

Article

Multi-Area Sampling-Based Spatiotemporal Trajectory Planning for Autonomous Driving in Dynamic On-Road Scenarios

Shuhuan Ma, Zhiqiang Ning, Lixin Wei * and Pengpeng Chai

Department of Mechanical Engineering, Shanxi Institute of Technology, Yangquan 045008, China; mashuhuan@sxit.edu.cn (S.M.); ningzhiqiang@sxit.edu.cn (Z.N.); chaipengpeng@sxit.edu.cn (P.C.)

* Correspondence: weilixin@sxit.edu.cn

Abstract: This paper focuses on the spatiotemporal trajectory planning problem faced by autonomous driving with a dynamic on-road situation. To solve the swing problem which is caused by the motions of obstacles, a multi-area sampling method is proposed. The main idea is sampling endpoints in a series of defined areas at a fixed time interval, which will generate suitable trajectories with speed information to deal with complex maneuver tasks. Considering the driving safety and comfort, the cost function is designed deliberately for the generated trajectories in each area to evaluate the behaviors of the automobile. Then, the best trajectory in the whole course is found by the dynamic programming-based approach, which is presented to optimize the problem-solving process and at the same time reduce the computational burden which is brought about by the multi-area sampling method. Finally, the effectiveness of the proposed trajectory planning method is demonstrated in different overtaking scenarios of structured roads.

Keywords: autonomous driving; trajectory planning; dynamic programming



Citation: Ma, S.; Ning, Z.; Wei, L.; Chai, P. Multi-Area Sampling-Based Spatiotemporal Trajectory Planning for Autonomous Driving in Dynamic On-Road Scenarios. *World Electr. Veh. J.* **2024**, *15*, 547. <https://doi.org/10.3390/wevj15120547>

Academic Editors: Guoxing Bai and Michael Fowler

Received: 21 October 2024

Revised: 19 November 2024

Accepted: 20 November 2024

Published: 23 November 2024



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

1. Introduction

Autonomous vehicles have great potential to tackle complex driving tasks, especially in dangerous and difficult conditions. Owing to its superiority in terms of relieving traffic congestion, enhancing vehicle and road utilization rates, and improving ride comfort and handling stability, autonomous driving has attracted significant attention from both academia and industry. Benefiting from the development of artificial intelligence and computer vision, the preliminary success of autonomous vehicle has been achieved in recent decades [1–5].

The decision-making and planning layer, as one of the most important parts of autonomous driving, serves as a connecting link between environmental perception and motion control. It generates appropriate driving behavior to cope with the changeable transportation condition, according to the information from the perception layer [6]. As the main factor differentiating the autonomous vehicle from the traditional one, the decision-making and planning layer has a direct impact on driving safety and comfort.

The goal of global planning is to identify the optimal route from the starting position to the target position under specific conditions with the support of map information. Local planning, which creates the reference trajectory for the vehicle to follow, is of great significance for driving safety, particularly in unknown and dynamic conditions. Consequently, global planning is a type of prior planning that is both global and optimal. However, it becomes vulnerable and risky when applied in changing situations. In contrast, local planning focuses on local environmental details, enabling the vehicle to have excellent obstacle-avoidance capabilities. In this paper, by taking into account the motion constraints of the vehicle and obstacles, we explore local trajectory planning to enhance the driving performance within dynamic on-road scenarios.

Owing to the high requirement in industry and challenge in academic research, a great quantity of literature has been devoted to dealing with the trajectory planning problem with

autonomous driving in past decades [7–15]. These methods can be mainly divided into two categories: numerical optimization-based approaches and sampling-based approaches.

Numerical optimization-based approaches obtain the solution by means of a differentiable function subject to various constraints. As a result, road conditions, static and dynamic obstacles, and the physical limitations of the ego vehicle can be readily considered. In [16], a practical path planner under dynamic conditions is proposed, with road information being detected in real-time. To obtain the optimal solution, the numerical nonlinear optimization algorithm is introduced to enhance the quality of the trajectory. Ref. [17] describes a computationally efficient planner for autonomous on-road driving in dynamic conditions. Firstly, a rough trajectory is generated by searching in spatiotemporal space with the dynamic programming method. Then, a focused trajectory search is performed to find the best solution according to the smoothness of the path. In [18], a variational formulation is used to design the trajectory planner. The obstacle constraints are regarded as polygons, and an objective function is deliberately designed to describe the physical limits and comfort. However, the non-convex problems, especially in constraints, are difficult to tackle and easily lead to a local optimum in the process of planning.

Sampling-based approaches generate a series of candidate trajectories by randomly sampling in the state space. Then, the optimal trajectory, which can minimize a elaborately designed cost function of safety and comfort, is selected from the candidate trajectory set. As a typical sampling-based method, the Rapidly Exploring Random Tree (RRT) approach can give a fast solution in multi-dimensional problems. In [19], a real-time motion planning framework, which improves the RRT approach by closed-loop prediction, is presented for autonomous vehicles in structured environments. Considering the half-car dynamical model, a fast local steering algorithm is introduced in [20] for motion planning based on the RRT* approach. Though the RRT approach is suitable for global and local planning, the obtained path is not smooth enough.

Compared with the RRT approach, the state lattice method is suitable for local and unknown environments. Ref. [21] proposes a state space sampling method, where the vehicle model and constraints are taken into consideration, for highly constrained environments. Considering time and velocity dimensions, ref. [22] proposes a spatiotemporal lattice planner for autonomous vehicle to deal with the moving obstacles in dynamic environments. The state space is redesigned to adapt to the road direction so that the computational complexity can be reduced with fewer endpoints. Ref. [23] proposes a trajectory planning framework for highway driving in structured environment based on state lattice. It uses a search space representation to find solutions in both spatial and temporal spaces in real-time. In [24], an efficient motion planning framework with trajectory optimization is presented. Firstly, the cost function is designed to find the best solution based on state lattices. Then, an iterative approach is used to optimize both the path and velocity of the trajectory. To address the trajectory generation problem in dynamic conditions, a trajectory planning approach is presented in [25], which considers the lateral and longitudinal movement in the Frenet framework. However, the computational cost of the sampling based method is an intractable problem with the increase in sampling points.

Based on the previously mentioned work, a spatiotemporal trajectory planning algorithm for autonomous vehicles in dynamic on-road scenarios is proposed. To address the swing problem of the planned trajectory resulting from the change in the motion of obstacles, this paper puts forward a multi-area sampling method. The novelty of this method lies in sampling sets of endpoints within a series of defined areas at regular time intervals along the road. Moreover, the corresponding optimization method using dynamic programming is introduced. This approach is capable of generating more plentiful trajectories with speed information for autonomous vehicles within the planning horizon. In terms of the accurate estimation of obstacle movement [26–28], the unnecessary maneuver can be avoided, e.g., giving up overtaking when the obstacle vehicle in front changes lane. In addition, the offset of the trajectory polynomial can be relieved in this way so that the jerk-optimal trajectory is obtained. The multi-area sampling method provides rich

candidate trajectories, but also brings about computational load. To obtain the optimal solution and reduce the computational complexity, a cost function which considers safety and comfort is designed to evaluate the trajectories in each layer, and dynamic programming is introduced to calculate the total cost to find the best trajectory.

