Design and implementation of path planning algorithm for vehicle parking

YANG Yi(杨毅), QU Xin(屈新), ZHU Hao(朱昊)✉, ZHANG Lu(张鲁), LI Xing-he(李星河)

(School of Automation, Beijing Institute of Technology, Beijing 100081, China)

Abstract: Parking is an important and indispensable skill for drivers. With rapid urban development, the automatic parking assistant system is one of the key components in future automobiles. Path planning is always essential for solving parking problems. In this paper, a path planning method is proposed for parking using straight lines and circular curves of different radius without collisions with obstacles. The parking process is divided into two steps in which the vehicle reaches the goal state through the intermediate state from the initial state. The intermediate state will be selected from the intermediate state set with a certain criterion in order to avoid obstacles. Similarly, an appropriate goal state will be selected based on the size of the parking lot. In addition, an automatic parking system is built, which effectively achieves the parking lot perception, path planning and performs parking processes in the environment with obstacles. The result of simulations and experiments demonstrates the feasibility and practicality of the proposed method and the automatic parking system.

Key words: path planning; autonomous parking system; autonomous vehicle platform

CLC number: TP 311 **Document code**: A **Article ID**: 1004-0579(2016)04-0502-10

Autonomous vehicle technology including driving assistant system has been extensively studied in recent years. The technology improves the comfort and reduces the maneuver effort for drivers. Parking is one of the important and indispensable skills for drivers which need many maneuvers with limited vision. Parking situation such as heavy traffic, tiny parking lot appears frequently with rapid urban development which increases the difficulty of the parking process. Besides, parking may cause traffic congestion and traffic accidents. Therefore, the automatic parking assistant system is one of the key components in future automobiles with prospects and market value.

Path planning as an essential component for the automatic parking assistant system has been widely studied. Generally, it can be classified into three categories:

Sampling based method has been developed for a long time including rapidly-exploring random tree (RRT)[1-3], RRG[4], path planning using Voronoi uncertainty fields[5] and swarm algorithm[6] etc. Sampling based method selects the point in the map with a certain criterion from the initial state to the goal state. However, a path with uncertain performance, uncertain computation time and little regard of the vehicle constraints is generated with this method.

Traversal method[7-8] is developed in recent years. It is aimed at planning all reasonable paths near the parking lot by iteration and selecting the optimal path among them. This method takes

Received 2015-06-18

Supported by the National Natural Science Foundation of China (61473042, 61105092, 61173076); Beijing Higher Education Young Elite Teacher Project (YETP1215)

✉ Author for correspondence

E-mail: frankzhu@bit.edu.cn

DOI: 10.15918/j.jbit1004-0579.201625.0408

comprehensive consideration to select the path of the global optimum from all reasonable paths. But the procedure is complex and costs long computation time.

The third method is related with parking process which divides the path planning into two steps. The vehicle reaches the goal state from the initial state through the intermediate state. The third method can be classified into two categories according to the different curves used for path planning: circular curves and other types of curves. Refs. [9 – 11] use the combination of straight lines and circular curves. Most of these themes aim at non-obstacle parking environment. Other types of curves include Bezier curve[12], circular curve interpolation[13] and 5th polynomial curve[14] etc. This method is easily coded and of high reliability. However, the current algorithms may have the following drawbacks: Few researchers make discussions on the path planning from the initial state to the intermediate state; Several algorithms take only circular curves of minimum radius for path planning or a single vehicle state as the intermediate state or the goal state.

Considering reliability and less computation time of the third method, a path planning algorithmis proposed for parking in this paper which could circumvent the drawbacks of the current algorithms. First an appropriate goal state will be selected based on the size of the parking lot, then the intermediate state set will be calculated and the appropriate intermediate state will be selected according to the initial state and the parking environment, finally the path will be planned. To implement the method, an automatic parking system has been built. Simulation results show that the algorithm succeeds to plan the parking path with the consideration of vehicle restraints and the environment with obstacles. Experiment results on the autonomous vehicle validate the effectiveness and practicability of the proposed algorithm and the automatic parking system.

The remainder of this paper is organized as follows: Section1 presents the basic equations used to model the dynamics of autonomous vehicles and provides a method to describe the path planning for parking. The map for parking path planning, the strategy to select the goal state, the intermediate state set and the path planning from the intermediate state to the goal state are described in section 2. Section 3 presents the strategy to choose the appropriate intermediate state and the path planning from the initial state to the intermediate state; Section 4 shows the automatic parking system and the results of simulations and experiments. Finally, in section 5 the conclusion is drawn and future work direction is presented.

