

Article

Method for the Trajectory Tracking Control of Unmanned Ground Vehicles Based on Chaotic Particle Swarm Optimization and Model Predictive Control

Mengtao Jin ^{1,*}, Junmin Li ² and Te Chen ^{3,4} 

¹ School of Mechanical, Electrical and Automotive Engineering, Xuchang Vocational Technical College, Xuchang 461000, China

² School of Mechanical Engineering, Anyang Institute of Technology, Anyang 455000, China; userljm704@163.com

³ Automotive Engineering Research Institute, Jiangsu University, Zhenjiang 212013, China; ujschente@163.com

⁴ Key Laboratory of Advanced Manufacture Technology for Automobile Parts (Chongqing University of Technology), Ministry of Education, Chongqing 400054, China

* Correspondence: xcjinmt@126.com

Abstract: The symmetry principle has significant guiding value in vehicle dynamics modeling and motion control. In complex driving scenarios, there are problems of low accuracy and large time delay in the trajectory tracking control of unmanned ground vehicles. In order to solve this problem and improve the motion control of unmanned ground vehicles, a vehicle coordination control method based on chaotic particle swarm optimization (CPSO) and model predictive control (MPC) algorithms is proposed. To achieve coordinated control of vehicle trajectory tracking and yaw stability, a model predictive controller was designed with the objective of minimizing trajectory tracking errors and yaw stability tracking errors. The required front wheel angle and yaw torque control variables were obtained by solving nonlinear constraint optimization. At the same time, considering the problems of low computational efficiency, high solving time, and local optimization in model predictive control, a chaotic particle swarm optimization algorithm is introduced to solve the optimization constraint problem within model predictive control, thereby effectively improving the computational efficiency and accuracy of the model predictive trajectory tracking controller. The results show that compared with MPC, the multi-objective function optimization solution time and vehicle lane changing time of CPSOMPC improved by 24.51% and 7.21%, respectively, which indicates the coordinated control method that combines the CPSO and MPC algorithms can effectively improve trajectory tracking performance while ensuring vehicle lateral stability.

Keywords: trajectory tracking; coordinated control; model predictive control; vehicle stability



Citation: Jin, M.; Li, J.; Chen, T. Method for the Trajectory Tracking Control of Unmanned Ground Vehicles Based on Chaotic Particle Swarm Optimization and Model Predictive Control. *Symmetry* **2024**, *16*, 708. <https://doi.org/10.3390/sym16060708>

Academic Editor: Theodore E. Simos

Received: 7 May 2024

Revised: 26 May 2024

Accepted: 3 June 2024

Published: 7 June 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

With the rapid development of artificial intelligence and autonomous driving technology, intelligent vehicles have become a research hotspot in the fields of automobiles and transportation. However, in practical applications, intelligent vehicles need to face various complex driving scenarios, such as busy urban traffic, adverse weather conditions, and sudden traffic events. These scenarios place extremely high demands on vehicle control. The driving scenarios and dynamic coupling relationships of intelligent vehicles are relatively complex. At the same time, the addition of functional modules to electronic control systems further enhances the complexity and computational complexity of the motion control process [1,2]. Therefore, vehicle control systems have high requirements in terms of computational efficiency and real-time performance [3–5]. Trajectory tracking is one of the important aspects of autonomous vehicle technology. Intelligent vehicle trajectory tracking control aims to achieve precise tracking of the vehicle's predetermined trajectory through advanced control algorithms and the system's architecture, thus improving driving

safety, comfort, and efficiency. In recent years, researchers in this field have continuously improved tracking control technology through breakthroughs in sensor technology and innovation in control theory [6–8]. For model predictive control methods, it is necessary to deal with finite time domain optimization problems with future time steps, and their computational efficiency and optimization solving ability directly determine the control accuracy and real-time performance of the model predictive controller, which in turn affect the effectiveness of vehicle trajectory tracking and stability control [9,10]. Thus, the optimization and improvement of model predictive control algorithms have important practical significance.

The negative impact of time delay factors on intelligent vehicle trajectory tracking control is significant. The existence of time delay problems may lead to a decrease in a control system's performance and may even cause instability in vehicle motion control. Multiple compensation and control methods can be adopted for time delay problems [11–13]. The slow response and trajectory deviation caused by time delays may prevent vehicles from making timely and correct judgments and executing corresponding actions in emergency situations, thereby increasing the risk of accidents. Therefore, in intelligent vehicle trajectory tracking control, effective handling of time delay issues is an important link for ensuring effectiveness in vehicle trajectory tracking control and driving safety [14–16]. By optimizing the processing speed of sensor data, constructing predictive control algorithms with high solving ability, introducing robust control strategies, adjusting control parameters in real time, and establishing accurate models, the impact of time delay factors on control systems can be effectively mitigated, and the accuracy and stability of trajectory tracking can be improved. For the model predictive control algorithm in trajectory tracking control, the real-time performance of the control system can be improved by adjusting the optimization solution rate of the optimization predictive control algorithm [17,18].

At the same time, vehicle driving stability and chassis control methods within the trajectory tracking control process cannot be ignored. The trajectory tracking of intelligent vehicles mainly refers to the tracking of pre-set trajectories during the driving process, maintaining accurate matching with the trajectories [19–21]. Then, in order to achieve this goal, the automotive chassis control system needs to be able to accurately control the vehicle's dynamic behavior such as speed, steering, and drive distribution so that the vehicle can operate according to the expected trajectory. Chassis coordination control is the key to achieving precise control and driving safety [22,23]. In-depth research on advanced control algorithms and technologies is of great value for improving the processing ability and system performance of vehicle motion control systems in the face of time delay problems [24,25]. Common compensation methods for time delay problems include predictive control, the Smith predictor, etc. These methods offset the impact of time delays by predicting future states or adjusting system models. In addition, some advanced control methods such as adaptive control and fuzzy control can also effectively address the problem of time delay [26,27]. The particle swarm optimization algorithm, due to its inherent advantages of fewer adjustment parameters, fast solving speed, and global optimization, can effectively avoid the problems of large computational complexity and local optima faced by conventional model predictive controllers when solving constrained optimization problems [28,29]. The use of the particle swarm optimization algorithm to solve the constraint optimization problem in model predictive controllers can effectively integrate the advantages of the particle swarm optimization algorithm and model predictive control algorithm and improve the optimization solving ability and control application effect of model predictive controllers, thereby achieving improvements in unmanned vehicle trajectory tracking control accuracy and real-time control [30,31], which will have important research significance and application value.

Symmetry theory plays an important role in vehicle dynamics control. In terms of vehicle dynamics modeling, and in order to construct a simplified vehicle model, the principle of symmetry can be used to simplify the complexity of the model while maintaining its accuracy and effectiveness [32,33]. In terms of vehicle stability control, symmetry

theory can be used to guide the design and implementation of vehicle stability control. The stability control of vehicles is an important aspect of vehicle dynamics control, including longitudinal stability, lateral stability, and vertical stability. Through the principle of symmetry, control algorithms can be optimized, and vehicle dynamic stability boundaries can be set to improve the stability of the vehicle under various road conditions and driving conditions, thereby improving the driving safety of the vehicle. Therefore, the development and implementation of relevant algorithms can further improve the dynamic control effect and trajectory tracking accuracy of autonomous ground vehicles.

To improve the accuracy and real-time performance of unmanned vehicle trajectory tracking control, in this paper, a vehicle trajectory model predictive control method based on chaotic particle swarm optimization was studied. A trajectory tracking model predictive controller was constructed based on the dynamics model of unmanned vehicles, and the required front wheel angles for trajectory tracking and yaw stability control were obtained by combining the objective function and nonlinear constraint optimization solution method. On this basis, a nonlinear constraint optimization problem based on chaotic particle swarm optimization was studied to improve the solving efficiency and tracking control accuracy of the model predictive controller, thereby achieving effective improvement of unmanned vehicle trajectory tracking and yaw stability control effects. In this study, in order to address the limited efficiency of MPC multi-objective constraint function optimization and the real-time performance of vehicle motion control, the CPSO algorithm was introduced to significantly improve the optimization and solving ability of MPC, thereby enhancing the effectiveness of vehicle motion control.

The rest of this paper is organized as follows. The dynamics model of the unmanned ground vehicle is presented in Section 2. The trajectory tracking controller based on chaotic particle swarm optimization MPC is analyzed in Section 3. The verification of results is presented in Section 4, and the conclusive remarks and discussions are presented in Section 5.

