2022-01-0085 Published 29 Mar 2022

Path Planning Method for Perpendicular Parking Based on Vehicle Kinematics Model Using MPC Optimization

Zhuoren Li, Lu Xiong, Bo Leng, and Zhiqiang Fu Tongji University

Dequan Zeng and Yiming Hu East China Jiaotong University

Mingzhi Wu Nanchang Automotive Institute of Intelligence & New Energy

Citation: Li, Z., Xiong, L., Leng, B., Fu, Z. et al., "Path Planning Method for Perpendicular Parking Based on Vehicle Kinematics Model Using MPC Optimization," SAE Technical Paper 2022-01-0085, 2022, doi:10.4271/2022-01-0085.

Received: 27 Jan 2022 Revised: 27 Jan 2022 Accepted: 27 Jan 2022

Abstract

In recent years, intelligent driving technology is being extensively studied. This paper proposes a path planning method for perpendicular parking based on vehicle kinematics model using MPC optimization, which aims to solve the perpendicular parking task. Firstly, in the case of any initial position and orientation of the vehicle, judging whether the vehicle can be parked at one step according to the location of the parking place and the width of the lane, and then calculating the starting position for parking, and use the Bezier curve to connect the initial position and the starting position. Secondly, reference parking path is calculated according to the collision constraints of the parking space. Finally, because the parking path based on the vehicle kinematics model is composed of circle arcs and straight lines, the curvature of the path is discontinuous. The reference parking path is optimized using Model Predictive Control (MPC). The final path is easier to be followed, and thereby improving the accuracy of parking results. Several simulations and experiments were carried out for the scenarios of one step parking and multiple step parking respectively. The simulation results verify the effectiveness and real-time performance of the proposed method. The parking path is smooth and the vehicle can effectively follow it, thus the vehicle can be accurately parked at the desired location.

Introduction

With the development of autonomous driving technology, intelligent vehicle is envisaged as a promising technology. In recent years, autonomous driving technologies for various designated scenarios have gradually begun to be implemented. Among them, the automatic parking system is regarded as one of the most promising technologies, which can not only help people complete parking, but also make the parking process safer and more convenient. Thus, it has become a hot spot in academic and industry.

The path planning problem of automatic parking has been extensively studied. The pioneering motion planning methods are undoubtedly the Dubins curve method [1] and the RS curve method [2]. On this basis, a large number of scholars have applied the RS curve to the path planning research of automatic parking, and proposed a series of path planning methods based on geometric curves. The current common parking planning methods include traditional planning algorithms based on geometric curves or search algorithm, and learning-based algorithms using data training, such as reinforcement learning. Min et al. [3] search the global path by A* algorithm and used arc curves with different radii to adjust the parking path. Klaudt et al. [4] used the hybrid A* algorithm to search the entire driving and parking path. Kwonh et al [5] used clothoids with linear changes in curvature to plan, which makes the path more consistent with the actual driving process of the vehicle. Sui [6] uses circle as the parking path, and then smooths it through clothoids. Young [7] uses circle and straight lines to form a path, construct a nonlinear optimization problem with collision constraints of parking space boundary to solve the optimal path. Zou [8] proposed a reverse planning method started from the endpoint of parking, and then combined the geometric curve method with state space sampling method to generate a smooth parking area, which realize the dynamic adjustment of the parking arc trajectory. Zhang [9] proposed a model-based reinforcement learning method to learn parking strategies by performing data generation, data evaluation, and training the network.

Based on the existing parking planning algorithm, this paper proposes a path planning strategy for perpendicular parking that combines the geometric curve method and the optimization method. The planned path conforms to the driver's habits, has smooth curvature, and is easy to track. At the same time, this method required small computing resources. Firstly, the minimum road width required for one-step parking is calculated by considering the geometric position relationship between the initial position of the vehicle and the parking slot. Secondly, the parking path is calculated according to the collision constraint of the parking slot. Finally, we establish the vehicle kinematics error model, and use the MPC algorithm to optimize the parking path. The planned trajectory has continuous curvature and continuous speed, which is easy to track and control. In Sec. II, we describe the problem for parking path planning. Path optimization using MPC is discussion in Sec. III. Sec. IV are the simulation results and finally, Sec. IV provides the conclusion.

Parking Path Planning

In this section, we use circles and lines to compose a parking reference path based on the vehicle kinematics model, which is similar to the parking path during human operation. The rectangle is used instead of the vehicle.

The simplified model is shown in Fig. 1, in which W represent the maximum width of the vehicle, l_r is the distance from the rear axle to the rear of the car, l_f is the distance from the front axle to the front of the car, and L is the wheelbase.

