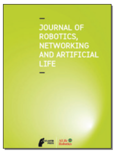
Journal of Robotics, Networking and Artificial Life
Vol. 7(3); December (2020), *pp.* 190–193
DOI: https://doi.org/10.2991/jrnal.k.200909.010; ISSN 2405-9021; eISSN 2352-6386
https://www.atlantis-press.com/journals/jrnal



Research Article

Apply Model-Free Adaptive Control Approach for Mobile Robot Path Following

Chia-Jen Lin[1], Chun-Chi Lai[2,*], Kuo-Hsien Hsia[2], Chin-Sheng Chen[1]

[1]*Graduate Institute of Automation Technology, National Taipei University of Technology, 1, Sec. 3, Zhongxiao E. Rd., Taipei, Taiwan 10608, R.O.C.*
[2]*Bachelor Program in Interdisciplinary Studies, National Yunlin University of Science and Technology, 123 University Road, Section 3,*
Douliou, Yunlin, Taiwan 64002, R.O.C.

ARTICLE INFO

Article History

Received 14 October 2019
Accepted 29 May 2020

Keywords

Mode-free adaptive control
dynamic window approach
path follower
mobile service robot

ABSTRACT

The state of the art of mobile robot path planning is composed with global planner and local planner. For example, the global planner majorly establishes a suitable path from a knowing map based on the shortest path. And the local planner which computes the velocity command that includes obstacle clearance and progress toward the goal. However, due to the 2D laser ranger on a mobile service robot just detects the part of the meal table or misses it, this will cause robot often hits the meal table. In this work we apply adaptive control method to feed the goals for Dynamic Window Approach's (DWA) computing. The experimental result shows that DWA will follow the pre-defined path closely and smoothly.

1. INTRODUCTION

With the rapid development of robotics and artificial intelligence technology, mobile service robots have begun to be used in our lives. Based on the design of the automatic navigation function, the mobile service robot is allowed to move autonomously and perform daily human services, such as automatic cleaning and automatic delivery. Based on Simultaneous Localization and Mapping (SLAM) [1], the autonomous navigation of mobile robot will become a reality. If exclude the SLAM, the remaining works of autonomous navigation are including path planning and path following.

The state of the art of mobile robot path planning is composed with global planner and local planner. For example, the global planner majorly establishes a suitable path from a knowing map. And the local planner not only follows the global path but also obstacle avoidance consideration. In this work, we apply the pre-defined path and design a path follower to replace the autonomous global planner to overcome the actual navigation problem when a mobile service robot is delivering the meal in a fast food restaurant.

2. MOBILE ROBOT PATH PLANNING

Path planning is the most basic part of mobile robot navigation. It refers to how the robot can find collision free waypoints as an appropriate motion path from the starting point to the end point in a working environment. In general, the path planning mainly involves these two major issues:

- Obstacles avoidance
- Optimizing the path

According to the environmental information, mobile robot path planning can be divided into global path planning and local planning as shown in Figure 1.

2.1. Global Planner

The global planner is a pre-planning method that depends on the analysis of known map. For example, the artificial potential field [2] algorithm was proposed in 1989. Its' concept is to apply a virtual attraction and repulsive force to a robot. The target gives the robot attractive force, and the obstacle is the repulsive force. After combining the force, a moving path of the robot to the target point can be obtained. The Rapid Exploration Random Tree [3] method was originally proposed in 1998. Its' idea is to growth with the initial point as the root in the space and add child nodes by random sampling. Connect the closest child node to the root node until the child gradually reaches the target point, and then a feasible path could be selected. The other well-known methods of safe path creation are such as the generalized Voronoi diagram [4] and the Probabilistic Roadmaps (PRMs) [5]. The PRMs applies a sampling to the environment in order for a graph to be created, each edge of which is considered safe for robot traversing and a graph search algorithm such as Dijkstra [6] or A* [7] is applied to the resulting graph to determine a path between the starting and goal.