The remainder of this paper is organized as follows. Section 2 introduces the system architecture of the trajectory planning method. The multi-area sampling method is described in Section 3. In Section 4, different cases are performed to verify the effectiveness of the planning method. Finally, the conclusion is presented in Section 5.

2. System Architecture

The objective of this system is to generate a feasible trajectory for the vehicle to track in the dynamic environment. A general description of the trajectory planning framework for the multi-area sampling method is outlined in Figure 1. This system is mainly composed of trajectory generation and trajectory evaluation.

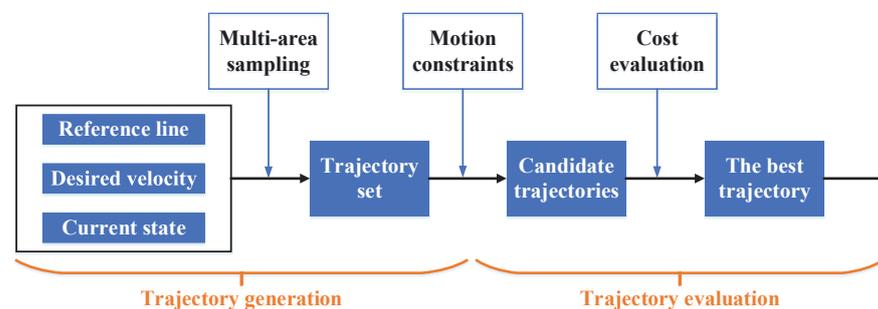


Figure 1. Frame diagram of the trajectory planning system.

According to the information of perception and decision, the necessary conditions including the reference line, desired velocity, and current state for trajectory generation are obtained. The novelty of this method is that it can sample numerous endpoints at fixed time intervals along the structured road. Considering the motion constraints of the vehicle, the trajectories which are beyond the maximum limit of velocity and acceleration will be eliminated from the trajectory set. Then, the feasible trajectories for autonomous vehicle are achieved, called candidate trajectories. More detailed information about multi-area sampling and trajectory generation is described in the next section. In order to obtain the best trajectory for tracking, the candidate trajectories need to be evaluated and optimized by the cost function which considers both safety and comfort. Then, the obtained trajectory is sent to the control module as the desired trajectory to track. More detailed information about trajectory evaluation and optimization is described in the next section.

3. Main Results

The trajectory generation strategy, which is composed of a multi-area sampling method, spatiotemporal trajectory generation, and motion constraints, is formulated in this section. For the on-road scenario, Cartesian coordinates will cause difficulties in dealing with the movement trajectory of the vehicle due to the curvy road. In this paper, Frenet coordinates are used instead of Cartesian coordinates, as shown in Figure 2. It is more convenient for us to use the distance along the road $s(t)$ and the lateral displacement with the reference line $l(t)$ to describe the vehicle's position on the road at time t . In this way, the curvy road is easily transformed into a straight road. The transformation relationships between the Frenet coordinates and Cartesian coordinates are described in [29].

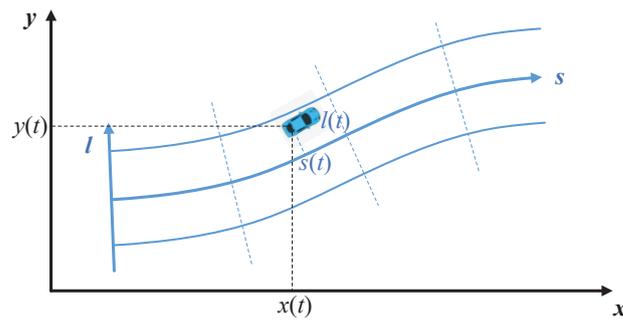


Figure 2. Frenet coordinates and Cartesian coordinates.

3.1. Multi-Area Sampling Method

To generate the trajectory set, the multi-area sampling method is proposed as illustrated in Figure 3. Different from the traditional method which chooses some terminal states on the road, we sample sets of endpoints in a series of defined areas at a fixed time interval T along the road. Each area is composed of a set of endpoints, where Δd is the distance of adjacent points along the reference line and Δl is the lateral offset between two points which is perpendicular to the reference line. The distance between each area Δs is defined as the length of the center position of each area, and it depends on the desired velocity v_{des} and the fixed time interval T :

$$\Delta s = v_{des} \cdot T \tag{1}$$

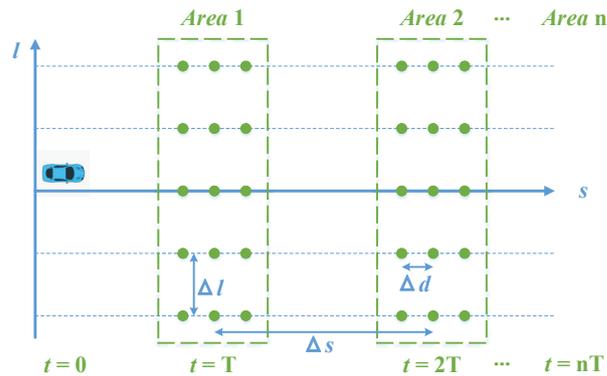


Figure 3. Multi-area sampling method.

Note that the local road is divided into n parts, and each part can generate a set of trajectories which are obtained by connecting the endpoints between adjacent areas. For the endpoints in each area, sampling along the direction of reference line will lead to different velocities due to the fixed time interval and different driving distances, and sampling in the direction perpendicular to the reference line will bring about lane change or obstacle avoidance. For the multi-area sampling on the road, it can describe more complex motion behaviors compared with the traditional sampling methods within the planning horizon. With this approach, the long-range trajectory planning is solved, and more abundant driving behaviors are provided for trajectory generation.

However, the multi-area sampling method can result in a complex computational problem, especially with the increment in areas on the road. In order to improve the computational efficiency and at the same time maintain the diversity of motion behaviors, the dynamic programming method is employed in this paper to tackle this problem. This topic is discussed in more detail in the next section.

