BULLETIN OF THE POLISH ACADEMY OF SCIENCES
TECHNICAL SCIENCES, Vol. 71(2), 2023, Article number: e144624
DOI: 10.24425/bpasts.2023.144624

Obstacle avoidance method of autonomous vehicle based on fusion improved A*APF algorithm

Yubin QIAN*, Hongtao SUN, and Song FENG

School of Mechanical and Automotive Engineering, Shanghai University of Engineering Science, Shanghai, China

Abstract. This paper proposes an autonomous obstacle avoidance method combining improved A-star (A*) and improved artificial potential field (APF) to solve the planning and tracking problems of autonomous vehicles in a road environment. The A*APF algorithm performs path planning tasks, and based on the longitudinal braking distance model, a dynamically changing obstacle influence range is designed. When there is no obstacle affecting the controlled vehicle, the improved A* algorithm with angle constraint combined with steering cost can quickly generate the optimal route and reduce turning points. If the controlled vehicle enters the influence domain of the obstacle, the improved artificial potential field algorithm will generate lane-changing paths and optimize the local optimal locations based on simulated annealing. Pondering the influence of surrounding participants, the four-mode obstacle avoidance process is established, and the corresponding safe distance condition is analyzed. A particular index is introduced to comprehensively evaluate speed, risk warning, and safe distance factors, so the proposed method is designed based on the fuzzy control theory. In the tracking task, a model predictive controller in the light of the kinematics model is devised to make the longitudinal and lateral process of lane-changing meet comfort requirements, generating a feasible autonomous lane-change path. Finally, the simulation was performed in the Matlab/Simulink and Carsim combined environment. The proposed fusion path generation algorithm can overcome the shortcomings of the traditional single method and better adapt to the dynamic environment. The feasibility of the obstacle avoidance algorithm is verified in the three-lane simulation scenario to meet safety and comfort requirements.

Key words: obstacle avoidance; path planning; fuzzy control; tracking control.

1. INTRODUCTION

A previous study has shown that driver errors account for 90% of the causes of collisions [1]. An intelligent vehicle is a self-planning machine that works in a complex environment, reducing traffic accidents in the future transportation system. As a complex system that combines software and hardware, driverless cars require the coordination of multiple modules, such as onboard hardware, sensor integration, perception prediction, and control plans to ensure safe and reliable operation [2]. As for lane changes, decision, planning, and control are very important to solve the overtaking problem by considering the surrounding environment [3]. Various advanced driving assistance systems (ADAS) were devised to reduce the probability of crashes and enhance the safety performance of vehicles [4]. However, autonomous obstacle avoidance technology that can adapt to various environmental conditions still needs to be studied. Lane change is a crucial issue for the stability of traffic flow. Indiscreet operations (such as sudden insertion into adjacent lanes) can disrupt regular traffic and even create serious traffic crashes [5]. Obstacle avoidance is a complex movement process involving longitudinal and lateral speed changes and approaching other moving vehicles [6, 7].

In the existing research, fuzzy logic considers various aspects of overtaking and was applied to relevant research on decision-making [8]. In addition, a multiple-goal reinforcement learning framework was proposed to plan over-taking maneuvers [9]. A two-step algorithm was devised for keeping the intricacy degree of the partially observable Markov decision process at a lower level for real-time decision-making when moving [10]. Due to a large amount of calculation, these methods have specific difficulties when applied to vehicles. For the problem of path planning, work on autonomous vehicle path planning can be traced back to the 1980s, focusing on calculating the fast, ideal, and crash-free trajectory from the start station to the destination [11]. To plan the optimal moving path quickly and effectively, the primary goal is to ensure that the intelligent vehicle reaches its destination safely. The global planning task based on environmental perception and the local planning task using sensors are two significant parts of the planning task.

The A* algorithm was initially published in 1968 by Peter Hart, an extension of the Dijkstra algorithm [12]. In a static environment, the A* is the most effective straight search algorithm for resolving the shortest path problem. To ensure an algorithm suitable for curved road planning, Pang *et al*. proposed a novel method with equal-step sampling [13]. LIU proposed to combine Euclidean distance and point-to-line distance to reduce the number of search points [14]. By dynamically estimating heuristic weights, two-stage A* can regulate the optimal generation of the trajectory according to the complexity of the planned request [15]. However, a single A* algorithm is hard to apply in a dynamic environment or needs complex changes to achieve this goal, which reduced the fast response characteristics of intelligent vehicles. The artificial potential field (APF)

*e-mail: qianyb@sues.edu.cn

Manuscript submitted 2022-08-03, revised 2022-12-31, initially accepted for publication 2023-02-06, published in April 2023.

path planning algorithm is a method using virtual force proposed by Khatib [16], so the movement of the monitored object is controlled by the resultant force where the target point generates the attraction, and the obstacle generates the repulsion. Although the planned path is generally smooth and safe, this method has zero points of force field [17]. Due to the simplicity and elegance of this algorithm, the APF is currently still a standard method for solving local obstacle avoidance planning problems. PBPF is a pseudo-bacterial genetic algorithm (PBGA) and a fitness function based on the concept of a potential field to construct a viable path in a dynamic environment, improving the conventional method [18]. Successful lane change operations possess path generation and need to spread to path tracking tasks, ensuring the vehicle is as close as possible to the present trajectory and meets the dynamics [19]. Manuel applied the fusion algorithm to the path planning of the robot, using the A* algorithm to get the best path and the APF controller to achieve path tracking to avoid obstacles [20]. Chunyu Ju applied the fused A* and APF algorithms to path planning, combining the advantages of both algorithms to get a reasonable path [21]. Tracking methods usually contain fuzzy logic control [22], sliding mode control [23], and yaw stability control [24]. The MPC controller is an effective tool [25], which has better anti-interference performance and control performance and is more widely used [26].

To solve the path planning problem of vehicles under the dynamic environment in the process of avoiding obstacles, combining decision and tracking control modules, this paper proposes an autonomous obstacle avoidance method, the overall architecture of the proposed automated system is shown in Fig. 1.

The environment perception module uses multi-sensor or wireless communication to accurately transmit the required information. The method of decision is designed based on fuzzy control by introducing the comprehensive decision index to consider velocity, risk warning, and safety distance. The path planning task is based on the A*APF algorithm. First, we make improvements to the A* and APF algorithms and then apply the fusion of the two algorithms to the obstacle avoidance scenario. A new cost function is constructed and path turns are reduced when the controlled vehicle does not enter the influence range of the obstacle. The optimal path is quickly generated with the improved A* algorithm. When the obstacle obstructs the vehicle progress, the improved APF can generate the obstacle avoidance path and optimize the trap points in time by simulated annealing to prevent the algorithm from confusion. Furthermore, a multi-constraint MPC controller is constructed based on the kinematics model to guarantee the desired path has the feasibility and stability requirements, and the vehicle state parameters are fed to the algorithm, this process is repeated at a fixed sample time and updated.

The remainder of the paper is as follows. Section 2 establishes the lane-changing path generation. Section 3 designs the tracking control based on MPC. Section 4 carries on the simulation to verify the feasibility of the algorithm, and Section 5 draws the corresponding conclusions.

2. PATH PLANNING METHOD BASED ON FUSION IMPROVED A*APF ALGORITHM

2.1. Basic A* algorithm

The A* algorithm divides the feasible area by detecting environmental information to establish a square grid, finding the shortest path composed of coordinate points. Since the study of vehicle motion planning, this paper assumes that the vehicle can accurately perceive the information of the external environment through sensors, converting path planning into an issue of searching for barrier-free areas in a known environment model.

As the most effective heuristic search in a static environment, the A* algorithm integrates the superiorities of the Dijkstra algorithm and the breadth-first search (BFS), widely used in the global path planning of robots or intelligent vehicles [27]. The A* algorithm first creates an OPEN list and saves eight adjacent grids as the nodes to be calculated, which are also named the current nodes. Analyzing these nodes in the OPEN list through the cost function, we select the path point with the lowest mobile cost and delete it from the OPEN list. Meanwhile, a close list is created to store all path points and obstacle points. The picture shows the geometric implication of the cost function.

