World Electric Vehicle Journal

MDPI

Article

Anti-Rollover Trajectory Planning Method for Heavy Vehicles in Human–Machine Cooperative Driving

Haixiao Wu *, Zhongming Wu, Junfeng Lu and Li Sun

College of Mechanical and Electrical Engineering, Jinling Institute of Technology, Nanjing 211169, China; zhmingwu@jit.edu.cn (Z.W.); ljf@jit.edu.cn (J.L.); sunli790124@jit.edu.cn (L.S.)
* Correspondence: wuhaixiao@jit.edu.cn

Abstract: The existing trajectory planning research mainly considers the safety of the obstacle avoidance process rather than the anti-rollover requirements of heavy vehicles. When there are driving risks such as rollover and collision, how to coordinate the game relationship between the two is the key technical problem to realizing the anti-rollover trajectory planning under the condition of driving risk triggering. Given the above problems, this paper studies the non-cooperative game model construction method of the obstacle avoidance process that integrates the vehicle driving risk in a complex traffic environment. Then it obtains the obstacle avoidance area that satisfies both the collision and rollover profit requirements based on the Nash equilibrium. A Kmeans-SMOTE risk clustering fusion is proposed in this paper, in which more sampling points are supplemented by the SMOTE oversampling method, and then the ideal obstacle avoidance area is obtained through clustering algorithm fusion to determine the optimal feasible area for obstacle avoidance trajectory planning. On this basis, to solve the convergence problems of the existing multi-objective particle swarm optimization algorithm and analyze the influence of weight parameters and the diversity of the optimization process, this paper proposes an anti-rollover trajectory planning method based on the improved cosine variable weight factor MOPSO algorithm. The simulation results show that the trajectory obtained based on the method proposed in this paper can effectively improve the anti-rollover performance of the controlled vehicle while avoiding obstacles.

Keywords: trajectory planning; heavy vehicles; obstacle avoidance; rollover prevention; risk clustering fusion; particle swarm optimization

check for updates

Citation: Wu, H.; Wu, Z.; Lu, J.; Sun, L. Anti-Rollover Trajectory Planning Method for Heavy Vehicles in Human–Machine Cooperative Driving. *World Electr. Veh. J.* **2024**, *15*, 328. https://doi.org/10.3390/wevj15080328

Academic Editor: Peter Van den Bossche

Received: 12 June 2024
Revised: 13 July 2024
Accepted: 17 July 2024
Published: 24 July 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

1.1. Related Works

The current research on trajectory planning is mainly based on self-driving vehicles [1,2]. The principle is that when the distance between the controlled vehicle and the obstacle vehicle ahead is less than the safe distance, a series of candidate trajectories are first generated by algorithms such as quintic polynomials [3]. Then, it is optimized to obtain an ideal trajectory that satisfies both the safety requirements of obstacle avoidance and the vehicle dynamics constraints, according to the vehicle dynamic constraints [4]. That is, whether to perform trajectory planning and how to plan to obtain the trajectory are completely determined by the algorithm. Furthermore, the main goal of the planning algorithm is obstacle avoidance safety, and the stability of vehicle dynamics is merely used as a constraint [5,6].

Much current research highlights the importance of considering the human factor in autonomous driving systems. Human–machine cooperative driving models have emerged as a bridge between fully autonomous systems and traditional vehicle operation, emphasizing the role of the driver in decision-making processes, especially in ambiguous or critical situations [7–9].

Some current research also analyzes the challenges specific to heavy vehicles, such as the risk of rollover and load transfer dynamics, which add layers of complexity to

trajectory planning [10–12]. These studies provide insights into the unique considerations required for heavy-duty vehicles and contribute to the development of specialized trajectory planning methods.

1.2. Research Focus and Contributions

Unlike conventional autonomous driving trajectory planning, this paper focuses on heavy vehicles in human–machine cooperative driving scenarios, including the trajectory planning module and shared control rights between the driver and the system. Normally, the module passively tracks the driver's actions unless the vehicle's risk threshold is breached, in which case it actively intervenes to mitigate driving risks.

Therefore, compared with the existing planning algorithm, the main goal of which is obstacle avoidance safety, human–machine cooperative driving of heavy vehicles not only needs to consider the risk of rollover and the uncertainty of heavy vehicle load transfer but also needs to consider avoiding collisions with obstacles in the environment, which will cause more serious traffic accidents. That is to say, the main aim of the anti-rollover trajectory planning of driving risk fusion in this paper is to prevent the occurrence of rollover and to avoid the risk of collision in the process of anti-rollover trajectory planning. The contradiction between anti-rollover and anti-collision is the cause of the driving strategy, give way or don't give way, commonly adopted by drivers in emergencies. To sum up, coordinating the game relationship between obstacle avoidance safety and anti-rollover performance and establishing an anti-rollover trajectory planning method that integrates various driving risk factors are key to realizing anti-rollover trajectory planning under driving risk trigger conditions.

Given the above problems, this paper proposes the anti-rollover trajectory planning method for heavy vehicles in the scenario of human–machine cooperative driving. SAE International (Society of Automotive Engineers), an international automotive organization, has defined six different levels of autonomous driving, or independent, in vehicles (SAE, 2014) [13]. According to this definition, the method proposed in this paper is aimed at level 3 vehicles. Firstly, a non-cooperative game model integrating obstacle avoidance and rollover prevention was constructed, and the obstacle avoidance area that satisfies both collision and rollover benefits was obtained by way of the Nash equilibrium. Secondly, considering the sparsity of Nash equilibrium points in the whole region, a clustering fusion method, Kmeans-SMOTE, was proposed to directly fit the unconnectable regions. This method used the SMOTE oversampling method to supplement more Nash equilibrium point sampling points and then obtained the ideal obstacle avoidance area through clustering algorithm fusion to determine the optimal feasible area for obstacle avoidance trajectory planning. Thirdly, aiming to solve the convergence problems of the multi-objective particle swarm optimization algorithm, weight parameter analysis, and optimizing process diversity, an improved cosine change weight factor MOPSO algorithm was proposed to effectively improve the convergence of the anti-rollover trajectory planning algorithm. In general, the main contributions of this paper are as follows.

(1) A non-cooperative game model integrating obstacle avoidance and rollover prevention is proposed to obtain the area of the ideal trajectory that can satisfy both anti-collision and anti-rollover benefits.
(2) The Kmeans-SMOTE risk clustering fusion method is established to determine optimal feasible regions for obstacle avoidance trajectory planning, which supplements more Nash equilibrium sampling points through the SMOTE oversampling and obtains the ideal obstacle avoidance area through clustering algorithm fusion.
(3) An improved cosine change weight factor MOPSO algorithm is proposed to effectively improve the convergence of the anti-rollover trajectory planning algorithm.

The remaining sections of this paper are arranged as follows: the second section discusses the model building, the third section is the anti-rollover trajectory planning method for heavy vehicles triggered by driving risk, the fourth section is the simulation analysis, and the fifth section is the conclusion.