1 Modeling and problem description

1.1 Dynamics of autonomous vehicle modeling

The vehicle mentioned in this paper is of the Ackermann steering structure. The Ackerman steering vehicle kinematic model is shown in Fig. 1. The vehicle rotates with a certain radius at the center O due to the angle between the front wheels and the direction of the vehicle. During the rotation of the vehicle, r_e, r_i, r and θ_{steer} are defined as shown in Fig. 1. The relationship between r_e, r_i, θ_{steer} and r is shown as

$$\begin{cases} r_i = r - d_{car}/2 \\ r_e = \sqrt{(r + d_{car}/2)^2 + (L_{car} - L_{rear})^2} \\ \theta_{steer} = \arctan((L_{car} - L_{front} - L_{rear})/r) \end{cases} \tag{1}$$

When $\theta_{steer} = \theta_{steer,max}$, the vehicle will rotate

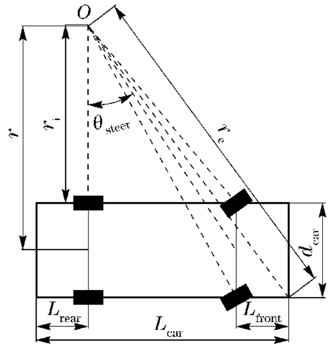


Fig. 1 Vehicle model with Ackerman steering structure

with the minimum radius due to vehicle constraints. Define r as $r_{\min}$ when the vehicle rotates with the minimum radius.

There have been many studies on the kinematic model of the vehicle. The kinematic model is shown as follows which implies that the trajectory of the vehicle is a circular curve:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{\theta}_{\text{steer}} \end{bmatrix} = \begin{bmatrix} \cos\theta \\ \sin\theta \\ \tan\theta_{\text{steer}}/(L_{\text{car}} - L_{\text{front}} - L_{\text{rear}}) \\ 0 \end{bmatrix} v + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} w \tag{2}$$

Define the coordinate of the midpoint of the rear axle as (x, y), the heading angle of the vehicle as θ. The speed of the vehicle is v and the angular velocity of θ_{steer} is w[15].

1.2 Parking path planning description

Define the vehicle state as (x, y, θ). Parking process is that steering the vehicle from the initial state to the goal state.

To solve the existing problems the current algorithms possess, the proposed method hasthe following features:

① To make it closer to the vehicle steering theory and easier to motion control, the combination of straight lines and circular curves with different radii satisfying $r \geqslant r_{\min}$ are used for path planning.

② The parking process keeps a larger distance between the vehicle and obstacles or the parking line markings with less maneuver and short travel distance.

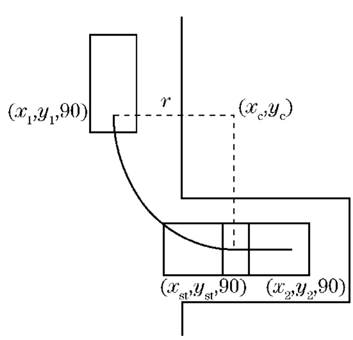
③ The intermediate state is selected according to the initial state and the parking environment. The goal state is selected based on the size of the parking lot.

④ The method has less computation time.

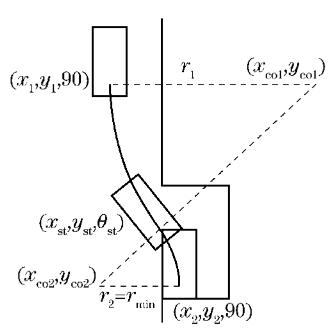
2 Goal state and the intermediate state set

When performing parking process, people are used to steering the vehicle to an appropriate intermediate state first, and then perform backward maneuver in one trial or repeated maneuvers in n trials until the vehicle reaches the goal state.

Define the vehicle state which could perform one trial parking process without the collision with the obstacles and the parking line markings as the intermediate state. Define all the intermediate state as the intermediate state set. As shown in Fig. 2, (x_1, y_1, θ_1) is an example of the intermediate state.



(a) Garage parking process



(b) Parallel parking

Fig. 2 Parking process

The intermediate state of the proposed algorithm is parallel to the curb. For garage parking, as shown in Fig. 2a, the vehicle reaches the goal state from the intermediate state with a circular trajectory. For parallel parking, as shown in Fig. 2b, the vehicle reaches the goal state from the intermediate state in one trial with the minimum radius during the second turning process.

Fig. 3 shows the map mentioned in this paper. It is required that $x_{\text{lot1}} = x_{\text{lot4}}$, $x_{\text{lot2}} = x_{\text{lot3}}$, $y_{\text{lot1}} = y_{\text{lot2}}$, $y_{\text{lot3}} = y_{\text{lot4}}$. The line connecting vertex 1 and vertex 4 is the entry of the parking lot. The vehicle is located on the left side of the parking lot.