2. Dynamics Model of an Unmanned Ground Vehicle

Figure 1 shows a schematic diagram of the monorail dynamics model for autonomous vehicles. This study only considers the dynamic control problem of intelligent vehicles in the horizontal motion plane and does not involve the vertical motion characteristics and dynamic control effects of the vehicle. We must establish a vehicle dynamics equation that considers lateral and yaw degrees of freedom to characterize the dynamic relationship between vehicle state parameters and construct a dynamic coordinate system xoy , where the origin of the dynamic coordinate system coincides with the vehicle center of mass, and the x -axis and y -axis represent the longitudinal and lateral motion directions of the vehicle, respectively. Assuming the tyre characteristics are consistent, we can ignore the dynamic equations of the vehicle suspension system and the vehicle's motion in pitch, roll, and vertical directions and only discuss the vehicle's motion in the xoy plane. Thus, the dynamic model equations of unmanned vehicles can be represented as

$$m\dot{v}_x = mv_y\gamma + 2F_{xf}\cos\delta_f - 2F_{yf}\sin\delta_f + 2F_{xr}, \quad (1)$$

$$m\dot{v}_y = -mv_x\gamma + 2F_{xf}\sin\delta_f + 2F_{yf}\cos\delta_f + 2F_{yr}, \quad (2)$$

$$I_z\dot{\gamma} = l_f F_{yf}\cos\delta_f - l_r F_{yr} + M_z, \quad (3)$$

where m is the vehicle mass; v_x is the longitudinal speed; v_y is the lateral speed; γ is the yaw rate of the vehicle; l_f and l_r , respectively, represent the distance from the vehicle's center of mass to the front and rear axes; I_z is the moment of inertia around the z axis and the heading angle of the vehicle; M_z is the external yaw moment; F_{xf} and F_{xr} , respectively, represent the longitudinal tyre force of the front and rear axes; and F_{yf} and F_{yr} represent

front and rear lateral tyre forces, respectively. At the same time, the tyre model based on the empirical magic formula is used to estimate the longitudinal and transverse tyre forces:

$$Y(\lambda) = D \sin\{\text{Carctan}[B\lambda - E(B\lambda - \arctan(B\lambda))]\}, \quad (4)$$

where $Y(\lambda)$ is the output of the model and is used to represent the longitudinal force, transverse force or aligning moment of the tyre; B is the stiffness coefficient; C is the curve shape coefficient; D is the peak coefficient; E is the curve curvature coefficient; and λ is the side deflection angle or longitudinal slip rate of the tyre. The tyre sideslip angle can be expressed as

$$\begin{aligned} \alpha_f &= (l_f \gamma + v_y) / v_x - \delta_f, \\ \alpha_r &= (v_y - l_r \gamma) / v_x \end{aligned} \quad (5)$$

where δ_f is the front wheel steering angle, and α_f and α_r are the front and rear axle tyre sideslip angle, respectively. The vehicle coordinates in the geodetic coordinate system can be expressed as

$$\begin{aligned} \dot{Y} &= \dot{x} \sin \varphi + \dot{y} \cos \varphi \\ \dot{X} &= \dot{x} \cos \varphi - \dot{y} \sin \varphi \end{aligned} \quad (6)$$

where, $\dot{x} = v_x$, $\dot{y} = v_y$, $\dot{\varphi} = \gamma$.

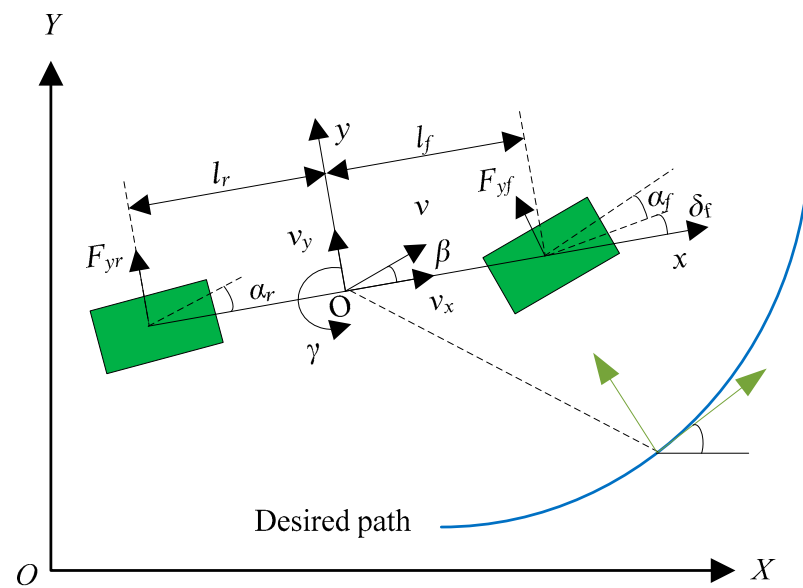


Figure 1. Vehicle dynamic model.

3. Trajectory Tracking Controller Based on Chaotic Particle Swarm Optimization MPC

3.1. Trajectory Tracking Control Method Based on MPC

The actual driving scene of unmanned vehicle is complex and changeable, so it is necessary to consider the influence of many nonlinear interference factors on the model accuracy. At the same time, for the model prediction controller, the influence of nonlinear factors on the control accuracy should be dealt with in the design process. Combining Equations (1)–(3) and (6), and in order to improve the control accuracy of the vehicle motion controller, the vehicle model is linearized by Taylor expansion so that the linear time-varying vehicle model can be expressed as

$$\begin{aligned} \dot{x}_k &= f(x_k, u_k) = A_k x_k + B_k u_k, \\ y &= C_k x_k \end{aligned} \quad (7)$$

where $x_k = [\dot{x} \ \dot{y} \ \varphi \ \gamma \ Y \ X]^T$ is the system state quantity, $u_k = [\delta_f \ M_z]^T$ is the control input, $A_k = \frac{\partial f}{\partial x}$ is the state transition, $B_k = \frac{\partial f}{\partial u}$ is the input matrix, and $C_k = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$ is the measurement matrix.

According to Equation (7), the prediction equation does not constrain the increment of vehicle control accurately enough. In order to solve this problem, the incremental model is proposed as $\Delta x(k+1) = A_k \Delta x(k) + B_k \Delta u(k)$, and the system state with increments can be represented as $x'(k|k) = [x(k|k) \ u(k-1|k)]^T$. The system model in Equation (7) can then be extended as

$$\begin{aligned} x'(k+1|k) &= A'_k x(k|k) + B'_k \Delta u(k|k) \\ \eta(k|k) &= C_k x(k|k) \end{aligned} \quad (8)$$

where $A'_k = \begin{bmatrix} A_k & B_k \\ 0_{1 \times 6} & 1 \end{bmatrix}$, $B'_k = [B_k \ 1]^T$, $\eta(k|k)$ is the output at the current time.

The prediction range and control range are defined as h_p and h_c , respectively, and meet the conditions that the output of the next time in the system can be obtained by using the state variable and the control variable of the current time. The calculation formula of the system prediction equation at the k th sampling time can be expressed as

$$Y(k+1|k) = \Psi_{\xi} \xi(k) + \Theta_{\xi} \Delta U(k), \quad (9)$$

where $Y(k+1|k)$ is the system output at the k th sampling time.

In order to realize the coordinated control of vehicle trajectory tracking and yaw stability, the objective function is designed as

$$J(k) = \sum_{j=1}^{h_p} q_1 \|\Delta e(k+j|k)\|_Q^2 + \sum_{j=0}^{h_c-1} q_2 \|\Delta u(k+j|k)\|_R^2 + q_3 \varepsilon^2, \quad (10)$$

where $\Delta e(k+j|k)$ is the tracking error of vehicle trajectory and yaw velocity; $\Delta u(k+j|k)$ is the incremental control input; Q and R are the state increment weighting matrix and control increment weighting matrix, respectively; ε is the relaxation factor; and q_1 , q_2 and q_3 are the system adjustment parameters. In the model prediction controller, the trajectory tracking error and yaw stabilization tracking error are included in Δe , and the accuracy of target tracking control is reflected. Δu is used to represent the control effect of front wheel steering angle and extra yaw moment. The purpose of the controller is to make the tracking error $\|\Delta e(k+j|k)\|_Q^2$ as close to zero as possible. At this point, the objective function in (13) can be reduced to an expression for solving the control increment

$$J(\xi, u, \Delta U) = \Delta U_t H_t \Delta U_t^T + G_t \Delta U_t^T, \quad (11)$$

where $H_t = \begin{bmatrix} \Theta_{\xi}^T Q \Theta_{\xi} & 0 \\ 0 & \rho \end{bmatrix}$ is the positive definite matrix and used to adjust the control rate of change; $G_t = [2E_t^T Q \Theta_{\xi} \ 0]$ and E_t are the tracking errors in the prediction domain; and H_t and G_t are the input for quantum particle swarm optimization.

