2017 9th International Conference on Intelligent Human-Machine Systems and Cybernetics

Trajectory Planning for Autonomous Ground Vehicles Driving in Structured Environments

Chao Li, Xiaohui Li, Junxiang Li, Qi Zhu, and Bin Dai
the College of Mechatronic Engineering and Automation
National University of Defense Technology
Changsha, P. R. China
Email: lichao.bit@foxmail.com, xiaohui_lee@outlook.com
junxiangli90@gmail.com, zhq_cs@126.com, bindai.cs@gmail.com.

***Abstract*—To solve the problem of trajectory planning for Autonomous Ground Vehicles (AGVs) in structured environments, a kinematically-feasible trajectory planning approach based on road models is proposed. In order to cope with obstacle avoidance on roads, we develop an efficient path generation method, using the piecewise spiral curve to generate a set of continuous curvature paths which satisfy the constraints of the start and end point boundaries. Based on the proposed optimization function, the optimal path is selected. Compared with the model-based predictive trajectory planner, the method proposed in this paper can effectively avoid the problem of slow convergence or no feasible solution. The experimental results show that: The proposed method not only retains most of the advantages of model-based predictive trajectory planning, but also reduces the computational complexity to meet the real-time requirements. The curvature of the generated path is continuous and is suitable for the actual control of the vehicle. Simulation results show that the proposed method can track the reference path smoothly and avoid static obstacles successfully.**

***Keywords*-Autonomous ground vehicle, sampling-based motion planning, local path generation, obstacle avoidance.**

I. Introduction

Autonomous driving technology is considered to have great potential in improving driving safety and comfort. In the last ten years, the unmanned car has made remarkable achievements. But so far, the autonomous driving technology for complex environment has not been mature. As a key part of autonomous driving technology, there are many problems that need to be solved in path planning [1] [2]. The path planner must be able to generate an executable path quickly and efficiently in a variety of complex scenarios in real time. In the structured environments, the geometric information of the reference path has a strong constraint on the local path of the vehicle, so it is necessary for the path planner to consider the geometric constraint of the reference path when generating the local path. The method of obtaining the reference path according to the perception information, such as lanes and curbs, cant guarantee the continuity and smoothness of the curvature and heading angle, which is not suitable for vehicle driving. Therefore, it is necessary to generate kinematic feasible paths using local path planner. In this paper, we focus on solving the problem of AGVs path planning problem in structured environments. On the basis of sate-space sampling, we consider the constraints of vehicle position and heading to generate a set of kinematically-feasible paths in real time.

II. Related Work

Local planning, which is also known as the motion planning, is an important part of the planning module. Local planning generates feasible paths in real time in the circumstance of dynamic traffic participants. The common method using in local planners can be classified into two types. One is the optimization method considering the boundary constraint [3], [4], the other is the path generation method based on sampling [5]. The sampling based path generation method can be divided into two categories: state space sampling, control space sampling [6].

Von et al. proposed a local path generation algorithm based on arc tentacles, which generates a series of candidate arcs with different radius after discretizing the control space [7]. Himmelsbach et al. improved this method using Clothoid as trajectory primitive while taking into account the current curvature and the velocity of the AGVs [8]. Because the control space sampling has the characteristics of simple calculation and good real-time performance, this method is widely used in local planning of AGVs [7]. However, this method also has inherent shortcomings. For example, the generated trajectory shape is too simple to deal with complex obstacle constraints. In addition, since this method does not take into account the end state constraints of the trajectory, a large number of computational resources are consumed on the generation and evaluation of non-executable trajectory.