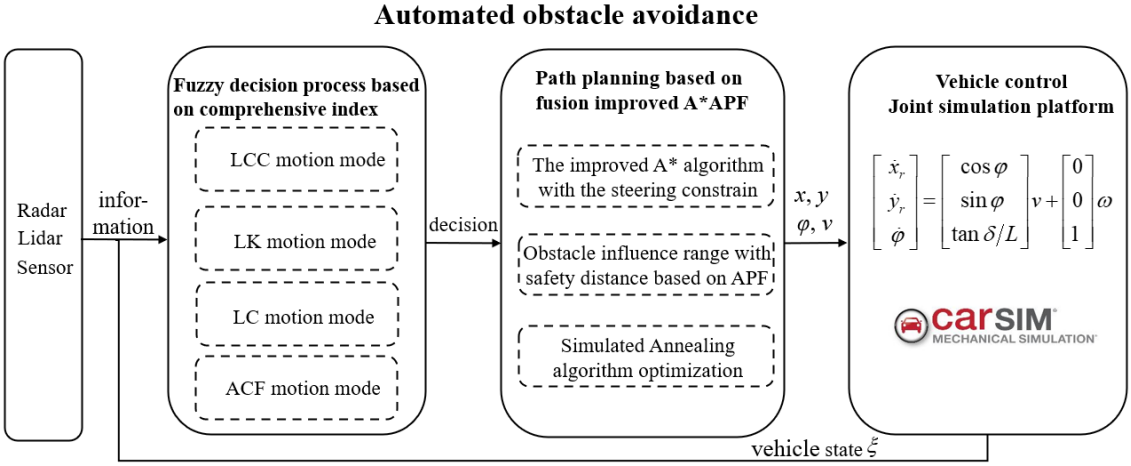

Fig. 1. Overall architectureof the automated system

As shown in Fig. 2, the point (x_s, y_s) in the left figure indicates the starting point and (x_m, y_m) indicates the current point to be calculated. In the right figure, the point (x_e, y_e) denotes the target point. The cost function is usually written as:

$$f(n) = g(n) + h(n), \tag{1}$$

where $g(n)$ denotes the actual movement cost from the starting position to the intermediate state n, and $h(n)$ denotes the estimated movement cost of the best path from the intermediate state n to the target position. In the conventional algorithm, $g(n)$ is defined by the Euclidean distance. Manhattan distance defines $h(n)$.

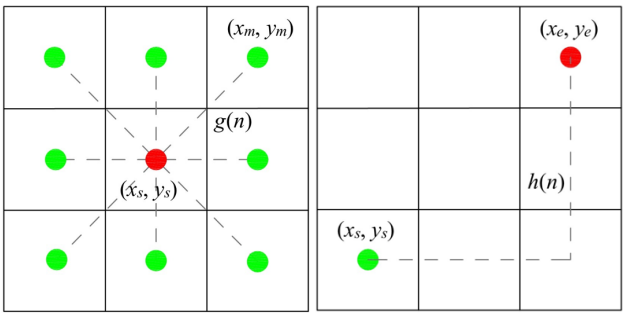

Fig. 2. The cost function

2.2. The improved A* algorithm with the steering constraint

However, as the shortest path planning, the traditional A* has many turns and preferentially follows a straight line between two points. Intelligent vehicles are constrained to drive in the lane without changing direction at will. Ensuring the nodes parallel to the lane have a smaller cost value, this paper defines the steering cost function p(n) considering the limit of steering angle, such as in Fig. 3. It introduces it into $f(n)$, reducing unnecessary steering:

$$p(n) = \begin{cases} \dfrac{\theta_g(n)}{\delta_{\max}} g(n) + \dfrac{\theta_h(n)}{\delta_{\max}} \dfrac{1}{h(n)} & \theta(n) \in \left(0, \dfrac{\pi}{2}\right), \\ 0 & \theta(n) = 0,\ \theta(n) = \infty, \end{cases} \tag{2}$$

where $\delta_{\max}$ is the maximum front-wheel steering angle, θ_g is the angle between the starting point and the current node, θ_h is the angle between the current point and the goal destination:

$$\theta_g(n) = \arctan\left(\left|\frac{y_m - y_s}{x_m - x_s}\right|\right), \tag{3}$$

$$\theta_h(n) = \arctan\left(\left|\frac{y_m - y_{\text{end}}}{x_m - x_{\text{end}}}\right|\right). \tag{4}$$

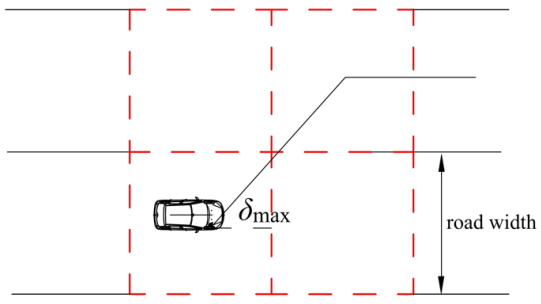

Fig. 3. The limit steering angle in A* algorithm

Thus, the novel cost function includes moving cost and steering cost is designed by:

$$f(n) = g(n) + h(n) + p(n). \tag{5}$$

In other words, vehicle motion constraints are considered to improve the A algorithm, which is suitable for the road structure environment, reducing path turning points, and realizing global path planning when no obstacles are affected.

2.3. Obstacle avoidance method based on improved APF algorithm

The improved A* algorithm can quickly plan an optimal path in a static environment. However, in a complex dynamic environment, since the local information changes in the case of the update of obstacle positions, the A* algorithm can no longer perform obstacle avoidance decisions well. A local path-planning algorithm to avoid dynamic obstacles in real time is necessary for an autonomous vehicle.

2.3.1. Basic artificial potential field algorithm

In previous research, Khatib first proposed the artificial potential field algorithm (APF), in which the spatial motion of an object was transformed into the force motion of a particle. In APF, the attractive potential field U_{att} acts on the global environment, and the farther the object is from the target point, the more attractive force it receives. In the obstacle avoidance control algorithm, to ensure the vehicle complies with the desired path from the start location to the destination, the attractive potential field can be written as:

$$U_{\text{att}} = \frac{1}{2} k_{\text{att}} \rho_s^2, \tag{6}$$

where k_{att} is the attractive constant, ρ_s is the distance between the controlled object and the destination. The attractive force acting on the controlled object is calculated as:

$$F_{\text{att}} = k_{\text{att}} \rho_g. \tag{7}$$

The repulsive potential field U_{rep} merely works in a limited range. If the object moves within this range, the repulsive force generated by the APF will prevent it from approaching the obstacle and form a collision-free motion trajectory:

$$U_{\text{rep}} = \begin{cases} \dfrac{1}{2} k_{\text{rep}} \left(\dfrac{1}{\rho_{\text{ob}}} - \dfrac{1}{\rho_0}\right)^2 & \rho_{\text{ob}} \le \rho_0, \\ 0 & \rho_{\text{ob}} > \rho_0, \end{cases} \tag{8}$$

where k_{rep} is the coefficient of the repulsive potential field, ρ_{ob} is the distance calculated from the object to the obstacle, ρ_0 is

the repulsive potential field range. In the U_{rep}, its radius defines the scope of the repulsive force. The controlled object is attracted towards the final destination by the force F_{att}. If the distance ρ_{ob} is minor than ρ_0, the repulsive force is used to avoid the collision. To sum up, since the principle of the APF algorithm is an analog to driver obstacle avoidance behavior, the APF is an ideal method to plan a safe local path. However, the traditional algorithm cannot be directly applied to traffic scenarios and needs to be improved.

2.3.2. Safety distance model

The traditional APF algorithm sets ρ_0 as a fixed value and ignores the relative motion between objects. This paper considers the safety distance between vehicles to define the potential field radius and improves the timeliness of the algorithm under the premise of satisfying safety. Emergency braking and steering are essential operations for autonomous vehicles to avoid collisions with the preceding vehicle. Besides, vehicles generally do not brake at maximum deceleration in the case of comfort considerations.