[*] Corresponding author. Email: cclai@yuntech.edu.tw

In nowadays, the occupancy grid map [8] is popular to represent the posterior probability model of environment. And the graph search algorithm Dijkstra or A* is also suit for finding the optimal path which is a collection of cells. After the success of finding the global path from the start to the goal, all the selected cells are translated into positions as the point vector.

2.2. Local Planner

In order to transform global path into a suitable waypoints, local planner applies sensors to detect the robot's environment to obtain pose or geometric information of obstacles. Therefore, using the updated local sensing information, the local planner generates an obstacle avoidance strategy and attempts to match the trajectory provided by the global planner. Some local trajectory generation of local planner may apply Bezier lines [9], arcs, segments [10], or splines [11]. In nowadays, one of the most popular local path plan method is Dynamic Window Approach (DWA) [12], the basic idea of the DWA algorithm is as follows:

- The DWA method is a local response obstacle avoidance controller, which searches for the best action (translation velocity v and angular velocity ω) in the velocity space.
- Considering the dynamic constraints of the robot to reduce the sample space.
- The trajectories corresponding to different velocities can be represented by a series of arcs with different curvatures.
- In order to obtain a fast response, it is assumed that the speed of the robot is constant in all future intervals.

The sampled trajectories are evaluated by an objective function $O(v, \omega)$, which is consisted with the goal heading, obstacle clearance, and velocity as below criteria:

$$\begin{aligned} O(v,\omega) = \sigma(\alpha \times \text{heading}(v,\omega) + \beta \times \text{dist}(v,\omega) \\ + \gamma \times \text{velocity}(v,\omega)) \end{aligned} \tag{1}$$

The heading (v, ω) is the angle difference between robot current head orientation and robot to goal direction, the dist (v, ω) is the distance to the closest obstacle, and the velocity (v, ω) presents the forward velocity of the robot with kinematic constraints.

2.3. Problem in Actual Navigation

In practical applications, the navigation package released in ROS [13] framework are applied to an automatic meal delivery robot in a fast food restaurant. Due to the pedestal column of meal table in the fast food restaurant is shown in Figure 2. Although this pedestal column could be sensed via the laser ranger equipped on robot platform, but sometimes it misses the detection and the upper body of robot hits the corner of table that causes the meal deliver task to be failed. In this work, we apply the pre-define path and path follower to replace the autonomous global planning to overcome the actual navigation problem which causes by the meal table.

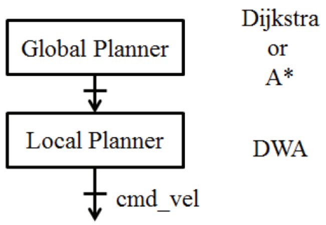


Figure 1 Mobile robot path planning flow diagram.

3. PATH FOLLOWER DESIGN

There are four sequences for path follower:

- Get the pre-defined path from file or Topic.
- Prune the Waypoints: Erase the unsuitable waypoints in the path.
- Segment Regulation: Adjust the segment length between waypoints.
- Send out or publish the modified path for DWA.

3.1. Prune the Waypoints

The first work in prune the waypoint is to check if the waypoint should be discarded. Assume for a given path with five waypoints $P_0 \sim P_4$, if the initial waypoint P_0 is in front of robot as shown in Figure 3a the angle θ between $\bar{v}_1$ and $\bar{v}_2$ will greater than $\pi/2$, where $\bar{v}_1$ is the vector of robot pose and P_0 and $\bar{v}_2$ is the vector of P_0 and P_1. On the other hand, if the angle θ between $\bar{v}_1$ and $\bar{v}_2$ are smaller than $\pi/2$ as shown in Figure 3b, then the P_0 point will be deleted and the point data will be recorded as P_{past}. In Figure 3c, L_1 is the distance between robot center pose and P_0. L_2 is distance between robot center pose and P_{past}. When $L_1 > L_2$ but robot cannot turn to P_0 due to the local obstacle. In this situation the P_0 point will also be deleted and the point data will be recorded as P_{past}.