In order to solve the problems existing in the control space sampling, some scholars proposed the method of state space sampling. The state-space sampling method first discretizes state space, and the local path generation problem is formulated into a Two-Point Boundary Value Problem(BVP). Kelly et al. expressed the steering control command of car-like robot as a polynomial about the arc length and solved

978-1-5386-3022-8/17 $31.00 © 2017 IEEE
DOI 10.1109/IHMSC.2017.125

the unknown parameters using the gradient-based shooting method [9] [10]. The curvature of the path obtained by this method is polynomial of arc length, which is convenient for vehicle tracking. The AGV platform developed by Carnegie Mellon University using this local planning method won the championship in DARPA Urban Challenge [11]. However, due to the nonlinearity of the vehicle model, the shooting method used to sometimes can't converge or converge too slow. In order to solve this problem, Ferguson et al. applied an initial parameter lookup table to improve the convergence rate of the algorithm [12]. Kanayama et al. used a piecewise spiral curve to connect two points on the plane while ensuring the generated path curvature is continuous [13]. But this method only considers the curvature constraint without considering the existence of obstacles. In order to deal with the obstacles in real time, we develop a sampling-based piecewise spiral curve path generation method to generate a series of kinematically-feasible path while satisfying the reference path constraints.

The remainder of this article is organized as follows. Section 3 mainly describes the method of generating local path by using piecewise spiral curve, and compares the experimental result with the path generation method based on model-based predictive path planner. Section 4 shows the simulation results. Conclusions and future work are discussed in Section 5.

III. Sampling-based piecewise spiral curve path generation method

Because the state-space sampling has the advantage of making full use of the geometric constraint of the reference path and good consistency of the spatial distribution of the generated path, it attracts a lot of scholars to study it. The overall idea of state space sampling is that the end of the trajectory is first sampled and then the trajectory generation problem is transformed into a Two-Point Boundary Value Problem [14]. In this part, we propose a sampling-based piecewise spiral curve path generation method .The generation method mainly includes three steps. First, sampling along the reference path. Second, generating the candidate path. Last, selecting the optimal path based on optimization function.

A. *Sampling the final state along the reference path*

In the actual environment, there are often a variety of obstacles around the reference path, so the local motion planner needs to generate a set of candidate path in real time to ensure its flexibility and feasibility. The traditional pure pursuit algorithm has the advantage of good real time while considering the position constraint of the preview point. However, this method ignores the heading angle constraint of the preview point, so that the generated candidate path cant be consistent with the heading of the reference path at the preview point. When the speed of the vehicle is high or the curvature of the reference path is too large, it is likely to cause problems such as sudden steering or overshoot of the vehicle. In order to ensure that the generated candidate path and reference path are conformal, we add the heading constraint to the final sampling state. Given vehicle initial state $[x_I, y_I, \theta_I]$,we sample a set of final states $[x_T, y_T, \theta_T]$ as showing in Fig. 1.In order to ensure the candidate path and the reference path are aligned, the heading of the final sampling state is equal to the heading of the nearest point on the reference path.

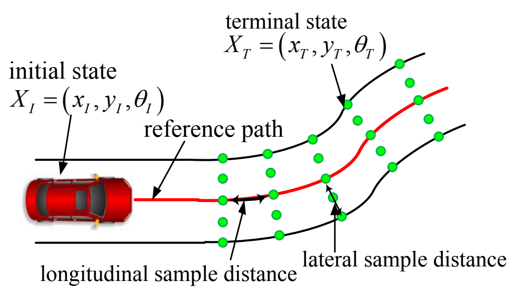

Figure 1: State space sampling.

B. *Piecewise spiral curve path generation method*

In this paper, the heading angle of the generated path is piecewise cubic function to the length of path. As showing in Fig.2 , for two points: $P_I(x_I, y_I, \theta_I)$, $P_T(x_T, y_T, \theta_T)$,if $\alpha 1 = \alpha 2$, we define these two points as symmetric configuration. According to the nature of spiral curve, two points which are symmetric configuration, we can use spiral to connect two points:

$$
\begin{aligned}
& l = \frac{d}{D(\alpha)} \\
& \kappa(s) = \frac{6\alpha D(\alpha)^3}{d^3}\left(\frac{d^2}{4D(\alpha)^2} - s^2\right) \\
& D(\alpha) = 2\int_0^{1/2} \cos\left(\alpha\left(3/2 - 2s^2\right)s\right)ds
\end{aligned}
\tag{1}
$$

Where l is the length of the curve, $\kappa(s)$ is the curvature, d is the straight distance between two points , $\alpha = \theta_T - \theta_I$.