As shown in Fig. 4, when the ego vehicle changes lanes, it needs to maintain a certain initial distance R_{lcc} from the vehicles behind the adjacent lanes to prevent collisions during the entire process. The critical collision distance in this scenario is defined as R_{S1}, which can be expressed as:

$$R_{\text{lcc}} + L + x_e > R_{S1}\,, \tag{9}$$

where x_e presents the longitudinal displacement of the ego car (orange) during this stage, L denotes the length of the vehicle. Namely, to ensure a smooth process, the critical collision avoidance distance is defined as the assumption that the orange car starts to change lanes and the blue car can simultaneously decelerate to a stop to avoid a collision:

$$R_{S1} = x_e + \frac{v_b^2}{2a_b} + v_b t_0\,. \tag{10}$$

Subscript b means the blue vehicle, v_b presents the vehicle speed moving in the adjacent lane, a_b is the deceleration considering passenger comfort (generally limited to 0.5g), t_0 denotes the reaction time of drivers. The longitudinal distance x_e during the lane change:

$$x_e = v_e t + \int_0^t \int_0^t a_e(\tau)\,\mathrm{d}\tau\,\mathrm{d}t, \tag{11}$$

where v_e denotes the velocity of the ego car, t presents the current moment ($t <$ TTC), and the relationship between speed and TTC details in [28]. a_e denotes the acceleration of the ego vehicle. After formula transformation, the lane-changing check (LCC) condition is defined as:

$$R_{\text{lcc}} > \frac{v_b^2}{2a_b} + v_b t_0 - L. \tag{12}$$

In addition, if other vehicles are moving in adjacent lanes, the accommodating spacing formed in the middle becomes the judging factor on whether to perform a lane change. According to previous research, after changing into the adjacent lane, the position between the ego vehicle and the updated front obstacle is also a fundamental problem [29]. A specific initial distance in this scenario R_{lcc} is reserved to prevent collisions. The longitudinal distance of the integrated lane change process and braking distance avoid a forward collision when the controlled vehicle updates another lane. The R_{S2} can be expressed as:

$$R_{\text{S2}} = x_e + \frac{v_e^2}{2a_{\text{eb}}} + v_e t_0\,, \tag{13}$$

where a_{eb} is the braking deceleration considering passenger comfort in this scenario, when there is accommodating space in adjacent lanes, the condition is defined as:

$$R_{\text{lcc}} > R_{S2}\,. \tag{14}$$

If the autonomous lane change algorithm observes that the vehicle fails to satisfy the LCC, it means the vehicle fails to realize the lane change. The system will start the adaptive car-following (ACF) mode, adaptively follow the obstacle in front and maintain a certain safe distance. Similar to the longitudinal braking distance model in [30], R_b can be described as:

$$R_b = \frac{v_e^2}{2a_{\text{eb}}} + (v_e - v_{\text{ob}})\,T_t - \frac{v_{\text{ob}}^2}{2a_{\text{ob}}}\,, \tag{15}$$

where v_{ob} is the velocity of the front obstacle, T_t is the delay time of the braking device, a_{ob} is the braking deceleration of the obstacle. As presented in equation (15), this paper calculates the minimum longitudinal distance to set the dynamic field radius of the APF. The defined LCC is used as the condition of whether the actuator discussed in the next section can perform a lane change:

$$r = R_b\,. \tag{16}$$

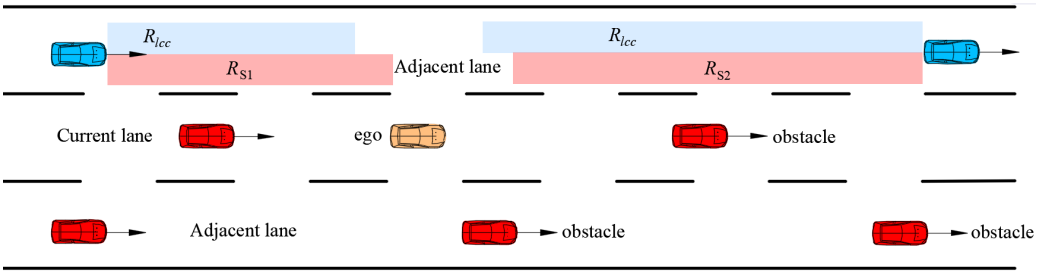

Fig. 4. Lane-changing condition

The real-time distance between the autonomous vehicle and the obstacle can be obtained by various sensors, which is counted as d. When $d < r$, the repulsive force experienced by the autonomous vehicle can be expressed as:

$$U_{\text{rep}} = \begin{cases} \frac{1}{2} k_{\text{rep}} \left(\frac{1}{d} - \frac{1}{r} \right)^2 & d \le r, \\ 0 & d > r, \end{cases} \tag{17}$$

$$F_{\text{rep}} = \begin{cases} \frac{k_{\text{rep}}}{d^2} \left(\frac{1}{r} - \frac{1}{d} \right) & d \le r, \\ 0 & d > r, \end{cases} \tag{18}$$

where r denotes the influence range of the obstacle.

2.3.3. Simulated annealing algorithm optimization

In the traditional APF, the algorithm falls into a local minimum position where the repulsive force is equal to the attractive force with an opposite direction. The simulated annealing (SA) algorithm was proposed by N. Metropolis. Because the SA algorithm can jump out of the trap probabilistically at the optimal position, it is a random optimization method [31].

When the vehicle is subjected to repulsive and gravitational forces of equal magnitude and the opposite direction in the potential field. Set the point as point x and set a relatively high initial temperature T. At this point, the algorithm will choose a random angle θ_{ran} forward to obtain a random point x_1. Then compare the total potential field (PF) of point x with that of point x_1. If the total PF of point x_1 is less than that of point x, the algorithm accepts x_1 as the next path point. Otherwise, the algorithm accepts this point as the next path point with probability P, with T decreasing in a certain way simultaneously. Finally, the iterative calculation helps the algorithm escape the optimal local value. In consideration of vehicle stability, the stability coefficient K is an essential factor affecting steering stability, which is divided into understeer, neutral steering, and excessive steering:

$$\delta = \frac{r(1 + Ku^2)}{u/l}, \tag{19}$$

where r presents the yaw angular velocity, u presents the velocity at the center of mass, l is the wheelbase. To make the algorithm satisfy the smooth steering and driving safety during local path planning, the random angle θ_{ran} needs to meet certain constraints:

$$\delta_{\min} \le \theta_{\text{ran}} \le \delta_{\max}. \tag{20}$$

A single A* algorithm fails to avoid dynamic obstacles in practical applications, which means path planning failure. Furthermore, if APF is used alone for local path planning, it fails to perceive global information, increasing calculated cost and causing low efficiency. To sum up, this paper uses the A*APF path planning algorithm based on combining the improved A* method and the improved APF method, which is presented in Fig. 5.

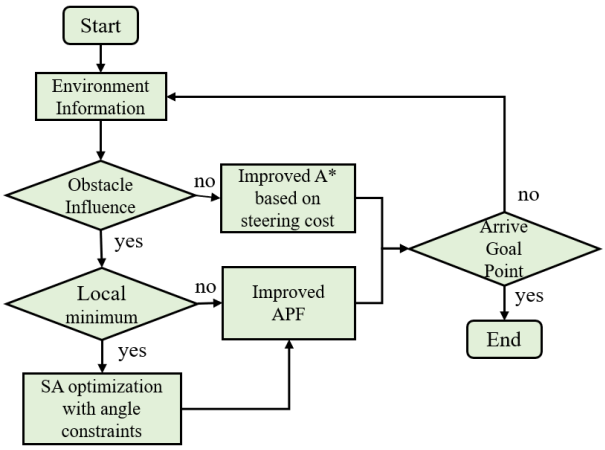

Fig. 5. Path generation process

3. FUZZY DECISION PROCESS BASED ON COMPREHENSIVE INDEX

There are common incentives for changing lanes, such as obstacles ahead or other vehicles inserted into the driving lane. Then, the driver will determine whether the lane change is feasible and safe, according to the traffic movement in the adjacent lanes, and make the final decision. This paper introduces a comprehensive decision index (CDI) to evaluate whether the driver can easily change the surrounding environment of the lane [32]. The first index is the speed advantage that drivers expect. When v_e in the original lane is lower than v_b in the adjacent lane, the driver is likely to have the motivation to change lanes. This paper quantifies the necessity of this working condition with speed:

