Hindawi
Mathematical Problems in Engineering
Volume 2019, Article ID 8164609, 10 pages
https://doi.org/10.1155/2019/8164609

Research Article

Application of Improved Particle Swarm Optimization in Vehicle Crashworthiness

Dawei Gao, Xiangyang Li, and Haifeng Chen

School of Mechanical Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China

Correspondence should be addressed to Dawei Gao; gddwww1999@163.com

Received 20 December 2018; Revised 17 February 2019; Accepted 24 February 2019; Published 17 March 2019

Academic Editor: Leandro F. F. Miguel

In the optimization design process, particle swarm optimization (PSO) is limited by its slow convergence, low precision, and tendency to easily fall into the local extremum. These limitations make degradation inevitable in the evolution process and cause failure of finding the global optimum results. In this paper, based on chaos idea, the PSO algorithm is improved by adaptively adjusting parameters r_1 and r_2. The improved PSO is verified by four standard mathematical test functions. The results prove that the improved algorithm exhibits excellent convergence speed, global search ability, and stability in the optimization process, which jumps out of the local optimum and achieves global optimality due to the randomness, regularity, and ergodicity of chaotic thought. At last, the improved PSO algorithm is applied to vehicle crash research and is used to carry out the multiobjective optimization based on an approximate model. Compared with the results before the improvement, the improved PSO algorithm is remarkable in the collision index, which includes vehicle acceleration, critical position intrusion, and vehicle mass. In summary, the improved PSO algorithm has excellent optimization effects on vehicle collision.

1. Introduction

With the increase of car ownership, traffic accidents have occurred from time to time, and the passive safety performance of cars in collisions has attracted more and more attention. Improve the crashworthiness of vehicles and reduce the occupant injury risk in the traffic accident and property loss becomes an important goal of automotive design. Therefore, many researchers and manufacturers have conducted research on improving the collision safety of vehicles. At present, the crashworthiness of the vehicle is mainly improved in the following three ways. Improve manufacturing processes, optimize structural design, and adjust material usage. Among them, the use of higher quality materials and improved processing technology can significantly improve the collision performance, but it will also lead to an inevitable increase in manufacturing costs. While in practical applications, optimizing the body design, especially the structural design of the front of the car, is the most widely used improvement. In the front part of the car, the structure of the bumper, energy absorbing box, front longitudinal beam, fender, hood, and other components will impact the safety of the collision, which requires the use of algorithms in the design process, to carry out multiobjective optimization on these structures [1, 2].

To this end, many researchers have contributed to this aspect, and a variety of intelligent algorithms have emerged. Fonseca and Fleming [3] first proposed a multiobjective genetic algorithm based on Pareto optimal concept in 1993. Later, Sun et al. [4] used the nondominated sorting genetic algorithm (NSGA-II) to optimize the shape of the thin-walled tube, which improved the absorbed energy and reduced the peak impact force. Kirkpatrick [5] proposed a simulated annealing algorithm for large-scale combination optimization problems in 1983. Suppapitnarm et al. [6] improved the optimization performance of the algorithm in engineering practice by improving the acceptance criteria and cooling accuracy table of the simulated annealing algorithm. The Italian scholar Dorigo [7] was inspired by the ant colony searching for food behavior in nature and proposed the ant colony algorithm (ACA). Karl [8] et al. further introduced the concept of pheromone vector into the ant colony algorithm to coordinate the relationship between the various targets and deal with multiobjective asset selection. The particle swarm algorithm appeared relatively late, which was first used by Jacqueline [9] in multiobjective optimization in 1999. Fang [10] et al. used the particle swarm optimization algorithm together with the response surface approximation model to optimize the multiobjective reliability of the vehicle door and achieved good results.

In recent years, several intelligent algorithms, such as particle swarm optimization (PSO), genetic algorithms, and simulated annealing algorithms, have received widespread attention in the area of scheduling [11–13]. Research shows that these algorithms suffer from defects when applied to complex optimization problems. Hence, the optimization results for a single technique tend to be unsatisfactory. Therefore, algorithm improvements based on fusion ideas have been proposed by domestic and foreign scholars [14, 15].

PSO has several advantages, including fast convergence, few setting parameters, and simple and easy implementation; hence, it can be used to solve nonlinear, nondifferentiable, and multipeak optimization problems, particularly in science and engineering fields [16, 17]. However, the algorithm suffers from major disadvantages, including "premature" phenomena and "unripe" phenomena [18, 19]. The insufficient convergence speed and insufficient accuracy of PSO hinder it from meeting requirements. In addition, its global search ability is poor, and it easily falls into the local extremum [20]. These drawbacks make degradation inevitable in the evolution process and cause the failure of finding the global optimum results. The features exhibited by chaotic thoughts show that chaotic sequences appear to be particularly random, the tracks of chaotic variables are regular, and the entire search space can be searched without repetition; therefore, chaotic sequences are superior to blind random search and avoid the disadvantage of evolutionary algorithms of falling into the local optimum [21, 22].

It is because of the aforementioned defects of the PSO algorithm that the present study introduces the chaos idea and proposes an improved PSO algorithm that uses chaos theory to dynamically adjust its parameters r_1 and r_2. In order to verify the effect of the method in actual optimization, eight sheets which have great influence on the frontal collision safety performance of a minibus were selected, and the thickness of the eight panels was used as the design variable. The peak acceleration a_{max} at the lower end of the B-pillar, the mass m of the vehicle, the backward intrusion amount D1 at the dashboard tube beam, the backward intrusion amount D2 at the steering column hole of the dash panel, and the backward intrusion amount D3 at the clutch plate of the lower dash panel were used as a study response. The improved particle swarm optimization algorithm is used for multiobjective optimization design. Compared with the traditional particle swarm optimization algorithm, the results show that the improved one is superior to the traditional particle swarm optimization algorithm in terms of optimization effect and solution speed.