If the given two points P_I and P_T are not symmetric configuration, we cant use spiral to connect two points directly. We need to find an intermediate point O ,which makes $P_I O$ and $P_T O$are symmetric configuration respectively.

When $\theta_I = \theta_T$, the set of intermediate points are :

$$
U1 = \{(x,y) \,|\, (x - x_I)(y - y_I) = (x - x_T)(y - y_T)\} \tag{2}
$$

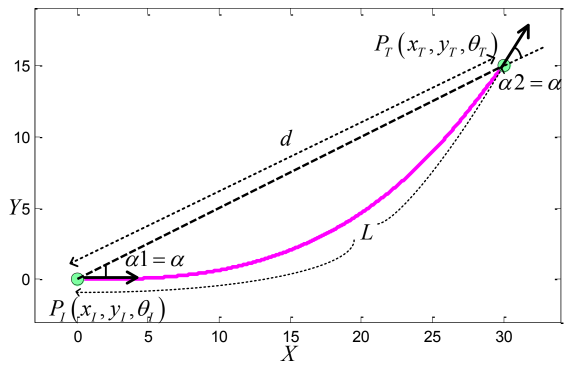

Figure 2: Symmetric configuration.

When $\theta_I \neq \theta_T$, the set of intermediate points are:

$$U2 = \left\{ (x,y) \,|\, (x-x_I)(x-x_T) + (y-y_I)(y-y_T) * \tan\left(\frac{\theta_T-\theta_I}{2}\right) = (x-x_I)(y-y_T) - (x-x_T)(y-y_I) \right\} \quad (3)$$

Algorithm 1 gives a detailed description of the path generation method based on piecewise spiral curves. As showing in Fig.3, we demonstrate a comparison of our methods and model predictive local path generation method proposed by Li et al [14]. The path generation method based on model predictive, because of the strong nonlinearity of the system model, can't be solved by analytical method, so it needs to be solved by numerical integration method. In practical application, this method may can't get feasible solution or converge slowly. In this paper, we use the piecewise spiral curve as the path primitive which avoids solved by iteration, so it has the advantage of low computational cost. In addition, we can find that, by the given initial and goal state, our method can get a set of continuous curvature and satisfy constraints of initial and target state, to ensure the flexibility of the candidate path.

C. Path evaluation

The generated candidate paths are first checked against obstacles explicitly to ensure safety. The candidate path with minimum cost is selected as the optimal path. We design a cost function to evaluate the security, smoothness and consistency of the planning results:

$$cost = \omega_1 J_p + \omega_2 J_s + \omega_3 J_d + \omega_4 J_c + \omega_5 J_o \quad (4)$$

Where J_p is the length cost of candidate path. The purpose of this term is to punish the vehicle reactive behavior caused by the short path.

$$J_p = \frac{s_{\max} - s}{s_{\max} - s_{\min}} \quad (5)$$

Algorithm 1 Path generation method based on piecewise spiral curve.

```
Input: Initial state P_I(x_I, y_I, θ_I) Final state P_T(x_T, y_T, θ_T)
Output: Candidate path parameters
// determine whether the two points are symmetric configurations
if The two points are symmetric configurations then
    Connect the two points using cubic spiral directly
    Output: Candidate path parameters
end if
if θ_I = θ_T then
    Sampling intermediate point from U1
    Connect each symmetric configuration
    Output: Candidate path parameters
end if
if θ_I ≠ θ_T then
    Sampling intermediate point from U2
    Connect each symmetric configuration
    Output: Candidate path parameters
end if
```

