IJRCS

International Journal of Robotics and Control Systems
Vol. 3, No. 2, 2023, pp. 161-170
ISSN 2775-2658
http://pubs2.ascee.org/index.php/ijrcs

ASCEE

Artificial Potential Field Path Planning Algorithm in Differential Drive Mobile Robot Platform for Dynamic Environment

Maulana Muhammad Jogo Samodro [a,1], Riky Dwi Puriyanto [a,2*], Wahyu Caesarendra [b,c,3]

[a] Department of Electrical Engineering, Universitas Ahmad Dahlan, Indonesia
[b] Faculty of Integrated Technologies, Universiti Brunei Darusalam, Jalan Tungku Link, BE1410 Brunei Darussalam
[c] Faculty of Mechanical Engineering, Opole University of Technology, 76 Proszkowska St., 45-758 Opole, Poland
[1] jogo.samodro@gmail.com; [2] rikydp@ee.uad.ac.id; [3] wahyu.caesarendra@ubd.edu.bn
* Corresponding Author

ARTICLE INFO

Article history
Received February 12, 2023
Revised March 19, 2023
Accepted March 20, 2023

Keywords
Mobile robot;
Artificial Potential Field;
Kinematics model;
Path planning;
Dynamic environment

ABSTRACT

Mobile robots need path-planning abilities to achieve a collision-free trajectory. Obstacles between the robot and the goal position must be passed without crashing into them. The Artificial Potential Field (APF) algorithm is a method for robot path planning that usually used to control the robot for avoiding obstacles in front of the robot. The APF algorithm consists of an attractive potential field and a repulsive potential field. The attractive potential fields work based on the predetermined goals that generated to attract the robot to achieve the goal position. Apart of it, the obstacle generates a repulsive potential field to push the robot away from the obstacle. The robot's localization in producing the robot's position is generated by the differential drive kinematic equations of the mobile robot based on encoder and gyroscope data. In addition, the mapping of the robot's work environment is embedded in the robot's memory. According to the experiment's results, the mobile robot's differential drive can pass through existing obstacles. In this research, four test environments represent different obstacles in each environment. The track length is 1.5 meters. The robot's tolerance to the goal is 0.1 m, so when the robot is in the 1.41 m position, the robot's speed is 0 rpm. The safe distance between the robot and the obstacle is 0.2 m, so the robot will find a route to get away from the obstacle when the robot reaches that safe distance. The speed of the resulting robot decreases as the distance between the robot and the destination gets closer according to the differential drive kinematics equation of the mobile robot.

1. Introduction

The development of technology has progressed from time to time. The Industrial Revolution 4.0 is a significant achievement that has been made in the industrial world. Robotics was first developed by Computer Aided Manufacturing International (CAM-1). A robot is a device capable of performing functions performed by humans or equipment capable of working with intelligence similar to that of humans. The mobile robot is one field in the world of robotics that can move according to existing

http://dx.doi.org/10.31763/ijrcs.v3i2.944

environmental conditions. The mobile robot propulsion system uses several types, including differential, tricycle, synchronous, and holonomic drives.

Differential drives mobile robot (DDMR) is a type of robot that utilizes the speed of the left and right wheels of the robot. The DDMR is typically called nonholonomic because it consists of 2 main wheels which limit the robot to move in all directions [1]. DDMR is equipped with 2 ball casters in front and behind the robot, which function as a counterweight. The DDMR will go straight when both wheels rotate at the same speed. In addition, the DDMR will move in a curve with the direction of the track toward one of the slower-moving wheels.

Autonomous DMR requires navigation capabilities to complete its task toward the goal position. The required navigation can include mapping, localization, and path planning capabilities. Path-planning is one of the navigational capabilities that an autonomous DDMR needs to have. Path-planning is a method for determining the direction of motion of a robot [2]. Robots using the path planning method can decide and know the direction of movement to avoid collisions so they can reach their destination. In addition, path planning is needed for the robot to carry out environmental recognition for all information about obstacles and goals that can be known a priori [3], [4].

One of the path-planning algorithms that has a simple but reliable equation is the Artificial Potential Field (APF). The Artificial Potential Field algorithm is one of the algorithms used for path planning by utilizing artificial attractive and repulsive potential fields. The goal position generates an artificial attractive potential field, while the obstacle generates an artificial repulsive potential field. Therefore, the robot is able to go to the goal position while avoiding collisions with obstacles [5]. Robots that detect obstacles will produce a safe path to the goal position [6].

The APF algorithm is implemented in autonomous robots to improve driving safety [7], [8], [9]. Mobile robots are one type of robot that uses the APF algorithm in navigation systems [10], [11]. However, this algorithm has drawbacks i.e. it can be trapped in local minimum conditions [12], [13], [14], [15], [10], [11]. One potential method to solve the local minimum problem is by modifying the robot's position heading to obtain the next position coordinate.

