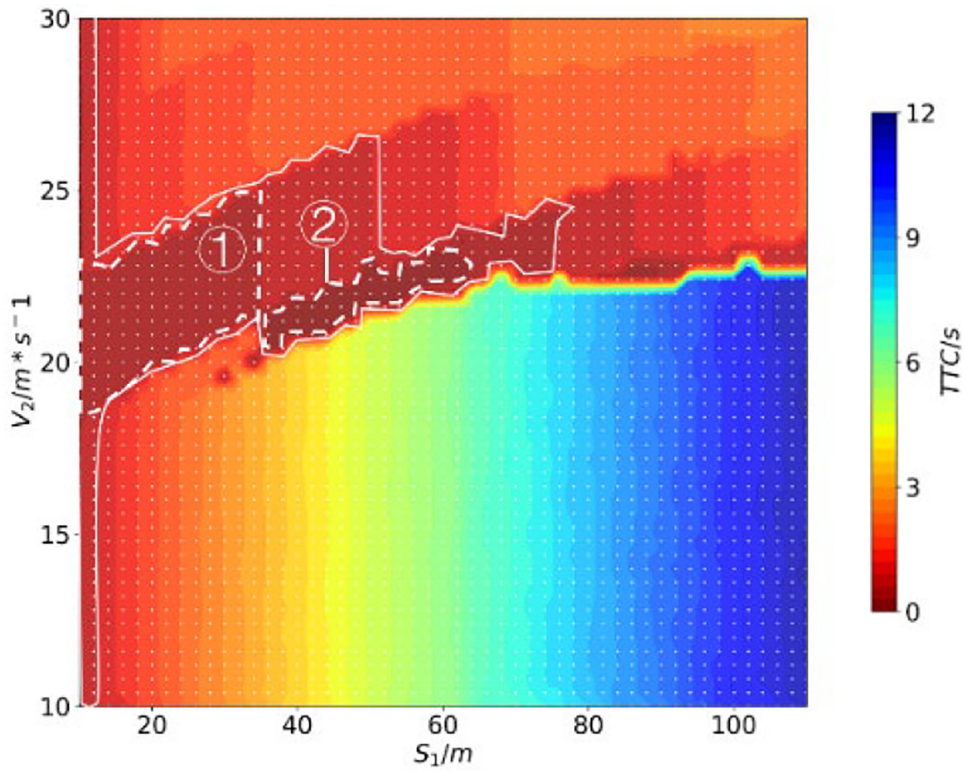
Notes: (a) MC; (b) PSO; (c) IPSO

Figure 7 Coverage of algorithms (Holder Table, 3,000 samples)



Journal of Intelligent and Connected Vehicles
Volume 5 · Number 3 · 2022 · 167–176

Figure 8 True Matrix of two-dimensional scenario (2,601 samples)



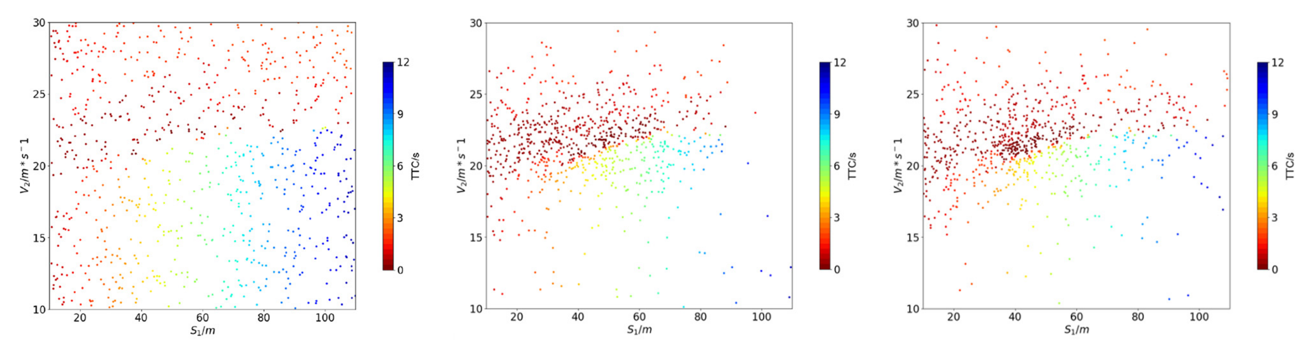
True Matrix shown in Figure 8. The deeper the red region is, the smaller the TTC is, which means the scenario is more critical. To compare the algorithms intuitively, sample distribution of all algorithms after 800 iterations is shown in Figure 9.