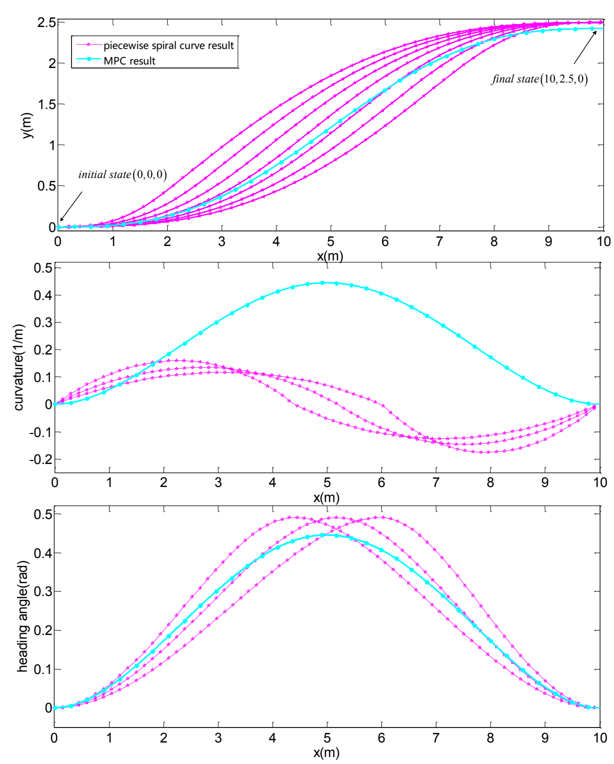

Figure 3: An example of the path generation method compare with model predictive method.

s_{max} is the upper limit of the path length threshold, s_{min} is the lower limit of the path length threshold, s is the path length.

J_s is the curvature cost of candidate path. The curvature of the path is directly related to the vehicles lateral control stability and comfortability.

$$J_s = \frac{1}{s_f} \int_0^{s_f} \frac{\kappa(s)}{\kappa_{max}} ds \tag{6}$$

s_f is the length of candidate path. κ is the curvature of the path. κ_{max} is the upper limit of curvature. The purpose of this term is to punish the curvature cost of candidate path.

J_d is the cost of cumulative lateral distance offset with respect to the reference path.

$$J_d = \begin{cases} \frac{1}{s_f} \int_0^{s_f} \frac{d}{d_{threshold}} ds\,, & d < d_{threshold} \\ \infty\,, & d \geq d_{threshold} \end{cases} \tag{7}$$

d is the lateral distance offset, $d_{threshold}$ is the width of the road, When the deviation is greater than the road width, the cost is infinite. The purpose of this term is to punish the lateral offset of the candidate path to the reference path.

J_c is the cost of consistency of the planning results. In order to prevent the vehicle from jittering, we evaluated the consistency of the alternative path and the current path.

$$J_c = d_s - d_0 \tag{8}$$

Where d_s is the lateral deviation of the next sampling point and d_0 is the lateral deviation of the current sampling point.

J_o is to evaluate the distance of the path from the obstacle. We consider the vehicle size to set the minimum threshold :d_{min} and the maximum threshold :d_{max} for the distance.

$$J_o = \begin{cases} 0\,, & d < d_{min} \\ \frac{d_{max} - d}{d_{max} - d_{min}}\,, & d_{min} < d < d_{max} \\ \infty\,, & d > d_{max} \end{cases} \tag{9}$$

IV. Experimental Results

In order to test the effectiveness of our local path planning algorithm, various simulation experiments in $MATLAB^{\circledR}$ are designed.

The experimental scenario design is shown in Fig.4. We set static obstacles along the reference road, while dynamic obstacles are not considered. The local path planning in the whole process adopts the local path planner proposed in Section 3. The lower figure shows a snapshot of state sampling and the process of selecting the optimal path. This paper assumes that the controller can track the local path accurately. The parameters of the path generation method are set as follows: $l_p \in [10m, 15m, 20m]$, $d_p = [-2m, 2m]$, $\Delta d_p = 0.5m$, $\kappa_{max} = 0.2m^{-1}$, $d_{min} = 1m$, $d_{max} = 2m$, $s_{max} = 35m$, $s_{min} = 10m$. As can be seen from the Fig.5, the vehicle can avoid static obstacles on the road, while the maximum curvature of the vehicle to meet the constraints (the maximum curvature of the experiment set to 0.2 / m). In addition, we can see from Fig.3 that the curvature generated by the piecewise spiral curve is continuous.