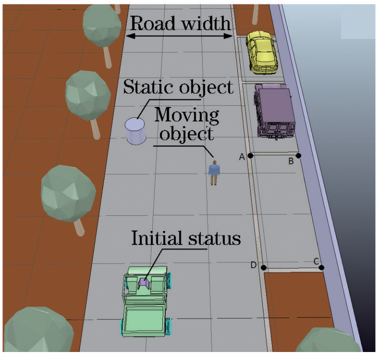


Fig. 3 Map mentioned in this paper

The algorithm proposed in this paper aims at path planning for one-trialparking. The minimum length $L_{\min}$ and width $d_{\min}$ of the parking lot for one trial parallel parking reaching the specific goal state is shown as follows

$$\begin{cases} L_{\min} = L_{\text{rear}} + \Delta L + \sqrt{r_e^2 - (r_i - \Delta d)^2} \\ d_{\min} = \sqrt{(r_{\min} + d_{\text{car}}/2)^2 + L_{\text{rear}}^2} - r_{\min} + d_{\text{car}}/2 + \Delta d \end{cases} \tag{3}$$

The parameters r_e and r_i in Eq. (3) are calculated by Eq. (1) under $r = r_{\min}$. The parameters ΔL and Δd are shown in Fig. 4. The minimum size of the parking lot can be calculated when $\Delta L = 0$, $\Delta d = 0$.

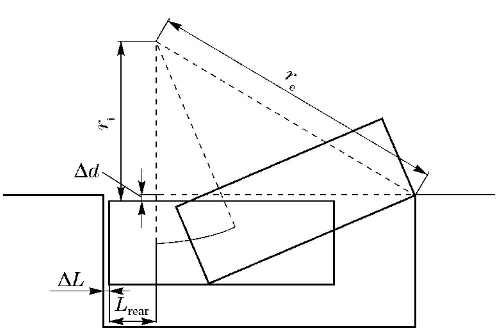


Fig. 4 Minimum length and width of the parking lot

Without loss of generality, the following discussion assumes that the parking lot is on the right of the vehicle. Then the goal state in the middle of the parking lot has priority for parallel parking which can be figured out by calculating ΔL in Eq. (3) on the condition of $\Delta d = (x_{\text{lot2}} - x_{\text{lot1}} - d_{\text{car}})/2$. The coordinates of the goal state can be figured out as shown as

$$\begin{cases} x_2 = (x_{\text{lot1}} + x_{\text{lot2}})/2 \\ y_2 = y_{\text{lot2}} - \sqrt{r_e^2 - (r_{\min} - (x_{\text{lot2}} - x_{\text{lot1}})/2)^2} \end{cases} \tag{4}$$

If the goal state collides with the parking lot, that is $y_2 < (y_{\text{lot3}} + L_{\text{rear}})$, then the goal state which is the most easily accessible with the coordinate shown as follows will be selected:

$$\begin{cases} x_2 = x_{\text{lot4}} + d_{\text{car}}/2 \\ y_2 = y_{\text{lot4}} + L_{\text{rear}} \end{cases} \tag{5}$$

For garage parking, the middle point of the parking lot will be selected as the goal state whose coordinate is shown as

$$\begin{cases} x_2 = (x_{\text{lot1}} + x_{\text{lot2}})/2 + L_{\text{car}}/2 - L_{\text{rear}} \\ y_2 = (y_{\text{lot1}} + y_{\text{lot4}})/2 \end{cases} \tag{6}$$

For parallel parking as shown in Fig. 2a, if the intermediate state $(x_1, y_1, 90)$ and the goal state $(x_2, y_2, 90)$ are already known, then the center of the first turning process $(x_{\text{co1}}, y_{\text{co1}})$ with radius r_1, the center of the second turning process $(x_{\text{co2}}, y_{\text{co2}})$ and the state $(x_{\text{st}}, y_{\text{st}}, \theta_{\text{st}})$ will be calculated using the following

$$x_{\text{co2}} = x_2 - r_{\min}, y_{\text{co2}} = y_2 \tag{7}$$

$$x_{\text{co1}} = \frac{(x_{\text{co2}}^2 - x_1^2 + y_{\text{co2}}^2 + y_1^2 - 2y_1 y_{\text{co2}} + 2x_1 r_{\min} - r_{\min}^2)}{2(x_{\text{co2}} - x_1 + r_{\min})},$$