3.2. Spatiotemporal Trajectory Generation

Through the connection of the endpoints in different areas, the trajectory set is constructed. In order to obtain the jerk-optimal trajectories as referred in [30], quintic polynomials are used to connect the two endpoints, where $P_{start} = [p_s, \dot{p}_s, \ddot{p}_s]$ is defined as the start state, and $P_{end} = [p_e, \dot{p}_e, \ddot{p}_e]$ is defined as the end state within the time interval $T = t_e - t_s$ in one dimension, and the function is given as

$$p(t) = \alpha_0 + \alpha_1 t + \alpha_2 t^2 + \alpha_3 t^3 + \alpha_4 t^4 + \alpha_5 t^5. \quad (2)$$

It should be noticed that the offset of the polynomial increases with time t , as shown in Figure 4. The different colors stand for the different extents of the lateral offset. Large offset will bring about the safety risk of driving, especially in lane change maneuvers or obstacle avoidance. Nevertheless, the long-range horizon should be considered in planning so that the autonomous vehicle can obtain high-quality trajectories to avoid unnecessary maneuvers on the road. In this paper, we can take advantage of the multi-area sampling method to solve this problem by selecting a suitable time interval T and the number of sampling areas n .

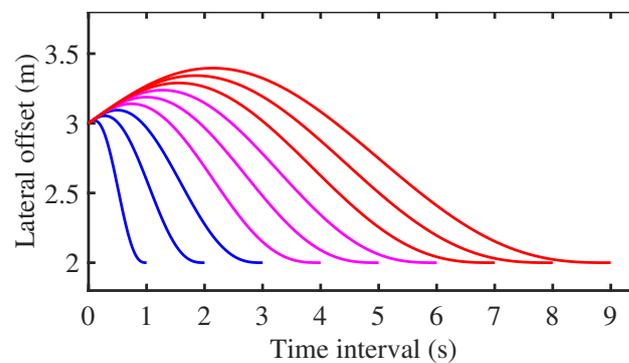


Figure 4. Lateral offset of quintic polynomial.

Since the spatiotemporal trajectory is divided into n parts, we define the state of the endpoint in area k as

$$X_k = [s_k, \dot{s}_k, \ddot{s}_k, l_k, \dot{l}_k, \ddot{l}_k, t_k]$$

where s_k is the distance along the road, l_k is the distance with the reference line, $t_k = kT$ is the time, $k = 1, 2, \dots, n$, and the initial state of the vehicle is X_0 . Then, the trajectory is determined by the endpoint sequence $\{X_0, X_1, \dots, X_n\}$.

In order to obtain the jerk-optimal trajectories, quintic polynomials are used to describe the curves. Because six parameters of the quintic polynomial are needed for calculation, six boundary conditions are used to solve this problem, namely, the initial position, velocity, and acceleration, and the terminal position, velocity, and acceleration. For the longitudinal movement, \dot{s}_k is set to a free variable to generate different velocities in response to complex situations so that the function can be changed into a quartic polynomial. Therefore, the spatiotemporal trajectory χ_k from area $k - 1$ to area k is represented as

$$\begin{aligned} s(t) &= \sum_{i=0}^4 a_i t^i \\ l(t) &= \sum_{i=0}^5 b_i t^i \end{aligned} \quad (3)$$

where a_i and b_i can be calculated by X_{k-1} and X_k . A trajectory set from one endpoint to the next area is generated as shown in Figure 5. The whole trajectory is $\Theta = \{\chi_1, \chi_2, \dots, \chi_n\}$.

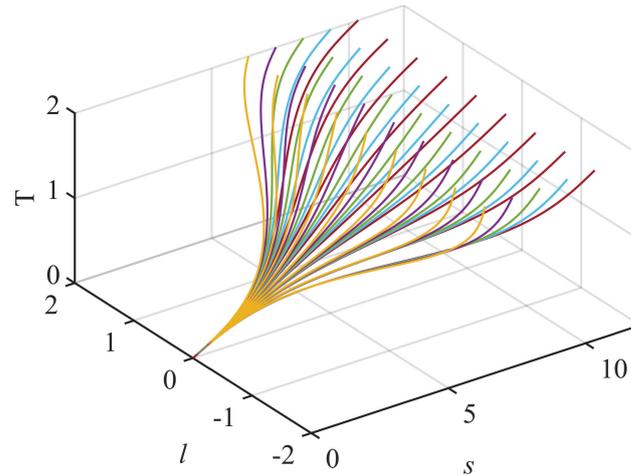


Figure 5. Trajectory generation from one point to next area.

3.3. Motion Constraints

The proposed trajectory generation method is just sampling along the road. Ignorance of the motion constraints will lead to invalid trajectory generation or even dangerous situation, especially for autonomous driving in the dynamic on-road scenarios. Therefore, the motion constraints must be considered so as to get rid of the unreasonable trajectories from the trajectory set.

Above all, the maximal speed is considered

$$v_{max} = \min\{v_{lim1}, v_{lim2}\} \quad (4)$$

where v_{lim1} is the speed limit of the road, and v_{lim2} is determined by

$$v_{lim2} \leq \sqrt{\frac{a_{lat}}{|\kappa|}} \quad (5)$$

where a_{lat} is the maximal lateral acceleration, and κ is the curvature value of the trajectory. The longitudinal acceleration \dot{v} is limited by the vehicle's physical constraints:

$$-a_{brake} \leq \dot{v} \leq a_{lon} \quad (6)$$

where a_{brake} is the maximal deceleration, and a_{lon} is the maximal longitudinal acceleration. In addition, the longitudinal and lateral accelerations should be limited to satisfy the circle of friction [31] which describes the safety region of the tire force:

$$a_{lon}^2 + a_{lat}^2 \leq \frac{\mu^2 F_z^2}{m^2} \quad (7)$$

where μ is the tire-road friction coefficient, F_z is the vertical load, and m is the mass of the vehicle.

3.4. Trajectory Evaluation

After the feasible trajectories are obtained, the procedure of evaluation and optimization is performed to select the best trajectory for the control layer to track. In this section, a cost function is designed to evaluate the trajectories in each area, and then the whole trajectory is optimized to find the minimum cost solution with dynamic programming. For the trajectory χ_k from area $k - 1$ to area k , the cost function J_k is defined as follows:

$$J_k(\chi_k) = \omega_1 J_{comf}(\chi_k) + \omega_2 J_{endp}(X_k) \quad (8)$$

where J_{comf} is a performance index which describes the driving comfort, J_{endp} is a term which represents the safety of the endpoint, and $\omega_i (i = 1, 2)$ is the corresponding weighted factor.

Owing to the human sensitivity to changes in acceleration, the driving comfort index J_{comf} is calculated as

$$J_{comf}(\chi_k) = c_1 \int_{t_{k-1}}^{t_k} \ddot{s}^2(\tau) d\tau + c_2 \int_{t_{k-1}}^{t_k} \ddot{l}^2(\tau) d\tau \quad (9)$$

where \ddot{s} and \ddot{l} are the longitudinal and lateral jerks, respectively, and c_1 and c_2 are the weighted factors.