The tyre angle constraint is defined as $|\alpha_{f,r}| \leq 2^\circ$, and the road adhesion coefficient constraint is defined as $\begin{cases} |a_y| \leq \mu_g g \\ a_{y,\min} - \varepsilon \leq a_y \leq a_{y,\max} + \varepsilon \end{cases}$. Then, the coordinated control problem of vehicle trajectory tracking and lateral stability can be transformed into an opti-

mization problem under constrained conditions, and the optimization objective function can be expressed as

$$\begin{cases} \min \Delta U_t H_t \Delta U_t^T + G_t \Delta U_t^T \\ \Delta U_t \\ s.t. \Delta U_{\min} \leq \Delta U_t \leq \Delta U_{\max} \\ U_{\min} \leq A_u \Delta U_t + U_t \leq U_{\max}, \\ U_{h,\min} \leq U_h \leq U_{h,\max} \\ U_{s,\min} - \varepsilon \leq U_s \leq U_{s,\max} + \varepsilon \end{cases} \quad (12)$$

where $U_{\min}$ and $U_{\max}$ are, respectively, the minimum and maximum values of the input quantity; $\Delta U_{\min}$ and $\Delta U_{\max}$ are the minimum and maximum increments, respectively; $Y_{h,\min}$ and $Y_{h,\max}$ are the minimum and maximum values, respectively, of the hard output limits; $Y_{s,\min}$ and $Y_{s,\max}$ are the minimum and maximum values, respectively, of the soft output limits; and A_u is the Kronecker product.

3.2. Nonlinear Constrained Optimization Calculation Based on Chaotic Particle Swarm Optimization

Traditional MPC requires solving a finite time range of open-loop optimization problems at each sampling moment, which is very time-consuming. Particle swarm optimization (PSO) can effectively improve the computational efficiency of the MPC optimal solution and reduce the output delay. The particle swarm optimization algorithm uses velocity and position to represent particle state and realizes global optimization through population initialization, fitness function calculation, and parameter update iteration. The update formula is as follows

$$\begin{aligned} v_{id}^{k+1} &= wv_{id}^k + c_1r_1(p_{id}^k - v_{id}^k) + c_2r_2(p_{gd}^k - v_{id}^k), \\ x_{id}^{k+1} &= x_{id}^k + v_{id}^{k+1} \end{aligned} \quad (13)$$

where v_{id}^k is the j -dimensional velocity of the i -th particle; x_{id}^k is the j -dimensional position of the i -th particle; w is the inertia weight coefficient; c_1 and c_2 are both learning factors greater than 0; and c_1 and c_2 satisfy the random distribution of $[0, 1]$.

However, the disadvantage of PSO is that it is easy for local minimum points to appear, resulting in limited local convergence and resolution space. When PSO is used to optimize multi-peak function, it can easily fall into local minima, which leads to slow convergence and means accurate results cannot be obtained. Chaos is a common nonlinear phenomenon in nature, and its behavior is random. Chaos has the characteristics of randomness, ergodicity, and regularity because of its exquisite internal structure. The chaos optimization algorithm can avoid the disadvantage of local extreme values of particle swarm optimization algorithm by maintaining population diversity.

The logical model involved in the chaotic optimization method can be expressed as

$$z_{n+1} = f(z_n) = \mu z_n(1 - z_n), \quad (14)$$

where μ is the control parameter and its value is 4, and z_n is the chaos variable value of the n th iteration. Chaotic variables are searched by their own laws of motion, and their randomness helps to improve the ability of large-scale search. Since the particle swarm optimization algorithm is prone to prematurity and local optimization problems, in addition, the uncertainty of the initial random particles can easily cause different degrees of difference in the iterative calculation results of the algorithm. Thus, the chaos variable can be converted into the linear mapping formula of the optimization variable and expressed as

$$x_{n+1} = x_{\min} + z_{n+1}(x_{\max} - x_{\min}), \quad (15)$$

where x_{min} and x_{max} are the minimum and maximum values of the optimization variable. The constrained optimization problem in Equation (12) can be expressed as follows

$$\begin{aligned} \min J(x), X &= [x_1, x_2, \dots, x_D] \\ \text{s.t. } x_i &\in [a_i, b_i], (i = 1, 2, \dots, D) \end{aligned} \quad (16)$$

where X is the input variable, D is the number of variables, $J(x)$ is the objective function, and $[a_i, b_i]$ is the range of x_i (that is, the constraints of the optimization objective function). The purpose of the optimization algorithm is to search for the optimal input to minimize the objective function by combining the objective function in Equation (10) and the optimization constraint of Equation (12).

The basic idea of chaotic particle swarm optimization is to introduce a chaotic state into the optimization variables. Firstly, the range of chaotic motion is transformed into the value range of optimization variables by the carrier method. Then, the obtained chaotic variables are represented as particles and searched by using cooperation and competition among particles. At the same time, small perturbations are added for each chaos variable and the optimal solution is searched by iteratively updating the velocity and position of the particles. The iterative formula of particle swarm optimization algorithm based on Tent mapping can be expressed as

$$z_{n+1} = \mu(1 - 2|z_n - 0.5|), 0 \leq z_0 \leq 1, \quad (17)$$

where μ is the control parameter from 0 to 1. If the value of μ is 1, the Tent map is in a completely chaotic state. According to the chaotic search method, a small chaotic disturbance is added to the current optimal solution:

$$z'_k = (1 - k_{z1})z^* + k_{z1}z_k, \quad (18)$$

where k_{z1} is the adjustment parameter between 0 and 1; z_k is the chaos variable at the k -th iteration; z'_k is the corresponding chaotic variable after adding a small disturbance to variable z_k and its chaotic sequence vectors generated by Tent map are $Z = (z_1, z_2, \dots, z_D)$ and $Z' = (z'_1, z'_2, \dots, z'_D)$ respectively; and the variable z^* is the optimal chaotic vector formed by mapping the current optimal vector. It can be expressed as

$$z^* = \frac{X^* - X_{min}}{X_{max} - X_{min}}. \quad (19)$$

In the initial search, if x_i changes a lot, then k_{z1} increases accordingly. During the search process, when x_i is close to the optimal, k_{z1} is gradually adjusted to be smaller so that the search can be optimized in the following determined region:

$$k_{z1} = 1 - \frac{k}{k_{max}}, \quad (20)$$

where k is the number of iterations, and k_{max} is the maximum number of iterations.

Finally, the iterative steps of the chaotic particle swarm optimization algorithm can be summarized as follows: (1) Initialize the chaotic variable, set the particle swarm value to 500, and generate the chaotic sequence based on Tent mapping. (2) The chaos variable is introduced into the optimization variable of Equation (16), and the carrier method is used to update it into the optimization variable. Meanwhile, the chaotic variable interval is transformed into the corresponding optimization variable interval. (3) Calculate the fitness function corresponding to the particle at the current iteration time. If the position of the particle after iteration is better than the individual historical optimal position or the population optimal position, the individual historical optimal value and the population optimal value of the particle are updated by comparing the results. (4) Update particle velocity and position according to Equation (13). (5) The particle swarm is arranged in ascending order according to the fitness value of the particle. If the current optimal value does not change

after a certain number of iterations and the current number $k \leq \frac{2}{3}k_{max}$ of iterations, the step (6) is continued; otherwise, step (8) is performed. (6) Chaotic perturbation is introduced, and the optimal particle position is updated by chaotic search. Chaotic perturbations in Equations (18)–(20) are added to the optimization variables, corresponding to 50% particles in the current group with small fitness values, and the mapping optimization variables are calculated. Then, the chaos search is iteratively performed until the termination condition is met. (7) The particles in the particle swarm are arranged in ascending order, and the average fitness of the population is calculated and compared with the optimal fitness. If the comparison difference is small enough to reach the preset termination condition, the search is stopped, and the optimal solution is returned; otherwise, step (8) is performed. (8) If $k = k + 1$, return to Step 3. Combined with the above content, the chaotic particle swarm optimization algorithm flow is shown in Figure 2.