Considering the low-speed characteristics of parking process, the movement of the vehicle satisfies the following three assumptions: (1) the wheel is in point contact with the ground; (2) the contact point between the tire and the ground rolls without relative sliding; (3) the vehicle only moves on a plane. At this time, the vehicle motion model is shown in Fig. 2 where x, y represents the horizontal and vertical coordinates of the vehicle position, θ is the current heading of the vehicle, v is the vehicle longitudinal speed and δ is the front wheel angle.

Analysis of Perpendicular Parking Process

The process of perpendicular parking is shown in Fig. 3., where p_1, p_2, p_3, p_4 are four corner points of the parking slot. The usual perpendicular parking path consists of an arc and a straight line.

However, when the width of the road where the vehicle is located is too narrow, the vehicle does not have enough space to drive into the parking space through a circular arc path. At this time, the vehicle needs multiple arc paths to adjust its position until it can drive into the parking space. The road constraint length required for one-step parking is,

$$l_{o1\_l1} = R - \frac{W}{2} - L_{safe1} \tag{1}$$

$$W_d = R_{\max} - l_{o1\_l1} \sin\alpha \tag{2}$$

$$R_{\max} = \sqrt{\left(R + \frac{W}{2}\right)^2 + \left(L + l_f\right)^2} \tag{3}$$

$$\alpha = \arccos\frac{R - \frac{W_l}{2}}{l_{o1\_p1}} \tag{4}$$

FIGURE 1 Simplified vehicle model.

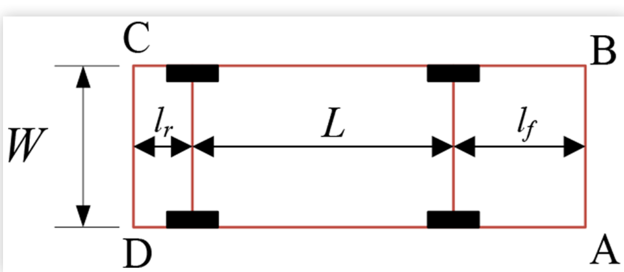


FIGURE 2 vehicle kinematics mode.

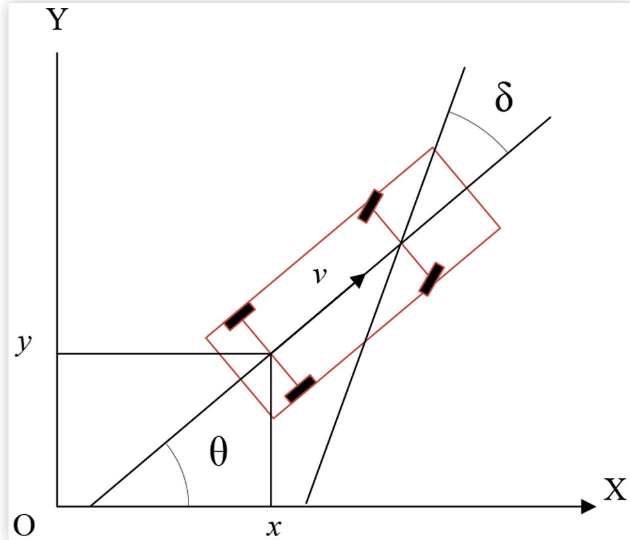


FIGURE 3 perpendicular parking process.

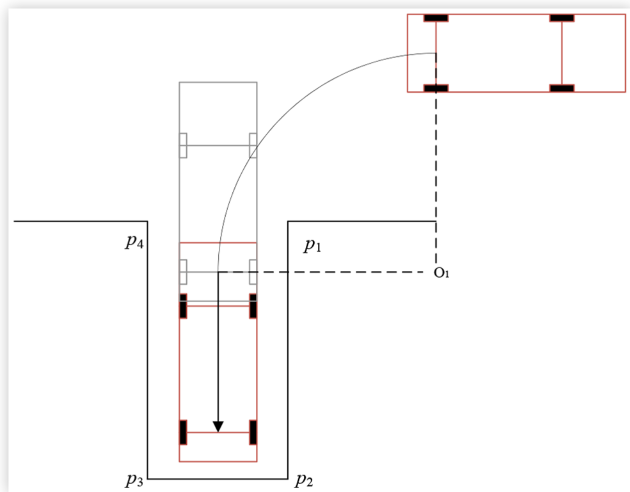
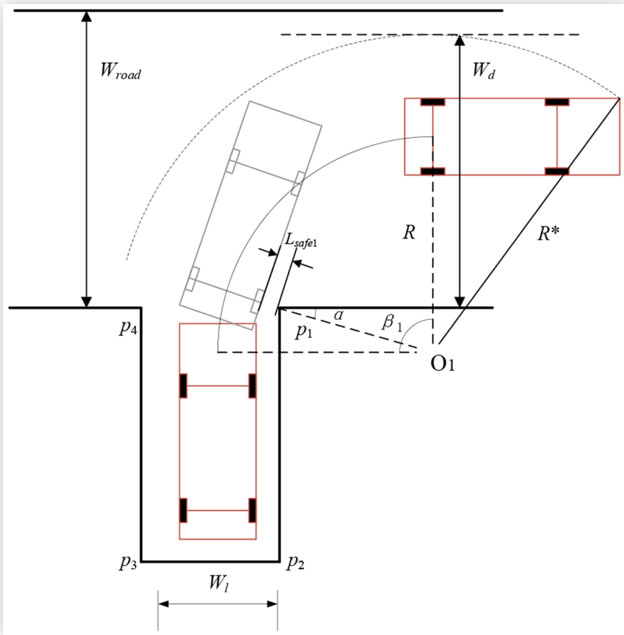


