2014 14th International Conference on Control, Automation and Systems (ICCAS 2014)
Oct. 22–25, 2014 in KINTEX, Gyeonggi-do, Korea

Trajectory Generation for a Car-Like Mobile Robot using Closed-Loop Prediction

Byungjae Park and Woo Yong Han*

Department of Vehicle-IT Convergence, ETRI,
Daejeon, South Korea (wyhan@etri.re.kr)

Abstract: This paper proposes a trajectory generation method using the closed-loop prediction to follow reference path, which are provided by a global path planner. Using the proposed method, smooth trajectories can be generated by considering the position of a mobile robot and its kinematic properties. The feasibility of the proposed method was verified by simulations in various situations.

Keywords: Autonomous navigation, path planning, mobile robot.

1. INTRODUCTION

Generating smooth trajectories is important for a non-holonomic mobile robot because the non-holonomic mobile robot can not move rapidly due to the its kinematic properties.

The polynomial spiral curve-based trajectory generation method can generate smooth trajectories [1]. This method generates a smooth trajectory to reach a given position using a single polynomial spiral curve consists of several parameters. The Bézier curve-based trajectory generation methods can generate smooth trajectories without any complex process because the continuity of a Bézier curve is guaranteed [2, 3].

Although the aforementioned methods can generate smooth trajectories, the kinematic properties of a non-holonomic mobile robot such as a maximum steering angle can not be considered directly.

In this paper, a trajectory generation method using the closed-loop prediction to follow reference path, which are provided by a global path planner. The proposed method can generate smooth trajectories by considering the kinematic property and position of a mobile robot.

The remainder of this paper is organized as follows. The closed-loop prediction is described in section 2. Simulation results are shown in section 3, and the conclusion follow in section 4.

2. CLOSED LOOP PREDICTION

The proposed method generates the trajectory of a vehicle using the close-loop prediction [4, 5]. The close-loop prediction consists of a low-level controller and vehicle model (Fig. 1). The low-level controller determines a control input to follow a given reference command, and the vehicle model estimate the state of a vehicle using the control input determined by the low-level controller.

A reference command r consists of the location of a waypoint and a desired velocity, the control input con-

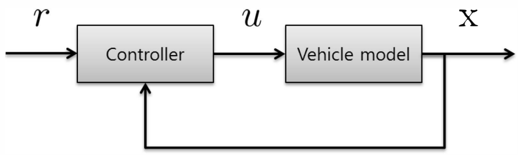


Fig. 1 Close-loop prediction: r is a reference command, u is a control input, and X is the state of a mobile robot. The close-loop prediction determines a control input iteratively by comparing a reference command and the state of a vehicle.

sists of the velocity and steering angle of a vehicle, and the state of a vehicle consists of the location and heading angle.

Using the close-loop prediction, we can generate feasible trajectories because the low-level controller and vehicle model can consider various properties of a vehicle such as the maximum steering angle, kinematic properties.

2.1 Controller

The proposed method uses a pure-pursuit algorithm to determine control input to follow the reference path [6,7]. The pure-pursuit algorithm determines the steering angle of a car-like mobile robot to approach a given goal in a circular motion (Fig. 2). The radius of a circular motion of a mobile robot is determined as follows:

$$R = \frac{l}{2\sin(\eta)}, \tag{1}$$

where l and η are a radial and angular errors, respectively. They are calculated calculated as follows:

$$l = \left((g_x - x(t))^2 + (g_y - y(t)^2)\right)^{\frac{1}{2}}, \tag{2}$$

$$\eta = \tan\left(\frac{\mathrm{g_y - y(t)}}{\mathrm{g_x - x(t)}}\right) - \theta(\mathrm{t}), \tag{3}$$

where $x(t)$, $y(t)$ and $\theta(t)$ are the position of a mobile robot. The relation between the steering angle of a car-like mobile robot and radius of a circular motion is repre-

This work was supported by the IT R&D program of MSIP/KEIT [10041417].