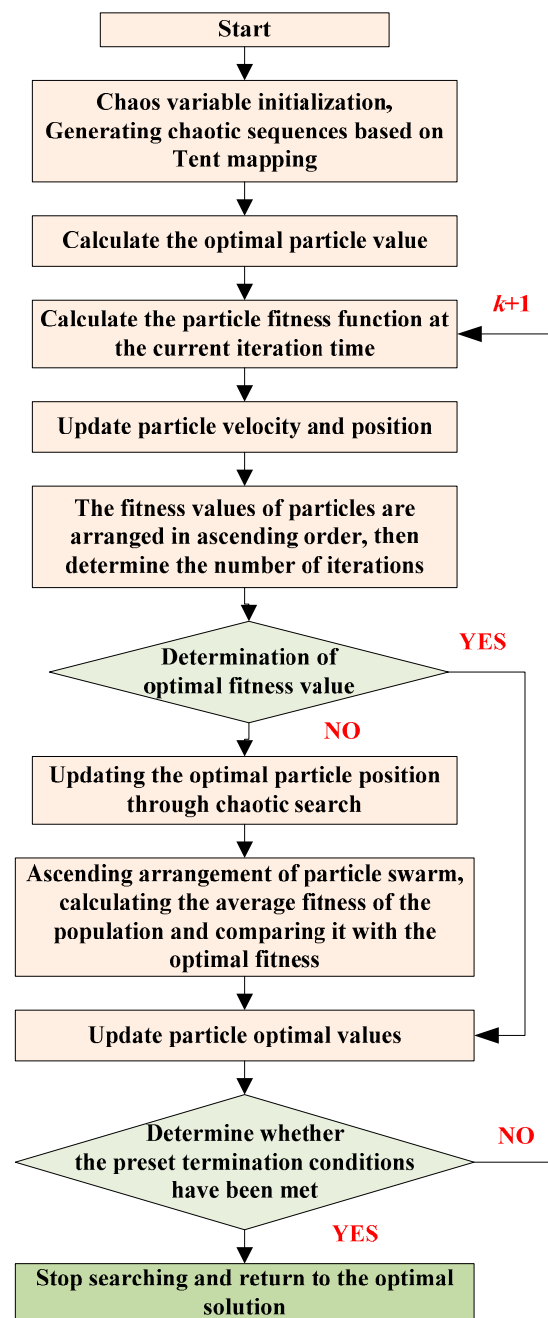


Figure 2. Process of chaos particle swarm optimization algorithm.

3.3. Tyre Force Distribution Method Considering Multi-Objective Optimization

According to the model prediction controller and chaotic particle swarm optimization algorithm, the required control values of front wheel angle and yaw moment are obtained. Next, the speed controller and the wheel hub motor torque optimization distribution method are designed to achieve the control requirements of yaw moment.

The first objective function for longitudinal vehicle driving control is

$$J_1 = (\bar{F}_x - F_{xd})^T W (\bar{F}_x - F_{xd}), \quad (21)$$

where $\bar{F}_x = [F_{x1} \ F_{x2} \ F_{x3} \ F_{x4}]^T$ represents the real-time tyre forces, F_{xd} the referenced tyre forces, and $W = \text{diag}[w_1 \ w_2 \ w_3 \ w_4]$ the assignment matrix.

Considering the load transfer issue of the tyre, the relationship of longitudinal force and vertical force is

$$\frac{F_{xfl}}{F_{zfl}} = \frac{F_{xfr}}{F_{zfr}} = \frac{F_{xrl}}{F_{zrl}} = \frac{F_{xrr}}{F_{zrr}}. \quad (22)$$

According to Equation (22), it can be deduced that $F_{xi} = \frac{F_{zi}}{\sum_{i=1}^4 F_{zi}} \sum_{i=1}^4 F_{xi} = \frac{F_{zi}}{mg} \sum_{i=1}^4 F_{xi}$. Thus,

the assignment coefficient of matrix W is

$$w_i = \frac{mg}{4F_{zi}}. \quad (23)$$

The second objective function for yaw moment control is

$$J_2 = \left(\bar{F}_x - B_x \begin{bmatrix} M_{zd} \\ F_{xd} \end{bmatrix} \right)^T \left(\bar{F}_x - B_x \begin{bmatrix} M_{zd} \\ F_{xd} \end{bmatrix} \right), \quad (24)$$

where B_x is the control matrix and can be written as $B_x = \begin{bmatrix} 1 & 1 & 1 & 1 \\ -\cos \delta_f b_s + l_f \sin \delta_f & \cos \delta_f b_s + l_f \sin \delta_f & -b & b \end{bmatrix}^T$, and M_{zd} is the desired yaw moment.

When the response capacity of the hub motor is fixed, the smaller the difference between the torque assigned to the hub motor and the current actual drive torque, the shorter the response time for the hub motor to reach the required drive torque, and the lower the instantaneous fluctuation and impact of the motor torque, which has certain significance for motor energy optimization and vehicle ride comfort. Therefore, the third objective function for suppressing instantaneous fluctuations and improving the motor response speed of in-motor torque is

$$J_3 = \sum_i^4 (\bar{F}_{xi} - F_{xa})^2, \quad (25)$$

where F_{xa} represents the actual tyre forces.

According to above three functions, the overall objective function is

$$\min J_0 = \min(k_1 J_1 + k_2 J_2 + k_3 J_3), \quad (26)$$

where k_1 , k_2 , and k_3 are regulating parameters.

Objective function 1 is designed to achieve longitudinal vehicle speed tracking control and obtain the overall longitudinal force driving demand. The overall longitudinal force driving demand can be expressed as

$$F_{xd} = F_c + F_f + F_w + F_i, \quad (27)$$

where F_c is the acceleration driving resistance, F_f is the friction rolling resistance, F_w is the air resistance, and F_i is the slope resistance. A PID controller is designed to realize the vehicle speed tracking control. The model equation of the controller can be expressed as

$$\frac{F_{xd} - F_f - F_w - F_i}{m} = k_p(v_{xd} - v_x) + k_i \int (v_{xd} - v_x) dt + k_d \frac{d(v_{xd} - v_x)}{dt}, \quad (28)$$

where k_p , k_i , and k_d are PID controller parameters. Thus, the required longitudinal force driving demand can be obtained according to (46).

In objective function 3, the relationship between the longitudinal force demand F_{xd} and the yaw moment demand M_z and the driving force of the four tyres can be expressed as

$$\begin{aligned} F_{xd} &= F_{x1} + F_{x2} + F_{x3} + F_{x4} \\ M_z &= (F_{x1} - F_{x2})l + (F_{x3} - F_{x4})l' \end{aligned} \quad (29)$$

By solving Equation (29), we can obtain $F_{x1} = \frac{F_{xd}}{2} + \frac{M_z}{2l} - F_{x2}$ and $F_{x3} = \frac{F_{xd}}{2} - \frac{M_z}{2l} - F_{x4}$. The combination of Formulas (27) and (29) is then obtained as follows:

$$J_3 = \left(\frac{F_{xd}}{2} + \frac{M_z}{2l} - F_{x2} - F_{a1} \right)^2 + (F_{x2} - F_{a2})^2 + \left(\frac{F_{xd}}{2} - \frac{M_z}{2l} - F_{x4} - F_{a3} \right)^2 + (F_{x4} - F_{a4})^2. \quad (30)$$

In combination with Equation (30), the partial derivative of F_{x2} and F_{x4} is set to 0, respectively, and the optimal solution below the objective function 3 is obtained:

$$\begin{aligned} F_{x2} &= \frac{F_{xd}}{4} + \frac{M_z}{4l} - \frac{F_{d1} - F_{d2}}{2} \\ F_{x4} &= \frac{F_{xd}}{4} - \frac{M_z}{4l} - \frac{F_{d3} - F_{d4}}{2} \end{aligned} \quad (31)$$

Thus, the tyre forces F_{x1} and F_{x3} are obtained as follows:

$$\begin{aligned} F_{x1} &= \frac{F_{xd}}{4} + \frac{M_z}{4l} + \frac{F_{d1} - F_{d2}}{2} \\ F_{x3} &= \frac{F_{xd}}{4} - \frac{M_z}{4l} + \frac{F_{d3} - F_{d4}}{2} \end{aligned} \quad (32)$$