FIGURE 4 constraints for one-step parking.



where, O_1 is the center of the first segment of the parking arc path, $R_{\max}$ is the distance from the outer corner of the vehicle head to the center O_1, R is the desired turning radius for parking path planning, here is set to 110% of the minimum turning radius, l_{o1_p1} is the distance from the center O_1 to the storage corner point p_1, W_l is the width of parking slot, and L_{safe1} is a minimum safety threshold. When the road width W_{road} in the current environment is longer than W_d, the vehicle can be parked into the storage space in one step, otherwise, multiple steps are required.

One-step Parking

When the vehicle can be parked at one step, the coordinates of parking arc path center O_1 is calculated according to the safety constraints of the parking slot. And then the coordinates of the parking start point S_1 is calculated based on O_1. The cubic Bezier curve[10] is used to connect the initial vehicle position point *start* to the point S_1, where (x_i, y_i) is the coordinates of corner point l_i given by the slot detection system [11].

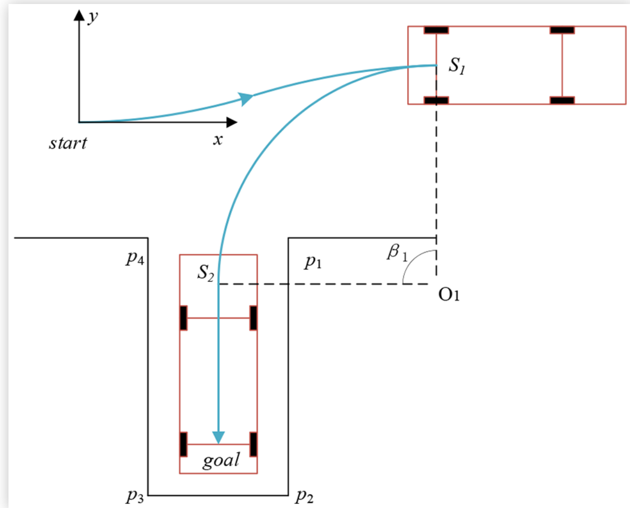
$$\begin{cases} x_{o1} = x_0 + l_{o1\_p1}\cos\alpha \\ y_{o1} = y_0 - l_{o1\_p1}\sin\alpha \end{cases} \tag{5}$$

and then the coordinates of parking start point S_1 is calculated as,

$$\begin{cases} x_{S1} = x_{o1} \\ y_{S1} = y_{o1} + R \end{cases} \tag{6}$$

β_1 is the central angle corresponding to the arc curve "S_1-S_2". When the vehicle reaches S2, it is connected the to the target position through a straight line "S_2-*goal*". Curve "*start*→S_1→S_2→*goal*" constitutes the reference parking path as shown in Fig. 5.

FIGURE 5 reference parking path of one-step parking.



Multi-step Parking

When the vehicle does not meet the requirements for one-step parking, multiple step parking paths need to be planned. In this paper, 3-segment parking method is adopted using three-segment arc path to adjust the position of the vehicle into the parking slot.

As shown in Fig. 6, firstly, let $W_d = W_{road} - L_{safe3}$, L_{safe3} is also minimum safety threshold to prevent the vehicle from being too close to the road boundary. And Then calculating the center O_1 of the first arc path according to the parking target position and the safety constraints of the parking slot,

$$\alpha = \arcsin\frac{R_{\max} - W_d}{l_{o1\_p1}} \tag{7}$$

$$\begin{cases} x_{o1} = x_0 + l_{o1\_p1}\cos\alpha \\ y_{o1} = y_0 - l_{o1\_p1}\sin\alpha \end{cases} \tag{8}$$

where, l_{o1_p1} is the distance between corner points p1 and p2.