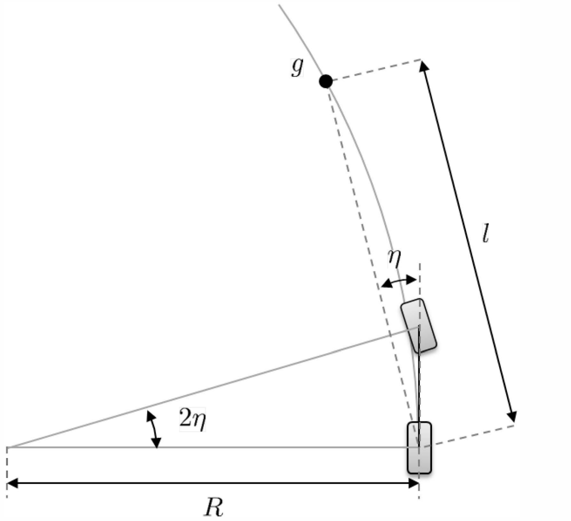


Fig. 2 Pure-pursuit algorithm concept: g is a given goal, l is a distance between the goal and current location of a car-like mobile robot, R is a radius of a circular motion, and η is a difference between the goal direction and heading direction of a car-like mobile robot.

sented as follows:

$$\phi = \tan^{-1}\left(\frac{\mathrm{L}}{\mathrm{R}}\right), \tag{4}$$

where L is a base length of a mobile robot. By plugging eqn. 1 into eqn. 4, the steering angle of a car-like mobile robot to reach a given goal can be determined as follows:

$$\phi = \tan^{-1}\left(\frac{2\,\mathrm{L}\sin(\eta)}{\mathrm{l}}\right), \tag{5}$$

2.2 Vehicle Model

The proposed method estimates the state of a car-like mobile robot using the following state transition model [4]:

$$\mathrm{x}(\mathrm{t}+\Delta\mathrm{t}) = \mathrm{x}(\mathrm{t}) + \int_{\mathrm{t}}^{\mathrm{t}+\Delta\mathrm{t}} \dot{\mathrm{x}}(\mathrm{t})\,\mathrm{dt}, \tag{6}$$

where $\mathrm{x}(\mathrm{t}) = [\mathrm{x}(\mathrm{t}), \mathrm{y}(\mathrm{t}), \theta(\mathrm{t})]^{\mathrm{T}}$ is a state of a car-like mobile robot. The state of a mobile robot at $t + \Delta t$ can be estimated from using using the following equation:

$$\begin{aligned} &\dot{\mathrm{x}}(\mathrm{t}) \\ &= \left[\dot{x}(t)\ \dot{y}(t)\ \dot{\theta}(t)\right]^{T} \qquad (7) \\ &= \left[v(t)\cos(\theta(\mathrm{t}))\ \mathrm{v}(\mathrm{t})\sin(\theta(\mathrm{t}))\ \tfrac{\mathrm{v}(\mathrm{t})}{\mathrm{L}}\tan(\phi(\mathrm{t}))\right]^{T}, \qquad (8) \end{aligned}$$

where $v(t)$ is the velocity of a car-like mobile robot, $\phi(t)$ is the steering angle of a car-like mobile robot. The velocity $v(t)$ is given, and the steering angle is determined by the controller using the pure-pursuit algorithm.

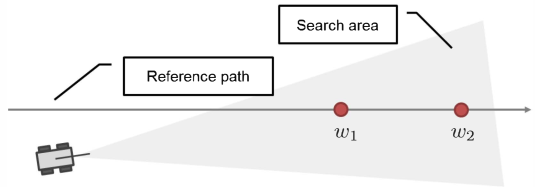


Fig. 3 Waypoint search: an appropriate waypoint is selected by considering the state of a car-like mobile robot, look-ahead distance, and maximum steering angle of a car-like mobile robot [8].

2.3 Waypoint Search

To follow a reference path safety using the proposed method, an appropriate reference command have to be given to the closed-loop prediction. As shown in Fig. 3, The reference command is used as a goal of the pure-pursuit algorithm. We uses waypoints on the reference path to give an appropriate reference command using the following equation [8]:

$$\begin{aligned} &w = \underset{w_i}{\mathrm{argmin}}\left(|w_i - \mathrm{x}(\mathrm{t})| - \mathrm{D}\right), \\ &\left(w_i \in P, \frac{w_i - \mathrm{x}(\mathrm{t})}{|w_i - \mathrm{x}(\mathrm{t})|}\cdot\delta_c \le G_{max}\right), \end{aligned} \tag{9}$$

