

New RRT-Based Method for Vehicle Path Planning in Curve Scenarios Considering Path Oscillations

Rui Zhang, He Guo , Miguel Angel Sotelo , *Fellow, IEEE*, Haiping Du, *Senior Member, IEEE*, Andriukaitis Darius , and Zhixiong Li , *Senior Member, IEEE*

Abstract—This paper proposes a bidirectional RRT* algorithm integrated with dynamic artificial potential fields (APF) in the Frenet coordinate system to address obstacle avoidance challenges for autonomous vehicles in curved road scenarios. By decoupling longitudinal (s) and lateral (d) motion, the Frenet framework simplifies road constraints while reducing search space dimensionality by 70%. The APF mechanism dynamically balances obstacle repulsion, reference trajectory attraction, and goal-driven forces through an S-shaped repulsion function, suppressing path oscillations and improving computational efficiency. Theoretical analysis proves probabilistic completeness and reduced time complexity ($O(m^2 \log m)$ vs. $O(n^2 \log n)$). Experimental validation demonstrates adherence to Ackermann steering constraints ($\kappa_{\max} \leq 0.25 \text{ m}^{-1}$), achieving a 15.9% reduction in path length compared to RRT*, 17.4% faster computation than DWA, and 100% obstacle avoidance success. Real-world tests on U-turn and multi-curve roads validate trajectory stability with mean curvature 0.06 m^{-1} (45% improvement over RRT*) and sub-40 ms planning cycles, enabling real-time navigation in safety-critical environments.

Index Terms—Autonomous vehicles, path planning, Frenet coordinate system, RRT algorithm, artificial potential field.

I. INTRODUCTION

A. Background

WITH the rapid advancement of autonomous vehicle technologies, the domain of self-driving vehicles has garnered widespread societal interest [1]. Many experts predict that autonomous vehicles will be commercialized in the near future,

Received 31 December 2024; revised 10 March 2025 and 10 June 2025; accepted 15 June 2025. Date of publication 1 July 2025; date of current version 18 December 2025. This work was supported in part by the Science and Technology Plan Project of Tianjin under Grant 18YFZCGX00380 and in part by the Science Program of Tianjin Education Commission under Grant 2020KJ120. The review of this article was coordinated by Dr. Bo Yu. (Rui Zhang and He Guo are co-first authors.) (Corresponding author: Zhixiong Li.)

Rui Zhang is with the School of Automobile and Transportation, Tianjin University of Technology and Education, Tianjin 300222, China (e-mail: zhangrui318@163.com).

He Guo is with the School of Control Science and Engineering, Tiangong University, Tianjin 300387, China (e-mail: 936515285@qq.com).

Miguel Angel Sotelo is with the Department of Computer Engineering, University of Alcalá, 28801 Alcalá de Henares (Madrid), Spain (e-mail: miguel.sotelo@uah.es).

Haiping Du is with the School of Electrical, Computer and Telecommunications Engineering, University of Wollongong, Wollongong, NSW 2522, Australia (e-mail: haiping_du@uow.edu.au).

Andriukaitis Darius is with the Faculty of Electrical and Electronics Engineering, Kaunas University of Technology, LT-51368 Kaunas, Lithuania (e-mail: darius.andriukaitis@ktu.lt).

Zhixiong Li is with the Faculty of Mechanical Engineering, Opole University of Technology, 45-001 Opole, Poland (e-mail: z.li@po.edu.pl).

Digital Object Identifier 10.1109/TVT.2025.3585155

heralding a transformative impact on daily life [2]. Path planning is a critical component of autonomous vehicle systems [3], influencing not only the efficiency of vehicle motion control but also the safety of navigation. As a result, path planning methodologies have become a focal point of global academic research. Among the various challenges in path planning, the ability to effectively and reliably navigate around obstacles detected by vehicle sensors stands out as a crucial requirement [4]. Therefore, exploring the complexities of obstacle-avoidance path planning for autonomous vehicles is both timely and essential.

Path planning for autonomous vehicles typically bifurcates into global and local paradigms. Global path planning generates coarse routes based on static maps (e.g., road networks), while local path planning dynamically adjusts trajectories using real-time sensor data to handle obstacles and road geometry [5]. Although local planning is critical for real-time obstacle avoidance, its application in high-curvature scenarios (e.g., sharp turns, winding roads) remains underexplored. Specifically, existing approaches face two intertwined challenges:

- 1) *Curvature-Kinematic Conflict*: Avoiding obstacles while adhering to curvature constraints (to prevent rollover) and vehicle kinematic limits (to ensure trackability).
- 2) *High-Frequency Demands on Speed and Path Quality*: Real-time replanning ($\geq 20 \text{ Hz}$) necessitates rapid computation to ensure safety, while simultaneously requiring smooth and dynamically feasible trajectories.

This work focuses on local path planning, where the algorithm leverages a global reference line (e.g., road centerline in Frenet coordinates) to constrain exploration. By integrating bidirectional RRT* with artificial potential fields (APF), our method dynamically balances computational efficiency and trajectory smoothness, addressing the limitations of both classical and learning-based approaches in curved environments.

B. Related Works

Contemporary path planning techniques span a spectrum of methodologies, each with distinct trade-offs in autonomous driving applications. Traditional optimization-based methods such as genetic algorithms (GA) [6] and artificial potential fields (APF) [9] have been widely adopted for their simplicity. Tang J et al. propose an improved Artificial Electric Field Algorithm (I-AEFA) with parameter adaptation, reverse learning, and Cauchy mutation for enhanced optimization and 3D robot path planning [11]. Pan Q et al. proposes an AC-GRN with elastic deformation for multirobot hunting, enhancing

cooperation through environmental sensing and adaptive information weighting [10]. GA explores solution spaces through evolutionary operations but struggles with computational efficiency in dynamic environments [7], while APF generates smooth paths via attraction/repulsion fields but often stagnates in local minima near complex obstacles [8]. Model Predictive Control (MPC) [12] dynamically optimizes trajectories but faces challenges in real-time performance due to non-convex constraints inherent to high-curvature roads [13].

Learning-based approaches, including neural networks [14] and reinforcement learning (RL) [15], offer adaptive navigation capabilities. For example, RL frameworks with velocity obstacle-shaped rewards [16] enhance multi-robot coordination, and hierarchical perception strategies [17] improve scalability in decentralized systems. However, these methods demand extensive training data and lack interpretability in safety-critical scenarios, limiting their practicality for real-world autonomous driving.

Among sampling-based algorithms, the Rapidly-exploring Random Tree (RRT) and its variants (e.g., RRT*[18], RRT-connect [19]) excel in unstructured environments by probabilistically exploring configuration spaces. Recent advancements include hybrid strategies like Zhang et al.'s A*-RRT fusion [20], Fan et al. [22] further accelerated convergence through bidirectional APF-RRT* for UAVs, demonstrating dual-tree exploration advantages. Despite these innovations, RRT-based methods remain limited in high-curvature scenarios: unguided sampling introduces path oscillations during rapid replanning (≥ 20 Hz) [23], generated trajectories often violate kinematic constraints [24], and Cartesian coordinate-based planning fails to decouple longitudinal/lateral motions in curved lanes [27].

The Frenet coordinate system, rooted in a reference line (e.g., road centerline), addresses these challenges by decoupling motion into longitudinal (s) and lateral (d) components. Pioneered by Werling et al. [25] in DARPA challenges, Frenet-based planning enables efficient obstacle avoidance through curvature-aligned coordinates. Subsequent work, such as Fan et al.'s DP-QP framework [26], optimized paths and speeds using quintic polynomials but faltered in multi-obstacle scenarios. Ji et al. [27] integrated APF into the Frenet frame to mitigate lane boundary effects but suffered from rigid repulsion parameters. Notably, while Frenet coordinates have been extensively studied for trajectory optimization and MPC-based planning [29], [30], [31], their integration with sampling-based algorithms like RRT remains underexplored. Existing RRT implementations predominantly operate in Cartesian coordinates, missing the opportunity to leverage Frenet's inherent decoupling of road-aligned motion—a gap our work bridges by proposing the first bidirectional APF-RRT* framework fully embedded in the Frenet frame.

C. Motivations and Contributions

In high-curvature road environments, autonomous vehicle path planning faces challenges in ensuring safety, real-time performance, and kinematic feasibility, with traditional methods often suffering from path oscillations, low search efficiency,

and constraint violations. To address these issues, this paper proposes a novel framework — the first bidirectional artificial potential field-guided RRT* (Bidirectional APF-RRT*) algorithm embedded in the Frenet coordinate system.

The main contributions of this work are summarized as follows.

1) A constrained sampling strategy based on the Frenet coordinate system is proposed to optimize the exploration process of the RRT* algorithm. The natural decoupling between longitudinal (s-direction) and lateral (d-direction) motion in the Frenet coordinate system is utilized. The sampling space is strictly limited to the drivable area. Specifically, the lateral sampling range is constrained within ± 1.5 meters, which corresponds to a typical lane width. The longitudinal direction advances along the reference path. Compared with traditional Cartesian-based RRT methods, the dimensionality of the search space is reduced by approximately 70%. The convergence efficiency of the algorithm in narrow and curved lanes is significantly improved.

2) An improved bidirectional RRT* algorithm integrating an Artificial Potential Field (APF) mechanism is designed. This algorithm incorporates obstacle repulsion forces, dynamic repulsion forces for moving obstacles, and target attraction forces during path expansion, with the force intensities dynamically adjusted based on the distance to obstacles. This mechanism not only effectively suppresses path oscillation phenomena but also prevents the algorithm from falling into local optima, while reducing the blindness of path exploration, thereby further enhancing the algorithm's convergence speed and stability.