Generally, the APF algorithm is tested through simulation on a particular robot. The APF algorithm has the advantage of being able to be implemented on mobile robots in real-time [16], [17], [18], [19], [20]. Integrating the APF algorithm on a real robot platform is needed to test the effectiveness of the robot in moving towards goals autonomously. The robot must know the environmental information around the robot to complete its task. In addition, the robot kinematics model used also needs to be integrated with the APF algorithm so that the robot runs appropriately.

This study contributes to integrating the APF algorithm on the DDMR platform. Environmental information around the robot is embedded in the robot's memory to get better results. Tests are carried out to see the robot's success in avoiding obstacles.

This paper consists of several sections. The first section presents the introduction. The second section is an explanation of the research method. The third section shows the result of the test. The fourth section is the conclusion of the research.

2. Method

2.1. Artificial Potential Field Algorithm

The Artificial Potential Field (APF) algorithm is a classic method that prevents robots from existing obstacles and decides which path the robot will pass. Two types of potential fields make up the APF algorithm, namely attractive potential fields and repulsive potential fields. The goal position generates an attractive potential field to attract the robot closer to the goal. The repulsive potential field is generated as an obstacle to keep the robot away. The equation for the APF potential field ($U_{APF}(x,y)$) can be seen in (1).

$$U_{APF}(x,y) = U_{att}(x,y) + U_{rep}(x,y) \tag{1}$$

where $U_{att}(x,y)$ and $U_{rep}(x,y)$ are attractive and repulsive potential fields of the APF algorithm, respectively. The attractive potential field equation ($U_{att}(x,y)$) can be seen in (2).

$$U_{att}(x,y) = \frac{1}{2}k_{att}D_g^2(x,y) \tag{2}$$

where $D_g(x,y)$ is Euclidean distance between the robot position and the goal position. Euclidean distance of $D_g(x,y)$ can be seen in (3). The k_{att} parameter is the attractive gain to determine the magnitude of the attractive field generated by the goal.

$$D_g(x,y) = \sqrt{\left(x - x_{goal}\right)^2 + \left(y - y_{goal}\right)^2} \tag{3}$$

The repulsive potential field equation ($U_{rep}(x,y)$) can be seen in (4).

$$U_{rep}(x,y) = \begin{cases} \frac{1}{2}k_{rep}\left(\frac{1}{D_o(x,y)} - \frac{1}{r}\right), & D_o(x,y) \leq r \\ 0, & D_o(x,y) > r \end{cases} \tag{4}$$

where $D_o(x,y)$ is Euclidean distance between the robot position and the obstacle position. Euclidean distance of $D_o(x,y)$ can be seen in (5). The k_{rep} parameter is the negative repulsive gain to determine the magnitude of the repulsive field generated by the obstacle. Parameter r indicates the specified safe distance between the robot and the obstacle. According to (4), the repulsive potential field will be active when the robot is very close to an obstacle ($D_o(x,y) \leq r$). The robot will not get a repulsive potential field at a safe distance ($D_o(x,y) > r$).

$$D_o(x,y) = \sqrt{(x - x_{obs})^2 + (y - y_{obs})^2} \tag{5}$$

The APF algorithm aims to get the minimum value at the goal position. Therefore, a negative gradient of $U_{APF}(x,y)$ in Equation (6) is required to get the minimum value.

$$F_{APF}(x,y) = -\nabla U_{att}(x,y) \tag{6}$$

The derivative function of $U_{APF}(x,y)$ on the x and y axes can be seen in Equations (7) and (8), respectively.

$$F_x(x,y) = \begin{cases} -k_{att}\left(x - x_{goal}\right) - \dfrac{\frac{1}{2}k_{rep}\left(\frac{1}{D_o(x,y)} - \frac{1}{r}\right)(x - x_{obs})}{D_o^3(x,y)}, & D_o(x,y) \leq r \\ -k_{att}\left(x - x_{goal}\right) , & D_o(x,y) > r \end{cases} \tag{7}$$

$$F_y(x,y) = \begin{cases} -k_{att}\left(y - y_{goal}\right) - \dfrac{\frac{1}{2}k_{rep}\left(\frac{1}{D_o(x,y)} - \frac{1}{r}\right)(y - y_{obs})}{D_o^3(x,y)}, & D_o(x,y) \leq r \\ -k_{att}\left(y - y_{goal}\right) , & D_o(x,y) > r \end{cases} \tag{8}$$

2.2. Differential Drive Mobile Robot

A differential drives mobile robot (DDMR) is a robot platform that uses different drives on the right and left wheels. The value of linear speed (v) and angular speed (ω) affects producing robot's motion. The robot will move straight when each wheel moves at the same speed. The robot will maneuver when the wheels move at different speeds. Robot maneuvers will follow the lower wheel rotation.

In this study, the DMMR design can be seen in Fig. 1. DDMR consists of 2 wheels connected to the actuator (dynamixel type xl-430-w250-t) and two ball casters as a counterweight. The diameter of the wheels used in DMMR is 0.068 m. The distance between the right and left wheels is 0.163 m. In addition, DDMR uses Open CR 1.0 as a controller. The gyroscope sensor embedded in the controller is used to get the robot's facing angle. The yaw (ψ) component of the gyroscope produces the angle by integrating the angular velocity produced by it.