$$y_{\text{co1}} = y_1 \tag{8}$$

$$r_1 = x_{\text{co1}} - x_1 \tag{9}$$

$$\begin{cases} x_{\text{st}} = \dfrac{r_{\min}}{\sqrt{(x_{\text{co1}} - x_{\text{co2}})^2 + (y_{\text{co1}} - y_{\text{co2}})^2}} \cdot (x_{\text{co1}} - x_{\text{co2}}) + x_{\text{co2}} \\ y_{\text{st}} = \dfrac{r_{\min}}{\sqrt{(x_{\text{co1}} - x_{\text{co2}})^2 + (y_{\text{co1}} - y_{\text{co2}})^2}} \cdot (y_{\text{co1}} - y_{\text{co2}}) + y_{\text{co2}} \\ \theta_{st} = \arctan((y_{\text{co1}} - y_{\text{co2}})/(x_{\text{co1}} - x_{\text{co2}})) + 90 \end{cases} \tag{10}$$

For garage parking as shown in Fig. 2b, if the intermediate state $(x_1, y_1, 90)$ and the goal state $(x_2, y_2, 180)$ are already known, then the center of the turning process (x_c, y_c) with radius r, the destination of the turning process $(x_{\text{st}}, y_{\text{st}})$ will be calculated by

$$r = y_1 - y_2 \tag{11}$$

$$x_c = x_1 + r, y_c = y_1, x_{st} = x_c, y_{st} = y_2 \qquad (12)$$

For each intermediate state (x_1, y_1, 90) in Fig. 2, it should avoid the collision with the obstacles and the parking line markings. The judgment of the intermediate state will be demonstrated in the following statement. Note that the number in Fig. 5 is of the related formula.

For the parallel parking shown in Fig. 5a, the paring process has to satisfies several conditions as follows.

The right rear tire of the vehicle should avoid the collision with the vertex 1 during the first turning process:

$$(x_{co1} - x_{lot1})^2 + (y_{co1} - y_{lot1})^2 < (r_1 - d_{car}/2)^2 \qquad (13)$$

The right front vertex of the vehicle should avoid the collision with the vertex 1 during the second turning process. The parameter r_e is calculated under $r = r_{min}$ by Eq. (1) in

$$(x_{co2} - x_{lot1})^2 + (y_{co2} - y_{lot1})^2 < r_e^2 \qquad (14)$$

The left front vertex of the vehicle should avoid the collision with the curb during the first turning process

$$x_{co1} - \sqrt{(r_1 + L_{car}/2)^2 + (L_{car} - L_{rear})^2} > (x_{lot1} - d_{road}) \qquad (15)$$

The radius of the first turning process has to satisfy

$$r_1 \geqslant r_{min} \qquad (16)$$

Similarly, for the garage parking process shown in Fig. 5b, the parking process has to satisfies several conditions as follows.

The right rear tire should avoid the collision with the vertex 1 of the parking lot during the turning process:

$$(x_c - x_1)^2 + (y_c - y_1)^2 < (r - d_{car}/2)^2 \qquad (17)$$

The left front vertex of the vehicle should avoid the collision with curb during the turning process

$$x_c - \sqrt{(r + L_{car}/2)^2 + (L_{car} - L_{rear})^2} > (x_{lot1} - d_{road}) \qquad (18)$$

The right rear vertex of the vehicle should avoid the collision with the parking line markings during the turning process

$$y_c - \sqrt{(r + L_{car}/2)^2 + (L_{rear})^2} > y_{lot4} \qquad (19)$$

The rear border of the vehicle should avoid the collision with the parking line markings after the turning process:

$$x_c < x_2 \qquad (20)$$

The radius of the turning process has to satisfy

$$r \geqslant r_{min} \qquad (21)$$

Searching and saving the intermediate position by Eqs. (13) – (21) after the appropriate goal state is selected shown in Eqs. (4) – (6). From Eqs. (7) – (12), the path from the intermediate state to the goal state for each intermediate state will be figured out. Define the vehicle states which could reach any intermediate state through a straight line parallel to the curb without the collision with obstacles as the extension area of the intermediate state set. An example of the extension area of the intermediate state set is shown in Fig. 6.