3) we propose an improved S-shaped attractive force function and curvature-adaptive sampling strategy specifically optimized for the Frenet coordinate system. This innovative combination enhances driving safety and passenger comfort through two key mechanisms: (1) The S-shaped function provides a smooth transition in the potential field, effectively suppressing oscillations near obstacles; (2) The adaptive sampling mechanism dynamically adjusts node density based on road curvature variations. Furthermore, we construct an attraction field based on the road centerline to ensure generated paths tightly adhere to reference trajectories, maintaining excellent path-following performance even in obstacle-free scenarios. Finally, we implement a comprehensive smoothing workflow integrating cubic B-spline interpolation and steering-angle-constrained pruning strategies, which significantly improves path smoothness and practical feasibility. Extensive simulations and real-vehicle experiments validate the algorithm's effectiveness, achieving 100% obstacle avoidance success rate in collision scenarios while meeting the <40 ms real-time planning cycle requirements for autonomous driving applications.

In the rest of this paper, Section II explains the Frenet Coordinate System for Sampling Space Reduction and conversation. Section III outlines the algorithmic principles and design used in this study. Section IV presents simulation experiments tailored to validate the proposed algorithm. Section V demonstrates the practical effectiveness of our approach through on-vehicle tests using the ROS platform. Section VI summarizes the paper and discusses future directions.

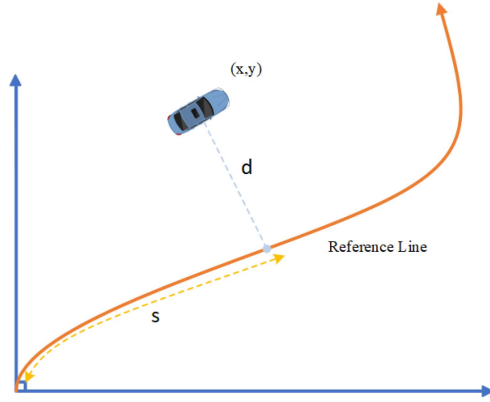


Fig. 1. The relationship between Cartesian coordinates and Frenet coordinates.

II. CONVERSION BETWEEN THE FRENETT COORDINATE SYSTEM AND THE CARTESIAN COORDINATE SYSTEM

A. Frenet Coordinate System for Sampling Space Reduction

In complex road environments, path planning methods based on the Cartesian coordinate system exhibit inherent limitations in encoding structured road features, such as lane boundaries and curvature. These limitations are reflected in inefficient exploration within the unbounded two-dimensional space, as well as an inability to guarantee that generated trajectories conform to road geometric constraints. To address this issue, the Frenet coordinate system is introduced by employing a reference line (e.g., the road centerline), through which the vehicle position is represented by two parameters with clear physical meanings: the longitudinal offset s , which denotes the arc length along the reference line, and the lateral offset d , defined as the signed perpendicular distance to the reference line, as illustrated in Fig. 1.

Existing studies have shown that the convergence speed of the RRT algorithm can be significantly improved by restricting the sampling domain [23]. For instance, Informed RRT reduces the sampling space to 50% through an elliptical sampling region [10]. However, additional geometric projection constraints are required in the traditional Cartesian coordinate system to achieve similar performance. In contrast, the Frenet coordinate system inherently provides structured sampling constraints, including lateral limits, longitudinal guidance, and obstacle projection, which enable the sampling region to be further reduced to less than 30%. As a result, the computational complexity is substantially decreased.

Compared to sampling strategies based on the traditional Cartesian coordinate system and improved methods such as Informed RRT*, the Frenet coordinate system is more suitable for structured road environments. As shown in Fig. 2, this approach significantly reduces the dimensionality of the sampling space through three coordinated mechanisms:

- 1) The lateral sampling is constrained within the range $d \in [-d_{\max}, d_{\max}]$ (e.g., ± 1.5 m, corresponding to standard lane widths), thereby eliminating geometrically infeasible regions.

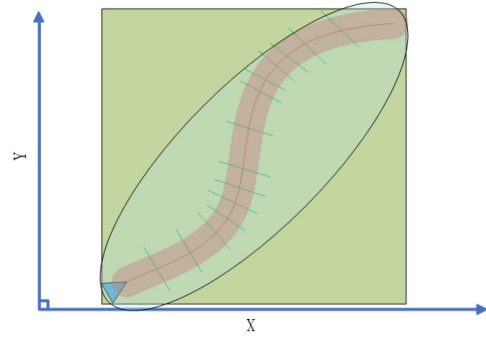


Fig. 2. Comparison of sampling regions between the two approaches.

- 2) Longitudinal sampling progresses monotonically along the road topology, ensuring that s increases continuously along the reference line.
- 3) An obstacle projection mechanism is employed to simplify collision detection.

Furthermore, this method does not require additional computation of road boundary constraints, which must be handled in other coordinate systems and consequently increases computational overhead.

Fig. 2 presents a comparison of the sampling regions: the green region represents the sampling area of the traditional Cartesian RRT (accounting for 100% of the original space), the blue region corresponds to the elliptical sampling area of Informed RRT* (approximately 60%), and the red region indicates the sampling area of the proposed method (less than 30%).

In terms of time complexity, the RRT* algorithm consists of four main steps: sampling, searching (for the nearest neighbor), extending (generating new nodes), and rewiring (optimizing paths via neighbor connections). The search step typically relies on spatial data structures (e.g., k-d trees), with a time complexity of $O(\log n)$. The rewiring step may require checking all existing nodes in each iteration to optimize neighbor paths, resulting in a worst-case complexity of $O(n)$. Collision detection can be considered $O(1)$ under efficient environmental representations (e.g., distance fields). Thus, for n nodes, the per-iteration complexity is $O(\log n + n) = O(n)$ (dominated by rewiring). The overall time complexity is $O(n^2)$; due to the impact of the search data structure, it is commonly expressed as $O(n^2 \log n)$ in practice.

$$T_{\text{RRT}^*} = O(n^2 \cdot (\log n)) \quad (1)$$

The Frenet coordinate-based RRT* algorithm significantly improves efficiency by constraining the sampling domain (e.g., parameterized along a reference path), which reduces inefficient exploration regions and concentrates node distribution around feasible paths. Let m denote the number of effective sampling nodes, where $m \ll n$ (e.g., m can decrease to $O(\sqrt{n})$ or lower in complex environments). This reduced sampling space decreases per-search time from $O(\log n)$ to $O(\log m)$ due to smaller data structures, while simultaneously reducing rewiring demands through fewer invalid nodes and smaller neighbor sets. Consequently, the overall time complexity drops from $O(n^2 \log n)$ for classical RRT* to $O(m^2 \log m)$. In highly complex environments where obstacle count $k \rightarrow n$, the classical method

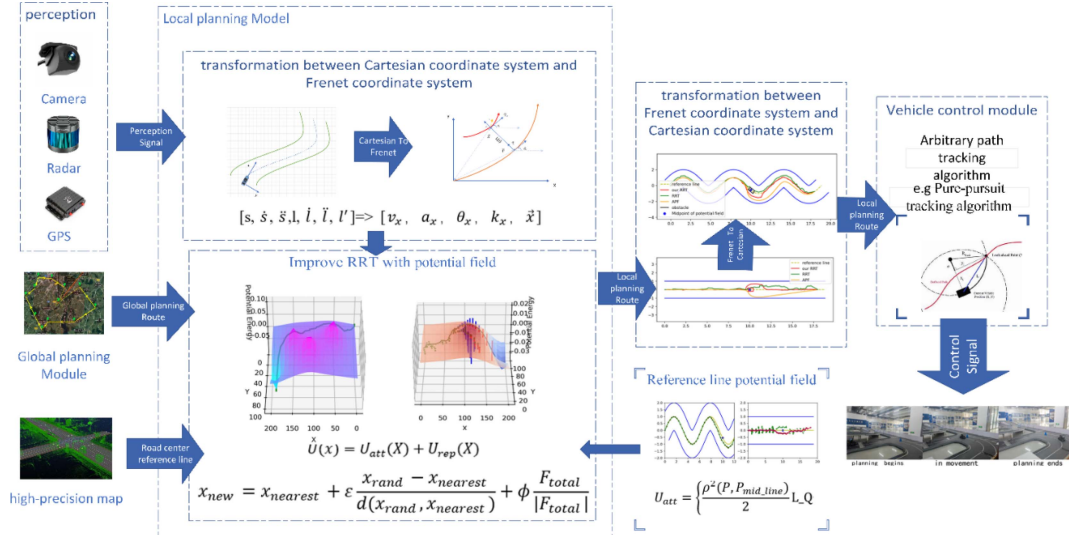


Fig. 4. Framework of the proposed path planning method about improved RRT.

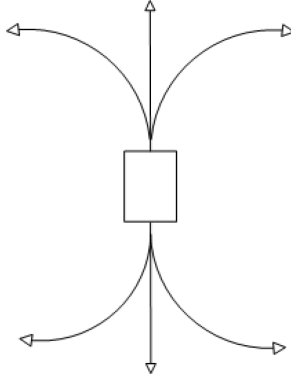


Fig. 5. Ackermann car restraint method.

The kinematic constraints for an Ackermann-steered vehicle are given in (6). In this context, $v_{\max}$ denotes the vehicle's maximum velocity, $\varphi_{\max}$ represents the maximum steering angle of the front wheels, $k_{\max}$ signifies the maximum path curvature, and $R_{\min}$ is the minimum turning radius of the vehicle.

$$\begin{cases} v \leq v_{\max} \\ \varphi \leq |\varphi_{\max}| \\ k \leq k_{\max} = \frac{\tan \varphi_{\max}}{b} = \frac{1}{R_{\min}} \end{cases} \quad (5)$$

B. Enhanced RRT* Algorithm

1) *APF-RRT* Algorithm Incorporating Artificial Potential Field Concepts:* Traditional Rapidly-exploring Random Tree (RRT) algorithms randomly determine the expansion direction during path searching, which can result in extended search times and redundant exploration points. In this study, we draw inspiration from the concepts of attractive and repulsive forces in artificial potential fields (APF). By applying resultant forces to newly generated nodes, we accelerate the convergence speed of

the algorithm. The principle of the APF algorithm is as follows:

$$F_{att}(X) = -\nabla U_{att}(X) = -k_{att} \cdot (X - X_0)^2 \quad (6)$$

$$U_{att}(X) = \frac{1}{2} k_{att} (X - X_0)^2 \quad (7)$$

Herein, F_{att} represents the attractive force vector acting on the robot, which is derived from the attractive potential function U_{att} , as shown in (6). This potential function decreases as the distance between the robot and the target decreases. Consequently, the robot moves in the direction of the steepest potential energy descent, effectively being guided towards the target position.

$$U_{rep}(X) = \begin{cases} \frac{1}{2} k_{rep} \left(\frac{1}{\rho(X, X_0)} - \frac{1}{\rho_0} \right)^2, & \rho(X, X_0) \leq \rho_0, \\ 0, & \rho(X, X_0) > \rho_0, \end{cases} \quad (8)$$

Additionally, ρ_0 is defined as the effective influence radius of the obstacle, a constant value beyond which the repulsive effect diminishes to zero. The function describes the shortest Euclidean distance from the robot to the nearest point on the obstacle, serving as a scalar quantity rather than a vector.