$$P_v = \frac{v_e}{v_b}. \tag{21}$$

The movement of autonomous vehicles needs to ensure the safety of surrounding vehicles. The common metrics used in its study are movement gap, time to collision (TTC), and time gap (TG) [33]. If there are no other participants in the current lane obstructing the controlled vehicle, its speed depends on the driver's intention and is within the roadway limits:

$$v_e = v_d, \quad 0 < v < v_{\max}, \tag{22}$$

where v_d denotes the intention of the driver, $v_{\max}$ is the maximum allowable speed. The second index is the car-following advantage that drivers expect to obtain. The leading obstacle car in the current lane decelerates, or other vehicles are inserted in front. The spacing between the ego car and the leading participant is shortened, and the need to change lanes is generated to avoid collisions or obtain a more significant space margin. This paper introduces the risk warming (rw) coefficient to quantify this indicator [28], which can be expressed as:

$$P_{rw} = \frac{a}{\text{TWH}} + \frac{b}{\text{TTC}}, \tag{23}$$

where $a = 1$, $b = 5$, TWH denotes time headway, and the specific calculation method and value selection are detailed in [30].

Through the above analysis in Fig. 4, the third index is the consideration of safety distance:

$$P_d = \frac{D}{R_{\text{lcc}}}. \tag{24}$$

Since lane-changing safety is most closely related to the vehicle behind in the target lane, D denotes the actual distance of R_{lcc}. When an autonomous vehicle is incapable of avoiding obstacles through active steering, deceleration braking will be another necessary maneuver [30]. So the CF condition is defined as:

$$R_{cf} < R_b. \tag{25}$$

As mentioned in the previous description, lane-changing is a multi-stage process (it needs to consider whether there is suitable accommodation spacing in the adjacent lanes). The four-mode autonomous avoiding process can be expressed as:

Mode 1, lane-keeping (LK): when the controlled vehicle is obstacle-free in the front and rear directions of the current lane, if the ego vehicle has no intention or conditions of lane-changing, it will continue to drive along the current lane without steering.

Mode 2, lane-changing check (LCC): if there is a feasibility of changing lanes, perform LCC to check whether there is a safe accommodation space in the adjacent lanes.

Mode 3, lane-changing (LC): if the LCC is satisfied, the controlled vehicle generates a lane-changing trajectory under the control of the proposed algorithm. After successfully changing the lane, the controlled vehicle will perform appropriate safe driving behaviors (i.e. return to the first mode).

Mode 4, adaptive car-following (ACF): if there exists an obstacle in front of the controlled vehicle, the system performs a new LCC detection and selects the LC mode when the requirements are met. Otherwise, it enters the CF mode to maintain the car-following state and a suitable safe distance.

The fuzzy system realizes convenient and fast prediction or control of uncertain reasoning, including fuzzification, fuzzy rules, fuzzy reasoning, and anti-fuzzification. This process is applied to human decision-making models to solve driving behavior characteristics that cannot be accurately expressed by mathematical models [34].

When constructing the affiliation function, the trapezoidal function is chosen for both the maximum and minimum intervals, and the trigonometric function is chosen for the intermediate value interval. The fuzzy subset P_v about speed is defined as {slow, medium, fast} and the fuzzy subset P_{rw} is defined as {small, medium, large}, which indicates that the degree of danger is small, medium, and large. In addition, the fuzzy subset P_d is defined as {extremely dangerous, dangerous, medium, safe, extremely safe}. Finally, the fuzzy subset CI is defined as NL (negative large), NM (negative minor), Z (zero), PM (positive minor), PL (positive large), and the corresponding fuzzy rules are detailed in [32].

The three parameters designed are the inputs of the module. First, obtain environmental information based on various sensors and wireless communication technologies, and fuzz the parameters into qualitative expressions. Then, make inferences based on the designed fuzzy rules to obtain the possibility of changing lanes; finally, use the center of gravity method to resolve the fuzziness to obtain the quantitative. The output value is a comprehensive decision factor and Fig. 6 shows its membership function.

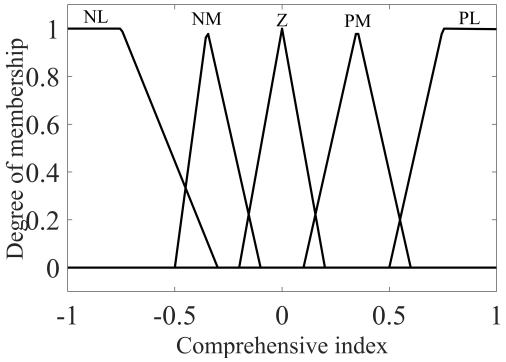

Fig. 6. Membership function

A positive value indicates that the driver is more likely to change lanes, while a negative value indicates a lower possibility. In addition, since this paper studies lane-changing behavior in a stable state, the autonomous system performs a steering operation when it is safe enough. The following methods are made for the decision-making after fuzzy control: {if CI = PL or PM, lane-changing; if CI = NL, NM, and Z, car-following}.

Human driving behavior is an essential factor affecting vehicle safety. In the autonomous vehicle control system, reflecting the driver's behavior and decision is vital to research. According to the research, driving behavior is separated into three levels: strategy, tactic, and management. [35]. An automated operating system based on human behavior must be constructed at a certain level. Figure 1 shows the framework structure of the autonomous obstacle avoidance algorithm, while Figure 7 shows the process of decision-making in the framework.

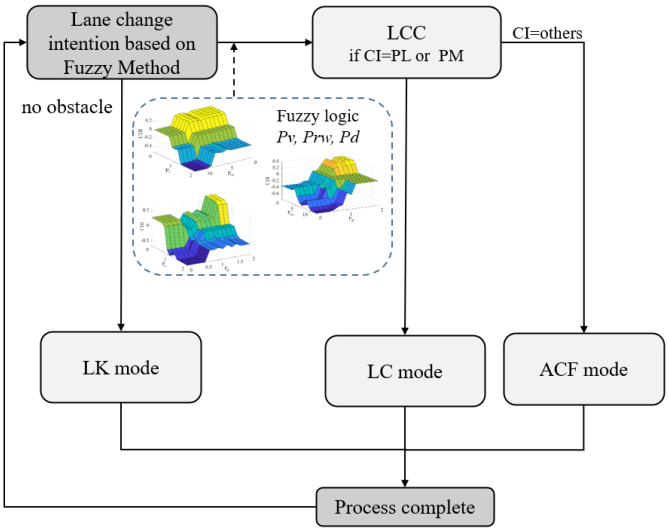

Fig. 7. The process of decision

4. TRACKING CONTROL BASED ON MODEL PREDICTIVE CONTROL

4.1. Vehicle kinematics model

Kinematics is the research about the movement rules from a geometric perspective, describing the position and speed changes of objects over time. Since the actual vehicle dynamics are complex, a simplified kinematics structure used in the path planning algorithm is feasible, ensuring the desired path has certain kinematic constraints. In addition, the dynamic factors of vehicle stability control are generally not considered in the case of low speed and good road conditions. So, the tracking controller based on the kinematic model has reliable control performance. The vehicle kinematics model that satisfies Ackerman steering is:

Figure 8 depicts the model fixed to the earth coordinate system *XOY*, where φ denotes the heading angle, δ denotes the front wheel steering angle, L is the wheelbase. In the center of the axle, (x_r, y_r) and (x_f, y_f) are the coordinates of the rear axle and the front axle, respectively, v is the vehicle speed of the rear axle. Then, the motion constraint equation is as follows:

$$\begin{aligned} \dot{x}_f \sin(\varphi+\delta) - \dot{y}_f \cos(\varphi+\delta) = 0, \\ \dot{x}_r \sin\varphi - \dot{y}_r \cos\varphi = 0. \end{aligned} \tag{26}$$