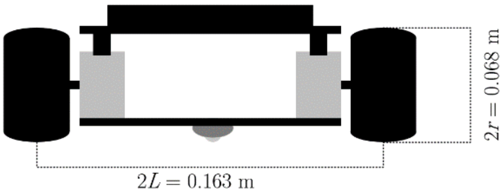


Fig. 1. Design of differential drives mobile robot (DDMR)

The DDMR moves using the kinematic model according to Equation (9). The distance between the right (Δd_r) and left (Δd_l) wheels are obtained by calculating the dynamixel motor encoder and the wheel's circumference. The gyroscope value representing the robot's heading is used to replace the equation $(\Delta d_l - \Delta d_r)/2L$ so that the error value due to wheel slip can be reduced. The value of θ influences the value of $F_x(x,y)$ and $F_y(x,y)$ from Equations (7) and (8).

$$f(x,y,\theta,\Delta d_r,\Delta d_l) = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} + \begin{bmatrix} \dfrac{\Delta d_r + \Delta d_l}{2}\cos\left(\theta + \dfrac{\Delta d_l - \Delta d_r}{4L}\right) \\ \dfrac{\Delta d_r + \Delta d_l}{2}\sin\left(\theta + \dfrac{\Delta d_l - \Delta d_r}{4L}\right) \\ \dfrac{\Delta d_l - \Delta d_r}{2L} \end{bmatrix} \tag{9}$$

The block diagram of the developed system can be seen in Fig. 2. Initial, obstacle, and goal position information is embedded in the robot's memory. The path planning algorithm uses the position coordinates as input to generate $F_x(x,y)$ and $F_y(x,y)$ values. The position coordinates are also sent to the DDMR kinematics equation to get the next robot position value. According to the previous explanation, $F_x(x,y)$ and $F_y(x,y)$ are also sent to the DDMR kinematics model to produce the appropriate robot heading values for avoiding obstacles.

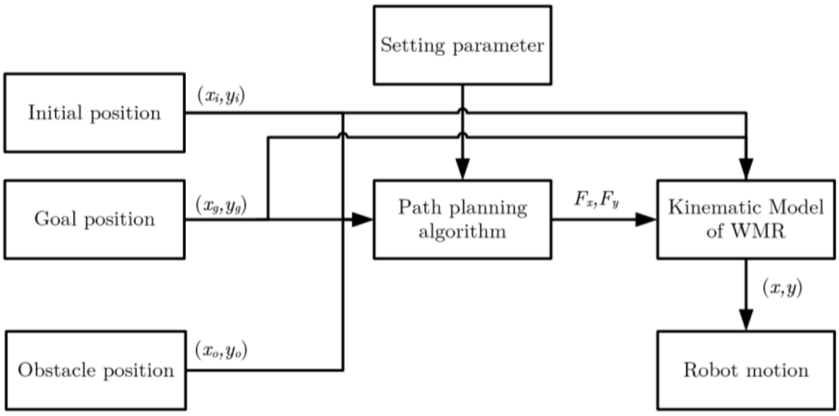


Fig. 2. Block diagram of the system

3. Results and Discussion

3.1. Environmental Setting

According to Fig. 3, the floor used in this test is carpet. This aims to reduce encoder errors due to wheel slip. The initial position and destination position are marked in green (left side). The distance of the robot's trajectory from the initial position to the final position is 1.5 meters. The robot's goal is marked with a green straight line as shown at the right side of Fig. 3.

Fig. 3. Condition of the environment

Obstacles will be between the starting line and the goal on the robot's path. The robot's environment consists of 4 obstacles on the robot's path. The obstacle to the robot has been initialized in the robot program. The obstacle initialization is in the coordinate position of (x, y) robot. A detail of the environment and the obstacle coordinate is presented in Table 1.

Table 1. The environment of the test

Environment	Num. of Obstacle	Obstacle Position	Goal Position
1	1	(0.5, 0)	(1.5, 0)
2	2	(0.4, 0); (1, 0)	(1.5, 0)
3	3	(0.2, 0); (0.6, 0); (1, 0)	(1.5, 0)
4	4	(0.2, 0); (0.5, 0.2); (0.8, 0.1); (1.2, 0)	(1.5, 0)