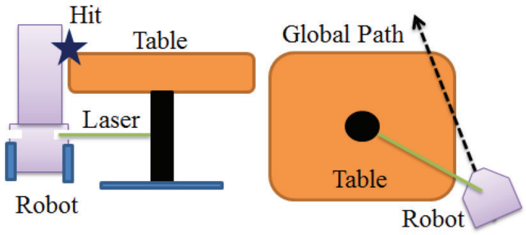


Figure 2 Auto global path planning problem: the shortest global path often causes mobile service robot to hit the meal tablet.

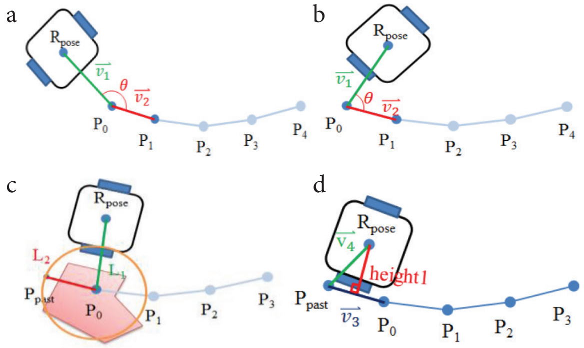


Figure 3 Prune the waypoints of default path.

3.2. Segment Regulation

At this stage, we need an indicator as the criterion for judging how close that robot is following the original path. Calculate the vertical distance "height1" as shown in Figure 3d between the robot pose and the segment of P_{past} to P_0 as the performance criteria for following the line. The "height1" distance can be calculated as below equation:

$$\vec{v}_3 \times \vec{v}_4 = |\vec{v}_3||\vec{v}_4|\sin\theta \tag{2}$$

$$\text{height1} = \frac{\vec{v}_3 \times \vec{v}_4}{|v_3|} \tag{3}$$

3.2.1. *Model-free adaptive control*

Model-free adaptive control was proposed by Hou and Huang [14] which can realize the parameter adaptive control and structure adaptive control of the controlled system which makes it more applicable for many practical plants.

$$u(k) = u(k-1) + \frac{\rho\phi(k)}{\lambda + \phi^2(k)}(y \ast (k+1) - y(k)) \tag{4}$$

$$\phi(k) = \phi(k-1) + \frac{\eta\Delta u(k-1)}{(\mu + \Delta u^2(k-1))(\Delta y(k) - \phi(k-1)\Delta u(k-1))} \tag{5}$$

$$\phi(k) = \phi(1), \text{ if } \phi(k) \le \varepsilon \text{ or} |\Delta u(k-1)| \le \varepsilon \tag{6}$$

From the above Eqs. (4)–(6), consider the previously calculated height1 as $y(k)$ and $u(k)$ as control variables to regulate the length of the path.

4. EXPERIMENTAL RESULTS

The experiment is firstly to simulate in ROS rviz user interface for verifying the path following result. In Figure 4a, the predefined path is shown as a green circle with a radius of 1 m. In Figure 4b, the blue circular path represents the smooth following result. Figure 4c shows the path following error with the time index. The maximum distance error of following the circle is about 0.0562 m and the mean distance error is about 0.0238 m as shown in Table 1. Besides, Figure 5a shows the service robot in a real fast-food restaurant, and the pre-defined green path is established to move around the restaurant as shown in Figure 5b. The service robot follows the path well and finally reaches the pre-defined goal as shown in Figure 5c to 5d.