2. Improved PSO Algorithm

In 1995, a group of researchers led by Kennedy and Eberhart developed a new calculation method for the study of bird predation behavior. This calculation method uses the iterative changes of particles to simulate bird predation behavior and searches a certain number of particles for optimal particles in the spatial solution set [23, 24].

The PSO algorithm is similar to bird predation. Each solution to the optimization problem is similar to a bird that is preying on a defined space; in this scenario, the bird is the "particle." Each particle has its own position, and the ith particle position is expressed as $x_i = (x_{i1}, x_{i2}, \ldots, x_{iD})$. The speed of flight is expressed as $= \ (v_{i1}, v_{i2}, \ldots, v_{iD})$. Moreover, each particle has an adaptive value determined by the objective function. In each iteration of the particle swarm, the particle needs to determine two extreme values: the individual extremum pbest (the optimal solution found by the particle itself) and the global extremum gbest (the optimal solution found by the group). After determining these two extreme values, the particle's d-dimensional (1 ≤ d ≤ D) velocity v_{id} and location x_{id} are updated according to the following equation:

$$\begin{aligned} v_{id}(t+1) = \omega \times v_{id}(t) + c_1 \times r_1 \times \left[\mathrm{P}_{id}(t) - \chi_{id}(t)\right] \\ + c_2 \times r_2 \times \left[\mathrm{P}_{gd}(t) - \chi_{id}(t)\right] \end{aligned} \tag{1}$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1), \tag{2}$$

where ω is the inertia weight, c_1 and c_2 are acceleration constants, and r_1 and r_2 are random values that vary between $[0, 1]$.

2.1. Previous Algorithm Defect. The PSO algorithm can easily fall into the local optimal solution in the search process. According to (1), when $x_{id} = p_{id} = p_{gd}$, the flight speed of the particles depends only on ω and v_{id}. If the values of v_{id} and ω are unequal to zero, then $x_i(t+1) \neq x_i$, and the particles fly away from the original trajectory. If v_{id} is equal to zero, then $v_{id}(t+1)$ is also equal to zero. Once all the particles achieve p_{gd}, the particle stops flying and converges to the local optimal solution.

At each time t, the particles search within a certain range of space under the influence of group information and their own historical experience. When a superior solution is found, the guidance information is updated, a new search is launched, and the optimization process is continued. However, the convergence speed in the late search slows down given that the flight direction of all particles is determined according to their entire experience. After reaching a certain level, the algorithm falls into stagnation, thereby hindering the achievement of an accurate solution.

Although some achievements have been made in the convergence analysis of the PSO algorithm, the premise is based on the assumption that the individual and global extremum values are constant. For the relevant conclusions of parameter selection, the proposed parameter values are highly unsatisfactory in practical applications despite the existence of a certain reference role. According to the basic principle of the PSO algorithm, correlations exist among the method parameters, especially the inertia weight and learning factor. Their mutual influence is also highly important.

However, this mutual influence is often ignored, which leads to debatable results [25]. The experiment study [26] found that the bigger the value of ω, c_1, and c_2, the greater the probability of no convergence, and the value of ω has the largest effect. Choosing the appropriate ω, c_1, and c_2 can thus guarantee the convergence of particles. Therefore, it is necessary to improve the global search ability of the PSO algorithm. The optimization of vehicle collision performance requires an algorithm with a fast running speed to meet the scheduling aging requirements and rapid convergence to the global optimal solution to ensure scheduling accuracy. Therefore, the PSO algorithm needs to be improved such that it eliminates inferior solutions as soon as possible, avoids overiteration near these inferior solutions, and solves the collision performance optimization model [27] with many decision variables and complex scenes.

2.2. Algorithm Improved by Chaos Thought. Chaos is a universal nonlinear phenomenon whose behavior is complex and similar to random phenomena but has delicate internal regularity. Optimization search using chaotic variables is superior to blind random search as it can avoid the disadvantage of evolutionary algorithms of falling into the local optimum due to the ergodicity of chaos [28]. A chaotic idea is used to adaptively adjust the parameters related to particle velocity update. When generating chaotic sequences, the following logistic model is used:

$$\lambda_i^{t+1} = \mu \times \lambda_i^t \times \left(1 - \lambda_i^t\right); \quad i = 1, 2, \ldots, \text{n}, \tag{3}$$

where λ_i^t is the value of λ_i after the chaotic evolution in step t, $\lambda_i \in [0, 1]$, and $1 \le \mu \le 4$. When $\mu = 4$ and λ_i is not 0.25, 0.5, and 0.75, the system exhibits complete chaotic characteristics, the generated chaotic sequence shows excellent randomness, and the trajectory of the chaotic variable can traverse the entire search space without repetition. The chaotic optimization of parameters r_1 and r_2 is as follows:

$$\begin{gathered} r_i\,(\text{t}+1) = 4.0 \times r_i\,(\text{t}) \times \left(1 - r_i\,(\text{t})\right) \\ r_i\,(\text{t}) \in (0, 1.0), \quad i = 1, 2. \end{gathered} \tag{4}$$

The inertia weight ω can balance the global and local searches of the PSO algorithm by controlling the effect of historical speed on the current velocity of the particle. Simultaneously, a suitable ω can diminish the time needed to find the optimal solution. The inertia weight setting for each particle is as follows:

$$\omega\,(\text{t}) = \omega_{max} - \left(\omega_{max} - \omega_{min}\right) \times \frac{t}{t_{max}}, \tag{5}$$

where ω_{max} and ω_{min} are the maximum and minimum values of the inertia weight, respectively, and t and t_{max} are the current algebra and maximum number of iterations, respectively [29]. The execution process of chaotic particle swarm optimization algorithm is as follows: Firstly, the chaotic idea is used to dynamically adjust the parameters r_1 and r_2 of the particle swarm optimization algorithm to produce a good group. Then, in the evolution, formula (1) and formula (2) are used to lead the particle swarm to the optimal solution search.

