Development of Point to Point Algorithm AMR Navigation

Minh Khong, Van-Luc Ngo, Minh-Hiep Le, Van-Tung Nham, and Quoc-Dat Do

Abstract AMR (Autonomous Mobile Robots) are used popular increasingly, especially in industry to solve logistical problems in factories. The navigation algorithm affects not only the motion characteristics, but also the specifications of AMR. The simpler the algorithm, the smaller the computational volume, the less demanding the hardware, and the lower the cost of AMR. The smoother the movement, the less energy is consumed and the larger the cargo capacity. By simulation, this study will analyze the characteristics of some algorithms currently being used for AMR such as Pure pursuit or Follow the Carrot Algorithms. The advantages and disadvantages of each algorithm will be analyzed to proposing a new algorithm that can overcome the disadvantages of the algorithms being used for AMR navigation. The algorithm was developed in this research has been tested on Phenikaa-X AMR, sensory evaluation gives better results.

Keywords AMR · Autonomous mobile robots · Point to point · Navigation algorithm · Logistics in factories · Pure pursuit · Follow the carrot

1 Introduction

The problem of control a AMR moving from point to point is one of the most common problems in industry such as moving from a station to a work place or performing logistical tasks between two specified location in factories. There are

M. Khong · V.-L. Ngo (✉)
Faculty of Mechanical Engineering and Mechatronics, Phenikaa University, Hanoi 12116, Vietnam
e-mail: luc.ngovan@phenikaa-uni.edu.vn

V.-T. Nham · Q.-D. Do
Faculty of Vehicle and Energy Engineering, Phenikaa University, Hanoi 12116, Vietnam

M. Khong · V.-L. Ngo · M.-H. Le · V.-T. Nham · Q.-D. Do
Phenikaa Research and Technology Institute (PRATI), A&A Green Phoenix Group JSC, No. 167 Hoang Ngan, Trung Hoa, Cau Giay, Hanoi 11313, Vietnam

B. T. Long et al. (eds.), *Proceedings of the 3rd Annual International Conference on Material, Machines and Methods for Sustainable Development (MMMS2022)*, Lecture Notes in Mechanical Engineering, https://doi.org/10.1007/978-3-031-57460-3_25

many algorithms that can be applied to control AMR move point to point, such as Pure Pursuit [1], Follow the Carrot [2], Stanley, MPC, etc. Each algorithm has its own advantages and disadvantages and is suitable for a type of AMR structure. The model of AMR in this research is a robot that uses two independent control wheels on both sides to navigate and adjust speed based on changing the speed of each wheel. The Pure Pursuit and Follow the Carrot algorithm which is described in Fig. 1 are more suitable to navigate this type of AMR. By simulation, this research will analyzed motion characteristics of this type of AMR using Follow the Carrot algorithm to develop a better navigation algorithm.

Pure Pursuit is a navigation algorithm for AMR in particular and mobile devices moving in general, first introduced in 1992 [1]. Until now, this algorithm is still an effective tracking algorithm used in navigation and control for autonomous vehicles [3–6].

Figure 1a shows the geometry of Pure Persuit algorithm (a). With this algorithm, the robot will move on a circular path with radius (r) and depends on the look ahead distance (l), from the geometry on Fig. 1 the relationship between r and l can be determined as follows:

$$x^2 + y^2 = l^2\, and\, x + d = r \Rightarrow r = \frac{l^2}{2x}$$

If the lookahead distance (l) is small, the moving robot has a small deviation from the target path but fluctuates more than the large value (l) and vice versa. If the look ahead distance (l) is reduced to allow the robot to close the target path, the robot will fluctuate a lot as shown in Fig. 2, which may affect the wares it carries. Conversely, if look ahead distance (l) is increase to reduce oscillation, the robot will not move closed to the target path and may deviate from the space that allows the robot to operate. Thus, this algorithm is not really suitable for robots operating in factories to do logistical tasks in some case.