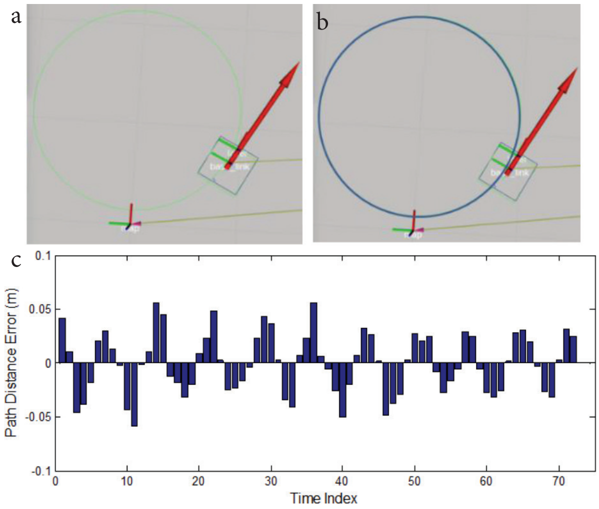


Figure 4 Robot path following simulation: (a) circular path generation, (b) circular path following simulation, (c) path following error.

Table 1 The circular path following error

Maximum distance error (m)	Mean distance error (m)
0.0562	0.0238

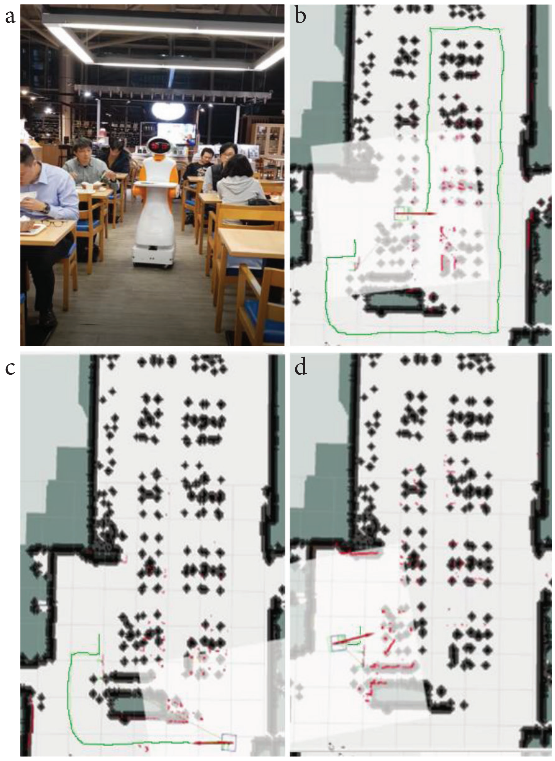


Figure 5 Robot path following demo: (a) service robot in a real fast-food restaurant, (b) service robot starts to follow a long path, (c) robot turn in right-down corner, (d) robot will reach final goal.

5. CONCLUSION

The state of the art of mobile robot path planning is composed with global planner and local path planner. However for meal deliver application in a real fast food restaurant, the mobile service robot often hits the corner of the meal table due to the 2D laser ranger just detects the part of the meal table or misses it. In this work, we apply the pre-defined path and path follower to replace the autonomous global planning to overcome the actual navigation problem which causes by the meal table. The experimental result shows that DWA can also follow the pre-defined path closely and smoothly.

CONFLICTS OF INTEREST

The authors declare they have no conflicts of interest.

REFERENCES

[1] H. Durrant-Whyte, T. Bailey, Simultaneous localization and mapping: part I, IEEE Robot. Autom. Mag. 13 (2006), 99–110.

[2] O. Khatib, Real-time obstacle avoidance for manipulators and mobile robots, Proceedings. 1985 IEEE International Conference on Robotics and Automation, IEEE, St. Louis, MO, USA, 1985, pp. 500–505.

[3] S.M. LaValle, Rapidly-exploring random trees: a new tool for path planning, Computer Science Dept., Iowa State University, 1998.

[4] S. Garrido, L. Moreno, M. Abderrahim, F. Martin, Path planning for mobile robot navigation using Voronoi diagram and fast marching, 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, Beijing, China, 2006, pp. 2376–2381.