2. The Rollover Dynamics Model of Heavy Vehicles

Aiming to study the dynamic performance of heavy vehicles during obstacle avoidance, this section establishes a rollover dynamics model, including four degrees of freedom: longitudinal, lateral, yaw, and roll [14,15]. The longitudinal motion of the vehicle is analyzed, and the forces in the x direction are shown in the following equation.

$$ma_x - mv_y r = \sum F_x \tag{1}$$

where m is the mass of the vehicle, a_x is the longitudinal acceleration, v_y is the lateral velocity, and r is the yaw rate.

Next the lateral motion of the vehicle is analyzed, and the forces in the y direction are shown in the following equation.

$$ma_y - m_s e\ddot{\varphi} = \sum F_y \tag{2}$$

where a_y is the lateral acceleration, m_s is the sprung mass, e is the distance from the sprung mass to the roll center, and $\ddot{\varphi}$ is the roll angular acceleration.

This study examined the rotational behavior of the vehicle during turning maneuvers with a specific focus on yaw motion. The analysis is executed through a moment equilibrium assessment centered on the vertical axis, denoted as the z-axis. The yaw dynamics were scrutinized to ensure moment balance around this axis. The pertinent equation is presented below.

$$I_z\dot{r} = \sum M_z \tag{3}$$

where I_z is the moment of inertia of the z-axis in vehicle mass around vehicle coordinates, and $\dot{r}$ is the yaw angular acceleration.

The roll motion of the whole vehicle is analyzed, that is, whether the moment balances on the x-axis. The equation is shown below.

$$I_{xz}\ddot{\varphi} - m_s e a_y = \sum M_x \tag{4}$$

where I_{xz} is the moment of inertia of the x-axis in vehicle mass around vehicle coordinates, and m_s is the sprung mass of the vehicle.

The lateral acceleration at the center of mass of the vehicle is shown in the following equation.

$$a_y = \dot{v}_y + v_x r \tag{5}$$

where v_x represents the longitudinal rate.

A linear three-degree-of-freedom vehicle model, including yaw, lateral, and roll motions, is established. The lateral motion model of the vehicle is analyzed, as shown in the following equation.

$$mu(\dot{\beta} + \gamma) - m_s h_s \ddot{\phi} = C_f(\beta + \frac{L_f}{u}\gamma - \delta_f - R_f\phi) + C_r(\beta - \frac{L_r}{u}\gamma - \delta_r - R_r\phi) \tag{6}$$

where m is the mass of the whole vehicle, u is the longitudinal speed, β is the sideslip angle, γ is the yaw rate, h_s is the roll arm, ϕ is the body roll angle, C_f and C_r are the comprehensive sideslip stiffnesses of the front and rear axles, respectively, L_f and L_r are the distances from the center of mass to the front and rear axles, respectively, δ_f and δ_r are the front and rear wheel angles of the vehicle, respectively, and R_f and R_r are the front and rear wheel roll deflection coefficients, respectively.

The yaw motion model of the vehicle is analyzed, as shown in the following equation.

$$I_{zz}\dot{\gamma} - I_{xz}\ddot{\phi} = C_f L_f(\beta + \frac{L_f}{u}\gamma - \delta_f - R_f\phi) - C_r L_r(\beta - \frac{L_r}{u}\gamma - \delta_r - R_r\phi) \tag{7}$$

where I_{zz} is the rotational inertia of the whole vehicle around the z-axis, and I_{xz} is the inertia product.

The roll motion model of the vehicle is analyzed, as shown in the following equation.

$$I_{xx}\ddot{\phi} - I_{xz}\dot{\gamma} - m_s h_s u(\dot{\beta} + \gamma) = m_s g h_s \phi - C_\phi\dot{\phi} - K_\phi\phi \tag{8}$$

where I_{xx} is the moment of inertia of the sprung mass around the x-axis, g is the gravitational acceleration, K_ϕ is the sum of the front and rear suspension roll stiffness, and C_ϕ is the sum of the front and rear suspension roll damping.

3. Anti-Rollover Trajectory Planning Method for Heavy Vehicles Considering Driving Risk Triggers

3.1. Obstacle Avoidance Area Based on Clustering Fusion of Driving Risk

At this stage, research has only considered the impact of security to determine the starting point, end point, and corresponding candidate trajectory of the trajectory. In the anti-rollover trajectory planning problem of heavy vehicles in the human–machine cooperative driving scenario studied in this paper, it is necessary to consider both anti-rollover performance and obstacle avoidance performance [16]. When the driver's behavior triggers the rollover risk threshold, the planning module replaces the driver to plan the anti-rollover trajectory; when the rollover risk drops below the threshold, the trajectory planning task is terminated, and the driver returns to control. The candidate trajectories may be of different lengths, and there is no fixed endpoint, as shown in Figure 1. It is difficult to determine the trajectory selection index through the existing known starting point and ending point. Therefore, to solve this problem, it is necessary to determine the candidate area in the environment that can balance the rollover risk and the collision risk instead of simply determining the candidate trajectory, and then an optimal trajectory in the candidate area is obtained through using the corresponding optimization method.

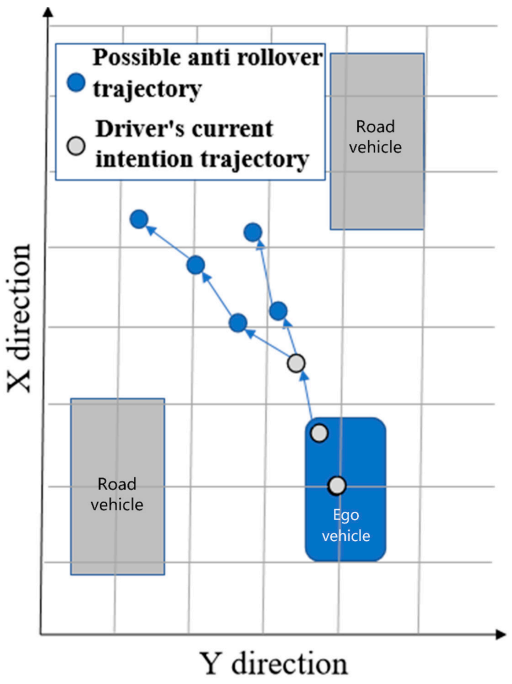


Figure 1. The difference between anti-rollover planning and traditional obstacle avoidance planning.

In this section, the collision risk and rollover risk are regarded as the participants, and the relative position of the vehicle is regarded as the strategy set of each participant in the game process. Aiming to determine feasible regions for obstacle avoidance trajectory planning, the game model construction method for the obstacle avoidance process based on integrated prediction of driving risk in the time domain solves the Nash equilibrium among different strategies and establishes the mapping relationship between the Nash equilibrium and the trajectory that can obtain high returns for participants [17]. The risk clustering fusion of the game relationship is shown in Figure 2. The blue dots and curves represent the sparse trajectories before clustering fusion, the yellow dots and arrows represent the candidate trajectory set obtained after clustering fusion, and the blue area represents the candidate trajectory space.