The coordinates of the parking start point S1 are given in the same way as the one-stage parking, and then the central angle corresponding to each arc path are calculated by,

$$\begin{cases} \beta_1 = \gamma - \angle S_2O_1C \\ \beta_2 = \arcsin\left(\dfrac{R + \dfrac{W_l}{2} - l_{o1\_34}}{2R} + \sin\beta_1\right) - \beta_1 \\ \beta_3 = \dfrac{\pi}{2} - \beta_2 - \beta_1 \end{cases} \tag{9}$$

where,

$$\gamma = \arcsin\frac{l_{o1\_34} - L_{safe2}}{\sqrt{\left(R + \dfrac{W}{2}\right)^2 + lr^2}} \tag{10}$$

FIGURE 6 parking process of multi-step parking.

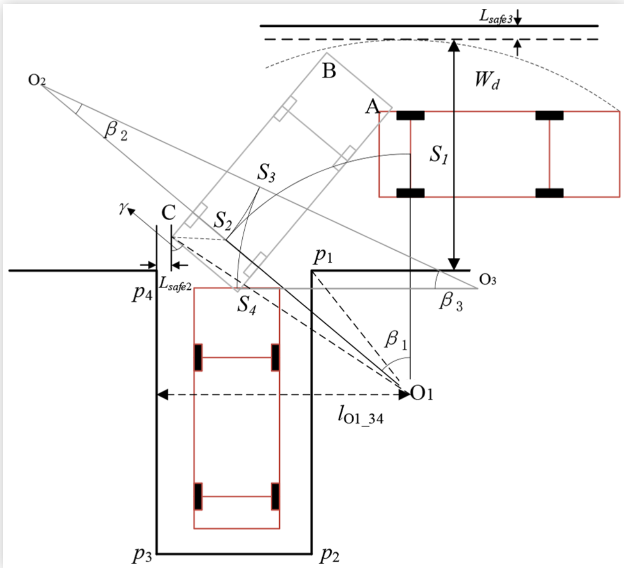
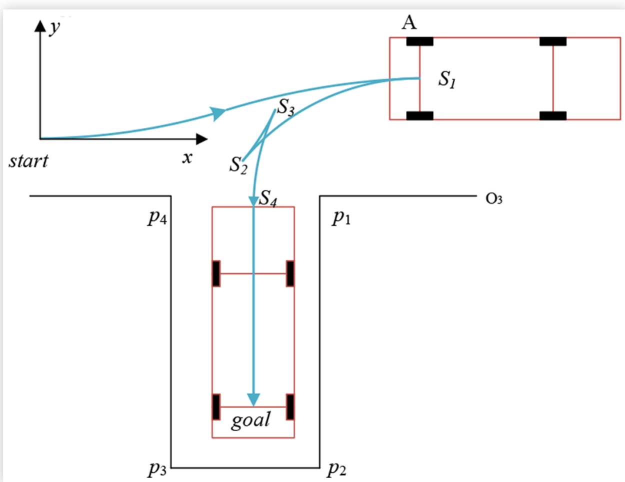


FIGURE 7 reference parking path of multi-step parking.



$$\angle S_2 O_1 C = \arccos \frac{2R+W}{\sqrt{(2R+W)^2 + 4lr^2}} \tag{11}$$

where L_{safe2} is a safety threshold to prevent the rear corner point of the vehicle from being too close to the lot boundary p_3-p_4, and $l_{o1\_34} = R + W_l/2$.

According to the three central angles β_1, β_2 and β_3, three circular arc paths S_1S_2, S_2S_3 and S_3S_4 are planned to ensure that the vehicle can reach the center line of the parking slot after driving through these paths. Finally, a straight line is used to make the vehicle reach the parking target position. The planned reference parking path for multi-step parking is shown in Fig. 7.

Optimized Path Planning

Linearization of Vehicle Kinematics Model

Through the planning algorithm in Section II, a reference parking trajectory can be obtained. However, the quality of the trajectory is poor in two parts: one is that the curvature of the trajectory segments traveling in the same direction is discontinuous; the second is that the actuator limit is not considered in reference to the vehicle speed and front wheel angle, and there are emergency stop and emergency steering operations. In this section, a lateral optimized path planner is introduced to calculate the optimal trajectory.