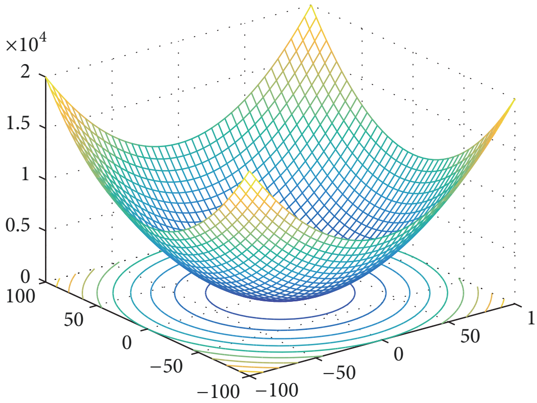


Figure 1: 3D graphics of Sphere function.

2.3. Performance Test of Improved Algorithm. This study uses four typical mathematical standard test functions to test and evaluate the improved PSO algorithm and thus verify its optimization performance.

2.3.1. Sphere Function

$$f\,(x) = \sum_{i=1}^{n} x_i^2. \tag{6}$$

The function is a unimodal one that is mainly used to test the optimization precision of the algorithm. The search space and global optimal solution are $-100 \le x_i \le 100$ and $f(x) = 0$; $x_i = 0$, respectively. The 3D graphical representation of the function is shown in Figure 1.

2.3.2. Rosenbrock Function

$$f\,(x) = \sum_{i=1}^{n-1} \left(100\left(x_i^2 - x_{i-1}\right)^2 + \left(x_i - 1\right)^2\right). \tag{7}$$

This function is a multipeak test one. A particularly narrow trough exists between the local optimum and the global optimal solution. Moreover, the function is commonly used to test the execution performance of the algorithm. The search space and global optimal solution are $-10 \le x_i \le 10$ and $f(x) = 0$; $x_i = 0$, respectively. The 3D graphical representation of the function is shown in Figure 2.

2.3.3. Rastrigin Function

$$f\,(x) = 100n + \sum_{i=1}^{n} \left[x_i^2 - 10\cos\left(2\pi x_i\right)\right]. \tag{8}$$

The function is a multipeak test function, and the distribution of minimum values is regular. The search space and global

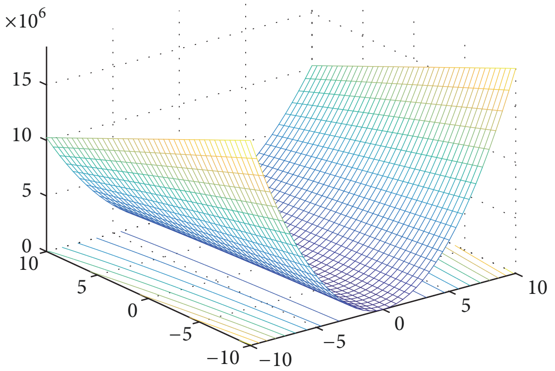


Figure 2: 3D graphics of Rosenbrock function.

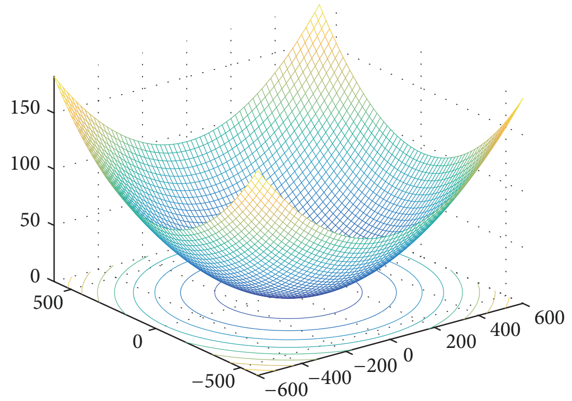


Figure 4: 3D graphics of Griewank function.

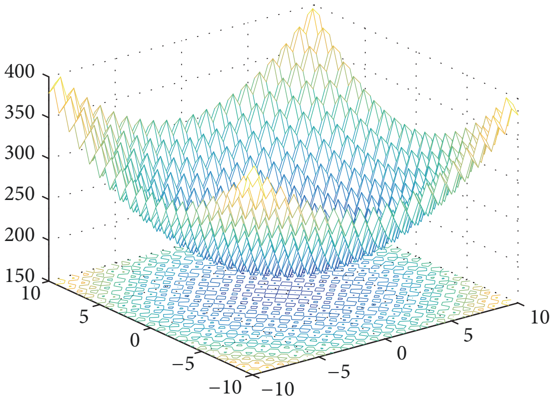


Figure 3: 3D graphics of Rastrigin function.

optimal solution are $-10 \leq x_i \leq 10$ and $f(x) = 0$; $x_i = 0$. The 3D graphical representation of the function is shown in Figure 3.

2.3.4. Griewank Function

$$f(x) = \frac{1}{4000}\sum_{i=1}^{n} x_i^2 + \prod_{i=1}^{n} \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1. \qquad (9)$$

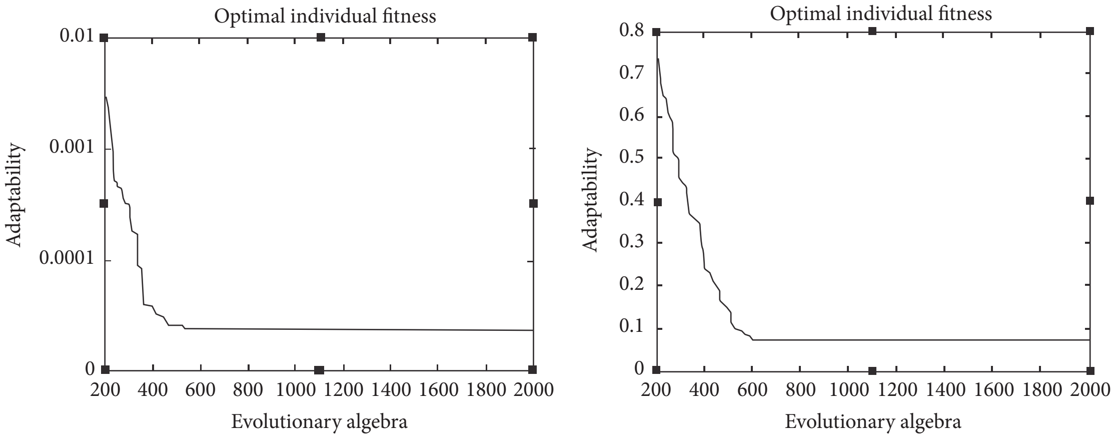
Similar to the Rastrigin function, the Griewank function has multiple minimum values that exhibit a regular distribution. The search space and global optimal solution are $-600 \leq x_i \leq 600$ and $f(x) = 0$; $x_i = 0$, respectively. The 3D graphical representation of the function is shown in Figure 4.