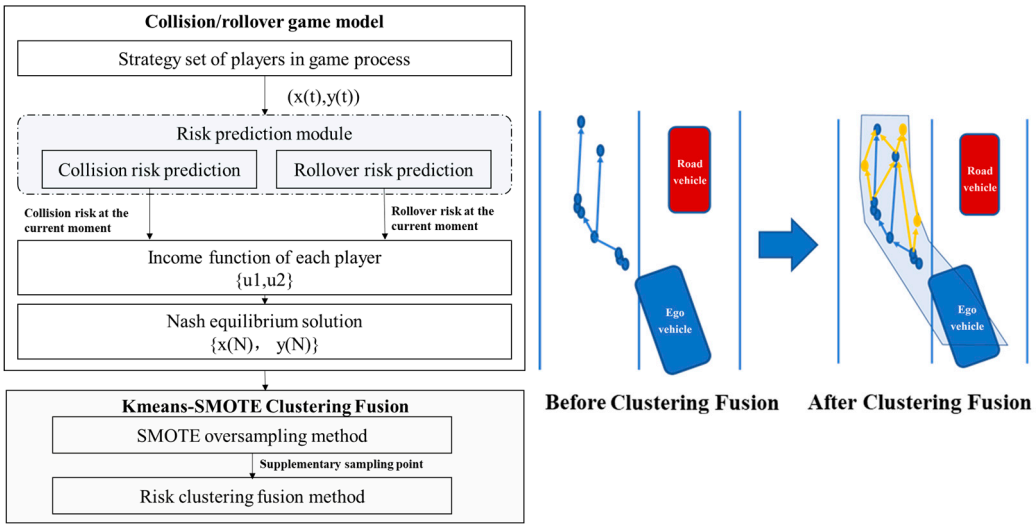


Figure 2. Risk cluster fusion based on the game relationship between collision and rollover. (**a**) Framework of risk cluster fusion; (**b**) Comparison before and after cluster fusion.

Firstly, the quantitative evaluation method of the rollover risk and collision risk in the process of obstacle avoidance is analyzed [18]. The two indicators based on the artificial potential field method and the zero moment point method are normalized to quantitatively evaluate the rollover risk and collision risk. The equation is as follows.

$$\begin{cases} f(U) = \sum\limits_{i} U_{P_j} + U_K + U_G \\ f(ZMP) = \dfrac{1}{mg}[-m_u a_y h_u + m_s g h \sin\varphi - m_s a_y (h_r + h\cos\varphi) + I_x \ddot{\varphi}] \end{cases} \tag{9}$$

where U_{P_j}, U_K, and U_G are, respectively, the obstacle vehicle repulsion field, the road boundary field, and the virtual gravitational field of the driving target.

Secondly, the game model of the anti-rollover trajectory planning process is analyzed and established. The collision risk includes the collision risk with the obstacle vehicle in the vehicle lane and the collision risk with the vehicle in the adjacent lane caused by the rollover risk. That is, the total number of collision risk participants in a prediction time domain p is unknown, and their strategies are all points in the environment at the current moment, which is shown in the next equation.

$$s_i \in \{(x_i, y_i) | x_i, y_i \in P\} \tag{10}$$

Among them, the payoff function for rollover risk participants is shown in the following equation.

$$U_{rr} = \sum_{j=1}^{j=j_{end}} u_j^{rr}(s_j^{rr}) = \sum_{j=1}^{j=j_{end}} -|f_{rr}(s_j^{rr}) + \Delta f_{rr}(s_j^{rr})| \tag{11}$$

where j is the influence of j-th strategy on the payoff function for the i-th participant.

The number of collision risk participants is uncertain; there are as many collision risk participants as potential obstacles in the environment, so the payment function is defined as shown in the following equation.

$$U_{cr}^{(k)} = \sum_{j=1}^{j=j_{end}} u_j^{cr}(s_j^{cr}) = \sum_{j=1}^{j=j_{end}} -|f_{cr}(s_j^{cr}) + \Delta f_{cr}(s_j^{cr})| \tag{12}$$

where k represents the k-th obstacle.

Because only a sparse set of points is obtained by solving the game model, a risk clustering fusion method based on Kmeans-SMOTE is proposed in this section to obtain candidate trajectory regions [19,20]. The process of data transfer, the generation of candidate trajectory generation, and the region's generation are shown in Figure 3. The discrete trajectory that is set as the clustering object is clustered into the primary candidate trajectory space by using the k-means clustering algorithm, which provides conditions for the increase of the sample space. Then the SMOTE oversampling method is used to add minority class samples to combat small separation issues, which can obtain the filled candidate trajectory set. On this basis, the risk judgment module will re-judge the risk of the newly generated trajectories and propose trajectories with rollover risk and collision risk. The straight-road clustering process cannot generate new spaces, and the loop terminates.

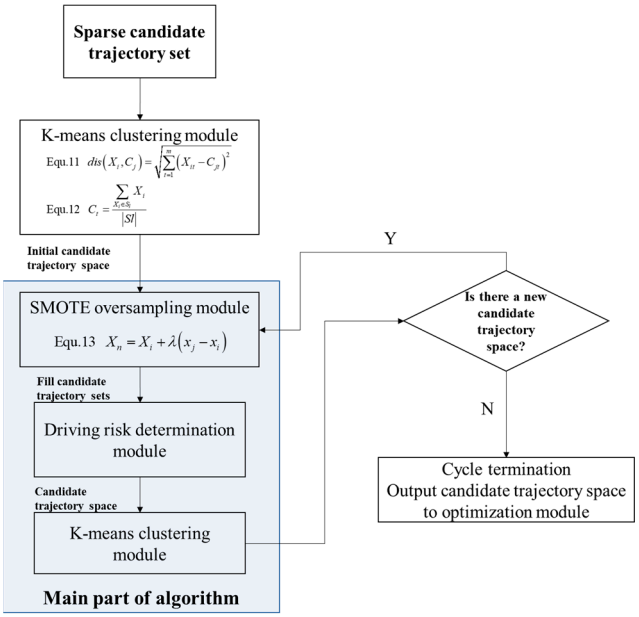


Figure 3. Risk clustering fusion method based on Kmeans-SMOTE.

The main part of the risk clustering fusion method based on Kmeans-SMOTE includes three steps: clustering, oversampling, and driving risk determination. The specific implementation is as follows.

The first step is clustering. For each sample x in the minority class, the k-nearest neighbors are obtained, which is the distance to all the minority class samples based on the Euclidean distance. In the clustering step, the way to k-means is used to cluster into k groups, and filtering selection is used for oversampled cluster clusters for oversampling to retain clusters with a high proportion of minority class samples. The given data sample X is assumed to contain n objects, each of which has attributes of m dimensions, namely $X = (X_1, X_2, X_3, \ldots, X_n)$. The goal of the k-means algorithm is to gather r objects into the k clusters according to the similarity of each other, and each object merely belongs to a cluster whose distance to the center of the cluster is the smallest. The k cluster centers, $(C_1, C_2, C_3, \ldots, C_k)$, $1 < k \leq n$, are initialized to calculate the Euclidean distance from each object to each cluster center. The equation is shown below.

$$dis(X_i, C_j) = \sqrt{\sum_{t=1}^{m}(X_{it} - C_{jt})^2} \tag{13}$$

where X_i represents the i-th object, $1 < i < n$. C_j represents the j-th cluster center. X_{it} represents the t-th attribute of the i-th object, $1 \leq t \leq m$. C_{jt} represents the t-th attribute of the j-th cluster center.