According to the vehicle kinematics mode in Section II, the state vector is set to $[x\, y\, \theta]^{\mathrm{T}}$ and the control vector to $[v\, w]^{\mathrm{T}}$. The kinematics model of the vehicle when parking is,

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos\theta \\ \sin\theta \\ 0 \end{bmatrix} v + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \omega = \begin{bmatrix} v\cos\theta \\ v\sin\theta \\ v\dfrac{\tan\delta}{L} \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix} \tag{12}$$

at the given reference trajectory point $\chi_r = [x_r\, y_r\, \theta_r]^{\mathrm{T}}$, the system error model can be obtained by performing a first-order Taylor expansion on the system,

$$\dot{\tilde{\chi}} = \begin{bmatrix} \dot{x} - \dot{x}_r \\ \dot{y} - \dot{y}_r \\ \dot{\theta} - \dot{\theta}_r \end{bmatrix} = \begin{bmatrix} 0 & 0 & -v_r \sin\theta_r \\ 0 & 0 & v_r \cos\theta_r \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x - x_r \\ y - y_r \\ \theta - \theta_r \end{bmatrix} + \begin{bmatrix} \cos\theta_r & 0 \\ \sin\theta_r & 0 \\ \dfrac{\tan\delta_r}{L} & \dfrac{v_r}{L\cos^2\delta_r} \end{bmatrix} \begin{bmatrix} v - v_r \\ \delta - \delta_r \end{bmatrix} \tag{13}$$

The equation (14) can be written after discretization as,

$$\tilde{\chi}(k+1) = A(k)\tilde{\chi}(k) + B(k)\tilde{u}(k) \tag{14}$$

where,

$$A(k) = \begin{bmatrix} 1 & 0 & -v_r(k)\cdot T\cdot \sin\theta_r(k) \\ 0 & 1 & v_r(k)\cdot T \cdot \cos\theta_r(k) \\ 0 & 0 & 1 \end{bmatrix} \tag{15}$$

$$B(k) = \begin{bmatrix} T\cdot\cos\theta_r(k) & 0 \\ T\cdot\sin\theta_r(k) & 0 \\ \dfrac{T\cdot\tan\delta_r(k)}{L} & \dfrac{T\cdot v_r(k)}{L\cos^2\delta_r(k)} \end{bmatrix} \tag{16}$$

where T is the sampling time. The reference speed $v_r(k)$ was set to 1m/s and steer angle $\delta_r(k)$ was obtained by,

$$\delta(k)=\arctan\frac{\left(\theta_r(k+1)-\theta_r(k)\right)L}{\sqrt{\left(x_r(k+1)-x_r(k)\right)^2+\left(y_r(k+1)-y_r(k)\right)^2}} \quad (17)$$

Optimization Problem Formulation

Let the control increment be,

$$\Delta u(k)=\tilde{u}(k)-\tilde{u}(k-1) \quad (18)$$

and let the state vector be,

$$\tilde{\xi}(k)=\begin{bmatrix}\tilde{\chi}(k+1)\\ \tilde{u}(k)\end{bmatrix} \quad (19)$$

To describe the system in unified state-space, the transfer matrix and state space function can be augmented as

$$\tilde{\xi}(k)=\begin{bmatrix}A(k) & B(k)\\ 0 & I_{2\times2}\end{bmatrix}\begin{bmatrix}\tilde{\chi}(k)\\ \tilde{u}(k-1)\end{bmatrix}+\begin{bmatrix}B(k)\\ I_{2\times2}\end{bmatrix}\Delta\tilde{u}(k) \quad (20)$$

Let the prediction horizon of the model predictive control be Np (≥1), and the control horizon size is Nc (Np≥Nc≥1). Then, the system state quantity in the prediction time domain is,

$$X_t=A_t\tilde{\xi}(k)+B_t\Delta U_t \quad (21)$$

where $X_t=\left[\tilde{\xi}(k+1)\ \tilde{\xi}(k+2)\cdots\tilde{\xi}(k+N_p)\right]^T_{1\times N_p}$ and $\Delta U_t=\left[\Delta u(k)\ \Delta u(k+1)\cdots\Delta u(k+N_c-1)\right]^T_{1\times N_c}$. The general MPC problem with constraints can be formulated as,

$$\begin{aligned}&\min: J=X_t^TQ_tX_t+\Delta U_t^TR_t\Delta U_t\\ &\text{subject to}:\begin{cases}\Delta u_{\min}\le\Delta u(k)\le\Delta u_{\max}\\ u_{\min}\le\tilde{u}(k)\le u_{\max}\\ \tilde{\xi}_{\min}\le\tilde{\xi}(k)\le\tilde{\xi}_{\max}\end{cases}\end{aligned} \quad (22)$$

where Q_t∈5×5 and R_t∈2×2 are the weighting matrices. $\Delta u_{\max}$ and $\Delta u_{\min}$ are limits of control increment vector, $u_{\max}$ and $u_{\min}$ are limits of control input vector, the $\tilde{\xi}_{\min}$ and $\tilde{\xi}_{\max}$ are limits of state vector. The objective of MPC problem here is to find an optimal trajectory to minimize its deviation from the reference path and control cost and subject to state constraints and control increment constraints.