With regard to the safety of the endpoint X_k , the cost function J_{endp} is composed of three parts: the distance with reference line in lateral direction, the distance with the desired position in longitudinal direction, and the distance to the obstacle at time t_k :

$$J_{endp}(X_k) = m_1 |l_k| + m_2 |s_k - s_{des}| + m_3 J_{obs} \quad (10)$$

where $m_i (i = 1, 2, 3)$ is the corresponding weighted factor. $s_{des} = v_{des} \cdot t_k$ is the desired position, and J_{obs} is expressed as follows:

$$J_{obs} = \begin{cases} e^{-\frac{|s_k - s_{obs}|}{\lambda}} & |l_k - l_{obs}| \leq l_{thre} \& |s_k - s_{obs}| \leq s_{thre} \\ 0 & otherwise \end{cases} \quad (11)$$

where λ is the bandwidth of the exponential function, (s_{obs}, l_{obs}) is the position of the obstacle in the Frenet frame, and s_{thre} and l_{thre} are the thresholds of penalty in the longitudinal and lateral directions.

By setting the weight coefficients, it is easily to realize that the distance factor between the ego vehicle and obstacle in the area of the obstacle potential field holds the dominant position in the cost function, and the comfort index works outside of that area.

3.5. Trajectory Optimization with Dynamic Programming

After evaluation, the cost of each trajectory in each area is obtained. The objective is to find the best trajectory Θ^* which consists of n segments $\{\chi_1^*, \chi_2^*, \dots, \chi_n^*\}$ to minimize the total cost function as follows:

$$\min J_{total}(\Theta) = \min \sum_{i=1}^n J_i(\chi_i) \quad (12)$$

where J_{total} is the total cost from the start position X_0 to the end position X_n .

Note that the trajectory optimization is converted into the multi-stage decision process, and the dynamic programming method is introduced to solve this problem. The cost function from area k to area n is defined as

$$V_k = \sum_{i=k+1}^n J_i(\chi_i) \quad (13)$$

where $V_0 = J_{total}$ represents the total cost function.

For the best trajectory $\{\chi_{k+1}^*, \chi_{k+2}^*, \dots, \chi_n^*\}$ from the endpoint X_k in area k to the terminal point X_n in area n , the cost function can be calculated by

$$V_k^* = \min V_k = \min (J_{k+1}(\chi_{k+1}) + V_{k+1}^*) \quad (14)$$

where V_{k+1}^* is the minimum cost from X_{k+1} to the end state X_n . It means that we can calculate the best solution from area k to area n by the minimum cost in area $k + 1$. Therefore, (12) is solved according to (8) and (14) as follows:

$$\begin{cases} V_0^* = \min J_{total}(\Theta) = \min(J_1(\chi_1) + V_1^*) \\ V_1^* = \min(J_2(\chi_2) + V_2^*) \\ \dots \\ V_{n-1}^* = \min(J_n(\chi_n) + V_n^*) \end{cases} \quad (15)$$

where $V_n^* = V_n = 0$ is the terminal cost in area n . Then, the optimal trajectory is obtained as

$$\Theta^* = \arg \min \sum_{i=1}^n J_i(\chi_i). \quad (16)$$

Note that the computational complex is mainly affected by the number of sampling areas, which bring abundant maneuvers for autonomous driving to adapt to complications. Therefore, a suitable number of sampling areas n should be selected in the planning horizon to trade off between the calculation efficiency and trajectory diversity.

3.6. Computational Complexity Analysis

The multi-area sampling method provides rich candidate trajectories but also brings computational burden. In fact, the computation of trajectory planning is mainly dependent on the calculation of the cost function for each trajectory. Therefore, the number of trajectories has an important impact on the computational burden. For the traversing search method, the number of trajectories N'_{traj} which is needed for calculation is as follows:

$$N'_{traj} = m^n, \quad (17)$$

where n is the number of sampling areas and m is the number of endpoints in each area. Obviously, the amount of calculation increases exponentially as the sampling area grows. In this work, the cost function of the trajectory in each area is calculated, and the dynamic programming method is used to solve the optimization problem. In this way, the number of trajectories N_{traj} which is needed to calculate is

$$N_{traj} = (n - 1) \cdot m^2 + m. \quad (18)$$

Figure 6 shows the changes in the computational burden in different parameter sets.

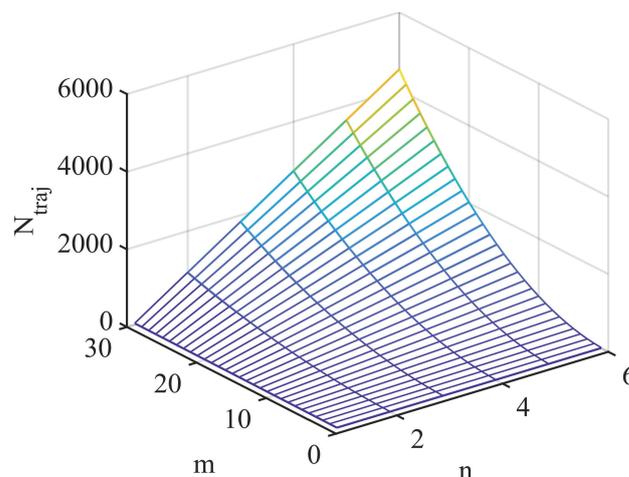


Figure 6. Computational complexity.

The method proposed in this work can generate rich spatiotemporal trajectories, which can provide a reasonable maneuver for obstacle avoidance. Meanwhile, the offset of the trajectory polynomial can be relieved. The multi-area sampling provides rich candidate trajectories but also brings computational burden. Note that the number of generated

trajectories affects the computational complexity because the cost function J_{total} is needed to evaluate for each trajectory. According to Equation (17), the computational complexity is mainly affected by the number of sampling areas n and endpoints m by the traversing search method. By contrast, the proposed method as shown in Equation (18) significantly reduces the computational complexity with the increasing of n . The problems such as combination explosion and calculation increasing as exponential type are avoided.

For the conventional method, the trajectory planning problem is decoupled into two parts: path planning and velocity planning. The path planning algorithm calculates a batch of feasible paths, and then the velocity planning algorithm calculates a reasonable speed for the optimal path which is selected from the candidate set based on the motion information of the obstacles. The conventional method brings convenience for the trajectory planning in the design process, but some limitations are unavoidable. On the one hand, the search algorithm or the optimization algorithm needs to be executed both in the path planning and the velocity planning, and it requires a significant amount of computing resources. On the other hand, the swing problem caused by the motion of obstacle vehicle becomes very difficult to be treated because the lane changing behavior is only implemented in the path planning part, which ignores the temporal feature.

The multi-area sampling method proposed in this paper is a spatiotemporal coupling approach, which is implemented in space and time simultaneously, and the multi-area indicates sampling at different time intervals. Consequently, the search algorithm or the optimization is executed only once per cycle, directly obtaining the desired trajectory that encompasses position and velocity information. Since sampling is carried out in the three-dimensional space–time domain, the future motion of obstacles can be easily considered during the planning process. Therefore, by taking into account the positional relationship between the ego vehicle and obstacles at different time points within the cost function, the swing problem is effectively alleviated.