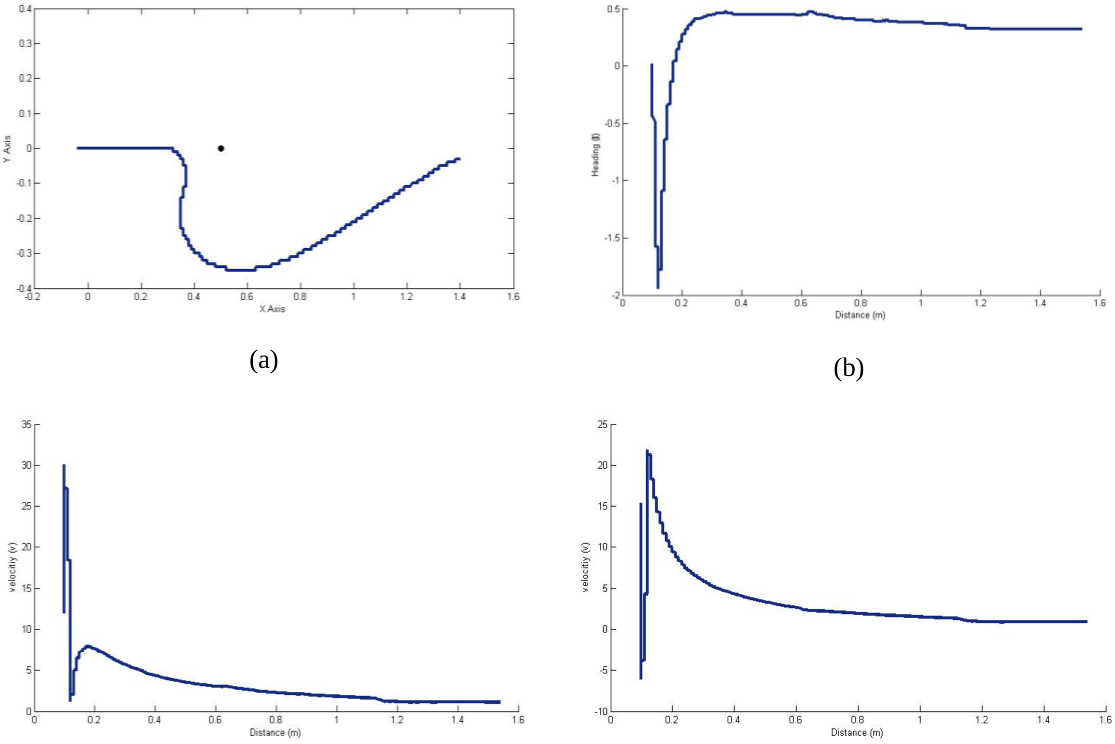
3.2. Trajectory of DDMR

Robot testing is carried out by introducing four environments that have been initialized at the beginning. Robot obstacles are located between the starting line and the goal. The robot's initial position is (0, 0), and the robot's destination line is (1.5, 0). The first test of the robot with an environment that has one obstacle. The position of the robot on the Cartesian plane is (0.5, 0) in meters. Fig. 4 shows the test results in the first environment. Fig. 4(a) shows the trajectory generated by the robot in avoiding obstacles at position (0.5, 0). The safe distance value of the robot to the obstacle (r) is 0.2 m. Therefore, the robot will move away from the obstacle when the distance between the robot and the obstacle is less than or equal to 0.2 m. The tolerance value of the robot's distance to the goal is 0.1 m. Therefore, the robot will stop at a distance of less than 0.1 m.

Fig. 4(b) shows the robot's heading toward the goal position. The initial initialization of the robot is always at 0 rad. The robot moves from an angle of 0 rad, then moves and faces at -1.9 rad. Obstacles on the robot's trajectory amount to 1 obstacle. Significant movement direction changes occur at position (0.31, 0), where the robot avoids obstacles. In addition, the robot starts moving towards the goal at position (0.57, -0.35). The robot stops at an angle of 0.32 rad.

Fig. 4(c) is a graph of the speed at the left wheel. The initial speed of the left wheel is 15 rpm. After that, the left wheel decreased because the robot made a turn maneuver. The highest left wheel speed of the robot when maneuvering is 30 rpm. Fig. 4(d) is a graph of the speed of the right wheel.

The initial speed of the right wheel is 15 rpm. After that, the right wheel decreased because the robot made a turn maneuver. The highest speed of the robot's right wheel when maneuvering is 22 rpm. The direction of motion influences significant speed changes in the robot. When the robot rotates to the left, the speed of the left wheel will decrease. Conversely, the left wheel speed will increase when the robot turns to the right.



(a) (b)