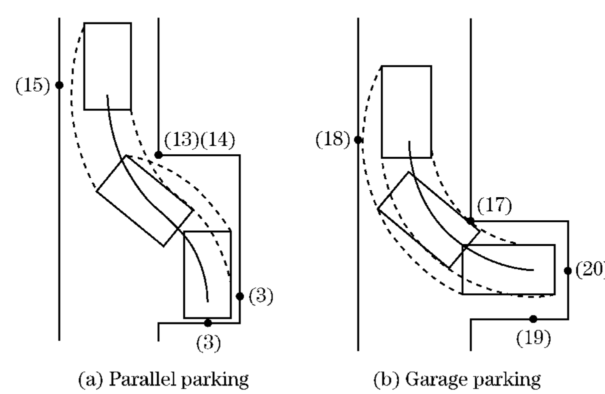


Fig. 5 Collision situation for parking

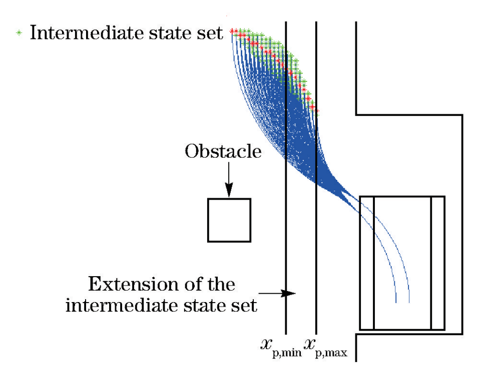
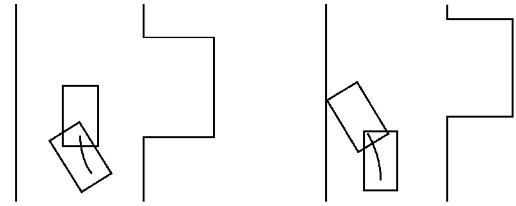


Fig. 6 Intermediate state set and its extension area

3 Intermediate state selection and path planning

The intermediate state set and the path from any intermediate state to the goal statehave been demonstrated in section 2. In this section, we will introduce the strategy of the intermediate state selection and the path planning from the initial state to the intermediate state.

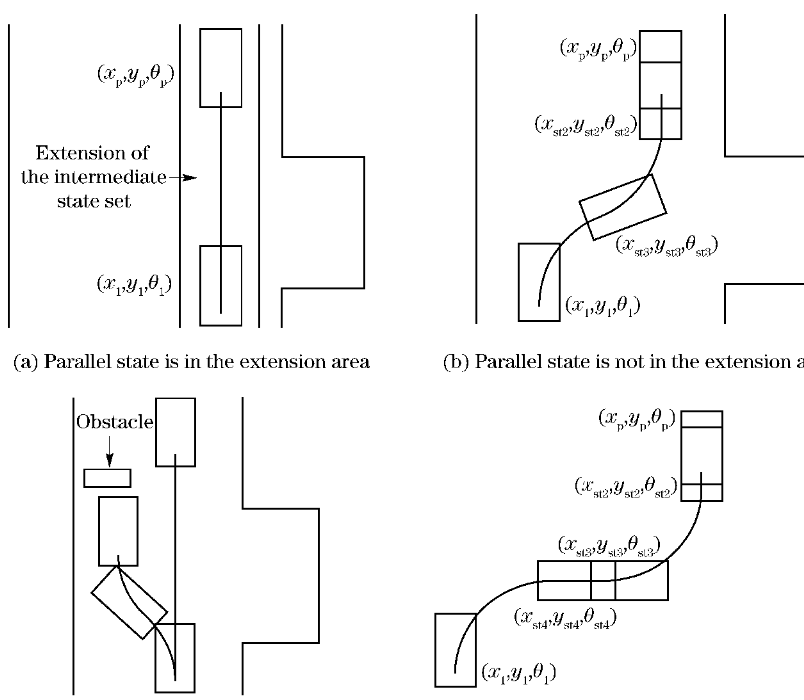
People usually make the vehicle parallel to the curb if the initial state of the vehicle is not parallel to the curb. Similarly, the algorithm will make the vehicle parallel to the curb first witha minimum radius if the initial state of the vehicle is not parallel to the curb. The forward maneuver path without the collision with obstacles or parking line markings takes the priority as shown in Fig. 7a. Otherwise, a backward maneuver path will be planned as shown in Fig. 7b. The vehicle state parallel to the curb is defined as the parallel state.



(a) Forward maneuver (b) Backward maneuver

Fig. 7 Parallel state

Then if the parallel state is in the extension area of the intermediate state set, a straight line connecting the parallel state and the intermediate state will be planned as shown in Fig. 8a, otherwise a path including two circular curves of minimum radius and a straight line will be planned as shown in Fig. 8b. Besides if the path shown in Fig. 8b collides with obstacles, then a similar backward maneuver path will be planned as shown in Fig. 8c. If the parallel state is far from the extension area of the intermediate state, a straight line will be added between two circular curves as shown in Fig. 8d.