Taking the partial derivative of subobjective functions J_1 and J_2 , we have $\frac{\partial J_1}{\partial \bar{F}_x} = W(\bar{F}_x - F_{xd})$, $\frac{\partial J_2}{\partial \bar{F}_x} = \left(\bar{F}_x - B_x \left[\frac{M_{zd}}{F_{xd}} \right] \right)$. According to $\frac{\partial J_f}{\partial \bar{F}_x} = k_1 \frac{\partial J_1}{\partial \bar{F}_x} + k_2 \frac{\partial J_2}{\partial \bar{F}_x} = 0$ and calculating the extreme value, the optimal result of tyre force distribution is

$$\bar{F}_x = (k_1 W + k_2 I)^{-1} \left[k_1 W_i F_{xd} + k_2 B_x \left[\frac{M_{zd}}{F_{xd}} \right] \right]. \quad (33)$$

In summary, the final comprehensive tyre force optimization distribution results can be derived:

$$\bar{F}_x = (k_1 W + k_2 I)^{-1} \left[k_1 W_i F_{xd} + k_2 B_x \left[\frac{M_{zd}}{F_{xd}} \right] \right] + k_3 \begin{bmatrix} \frac{F_{xd}}{4} + \frac{M_z}{4l} + \frac{F_{d1} - F_{d2}}{2} \\ \frac{F_{xd}}{4} + \frac{M_z}{4l} - \frac{F_{d1} - F_{d2}}{2} \\ \frac{F_{xd}}{4} - \frac{M_z}{4l} + \frac{F_{d3} - F_{d4}}{2} \\ \frac{F_{xd}}{4} - \frac{M_z}{4l} - \frac{F_{d3} - F_{d4}}{2} \end{bmatrix} \quad (34)$$

Combined with the above content, the trajectory tracking and yaw stability control strategies of unmanned vehicles can be obtained, as shown in Figure 3.

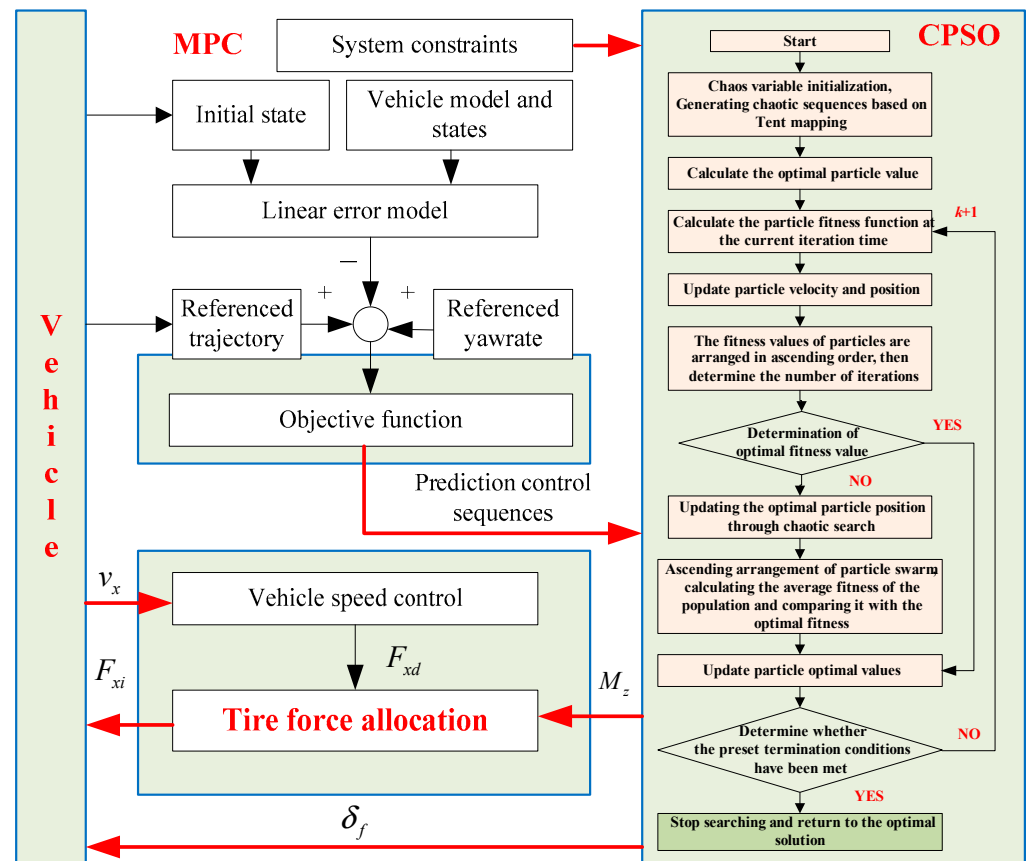


Figure 3. Overall vehicle control strategy.

4. Result Verification

In order to verify the effectiveness of the predictive control method based on a chaotic particle swarm optimization model in vehicle trajectory tracking and yaw stability control, the simulation test and verification under lane changing conditions were carried out in a Carsim and Simulink co-simulation environment. In the simulation condition, the reference trajectory was set as a typical lane changing trajectory condition, the target speed was set to 30 km/h, the control time domain and prediction time domain of the model prediction controller were set to 0.1, the chaotic particle swarm size was set to 500, and the maximum number of iterations was set to 500.

The comparison results of vehicle trajectories obtained are shown in Figure 4. As can be seen from Figure 4, the model predictive control method can achieve accurate tracking of the reference trajectory in lane changing conditions, among which the introduction of chaotic particle swarm optimization algorithm further improves the trajectory tracking accuracy. In contrast, the tracking results using only the model predictive control method show a more obvious time-lag phenomenon, the tracking accuracy is relatively low, and the recovery rate of the vehicle trajectory is also relatively lagging. At this time, the resulting comparison results of vehicle heading angle are shown in Figure 5. The tracking results of the vehicle heading angle are consistent with the variation trend of vehicle trajectory. Compared with a single model predictive control algorithm, the combination of model predictive control and chaotic particle swarm optimization algorithm can effectively improve the rolling update calculation rate of the controller, so that the change of vehicle heading angle can track the reference value faster, and the tracking real-time and tracking accuracy have been significantly improved.

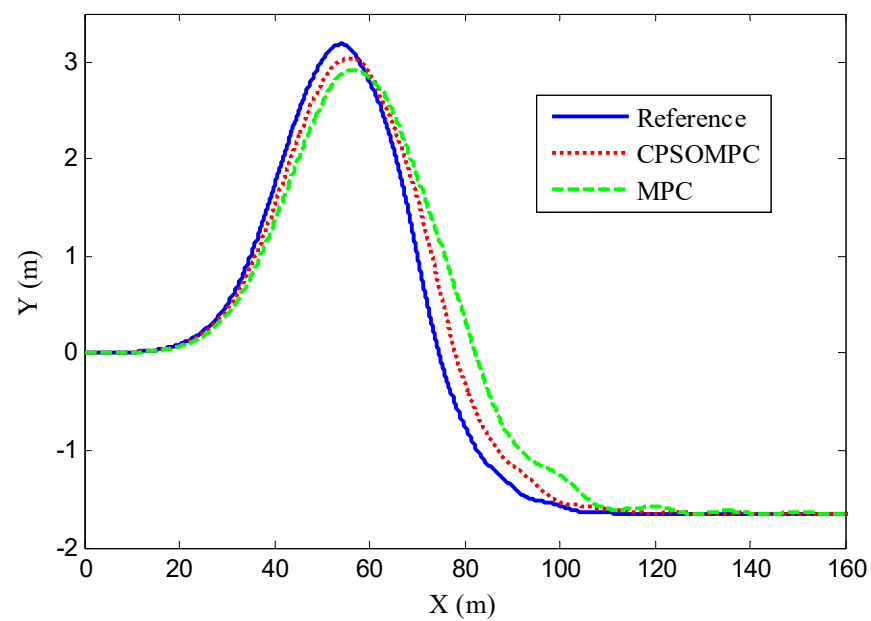


Figure 4. Comparison of vehicle driving trajectories.