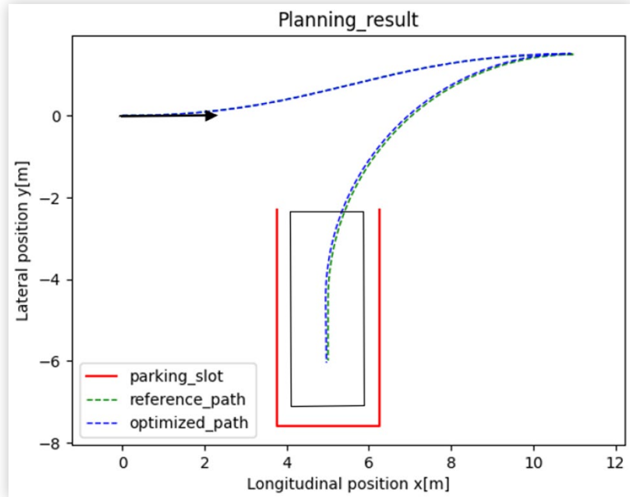
$$u^*(k)=u_r(k-1)+\tilde{u}(k-1)+\Delta U_t(1) \quad (23)$$

The obtained optimal control sequence $u^*(k)$ is substituted into the equation (11) and iteratively solved to update the pose of the vehicle. In this loop iteration, the optimal trajectory smoothed by the MPC can be obtained.

Simulation Results

In order to verify the effectiveness of the planning and control algorithms above, the simulation and experiment were designed respectively. Simulations were conducted in the C++ 11/Linux. OSQP SOLVER was used for solving MPC problem at each step. The parameters required by the MPC were selected as follows:

FIGURE 8 planning results of one-step parking.



Prediction and control horizon: $N_p = N_c = 20$, sampling time T = 0.1s, $\Delta u_{\max}$ =[1.0 0.6]^T, $\Delta u_{\min}$ =[−1.0 −0.6]^T, $u_{\max}$ = [2.5 0.6]^T, $u_{\min}$ = [−2.5 -0.6]^T, $\tilde{\xi}_{\min}$ = [−0.5 −0.5 −0.15]^T and $\tilde{\xi}_{\max}$ = [0.5 0.5 0.15]^T. Q_t = diag{10,10,25,0,25} and R_t = diag{2, 10}. The weight Q_t was chosen to prioritize heading(θ) and Steering angle inputs were also penalized more than acceleration in R_t to achieve smooth trajectories.

The simulation is conducted in a perpendicular parking scenario. The size of the parking slot is 5.3 meters in length and 2.5 meters in width. The length of the vehicle is 4.8 meters and width is 1.8 meters. The minimum turning radius is 5.5 meters. Taking the center of the rear axle of the vehicle as the origin of the coordinates, the longitudinal direction of the vehicle is the x direction, and the lateral direction is the y direction.

Fig. 8. shows the planning results of one-step parking scenario. In this scenario, the lateral distance from the initial position of the vehicle to the parking slot is 2.3 meters. The width of the road is 5 meters, which meets the requirements for one-step parking. The green dashed curve represents the reference path based on geometric curve planning, and the blue dashed curve represents the path optimized by MPC. The deviation (*dx*, *dy*, *dheading*) between the end of the optimized parking path and the end of the reference path is (0.03m, 0.05m, 0.36°), which meas 0.03 meters laterally and 0.05 meters longitudinally, and the heading deviation of the end point is only 0.36 degrees. For this deviation, it is still a good parking stop position. Fig. 9 is the curvature of the planned path. The green curve is the curvature of the path before optimization. It is shown that the path composed of geometric curves has obvious curvature change at the junction of the arc and the straight line. The blue curve is the curvature of the optimized path, and is continuous and smooth at the location where the curvature changes suddenly in green curve.

And Fig. 10 - Fig. 13 show the planning results of multiple step parking scenarios with different initial position of the vehicle. Same as one-step parking, Fig. 10 and Fig. 12 are the

FIGURE 9 curvature of the planned path of one-step parking.