4. Simulation Verification

To verify the efficiency of the proposed trajectory planning method, simulations with different cases are performed in the dynamic on-road scenarios based on two-lane roads. The parameters of the multi-area sampling method in the test are listed in Table 1, and other practical values for simulation are presented in Table 2.

Table 1. Parameters of the multi-area sampling method.

Symbol	Description	Value	Unit
v_{des}	desired velocity	30	km/h
n	number of sampling area	3	-
T	time interval of sampling	3	s
Δl	lateral offset of sampling points	0.5	m
Δd	longitudinal distance of sampling points	3	m

Table 2. Parameters of the simulation.

Symbol	Description	Value	Unit
v_{max}	maximal velocity	50	km/h
a_{brake}	maximal deceleration	3	m/s^2
a_{lon}	maximal longitudinal acceleration	2	m/s^2
a_{lat}	maximal lateral acceleration	2	m/s^2
R_{ego}/R_{obs}	radius of rough boundary of vehicle	3	m
r_{ego}/r_{obs}	radius of elaborate boundary of vehicle	1.5	m

The time interval T and the number of sampling area n are closely related to the computational complexity and the performance of the trajectory generation. Considering that the lateral offset is gradually increased with the time interval extension, as shown in

Figure 4, the time interval T should not be too large. In this paper, $T = 3$ is selected to alleviate the lateral offset of the quintic polynomial. Nevertheless, the planning horizon should be extended far enough to ensure safety along the road. The number of sampling area n can help us to extend the planning horizon. In this test, $n = 3$ is selected because two points are taken into account. First, the increase in sampling area n will bring computational burden in the planning process. Second, the sensing range of the perception system will limit the number of sampling areas.

A kinematic model of vehicle is introduced in this section to track the planned trajectory, which is formulated as follows:

$$\begin{cases} \dot{x}(t) = v(t) \cdot \cos \theta(t) \\ \dot{y}(t) = v(t) \cdot \sin \theta(t) \\ \dot{\theta}(t) = \frac{v(t)}{L} \cdot \tan \delta(t) \\ \dot{v}(t) = a(t) \end{cases} \quad (19)$$

where (x, y) is the position coordinate of the vehicle in the Cartesian frame, and v and a are the vehicle velocity and acceleration, respectively. θ is the orientation of the vehicle, L is the length of the wheel base, and δ is the front wheel steering angle. In this research, the PID controller is used to control the velocity in the longitudinal direction. In the lateral direction, the pure pursuit controller is introduced to track the obtained trajectory.

4.1. Case 1: Obstacle Avoidance by Lane Changing Behavior

In this scenario, the obstacle with a low speed (12 km/h) is moving along the road in front of the ego vehicle. The lane change maneuver is performed according to the planned trajectory to overtake the obstacle. Figure 7 shows the paths of the planned trajectory and obstacle in Cartesian coordinates, and the black curve is the reference line which represents the center line of the road in this paper. The velocity information and acceleration of the vehicle are shown in Figure 8. The movement of the ego vehicle is demonstrated in Figure 9, where the blue one is the ego vehicle and the green one is the obstacle vehicle.

From the simulation results, we can see that the planned trajectory avoids the dynamic obstacle in a smooth way, and the velocity of the vehicle is maintained at 30 km/h in the whole process. The trajectory tracking task is easily implemented by the pure pursuit controller in lateral control and PID controller in longitudinal control. The acceleration of the vehicle is smooth, which means that jerk is small, and therefore comfort is guaranteed.

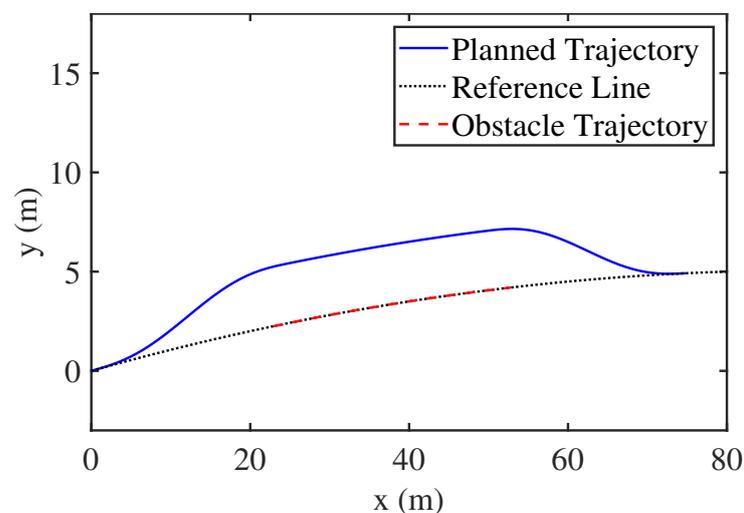


Figure 7. Paths in Cartesian coordinates in case 1.

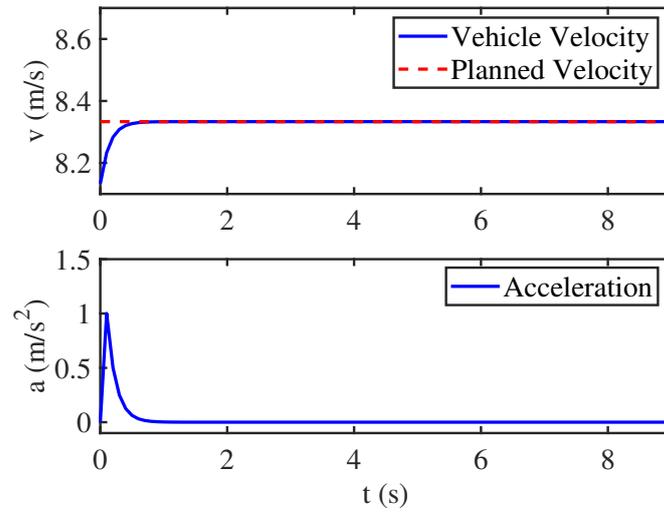


Figure 8. Velocity and acceleration in case 1.

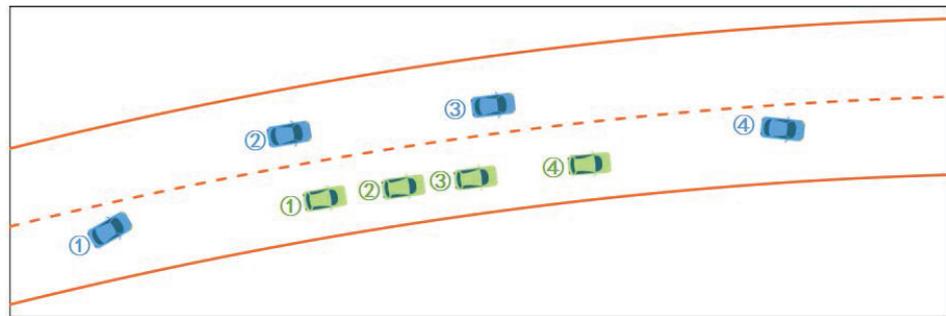


Figure 9. The obstacle avoidance by lane changing behavior.