The particle swarm and improved PSO algorithms are applied to run the four test functions. The initial parameters of the two algorithms are set as follows: the inertia weight of the particle swarm algorithm is 0.9, and the acceleration constant is $c_1 = c_2 = 1.4$. The maximum and minimum inertia weight values of the improved PSO algorithm are $\omega_{max} = 0.9$ and $\omega_{min} = 0.4$, respectively; the acceleration constant is $c_1 = c_2 = 1.4$; and $\mu = 4$. The two optimization algorithms have 0.5 step size, 2000 iterations, and 10 particles. Each test function is tested 20 times, and the optimal and worst solutions and the average value of the solution are obtained. The performance comparison between the two algorithms under different test functions is shown in Table 1. At the same time, the average value of the fitness function curve obtained in 20 tests was taken to generate a fitness function curve as shown in Figures 5–8.

The test results are shown in Figures 5–8. For the Sphere function, the improved PSO algorithm presents a remarkable reduction in the fitness value over the PSO algorithm. By the improved algorithm, the fitness value is close to zero, and the convergence speed is slightly increased. For the Rosenbrock function, the performance of the improved PSO algorithm is slightly improved, and the fitness value is diminished. However, the convergence speed of the improved PSO algorithm under this function is not as good as that of the PSO algorithm. For the Rastrigin function, the improved PSO algorithm is excellent in terms of fitness value and convergence speed. In addition, a large degree of improvement is observed. For the Griewank function, the improved PSO algorithm is slightly better than the PSO algorithm in terms of fitness value and convergence speed, but the difference is average. Evidently, for a single optimization, the improved PSO algorithm is better than the PSO algorithm under the four test functions.

As shown in Table 1, for the four test functions, the optimal and average solutions of the improved PSO algorithm are better than those of the PSO algorithm. Moreover, the difference between the optimal and worst solutions obtained by the improved PSO algorithm is smaller than that obtained by the PSO algorithm. These results indicate that the improved PSO algorithm is more stable than the PSO algorithm is in the optimization process.

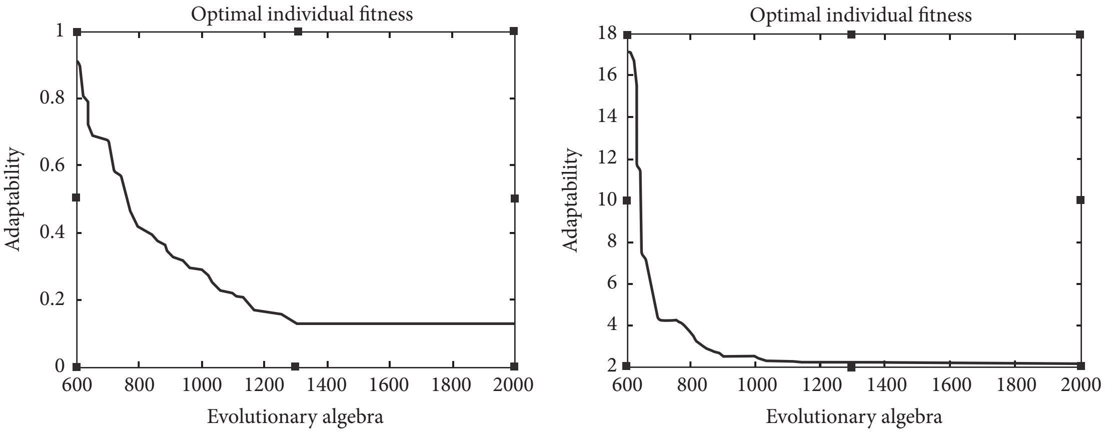
Based on the four common test functions, the improved PSO algorithm proposed in this study exhibits an improvement in global search ability and search speed more than the PSO algorithm.



(a) Optimal individual fitness for improved PSO algorithm

(b) Optimal individual fitness for PSO

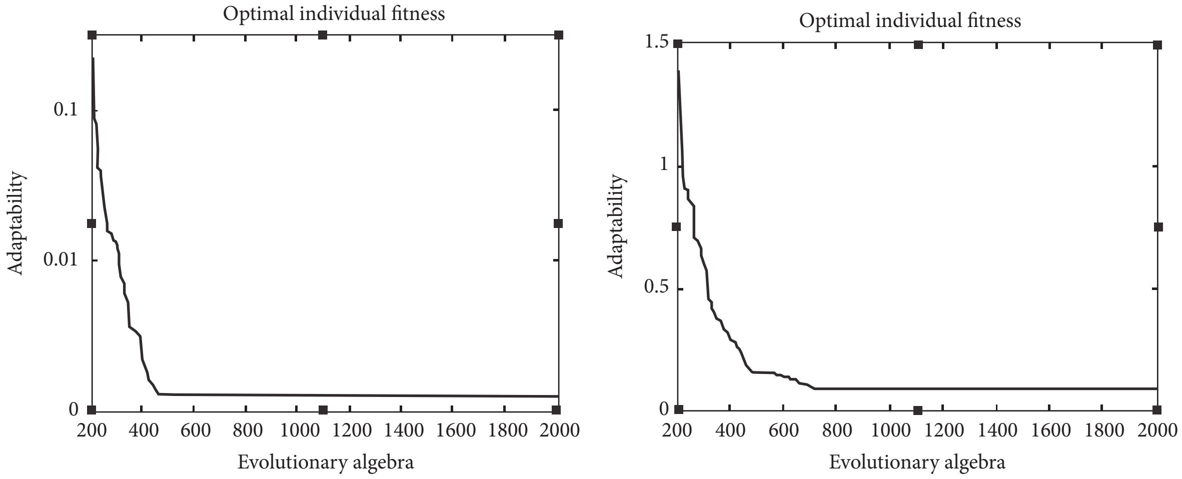
Figure 5: Test results of Sphere function.