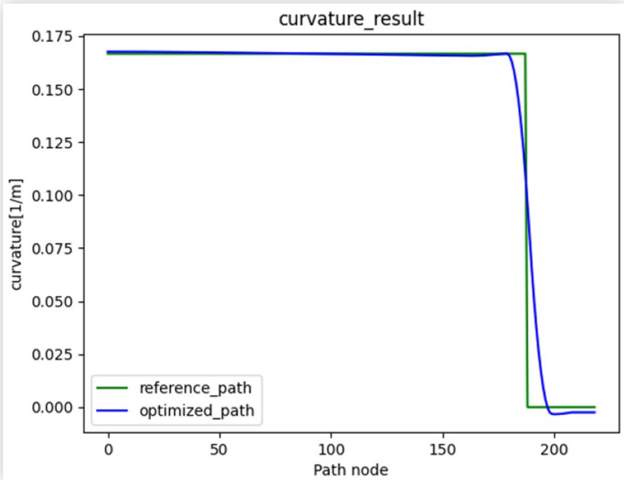


FIGURE 10 planning results of multi-step parking in scenario1.

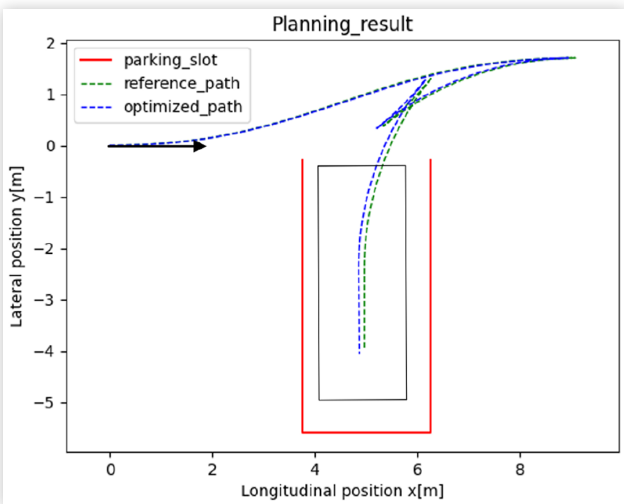


FIGURE 11 curvature of multi-step parking path in scenario1.

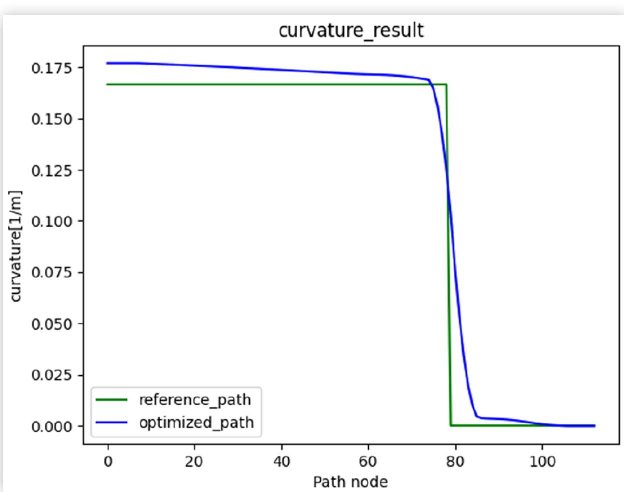


FIGURE 12 planning results of multi-step parking in scenario2.

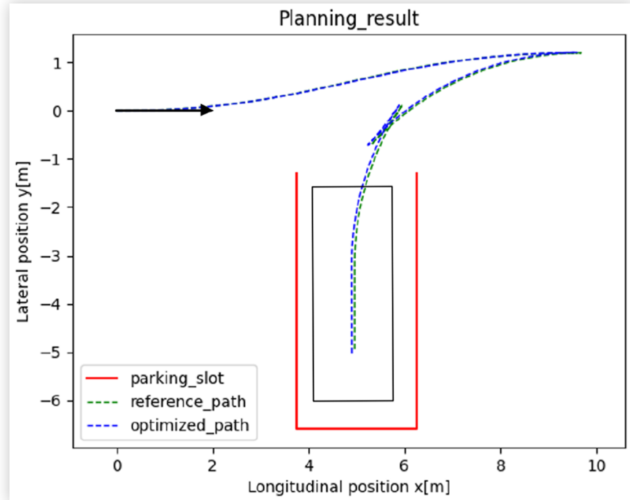
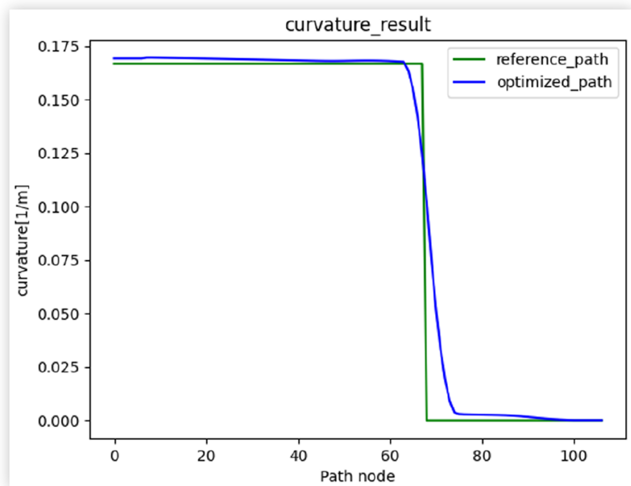