The distance between each object and each cluster center is compared in turn, and the object is assigned to the cluster with the nearest cluster center, which can obtain k clusters, namely $\{S_1, S_2, S_3, \ldots, S_k\}$. The K-means algorithm uses the center to define the prototype of the cluster. The center of the cluster is the mean value of all the objects in each dimension of the clusters. The calculation equation is as follows.

$$C_t = \frac{\sum\limits_{X_i \in S_l} X_i}{|S_l|} \tag{14}$$

where C represents the center of the first cluster, $1 < l \leq k$. $|S_l|$ represents the number of objects in the first cluster, and X_i represents the i-th object in the first cluster, $1< i \leq |S_l|$.

The second step is oversampling. Firstly, more samples are assigned to clusters where the few samples are sparsely distributed by assigning the number of synthetic samples. The sample imbalance ratio is set to determine the sampling magnification, N, according to the sample imbalance ratio. For each minority class sample, x, it randomly selects several samples from its k-nearest neighbors and assumes it as x_n. Then, the initial candidate trajectory space obtained by clustering is filled based on the smote oversampling algorithm. The purpose of the smote oversampling algorithm is to synthesize a new minority sample. The synthesis strategy is that a point on the connection between a and b is selected as a newly synthesized minority class sample where a is each minority class sample and b is the nearest sample of it. SMOTE is applied in each selected cluster to achieve the target ratio of minority and majority instances. SMOTE uses the samples x_i and x_j screened from a small sample set and the corresponding random numbers, $0 < \lambda < 1$. A new sample is constructed through the relationship between the two samples, as shown in the following equation.

$$X_n = X_i + \lambda(x_j - x_i) \tag{15}$$

The third step is to use the driving risk judgment conditions proposed in Section 3 to judge the filled candidate trajectory set obtained by oversampling in the previous step and eliminate the candidate trajectories with driving risk.

The fourth step is to compare the candidate space formed by the trajectory set after removing the risk trajectory from the initial space. If there is a difference, the newly generated space is assigned to X, and the next cycle is repeated. If there is no difference, the cycle is terminated, and the area obtained by clustering is used as a candidate area, which is outputted to the subsequent trajectory optimization module to obtain the undetermined trajectory in the area through the optimization algorithm.

3.2. Multi-Objective Optimization Method for Anti-Rollover Trajectory

In this section, a MOPSO algorithm with a cosine variable weight factor is proposed by analyzing the shortcomings of the existing particle swarm optimization algorithm [21,22]. On this basis, considering the load transfer characteristics of heavy vehicles, the anti-rollover trajectory planning objectives and constraints are established to realize the anti-rollover trajectory planning when the driving risk is triggered.

The basic particle swarm optimization algorithm is analyzed. During each search generation, each particle constantly updates its position and velocity state by pursuing the optimal solution found by itself, pbest, and the optimal solution found by the entire particle swarm, gbest. The equation is as follows.

$$v_{i,j}(t+1) = wv_{i,j}(t) + c_1 r_1[p_{i,j} - x_{i,j}(t)] + c_2 r_2[p_{g,j} - x_{i,j}(t)] \tag{16}$$

$$x_{i,j}(t+1) = x_{i,j}(t) + v_{i,j}(t+1), j = 1, 2, \cdots, d \tag{17}$$

where w is the inertia weight value, c_1 and c_2 are the learning factor, and its value is positive, and r_1 and r_2 are uniformly distributed random numbers between 0 and 1.

Some control parameters of PSO itself determine its performance to a large extent. These parameters include the number of particles, the maximum speed of particles, learning

factors and weight coefficient values, and so on. In the later stage of algorithm convergence, due to the decrease of particle swarm diversity, the algorithm falls into the local optimum, and it is difficult for particles to jump out in the poor search area, forming a search stagnation. In view of the above shortcomings, this paper solves the shortcomings of the traditional PSO algorithm by improving the parameters of the updated formula. The cosine variable weight factor proposed in this paper is shown in the following equation.

$$\omega(k) = \frac{\omega_{\max} - \omega_{\min}}{2} \cos(\pi \frac{k}{k_{\max}}) + \frac{\omega_{\max} + \omega_{\min}}{2} \tag{18}$$

where $k_{\max}$ is the number of final iterations. k is the number of iterations of this algorithm, and $\omega(k)$ is the inertia weight factor corresponding to the k-th iteration. According to the above equation, the variation curves of different weight factors with the number of iterations are drawn, as shown in Figure 4.

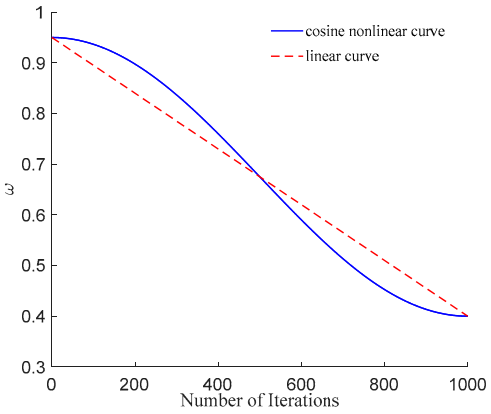
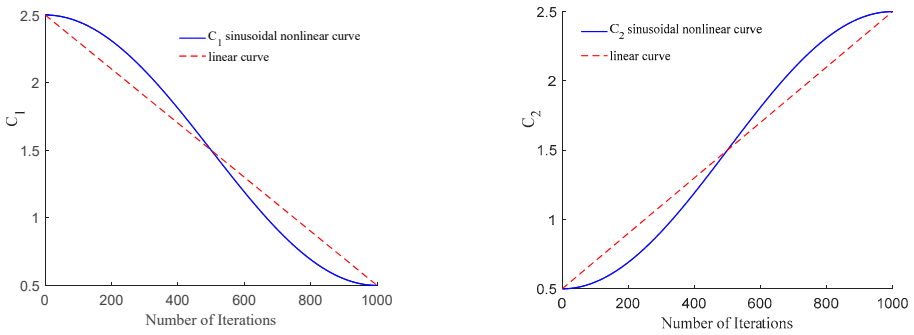


Figure 4. Weight factor change curve.

In terms of the calculation of the acceleration factor, an adaptive acceleration factor based on a sine function was designed, and its mathematical expression is as follows.

$$\begin{cases} c_1(k) = c_a \sin(\frac{\pi}{2} \frac{\frac{k_{\max}}{2} - k}{\frac{k_{\max}}{2}}) + c_b \\ c_2(k) = c_\alpha \sin(\frac{\pi}{2} \frac{k - \frac{k_{\max}}{2}}{\frac{k_{\max}}{2}}) + c_\beta \end{cases} \tag{19}$$

where c_a, c_b, c_α, c_β are parameters to be determined. Higher optimization results can be obtained when c_1 takes values from [0.5, 2.5] and c_2 takes values from [0.5, 2.5]. So, c_a = 1, c_b = 1.5, c_α = 1 and c_β = 1.5 can be determined. Figure 5 shows the curves of the acceleration factor c_1 and c_2 changing with the number of iterations.



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

Figure 5. Variations curves of the acceleration factor c_1 and c_2 changing with the number of iterations. (**a**) The variations curves of c_1. (**b**) The variations curves of c_2.