With the threshold set as 0.3 s, two modalities of scenarios are found, as shown in the white dashed outline in Figure 8. Therefore, the test processes of these two modalities of the critical-scenarios are observed. In the first modality in dotted area 1, E brakes immediately because V_0 is greater than V_1, and S_1 is small in the initial phase. The second modality in dotted area 2 is shown in Figure 10, which is divided into six phases: In the first phase, the scenario initializes (Figure 10a); in the second phase (Figure 10b), C_1 brakes; in the third phase (Figure 10c), E starts to brake when it approaches C_1, and E cannot change lanes immediately as C_2 is close to it. In the fourth phase (Figure 10d), E decelerates slowly and approaches C_1 until C_2 passes C_1. In the fifth phase (Figure 10e), E changes lanes as the next lane is unoccupied; in the sixth phase (Figure 10f), E completes lane changing and continues to go straight.

The two modalities of the critical-scenarios discovered based on the True Matrix are both dangerous enough to meet the test requirements.

With the threshold set as 0.6 s, a bigger region that includes the above scenarios is filtrated, as shown in the white solid line outline in Figure 8.

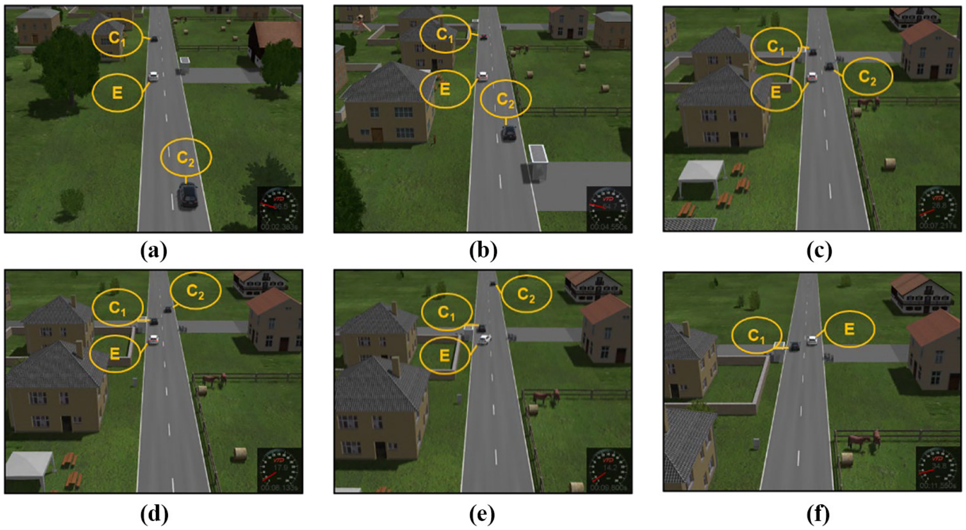
Figure 9 Samples distribution of algorithms (two-dimensional scenario, 800 samples)



(a) **(b)** **(c)**

Notes: (a) MC; (b) PSO; (c) IPSO

Figure 10 Test process of the second modality of the critical-scenarios



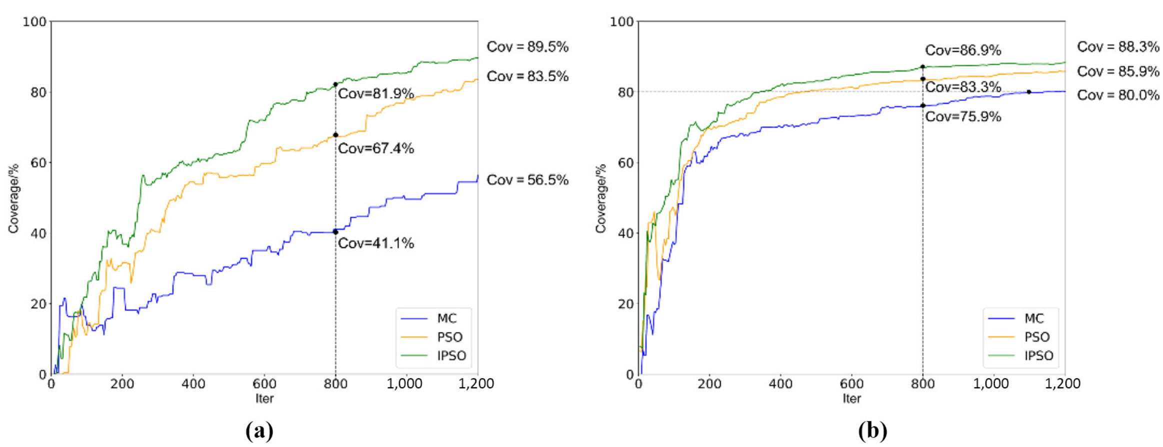
Notes: (a) initialization; (b) C1 braking; (c) E approaching C1; (d) C2 passing C1; (e) Echanging lane; (f) E keeping straight

According to calculating the critical-scenarios' coverage with different thresholds, the virtual test results of the two-dimensional scenario are shown in Figure 11. The analysis is as follows:

Overall, with the increase of iterations, the coverage of each algorithm shows a rising trend. However, at the beginning of the iteration, the coverage fluctuates wildly, which is considered to be due to the inaccurate fitting with few initial samples. The coverage of IPSO increases rapidly in the initial phase, and we believe it is due to the improved initialization mechanism based on the Latin hypercube sampling method.