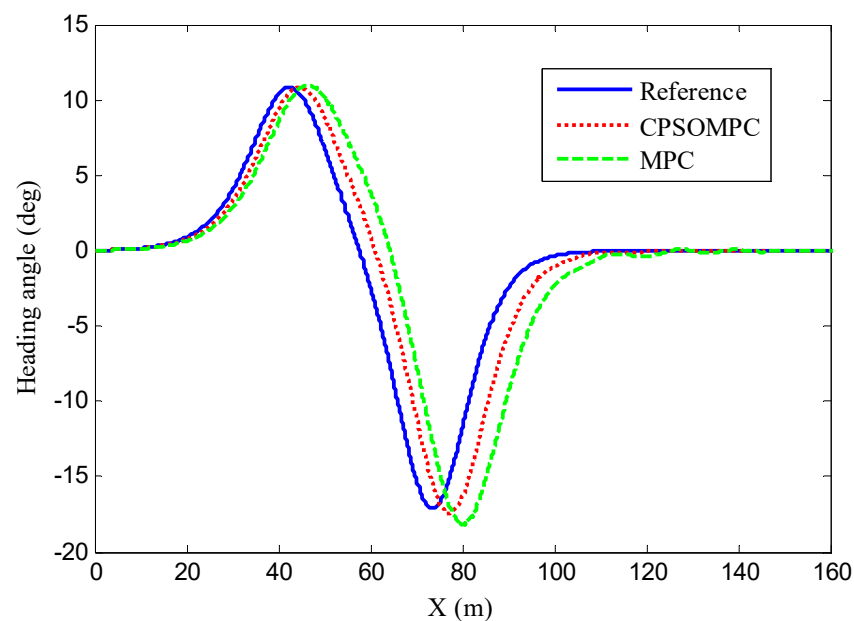


Figure 5. Comparison of vehicle heading angle.

The comparison results of vehicle trajectory tracking errors are shown in Figure 6. According to the comparison results of the lateral deviation of trajectory tracking, it can be seen that the model predictive control method combined with chaotic particle swarm optimization algorithm has a more obvious inhibition effect on the lateral deviation of trajectory tracking and has better performance in terms of fluctuation amplitude and convergence speed. In contrast, the single model predictive control method has relatively large error fluctuation, and its error convergence trend is obviously more lagging. The comparison results of course deviation also show the same change trend. The introduction of the chaotic particle swarm optimization algorithm can significantly reduce the amplitude of course deviation and accelerate the convergence speed.

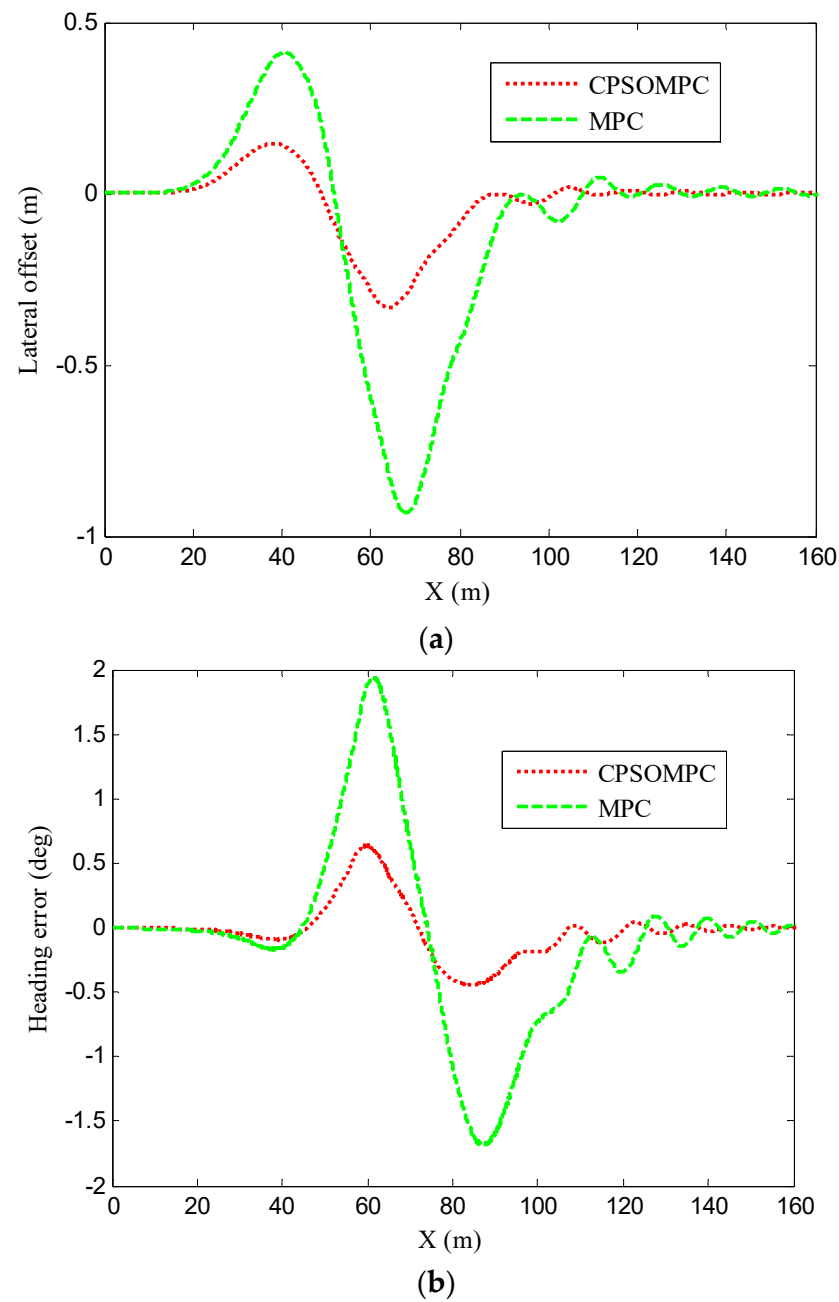


Figure 6. Comparison of vehicle trajectory tracking errors: (a) lateral deviation; (b) heading deviation.

The comparison results of vehicle yaw angular speed are shown in Figure 7. It can be seen from the figure that the vehicle yaw speed under the two control algorithms can track and control the reference yaw speed on the overall trend so as to maintain the vehicle yaw stability. The vehicle yaw velocity obtained by the model prediction controller based on chaotic particle swarm optimization algorithm is relatively better in its tracking effect; the change of vehicle yaw velocity is more stable on the whole, and the tracking control precision is relatively higher.

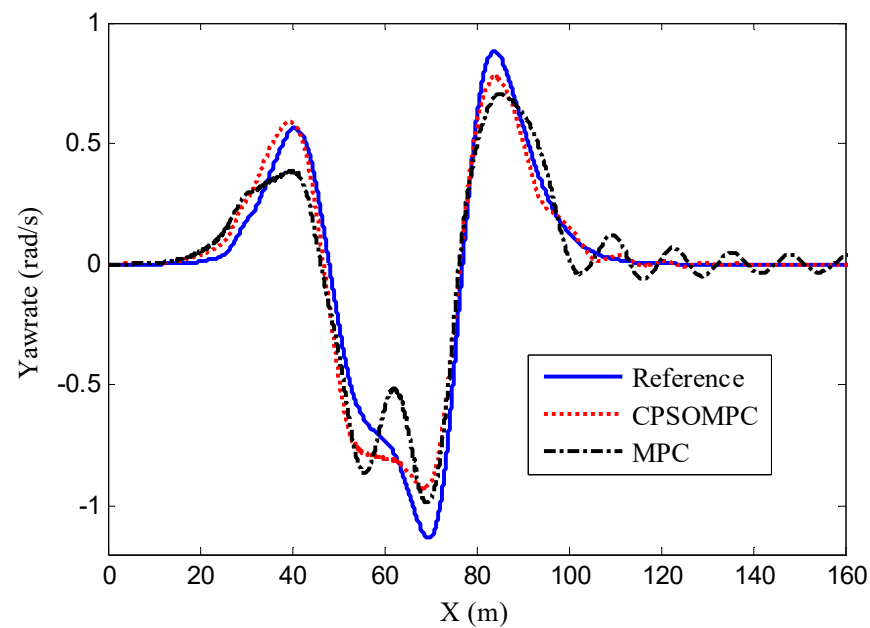


Figure 7. Comparison of vehicle yaw rate.

The results of tyre force optimization distribution are shown in Figure 8. It can be seen from the figure that the changing trend of four-wheel tyre force is directly related to the changing trend of vehicle driving trajectory and yaw angle speed. When the vehicle needs to achieve rapid lane change cruising through steering control, the tyre force of the vehicle also increases rapidly, providing sufficient driving force to the tyres to overcome the additional steering resistance. During the rapid lane change, the tyre force of the right front wheel is greater than that of the left front wheel, which helps to overcome the steering resistance of the vehicle during the lane change steering. At the same time, the tyre force of the right rear wheel is less than that of the left rear wheel, which helps to provide enough additional yaw moment of the vehicle while overcoming steering resistance, thus achieving yaw stability control while achieving steering maneuverability of the vehicle.