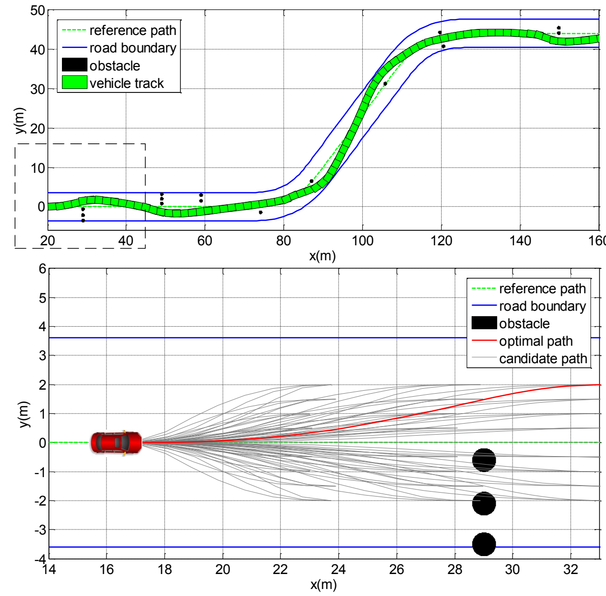

Figure 4: Simulation results for path following with static obstacles avoidance.

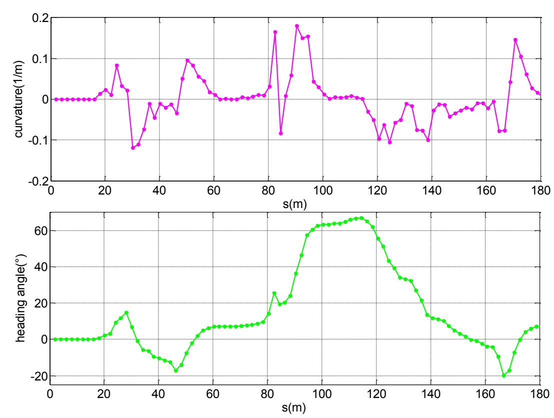

Figure 5: The curvature and heading angle of the path following simulation results.

In order to verify the flexibility of our proposed approach in complex environments, we designed the experimental scenario as shown Fig.6. Compared with the path generation method based on model predictive, piecewise spiral curve path generation method can still generate a large number of feasible candidate paths in the environment of complex obstacles while the other can't. As we can see in Fig.7, we can change the shape of the generated path by changing the intermediate point.

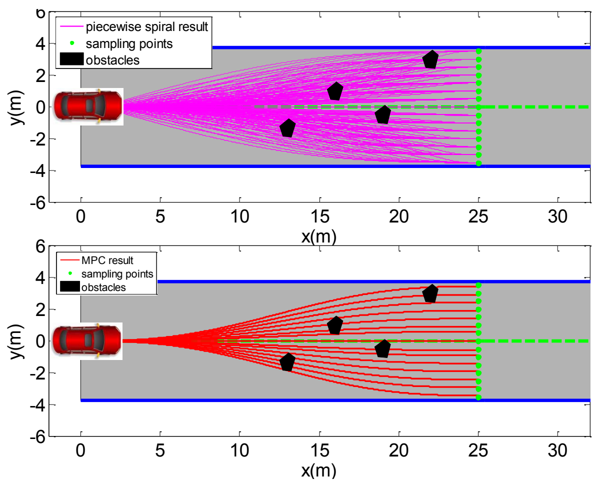

Figure 6: Obstacle avoidance experiment under complex obstacles.

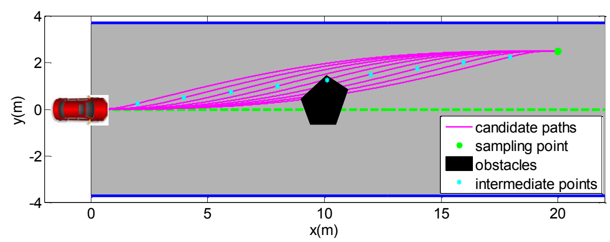

Figure 7: Changing the intermediate point.

In order to verify the real-time performance of our proposed approach, we counted the time of the path generation time in Fig.6. The results are shown in Tab.I. From the results we can find that it takes only 0.037 seconds to generate 127 candidate paths, which proves that our method can meet the real-time requirements.