It can be seen from the change curves of the weight factor and the acceleration factor that in the early stage of the PSO algorithm iteration, the weight factor is larger, the acceleration factor is larger, and the acceleration factor is smaller, which is conducive to improving the global learning of the PSO algorithm. In the later stage, the weight factor is smaller, the acceleration factor is smaller, and the acceleration factor is larger, which is beneficial to improve the local learning ability of the PSO algorithm.

The new weight update equation and acceleration factor update equation are substituted into the original update equation to obtain a new update equation, which is shown below.

$$v_{i,j}(t+1) = w(k)v_{i,j}(t) + c_1(k)r_1[p_{i,j} - x_{i,j}(t)] + c_2(k)r_2[p_{g,j} - x_{i,j}(t)] \tag{20}$$

$$x_{i,j}(t+1) = x_{i,j}(t) + v_{i,j}(t+1), j = 1, 2, \cdots, d \tag{21}$$

The specific process of anti-rollover trajectory planning based on the cosine change weight factor MOPSO algorithm is as follows.

Firstly, the current velocity and position of each particle i in the particle population are initialized, which are shown below.

$$\begin{cases} \mathbf{V}_i = \begin{pmatrix} v_{i,1} & v_{i,2} & \cdots & v_{i,e} \end{pmatrix} \\ \mathbf{X}_i = \begin{pmatrix} x_{i,1} & x_{i,2} & \cdots & x_{i,e} \end{pmatrix} \end{cases} \tag{22}$$

where $\mathbf{V}_i$ and $\mathbf{X}_i$ represent the velocity and position information of the i-th particle in the e-dimensional space, respectively.

Secondly, the fitness function value of each particle i is evaluated, the current position and speed of particle i are saved to the pbest solution, and the speed and position of the best particle among them are saved to the gbest value. In the process of trajectory planning, it is not only necessary to ensure the vehicle's obstacle avoidance performance, but also to reduce the risk of the vehicle's rollover. Therefore, when designing the fitness function, in addition to considering the artificial potential field and rollover index, it is also necessary to introduce the side slip angle of the center of mass and the yaw rate as the optimization objective function, as shown in the following equation.

$$J = \min[k_1 f(U) + k_2 f(ZMP) + k_3 f(\beta) + k_4 f(r)] \tag{23}$$

where U is the safety index, ZMP is the rollover index, β is the sideslip angle of the center of mass, and r is the yaw rate.

Thirdly the velocity and position of the current particle are updated based on the cosine change weight factor, as shown in the above equation.

Fourthly, the function fitness values of each particle i are compared with the passing best position pbest. If its fitness value is better, the previous best position's pbest value is replaced with the current position.

Fifthly, the pbest value obtained by each particle i is compared with the gbest value in the entire population to update the gbest value in between.

Sixthly, when the stop condition is reached during the running of the algorithm, including the pre-specified algorithm operation accuracy or the number of particle search iterations, the search work of particle i ends, and the independent variable value corresponding to the optimal solution of the objective function is outputted. Otherwise, return to 3 and continue the search work.

4. Simulation Analysis

4.1. Analysis of Risk Clustering Fusion Results

Aiming to verify the effectiveness of the driving risk clustering fusion algorithm proposed in this chapter, the designed simulation scene is shown in Figure 6. The scene contains a large number of curves. On the one hand, it mainly verifies the anti-rollover capability of the vehicle. On the other hand, it uses curves to simulate the driver's emergency avoidance behavior after discovering obstacles. The curves within the simulated scenario span a range of radii from 25 m to 650 m, embodying a wide array of curvatures

that mirror the majority of road conditions encountered in practical engineering contexts. This approach enables a comprehensive evaluation of the vehicle's performance across diverse turning scenarios, ultimately ensuring the robustness of the driving risk clustering fusion algorithm under investigation. The road surface adhesion coefficient is set as 0.8, and the driver model is used to control the steering wheel angle during the simulation process shown in the equation, in which the principle is to output the input size of the steering wheel angle according to the deviation of displacement and the angle between the preview point and the actual trajectory.

$$\varepsilon_y = Y_{\mathrm{d}}(t + T_{\mathrm{p}}) - (Y(t) + T_{\mathrm{p}}\dot{Y}(t)) \tag{24}$$

where Y_{d} is the ideal lateral displacement and T_{p} is the preview time.

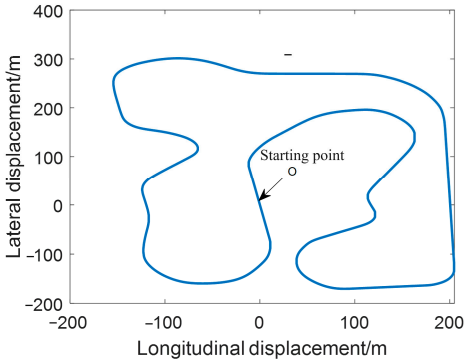
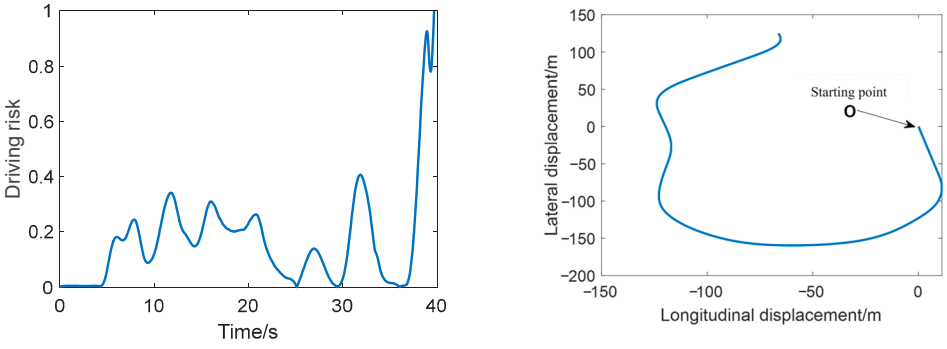


Figure 6. The simulation scene of the driver model.

The driving risk triggering situation when the longitudinal vehicle speed is 50 km/h is analyzed, as shown in Figure 7. From the figure, the simulation terminates when the time is about 40 s. Since the simulation environment does not contain obstacles, it means that the vehicle has rolled over at this time, and the risk value of the vehicle rollover is 1. Therefore, in this section, the vehicle position at 40 s is used as the starting point to verify the effectiveness of the algorithm proposed in this paper for reducing vehicle driving risk and preventing rollover.



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

Figure 7. Analysis of driving risk triggering situation in the simulation process. (**a**) Risk of rollover. (**b**) Displacement of vehicle.

Based on the algorithm designed in this paper, the vehicle position when the driving risk is triggered at 40 s is used as the starting point for obstacle avoidance trajectory planning. The purpose of the planning is to reduce the driving risk to a level where rollover and collision will not occur. The results are shown in Figure 8. A total of four primary trajectories that meet the requirements are found to form a sparse candidate trajectory set, that is, all trajectories in the set can reduce the driving risk of the controlled vehicle

below the threshold, and the starting point and end point of the planned trajectory are the switching points of human–machine driving authority.