The repulsive force acting on the robot due to an obstacle can be derived from the negative gradient of the repulsive potential field, as shown in (9).

In the context of the artificial potential field (APF), k_{rep} represents the repulsive coefficient that governs the intensity of the repulsion field. Let x denote the vector coordinates of the robot's position, and X_0 represent the vector coordinates of the obstacle's position. Additionally, ρ_0 is defined as the effective influence radius of the obstacle, a constant value beyond which the repulsive effect diminishes to zero. The function $\rho(X, X_0)$ describes the shortest Euclidean distance from the robot to the nearest point on the obstacle, serving as a scalar quantity rather than a vector. The repulsive force acting on the robot due to an obstacle can be derived from the negative gradient of the

repulsive potential field, as shown in (10):

$$F_{rep}(X) = \begin{cases} k_{rep} \left(\frac{1}{\rho(X, X_0)} - \frac{1}{\rho_0} \right), & \rho(X, X_0) \leq \rho_0, \\ 0, & \rho(X, X_0) > \rho_0. \end{cases} \quad (9)$$

$$U(X) = U_{att}(X) + U_{rep}(X) \quad (10)$$

$$F(X) = F_{att}(X) + \sum_{i=1}^n F_{rep}(X) \quad (11)$$

In conventional RRT algorithms, the increment of node expansion is determined solely by the direction from the nearest node x_{near} to the random sample point x_{rand} a fixed step size. However, this approach can lead to inefficient exploration and suboptimal paths.

In this study, we propose an enhanced method where the growth increment of each new node is co-determined by both the attractive and repulsive forces perceived by the robot at the position $x_{nearest}$, as well as the direction relative to the random point. This integration allows for more intelligent path planning that dynamically adjusts based on environmental constraints and target proximity.

The formula for generating the new node x_{new} is given by (12):

$$x_{new} = x_{nearest} + \varepsilon \frac{x_{rand} - x_{nearest}}{d(x_{rand}, x_{nearest})} + \phi \frac{F_{total}}{|F_{total}|} \quad (12)$$

In this context, F_{total} denotes the aggregate of attractive and repulsive forces experienced by the robot at text $x_{nearest}$; ε signifies the stochastic component; and represents the potential field influence. It is crucial to underscore that for the algorithm to achieve probabilistic completeness—thereby ensuring that the random tree can attain any collision-free configuration within the space—the relationship between φ and ε adhere to $\varphi < \varepsilon$. Contrary to conventional Rapidly-exploring Random Trees (RRTs), which expand in fixed increments, our RRT variant's expansion distance is dynamically determined by integrating both the stochastic and potential field components.

Upon nearing obstacles, the orientation of the potential field component is predominantly dictated by repulsive forces, resulting in a diminished step size toward the obstacle. This mechanism enables the random tree to adeptly traverse through confined passages. Conversely, when nodes are situated further from obstacles, the directionality of the potential field component is chiefly governed by attractive forces, thereby encouraging an increased step size towards the target. Consequently, this approach facilitates a more rapid progression towards the objective while maintaining safety margins from obstacles. This adaptive strategy not only enhances the efficiency of the exploration process but also ensures robust performance in complex environments with varying degrees of spatial constraints.

In the context of artificial potential field (APF) methods, the gravitational force is designed to facilitate rapid convergence towards the target when the robot is at a considerable distance. As the robot approaches the target, it becomes necessary to increase the repulsive forces to prevent overshooting. In contrast, in traditional Rapidly-exploring Random Tree (RRT) algorithms, nodes near the final destination are directly connected to the

latest node once a feasible path to the goal is identified. Consequently, the influence of potential field forces on nodes within an RRT framework differs from that observed in conventional RRT implementations.

The gravitational and repulsive functions within the APF framework are mathematically defined by (14) and (15), respectively:

$$F_{att} = F_{att}^* \frac{x_{goal} - x_{nearest}}{d(x_{goal}, x_{nearest})} \quad (13)$$

$$F_{rep} = \begin{cases} l \frac{F_{rep}^*}{1 + e^{(2|d_{min}|/d_{rep}^* - 1)k}} \frac{d_{min}}{|d_{min}|} & |d_{min}| \leq d_{rep}^* \\ 0 & |d_{min}| > d_{rep}^* \end{cases} \quad (14)$$

Equation (14) defines the repulsive force F_{att}^* which is a constant representing the gravitational pull towards the target. The term $x_{goal} - x_{nearest}$ indicates the direction of growth from the nearest point on the RRT tree towards the goal. As the distance between the goal and the growth point increases, the influence of this attractive force decreases, ensuring that the robot does not move too aggressively when far from the target. Conversely, as the robot approaches closer to the goal, the attractive force gradually increases, thereby resolving the issue of unreachable targets often encountered in traditional APF methods.

Equation (14) defines the repulsive force F_{rep}^* where d_{rep}^* is a threshold distance, and d_{min} represents the minimum distance between the obstacle and the growth point. The term $\frac{d_{min}}{|d_{min}|}$ specifies the direction of the repulsive force away from the obstacle. The exponential function $e^{(2|d_{min}|/d_{rep}^* - 1)k}$ acts as an S-shaped curve, rapidly increasing as d_{min} approaches the obstacle, thereby limiting the maximum repulsive force to the predefined constant F_{rep}^* . When d_{min} approaches the threshold d_{rep}^* , the repulsive force diminishes towards zero, ensuring smooth transitions and preventing abrupt changes in the robot's trajectory.

Additionally, to enhance the algorithm's responsiveness to dynamic obstacles, a repulsive velocity factor is introduced in this study, as described by the following equation. The designed varying potential field force is expressed as equation:

$$F_{vel} = \begin{cases} k_t |v_r - v_0| \cos \alpha, & \alpha \in [-\frac{\pi}{2}, \frac{\pi}{2}] \text{ } |d_{min}| \leq d_{rep}^* \\ 0, & |d_{min}| > d_{rep}^* \end{cases} \quad (15)$$

Among them, v_r represents the speed of the obstacle, v_0 represents the speed of the self-vehicle, and α represents the angle between the speed of the obstacle and the speed of the self-vehicle.

By incorporating these adaptive attractive and repulsive forces into the RRT framework, the algorithm not only overcomes the problem of unreachable goals but also ensures the generation of smooth paths without sudden discontinuities. This approach significantly enhances the navigational capabilities of robots in complex environments, providing both safety and efficiency in path planning.

Fig. 6 illustrates the principle and methodology of integrating potential fields into the path planning process for newly generated nodes within a random tree structure. In this enhanced

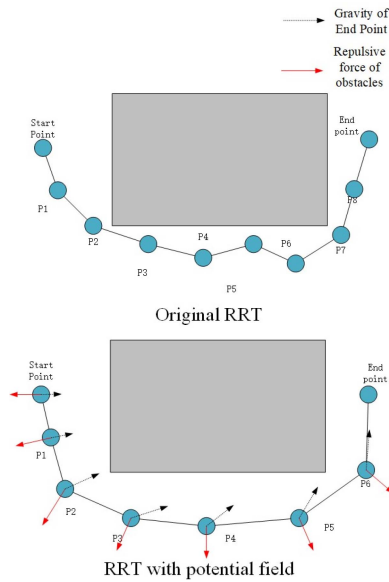


Fig. 6. Method of adding a force field.

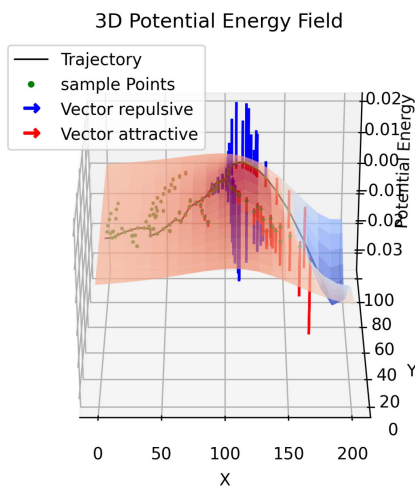


Fig. 7. The effect of RRT random points in the field of artificial potential field.

approach, each new node experiences both a repulsive force emanating from the center of nearby obstacles and an attractive force directed towards the goal. Visually, the repulsive forces are denoted by red arrows pointing away from the obstacle's center, symbolizing the force that pushes the robot away from obstacles. Conversely, the attractive forces are represented by black dotted lines converging on the goal endpoint, guiding the robot towards its destination.

Compared to the conventional RRT algorithm, which lacks the influence of a repulsive field, this integrated method significantly improves the quality of obstacle avoidance and notably reduces the number of nodes required for effective path generation. The introduction of these forces facilitates more direct and efficient paths, minimizing unnecessary detours and computational overhead.