The yaw rate of the vehicle can be written as:

$$\omega = \frac{v}{L}\tan\delta. \tag{27}$$

The relationship between the wheel angle δ and the turning radius R is written as:

$$\tan\delta = \frac{L}{R}. \tag{28}$$

The two-degree-freedom vehicle kinematics model is formulated as follows:

$$\dot{\xi} = \begin{bmatrix} \dot{x}_r \\ \dot{y}_r \\ \dot{\varphi} \end{bmatrix} = \begin{bmatrix} \cos\varphi \\ \sin\varphi \\ \tan\delta/L \end{bmatrix} v. \tag{29}$$

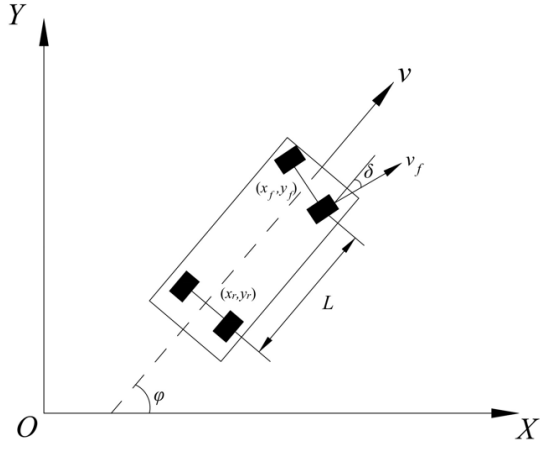

Fig. 8. Vehicle kinematics model

4.2. Tracking controller design

The system state ξ can be defined as $[x, y, \varphi]$, thus the discrete linear system is presented explicitly as:

$$\xi(k+1) = \begin{bmatrix} x_r(s+1) \\ y_r(s+1) \\ \varphi(s+1) \end{bmatrix} = \begin{bmatrix} x_r(s)+\Delta t \cdot v(s)\cos(\varphi(s)) \\ y_r(s)+\Delta t \cdot v(s)\sin(\varphi(s)) \\ \varphi(s)+\Delta t \cdot v(s)/L\tan(\varphi(s)) \end{bmatrix}, \tag{30}$$

where Δt is the discrete step, it was set to 0.05 in the experiment; $\xi(s+1)$ denotes the system state of step $s+1$. Restrictions of the variable in the MPC are designed in [36], containing vehicle speed and front-wheel steering angle. So the equation should be further expressed as follows:

$$\chi(k+1) = \begin{bmatrix} x_r(s+1) \\ y_r(s+1) \\ \varphi(s+1) \\ v(s+1) \\ \delta(s+1) \end{bmatrix} = \begin{bmatrix} x_r(s)+\Delta t \cdot v(s)\cos(\varphi(s)) \\ y_r(s)+\Delta t \cdot v(s)\sin(\varphi(s)) \\ \varphi(s)+\Delta t \cdot v(s)/L\tan(\varphi(s)) \\ v(s)+\Delta v(s+1) \\ \delta(s)+\Delta\delta(s+1) \end{bmatrix}, \tag{31}$$

where $\Delta u = [\Delta v(s), \Delta\delta(s)]^{\mathrm{T}}$, the increment of the control variable, is vehicle speed and wheel angle at s-th step, respectively. In the path tracking control of autonomous vehicles, this paper uses acceleration and yaw rate as the control variables, and the kinematics model can be converted as follows:

$$\begin{bmatrix} \dot{x}_r \\ \dot{y}_r \\ \dot{\varphi} \end{bmatrix} = \begin{bmatrix} \cos\varphi \\ \sin\varphi \\ \tan\delta/L \end{bmatrix} v + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \omega. \tag{32}$$

In terms of the longitudinal and lateral control of the targeted car in the autonomous lane-changing framework, an objective function shows the tracking manifestation of desired trajectory and velocity:

$$\begin{aligned} \min J(s) = &\sum_{i=1}^{Np} \|e_x(i) - S\|_{W_X}^2 + \sum_{i=1}^{Np} \|e_y(i)\|_{W_Y}^2 \\ &+ \sum_{i=1}^{Np} \|e_v(i)\|_R^2 + \sum_{i=0}^{Nc-1} \|\Delta u(s+i,s)\|_P^2, \end{aligned} \tag{33}$$

where N_p is the prediction scope, N_c is the control scope and $Np<Nc$, $e_x(i)$, $e_y(i)$ is the longitudinal and lateral error between the controlled object and the leading obstacle, respectively. $e_v(i)$ represents the error between the current speed and the desired speed. W_X, W_Y, R, Q, and P are the weighting values of the optimization problem. S represents the longitudinal distance that should be maintained between two vehicles. In addition, $t+s,t$ indicates the predicted value at s steps after t, and the control increment $\Delta u(t+s,t)$ needs to be minimized.

It should be noted that when the controlled object is in a single-car state, or it has developed into the leading object, the $e_x(i)$ and $e_y(i)$ should be zero, and the speed error indicated by e_v is transformed into the preset desired speed of the driver. When the system is in LC mode, the longitudinal and lateral errors (i.e. $e_x(i)$ and $e_y(i)$) need to be calculated. The required

speed depends on the front leading vehicle this time. The corresponding matrix and derivation process are detailed in [36]. Moreover, to guarantee the tracking control process stability, the establishment of constraint conditions needs to consider the control variable and its increment. The speed and wheel angle control of the autonomous vehicle are:

$$\begin{aligned} -0.2\,\mathrm{m/s} \le \mathrm{v} - \mathrm{v_d} \le 0.2\ \mathrm{m/s}, \\ -0.05\,\mathrm{m/s} \le \Delta \mathrm{v} \le 0.05\ \mathrm{m/s}, \\ -25^\circ \le \delta \le 25^\circ, \\ -0.47^\circ \le \Delta\delta \le 0.47^\circ, \end{aligned} \tag{34}$$

where $v_{\max}$ denotes the maximum speed allowed in the lane, v_d is the expected speed, and Δv is its increment per control cycle. Combining the cost function and restrictions, the MPC tracking controller must settle the below optimization complication in each cycle:

$$\begin{aligned} &\min J, \\ &s.t.\ \Delta U_{\min} \le \Delta U_t \le \Delta U_{\max}, \\ &U_{\min} \le A\Delta U_t + U_t \le U_{\max}, \end{aligned} \tag{35}$$

where $U_t = 1_{Nc} \otimes u(s-1), 1_{Nc}$ is a column vector with Nc rows, $u(s-1)$ denotes the actual control variable at the former moment, ΔU_t is the predicted input increments before $Nc-1$ steps, A is a special identity matrix, $U_{\min}$ and $U_{\max}$ is the minimum and maximum set of control variable in the control scope, respectively. After each control cycle, a series of input increments in the control scope is acquired:

$$\Delta U_t^* = [\Delta u_t^*\ \Delta u_{t+1}^*, \ldots, \Delta u_{t+Nc-1}^*]^T. \tag{36}$$

Then, MPC can repeat this process after conducting the later control cycle, achieving the trajectory tracking assignment of the vehicle in this loop:

$$u(t) = u(t-1) + \Delta u_t^*. \tag{37}$$

5. SIMULATION EXPERIMENT ANALYSIS

It is necessary to build a suitable simulation scenario to verify the lane-changing system. The effectiveness of the proposed lane-changing algorithm is simulated in a three-lane straight road, setting the width of each lane to 3.5 m. Active steering and the emergency braking are both effective operations for vehicles to take obstacle avoidance behaviors. In order to show the response ability of the proposed algorithm in a complex road environment, constructed scenario 1 describes that the intelligent vehicle first takes the brakes to adaptively follow the vehicle in front, and waits, completing the process of changing lanes and avoiding obstacles ahead.