- With 800 samples, the coverage of IPSO is about 14% higher than PSO and 40% higher than MC when the threshold is set as 0.3 s, and 3% higher than PSO and 9% higher than MC when the threshold is set as 0.6 s. It shows that the stricter the threshold is, the fewer the critical-scenarios that are in the search space, which means the search is more difficult; therefore, more highlights the advantages of IPSO. Because IPSO searches for the critical-scenarios more comprehensively and quickly by the initialization mechanism combined with the improved velocity updating mechanism based on neighborhood operator.
- With 1,200 samples, the coverage of IPSO is about 6% higher than PSO and 33% higher than MC when the threshold is set as 0.3 s, and 3% higher than PSO and 8% higher than MC when the threshold is set as 0.6 s. It is analyzed that when the threshold is set as 0.3 s, the regions of critical scenarios are close, which reduces the difficulty of search compared with the four scattered regions of the Holder Table.

Figure 11 Coverage of algorithms (two-dimensional scenario, 1,200 samples)



Notes: (a) Threshold = 0.3 s; (b) threshold = 0.6 s

In summary, IPSO shows a better performance on efficiency and coverage in the multimodal critical-scenarios search. With a stricter threshold (0.3 s) and fewer samples (800), the coverage of IPSO is 14% higher than PSO and 40% higher than MC, which means IPSO can accelerate the process of AVs' V&V effectively.

5. Discussion

5.1 Trade-off between exploration and exploitation

The "no free lunch" theorem tells us no algorithm performs well on any optimization problems. We need to trade off between exploration and exploitation based on the problem. For the critical-scenarios search, the algorithm's exploration ability should be improved to guarantee AVs' safety. However, most of the research adopts the optimization algorithm to search for scenarios with more exploitation than exploration, which may lead to poor algorithm performance on this problem.

5.2 Cloud computing platforms with swarm intelligent algorithms

With the popularity of cloud computing platforms, swarm intelligence algorithms will become more efficient in critical-scenarios search than other optimization algorithms like Bayesian optimization as the former can realize parallel computing, but the latter relies more on the prior knowledge when searching the critical-scenarios.

6. Conclusions

In this paper, we transformed the critical-scenarios search into an optimization problem and proposed IPSO to search modalities of critical-scenarios with F1 quantifying the coverage. To enhance the exploration ability for a higher coverage, PSO was improved in three phases. The metric F1 was firstly adopted to quantify the critical-scenarios' coverage in detail by comparing fitted space with ground truth pixel-by-pixel, and it was used to evaluate the optimization algorithms' performance in critical-scenarios search.

To verify the efficiency and the application prospect of the method, experiments on the multimodal artificial function and critical-scenarios search were carried out with MC and PSO as baselines. Results of the artificial function show that the proposed method can significantly improve the search effectiveness and critical-scenarios' coverage in the multimodal search space. Results of the critical-scenarios search of the decision-making system show that the proposed method can search for multimodal critical-scenarios of AVs effectively, which accelerates the process of V&V and guarantees the automated driving safety.

In future work, we will focus on the following aspects: First, for the neighborhood operation in IPSO, the constant hypersphere results in slow convergence, which may weaken the exploitation ability. Variable hypersphere for the neighborhood operation will be designed further to improve the tunability of exploration ability. Second, calculation of the coverage in high dimensional search space is faced with a heavy burden in matrix comparison. A new metric will be studied for the high dimensional search space.

References

Bak, E.N., Saraçoğlu, R. and Şehribanoğlu, S. (2019), "Optimization of holder table test function using dynamic mutated genetic algorithm", *Proceedings Book*, p. 221.

Batsch, F., Daneshkhah, A., Cheah, M., Kanarachos, S. and Baxendale, A. (2019), "Performance boundary identification for the evaluation of automated vehicles using gaussian process classification", *Intelligent Transportation Systems (ITSC), 2019 IEEE Intelligent Transportation Systems Conference on IEEE*, pp. 419-424.

Beglerovic, H., Stolz, M. and Horn, M. (2017), "Testing of autonomous vehicles using surrogate models and stochastic optimization", *Intelligent Transportation Systems (ITSC), 2017 IEEE 20th International Conference on IEEE*, pp. 1-6.