(a) Parallel state is in the extension area (b) Parallel state is not in the extension area

(c) Backward maneuver path (d) Parallel state is far from the extension area

Fig. 8 Path planning from the initial state to the intermediate state

Considering the uncertainty of the vehicle motioncontrol, the intermediate state in the middle of the intermediate state set is recommended.

The vehicle states as (x_{st3}, y_{st3}), (x_{st4}, y_{st4}), y_{st2}, θ_{st3} and θ_{st4} are shown in Fig. 8b and Fig. 8d can be calculated as follows.

As shown in Fig. 8b, if $\Delta x = (x_{st2} - x_1) \leqslant 2r_{min}$:

$$\begin{cases} x_{st3} = x_1 + \Delta x/2 \\ y_{st3} = y_1 + \sqrt{r_{min}^2 - (r_{min} - \Delta x/2)^2} \\ \theta_{st3} = 90 - \arccos((r_{min} - \Delta x/2)/r_{min}) \\ y_{st2} = y_1 + 2\sqrt{r_{min}^2 - (r_{min} - \Delta x/2)^2} \end{cases} \tag{22}$$

As shown in Fig. 8d, if $\Delta x = (x_{st2} - x_1) > 2r_{min}$:

$$\begin{cases} x_{st3} = x_1 + r_{min},\ y_{st3} = y_1 + r_{min}, \theta_{st3} = 0 \\ x_{st4} = x_1 + \Delta x - r_{min},\ y_{st4} = y_1 + r_{min}, \theta_{st4} = 0 \\ y_{st2} = y_1 + 2r_{min} \end{cases} \tag{23}$$

The backwarel maneuver path can be calculated similarly.

The path from the initial state to the intermediate state is planned by Eqs. (22) (23). In section 2, the path planning from the intermediate state to the goal state is illustrated by Eqs. (7) – (12). As a consequence, the entire parking path can be obtained.

4 Simulation and experiment results

4.1 Simulation results

The path planning method is programmed using MATLAB. The simulation assumes the vehicle initial state and the map is already known, the result of the simulation for different initial states and different environments will be demonstrated in the following statement.

All examples were simulated on a Windows 7 system with a 2.2 GHz Intel Core 2 Duo CPU T6600 processor and 4 GB of memory. All examples finished in less than 1 s. The parameters of the simulation are shown in Tab. 1. The parameters of the vehicle are shown in Tab. 2.

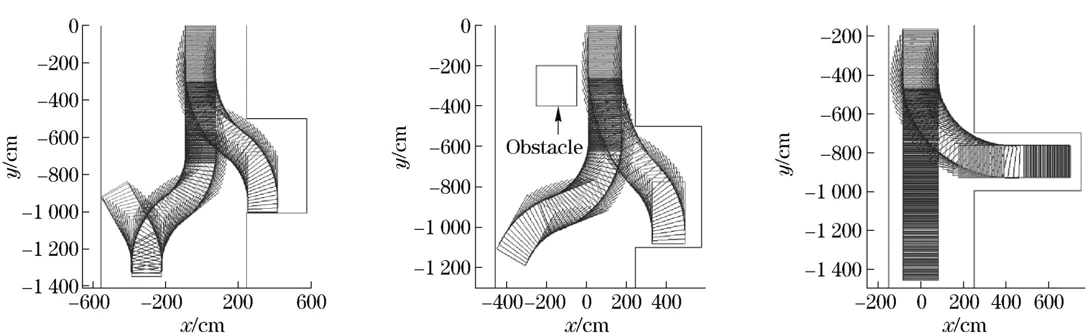
Tab. 1 Simulation parameters

No.	L_{lot}/cm	d_{lot}/cm	Road width/cm
1	510	330	800
2	600	330	800
3	300	500	400

Tab. 2 Vehicle parameters

L_{car}/cm	d_{car}/cm	L_{front}/cm	L_{rear}/cm	$\theta_{steer,max}$/(°)
308	165	60	55	26.34

As shown in Fig. 9a, it is a tiny parking lot case result in choosing the goal state shown in Eq. (5). Besides parking for such an initial state of the vehicle is really tough, the vehicle has to drive backward at first, then moves to the extension area of the intermediate state through two circular curves. Fig. 9b shows the parking process in an environment with obstacles and the goal state in the middle of the parking lot shown in Eq. (4) is selected because of the spacious parking lot. The simulation result of the garage parking is shown in Fig. 9c.



(a) Parallel parking under tough situation (b) Parallel parking with obstacles (c) Garage parking

Fig. 9 Simulation results

4.2 Autonomous vehicle platform

The proposed parking plan method and the parking system will be tested on the IN2BOT autonomous vehicle platform as shown in Fig. 10.