FIGURE 13 curvature of multi-step parking path in scenario2.



results of the parking path, and their deviation of end point are (0.07m, 0.08m, 0.65°) and (0.04m, 0.05m, 0.29°). Fig. 11 and Fig. 13 are corresponding curvature results. Compared with the green curvature curve, the blue curvature curve which represent the path optimized by MPC is smoother in the original two position where the reference paths have sudden curvature changes. Such a result will bring a smaller error when the vehicle is actually tracking the path, so as to achieve a good actual parking effect.

Conclusions

In this paper, we propose a path planning method that combines geometric curves and MPC optimization method for perpendicular parking scenario. According to the geometric relationship between the vehicle and the parking slot, a parking reference path consisting of Bezier curve, circles and line is designed. Based on the vehicle kinematics model, the MPC method is used to iteratively optimize the parking reference path. The effectiveness of the method is verified in one-step and multi-step parking scenario. The simulation results show that, compared with the parking path composed of pure geometric curves, the optimized path has a smoother curvature and does not produce sudden changes in curvature. In the future, we will add path tracking algorithms to supplement the corresponding experimental results based on our electric vehicle platform.

References

1. Dubins, L.E., "On Curves of Minimal Length with a Constraint on Average Curvature, and with Prescribed Initial and Terminal Positions and Tangents," *American Journal of Mathematics* 79, no. 3 (1957): 497-516.
2. Reeds, J.A. and Shepp, L.A., "Optimal Paths for a Car that Goes Both Forwards and Backwards," *Pacific Journal of Mathematics* 145, no. 2 (1990): 367-393.
3. Min, K. and Choi, J. "Design and Implementation of an Intelligent Vehicle System for Autonomous Valet Parking Service," in *2015 10th Asian Control Conference (ASCC)*, 2015, pp. 1-6, doi:10.1109/ASCC.2015.7244802.
4. Klaudt, S., Zlocki, A., and Eckstein, L., "A-Priori Map Information and Path Planning for Automated Valet-Parking," *2017 IEEE Intelligent Vehicles Symposium (IV)* (2017): 1770-1775.
5. Kwon, H. and Chung, W., "Performance Analysis of path Planners for Car-Like Vehicles Toward Automatic Parking Control," *Intelligent Service Robotics* 7, no. 1 (2014): 15-23.
6. Liyang, S., Yu, H., Xuezhi, C., Changhao, J., and Miaohua, H., "Path Planning Based on Clothoid for Autonomous Valet Parking," in *2020 17th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP)*, 2020, pp. 389-393.
7. Jeong, Y.W., Heo, H.W. and Chung, C.C., "A Local Path Planning for Perpendicular Parking in Limited Parking Spaces," in *2020 20th International Conference on Control, Automation and Systems (ICCAS)*, 2020, pp. 38-40, doi:10.23919/ICCAS50221.2020.9268290.
8. Zou, R., Wang, S., Wang, Z., Zhao, P., and Zhou, P., "A Reverse Planning Method of Autonomous Parking Path," in *2020 5th Asia-Pacific Conference on Intelligent Robot Systems (ACIRS)*, 2020, pp. 92-98, doi:10.1109/ACIRS49895.2020.9162616.
9. Zhang, J., Chen, H., Song, S., and Hu, F., "Reinforcement Learning-Based Motion Planning for Automatic Parking System," *IEEE Access* 8 (2020): 154485-154501.
10. Yang, K. and Sukkarieh, S., "3D Smooth Path Planning for a UAV in Cluttered Natural Environment," in: , *Intelligent Robots and Systems, Iros, IEEE/RSJ International Conference on*, (Nice: IEEE, 2008), 794-800.
11. Yu, Z., Gao, Z., Chen, H., and Huang, Y., "SPFCN: Select and Prune the Fully Convolutional Networks for Real-Time Parking Slot Detection," *IEEE Intelligent Vehicles Symposium (IV)* (2020, 2020): 445-450.

Contact Information

Zhuoren Li
School of Automotive Studies, Tongji University, Shanghai 201804, China. (zrli_96@163.com).

Acknowledgments

This work is supported by Shanghai Municipal Science and Technology Major Project 2021SHZDZX0100，the Fundamental Research Funds for the Central Universities and the Perspective Study Funding of Nanchang Automotive Institute of Intelligence and New Energy . Tongji University (TPD-TC202110-08).

ISSN 0148-7191