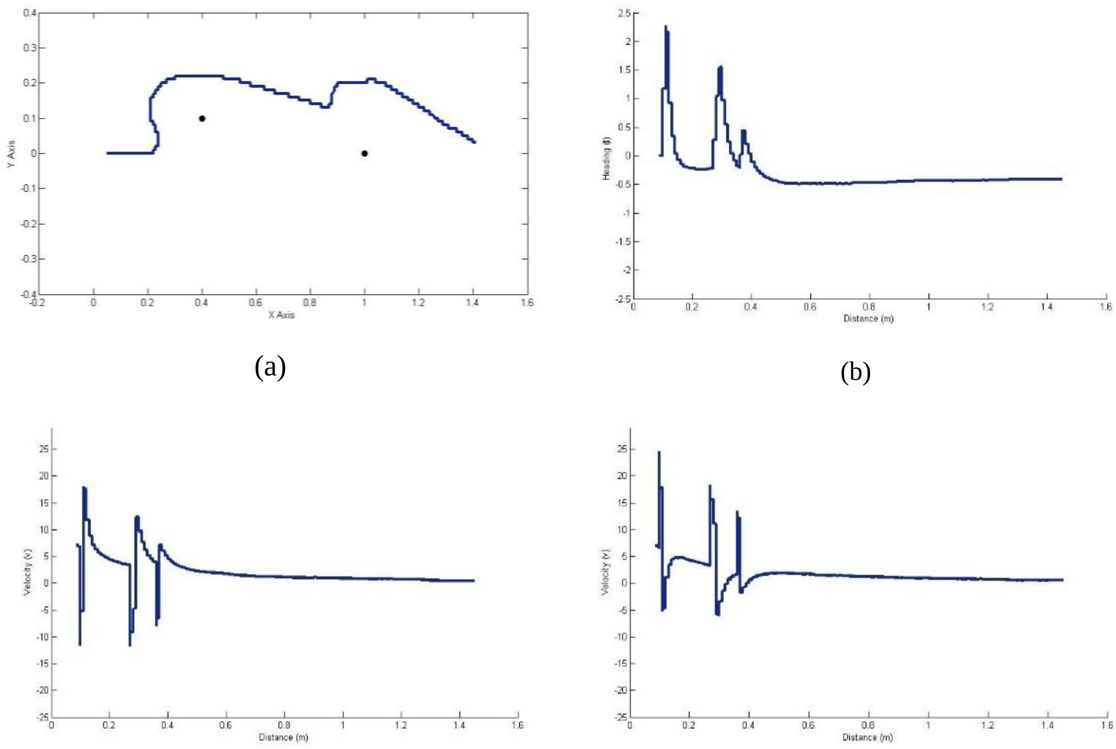
(c) (d)

Fig. 4. Result of the first test using 1 obstacle: (a) Trajectory in environment 1; (b) Heading robot in environment 1; (c) Speed of left wheel in environment 1; (d) Speed of right wheel in environment 2

The second experiment was carried out in an environment that had two obstacles. The position of the obstacles in the environment is (0.4, 0.1) and (1, 0) in meters. Fig. 5(a) is a trajectory robot from DMR kinematic data acquisition. The robot's initial position is (0, 0), with the goal position (1.5, 0). The safe value between the robot and each obstacle is 0.2 m. According to Fig. 5(a), the maneuver occurs when the robot is in position (0.2, 0) to get away from obstacles in position (0.4, 0). After passing the first obstacle, the robot will approach the goal position (1.5, 0). The robot turns again at position (0.87, 0.14) to avoid obstacles (1, 0). The robot reaches the goal and stops at coordinates (1.41, 0.03).

Fig. 5(b) shows the resulting robot heading angle in environment 2. The initial initialization of the robot in the second test is 0 rad. The robot moves from an angle of 0 rad and then maneuvers to the left when it is in position (0.2, 0). This is caused by the first obstruction at position (0.4, 0). At position (0.87, 0.14), the robot changes its direction of motion because of an obstacle at position (1, 0). After passing all the existing obstacles, the robot is able to reach the goal position. The robot stops at -0.41 rad.

Fig. 5(c) is a graph of the speed at the left wheel. The initial speed of the left wheel is 7.3 rpm, then it decreases because the robot performs a turn maneuver. The highest left wheel speed of the robot when maneuvering is 30 rpm. Fig. 5(d) is a graph of the speed at the right wheel. The initial speed of the right wheel is 7.2 rpm, then it decreases because the robot performs a turn maneuver. The highest speed of the robot's right wheel when maneuvering is 22 rpm.



(a) (b)

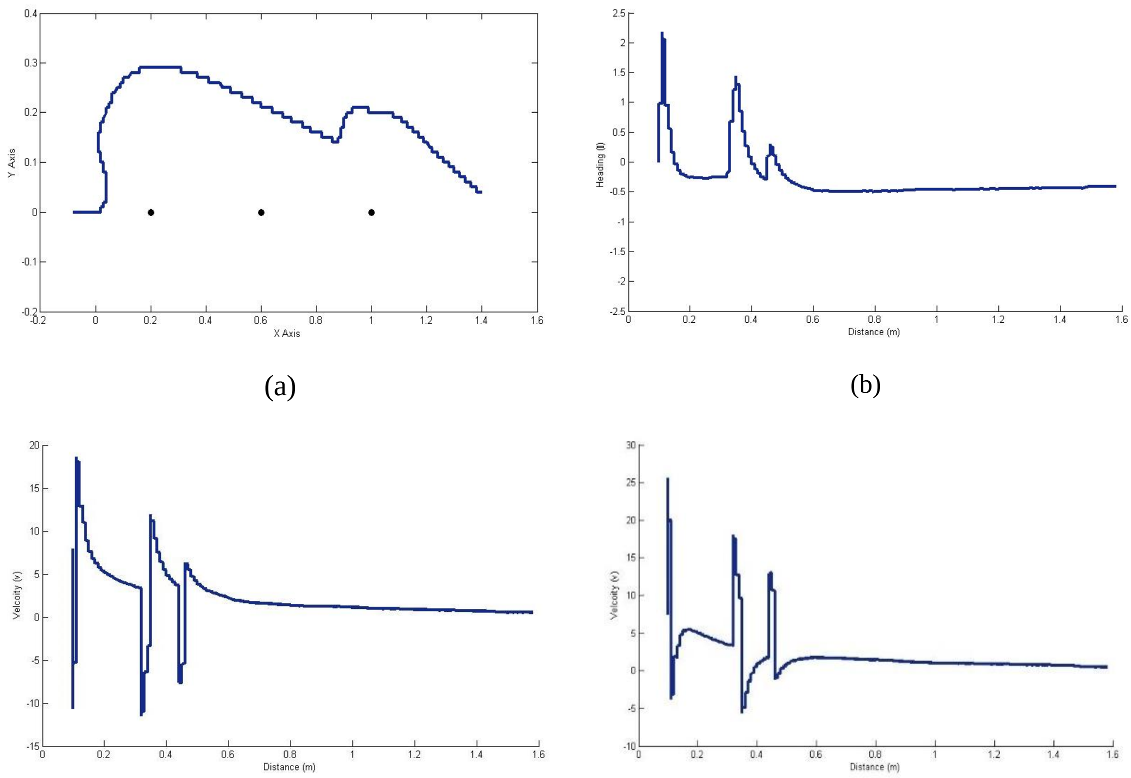
(c) (d)