Fig. 7 provides a detailed visualization of the magnitude of the potential field experienced by the random tree nodes at varying

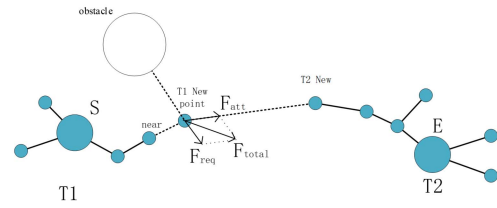


Fig. 8. Planning path of the original RRT algorithm.

distances from both repulsive and attractive sources. When a green random point enters the influence range of the potential field, it encounters a blue-colored region indicating the strength of the repulsive force exerted by the obstacle. Simultaneously, a red-colored region signifies the attractive force pulling the node towards the goal's potential field. This color-coded representation effectively communicates how the balance between repulsion and attraction dynamically influences the direction and step size of the node's expansion, ensuring smooth and safe navigation through complex environments.

2) *Bidirectional APF-RRT* Strategy*: Building upon the APF-RRT* framework, this paper introduces a bidirectional tree generation strategy. The incorporation of a dual-tree growth approach into the APF-RRT* methodology significantly accelerates the exploration process. As illustrated in Fig. 8, during each iteration cycle, Tree T1 and Tree T2 are generated alternately, with each tree expanding towards the most recent node of the opposite tree as its target. This bidirectional expansion not only enhances the efficiency of space exploration but also optimizes the path planning process by leveraging the complementary advantages of both trees.

Compared to the unidirectional APF-RRT* algorithm, the proposed method grows two trees simultaneously—one from the starting point and the other from the goal—expanding them alternately. This approach significantly reduces the time required to cover the search space. By growing the trees toward each other, the probability of their convergence is increased, enabling the algorithm to find a feasible path more quickly. In environments with narrow passages, the bidirectional growth allows the two trees to approach the passage from both ends, making it easier to traverse the constrained space. This overcomes the inefficiency of a single tree, which may struggle due to low sampling probabilities in such scenarios. Furthermore, the alternating expansion strategy dynamically balances the exploration efforts of the two trees, preventing either tree from overly focusing on a specific region and thereby improving the overall rationality and quality of the generated path. The Pseudocode of the Bidirectional APF-RRT* algorithm is in Algorithm 1.

C. Improvements of Bidirectional APF-RRT* in the Frenet Coordinate System

Traditional RRT algorithms are predominantly designed for Cartesian coordinates. By adapting these algorithms to the Frenet coordinate system, we can leverage specific characteristics of this coordinate system to further optimize algorithm performance.

Algorithm 1: Bidirectional APF-RRT*.

```

1: Input: start, goal, obstacle
2: Output: path
3:  $T_1 \leftarrow \text{new RRTStarTree}(s_{\text{start}}, d_{\text{start}})$ 
4:  $T_2 \leftarrow \text{new RRTStarTree}(s_{\text{goal}}, d_{\text{goal}})$ 
5: while not done do
6:    $\text{step\_size} \leftarrow \text{adjust\_step\_size\_based\_on\_potential\_field}(T_1, T_2, s_{\text{obs}}, d_{\text{obs}})$ 
7:   if iteration % 2 == 0 then
8:      $\text{extend\_tree}(T_1, T_2.\text{latest\_node}, \text{step\_size}, s_{\text{obs}}, d_{\text{obs}})$ 
9:      $\text{rewire}(T_1)$ 
10:  else
11:     $\text{extend\_tree}(T_2, T_1.\text{latest\_node}, \text{step\_size}, s_{\text{obs}}, d_{\text{obs}})$ 
12:     $\text{rewire}(T_2)$ 
13:  end if
14:  if connection_found( $T_1, T_2$ ) then
15:    path  $\leftarrow \text{connect\_trees}(T_1, T_2)$ 
16:    return path
17:  end if
18:  iteration  $\leftarrow$  iteration + 1
19:  Increment the iteration counter
20: end while

```

During driving, deviating from the road centerline can result in insufficient space to safely avoid obstacles, necessitating larger steering angles to navigate around them. Such behavior not only inconveniences other drivers but also increases the risk of misjudgment, potentially leading to traffic accidents. Additionally, frequent deviations may trigger lane-departure warnings, causing conflicts between various modules within the driving system. Therefore, it is imperative that the chosen trajectory remains as close to the road centerline as possible to ensure safe and efficient navigation.

In this study, we enhance the lane-centering capability of autonomous vehicles in obstacle-free segments by introducing a micro attraction potential field consistently pointing towards the road's centerline. This approach supplements the established repulsion fields for obstacles and attraction fields for the end-point. The direction of this potential field aligns with the tangent at the vehicle's corresponding point (S,0) on the road reference line. To flexibly adjust the strength of the centerline attraction, we introduced a variable adjustment factor, which has a default value of 0.1, as shown in (16):

$$U_{\text{att}} = \left\{ \frac{\rho^2(P, P_{\text{midline}})}{2} \right\} L \quad (16)$$

Additionally, when the repulsion field of an obstacle overlaps with the centerline attraction field, it can adversely affect obstacle avoidance performance. To address this, within the influence zone of the obstacle, we've established a void zone for the centerline attraction field, where its strength is set to zero to ensure it doesn't interfere. The comparison between the scenarios before and after introducing the reference line can be seen in Fig. 9 and Fig. 10. The left figure depicts the planning outcome in the

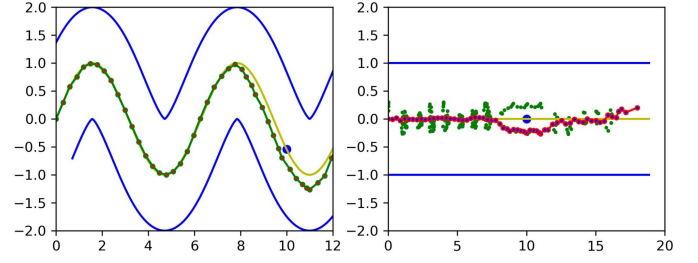


Fig. 9. Planned route after adding centerline gravity field.

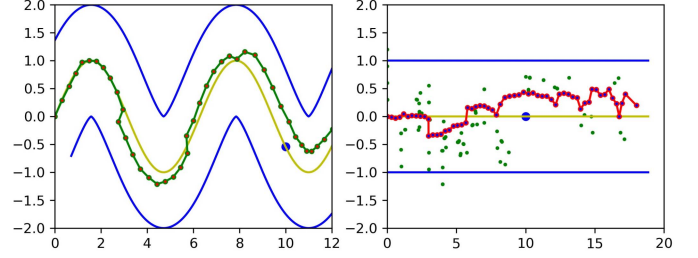


Fig. 10. Obstacle avoidance route without adding centerline.

Cartesian coordinate system, while the right one shows the result in the Frenet coordinate system.

Finally, inspired by other scholars' work on non-uniform sampling in RRT algorithms [28], this paper proposes a method where the adaptive sampling density dynamically adjusts according to the curvature of the reference line, reflecting human-like driving behavior. Specifically, the sampling density is positively correlated with the absolute value of the curvature: higher sampling density is applied in regions with high curvature, while lower sampling density is used in regions with low curvature. Let the curvature of the reference line be denoted as $k(s)$, and let the sampling interval be Δs , which is determined as shown in (17). Here, α represents the curvature sensitivity coefficient, governing the rate at which the sampling density changes with respect to curvature.

$$\Delta s = \frac{\Delta s}{1 + \alpha |k|} \quad (17)$$

When generating candidate sampling points along the reference line, the position of the next sampling point is dynamically adjusted based on the curvature $k(s_i)$ at the current arc length position s_i , as shown in the (18).

$$s_{i+1} = s_i + \Delta s(k(s_i)) \quad (18)$$

This strategy results in a higher density of sampling points in curved regions (high κ) and a sparser distribution of sampling points in straight regions (low κ).

D. Post-Processing Optimization Strategy for Paths

Despite numerous enhancements to the RRT* algorithm aimed at generating smoother trajectories, paths produced by this sampling-based method remain unsuitable for direct tracking by autonomous vehicle algorithms due to inherent limitations of the RRT framework. Therefore, a secondary optimization phase is

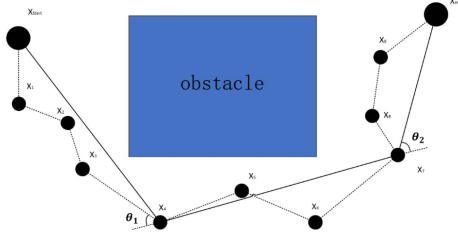


Fig. 11. Refined trajectory suitable for autonomous navigation.

essential to refine the generated path before it can be effectively utilized.

Given that the sampling nature of the RRT algorithm leads to an abundance of redundant waypoints, an initial refinement involves applying a pruning strategy constrained by a maximum steering angle limit. This limit is set to $\frac{\pi}{4}$ radians. As illustrated in Fig. 11, X_{start} and X_{end} denote the start and goal nodes, respectively, while X represents various valid nodes along the path, and symbolizes the vehicle's turning angle.

The process begins by connecting X_{start} directly to X_{end} to form an initial path. If this line segment does not intersect any obstacles, it is considered the optimal path. However, if obstacles are present, the algorithm then connects the preceding valid node X_9 to X_{start} and checks two conditions: whether the line segment between these points is obstacle-free and whether the turn angle formed by X_{end} , X_9 and X_{start} is less than the maximum steering angle of $\frac{\pi}{4}$. If both conditions are satisfied, X_{start} becomes the parent node of X_9 . The algorithm iteratively applies this check until a node (X_4 , for instance) meets all criteria. It then selects X_4 as the new starting point and repeats the optimization process until the path from X_{start} to X_{end} is both obstacle-free and adheres to the maximum steering angle constraint.

The result of this optimization process is depicted in Fig. 11, showcasing a refined trajectory suitable for autonomous navigation.

Following the application of the Pruning Strategy with Maximum Steering Angle Limitation, this paper further employs cubic B-spline curves to smooth the generated path. Compared to Bezier curves, cubic B-splines offer greater flexibility in deployment, enhanced stability, and higher continuity, making them particularly advantageous for the refinement of trajectories intended for autonomous vehicle navigation. The Pseudocode of the Path Planning In Frenet Frame in Algorithm 2.