The IN2BOT is modified on a Polaris Ranger EV, which is an all-terrain vehicle. The parameters of the vehicle are shown in Tab. 2. The vehicle is equipped with several cameras to get the image information around the vehicle, traffic signs and traffic lights. A Velodyne 3D LIDAR provides the point clouds of the environment. INS and GPS provides the position and direction of the vehicle. A speedometer provides the measurement of driving speed and mileage. The vehicle is also equipped with three motors for throttle, steering and brake. IN2BOT has successfully accomplished the "Future Challenge 2014" autonomous vehicle contest for 20 km missions on urban roads and rural roads. In addition, a parallel parking mission is set in the contest of urban roads. The system to accomplish this mission is the proposed automatic parking system.

Fig. 10 IN2BOT autonomous vehicle platform

4.3 Automatic parking system

The proposed automatic parking system is a subsystem of the autonomous driving system. It contains three modules which enable the vehicle to perform parking process. The diagram of the automatic parking system is shown in Fig. 11.

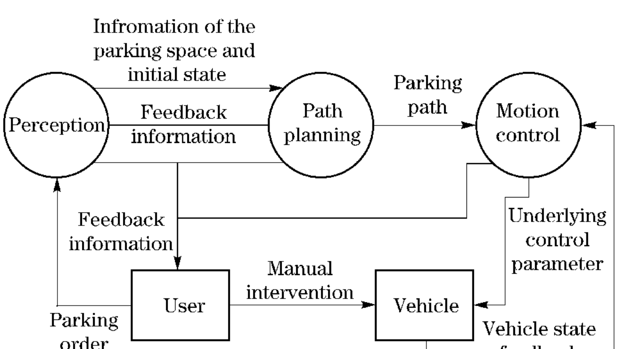


Fig. 11 Automatic parking system

When the drivercommands the parking order, the perception module will search the parking lot while the vehicle is steering forward and get the coordinate of the vertices of the parking lot and the initial state of the vehicle. The parking lot search is based on the panoramic vision sensor on the top of the vehicle. The result of the parking lot search is shown as the quadrangle in Fig. 12.

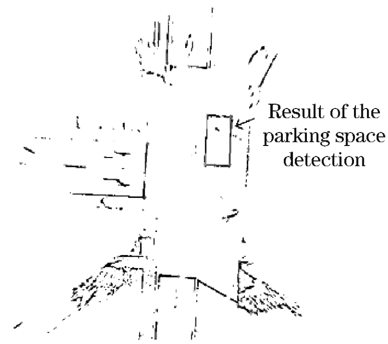


Fig. 12 Result of the parking lot perception

Then the path planning module will process the information received to get the map and plan path using the proposed path planning algorithm. An initialization process including a rectangular process and a rotation process is needed because of the non-rectangular parking space. If an appropriate-sized parking lot is detected and a reasonable path is planned, then a conformation will be submitted to the perception module, otherwise a request of parking lot search will be sent. Note that the perception module and the path planning module would interact frequently until the parking path is planned or the parking lot is abandoned if an appropriate-sized parking lot is detected without a parking path because of the obstacles in the environment.

The motion control module calculates the related control parameters, such as speed and steering angle, and performs the automatic parking process using mileage and navigation information. Because of the circular curves we use, steering angle for each part of path will be figured out easily. In addition, a closed-loop controller on heading angle is implemented because little bias on heading angle will cause huge error onthe lateral displacement.

The user can ensure the vehicle state from the feedback information sent from three modules and intervene the automatic parking at any time during the entire process of parking. The specific examples and experiment results are demonstrated in section 4. 4.

4. 4 Experiment results

Three experiments are performed for parking. Experiment 1 is parallel parking, Experiment 2 and 3 are garage parking. Experiment 2 has the situation shown in Fig. 9b which the parallel state is not in the extension area of the intermediate state set. The parameter and the results of the experiments are shown in Tab. 3 and Tab. 4. The result of the path planning is shown as the solid curves in Fig. 13.

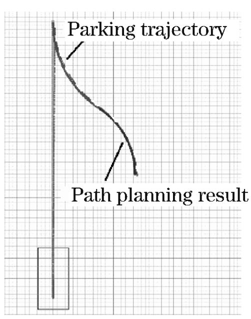
Parking trajectory is shown as thedash lines in Fig. 13. The parking process is shown in Fig. 14. The result of the experiments shows that the correctness and practicability of the path planning method and the automatic parking system.