Fig. 5. Result of the second test using 2 obstacles: (a) Trajectory in environment 2; (b) Heading robot in environment 2; (c) Speed of left wheel in environment 2; (d) Speed of right wheel in environment 2

Fig. 6(a) shows the robot's trajectory from environment 3. The initial positions of the robot and goal positions are (0, 0) and (1.5, 0), respectively. Obstacles are in positions (0.2, 0), (0.6, 0), and (1, 0), with a safe distance to each obstacle of 0.2 m. According to Fig. 6(a), the robot changes direction at position (0.01, 0) due to an obstacle at position (0.2, 0). Then the robot returns to its destination route to the goal. The trajectory robot still goes to the goal position even though there is an obstacle at (0.6, 0). The obstacle distance causes this to be greater than that which has been determined. At position (0.87, 0.14), the robot experiences a left turn maneuver because there is a third obstacle, namely at position (1, 0). Therefore, the robot moves away from the obstacle and stops at position (1.4, 0.04). The last position of the robot shows that the resulting error to goal value is less than the tolerance value of 0.1 m.

Fig. 6(b) shows the robot's heading angle in environment 3. The initial initialization of the robot's heading angle is 0 rad. The robot moves from an angle of 0 rad and then changes position (0.01, 0). This is caused by the first obstacle at position (0.2, 0). After successfully passing the first obstacle, the robot maneuvers towards the goal at position (0.02, 0.11). When passing the second obstacle, the robot does not make a large enough angle change because the robot's distance is greater than the distance at the second obstacle. At position (0.87, 0.14), the robot again experiences a change in heading angle because there is an obstacle at position (1, 0). The robot goes to its destination after passing through all the obstacles. The robot stops at an angle of -0.41 rad.

Fig. 6(c) and Fig. 6(d) are graphs of the speeds at the left and right wheels. The initial speed of the left wheel is 7.89 rpm. It has decreased because the robot is maneuvering the turns. The initial speed of the right wheel is 7.86 rpm. The highest left wheel speed of the robot when maneuvering is 18.57 rpm. In addition, the highest speed of the robot's right wheel when maneuvering is 25.61 rpm.



(a) (b)

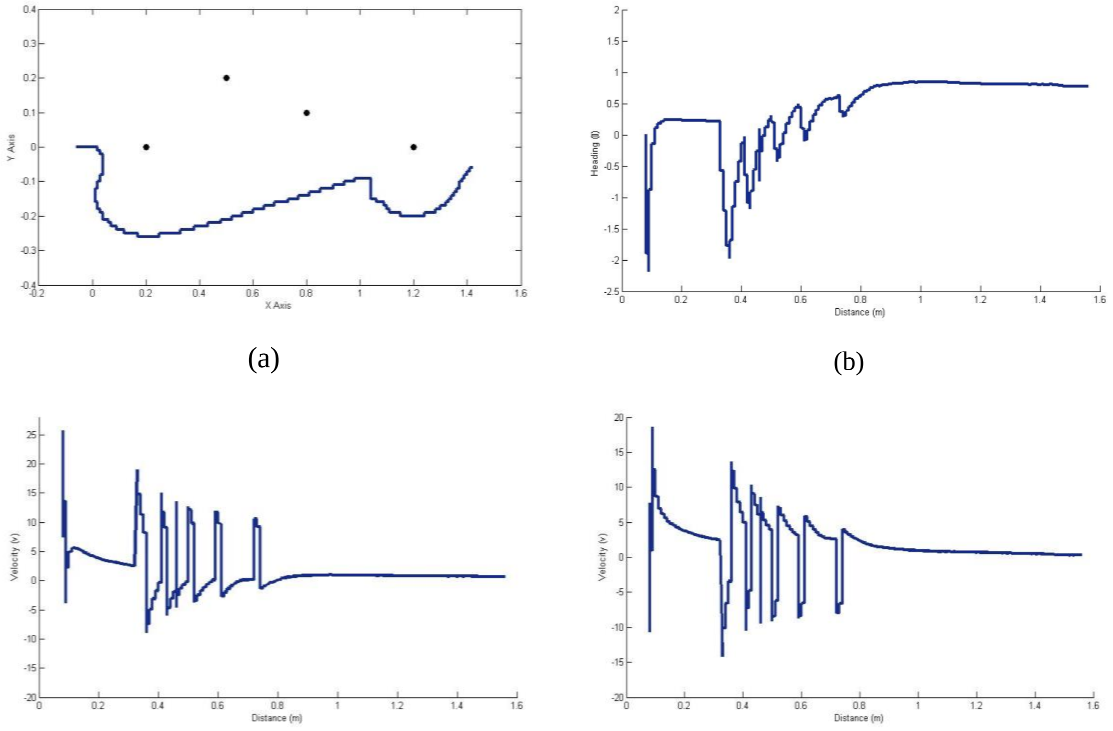
(c) (d)