IV. EXPERIMENT RESULTS AND ANALYSIS

A. Mixed Experiment

To evaluate algorithm performance, we initially conducted experiments in a simulated environment containing both static and dynamic obstacles. Benchmark comparisons were performed against state-of-the-art planning algorithms, including RRT*, Artificial Potential Field (APF), Dynamic Window Approach (DWA), and Double Deep Q-learning (DDQN). All experiments, including those presented in subsequent sections, were executed on a computational platform equipped with a 12th Gen Intel

Algorithm 2: Path Planning in Frenet Frame.

```

1: Input: start, goal, map, reference line
2: Output: path
3:  $(s_{obs}, d_{obs}) = \text{Cartesian\_to\_Frenet}(\text{obstacles\_positions})$ 
4:  $(s_{start}, d_{start}) = \text{Cartesian\_to\_Frenet}((x_{start}, y_{start}))$ 
5:  $(s_{goal}, d_{goal}) = \text{Cartesian\_to\_Frenet}((x_{goal}, y_{goal}))$ 
6:  $\text{path\_frenet} = \text{BiDirectional\_APFRRTStar}(s_{start}, d_{start}, s_{goal}, d_{goal}, s_{obs}, d_{obs})$ 
7:  $\text{path\_optimized} = \text{PathOptimization}(\text{path\_frenet})$ 
8:  $\text{path\_optimized} = \text{Frenet\_to\_Cartesian}(\text{path\_optimized\_frenet})$ 
9: return  $\text{path\_optimized}$ 
    
```

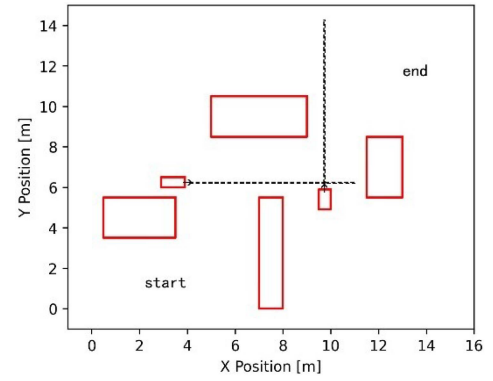


Fig. 12. Experiment A setup.

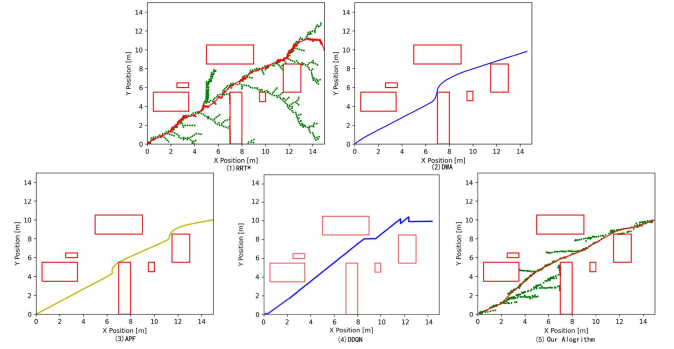


Fig. 13. Simulation results.

Core™ i7-12700H processor (base frequency: 2.30 GHz) and 40 GB RAM.

The experimental arena comprised a 15×15 unit² planar workspace containing four rectangular static obstacles of heterogeneous dimensions and two dynamic obstacles moving at 0.5 m/s. For consistency, all trials commenced from the start point at coordinates (0, 0) and concluded at the end point located at (15, 10). The experimental setup is illustrated in Fig. 12.

Fig. 13(1)–(5) respectively illustrate the planning performance of the RRT* algorithm, DWA algorithm, APF algorithm, Double Deep Q-learning algorithm and the proposed algorithm in the aforementioned simulation environment. Fig. 13(1)

TABLE I
COMPARISON OF ALGORITHMS IN SCENARIOS

Result		
RRT*	single-planning cycle time(s)	1.21
	Average path length (m)	21.14
	success rate (%)	92
DWA	single-planning cycle time (s)	0.046
	Average path length (m)	18.1
	success rate (%)	58
APF	single-planning cycle time(s)	0.02
	Average path length (m)	18.6
	success rate (%)	1.34
DDQN	single-planning cycle time(s)	0.041
	Average path length (m)	19.6
	success rate (%)	100
Our Alogrithm	single-planning cycle time (s)	0.039
	Average path length (m)	17.79
	success rate (%)	100

presents the simulation results of the RRT* algorithm. The algorithm exhibits poor search efficiency and generates relatively long paths, which fail to meet the requirements for real-time obstacle avoidance. Fig. 13(2) displays the results of the DWA algorithm, a widely used local path planning method. While its planning outcomes are relatively satisfactory, the performance heavily depends on predefined parameters such as the velocity space size and prediction horizon. In complex environments, this often necessitates a trade-off between planning quality and computational speed. Fig. 13(3) shows the performance of the Artificial Potential Field (APF) method. Although the APF method demonstrates both high-quality path planning and computational efficiency, extensive experiments reveal that it suffers from issues such as local minima and unreachable target points, leading to a low planning success rate. Fig. 13(4) demonstrates the performance of the DDQN algorithm. This algorithm, which is based on deep reinforcement learning, can generate safe spatial trajectories. However, it still has certain deficiencies in terms of the length of the generated path distance, and its performance highly depends on the number of pre-training sessions. Finally, Fig. 13(5) showcases the performance of our proposed algorithm. By integrating potential field concepts with sampling-based planning, the algorithm significantly reduces sensitivity to parameter settings while achieving superior planning results.

Each algorithm underwent 20 independent trials with key performance metrics recorded, including single-planning cycle time, mean path length, and success rate. As detailed in Table I, our proposed algorithm demonstrated 15.9% shorter average path length compared to RRT*, 45% higher success rate than APF, and 17.4% faster computation speed relative to DWA, while maintaining 100% success rate across all trials.

To further validate the obstacle avoidance performance of the proposed path planning algorithm in dynamic environments and its sensitivity to potential field parameters, Environment B was designed for systematic evaluation. Environment B adopts the same 15×15 unit² planar workspace as Environment A. The initial position of the test vehicle is set to (0, 5.5), with the target point located at (14.5, 5.5). The forward speed of the vehicle is maintained at a constant 1.5 m/s. The workspace contains four

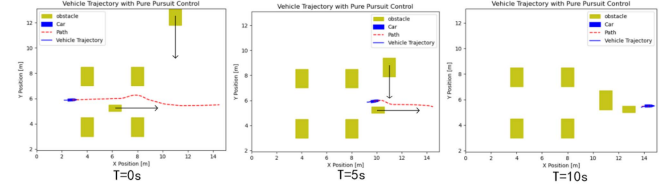


Fig. 14. Simulation results.

TABLE II
IMPACT OF KREP VALUES ON PATH PLANNING

Result		
krep=0.1	Path Length (m)	14.70
	Computation Time (s)	0.010
krep=0.01	Path Length (m)	14.50
	Computation Time (s)	0.006
krep=0.001	Path Length (m)	14.24
	Computation Time (s)	0.006
krep=0.0001	Path Length (m)	14.25
	Computation Time (s)	0.007

rectangular static obstacles of identical dimensions, as well as two dynamic obstacles moving at a constant speed of 1 m/s. It includes direction Angle modeling, in which one follows a trajectory exactly opposite to that of the test vehicle, and the other advances perpendicularly to the test vehicle. Fig. 14 demonstrate that the proposed algorithm successfully accomplished the obstacle avoidance task in the dynamic environment.

To further verify the sensitivity of the algorithm to key potential field parameters, four groups of comparative experiments were conducted with varying values of the repulsive field gain coefficient k_{rep} (0.1, 0.01, 0.001, and 0.0001). Each parameter set was independently executed 10 times, and statistical averages were calculated. The recorded metrics include path length (in units) and planning time per iteration (in seconds), as summarized in Table II.

Experimental results show that as k_{rep} decreases from 0.1 to 0.0001, the path length exhibits a monotonically decreasing trend, with a total variation of 3.1%. Notably, when $k_{rep} \leq 0.001$, the path length converges, with only a 0.07% difference between the 0.001 and 0.0001 groups. This indicates that the contribution of the repulsive field to path optimization becomes saturated. A higher repulsive coefficient ($k_{rep} = 0.1$) leads to a significant increase in computation time—66.7% longer than the baseline group—which is directly related to the increased number of iterations caused by the high gain value. When $k_{rep} \leq 0.001$, the computation time remains stable within the range of 0.006–0.007 seconds, meeting the requirements for real-time planning. Over the entire tested range of $k_{rep} \in [0.0001, 0.1]$, the path length variation remains within 3.1%, demonstrating the algorithm's robustness to potential field parameters.

B. Ablation Experiment

To elucidate the contributions of individual components within the proposed algorithm, we have systematically decomposed the suggested enhancements and performed a series of ablation studies. The experiments were divided into five groups:

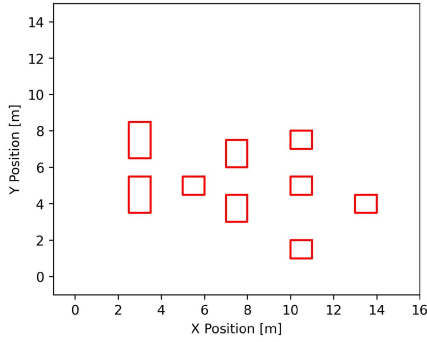


Fig. 15. Experiment B setup.

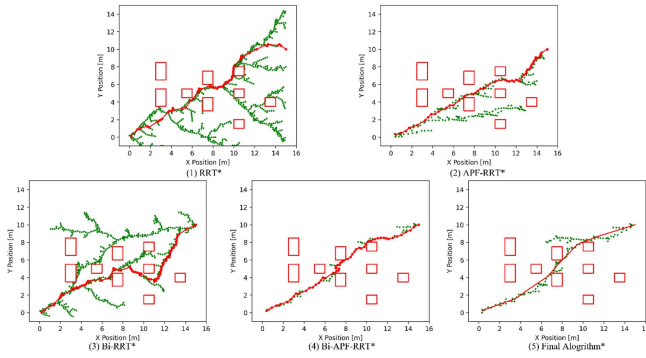


Fig. 16. Ablation simulation results.