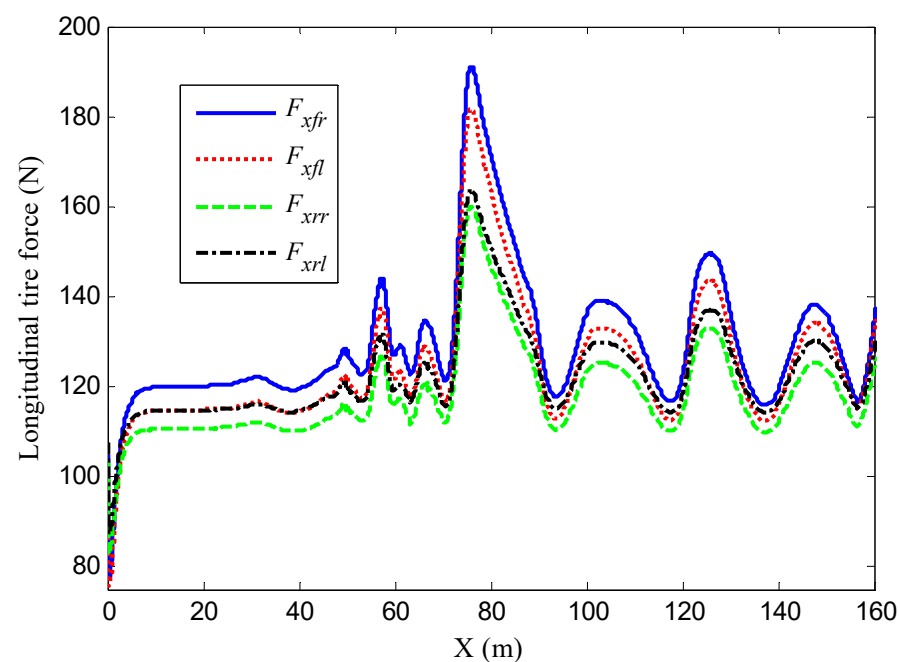


Figure 8. Optimization distribution results of tyre force.

In order to more clearly and intuitively reflect the direct role of the proposed CPSO algorithm in improving the MPC solving ability and thereby affecting the vehicle trajectory tracking and yaw stability control effects, further quantitative statistical analysis and comparison of control effects were conducted. We calculated and compared the lateral offset and heading error related to trajectory tracking performance under the conventional MPC algorithm, PSOMPC algorithm, and CPSOMPC algorithm; the yaw rate error related to vehicle stability; and the optimization solution time and lane changing time directly reflecting the time delay problem. The comparison results are listed in Table 1. As shown in Table 1, compared with conventional MPC algorithms, the average lateral offset and evaluation heading error of the PSOMPC and CPSOMPC algorithm are significantly reduced, the trajectory tracking error is significantly suppressed, and the trajectory tracking accuracy is improved. Moreover, the tracking error of the yaw rate is relatively small at this time, and the tracking control effect of the vehicle's yaw stability is improved. In addition, the optimization problem solving times under the PSOMPC and CPSOMPC optimization algorithm were 430 ms and 382 ms, respectively, which is significantly smaller than the 506 ms under the MPC algorithm. The times required for the vehicle's trajectory to complete lane changing were 12.36 s, 12.59 s and 13.32 s, respectively. The multi-objective function optimization solution time and vehicle lane changing time improved by 24.51% and 7.21%, respectively. This indicates that the application of the CPSO optimization algorithm effectively improves efficiency in solving the constraint objective function and reduces the controller operation time, thereby reducing the time required for vehicle lane changing and stabilization and effectively improving the maneuverability and real-time performance of vehicle motion control.

Table 1. Quantitative comparative analysis of control results.

	Average Lateral Offset (m)	Average Heading Error (deg)	Average Yaw rate Error (rad/s)	Optimal Calculation Time (ms)	Lane Changing Time (s)
MPC	0.0331	0.1296	0.0269	506	13.32
PSOMPC	0.0176	0.0593	0.0211	430	12.59
CPSOMPC	0.0136	0.0307	0.0183	382	12.36

5. Conclusions and Discussion

In order to improve the precision and real-time control of unmanned vehicle trajectory tracking, a coordinated control method of vehicle trajectory tracking and yaw stability based on chaotic particle swarm optimization and a model predictive control algorithm was proposed. A discrete linear time-varying vehicle dynamics equation was constructed based on the unmanned vehicle dynamics model, and a model prediction controller covering multi-objective constrained optimization was designed. On this basis, in order to avoid local optimization of multi-objective constraint optimization problems and improve the efficiency of optimization constraint solving, a nonlinear constraint optimization method based on chaotic particle swarm optimization algorithm was designed to improve the optimization solution effect of a model prediction controller, so as to effectively improve the trajectory tracking and yaw stability control effect. The results show that the trajectory tracking control method combined with chaotic particle swarm optimization algorithm and model predictive control algorithm can effectively improve the control precision and efficiency of the trajectory tracking process while maintaining the stability of vehicle yaw.

At present, the combination of the model predictive control and particle swarm optimization algorithms also encounters some limitations in solving the problem of time delay in intelligent vehicle trajectory tracking control. Model dependency is one of the most important factors. The performance of MPC largely depends on the vehicle dynamics model used. If the model is inaccurate or fails to fully consider the complexity of actual vehicle behavior, the effectiveness of the control algorithm may be affected. In addition, the accuracy of the model will also change with changes in vehicle status, environmental con-

ditions, and usage time, which further increases the difficulty of control and optimization. The related research appears to be of great significance. Meanwhile, robustness is also a key issue that urgently needs to be overcome. In intelligent vehicle trajectory tracking control, the system often faces various uncertainties due to changes in road conditions, vehicle status, sensor noise, and other factors. However, traditional model predictive control algorithms are often designed and optimized based on idealized models, with insufficient consideration of uncertainty factors in actual systems, which may lead to poor robustness of the algorithm in practical applications.

In future research, the time delay problems in intelligent vehicle dynamics control systems will be an important issue. We believe that targeted research can be conducted from three aspects: communication delays, actuator delays, and control system delays. By analyzing the mechanism of induced time delay, the characteristics of time delay can be effectively characterized, and a refined time delay model can be established. Then, combined with the mechanism model, a high-performance controller for intelligent vehicle dynamics time delay control can be designed, and advanced control algorithms can be matched and optimized based on the actual dynamic characteristics of the vehicle, thereby achieving real-time performance improvement of the control system. The problem of time delay in intelligent vehicle dynamics control is a complex and important research topic. By deeply analyzing the phenomenon and impacts of time delay, summarizing existing control strategies, developing advanced control methods and real-time performance optimization strategies, balancing vehicle safety and stability indicators, and combining collaborative design factors in software and hardware, we can provide more reliable, efficient, and safe solutions for future intelligent control of vehicle dynamics.

Author Contributions: Conceptualization, M.J., T.C. and J.L.; methodology, M.J., T.C. and J.L.; software, M.J. and T.C.; validation, M.J., T.C. and J.L.; writing—original draft preparation, M.J. and J.L.; writing—review and editing, M.J., T.C. and J.L.; visualization, M.J. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China (No. 52202472); the Opening Foundation of Key Laboratory of Advanced Manufacture Technology for Automobile Parts, Ministry of Education (No. 2023KLMT06); the Key Research Project Plan for Higher Education Institutions in Henan Province (No. 23B460002); and the Jiangsu Funding Program for Excellent postdoctoral Talent (No. 2022ZB660).

Data Availability Statement: The authors confirm that the data supporting the findings of this study are available within the paper.