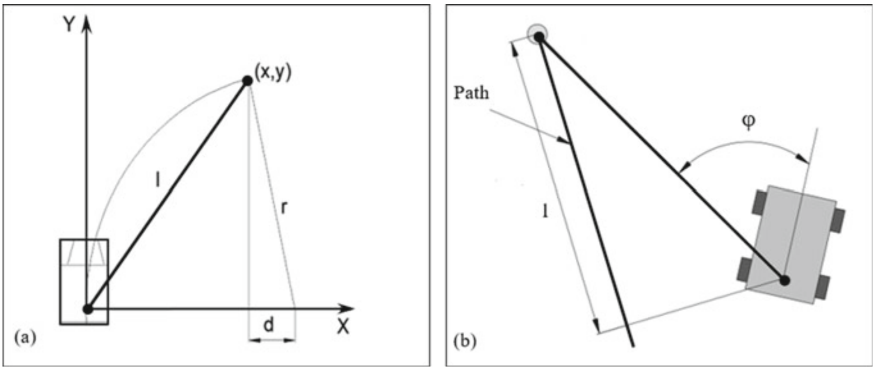


Fig.1 Algorithm geometry of Pure Pursuit (**a**) and Follow the Carrot (**b**) [1, 2]

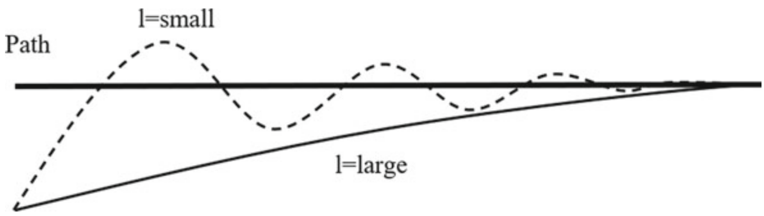


Fig. 2 The response of the Pure Pursuit algorithm tracker [1]

Compared with the Pure Pursuit algorithm, the Follow the Carrot algorithm has a smaller computational volume because it does not have to perform the coordinate system conversion operations, but it requires a device to determine the direction of the robot at the current location [1]. With the algorithm of Follow the Carrot, the robot always tends to turn towards the goal point by calculating the angle deviation of direction (φ) between the robot direction and the direction to the goal point before moving towards to the goal point as described in Fig. 1b. Theoretically, the Follow the Carrot algorithm can give more accurate results. However, experiments have shown that due to inertia, accuracy of the system, etc., the robot tends to rotate over the direction to the goal point and the motion is wobbly when traveling at the high of velocity [2].

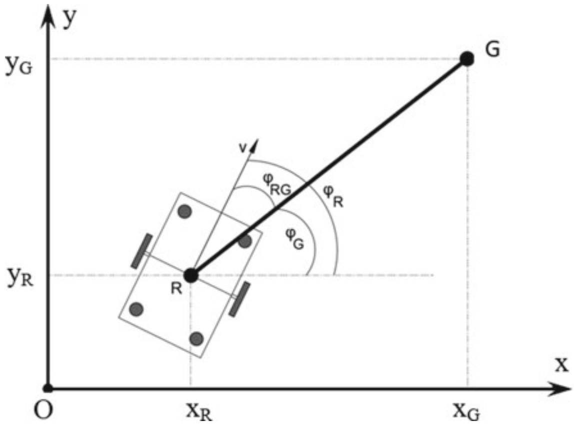
The above analysis shows that with the Pure Pursuit algorithm the computation volume is high and the fluctuations are large or larger deviated to the target path. For the Follow the Carrot algorithm, although the calculation volume is smaller, but it requires equipment to determine the direction of the robot at the current position and still vibrates when moving at high velocity. This study will propose an algorithm developed on the idea of taking advantage and limiting the disadvantages of Pure Pursuit an Follow the Carrot algorithm.