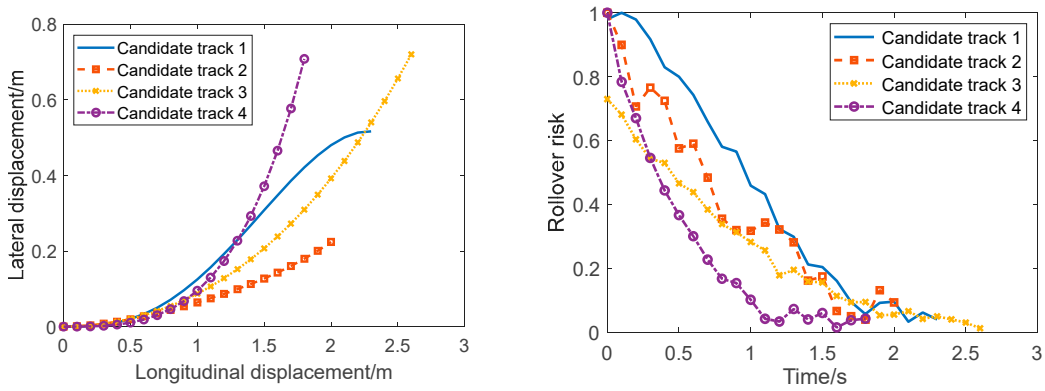


Figure 8. Sparse set of candidate trajectories and their driving risks. (**a**) Sparse set of candidate trajectories. (**b**) Candidate trajectory rollover risk.

Then, when the longitudinal vehicle speed is selected as 35 km/h, the triggering situation of rollover risk is analyzed, which is shown in Figure 9. For the convenience of analysis, the rollover risk level is divided into five levels, that is, level 1 means 0.2 to 0.4, level 2 means 0.4 to 0.6, level 3 means 0.6 to 0.8, and level 5 means 0.8 to 1. Therefore, the purpose of trajectory planning is to reduce the global rollover risk of the controlled vehicle to below level 3 for passage safety.

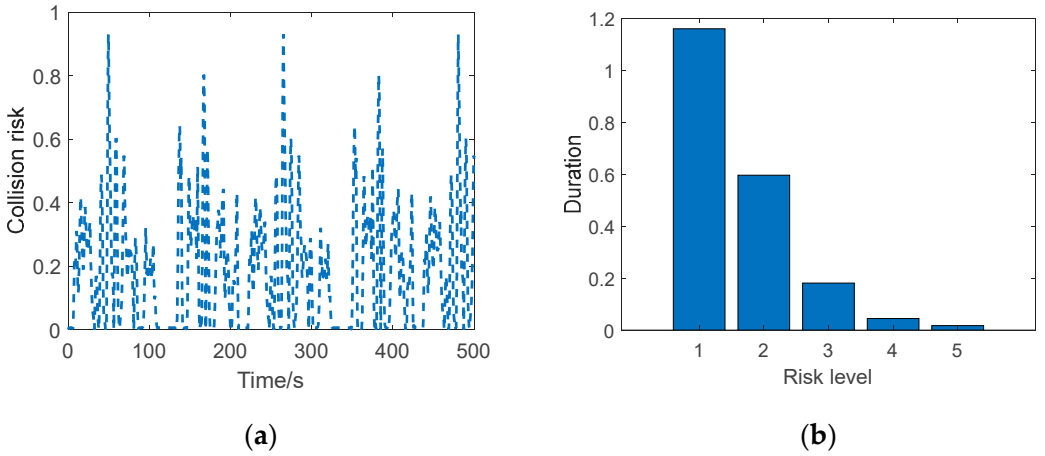


Figure 9. Analysis of driving risk triggering situation in the simulation process. (**a**) Risk of rollover. (**b**) Risk Statistics.

4.2. Analysis of Anti-Rollover Trajectory Optimization Results

Taking the simulation environment and trajectory planning results established in Section 4.2 as the input, the effectiveness of the trajectory optimization algorithm proposed in this section is verified. As depicted in Figure 10, it can be seen that compared with the basic PSO algorithm, the convergence speed of the CPSO algorithm proposed in this paper is significantly improved, and it effectively avoids the situation wherein the basic PSO algorithm stops near the local optimal solution.

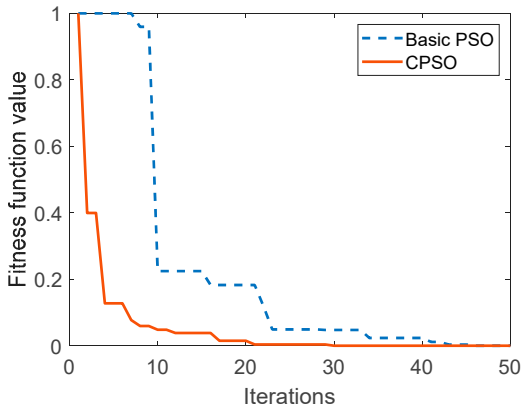


Figure 10. Comparison of optimization results between CPSP and basic PSO.

The risk values, their deviations, and the risk levels of the CPSO and basic CPSO algorithms are shown in Figure 11. By comparing the risk value of the optimized trajectory, the average driving risk obtained by the CPSO algorithm proposed in this paper is reduced by 23.84%, and the peak value is reduced by 33.19%, compared with the results obtained by the basic PSO algorithm. The duration of the risk level above level 1 is significantly reduced, indicating the effectiveness of the method proposed in this paper in reducing driving risks.

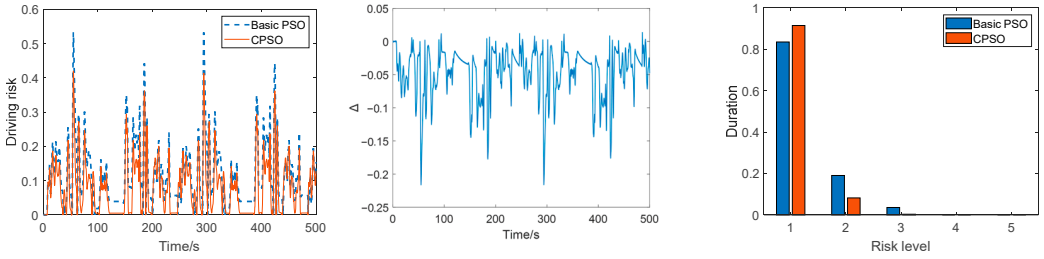


Figure 11. Risk comparison after optimization. (**a**) Risk comparison. (**b**) Risk value deviations. (**c**) Comparison of risk levels.

Then, the vehicle dynamics parameters corresponding to the optimized trajectory are compared. It can be seen in Figure 12 that the peak value of the lateral acceleration obtained by the method proposed in this paper is reduced by 33.58%, and the average value is reduced by 29.06%; the peak value of the roll angle is reduced by 18.26%, and the average value is reduced by 13.82%, compared with the results obtained by the basic PSO algorithm. This shows that the method proposed in this paper can effectively reduce the probability of instability and rollover of the controlled vehicle.