4.2. Case 2: Obstacle Avoidance by Lane Keeping Behavior

In this case, the obstacle moves along the road at a low speed (12 km/h) at the initial stage, then it changes lane in a short time and drives on the new road. Figure 10 gives the paths of the planned trajectory and obstacle in Cartesian coordinates. In Figure 11, the velocity information and acceleration of the vehicle are illustrated. In Figure 12, the ego vehicle (blue car) with the proposed method overtakes the obstacle vehicle (green car) without a lane change maneuver. Therefore, the swing problem is mitigated.

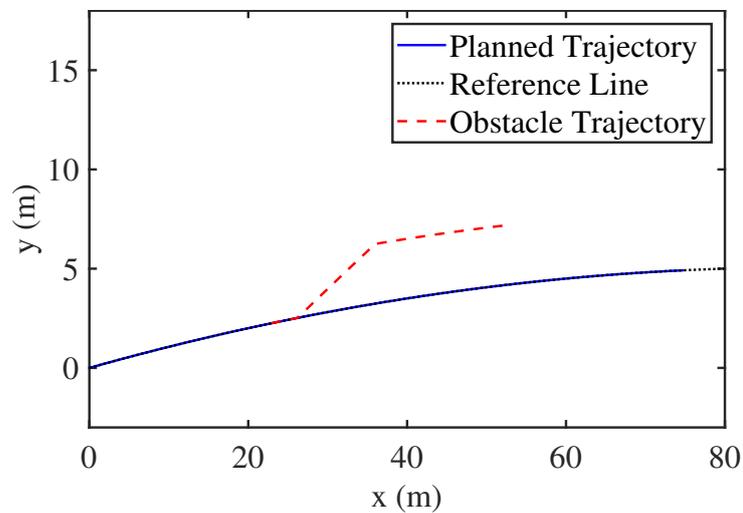


Figure 10. Paths in Cartesian coordinates in case 2.

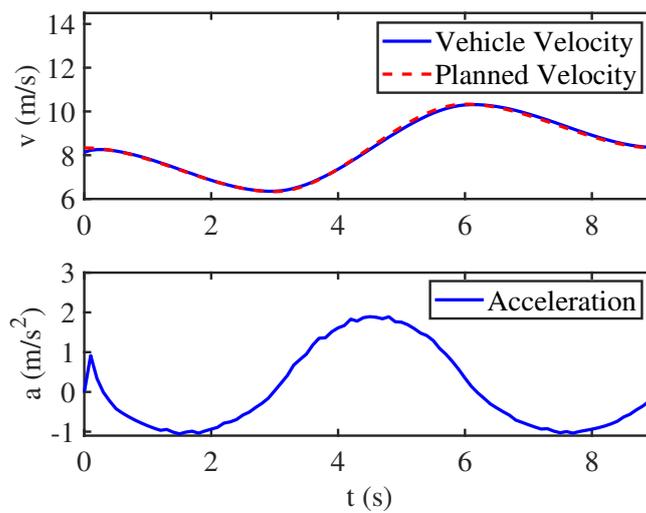


Figure 11. Velocity and acceleration in case 2.

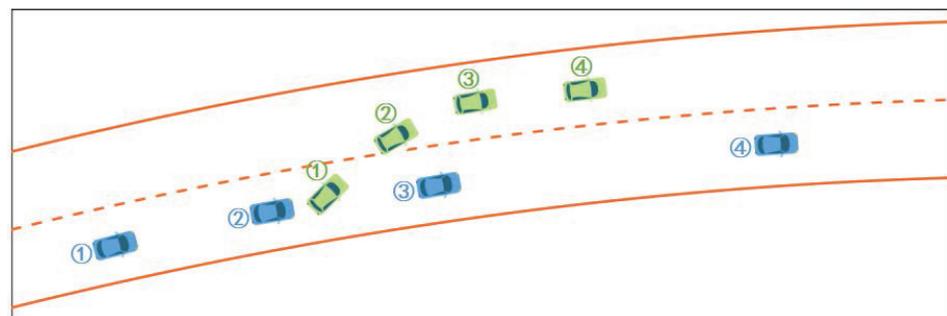


Figure 12. The swing problem is mitigated with the proposed method.

Compared with the overtaking behavior by the lane change maneuver in Case 1, the ego vehicle overtakes the obstacle by lane keeping in this scenario. Note that the velocity of the vehicle slows down during the initial stages, then the vehicle accelerates beyond 36 km/h. Finally, the speed is reduced to the desired velocity. It means that the ego vehicle avoids the low-speed obstacle by deceleration at the beginning, and then a brief acceleration occurs to make up for the deviation from the desired distance. The reason is that the motion of the obstacle is taken into account in the planning. After that, a suitable trajectory is generated based on the multi-area sampling method. In this way, the swing problem which caused by the change in the obstacle's movement is solved.

4.3. Obstacle Avoidance in Complex Conditions

To further verify the effectiveness of the proposed algorithm, simulation results in complex conditions are shown in Figures 13–16. In these figures, the blue car is our ego vehicle, and the others are the obstacle vehicles. Figure 13 shows the lane keeping scenario in the automatic driving. Owing to the obstacle vehicle in front driving at a high speed and having a safe distance, the ego vehicle keeps going along the road instead of overtaking. Figures 14 and 15 indicate that the ego vehicle drives on the crowded road. Based on the different position and the motion state of the obstacle vehicle, the ego vehicle changes lanes and overtakes obstacle vehicles at a high speed. Figure 16 illustrates that the ego vehicle overtakes obstacle vehicles under the condition that an obstacle vehicle changes lane at low speed. Considering the lane change maneuver of the obstacle in front, the ego vehicle keeps going straight the first time and then executes a lane change maneuver to avoid the obstacle vehicles on the crowded road.

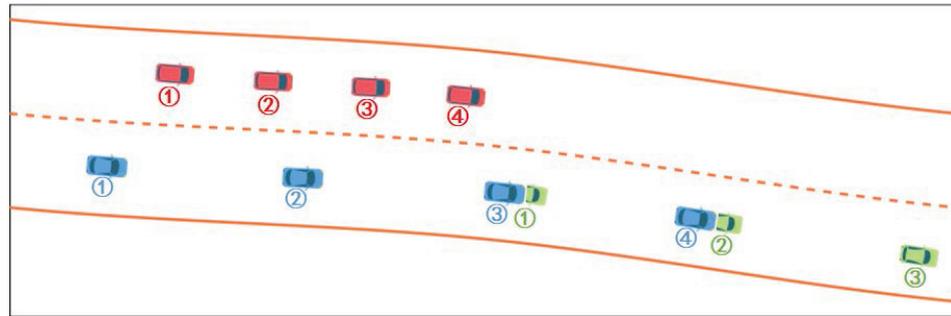


Figure 13. The lane keeping scenario.