2 Development of Point to Point Navigation Algorithm

The proposed algorithm in this study comes from the idea of developing the advantages and limiting the disadvantages of the Pure Pursuit an Follow the Carrot algorithm to achieve the goal of better control and reduce the cost of AMR. Accordingly, this algorithm will control the robot to move with large of look ahead distance to get smoother movement. In order to reduce the error of the robot trajectory and the target path, both linear and angular velocity of robot are controlled for the robot to move gradually to close to the target path. With the this algorithm the robot also does not need equipment to determine the direction of the robot at the current position, Fig. 3 show geometry of proposal algorithm.

Base on the geometry of proposal algorithm, some parameter can be calculated:

Fig. 3 Geometry of proposal algorithm



- The distance from the robot to the goal point G: $l_{RG} = \sqrt{(x_G - x_R)^2 + (y_G - y_R)^2}$.
- Deviation angle between the vehicle direction and the line connecting the vehicle to the goal point: $\varphi_{RG} = \varphi_R - \varphi_G = \varphi_R - a\tan(\frac{y_G - y_R}{x_G - x_R})$.

The linear and angular velocity (v, w) should be limited to not exceed the allowable value (v_{max}, w_{max}) and the magnitude is proportional to the distance from the robot position (R) to the goal point (G).

Choose the motion control law for (v) and (w) of the robot is:
$$\begin{cases} v = v_{max} sin\left(\frac{l_{RG}}{l_{max}}\frac{\pi}{2}\right) \\ w = w_{max}\sin\left(\frac{\varphi_{RG}}{\varphi_{max}}\frac{\pi}{2}\right) \end{cases}$$
The meaning of the above parameters is explained as follows:

- l_{RG} is the distance from the current point of robot (R) to the goal point (G).
- l_{max} is the maximum of l_{RG} or l_{max} is the distance from the start point of robot to goal point (G).
- $\varphi_R, \varphi_G \, and \, \varphi_{RG}$ are explained and illustrated in Fig. 3 and φ_{max} is maximum of φ_{RG}.

The linear and angular velocity of robot will gradually decrease as the robot reaches the goal point (G) and does not exceed the allowed value (v_{max}, w_{max}). The more moving, the more w decreases and the robot direction closes to the target path. At the time robot reaches the goal point (G) then $l_{RG} = 0 and \varphi_{RG}$ so v = 0 and w = 0, in other words the robot will stop at the goal point (G) in the direction of the target path.

To evaluate the purpose response of the proposed algorithm, this study will perform simulations with the following example parameters:

- The coordinates of the starting point of the robot: $\begin{cases} x_R = 0(m) \\ y_R = 0(m) \\ \varphi_R = 90^\circ \end{cases}$
- The coordinates of the goal point of the robot: $\begin{cases} x_G = 2(m) \\ y_G = 3(m) \end{cases}$

3 Results and Discussion

Figure 4 shows the movement of the robot with the proposal and Pure Pursuit algorithm. This result shows that the robot controlled by the proposed algorithm follows the target path more closely and stops when it reaches the goal point. In contrast, the robot controlled by the Pure Pursuit algorithm is more deviated from the target path and it did not stop at the goal point. The variation of l_{RG} with robot used the proposal and Pure Pursuit algorithm as shown in Fig. 5 indicate that that the robot controlled by the Pure Pursuit algorithm reached the goal point earlier at the time of 4.2 s, and then went over it again because it can stopped. The robot controlled by the proposal algorithm arrives at the goal point later, but the robot is getting more and more closer to the goal point. The proposal with the control law of the robot as proposed above, both the angular velocity (w) and the linear velocity (v) decrease with time, making the robot reach the goal point later.