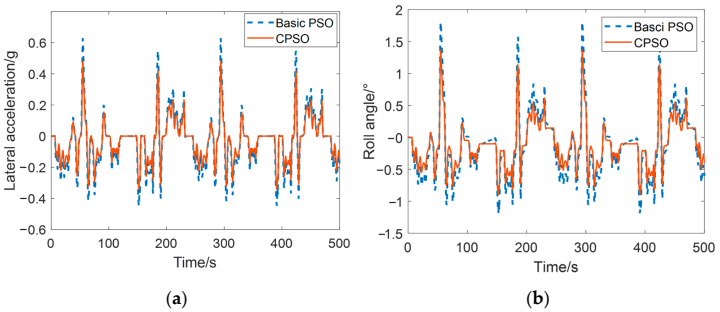
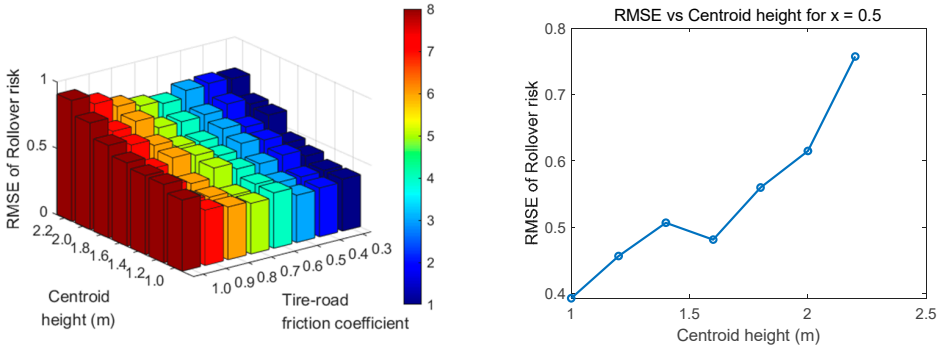


Figure 12. Comparison of dynamics parameters after optimization. (**a**) Comparison of lateral acceleration. (**b**) Comparison of roll angle.

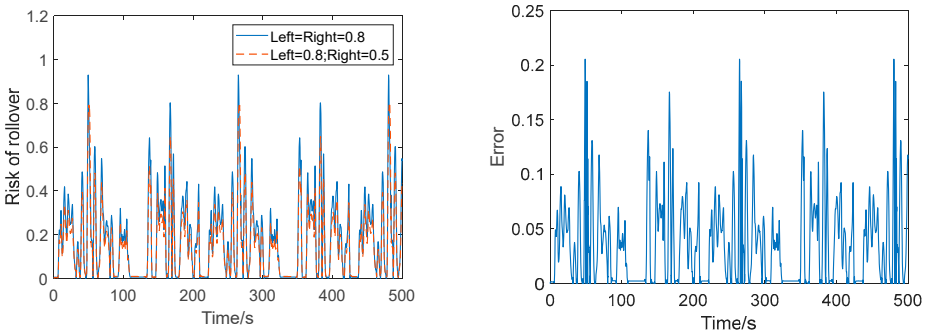
To verify the applicability of the algorithms proposed in this paper under different coefficients of adhesion and different centroid heights, simulations were conducted using the vehicle model, with the results shown in Figure 13. As can be seen from Figure 13a that when the coefficient of adhesion decreases, the mean square error of the rollover risk also decreases. When the coefficient of adhesion is 0.5, the mean square error of the rollover risk at different center-of-gravity heights is shown in Figure 13b. Analyzing the causes of the results, the smaller the coefficient of adhesion, the more likely the vehicle is to skid, and the rollover risk is paradoxically smaller. When the center-of-gravity height varies, there is an overall trend that the lower the center of gravity, the smaller the rollover risk.



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

Figure 13. Rollover risk at varying friction coefficients and centroid heights. (**a**) Rollover risk. (**b**) Rollover risk vs. centroid height.

The rollover risk when the coefficients of adhesion on both sides are inconsistent is shown in Figure 14, with the left side at 0.8 and the right side at 0.5, and compared with the simulation results when both are at 0.8, it can be observed that although there are significant fluctuations, there is no obvious increase in rollover risk. This indicates that the algorithms proposed in this work can still ensure good control effects when the coefficients of adhesion on both sides are inconsistent.



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

Figure 14. Asymmetric adhesion rollover risk. (**a**) Risk of rollover. (**b**) Error of risk.

The results indicate that under various coefficients of adhesion and center-of-gravity heights, the rollover risk of the vehicle remains within safe limits. This finding is crucial as it demonstrates the robustness of the algorithms across a range of conditions that could potentially affect vehicle stability. By maintaining the rollover risk within an acceptable threshold, the proposed methods ensure that safety is upheld, even in challenging scenarios with reduced traction or altered weight distribution. The comprehensive analysis and simulation outcomes provide a strong foundation for the further development and application of these algorithms in real-world human–machine cooperative driving systems.

The projected polygon method has been widely recognized and utilized in existing literature to assess stability [23]. Further analysis of the vehicle rollover risk during simula-

tion was conducted using the projected polygon method. Initially, the normalized critical stability boundary of the vehicle was obtained. Subsequently, the discrete data obtained during the full-condition simulation were processed and projected, yielding the results shown in Figure 15. In this figure, the blue solid line represents the critical stability boundary, while the red dots indicate the ends of the resultant vectors combining the longitudinal and lateral forces of the vehicle. Based on the analysis of the results, it is evident that the ends of the resultant vectors consistently remained within the critical stability boundary throughout the entire simulation process, indicating that the vehicle maintained stability and was not subject to any rollover risks throughout. This further validates the effectiveness of the present work.

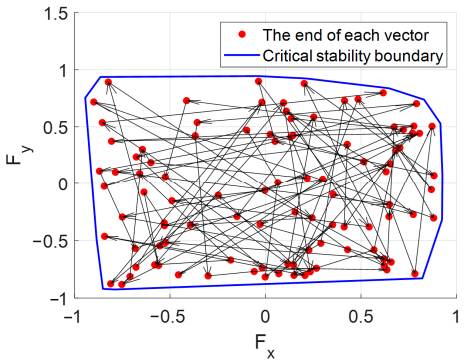


Figure 15. Risk analysis using projected polygon method.

5. Conclusions

This research considered the safety and rollover stability issues in the process of obstacle avoidance trajectory planning, studied the non-cooperative game model construction method of the obstacle avoidance process with fusion risk, and proposed a Kmeans-SMOTE risk clustering fusion method for the obstacle avoidance trajectory planning to determine the optimal feasible region. On this basis, the improved artificial potential field index and rollover dynamics index of heavy trucks were established, and an obstacle avoidance trajectory planning method integrated with rollover dynamics was proposed, which can ensure vehicle safety and traffic efficiency while reducing rollover risk and obstacle avoidance trajectory.

The results show that compared with the basic PSO trajectory planning method, the average driving risk obtained by the method proposed in this paper was reduced by 23.84%, and the peak value was reduced by 33.19%. In addition, in terms of dynamics parameters, the peak value of lateral acceleration was reduced by 33.58%, and the average value was reduced by 29.06%; the peak value of roll angle was reduced by 18.26%, and the average value was reduced by 13.82%, which indicates that the method proposed in this paper effectively reduces the probability of the controlled vehicle's instability and rollover.