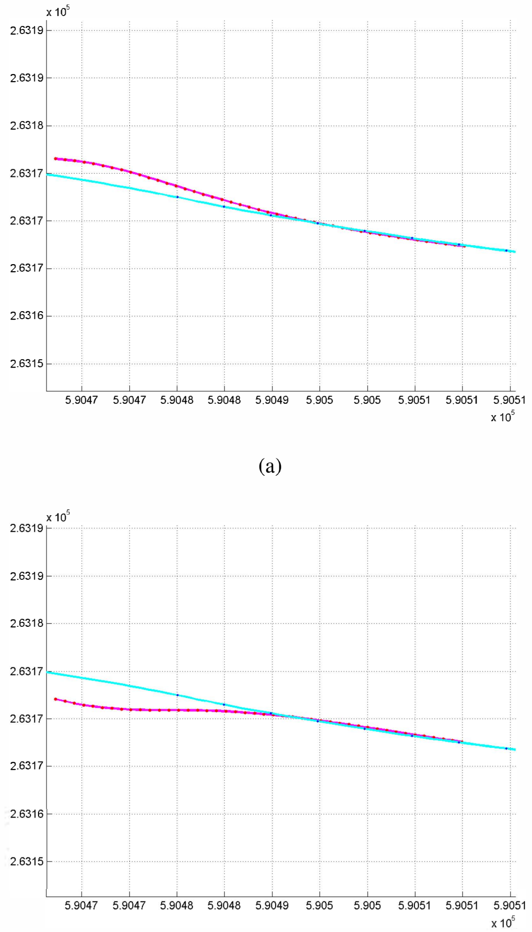
where w_i is the waypoint of a reference path P, D is a look-ahead distance, δ_c is the unit vector of a car-like mobile robot's heading direction, and G_{max} is the maximum steering angle of a car-like mobile robot. The look-ahead distance D is important for the pure-pursuit algorithm to determine appropriate control inputs to follow the reference path P smoothly. If the look-ahead distance is too short, the pure-pursuit algorithm induces oscillations. On the other hands, if the look-ahead distance is too long, the pure-pursuit algorithm can not follow the reference path. To determine the appropriate look-ahead distance, the following equation is used [6, 7]:

$$D(v(t)) = \begin{cases} D_{min} & \\ \quad \text{if } v(t) \le v_{lower} & \\ \alpha\cdot v(t) + \beta & \\ \quad \text{if } v_{min} < v(t) \le v_{upper} & \\ D_{max} & \\ \quad \text{if } v_{upper} < v(t) & \end{cases}, \tag{10}$$

where D_{min} and D_{max} are the minimum and maximum look-ahead distances, respectively, v_{lower} and v_{upper} are the corresponding upper and lower velocity of a car-like mobile robot, respectively, and α and β are coefficients to determine the look-ahead distance.

3. SIMULATION RESULTS

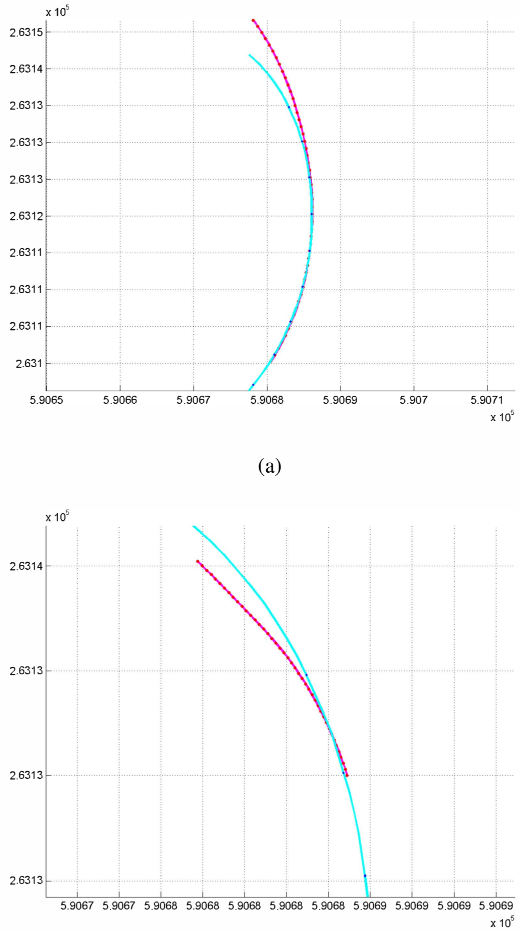
The simulations were conducted to verify that the proposed method can generate smooth trajectories to follow reference paths. The reference paths were acquired by



(a)

(b)

Fig. 4 Simulation results in straight road: Cyan line and magenta line show the reference path and generated trajectory, respectively.



(a)

(b)

Fig. 5 Simulation results in curved road: Cyan line and magenta line show the reference path and generated trajectory, respectively.

estimating the position of vehicle using GPS and IMU data [9, 10] . In the simulations, the initial states of a car-like mobile robot were set randomly around the reference paths to verify that the proposed method can generate trajectories to follow the reference paths by considering kinematic properties. The maximum steering angle G_{max} and the maximum steering angular velocity $\dot{G}_{max}$ were set as $0.349\ rad.$ and $0.175\ rad./s$, respectively, and the length of the trajectory was set as $5\ v(t)$.