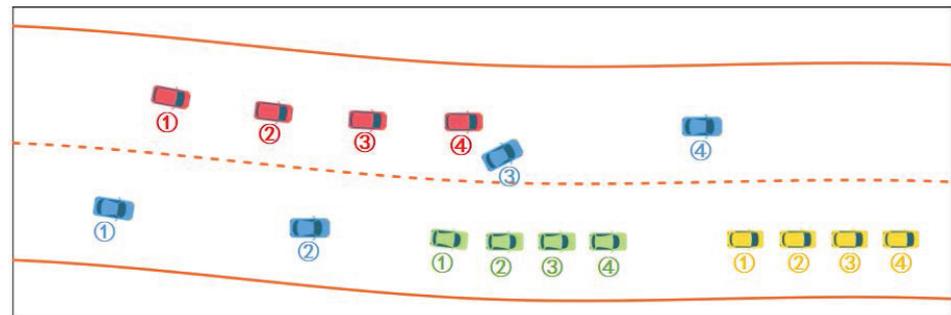


Figure 14. The lane change maneuver in a complex condition.

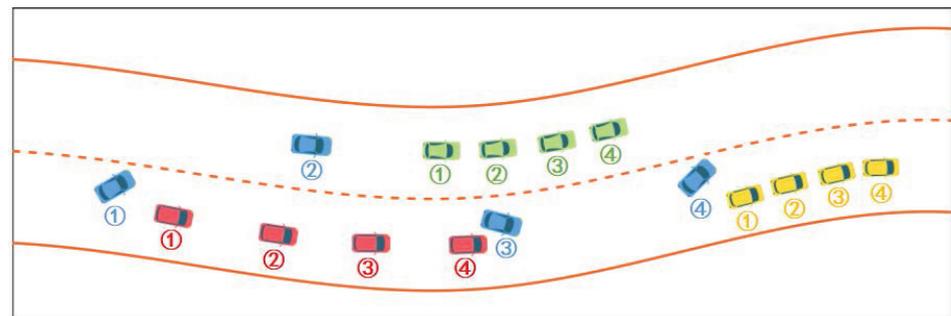


Figure 15. Continuous lane change maneuvers in a complex condition.

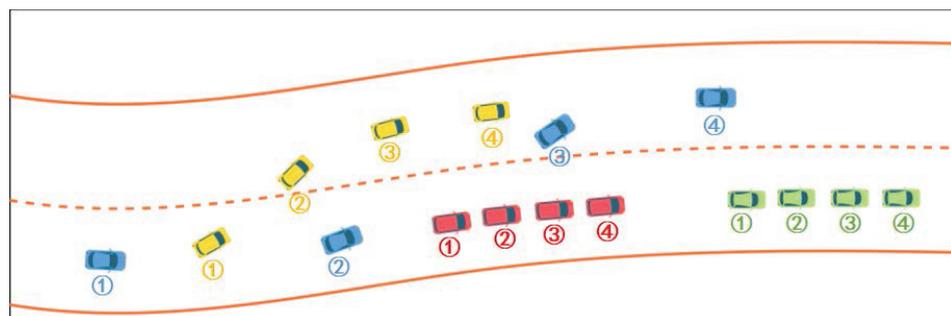


Figure 16. The lane change maneuver under the condition that the obstacle changes lane at a low speed.

5. Conclusions

This paper presents a multi-area sampling-based spatiotemporal trajectory planning method for autonomous vehicles to deal with the on-road driving problems in dynamic environments. Based on the multi-area sampling method, the structured road in the planning horizon is divided into different areas. The cost function which considering safety and comfort is designed for the trajectories in each layer. To calculate the best trajectory and at the same time reduce the computational load, dynamic programming is introduced

to solve these problems. The swing problem of the planned trajectory in dynamic scenario is solved by the proposed trajectory planning framework, and the safety and comfort are guaranteed simultaneously. The effectiveness of the proposed planning method is demonstrated via tests in different on-road scenarios. With the mature perception and control technologies of the autonomous driving system, the trajectory planning method proposed in this manuscript can be effectively deployed on intelligent vehicles. However, how to accurately predict the movement of obstacles has a significant impact on the proposed trajectory planning method. Although some results have been made in this field, trajectory prediction under complex conditions remains an open problem. For future work, a high-performance prediction algorithm would be further explored.

Author Contributions: Conceptualization, S.M. and L.W.; methodology, Z.N.; software, P.C.; validation, S.M., Z.N. and L.W.; formal analysis, S.M.; investigation, L.W.; resources, Z.N.; data curation, P.C.; writing—original draft preparation, S.M.; writing—review and editing, P.C.; visualization, Z.N.; supervision, L.W.; project administration, S.M.; funding acquisition, S.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Scientific and Technological Innovation Programs of Higher Education Institutions in Shanxi (Grant 2022L596).