The above simulation results show the effectiveness of our path planning approach.

V. Conclusions

This paper studies the planner of autonomous vehicle in structured environments. We proposed a state-space sampling based method is used to generate candidate paths which are conformal to the reference path. This method can consider a variety of non-holonomic constraints, such as curvature upper limit, curvature change rate, etc. Because our method avoids solving the nonlinear equation, the real-time performance of the algorithm is improved. The simulation results show that it can track the reference path and smoothly as well as avoiding static obstacles. Future work is mainly focused on dynamic obstacle avoidance and the optimization of reference path consider the constraints of static obstacles and road boundary.

Table I: Piecewise spiral curve parameters setting and calculation time

Meaning	Value
Initial state	$(0,0,0)$
Final state	$(20,2.5,0)$
Integral step	$0.01m$
Number of paths	127
Calculation time	$0.037s$

Acknowledgment

This work was supported by the China Scholarship Council and the National Nature Science Foundation of China under No.61375050 and No.91220301.

References

[1] J. Mcdonald, *International Machine Vision and Image Processing Conference-Imvip 2007*. IEEE Computer Society Press, 2008.

[2] J. J. Leonard, "Challenges for autonomous mobile robots," in *Machine Vision and Image Processing Conference, 2007. IMVIP 2007. International*. IEEE, 2007, pp. 4–4.

[3] G. Bevan, H. Gollee, and J. O'reilly, "Trajectory generation for road vehicle obstacle avoidance using convex optimization," *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, vol. 224, no. 4, pp. 455–473, 2010.

[4] Y. Yoon, J. Shin, H. J. Kim, Y. Park, and S. Sastry, "Model-predictive active steering and obstacle avoidance for autonomous ground vehicles," *Control Engineering Practice*, vol. 17, no. 7, pp. 741–750, 2009.

[5] M. McNaughton, "Parallel algorithms for real-time motion planning," 2011.

[6] T. M. Howard, C. J. Green, A. Kelly, and D. Ferguson, "State space sampling of feasible motions for high-performance mobile robot navigation in complex environments," *Journal of Field Robotics*, vol. 25, no. 6-7, pp. 325–345, 2008.

[7] F. Von Hundelshausen, M. Himmelsbach, F. Hecker, A. Mueller, and H.-J. Wuensche, "Driving with tentacles: Integral structures for sensing and motion," *Journal of Field Robotics*, vol. 25, no. 9, pp. 640–673, 2008.

[8] M. Himmelsbach, T. Luettel, F. Hecker, F. von Hundelshausen, and H.-J. Wuensche, "Autonomous off-road navigation for mucar-3," *KI-Künstliche Intelligenz*, vol. 25, no. 2, pp. 145–149, 2011.

[9] A. Kelly and B. Nagy, "Reactive nonholonomic trajectory generation via parametric optimal control," *The International Journal of Robotics Research*, vol. 22, no. 7-8, pp. 583–601, 2003.

[10] B. Nagy and A. Kelly, "Trajectory generation for car-like robots using cubic curvature polynomials," *Field and Service Robots*, vol. 11, 2001.

[11] C. Urmson, J. Anhalt, D. Bagnell, C. Baker, R. Bittner, M. Clark, J. Dolan, D. Duggins, T. Galatali, C. Geyer *et al.*, "Autonomous driving in urban environments: Boss and the urban challenge," *Journal of Field Robotics*, vol. 25, no. 8, pp. 425–466, 2008.

[12] D. Ferguson, T. M. Howard, and M. Likhachev, "Motion planning in urban environments," *Journal of Field Robotics*, vol. 25, no. 11-12, pp. 939–960, 2008.

[13] Y. Kanayama and B. I. Hartman, "Smooth local path planning for autonomous vehicles," in *Robotics and Automation, 1989. Proceedings., 1989 IEEE International Conference on*. IEEE, 1989, pp. 1265–1270.

[14] X. Li, Z. Sun, D. Cao, D. Liu, and H. He, "Development of a new integrated local trajectory planning and tracking control framework for autonomous ground vehicles," *Mechanical Systems and Signal Processing*, vol. 87, pp. 118–137, 2017.