The following Fig. 9 and Table 2 reveal the initial state of the ego vehicle and surrounding participants. The ego vehicle is driving in the middle lane, vehicle D is the obstacle ahead, the three cars A, B, and C are in the left lane, and E, F, and G are driving in the right lane. All simulation objects choose the C-class passenger vehicle that comes with Carsim, and some key vehicle parameters are shown in Table 1. The framework of co-simulation is shown in Fig. 10. Vehicles can receive available

Table 1
Simulation vehicle parameter setting

Parameters	Value
Overall mass	1274 kg
Height of center of mass	540 mm
Distance between center of mass and front axle	1016 mm
Wheelbase	2578 mm
Tire center height	316 mm
X-axis inertia	606 kg·m^2
Y-axis inertia	1523 kg·m^2
Z-axis inertia	1523 kg·m^2

Table 2
Simulation environment settings in scenario 1

Vehicle	X Position (m)	Y Position (m)	Speed (m/s)
Ego	0	0	20
A	20	3.5	20
B	80	3.5	20
C	120	3.5	20
D	50	0	10
E	0	–3.5	8
F	20	–3.5	8
G	40	–3.5	8

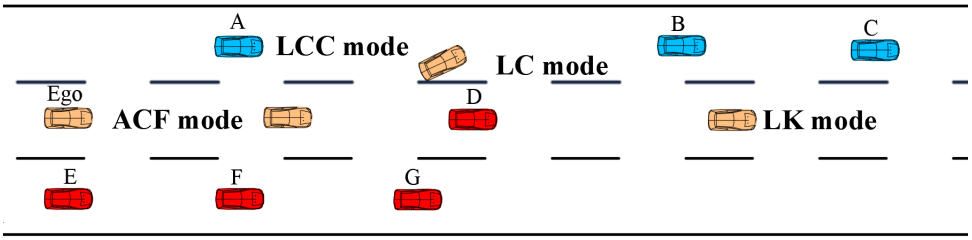

Fig. 9. Simulated environment in scenario 1

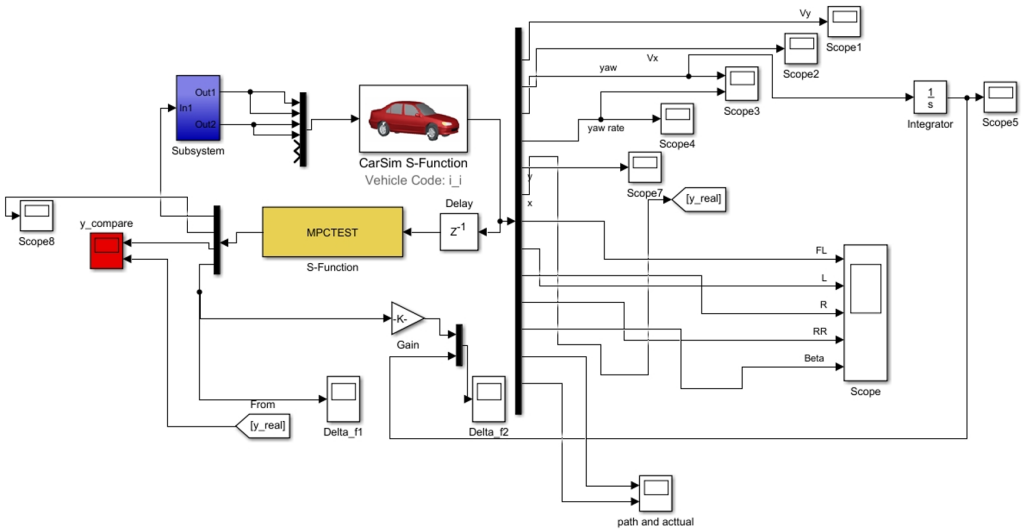

Fig. 10. Co-simulation structure

information through various sensors and wireless communications.

The simulation result is shown in Figs. 11–13. The planned lane change path is smooth in Fig. 11; the controller T can track the desired path and generate the actual trajectory that meets the stability requirements. Also, the steering angle and the lateral acceleration in Fig. 12 vary within a limited range. At this time, when the vehicle is turning left, the value is recorded as a positive value; otherwise, it is recorded as a negative value. There will be two changes in the state of motion, and the trends of these two changes are roughly symmetrical, and the directions are precisely opposite, showing the production of overtaking and returning to the original lane.

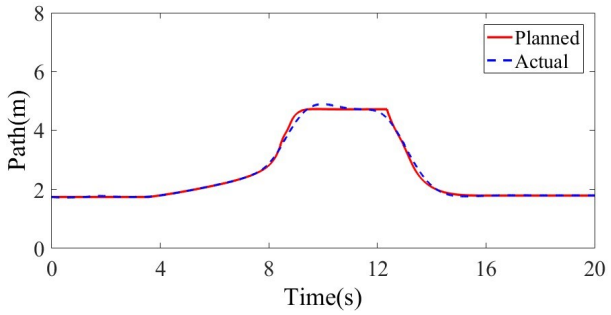

Fig. 11. Planned path and actual path in scenario 1

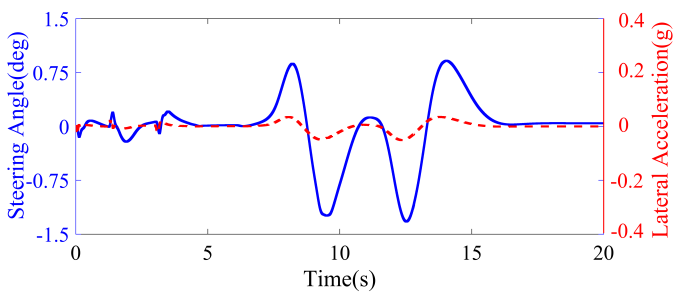

Fig. 12. Steering angle and lateral acceleration in scenario 1

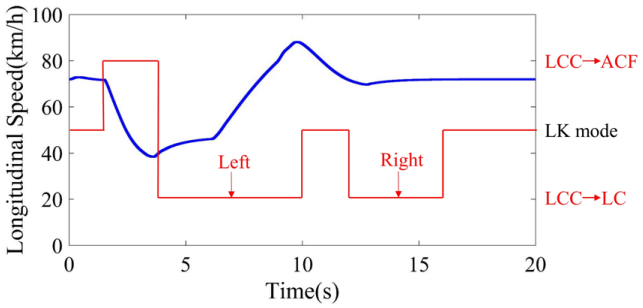

Fig. 13. Longitudinal speed and mode selection in scenario 1

The longitudinal speed change and the mode selection of the entire lane-changing process are shown in Fig. 13. After the longitudinal speed of the controlled vehicle has decreased for a period, it will accelerate when the lane change occurs and rise to the predetermined value until the lane-changing process is completed. In front of Ego in the middle lane, the speed of obstacle D is low. At the same time, car A driving in the left lane is closer to Ego, at this time the fuzzy decision system calculates that it cannot turn to change lanes immediately. It gets the instruction to execute ACF mode. Currently, Ego adaptively follows D. As the vehicle state changes, the decision system performs a lane change operation during the LCC check when the distance limit satisfies R_{lcc}, and the Ego vehicle will return to the original lane at a suitable position according to the proposed path generation algorithm. Since there are no other obstacles in the subsequent process, the Ego will maintain the driver's desired speed and perform LK driving. During this entire process, the fuzzy decision system can calculate a reasonable obstacle avoidance pattern based on the vehicle state. Thus, the algorithm can control the vehicle to complete the change to obstacle avoidance safely and stably.

The simulation process shows the operation of the four-mode autonomous obstacle avoidance system. LK is a mode that can keep driving in the current lane, without deviating from the

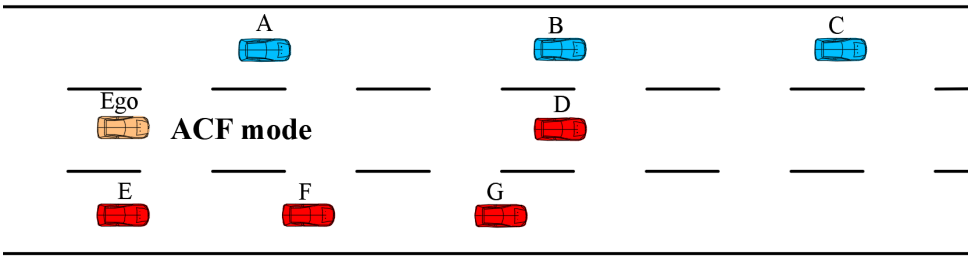

Fig. 14. Simulated vehicle parameters in scenario 2

course or other operations. As can be seen, LCC is an existential check process at the beginning of a lane change and when returning to the original lane. LC and ACF, respectively, indicate lane-changing trajectory and car-following trajectory under the guidance of A*APF. To sum up, the decision-making part based on the comprehensive index and fuzzy control is practical while generating a safe and comfortable trajectory.