Group (1) RRT*, Group (2) APF-RRT*, Group (3) double-RRT*, (4). double-APF-RRT* and Group (5). The experimental setup is illustrated in Fig. 15, where each configuration was tested under identical conditions to ensure comparability. The test environment consisted of a square space measuring 15x15 units, populated with nine rectangular obstacles of varying sizes. For consistency, all trials commenced from the start point at coordinates (0, 0) and concluded at the end point located at (15, 10).

Fig. 16(1) presents the simulation results for the RRT algorithm. Fig. 16(2) shows the simulation outcomes for the APF-RRT* algorithm, which exhibits significantly improved search efficiency and smoother paths compared to the original RRT* due to the added guidance of an artificial potential field (APF). Fig. 16(3) illustrates the performance of the Bi-RRT* algorithm, which employs a bidirectional strategy to achieve higher sampling efficiency than RRT*. However, the lack of APF guidance leads to excessively high path curvature. Fig. 16(4) corresponds to the Bi-APF-RRT algorithm, which adds APF guidance to the Bi-RRT* framework, thereby addressing the issues caused by the latter. Nevertheless, inherent limitations in the sampling algorithm result in less-than-optimal path smoothness. Fig. 19(5) represents the final version of the proposed algorithm, incorporating constraints on steering angles and utilizing cubic Bézier curve optimization for path planning. This approach yields smooth and fluent paths that meet the kinematic requirements of autonomous vehicles. Within this simulation environment, each algorithm was subjected to 20 repeated trials, with average performance metrics—such as runtime, planned path length, mean path curvature—calculated and

TABLE III
COMPARISON OF ALGORITHMS IN SCENARIOS

		Result
RRT*	Average time (s)	3.56
	Average path length (m)	20.81
	Average path curvature (m^{-1})	0.92
	curvature change rate(m^{-1})	0.786
APF- RRT*	Average time (s)	0.33
	Average path length (m)	19.21
	Average path curvature (m^{-1})	0.58
	curvature change rate(m^{-1})	0.34
Bi-RRT*	Average time (s)	0.91
	Average path length (m)	21.7
	Average path curvature (m^{-1})	1.34
	curvature change rate(m^{-1})	0.651
Bi-APF-RRT*	Average time (s)	0.08
	Average path length (m)	19.81
	Average path curvature(m^{-1})	0.73
	curvature change rate(m^{-1})	0.593
Final Algorithm	Average time (s)	0.09
	Average path length (m)	18.94
	Average path curvature(m^{-1})	0.06
	curvature change rate(m^{-1})	0.008

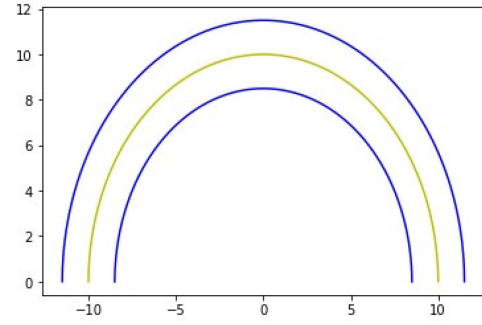


Fig. 17. U-turn curve experiment.

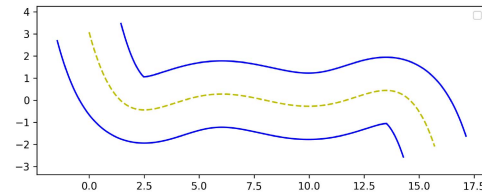


Fig. 18. Complex curve experiment.

mean curvature change rate. The detailed results are presented in Table III.

C. Typical Roadways Experiment

To further evaluate the performance of the enhanced method within the Frenet coordinate framework and to assess its path planning and obstacle avoidance capabilities in diverse and complex road environments, we conducted tests on two representative roadway configurations: a conventional U-turn and an intricately curved segment, as depicted in Figs. 17 and 18. Both scenarios feature a lateral extent of 20 meters and a lane width of 3 meters. In both scenarios, obstacles were positioned along the centerlines, and the enhanced RRT algorithm was employed to perform path planning.

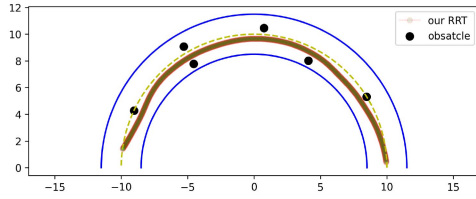


Fig. 19. U-turn path planning effect.

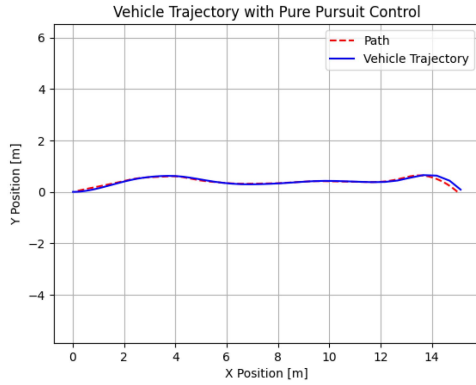


Fig. 20. U-turn road vehicle tracking performance.

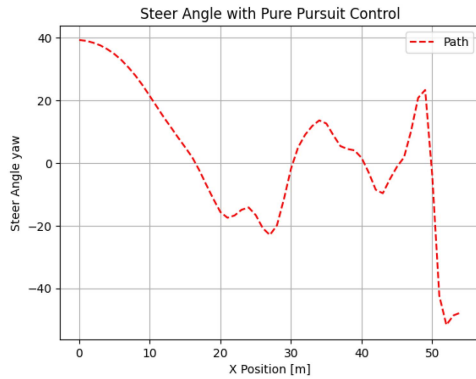


Fig. 21. U-turn road range of steering angles.

Notably, the essence of the Frenet coordinate system lies in curvature transformation. The use of a rectangular simplification model may lead to abrupt changes in curvature at right angles and along their adjacent sides, potentially causing deformation of obstacles and the vehicle, which can impact collision detection. Therefore, in our road experiments, we adopted a circular composite model to approximate the shapes of rectangular obstacles and vehicles. This approach provides a continuous and bounded curvature distribution, thereby minimizing abrupt changes in curvature. Such a design facilitates smoother path descriptions within the Frenet coordinate system, reducing errors associated with sharp curvature changes. In the U-turn road test, six obstacles, each with a diameter of 0.6 meters, were placed. The algorithm initially performed planning in the Frenet coordinate system, with the six obstacles positioned at (10.5, 1), (2.8, 0), (10.5, -0.5), (5.5, 1), (7.5, -0.5), and (13.5, 0). The planning start point was set at (0, 0), and the end point at (15, 0). After running the algorithm, the planned results were mapped back to the Cartesian coordinate system. As shown in Fig. 22, the

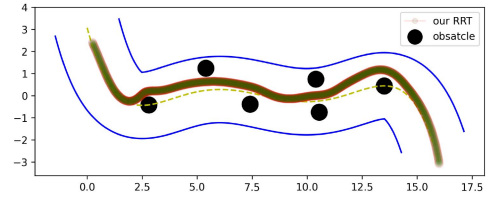


Fig. 22. Complex curve road planning effect.

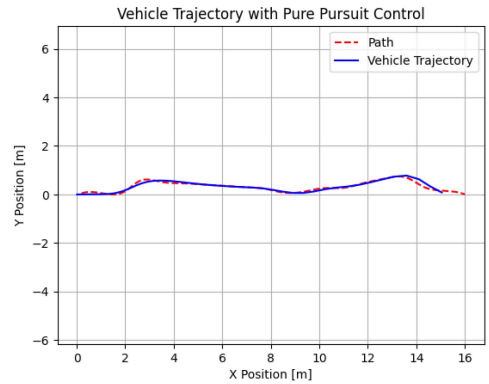


Fig. 23. Complex curve road vehicle tracking performance.

red line represents the obstacle-avoidance path planned by our method, the yellow dashed line serves as a reference, and the green line indicates the road centerline, which is used as a reference for our algorithm. The vehicle's length is 2 meters, width is 50 cm, initial speed is 2 m/s, and acceleration is set to 1.0 m/s². The trajectory tracking algorithm employed is the Pure Pursuit algorithm, with a lookahead distance parameter of 1.0 m. The vehicle tracking performance is illustrated in Fig. 20, while the range of steering angles is depicted in Fig. 21.

The experimental results demonstrate that in the U-turn scenario, the proposed algorithm successfully avoids obstacles while maintaining the vehicle as close to the road centerline as possible to ensure safety. Additionally, using the Pure Pursuit tracking algorithm, the autonomous vehicle successfully follows the planned path with stable steering angles, adhering to the vehicle's kinematic constraints.

In the complex curve road test, we again placed six obstacles but increased their diameter to 0.9 meters to heighten the planning challenge. The algorithm initially performed path planning in the Frenet coordinate system, with the six obstacles positioned at coordinates (10.5, 1), (2.8, 0), (10.5, -0.5), (5.5, 1), (7.5, -0.5), and (13.5, 0). The start point for planning was set at (0, 0), and the end point at (15, 0). After executing the algorithm, the planned path was mapped back to the Cartesian coordinate system. As shown in Fig. 25, the red line represents the obstacle-avoidance path planned by our method, the yellow dashed line serves as a reference, and the green line indicates the road centerline, which is used as a reference for our algorithm. The vehicle has a length of 2 meters, a width of 50 cm, an initial speed of 2 m/s, and an acceleration of 1.0 m/s². The trajectory tracking algorithm employed is the Pure Pursuit algorithm, with a lookahead distance parameter of 1.0 m. The vehicle tracking

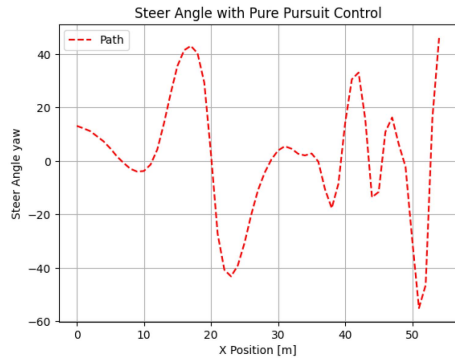


Fig. 24. Complex curve road range of steering angles.