(a) Optimal individual fitness for improved PSO algorithm

(b) Optimal individual fitness for PSO

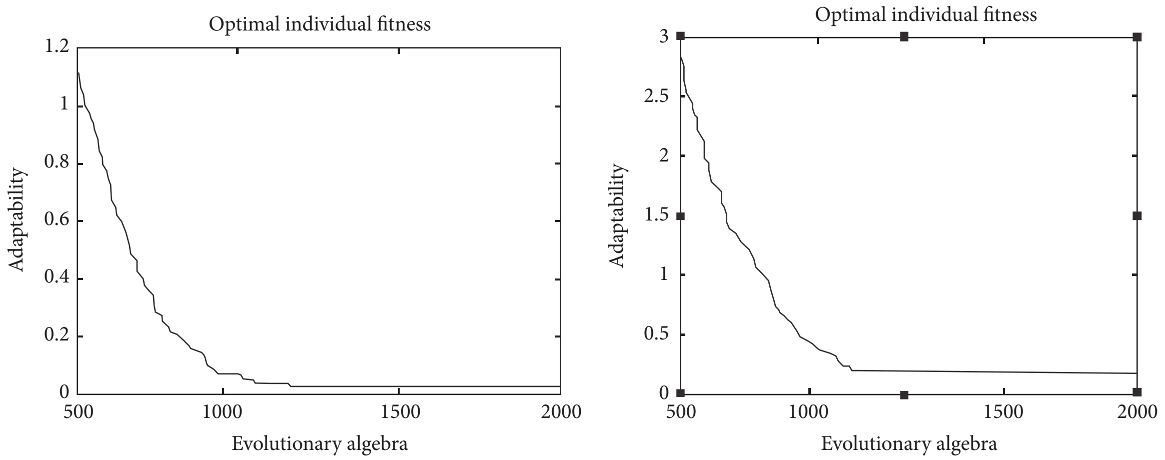
Figure 6: Test results of Rosenbrock function.



(a) Optimal individual fitness for improved PSO algorithm

(b) Optimal individual fitness for PSO

Figure 7: Test results of Rastrigin function.



(a) Optimal individual fitness for improved PSO algorithm

(b) Optimal individual fitness for PSO

Figure 8: Test results of Griewank function.

Table 1: Comparison of algorithm performance.

	Test function	Optimal solution	Worst solution	Average solution
Improved particle swarm optimization	Sphere	3.6×10^{-5}	4.5×10^{-5}	3.9×10^{-5}
Particle swarm optimization	Sphere	6.7×10^{-2}	8.2×10^{-2}	7.1×10^{-2}
Improved particle swarm optimization	Rosenbrock	1.2×10^{-1}	1.6×10^{-1}	1.4×10^{-1}
Particle swarm optimization	Rosenbrock	1.9	2.4	2.0
Improved particle swarm optimization	Rastrigin	9.8×10^{-4}	1.7×10^{-3}	1.3×10^{-3}
Particle swarm optimization	Rastrigin	9.7×10^{-2}	2.1×10^{-1}	1.8×10^{-1}
Improved particle swarm optimization	Griewank	4.2×10^{-2}	9.3×10^{-2}	5.8×10^{-2}
Particle swarm optimization	Griewank	4.0×10^{-1}	5.1×10^{-1}	4.8×10^{-1}

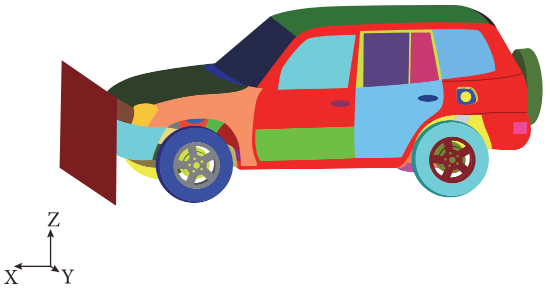


Figure 9: Finite element model of 100% frontal collision of minibuses.

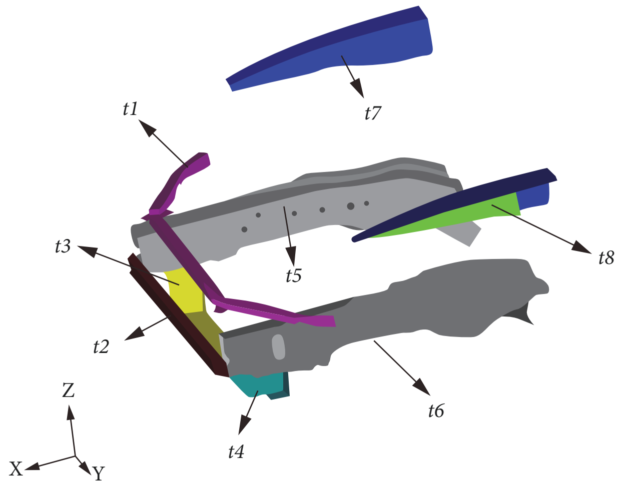


Figure 10: Design variables.

3. Multiobjective Optimization of Minibus Collision

3.1. Design Variables and Responses Functions. A finite element model of a 100% frontal overlap deformable barrier collision of a minibus is shown in Figure 9. According to C-NCAP's new vehicle evaluation index, under the premise of meeting the crash performance of the whole vehicle, the front part of the vehicle body should be crushed and fully deformed during the collision process so that substantial collision kinetic energy is absorbed and the acceleration peak at the lower end of the B-pillar is effectively decreased [30, 31]. Therefore, the thickness of the eight sheets in the front part of the vehicle body is selected as the design variable [32, 33], as shown in Figure 10. The initial values of the design variable are provided in Table 2.