Fig. 6. Result of the third test using 3 obstacles: (a) Trajectory in environment 3; (b) Heading robot in environment 3; (c) Speed of left wheel in environment 3; (d) Speed of right wheel in environment 3

Fig. 7(a) shows the trajectory of the robot in environment 4. The robot obstacles are in positions (0.2, 0), (0.5, 0.2), (0.8, 0.1), and (1.2, 0). The safe distance of the robot against each obstacle is 0.2 m. According to Fig. 7(a), the robot changes direction at position (0.01, 0) due to an obstacle at position (0.2, 0). Furthermore, the robot returns to its destination route, namely (1.5, 0). The robot remains towards the goal and is not affected by the second obstacle because the distance between the robot and the obstacle is greater than the value of r. The robot moves to avoid obstacle four at position (1.04, -0.1). The robot maneuvers to the right because there is a fourth obstacle which is at position (1.2, 0), so the robot stays away from the obstacle. The robot stops at position (1.42, -0.06) with an error tolerance of 0.1 m to the goal.

Fig. 7(b) shows the robot's heading angle in environment 4. Initialize the robot's heading angle in environment four at 0 rad. The robot's heading angle starts to change when it is in position (0.01, 0). This is because there is a single obstacle at position (0.2, 0). The robot is not affected by the second and third obstacles because the robot's trajectory toward the goal is at a value greater than r. The robot experiences a change in the heading angle value at the position (1.02, -0.09) because there is an obstacle at position (1.2, 0). After being able to pass all the obstacles that exist, the robot goes to the target. The robot stops at a heading angle of 0.77 rad.

Fig. 7(c) and Fig. 7(d) show graphs of the speeds at the left and right wheels in environment four. The initial speed of the left and right wheels is 7.7 rpm and 7.7 rpm, respectively. The value of the velocity decreases according to the DDMR kinematics equation. The highest left wheel speed of the robot when maneuvering is 25.65 rpm. The highest speed of the robot's right wheel when maneuvering is 18.61 rpm.



(a) (b)

(c) (d)

Fig. 7. Result of the fourth test using 4 obstacles: (a) Trajectory in environment 4;(b) Heading robot in environment 4; (c) Speed of left wheel in environment 4; (d) Speed of right wheel in environment 4

4. Conclusion

According to the presenter results, the Artificial Potential Field (APF) algorithm can be potentially applied to the Differential Drives Mobile Robot (DDMR) kinematic model robot. DDMR managed to avoid obstacles in point masses and reach the desired goal. DDMR can pass all prepared test environments. DDRR is able to reach a predetermined goal position with an error less than the tolerance of 0.1 m.

Author Contribution: All authors contributed equally to the main contributor to this paper. All authors read and approved the final paper.

Acknowledgement: The third author acknowledge the Polish National Agency for Academic Exchange (NAWA) No. BPN/ULM/2022/1/00139/U/00001 for partly financial support of the study.

Funding: This research was funded by Universitas Ahmad Dahlan via Biro Kemahasiswaan dan Alumni (BIMAWA).

Conflicts of Interest: The authors declare no conflict of interest.

References

[1] R. Siegwart and I. R. Nourbakhsh, *Introduction to Autonomous Mobile Robots*. MIT Press, 2004, https://books.google.co.id/books?id=4of6AQAAQBAJ.

[2] S. Nurmaini and B. Tutuko, “Intelligent robotics navigation system: Problems, methods, and algorithm,” *Int. J. Electr. Comput. Eng.*, vol. 7, no. 6, pp. 3711–3726, 2017, https://doi.org/10.11591/ijece.v7i6.pp3711-3726.

[3] H. Y. Zhang, W. M. Lin, and A. X. Chen, "Path planning for the mobile robot: A review," *Symmetry (Basel).*, vol. 10, no. 10, 2018, https://doi.org/10.3390/sym10100450.

[4] A. N. Atiyah, N. Adzhar, and N. I. Jaini, "An overview: On path planning optimization criteria and mobile robot navigation," *J. Phys. Conf. Ser.*, vol. 1988, no. 1, 2021, https://doi.org/10.1088/1742-6596/1988/1/012036.