Figures 6 and 7 show the comparison of the linear and angular velocity variation when robot is controlled by proposal and Pure Pursuit algorithm. The angular velocity of the proposal algorithm change faster in the first period, resulting in the robot quickly tracking to the path direction. This explains why the robot controlled by

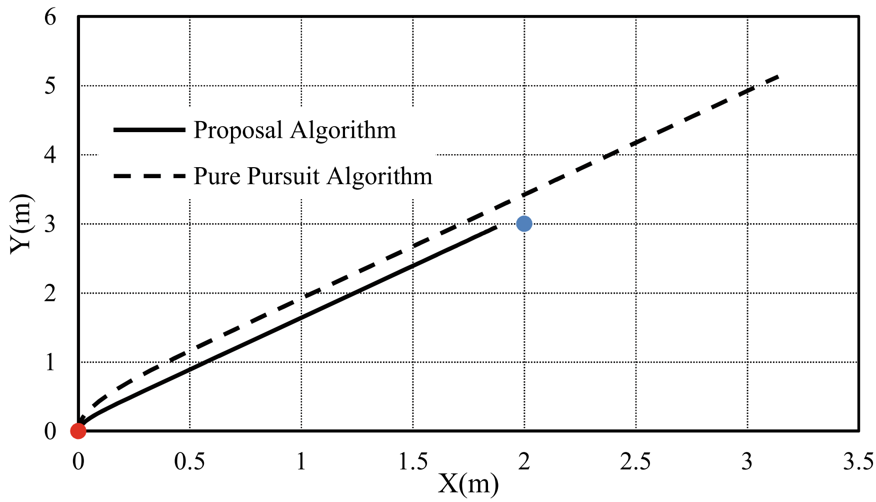


Fig.4 The movement of the robot with the proposal and Pure Pursuit algorithm

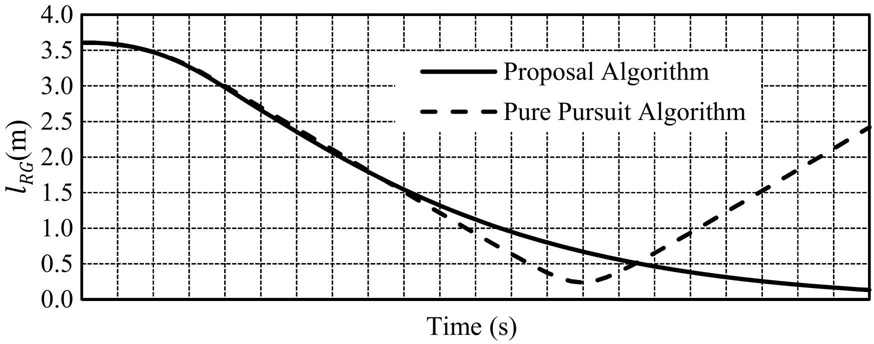


Fig.5 The variation of l_{RG} of the proposal and Pure Pursuit algorithm

the proposal algorithm closed to the target path compared to that controlled by pure pursuit algorithm. When robot is controlled by both algorithms then angular velocity is fast to zero but proposal algorithm driven angular velocity goes faster. This result shows that the proposal algorithm is better the Pure Pursuit algorithm in driven robot direction. Figure 7 shows that the linear velocity controlled by the proposal algorithm varies smoother than that controlled by the Pure Pursuit algorithm and steadily decreases towards zero as the robot approaches the goal point. The curve of linear velocity variation controlled by pursuit algorithm has a folding position. This feature indicates that the robot will be jerked at this position.

The proposal algorithm was tested to navigate the AMR which developed by Phenikaa-X company as shown in Fig. 8. Sensory evaluation gives better results such as moother speed change, AMR stops at target position. This is result of the velocity in the proposal algorithm varies with the sine function while the velocity in the PP and FTC algorithm is constant. The AMR stops at target position also consistent with the analysis above.