The obstacle avoidance process of smart vehicles includes active steering and emergency braking. In the first structured road scenario, this paper verifies the applicability of the algorithm in dealing with complex situations. In addition, this paper also constructed another scenario to verify the algorithm ability to adaptively follow the obstacle ahead when the steering conditions are not met, as shown in Fig. 14. The environment settings at this time are shown in Table 3.

Table 3
Simulation environment settings in scenario 2

Vehicle	X Position (m)	Y Position (m)	Speed (m/s)
Ego	0	0	20
A	15	3.5	15
B	60	3.5	15
C	105	3.5	15
D	50	0	15
E	0	–3.5	10
F	20	–3.5	10
G	40	–3.5	10

This scenario mainly pays attention to the ACF function of the algorithm to verify the ability of the autonomous vehicle to maintain an adaptive following when it is not steerable. The path planned by the algorithm and the actual trajectory constrained by the controller are shown in Fig. 15. Except for slight changes in the starting position of the path, the planned path can maintain the ability of the vehicle to move in a straight line.

As shown in Fig. 16, the steering angle and lateral acceleration of the vehicle at this time are expressed on the same coordinate axis. The trajectory after being constrained by the MPC controller can ensure that the vehicle driving parameters change smoothly within a certain range, which meets the comfort requirements.

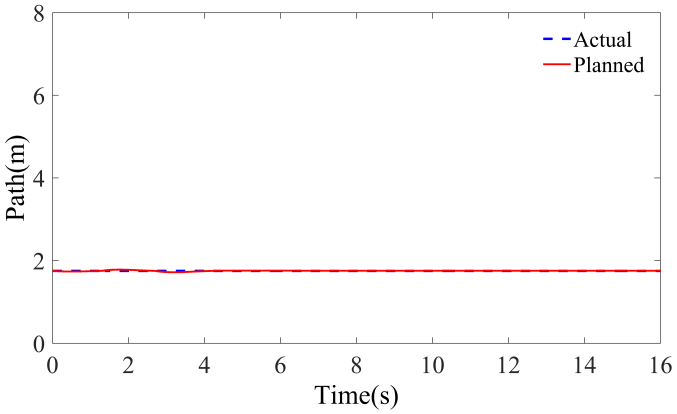

Fig. 15. Planned path and actual path in scenario 2

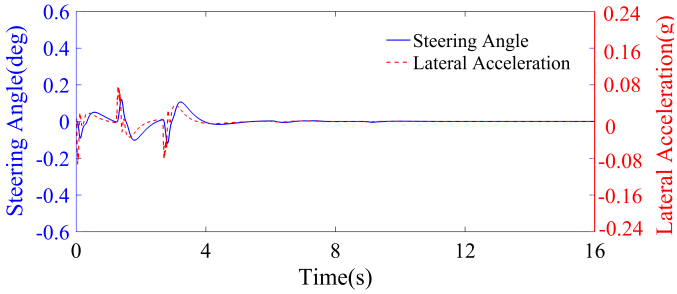

Fig. 16. Steering angle and lateral acceleration in scenario 2

In scenario 2, the vehicle longitudinal speed and mode switching are shown in Fig. 17. This scenario verifies the lane-keeping ability of the vehicle. Under the influence of obstacles constructed by the safe distance, the algorithm will switch to local obstacle avoidance planning. At this timeŁŹ the fuzzy de-

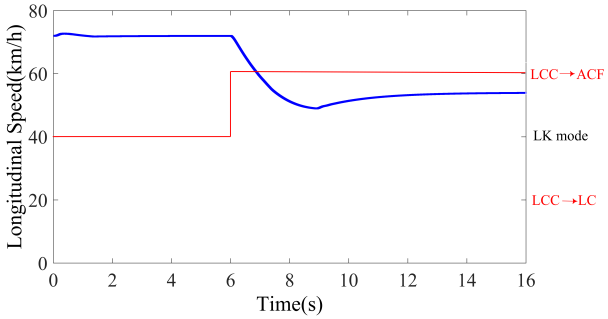

Fig. 17. Longitudinal speed and mode selection in scenario 2

cision system gets the obstacle avoidance mode according to the vehicle driving state and checks whether there is a possibility of steering. At this time the conditions are not met steering for a lane change, the obstacle avoidance mode is switched from LCC to ACF and continues to be maintained.

6. CONCLUSIONS

We propose an algorithm for automatic vehicle obstacle avoidance, including fuzzy decision-making, path planning, and tracking control. In detail, firstly, a four-mode avoiding system is set up, CI is introduced by integrating vehicle speed, risk warning, and safe distance factors, and fuzzy control theory is used to define the decision-making part. Secondly, this paper proposes a fusion A*APF algorithm. On the one hand, the optimal reference path based on the improved A* algorithm introduces a steering cost function, which makes the vehicle move in the direction parallel to the lane, reducing the turning of the path. On the other hand, the local obstacle avoidance principle is based on the improved APF algorithm, designing a safe path by introducing braking distance and resolving the local minimum problem by optimizing with the simulated annealing algorithm considering steering constraint. The effectiveness of the proposed algorithm is verified in the Matlab/Simulink and Carsim joint simulation. In the road structure environment, the obstacle avoidance algorithm can plan a safe path under the premise of considering the movement of the surrounding vehicles, while meeting the requirements of comfort and stability.

In the simulation, due to the error of the tracking controller, there are certain fluctuations when the system tracks the desired speed. In addition, this paper focuses on the obstacle avoidance process of the autonomous vehicle in the traffic environment, putting safety in a prior place, and the actual output is controlled by the tracking module. In future work, we will comprehensively evaluate the balance between comfort and safety at the planning level.

REFERENCES

[1] E. Fraedrich and B. Lenz, "Automated driving: Individual and societal aspects," *Transp. Res. Record*, vol. 2416, no. 1, pp. 64–72, 2014, doi: 10.3141/2416-08.

[2] J. Zhao, B. Liang, and Q. Chen, "The key technology toward the self-driving car," *Int. J. Intell. Syst.*, vol. 6, no. 1, pp. 2–20, 2018, doi: 10.1108/IJIUS-08-2017-0008.

[3] H. Chae and K. Yi, "Virtual Target-based Overtaking Decision, Motion Planning and Control of Autonomous Vehicles," *IEEE Access*, *IEEE Access*, vol. 8, pp. 51363–51376, 2020, doi: 10.1109/ACCESS.2020.2980391.

[4] D. González, J. Pérez, V. Milanés, and F. Nashashibi, "A Review of Motion Planning Techniques for Automated Vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 4, pp. 1135–1145, 2016, doi: 10.1109/TITS.2015.2498841.

[5] C. Rodemerk, S. Habenicht, A. Weitzel, H. Winner, and T. Schmitt, "Development of a general criticality criterion for the risk estimation of driving situations and its application to a maneuver-based lane change assistance system," in *IEEE Intelligent Vehicles Symposium*, 2012, pp. 264–269, doi: 10.1109/IVS.2012.6232129.

[6] L. Habel and M. Schreckenberg, "Asymmetric lane change rules for a microscopic highway traffic model," in *International Conference on Cellular Automata*, 2014, pp. 620–629, doi: 10.1007/978-3-319-11520-7_66.

[7] X. Sun *et al.*, "NFTSM control of direct yaw moment for autonomous electric vehicles with consideration of tire nonlinear mechanical properties," *Bull. Pol. Acad. Sci. Tech. Sci.*, vol. 69, no. 3, p. e137065, 2021, doi: 10.24425/bpasts.2021.137065.