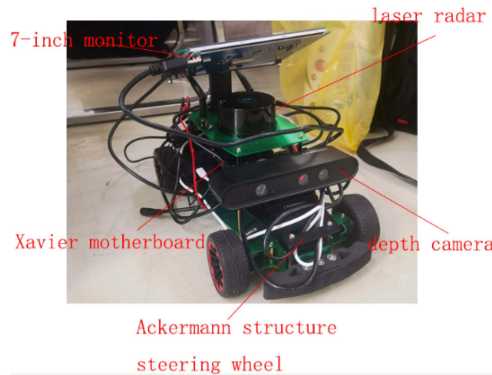


Fig. 25. Schematic diagram of intelligent ROS car structure.

performance is illustrated in Fig. 23, while the range of steering angles is depicted in Fig. 24.

The experimental results demonstrate that, even in environments with large obstacles and complex road conditions, the proposed algorithm successfully avoids obstacles while maintaining the vehicle as close to the road centerline as possible to ensure safety. Furthermore, utilizing the Pure Pursuit tracking algorithm, the autonomous vehicle is able to successfully follow the planned path with stable steering angles, adhering to the vehicle's kinematic constraints.

D. Real Experiment

In addition to the aforementioned simulated road experiments, this paper conducted additional simulation experiments under real conditions to investigate the planning speed and effectiveness of route planning on actual vehicles. A smart car with Ackermann steering structure was chosen for the real vehicle experiments, as shown in Fig. 25. The components of the smart car include a 7-inch display, a laser radar, an Xavier mainboard, a depth camera, and a servo motor for Ackermann steering. The car's system runs on Ubuntu 18.04 and is equipped with ROS-melodic as the autonomous driving system. In practical operation, the experiment utilized the car's built-in 4ROS laser radar for vehicle localization and mapping in indoor environments, and the road centerline calculation was achieved through the onboard binocular camera. To better detect the obstacle avoidance situations of the vehicle in a dynamic environment during the experiment, obstacles were not pre-detected in the mapping task but were detected during the planning process

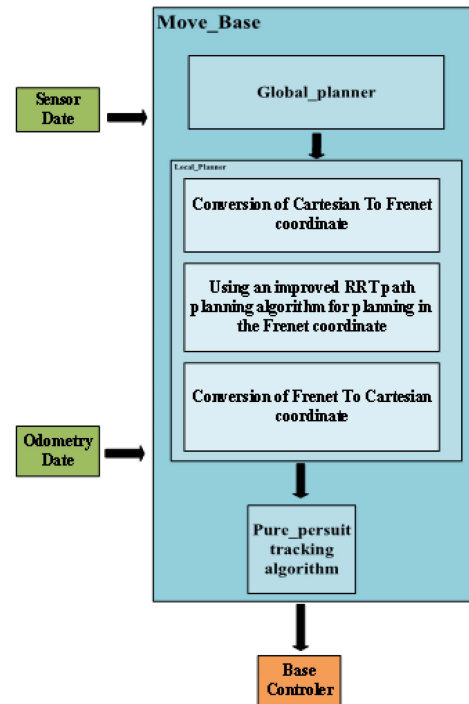


Fig. 26. Structure diagram of auto drive system.

using laser radar. The overall planning and control modules were written in C++ and executed on the Xavier mainboard.

The overall structure of the autonomous driving system is shown in Fig. 26 and consists of four parts: perception module, localization module, planning module, and control module. The algorithms proposed in this paper are deployed within the control module's Move_Base framework. The perception module and Odometry localization module transmit vehicle location and map information to the planning module. The global planner, Global_planner, is responsible for passing the rough route to the local path planner, Local_planner, and the algorithms proposed in this paper are deployed in the latter. Starting with the given global route as the centerline in the Frenet coordinate system, a coordinate transformation is performed to convert map information, including obstacles, from Cartesian coordinates to Frenet coordinates. The planning is then carried out using the algorithm described earlier, and the planned route is transformed back into Cartesian coordinates as point arrays and provided to the Pure-pursuit tracking algorithm. Finally, velocity commands are output to the vehicle controller to accomplish autonomous vehicle driving.

The results of the real vehicle experiments are shown in Fig. 27, Fig. 28 and Fig. 29. The experimental scenario was set as a U-turn road with a box-shaped obstacles placed on the road. The algorithm proposed in this paper demonstrated excellent navigation performance in actual obstacle avoidance scenarios, meeting the global navigation requirements of the Ackermann structure vehicle. The trajectories of the main vehicle and obstacles consistently maintained a safe distance. Due to consideration of vehicle dynamics, the performance of the main vehicle remained stable within reasonable ranges of yaw angle and yaw rate.

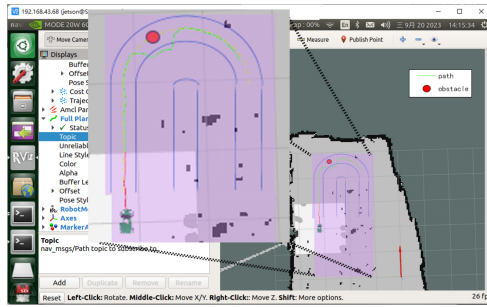
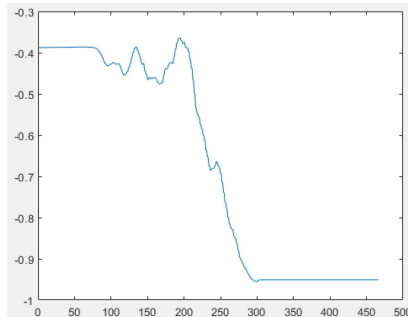
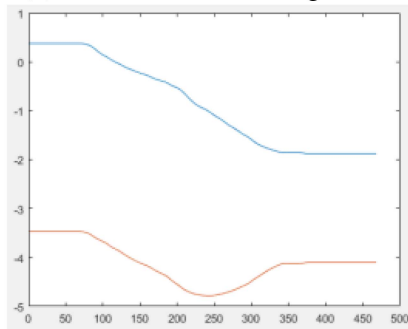


Fig. 27. Actual vehicle test results.



(1) Vehicle Deflection Angle Yaw



(2) Vehicle Displacement x y

Fig. 28. Vehicle parameters.



Fig. 29. Scene diagram of actual vehicle experiment.

The trajectories of the main vehicle and obstacles consistently maintained a safe distance. Due to the consideration of vehicle dynamics, the performance of the main vehicle remained stable within a reasonable range of yaw angle and yaw rate.

V. CONCLUSION

This study proposes a bidirectional APF-RRT* path planning algorithm embedded in the Frenet coordinate system. A constrained sampling strategy is developed to reduce search

space dimensionality by approximately 70%, leveraging the natural decoupling of longitudinal (s-direction) and lateral (d-direction) motions while restricting lateral sampling to ± 1.5 meters (typical lane width). An improved bidirectional RRT* algorithm is designed with integrated artificial potential field (APF) mechanisms, incorporating dynamically adjusted obstacle repulsion forces, moving obstacle repulsion forces, and target attraction forces to suppress path oscillations and avoid local optima. Furthermore, an S-shaped attractive force function and curvature-adaptive sampling strategy are proposed to ensure path adherence to reference lines, combined with cubic B-spline smoothing and steering angle-constrained pruning to enhance trajectory smoothness and feasibility. Extensive simulations and real-vehicle experiments validate the method's effectiveness. Although the approach demonstrates strong efficiency and constraint adaptation, geometric characteristics of the Frenet coordinate system in high-curvature scenarios offer optimization potential. Future work will explore dynamic curvature distribution strategies and benchmark curvature model corrections to improve geometric compatibility, alongside expanding dynamic obstacle prediction capabilities for enhanced collision avoidance.

REFERENCES

- [1] K. Muhammad, A. Ullah, J. Lloret, J. D. Ser, and V. H. C. de Albuquerque, "Deep learning for safe autonomous driving: Current challenges and future directions," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 7, pp. 4316–4336, Jul. 2021, doi: [10.1109/TITS.2020.3032227](https://doi.org/10.1109/TITS.2020.3032227).
- [2] A. Rasouli and J. K. Tsotsos, "Autonomous vehicles that interact with pedestrians: A survey of theory and practice," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 3, pp. 900–918, Mar. 2020, doi: [10.1109/TITS.2019.2901817](https://doi.org/10.1109/TITS.2019.2901817).
- [3] B. Li, Y. Ouyang, L. Li, and Y. Zhang, "Autonomous driving on curvy roads without reliance on frenet frame: A cartesian-based trajectory planning method," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 9, pp. 15729–15741, Sep. 2022, doi: [10.1109/TITS.2022.3145389](https://doi.org/10.1109/TITS.2022.3145389).
- [4] X. Zhang, A. Liniger, and F. Borrelli, "Optimization-based collision avoidance," *IEEE Trans. Control Syst. Technol.*, vol. 29, no. 3, pp. 972–983, May 2021, doi: [10.1109/TCST.2019.2949540](https://doi.org/10.1109/TCST.2019.2949540).
- [5] 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, Apr. 2016, doi: [10.1109/TITS.2015.2498841](https://doi.org/10.1109/TITS.2015.2498841).
- [6] Y. Volkan Pehlivanoglu and P. Pehlivanoglu, "An enhanced genetic algorithm for path planning of autonomous UAV in target coverage problems," *Appl. Soft Comput.*, vol. 112, 2021, Art. no. 107796, doi: [10.1016/j.asoc.2021.107796](https://doi.org/10.1016/j.asoc.2021.107796).
- [7] W. Huang, Z. Xu, and L. Zhu, "The minimum regret path problem on stochastic fuzzy time-varying networks," *Neural Netw.*, vol. 153, pp. 450–460, Sep. 2022, doi: [10.1016/j.neunet.2022.06.029](https://doi.org/10.1016/j.neunet.2022.06.029).
- [8] H. Gong et al., "Bidirectional jump point search path-planning algorithm based on electricity-guided navigation behavior of electric eels and map preprocessing," *Biomimetics*, vol. 8, no. 5, 2023, Art. no. 387, doi: [10.3390/biomimetics8050387](https://doi.org/10.3390/biomimetics8050387).
- [9] Y. Huang et al., "A motion planning and tracking framework for autonomous vehicles based on artificial potential field elaborated resistance network approach," *IEEE Trans. Ind. Electron.*, vol. 67, no. 2, pp. 1376–1386, Feb. 2020, doi: [10.1109/TIE.2019.2898599](https://doi.org/10.1109/TIE.2019.2898599).
- [10] Q. Pan et al., "Adaptive cooperative gene regulatory network optimized by elastic deformation algorithm for multirobot hunting[j]," *IEEE Syst. J.*, vol. 17, no. 3, pp. 4843–4854, Sep. 2023.
- [11] J. Tang et al., "An improved artificial electric field algorithm for robot path planning[j]," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 60, no. 2, pp. 2292–2304, Apr. 2024.
- [12] H. Chen and X. Zhang, "Path planning for intelligent vehicle collision avoidance of dynamic pedestrian using Att-LSTM, MSFM, and MPC