Two minimum objective functions are set to reduce collision acceleration and body mass, including acceleration

Table 2: Initial values of design variable.

Design variable	t1	t2	t3	t4	t5	t6	t7	t8
Initial value (mm)	1.10	1.25	1.10	1.10	1.65	1.63	0.89	0.70

Table 3: Training data.

(a) Design variable test

Number of groups	t1/mm	t2/mm	t3/mm	t4/mm	t5/mm	t6/mm	t7/mm	t8/mm
1	0.66	1.39	1.84	0.69	1.13	1.05	1.51	1.46
2	0.68	1.41	0.99	1.14	0.92	0.81	1.44	1.10
3	0.72	1.20	1.47	1.17	1.73	0.77	1.41	1.25
...	...	...	...	...	...	...	...	...
70	2.00	1.17	0.68	2.00	1.52	1.81	0.75	1.89

(b) Response test design

Number of groups	$a_{max}/(m/s^2)$	m/kg	D1/mm	D2/mm	D3/mm
1	54.8	717.5	86.5	240.0	230.0
2	52.9	716.9	84.5	209.5	210.0
3	55.3	717.7	92.1	246.2	237.7
...	...	...	...	...	...
70	46.5	717.4	96.5	261.8	262.0

peak a_{max} at the lower end of B-pillar and mass m of the vehicle. In order to control the damage of vehicle body deformation to the occupant's chest, head, foot, and other parts, the response function at relevant positions is restrained within acceptable level, such as the intrusion D1, D2, and D3 separately at the instrument panel tube beam, the steering column hole, and the lower dash panel clutch pedal [34, 35].

3.2. Approximate Model. This study uses the Latin hypercube test method [36] to obtain 70 sets of training data combined with the design variables and crash performance indicators of the vehicle finite element model (Table 3). On the basis of these data, this study establishes the Kriging approximation model [37]; in order to verify the prediction accuracy of the approximate model, 20 sample points were randomly selected in Table 3, and the accuracy of the approximate model of the established 100% frontal collision model was evaluated based on the results obtained from these sample points. The evaluation results show that the established approximate model satisfies the evaluation index of each response value of the 100% positive overlap collision Kriging approximation model (such as the average relative error RAAE ≥ 95% and coefficient of determination $R^2 \leq 15\%$). It is enough to replace the original finite element model and carry out the next multiobjective particle swarm optimization design [38–40].

3.3. Multiobjective Optimization Design. In the multiobjective optimization design, the acceleration peak at the lower end of the B-pillar and intrusion amount of the key part should be kept as small as possible, and the vehicle weight [41] should be reduced. The mathematical model expression of the multiobjective optimization design is as follows:

$$\begin{aligned}
&\min a_{max}\left(t_1, t_2, t_3 \ldots t_8\right) \\
&\min m\left(t_1, t_2, t_3 \ldots t_8\right) \\
&D_1\left(t_1, t_2, t_3 \ldots t_8\right) \leq 97.6mm \\
&D_2\left(t_1, t_2, t_3 \ldots t_8\right) \leq 260mm \\
&D_3\left(t_1, t_2, t_3 \ldots t_8\right) \leq 268.4mm \\
&t_1, t_2, t_3 \ldots t_8 \\
&\quad \in [1.1, 1.25, 1.1, 1.1, 1.65, 1.63, 0.89, 0.7] .
\end{aligned} \tag{10}$$

The PSO algorithm has an inertia weight of 0.9 and an acceleration constant of $c_1 = c_2 = 1.4$. The optimization algorithm has 0.5 step size, 2000 iterations, and 10 particles. The maximum and minimum inertia weight values of the improved PSO algorithm are $\omega_{max} = 0.9$ and $\omega_{min} = 0.4$, respectively; the acceleration constant is $c_1 = c_2 = 1.4$; and $\mu = 4$. The improved PSO algorithm also has 0.5 step size, 2000 iterations, and 10 particles. The approximate model is optimized, and the results are compared, as shown in Table 4.

As shown in Table 4, the approximate model is optimized by the PSO algorithm. The results of the collision show that although the collision evaluation parameters have different degrees of improvement, they are still not ideal. The optimized vehicle acceleration and key parts of intrusion will still cause considerable damage to the passengers inside the vehicle. After the optimization of the improved PSO algorithm for the approximate model, the peak acceleration

Table 4: Comparison of optimization results.

(a) Design variable

	t1/mm	t2/mm	t3/mm	t4/mm	t5/mm	t6/mm	t7/mm	t8/mm
Original finite element model	1.10	1.25	1.10	1.10	1.65	1.63	0.89	0.70
Particle swarm optimization	1.20	1.00	1.30	0.90	1.10	1.30	0.92	0.51
Improved particle swarm optimization	1.15	1.02	1.23	0.92	0.96	1.30	0.90	0.49

(b) Collision target response

	a_{max} /(m/s^2)	D1/mm	D2/mm	D3/mm	m/kg
Original finite element model	57.60	97.60	260.60	268.40	716.85
Particle swarm optimization	53.20	90.10	252.40	254.20	710.61
Improved particle swarm optimization	47.90	84.70	241.60	242.10	710.40
Particle swarm optimization percentage	-7.64%	-7.68%	-3.15%	-5.29%	-0.87%
Improved particle swarm optimization percentage	-16.84%	-13.22%	-7.29%	-9.80%	-0.90%

Table 5: Optimized response fitting effect table.