[8] J.E. Naranjo, C. Gonzalez, R. Garcia, and T. de Pedro, "Lane-change fuzzy control in autonomous vehicles for the overtaking maneuver," *IEEE Trans. Intell. Transp. Syst.*, vol. 9, no. 3, pp. 438–450, 2008, doi: 10.1109/TITS.2008.922880.

[9] D.C.K. Ngai and N.H.C. Yung, "A multiple-goal reinforcement learning method for complex vehicle overtaking maneuvers," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 2, pp. 509–522, 2011, doi: 10.1109/TITS.2011.2106158.

[10] S. Ulbrich and M. Maurer, "Probabilistic online POMDP decision making for lane changes in fully automated driving," in *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*, 2013, pp. 2063–2067, doi: 10.1109/ITSC.2013.6728533.

[11] Z. Shiller, Y.-R. Gwo, "Dynamic motion planning of autonomous vehicles," *IEEE Trans. Robot. Autom.*, vol. 7, no. 2, pp. 241–249, 1991, doi: 10.1109/70.75906.

[12] D. Ferguson and A. Stentz, "Anytime, dynamic planning in high-dimensional search spaces," in *IEEE International Conference on Robotics and Automation*, 2007, pp. 1310–1315, doi: 10.1109/ROBOT.2007.363166.

[13] Y. Pang, Z. Song, X. Li, and J. Pan, "Truncation error analysis on reconstruction of signal from unsymmetrical local average sampling," *IEEE Trans. Cybern.*, vol. 45, no. 10, pp. 2100–2104, 2014, doi: 10.1109/TCYB.2014.2365513.

[14] H. Liu and Y. Zhang, "ASL-DWA: An Improved A-Star Algorithm for Indoor Cleaning Robots," *IEEE Access*, vol. 10, pp. 99498–99515, 2022, doi: 10.1109/ACCESS.2022.3206356.

[15] K. Zhang, Y. Yang, M. Fu, and M. Wang, "Two-phase A*: A real-time global motion planning method for non-holonomic unmanned ground vehicles," *Proc. Inst. Mech. Eng. Part D-J. Automob. Eng.*, vol. 235, no. 4, pp. 1007–1022, 2021, doi: 10.1177/0954407020948397.

[16] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," in *Autonomous Robot Vehicles*, Springer, New York, NY, 1986, pp. 396–404, doi: 10.1007/978-1-4613-8997-2_29.

[17] H. Xizhi, J. Zhihui, and X. Congcong, "Vehicle Path Planning Fusion Algorithm Based on Road Network," in *Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, 2020, vol. 1, pp. 98–102, doi: 10.1109/ ITNEC48623.2020.9084895.

[18] U. Orozco-Rosas, K. Picos, and O. Montiel, "Hybrid path planning algorithm based on membrane pseudo-bacterial potential field for autonomous mobile robots," *IEEE Access*, vol. 7, pp. 156787–156803, 2019, doi: 10.1109/ACCESS.2019.2949835.

[19] T. Shim, G. Adireddy, and H. Yuan, "Autonomous vehicle collision avoidance system using path planning and model-predictive-control-based active front steering and wheel torque control," *Proc. Inst. Mech. Eng. Part D–J. Automob. Eng.*, vol. 226, no. 6, pp. 767–778, 2012, doi: 10.1177/0954407011430275.

[20] N.L. Manuel, N. İnanç, and M. Erten, "Control of mobile robot formations using A-star algorithm and artificial potential fields," *J. Mechatron. Electr. Power Veh. Technol.*, vol. 12, no. 2, pp. 57–67, 2021, doi: 10.14203/j.mev.2021.v12.57-67.

[21] C. Ju, Q. Luo, and X. Yan, "Path planning using artificial potential field method and A-star fusion algorithm," in *2020 Global Reliability and Prognostics and Health Management (PHM-Shanghai)*, 2020, pp. 1–7, doi: 10.1109/PHM-Shanghai49105.2020.9280929.

[22] V. Mohan, A. Rani, V. Singh, and F. Systems, "Robust adaptive fuzzy controller applied to double inverted pendulum," *J. Intell. Fuzzy Syst.*, vol. 32, no. 5, pp. 3669–3687, 2017, doi: 10.3233/JIFS-169301.

[23] S.H. Tabatabaei Oreh, R. Kazemi, and S. Azadi, "A sliding-mode controller for directional control of articulated heavy vehicles," *Proc. Inst. Mech. Eng. Part D-J. Automob. Eng.*, vol. 228, no. 3, pp. 245–262, 2014, doi: 10.1177/0954407013503628.

[24] W. Zhang, Z. Wang, L. Drugge, and M. Nybacka, "Evaluating model predictive path following and yaw stability controllers for over-actuated autonomous electric vehicles," *IEEE Trans. Veh. Technol.*, vol. 69, no. 11, pp. 12807–12821, 2020, doi: 10.1109/TVT.2020.3030863.

[25] G.V. Raffo *et al.*, "A predictive controller for autonomous vehicle path tracking," *IEEE Trans. Intell. Transp.*, vol. 10, no. 1, pp. 92–102, 2009, doi: 10.1109/TITS.2008.2011697.

[26] T. Tuan *et al.*, "Disturbance-Kalman state for linear offset free MPC," *Arch. Control Sci.*, vol. 32, no. 1, pp. 153–173, 2022, doi: 10.24425/acs.2022.140869.

[27] C. Urmson, "Self-driving cars and the urban challenge," *IEEE Intell. Syst.*, vol. 23, no. 2, pp. 66–68, 2008, doi: 10.1109/MIS.2008.34.

[28] S. Moon and K. Yi, "Human driving data-based design of a vehicle adaptive cruise control algorithm," *Veh. Syst. Dyn.*, vol. 46, no. 8, pp. 661–690, 2008, doi: 10.1080/00423110701576130.

[29] C. Yuan *et al.*, "Research on active collision avoidance algorithm for intelligent vehicle based on improved artificial potential field model," *Int. J. Adv. Robot. Syst.*, vol. 17, no. 3, p. 1729881420911232, 2020, doi: 10.1177/1729881420911232.

[30] M. Hassanzadeh, M. Lidberg, M. Keshavarz, and L. Bjelkeflo, "Path and speed control of a heavy vehicle for collision avoidance manoeuvres," in *Intelligent Vehicles Symposium*, 2012, pp. 129–134, doi: 10.1109/IVS.2012.6232254.

[31] M.P. Vecchi, S. Kirkpatrick, "Global wiring by simulated annealing," *IEEE Trans. Comput-Aided Des. Integr. Circuits Syst.*, vol. 2, no. 4, pp. 215–222, 1983, doi: 10.1109/TCAD.1983.1270039.

[32] J. Ding, R. Dang, J. Wang, and K. Li, "Driver intention recognition method based on comprehensive lane-change environment assessment," in *Intelligent Vehicles Symposium Proceedings*, 2014, pp. 214–220, doi: 10.1109/IVS.2014.6856483.

[33] T. Kondoh *et al.*, "Identification of visual cues and quantification of drivers' perception of proximity risk to the lead vehicle in car-following situations," *J. Mech. Syst. Transp. Logist.*, vol. 1, no. 2, pp. 170–180, 2008, doi: 10.1299/jmtl.1.170.

[34] F. Pan *et al.*, "Lane-changing risk analysis in undersea tunnels based on fuzzy inference," *IEEE Access*, vol. 8, pp. 19512–19520, 2020, doi: 10.1109/ACCESS.2020.2968584.

[35] A. Kondyli and L. Elefteriadou, "Driver behavior at freeway-ramp merging areas based on instrumented vehicle observations," *Transp. Lett.*, vol. 4, no. 3, pp. 129–142, 2012, doi: 10.3328/TL.2012.04.03.129-141.

[36] Z. Huang *et al.*, "Path planning and cooperative control for automated vehicle platoon using hybrid automata," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 3, pp. 959–974, 2018, doi: 10.1109/TITS.2018.2841967.