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

References

1. Chen, T.; Cai, Y.; Chen, L.; Xu, X. Trajectory and velocity planning method of emergency rescue vehicle based on segmented three-dimensional quartic Bezier curve. *IEEE Trans. Intell. Transp. Syst.* **2023**, *24*, 3461–3475. [\[CrossRef\]](#)
2. Wang, D.; Yang, F.; Gan, L.; Li, Y. Fuzzy prediction of power lithiumion battery state of function based on the fuzzy c-means clustering algorithm. *World Electr. Veh. J.* **2019**, *10*, 1. [\[CrossRef\]](#)
3. Zhao, W.; Wang, L.; Mirjalili, S. Artificial hummingbird algorithm: A new bio-inspired optimizer with its engineering applications. *Comput. Methods Appl. Mech. Eng.* **2022**, *388*, 114194. [\[CrossRef\]](#)
4. Zhang, L.; Zhao, Z.; Chai, J.; Kan, Z. Risk Identification and Analysis for PPP Projects of Electric Vehicle Charging Infrastructure Based on 2-Tuple and the DEMATEL Model. *World Electr. Veh. J.* **2019**, *10*, 4. [\[CrossRef\]](#)
5. Iora, P.; Tribioli, L. Effect of ambient temperature on electric vehicles' energy consumption and range: Model definition and sensitivity analysis based on nissan leaf data. *World Electr. Veh. J.* **2019**, *10*, 2. [\[CrossRef\]](#)
6. Chen, T.; Chen, L.; Xu, X.; Cai, Y.; Jiang, H.; Sun, X. Passive fault-tolerant path following control of autonomous distributed drive electric vehicle considering steering system fault. *Mech. Syst. Signal Process.* **2019**, *123*, 298–315. [\[CrossRef\]](#)
7. Khunkitti, S.; Siritaratiwat, A.; Premrudeepreechacharn, S. A Many-Objective Marine Predators Algorithm for Solving Many-Objective Optimal Power Flow Problem. *Appl. Sci.* **2022**, *12*, 11829. [\[CrossRef\]](#)
8. Ezugwu, A.E.; Agushaka, J.O.; Abualigah, L.; Mirjalili, S.; Gandomi, A.H. Prairie Dog Optimization Algorithm. *Neural Comput. Appl.* **2022**, *34*, 20017–20065. [\[CrossRef\]](#)

9. Jiang, Y.; Meng, H.; Chen, G.P.; Yang, C.N.; Xu, X.J.; Zhang, L.; Xu, H.J. Differential-steering based path tracking control and energy-saving torque distribution strategy of 6WID unmanned ground vehicle. *Energy* **2022**, *254*, 124209. [\[CrossRef\]](#)
10. He, S.; Xu, X.; Xie, J.; Wang, F.; Liu, Z. Adaptive control of dual-motor autonomous steering system for intelligent vehicles via Bi-LSTM and fuzzy methods. *Control. Eng. Pract.* **2023**, *130*, 105362. [\[CrossRef\]](#)
11. Li, Z.; Zhao, H.; Wang, Y.; Ren, Y.; Chen, Z.; Chen, C. Adaptive event-triggered control for almost sure stability for vehicle platooning under interference and stochastic attacks. *ISA Trans.* **2023**, *138*, 120–132. [\[CrossRef\]](#)
12. Wang, L.; Lan, J.; Li, M. PAFNet: Pillar Attention Fusion Network for Vehicle–Infrastructure Cooperative Target Detection Using LiDAR. *Symmetry* **2024**, *16*, 401. [\[CrossRef\]](#)
13. Khunkitti, S.; Premrudeepreechacharn, S.; Siritaratiwat, A. A two-archive Harris Hawk optimization for solving many-objective optimal power flow problems. *IEEE Access* **2023**, *11*, 134557–134574. [\[CrossRef\]](#)
14. Li, Z.; Wang, P.; Liu, H.; Hu, Y.; Chen, H. Coordinated longitudinal and lateral vehicle stability control based on the combined-slip tire model in the MPC framework. *Veh. Syst. Dyn.* **2021**, *161*, 107947.
15. Xu, X.; Jiang, X.; Xie, J.; Wang, F.; Li, M. Research on human driving characterised trajectory planning and trajectory tracking control based on a test track. *Int. J. Control. Autom. Syst.* **2023**, *21*, 1258–1272. [\[CrossRef\]](#)
16. Chen, T.; Cai, Y.; Chen, L.; Xu, X. Sideslip angle fusion estimation method of three-axis autonomous vehicle based on composite model and adaptive cubature Kalman filter. *IEEE Trans. Transp. Electr.* **2024**, *10*, 316–330. [\[CrossRef\]](#)
17. Xie, J.; Xu, X.; Wang, F.; Liu, Z.; Chen, L. Coordination control strategy for human-machine cooperative steering of intelligent vehicles: A reinforcement learning approach. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 21163–21177. [\[CrossRef\]](#)
18. Chen, T.; Chen, L.; Xu, X.; Cai, Y.; Jiang, H.; Sun, X. Estimation of longitudinal force and sideslip angle for intelligent four-wheel independent drive electric vehicles by observer iteration and information fusion. *Sensors* **2018**, *18*, 1268. [\[CrossRef\]](#)
19. Wang, R.R.; Zhang, H.; Wang, J.M.; Yan, F.J.; Chen, N. Robust lateral motion control of four-wheel independently actuated electric vehicles with tire force saturation consideration. *J. Frankl. Inst.* **2015**, *352*, 645–668. [\[CrossRef\]](#)
20. Wang, Z.J.; Zhou, X.Y.; Wang, J.M. Extremum-Seeking-Based Adaptive Model-Free Control and Its Application to Automated Vehicle Path Tracking. *IEEE-ASME Trans. Mechatron.* **2022**, *27*, 3874–3884. [\[CrossRef\]](#)
21. Liang, C.; Xu, X.; Wang, F.; Wang, S.; Zhou, Z. CAN-induced asynchronous random delays-considered mode transition system for DM-PHEV based on constrained output feedback robust control strategy. *IEEE Trans. Veh. Technol.* **2022**, *71*, 5995–6006. [\[CrossRef\]](#)
22. Xu, F.X.; Zhou, C.; Liu, X.H.; Wang, J. GRNN inverse system based decoupling control strategy for active front steering and hydro-pneumatic suspension systems of emergency rescue vehicle. *Mech. Syst. Signal Process.* **2022**, *167*, 108595. [\[CrossRef\]](#)
23. Chen, T.; Cai, Y.F.; Chen, L.; Xu, X.; Sun, X.Q. Trajectory tracking control of steer-by-wire autonomous ground vehicle considering the complete failure of vehicle steering motor. *Simul. Model. Pract. Theory* **2021**, *109*, 102235. [\[CrossRef\]](#)
24. Lee, D.; Yi, K.; Chang, S.; Lee, B.; Jang, B. Robust steering-assist torque control of electric-power-assisted-steering systems for target steering wheel torque tracking. *Mechatronics* **2018**, *49*, 157–167. [\[CrossRef\]](#)
25. Ma, X.; Guo, Y.; Chen, L. Active disturbance rejection control for electric power steering system with assist motor variable mode. *J. Frankl. Inst.* **2018**, *355*, 1139–1155. [\[CrossRef\]](#)
26. Montanino, M.; Punzo, V. On string stability of a mixed and heterogeneous traffic flow: A unifying modelling framework. *Transp. Res. Part B Methodol.* **2021**, *144*, 133–154. [\[CrossRef\]](#)
27. Jin, L.; Tian, D.; Liu, Y. Coordinated control of DDAS and ESC systems for electric vehicle driven by in-wheel motors. *J. Mech. Eng.* **2018**, *54*, 160–169. [\[CrossRef\]](#)
28. Chen, W.; Sun, X.; Wang, H. Extension coordinated control of automotive differential drive assisted steering system. *Sci. Sin. Tech.* **2017**, *47*, 324–335. [\[CrossRef\]](#)
29. Hu, C.; Qin, Y.; Cao, H.; Song, X.; Jiang, K.; Rath, J.J.; Wei, C. Lane keeping of autonomous vehicles based on differential steering with adaptive multivariable super-twisting control. *Mech. Syst. Signal Process.* **2019**, *125*, 330–346. [\[CrossRef\]](#)
30. Deng, H.; Zhao, Y.; Feng, S.; Wang, Q.; Zhang, C.; Lin, F. Torque vectoring algorithm based on mechanical elastic electric wheels with consideration of the stability and economy. *Energy* **2021**, *219*, 119643. [\[CrossRef\]](#)
31. Liu, Y.; Gao, H. Stability, scalability, speedability, and string stability of connected vehicle systems. *IEEE Trans. Syst. Man Cybern. Syst.* **2021**, *52*, 2819–2832. [\[CrossRef\]](#)
32. Wu, X.; Wang, Z.; Chen, S. Research on Mathematical Modeling of Critical Impact Force and Rollover Velocity of Coach Tripped Rollover Based on Numerical Analysis Method. *Symmetry* **2024**, *16*, 543. [\[CrossRef\]](#)
33. Küp, E.T.; Cebeci, S.; Bayram, B.; Aydın, G.; Bozkaya, B.; Akhavan-Tabatabaei, R. An Integrated Framework for Dynamic Vehicle Routing Problems with Pick-up and Delivery Time Windows and Shared Fleet Capacity Planning. *Symmetry* **2024**, *16*, 505. [\[CrossRef\]](#)

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.