Fig.4 shows simulation results of the proposed method in a straight section of the reference path. In this figure, the proposed method can generates the smooth trajectory to follow the reference path even though the initial position of the mobile robot is not on the reference path. Fig. 5 shows simulation results of the proposed method in a curved section of the reference path. This figure also shows that the proposed method can generates the smooth trajectory to follow the reference path. Fig. 6 shows simulation results of the proposed method in a highly-curved section of the reference path with a small curvature. This figure shows that the proposed method can generates the smooth trajectory to follow the reference path even though the reference path is highly-curved.

4. CONCLUSION

The proposed method generates smooth trajectories to follow a given reference path using the closed-loop prediction consists of a controller and motion model. Using the proposed method, smooth trajectories can be generated by considering the position of a mobile robot and its kinematic properties. The feasibility of the proposed method was verified by simulations in various situations.

The proposed method can be applied to local path planners to avoid unknown obstacles around a reference path.

REFERENCES

[1] T.Howard, C. Green, A. Kelly and D. Ferguson, ‘‘State Space Sampling of Feasible Motions for

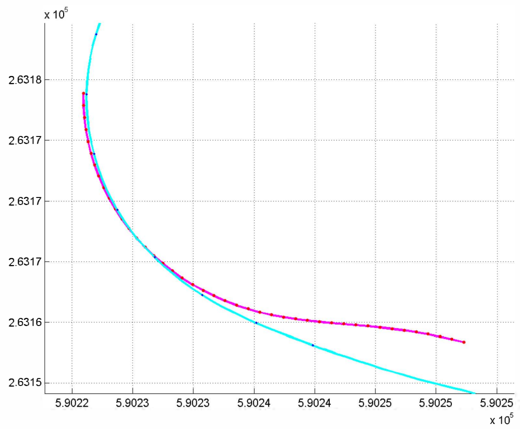


Fig. 6 Simulation result in highly-curved road with small curvature: Cyan line and magenta line show the reference path and generated trajectory, respectively.

High Performance Mobile Robot Navigation in Complex Environments,''Journal of Field Robotics. vol 25, no 6-7, pp. 325-345, 2008.

[2] S. Lee, C. Chun, T.-B. Kwon and S. Kang, "Bezier Curve-Based Path Planning for Robust Waypoint Navigation of Unmanned Ground Vehicle,'' *Journal of Institute of Control, Robotics and Systems*, vol. 17, no. 5, pp. 429-435, 2011.

[3] K. Yang and S. Sukkarieh, "Anytime Synchronized-Biased-Greedy Rapidly-exploring Random Tree Path Planning in Two Dimensional Complex Environments," *International Journal of Control, Automation, and Systems*, vol. 9, no. 4, pp. 750-758, 2011.

[4] U. Schwesinger, M. Rufli, P. Furgale and R. Siegwart, ''A Sampling-Based Partial Motion Planning Framework for System-Compliant Navigation along a Reference Path,'' In Proc. of IEEE Intelligent Vehicles Symposium, pp. 391–396, 2013.

[5] Y. Kuwata, J. Teo, G. Fiore, S. Karaman, E. Frazzoli and J. P. How, ''Real-time Motion Planning with Applications to Autonomous Urban Driving,'' IEEE Transactions on Control Systems Technology, vol. 17, no. 5, 1105 - 1118, 2009.

[6] S. Choi, J.-Y. Lee and W. Yu, ''Comparison between Position and Posture Recovery in Path Following'', in Proc. of International Conference on Ubiquitous Robot and Ambient Intelligence, 2009.

[7] J. M. Snider, ''Automatic Steering Methods for Autonomous Automobile Path Tracking,'' Tech Report CMU-RI-TR-09-08, Robotics Institute, Carnegie Mellon University, February, 2009.

[8] B. Park, Y. -C. Lee and W. Y. Han, ''Trajectory Generation Method Using Bezier Spiral Curves for High-Speed On-Road Autonomous Vehicles,'' to appear in proc. of IEEE International Conference on Automation Science and Engineering, 2014.

[9] M. Enkhtur, S. Y. Cho and K.-H. Kim, ''Modified Unscented Kalman Filter for a Multirate INS/GPS Integrated Navigation System,'' ETRI Journal, vol. 32, num. 5, pp. 943-946.

[10] S. Y. Cho and H. K. Lee, ''Modified RHKF Filter for Improved DR/GPS Navigation against Uncertain Model Dynamics,'' ETRI Journal, vol. 34, no. 3, pp. 379-387, 2012.