[5] L.E. Kavraki, M.N. Kolountzakis, J.C. Latombe, Analysis of probabilistic roadmaps for path planning, IEEE Trans. Robot. Autom. 14 (1998), 166–171.

[6] D.E. Knuth, A generalization of Dijkstra's algorithm, Inform. Process. Lett. 6 (1977), 1–5.

[7] P.E. Hart, N.J. Nilsson, B. Raphael, A formal basis for the heuristic determination of minimum cost paths, IEEE Trans. Syst. Sci. Cyber. 4 (1968), 100–107.

[8] S. Thrun, A. Bücken, Integrating grid-based and topological maps for mobile robot navigation, Proceedings of the Thirteenth National Conference on Artificial Intelligence (AAAI), Portland, OR, 1996, pp. 944–950.

[9] J.P. Rastelli, R. Lattarulo, F. Nashashibi, Dynamic trajectory generation using continuous-curvature algorithms for door to door assistance vehicles, 2014 IEEE Intelligent Vehicles Symposium Proceedings, IEEE, Dearborn, MI, USA, 2014, pp. 510–515.

[10] J.A. Reeds, L.A. Shepp, Optimal paths for a car that goes both forwards and backwards, Pac. J. Math. 145 (1990), 367–393.

[11] A. Piazzi, C.G. Lo Bianco, M. Bertozzi, A. Fascioli, A. Broggi, Quintic G^2-splines for the iterative steering of vision-based autonomous vehicles, IEEE Trans. Intell. Transport. Syst. 3 (2002), 27–36.

[12] D. Fox, W. Burgard, S. Thrun, The dynamic window approach to collision avoidance, IEEE Robot. Autom. Mag. 4 (1997), 23–33.

[13] Available from: http://wiki.ros.org/move_base.

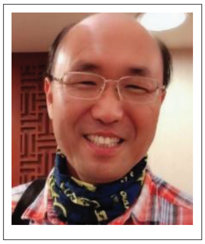
[14] Z.S. Hou, W. Huang, The model-free learning adaptive control of a class of SISO nonlinear systems, Proceedings of the 1997 American Control Conference, IEEE, Albuquerque, NM, USA, 1997, pp. 343–344.

AUTHORS INTRODUCTION

Mr. Chia-Jen Lin

He received the M.S. degree in 2004 from National Yang-Ming University, Taipei, Taiwan, where he is currently working toward the PhD degree in National Taipei University of Technology, Taipei, Taiwan. He is also a R&D manager in TECO Group Research Institute with TECO Electric and Machinery Co., Ltd. His research interests are AGV control and service robot techniques.

Dr. Kuo-Hsien Hsia



He received the PhD degree in Electrical Engineering from the National Sun Yat-Sen University, Taiwan, in 1994. He is currently an Associate Pofessor of National Yunlin University of Science and Technology in Taiwan. His research interests are in the area of artificial intellgience, fuzzy control and image processing.

Dr. Chun-Chi Lai

He received the PhD degree in Electrical Engineering from National Chung Cheng University, Chiayi, Taiwan, in 2014. He is currently an Assistant Professor of National Yunlin University of Science and Technology in Taiwan. His research interests include multisensor fusion and intelligent robotics.

Dr. Chin-Sheng Chang

He received the PhD degree in Mechanical Engineering from National Chiao Tung University, Hsinchu, Taiwan, in 1999. In 2002, he joined the Graduate Institute of Automation Technology, National Taipei University of Technology, Taipei, Taiwan, as an Assistant Professor, where he is currently a Professor and the Director. He has authored or coauthored more than 150 journal and conference papers and book chapters on the research. His research interests include intelligent motion control, robotics, mechatronics, and machine vision. Dr. Chen was recipient of the Outstanding Research Award from the College of Mechanical and Electrical Engineering, Taipei TECH, in 2013and 2014.