Tab. 3 Results of the parking lot detection

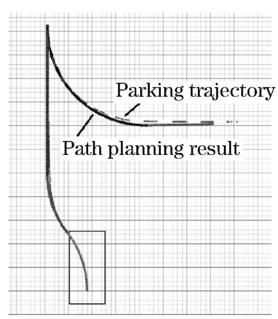
No.	Category	L_{lot}/cm	d_{lot}/cm	Road width/cm
1	True value	550	250	Large
	Detect result	533	239	1 189
2	True value	520	260	955
	Detect result	448	272	1 374
3	True value	520	260	955
	Detect result	496	248	1 101

Tab. 4 Error of the parking process

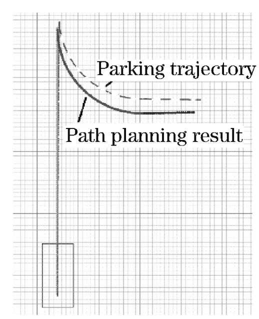
No.	Lateral error/ cm	Longitudinal error/ cm	Heading angle error/(°)
1	13	39	2. 06
2	39	41	0. 47
3	9	19	0. 70



(a) Experiment 1



(b) Experiment 2



(c) Experiment 3

Fig. 13 Result of the path planning and the parking trajectory


(a) Garage parking process


(b) Parallel parking process

Fig. 14 Result of the path planning and the parking trajectory

5 Conclusion and the future work

In this paper, a path planning methodis proposed for parking and an automatic parking system is established. The proposed method succeeds to plan path without the collision with the obstacles in different parking conditions. The result of experiments shows the correctness and the practicability of the automatic parking system.

The future work includesdeveloping path planning methods in complex environment such as maze and the parking system with more accurate steering.

References:

[1] Jaillet L, Cortés J, Siméon T. Sampling-based path planning on configuration-space costmaps[J]. Robotics, IEEE Transactions on, 2010, 26(4): 635 – 646.

[2] Han L, Do Q H, Mita S. Unified path planner for parking an autonomous vehicle based on RRT[C]// Robotics and Automation (ICRA), 2011 IEEE International Conference on, 2011: 5622 –5627.

[3] Devaurs D, Siméon T, Cortés J. Enhancing the transition-based RRT to deal with complex cost spaces[C]// Robotics and Automation (ICRA), 2013 IEEE International Conference on, 2013: 4120 –4125.

[4] Vasile C I, Belta C. Sampling-based temporal logic path planning[C]//Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on, 2013: 4817 –4822.

[5] Ok K, Ansari S, Gallagher B, et al. Path planning with uncertainty: Voronoi uncertainty fields[C]// Robotics and Automation (ICRA), 2013 IEEE International Conference on, 2013: 4596 –4601.

[6] Hanafy M, Gomaa M M, Taher M, et al. Path generation and tracking for car automatic parking employing swarm algorithm[C]//Computer Engineering & Systems (ICCES), 2011 International Conference on, 2011: 99 –104.

[7] Kim D, Chung W, Park S. Practical motion planning for car-parking control in narrow environment [J]. IET Control Theory & Applications, 2010, 4 (1): 129 –139.

[8] Lini G, Piazzi A, Consolini L. Multi-optimization ofsplines for autonomous parking[C]//Decision and Control and European Control Conference (CDC – ECC), 2011 50th IEEE Conference on, 2011: 6367 –6372.

[9] Hsieh M F. A parking algorithm for an autonomous vehicle[C]//2008 IEEE Intelligent Vehicles Symposium, 2008: 1155 –1160.

[10] Choi S, Boussard C, D'Andrea-Novel B. Easy path planning and robust control for automatic parallel parking[J]. World Congress, 2011, 18(1): 656 – 661.

[11] Vorobieva H, Glaser S, Minoiu Enache N, et al. Geometric path planning for automatic parallel parking in tiny spots[J]. Control in Transportation Systems, 2012, 13(1): 36 –42.

[12] Liang Z, Zheng G, Li J. Automatic parking path optimization based on Bezier curve fitting[C]//Automation and Logistics (ICAL), 2012 IEEE International Conference on, 2012: 583 –587.

[13] Vorobieva H, Minoiu-Enache N, Glaser S, et al. Geometric continuous-curvature path planning for automatic parallel parking[C]//Networking, Sensing and Control (ICNSC), 2013 10th IEEE International Conference on, 2013: 418 –423.

[14] Lyon D. Parallel parking with curvature and nonholonomic constraints[C]//Intelligent Vehicles' 92 Symposium, Proceedings of the IEEE, 1992: 341 – 346.

[15] De Luca A, Oriolo G, Samson C. Feedback control of a nonholonomic car-like robot[M]//Robot Motion Planning and Control. Berlin: Springer, 1998: 171 –253.

(**Edited by** Wang Yuxia)