This paper makes progress in trajectory planning for vehicles, but there is still significant room for improvement. A crucial area for future research is understanding the relationship between the effectiveness of the planning and the road surface's grip (adhesion coefficient). Future studies will focus on developing advanced methods to precisely estimate both the road's grip and the height of the vehicle's center of gravity. These advancements will allow the model's parameters to be adjusted dynamically (real-time) based on driving conditions. This will ultimately lead to more adaptable algorithms that can perform well in a wider range of situations.

In the future, further study on the impact of dynamic load uncertainty on driving risk will be carried out and the proposed algorithm will be applied to a real vehicle test for effectiveness verification.

Author Contributions: Conceptualization, H.W., Z.W., J.L. and L.S.; methodology, H.W.; software, Z.W. and J.L.; validation, L.S.; writing—original draft preparation, H.W.; writing—review and editing, H.W., Z.W., J.L. and L.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Jiangsu Outstanding Youth Fund Project under Grant BK20220078.

Data Availability Statement: The original contributions presented in the study are included in the article, further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Boggio, M.; Novara, C.; Taragna, M. Trajectory Planning and Control for Autonomous Vehicles: A "Fast" Data-Aided NMPC Approach. *Eur. J. Control* **2023**, *74*, 100857. [CrossRef]
2. Yan, Y.; Wang, J.; Wang, Y.; Hu, C.; Huang, H.; Yin, G. A Cooperative Trajectory Planning System Based on the Passengers' Individual Preferences of Aggressiveness. *IEEE Trans. Veh. Technol.* **2023**, *72*, 395–406. [CrossRef]
3. Lu, S.; Ding, B.; Li, Y. Minimum-Jerk Trajectory Planning Pertaining to a Translational 3-Degree-of-Freedom Parallel Manipulator Through Piecewise Quintic Polynomials Interpolation. *Adv. Mech. Eng.* **2020**, *12*, 460–470. [CrossRef]
4. Bianco, C.G.L.; Piazzi, A. Minimum-Time Trajectory Planning of Mechanical Manipulators under Dynamic Constraints. *Int. J. Control* **2010**, *75*, 967–980. [CrossRef]
5. Gao, Y.; Gordon, T.; Lidberg, M. Optimal Control of Brakes and Steering for Autonomous Collision Avoidance Using Modified Hamiltonian Algorithm. *Veh. Syst. Dyn.* **2019**, *57*, 1224–1240. [CrossRef]
6. Liu, L.-S.; Lin, J.-F.; Yao, J.-X.; He, D.-W.; Zheng, J.-S.; Huang, J.; Shi, P. Path Planning for Smart Car Based on Dijkstra Algorithm and Dynamic Window Approach. *Wirel. Commun. Mob. Comput.* **2021**, *2021*, 8881684. [CrossRef]
7. Bloch, M.; Eitrheim, M.; Mackay, A.; Sousa, E. Connected, Cooperative and Automated Driving: Stepping away from dynamic function allocation towards human-machine collaboration. *Transp. Res. Procedia* **2023**, *72*, 431–438. [CrossRef]
8. Li, A.; Ma, X.; Guo, J.; Zhang, J.; Wang, J.; Zhao, K.; Li, Y. Driver fatigue detection and human-machine cooperative decision-making for road scenarios. *Multimed. Tools Appl.* **2023**, *83*, 12487–12518. [CrossRef]
9. Jiang, B.; Li, X.; Zeng, Y.; Liu, D. Human-Machine Cooperative Trajectory Planning for Semi-Autonomous Driving Based on the Understanding of Behavioral Semantics. *Electronics* **2021**, *10*, 946. [CrossRef]
10. Lu, Y.; Zhang, J.; Li, H.; Han, Y. A novel adaptive model following controller to enhance steering and roll stability of heavy vehicle. *J. Mech. Sci. Technol.* **2021**, *35*, 5287–5297. [CrossRef]
11. Miyamoto, S.; Oya, M. Improved rollover prevention controller for heavy vehicles with varying velocity and values of vehicle parameters. *Artif. Life Robot.* **2022**, *27*, 521–527. [CrossRef]
12. Lu, Y.; Han, Y.; Huang, W.; Wang, Y. Sliding mode control for overturning prevention and hardware-in-loop experiment of heavy-duty vehicles based on dynamical load transfer ratio prediction. *Proc. Inst. Mech. Eng. Part K J. Multi-Body Dyn.* **2022**, *236*, 68–83. [CrossRef]
13. Wiseman, Y. Autonomous Vehicles. In *Encyclopedia of Information Science and Technology*, 5th ed.; IGI Global: Hershey, PA, USA, 2020; Volume 1, pp. 1–11.
14. Aghabayk, K.; Sarvi, M.; Young, W. Including Heavy Vehicles in a Car-Following Model: Modelling, Calibrating and Validating: Including Heavy Vehicles in a Car-Following Model. *J. Adv. Transp.* **2016**, *50*, 1432–1446. [CrossRef]
15. Wang, J.Y.; Tomizuka, M. Dynamic Analyses and Robust Steering Controller Design for Automated Lane Guidance of Heavy-Duty Vehicles. *Asian J. Control* **2010**, *2*, 140–154. [CrossRef]
16. Yang, X.; Chen, J.; Dang, Y.; Luo, H.; Tang, Y.; Liao, C.; Chen, P.; Cheng, K.-T. Fast Depth Prediction and Obstacle Avoidance on a Monocular Drone Using Probabilistic Convolutional Neural Network. *IEEE Trans. Intell. Transp. Syst.* **2019**, *22*, 156–167. [CrossRef]
17. Xie, N.G.; Sun, L.S.; Bao, J.H.; Fang, H. Multi-Objective Optimization Design of Arch Dam Shape Based on Unselfish Cooperation Game Model. *J. Hydraul. Eng.* **2007**, *38*, 1109–1114.
18. Zhao, S.; Zhao, J. Machine Learning Theory in the Strategic Management of Regional Risk Factors Measurement. *Mob. Inf. Syst.* **2021**, *2021 Pt 9*, 2770830.1–2770830.10. [CrossRef]
19. Arjun, P.; Manoj, K.G. Improved Hybrid Bag-Boost Ensemble with K-Means-SMOTE–ENN Technique for Handling Noisy Class Imbalanced Data. *Comput. J.* **2021**, *65*, 124–138.
20. Georgios, D.; Fernando, B.; Felix, L. Improving Imbalanced Learning through a Heuristic Oversampling Method Based on K-Means and SMOTE. *Inf. Sci.* **2018**, *465*, 1–20.
21. Soheyli, S.; Mayam, M.H.S.; Mehrjoo, M. Modeling a Novel CCHP System Including Solar and Wind Renewable Energy Resources and Sizing by a CC-MOPSO Algorithm. *Appl. Energy* **2016**, *184*, 375–395. [CrossRef]
22. Zhang, D.; Ren, Z.; Koh, C.-S. Multi-Objective Optimal Design of a NEMA Design D Three-phase Induction Machine Utilizing Gaussian-MOPSO Algorithm. *J. Electr. Eng. Technol.* **2014**, *9*, 184–189. [CrossRef]
23. Rinchi, M.; Pugi, L.; Bartolini, F.; Gozzi, L. Design of Control System to Prevent Forklift Capsize. *Int. J. Veh. Syst. Model. Test.* **2010**, *5*, 35–58. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.