- at unsignalized crosswalk,” *IEEE Trans. Ind. Electron.*, vol. 69, no. 4, pp. 4285–4295, Apr. 2022.
- [13] P. Wang, S. Gao, L. Li, B. Sun, and S. Cheng, “Obstacle avoidance path planning design for autonomous driving vehicles based on an improved artificial potential field algorithm,” *Energies*, vol. 12, no. 12, 2019, Art. no. 2342, doi: [10.3390/en12122342](https://doi.org/10.3390/en12122342).
- [14] M. A. V. J. Muthugala, S. M. B. P. Samarakoon, and M. R. Elara, “Toward energy-efficient online complete coverage path planning of a ship hull maintenance robot based on glasius bio-inspired neural network,” *Expert Syst. Appl.*, vol. 187, 2022, Art. no. 115940.
- [15] L. Chen et al., “STR: Spatial–Temporal RetNet for Distributed Multi-Robot Navigation,” *IEEE Trans. Autom. Sci. Eng.*, vol. 22, pp. 10429–10441, 2025.
- [16] R. Han et al., “Reinforcement learned distributed multi-robot navigation with reciprocal velocity obstacle shaped rewards,” *IEEE Robot. Autom. Lett.*, vol. 7, no. 3, pp. 5896–5903, Jul. 2022.
- [17] Y. Jia et al., “Hierarchical perception-improving for decentralized multi-robot motion planning in complex scenarios,” *IEEE Trans. Intell. Transp. Syst.*, vol. 25, no. 7, pp. 6486–6500, Jul. 2024.
- [18] S. Karaman, M. R. Walter, A. Perez, E. Frazzoli, and S. Teller, “Anytime motion planning using the RRT*,” in *Proc. 2011 IEEE Int. Conf. Robot. Automat.*, Shanghai, China, 2011, pp. 1478–1483, doi: [10.1109/ICRA.2011.5980479](https://doi.org/10.1109/ICRA.2011.5980479).
- [19] S. Klemm et al., “RRT*-Connect: Faster, asymptotically optimal motion planning,” in *Proc. 2015 IEEE Int. Conf. Robot. Biomimetics (ROBIO)*, Zhuhai, China, 2015, pp. 1670–1677, doi: [10.1109/ROBIO.2015.7419012](https://doi.org/10.1109/ROBIO.2015.7419012).
- [20] R. Zhang et al., “Intelligent path planning by an improved RRT algorithm with dual grid map,” *Alexandria Eng. J.*, vol. 88, pp. 91–104, 2024.
- [21] J. D. Gammell, S. S. Srinivasa, and T. D. Barfoot, “Informed RRT*: Optimal sampling-based path planning focused via direct sampling of an admissible ellipsoidal heuristic[C]/,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2014, pp. 2997–3004.
- [22] J. Fan, X. Chen, and X. Liang, “UAV trajectory planning based on bi-directional APF-RRT* algorithm with goal-biased,” *Expert Syst. Appl.*, vol. 213, 2023, Art. no. 119137.
- [23] M. Cao, X. Zhou, and Y. Ju, “Robot motion planning based on improved RRT algorithm and RBF neural network sliding,” *IEEE Access*, vol. 11, pp. 121295–121305, 2023.
- [24] S. Huang, “Path planning based on mixed algorithm of RRT and artificial potential field method,” in *Proc. 4th Int. Conf. Intell. Robot. Control Eng.*, Lanzhou, China, 2021, pp. 149–155, doi: [10.1109/IRCES3649.2021.9570910](https://doi.org/10.1109/IRCES3649.2021.9570910).
- [25] M. Werling, J. Ziegler, S. Kammel, and S. Thrun, “Optimal trajectory generation for dynamic street scenarios in a frenet frame,” in *Proc. 2010 IEEE Int. Conf. Robot. Automat.*, Anchorage, AK, USA, 2010, pp. 987–993, doi: [10.1109/ROBOT.2010.5509799](https://doi.org/10.1109/ROBOT.2010.5509799).
- [26] H. Fan et al., “Baidu apollo em motion planner,” 2018, *arXiv:1807.08048*.
- [27] J. Peng and M. Guo, “Local path planning for unmanned vehicles based on improved artificial potential field method in frenet coordinates,” *Acta Armamentarii*, vol. 45, no. 7, pp. 2097–2109, 2024, DOI: [10.12382/bgxb.2023.0305](https://doi.org/10.12382/bgxb.2023.0305).
- [28] J. Wang, T. Li, B. Li, and M. Q.-H. Meng, “GMR-RRT*: Sampling-based path planning using Gaussian mixture regression,” *IEEE Trans. Intell. Veh.*, vol. 7, no. 3, pp. 690–700, Sep. 2022.
- [29] J. Huang, Z. He, Y. Arakawa, and B. Dawton, “Trajectory planning in frenet frame via multi-objective optimization,” *IEEE Access*, vol. 11, pp. 70764–70777, 2023.
- [30] Z. Arjmandzadeh et al., “A comparative study on autonomous vehicle local path planning through model predictive control and frenet frame method,” *SAE Intl. J. CAV*, vol. 7, no. 4, pp. 447–458, 2024.
- [31] Z. Yu, M. Zhu, K. Chen, X. Chu, and X. Wang, “LF-Net: A learning-based frenet planning approach for urban autonomous driving,” *IEEE Trans. Intell. Veh.*, vol. 9, no. 1, pp. 1175–1188, Jan. 2024.



Rui Zhang received the Ph.D. degree in transportation engineering from the Wuhan University of Technology, Wuhan, China. She is currently an Associate Professor with the School of Automotive and Transportation, Tianjin University of Technology and Education, Tianjin, China. Her main research interests include intelligent transportation systems and intelligent vehicles.



He Guo received the B.S. degree from NanKai University, Tianjin, China, in 2019, and the M.S. degree from the Tianjin University of Technology and Education, Tianjin, China, in 2024. He is currently working toward the Doctor degree with Tiangong University, Tianjin. His research focuses on intelligent vehicles technology.



PORTATION SYSTEMS.

Miguel Angel Sotelo (Fellow, IEEE) received the Ph.D. degree in electrical engineering from the University of Alcalá (UAH), Madrid, Spain, in 2001. He is currently a Full Professor with the Department of Computer Engineering, UAH. His research interests include autonomous vehicles and prediction of intentions. He is the Past President of the IEEE Intelligent Transportation Systems Society. He was the Editor-in-Chief of IEEE Intelligent Transportation Systems Magazine, ITSS Newsletter, and an Associate Editor for IEEE TRANSACTIONS ON INTELLIGENT TRANS-

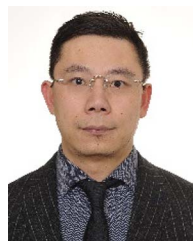


Wollongong, Wollongong, Australia. His current research interests include robotics and automation, vehicle dynamics and control systems, and electric vehicles. Prof. Du is an Associate Editor for IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, a Subject Editor of the *Journal of Franklin Institute*, and an Associate Editor for IEEE Control Systems Society Conference Editorial Board. He is a recipient of Fellow of the Institute of Engineering and Technology.

Haiping Du (Senior Member, IEEE) received the Ph.D. degree in mechanical design and theory from Shanghai Jiao Tong University, Shanghai, China, in 2002. He was a Postdoctoral Research Associate with Imperial College London, London, U.K., from 2003 to 2005, and with the University of Hong Kong, Hong Kong, from 2002 to 2003. He was a Research Fellow with the University of Technology Sydney, Ultimo, Australia, from 2005 to 2009. He is currently a Professor with the School of Electrical, Computer and Telecommunications Engineering, University of



Andriukaitis Darius received the Ph.D. degree in electrical and electronic engineering from the Kaunas University of Technology, Kaunas, Lithuania, in 2008. He is currently a Full Professor with the Faculty of Electrical and Electronics, Kaunas University of Technology. He is the author/coauthor of more than 60+ refereed journal papers. His research interests include finding solutions for the issues related to interactive electronic systems, integrated information systems, and WSN. He is the executive Editor in Chief of journal *Elektronika Ir Elektrotechnika*.



of Fellow of the Institute of Engineering and Technology.

Zhixiong Li (Senior Member, IEEE) received the Ph.D. degree in transportation engineering from the Wuhan University of Technology, Wuhan, China, in 2013. He was with Yonsei Frontier Lab, Yonsei University, Seoul, Republic of Korea. His research interests include intelligent vehicles and control, loop closure detection, and mechanical system modeling and control. Dr. Li is an Associate Editor for IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS and a Column Editor of IEEE Intelligent Transportation Systems Magazine. He is a recipient