Evaluation index	Approximate model prediction	Optimized finite element results
D1	84.7mm	86.2mm
D2	241.6mm	243.3mm
D3	242.1mm	243.4mm
a_{max}	47.9m/s^2	48.5m/s^2
m	710.4kg	710.41kg

at the lower end of the B-pillar is decreased to 47.9 m/s^2. The intrusion amount D1 of the instrument panel tube beam is decreased from 97.6mm to 84.7mm. Moreover, the backward intrusion amount D2 of the steering column hole is decreased from 260.6mm to 241.6mm, and the lower front panel clutch pedal is inwardly invaded. The amount of backward intrusion D3 of the lower front panel clutch pedal is decreased from 268.4mm to 242.1mm, and the entire vehicle quality is diminished from 716.85kg to 710.4kg. Relative to the PSO algorithm, the improved PSO has a remarkable improvement in terms of the effect of the performance evaluation index. This feature is in accordance with the expected results. After comprehensive consideration, this study finds that the improved PSO algorithm is better than the PSO algorithm for the Kriging approximation model. Moreover, the improved PSO algorithm is used as the optimization scheme for the 100% front collision of the car. The fitting effect of each response value after optimization is shown in Table 5. It can be seen that the fitting errors of each evaluation index are small. The approximate model prediction accuracy meets the requirements, and the obtained optimization result can represent the optimization result of the original finite element model.

4. Conclusion

(1) In this study, the PSO algorithm is improved by dealing with the method's various limitations, such as slow convergence speed, low precision, and tendency to easily fall into the local extremum. Then, the chaos idea is introduced, and the parameters related to particle velocity update are adaptively adjusted by using chaos thought to generate chaotic sequences. Four commonly used mathematical test functions with different characteristics are used to verify the performance of the improved PSO algorithm. The results show that the improved PSO algorithm has stronger optimization ability than the PSO algorithm.

(2) The PSO and improved PSO algorithms are applied to vehicle collision optimization. According to the comparison results, the evaluation index of the collision performance after optimization by the PSO algorithm improves, but the effect is unremarkable. This result indicates the need for further optimization. After using the improved PSO algorithm, the improvement effect of the vehicle crash performance evaluation index is remarkable. Therefore, the optimized design of the improved PSO algorithm is used as the final optimization method, thereby providing a new design for car crash optimization.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This project is supported by the National Natural Science Foundation of China (51305269) and funded by the Shanghai

Automotive Industry Technology Development Foundation (1744).

References

[1] S. Hou, Q. Li, S. Long, X. Yang, and W. Li, "Multiobjective optimization of multi-cell sections for the crashworthiness design," *International Journal of Impact Engineering*, vol. 35, no. 11, pp. 1355–1367, 2008.

[2] G. Sun, G. Li, S. Zhou, H. Li, S. Hou, and Q. Li, "Crashworthiness design of vehicle by using multiobjective robust optimization," *Structural and Multidisciplinary Optimization*, vol. 44, no. 1, pp. 99–110, 2011.

[3] C. M. Fonseca, "Genetic algorithms for multi-objective optimization: formulation, discussion and generalization," in *Proceedings of the Fifth International Conference on Genetic Algorithms*, pp. 416–423, San Mateo, USA, 1993.

[4] G. Sun, T. Pang, J. Fang, G. Li, and Q. Li, "Parameterization of criss-cross configurations for multiobjective crashworthiness optimization," *International Journal of Mechanical Sciences*, vol. 124-125, pp. 145–157, 2017.

[5] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 4598, pp. 671–680, 1983.

[6] A. Suppapitnarm, K. A. Seffen, G. T. Parks, and P. J. Clarkson, "Simulated annealing algorithm for multiobjective optimization," *Engineering Optimization*, vol. 33, no. 1, pp. 59–85, 2000.

[7] M. Dorigo, G. di Caro, and L. M. Gambardella, "Ant algorithms for discrete optimization," *Artificial Life*, vol. 5, no. 2, pp. 137–172, 1999.

[8] K. Doerner, W. J. Gutjahr, R. F. Hartl, C. Strauss, and C. Stummer, "Pareto ant colony optimization: a metaheuristic approach to multiobjective portfolio selection," *Annals of Operations Research*, vol. 131, no. 1–4, pp. 79–99, 2004.

[9] M. Jacqueline and C. Richard, *Application of particle swarm to multiobjective optimization*, Auburn: Auburn University, 1999.

[10] J. Fang, Y. Gao, G. Sun, N. Qiu, and Q. Li, "On design of multicell tubes under axial and oblique impact loads," *Thin-Walled Structures*, vol. 95, pp. 115–126, 2015.

[11] S. Pirmohammad and S. Esmaeili Marzdashti, "Crashworthiness optimization of combined straight-tapered tubes using genetic algorithm and neural networks," *Thin-Walled Structures*, vol. 127, pp. 318–332, 2018.

[12] N. Qiu, Y. Gao, J. Fang, G. Sun, Q. Li, and N. H. Kim, "Crashworthiness optimization with uncertainty from surrogate model and numerical error," *Thin-Walled Structures*, vol. 129, pp. 457–472, 2018.

[13] J. Fang, Y. Gao, G. Sun, Y. Zhang, and Q. Li, "Parametric analysis and multiobjective optimization for functionally graded foam-filled thin-wall tube under lateral impact," *Computational Materials Science*, vol. 90, pp. 265–275, 2014.

[14] D. Zhang, S. Cui, J. Cheng, W. Tian, and G. H. Su, "Improving the optimization algorithm of NTCOC for application in the HCSB blanket for CFETR Phase II," *Fusion Engineering and Design*, vol. 135, pp. 216–227, 2018.

[15] M. R. Bonyadi and Z. Michalewicz, "A locally convergent rotationally invariant particle swarm optimization algorithm," *Swarm Intelligence*, vol. 8, no. 3, pp. 159–198, 2014.

[16] K. R. Harrison, A. P. Engelbrecht, and B. M. Ombuki-Berman, "Self-adaptive particle swarm optimization: a review and analysis of convergence," *Swarm Intelligence*, vol. 12, no. 3, pp. 187–226, 2018.

[17] R. C. Eberhart and Y. H. Shi, "Particle swarm optimization: developments, applications and resources," in *Proceedings of the Congress on Evolutionary Computation*, vol. 1, pp. 81–86, Piscataway, NJ, USA, 2001.