Chen, J., Feng, T., Liu, L. and Wu, X. (2020), "An automatic generation method of concrete scenario for decision-making and planning system test", *Automobile Technology*, Vol. 36, pp. 45-50.

Feng, S., Feng, Y., Sun, H., Bao, S., Zhang, Y. and Liu, H.X. (2020), "Testing scenario library generation for connected and automated vehicles, part I: methodology ", *IEEE Transactions on Intelligent Transportation Systems*, Vol. 22 No. 3, pp. 1573-1582.

Huberman, B.A., Lukose, R.M. and Hogg, T. (1997), "An economics approach to hard computational problems", *Science*, Vol. 275 No. 5296, pp. 51-54.

Kalra, N. and Susan, M.P. (2016), "Driving to safety: how many miles of driving would it take to demonstrate autonomous vehicle reliability?", *Transportation Research Part A: Policy and Practice*, Vol. 94, pp. 182-193.

Kennedy, J. and Eberhart, R. (1995), "Particle swarm optimization", *International Conference on Neural Networks, ICNN 95, Proceeding of the IEEE*, IEEE, pp. 1942-1948.

Klischat, M., Liu, E.I., Holtke, F. and Althoff, M. (2020), "Scenario factory: creating safety-critical traffic scenarios for automated vehicles", *Intelligent Transportation Systems (ITSC), 2020 IEEE 23rd International Conference on IEEE*, pp. 1-7.

McKay, M.D., Beckman, R.J. and Conover, W.J. (1979), "A comparison of three methods for selecting values of input variables in the analysis of output from a computer code", *Technometrics*, Vol. 42, pp. 239-245.

Masuda, S., Nakamura, H. and Kajitani, K. (2018), "Rule-based searching for collision test cases of autonomous vehicles simulation", *IET Intelligent Transport Systems*, Vol. 12 No. 9, pp. 1088-1095.

Menzel, T., Bagschik, G. and Maurer, M. (2018), "Scenarios for development, test and validation of automated vehicles", intelligent vehicles symposium, IV 2018", *Proceeding of the IEEE*, IEEE, pp. 1821-1827.

Mullins, G.E., Stankiewicz, P.G. and Hawthorne, C.R. (2018), "Adaptive generation of challenging scenarios for testing and evaluation of autonomous vehicles", *Journal of Systems and Software*, Vol. 137, pp. 197-215.

Pilz, C., Steinbauer, G., Schratter, M. and Watzenig, D. (2019), "Development of a scenario simulation platform to support autonomous driving verification", *International Conference on Connected Vehicles, ICCVE, 2019 IEEE*

International Conference on Connected Vehicles and Expo on IEEE, IEEE, pp. 1-7.

Schöneburg, R. Hart, M. Feese, J. Mücke, S. and Richert, J. (2019), "ESF 2019 – experimental safety vehicle meets automated driving mode", Enhanced Safety of Vehicles, ESV 2019, no. 19-0042.

Schuldt, F., Reschka, A. and Maurer, M. (2018), "A method for an efficient, systematic test case generation for advanced driver assistance systems in virtual environments", *Automotive Systems Engineering II*, Springer, pp. 147-175.

Suganthan, P.N. (1999), "Particle swarm optimiser with neighbourhood operator", *Evolutionary Computation, CEC 99, Proceedings of the 1999 Congress on Evolutionary Computation*, pp. 1958-1962.

Tuncali, C.E., Fainekos, G., Ito, H. and Kapinski, J. (2018), "Simulation-based adversarial test generation for autonomous vehicles with machine learning components", *Intelligent Vehicles Symposium, IV 2018, Proceeding of the IEEE*, IEEE, pp. 1555-1562.

Vogel, K. (2003), "A comparison of headway and time to collision as safety indicators", *Accident Analysis & Prevention*, Vol. 35 No. 3, pp. 427-433.

Wolpert, D.H. and Macready, W.G. (1997), "No free lunch theorems for optimization", *IEEE Transactions on Evolutionary Computation*, Vol. 1 No. 1, pp. 67-82.

Zhang, L. (2005), *The Theorem and Practice upon the Particle Swarm Optimization Algorithm*, Zhejiang University, January.

Zhu, B., Zhang, P., Zhao, J., Chen, H. and Xu, Z. (2019), "Review of scenario-based virtual validation methods for automated vehicles", *China Journal of Highway and Transport*, Vol. 32, pp. 1-19.

Zhu, B., Zhang, P., Zhao, J. and Deng, W. (2021), "Hazardous scenario enhanced generation for automated vehicle testing based on optimization searching method", *IEEE Transactions on Intelligent Transportation Systems*, pp. 1-11.

Corresponding author

Junyi Chen can be contacted at: chenjunyi@tongji.edu.cn