[5] A. Koubaa *et al.*, *Robot Path Planning and Cooperation: Foundations, Algorithms and Experimentations*. Springer Cham, 2018, https://doi.org/10.1007/978-3-319-77042-0.

[6] G. Li, A. Yamashita, H. Asama, and Y. Tamura, "An efficient improved artificial potential field based regression search method for robot path planning," *2012 IEEE Int. Conf. Mechatronics Autom. ICMA 2012*, pp. 1227–1232, 2012, https://doi.org/10.1109/ICMA.2012.6283526.

[7] P. Wang, S. Gao, L. Li, B. Sun, and S. Cheng, "Obstacle avoidance path planning design for autonomous driving vehicles based on an improved artificial potential field algorithm," *Energies*, vol. 12, no. 12, 2019, https://doi.org/10.3390/en12122342.

[8] S. Wang, T. Zhao, and W. Li, "Mobile Robot Path Planning Based on Improved Artificial Potential Field Method," *2018 IEEE Int. Conf. Intell. Robot. Control Eng. IRCE 2018*, pp. 257–262, 2018, https://doi.org/10.1109/IRCE.2018.8492951.

[9] P. Yan, Z. Yan, H. Zheng, and J. Guo, "Real Time Robot Path Planning Method Based on Improved Artificial Potential Field Method," *Proc. 37th Chinese Control Conf.*, vol. 37, no. July 25-27, 2018, Wuhan, China, pp. 4814–4820, 2018, https://doi.org/10.23919/ChiCC.2018.8482571.

[10] S. M. H. Rostami, A. K. Sangaiah, J. Wang, and X. Liu, "Obstacle avoidance of mobile robots using modified artificial potential field algorithm," *Eurasip J. Wirel. Commun. Netw.*, vol. 2019, no. 1, 2019, https://doi.org/10.1186/s13638-019-1396-2.

[11] Z. Liu and T. Jiang, "Route planning based on improved artificial potential field method," *2017 2nd Asia-Pacific Conf. Intell. Robot Syst. ACIRS 2017*, pp. 196–199, 2017, https://doi.org/10.1109/ACIRS.2017.7986092.

[12] X. Gu, M. Han, W. Zhang, G. Xue, G. Zhang, and Y. Han, "Intelligent vehicle path planning based on improved artificial potential field Algorithm," *2019 Int. Conf. High Perform. Big Data Intell. Syst. HPBD IS 2019*, no. 2016, pp. 104–109, 2019, https://doi.org/10.1109/HPBDIS.2019.8735451.

[13] W. Li, C. Yang, Y. Jiang, X. Liu, and C. Y. Su, "Motion planning for omnidirectional wheeled mobile robot by potential field method," *J. Adv. Transp.*, vol. 2017, 2017, https://doi.org/10.1155/2017/4961383.

[14] Iswanto, A. Ma'arif, O. Wahyunggoro, and A. I. Cahyadi, "Artificial potential field algorithm implementation for quadrotor path planning," *Int. J. Adv. Comput. Sci. Appl.*, vol. 10, no. 8, pp. 575–585, 2019, https://doi.org/10.14569/IJACSA.2019.0100876.

[15] Z. Yujiang and L. Huilin, "Research on mobile robot path planning based on improved artificial potential field," *Math. Model. Eng.*, vol. 3, no. 2, pp. 135–144, 2017, https://doi.org/10.21595/mme.2017.19520.

[16] Y. Singh, S. Sharma, R. Sutton and D. Hatton "Path Planning of an Autonomous Surface Vehicle based on Artificial Potential Fields in a Real Time Marine Environment," *16th International Conference on Computer and IT Applications in the Maritime Industries (COMPIT 2017)*, pp. 1–10, 2017, http://hdl.handle.net/10026.1/8717.

[17] A. Lazarowska, "A Discrete Artificial Potential Field for Ship Trajectory Planning," *J. Navig.*, vol. 73, no. 1, pp. 233–251, 2020, https://doi.org/10.1017/S0373463319000468.

[18] P. Sudhakara, V. Ganapathy, B. Priyadharshini, and K. Sundaran, "Obstacle Avoidance and Navigation Planning of a Wheeled Mobile Robot using Amended Artificial Potential Field Method," *Procedia Comput. Sci.*, vol. 133, pp. 998–1004, 2018, https://doi.org/10.1016/j.procs.2018.07.076.

[19] W. Siming, Z. Tiantian, and L. Weijie, "Mobile Robot Path Planning Based on Improved Artificial Potential Field Method," *2018 IEEE Int. Conf. Intell. Robot. Control Eng.*, vol. 2, no. 1, pp. 29–33, 2018, https://doi.org/10.1109/IRCE.2018.8492951.

[20] K. Gao, D. Yan, F. Yang, J. Xie, L. Liu, R. Du, and N. Xiong, "Conditional artificial potential field-based autonomous vehicle safety control with interference of lane changing in mixed traffic scenario," *Sensors (Switzerland)*, vol. 19, no. 19, 2019, https://doi.org/10.3390/s19194199.