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

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

References

1. Katakazas, C.; Quddus, M.; Chen, W.H.; Deka, L. Real-time motion planning methods for autonomous on-road driving: State-of-the-art and future research directions. *Transp. Res. Part* **2015**, *60*, 416–442. [[CrossRef](#)]
2. Huang, T.; Pan, H.; Sun, W.; Gao, H. Sine Resistance Network-Based Motion Planning Approach for Autonomous Electric Vehicles in Dynamic Environments. *IEEE Trans. Transp. Electrification* **2022**, *8*, 2862–2873. [[CrossRef](#)]
3. Gonzalez, D.; Prez, J.; Milans, V.; Nashashibi, F. A review of motion planning techniques for automated vehicles. *IEEE Trans. Intell. Transp. Syst.* **2016**, *17*, 1135–1145. [[CrossRef](#)]
4. Sun, W.; Wang, X.; Zhang, C. A model-free control strategy for vehicle lateral stability with adaptive dynamic programming. *IEEE Trans. Ind. Electron.* **2020**, *67*, 10693–10701. [[CrossRef](#)]
5. Dixit, S.; Fallah, S.; Montanaro, U.; Dianati, M.; Stevens, A.; McCullough, F.; Mouzakitis, A. Trajectory planning and tracking for autonomous overtaking: State-of-the-art and future prospects. *Annu. Rev. Control* **2018**, *45*, 76–86. [[CrossRef](#)]
6. Wei, J.; Snider, J.M.; Kim, J.; Dolan, J.M.; Rajkumar, R.; Litkouhi, B. Towards a viable autonomous driving research platform. In Proceedings of the 2013 IEEE Intelligent Vehicles Symposium (IV), Gold Coast, QLD, Australia, 23–26 June 2013; pp. 763–770.
7. Glaser, S.; Vanholme, B.; Mammar, S.; Gruyer, D.; Nouveliere, L. Maneuver-based trajectory planning for highly autonomous vehicles on real road with traffic and driver interaction. *IEEE Trans. Intell. Transp. Syst.* **2010**, *11*, 589–606. [[CrossRef](#)]
8. Elbanhawi, M.; Simic, M. Sampling-based robot motion planning: A review. *IEEE Access* **2014**, *2*, 56–77. [[CrossRef](#)]
9. Schlechtriemen, J.; Wabersich, K.P.; Kuhnert, K.D. Wiggling through complex traffic: Planning trajectories constrained by predictions. In Proceedings of the 2016 IEEE Intelligent Vehicles Symposium (IV), Gothenburg, Sweden, 19–22 June 2016; pp. 1293–1300.
10. Cesari, G.; Schildbach, G.; Carvalho, A.; Borrelli, F. Scenario model predictive control for lane change assistance and autonomous driving on highways. *IEEE Intell. Transp. Syst. Mag.* **2017**, *9*, 23–35. [[CrossRef](#)]
11. Ji, J.; Khajepour, A.; Melek, W.W.; Huang, Y. Path planning and tracking for vehicle collision avoidance based on model predictive control with multiconstraints. *IEEE Trans. Veh. Technol.* **2017**, *66*, 952–964. [[CrossRef](#)]
12. Hu, X.; Chen, L.; Tang, B.; Cao, D.; He, H. Dynamic path planning for autonomous driving on various roads with avoidance of static and moving obstacles. *Mech. Syst. Signal Process.* **2018**, *100*, 482–500. [[CrossRef](#)]
13. Wang, Y.; Liu, Z.; Zuo, Z.; Li, Z.; Luo, X. Trajectory planning and safety assessment of autonomous vehicles based on motion prediction and model predictive control. *IEEE Trans. Veh. Technol.* **2019**, *68*, 8546–8556. [[CrossRef](#)]
14. Zhu, S.; Aksun-Guvenc, B. Trajectory planning of autonomous vehicles based on parameterized control optimization in dynamic on-road environments. *J. Intell. Robot. Syst.* **2020**, *100*, 1055–1067. [[CrossRef](#)]
15. Huang, Y.; Ding, H.; Zhang, Y.; Wang, H.; Cao, D.; Xu, N.; Hu, C. A motion planning and tracking framework for autonomous vehicles based on artificial potential field elaborated resistance network approach. *IEEE Trans. Ind. Electron.* **2020**, *67*, 376–1386. [[CrossRef](#)]

16. Dolgov, D.; Thrun, S.; Montemerlo, M.; Diebel, J. Path planning for autonomous vehicles in unknown semi-structured environments. *Int. J. Robot. Res.* **2010**, *29*, 485–501. [[CrossRef](#)]
17. Gu, T.; Dolan, J.M. On-road motion planning for autonomous vehicles. In Proceedings of the International Conference on Intelligent Robotics and Applications, Montreal, QC, Canada, 3–5 October 2012; pp. 588–597.
18. Ziegler, J.; Bender, P.; Dang, T.; Stiller, C. Trajectory planning for berth: A local, continuous method. In Proceedings of the 2014 IEEE Intelligent Vehicles Symposium Proceedings, Dearborn, MI, USA, 8–11 June 2014; pp. 450–457.
19. Kuwata, Y.; Teo, J.; Fiore, G.; Karaman, S.; Frazzoli, E.; How, J.P. Real-time motion planning with applications to autonomous urban driving. *IEEE Trans. Control Syst. Technol.* **2009**, *17*, 1105–1118. [[CrossRef](#)]
20. hwan Jeon, J.; Cowlagi, R.V.; Peters, S.C.; Karaman, S.; Frazzoli, E.; Tsiotras, P.; Iagnemma, K. Optimal motion planning with the half-car dynamical model for autonomous high-speed driving. In Proceedings of the 2013 American Control Conference, Washington, DC, USA, 17–19 June 2013; pp. 188–193.
21. Howard, T.M.; Green, C.J.; Kelly, A.; Ferguson, D. State space sampling of feasible motions for high performance mobile robot navigation in complex environments. *J. Field Robot.* **2008**, *25*, 325–345. [[CrossRef](#)]
22. Ziegler, J.; Stiller, C. Spatiotemporal state lattices for fast trajectory planning in dynamic on-road driving scenarios. In Proceedings of the 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, St. Louis, MO, USA, 10–15 October 2009; pp. 1879–1884.
23. McNaughton, M.; Urmson, C.; Dolan, J.M.; Lee, J. Motion planning for autonomous driving with a conformal spatiotemporal lattice. In Proceedings of the 2011 IEEE International Conference on Robotics and Automation, Shanghai, China, 9–13 May 2011; pp. 4889–4895.
24. Xu, W.; Wei, J.; Dolan, J.M.; Zhao, H.; Zha, H. A real-time motion planner with trajectory optimization for autonomous vehicles. In Proceedings of the 2012 IEEE International Conference on Robotics and Automation, Saint Paul, MN, USA, 14–18 May 2012; pp. 2061–2067.
25. Werling, M.; Kammel, S.; Ziegler, J.; Groell, L. Optimal trajectories for time-critical street scenarios using discretized terminal manifolds. *Int. J. Robot. Res.* **2012**, *31*, 346–359. [[CrossRef](#)]
26. Geng, X.; Liang, H.; Yu, B.; Zhao, P.; He, L.; Huang, R. A Scenario-Adaptive Driving Behavior Prediction Approach to Urban Autonomous Driving. *Appl. Sci.* **2017**, *7*, 426. [[CrossRef](#)]
27. Luo, W.; Yang, B.; Urtasun, R. Fast and Furious: Real Time End-to-End 3D Detection, Tracking and Motion Forecasting with a Single Convolutional Net. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018.
28. Mohajerin, N.; Rohani, M. Multi-Step Prediction of Occupancy Grid Maps With Recurrent Neural Networks. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019.
29. Werling, M.; Ziegler, J.; Kammel, S.; Thrun, S. Optimal trajectory generation for dynamic street scenarios in a frenet frame. In Proceedings of the IEEE International Conference on Robotics & Automation, Anchorage, AK, USA, 3–7 May 2010; pp. 987–993.
30. Takahashi, A.; Hongo, T.; Ninomiya, Y.; Sugimoto, G. Local path planning and motion control for agv in positioning. In Proceedings of the IEEE/RSJ International Workshop on Intelligent Robots & Systems (IROS 89) the Autonomous Mobile Robots & Its Applications, Tsukuba, Japan, 4–6 September 1989; pp. 392–397.
31. Zhang, J.; Sun, W.; Du, H. Integrated motion control scheme for four-wheel-independent vehicles considering critical conditions. *IEEE Trans. Veh. Technol.* **2019**, *68*, 7488–7497. [[CrossRef](#)]

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