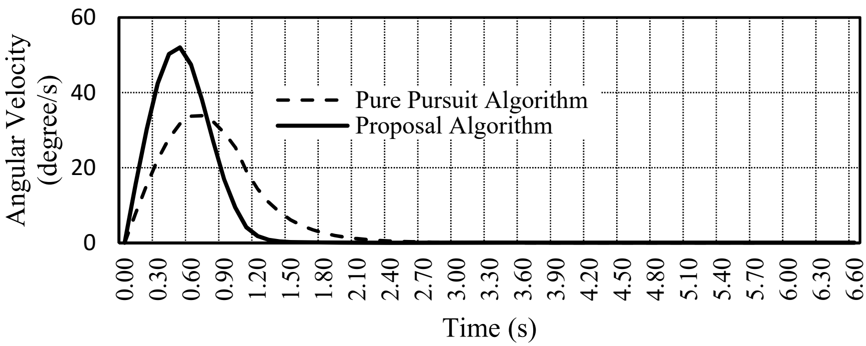


Fig. 6 The variation of angular velocity of the proposal and Pure Pursuit algorithm

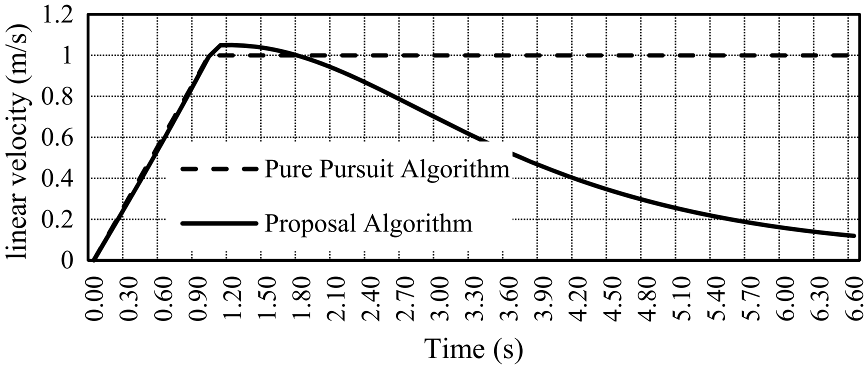


Fig. 7 The variation of linear velocity of the proposal and Pure Pursuit algorithm

Fig. 8 The AMR was developed by Phenikaa-X Company

4 Conclusions and Recommendations

This study has proposed a new algorithm to control AMR to move in point to point in straight path. The proposal algorithm is developed from those commonly used in AMR control. Simulation results have shown that the proposed algorithm controls the AMR better in tracking the path, navigation, and the robot can stop when it reaches the goal point. With this algorithm, the robot does not need a device to determine the direction of the robot at the current position as the Follow the Carrot algorithm. However, this study only consider base on theoretical analysis and simulation. The proposal algorithm was tested to navigate the real AMR and gives better results such as moother speed change, AMR stops at target position, these are the goals of this research. Further research should analyzed by experiment and compare with simulation results to have more accurate conclusions about the advantages of the proposed algorithm. The analysis and evaluation of the results also need to be analyzed more detailed such as the parameters of oscillation, jerking, productivity.

References

1. R. Craig Conlter, *Implementation of the Pure Pursuit Path Tracking Algorithm* (Robotics Institute Camegie Mellon University, 1992)
2. M. Lundgren, *Path Tracking for a Miniature Robot.* Excerpt from Master's thesis 2003 Supervisor: Thomas Hellström (2003)
3. S. Dominguez-Quijada, A. Ali, G. Garcia, P. Martinet, Comparison of lateral controllers for autonomous vehicle: experimental results, in *IEEE 19th International Conference on Intelligent Transportation Systems (ITSC 2016)* (2016)
4. W. Naeem, R. Sutton, S.M. Ahmad, *Pure Pursuit Guidance and Model Predictive Control of an Autonomous Underwater Vehicle for Cable/Pipeline Tracking* (Marine and Industrial Dynamic Analysis Research Group Department of Mechanical and Marine Engineering, 2003)
5. T. Hellström, O. Ringdahl, Follow the past: a path-tracking algorithm for autonomous vehicles. Int. J. Veh. Auton. Syst **4** (2006)
6. G. Klancar, A. Zdesar, S. Blazic, I. Skrjanc, *Wheeled Mobile Robotics* (Butterworth-Heinemann is an imprint of Elsevier, 2017)