[18] Z. Liu, J. Lu, and P. Zhu, "Lightweight design of automotive composite bumper system using modified particle swarm optimizer," *Composite Structures*, vol. 140, pp. 630–643, 2016.

[19] S. Guangyong, L. Guangyao, H. Shujian, Z. Shiwei, L. Wei, and L. Qing, "Crashworthiness design for functionally graded foam-filled thin-walled structures," *Materials Science & Engineering: A (Structural Materials: Properties, Microstructure and Processing)*, vol. 527, no. 8, pp. 1911–1919, 2010.

[20] N. Qiu, Y. Gao, J. Fang, G. Sun, and N. H. Kim, "Topological design of multi-cell hexagonal tubes under axial and lateral loading cases using a modified particle swarm algorithm," *Applied Mathematical Modelling*, vol. 53, pp. 567–583, 2018.

[21] K. Tatsumi, T. Ibuki, and T. Tanino, "Particle swarm optimization with stochastic selection of perturbation-based chaotic updating system," *Applied Mathematics and Computation*, vol. 269, pp. 904–929, 2015.

[22] J. Chuanwen and E. Bompard, "A hybrid method of chaotic particle swarm optimization and linear interior for reactive power optimisation," *Mathematics and Computers in Simulation*, vol. 68, no. 1, pp. 57–65, 2005.

[23] M. Chih, "Self-adaptive check and repair operator-based particle swarm optimization for the multidimensional knapsack problem," *Applied Soft Computing*, vol. 26, pp. 378–389, 2015.

[24] B. Zhao et al., "An improved particle swarm optimization algorithm for unit commitment," *International Journal of Electrical Power & Energy Systems*, vol. 28, no. 7, pp. 482–490, 2006.

[25] H. Shi, S. Liu, H. Wu et al., "Oscillatory particle swarm optimizer," *Applied Soft Computing*, vol. 73, pp. 316–327, 2018.

[26] A. Alfi and H. Modares, "System identification and control using adaptive particle swarm optimization," *Applied Mathematical Modelling*, vol. 35, no. 3, pp. 1210–1221, 2011.

[27] C. Qi, S. Yang, and F. Dong, "Crushing analysis and multiobjective crashworthiness optimization of tapered square tubes under oblique impact loading," *Thin-Walled Structures*, vol. 59, pp. 103–119, 2012.

[28] X. Yuan, Y. Yang, and H. Wang, "Improved parallel chaos optimization algorithm," *Applied Mathematics and Computation*, vol. 219, no. 8, pp. 3590–3599, 2012.

[29] S.-F. Dong, Z.-C. Dong, J.-J. Ma, and K.-N. Chen, "Improved PSO algorithm based on chaos theory and its application to design flood hydrograph," *Water Science and Engineering*, vol. 3, no. 2, pp. 156–165, 2010.

[30] G. Sun, H. Zhang, G. Lu, J. Guo, J. Cui, and Q. Li, "An experimental and numerical study on quasi-static and dynamic crashing behaviors for tailor rolled blank (TRB) structures," *Materials and Corrosion*, vol. 118, pp. 175–197, 2017.

[31] H. Fang, M. Rais-Rohani, Z. Liu, and M. F. Horstemeyer, "A comparative study of metamodeling methods for multiobjective crashworthiness optimization," *Computers & Structures*, vol. 83, no. 25-26, pp. 2121–2136, 2005.

[32] R. Lu, W. Gao, X. Hu, W. Liu, Y. Li, and X. Liu, "Crushing analysis and crashworthiness optimization of tailor rolled tubes with variation of thickness and material properties," *International Journal of Mechanical Sciences*, vol. 136, pp. 67–84, 2018.

[33] G. Sun, J. Tian, T. Liu, X. Yan, and X. Huang, "Crashworthiness optimization of automotive parts with tailor rolled blank," *Engineering Structures*, vol. 169, pp. 201–215, 2018.

[34] D. Abellán-López, M. Sánchez-Lozano, and L. Martínez-Sáez, "Frontal crashworthiness characterisation of a vehicle segment using curve comparison metrics," *Accident Analysis & Prevention*, vol. 117, pp. 136–144, 2018.

[35] S. Hou, D. Dong, L. Ren, and X. Han, "Multivariable crashworthiness optimization of vehicle body by unreplicated saturated factorial design," *Structural and Multidisciplinary Optimization*, vol. 46, no. 6, pp. 891–905, 2012.

[36] Q. He, D. Ma, Z. Zhang, and L. Yao, "Mean compressive stress constitutive equation and crashworthiness optimization design of three novel honeycombs under axial compression," *International Journal of Mechanical Sciences*, vol. 99, pp. 274–287, 2015.

[37] K. B. Soo, Y. B. Lee, and D. H. Choi, "Comparison study on the accuracy of metamodeling technique for non-convex functions," *Journal of Mechanical Science & Technology*, vol. 23, no. 4, pp. 1175–1181, 2009.

[38] Y. Zhang, G. Sun, X. Xu, G. Li, and Q. Li, "Multiobjective crashworthiness optimization of hollow and conical tubes for multiple load cases," *Thin-Walled Structures*, vol. 82, pp. 331–342, 2014.

[39] C. Maschio and D. J. Schiozer, "Probabilistic history matching using discrete Latin Hypercube sampling and nonparametric density estimation," *Journal of Petroleum Science & Engineering*, vol. 147, pp. 98–115, 2016.

[40] X. Song, G. Sun, and Q. Li, "Sensitivity analysis and reliability based design optimization for high-strength steel tailor welded thin-walled structures under crashworthiness," *Thin-Walled Structures*, vol. 109, pp. 132–142, 2016.

[41] F. Pan, P. Zhu, and Y. Zhang, "Metamodel-based lightweight design of B-pillar with TWB structure via support vector regression," *Computers & Structures*, vol. 88, no. 1-2, pp. 36–44, 2010.