Journal of Robotics and Control (JRC)
Volume 2, Issue 5, September 2021
ISSN: 2715-5072 DOI: 10.18196/jrc.25121

Implementation Kinematics Modeling and Odometry of Four Omni Wheel Mobile Robot on The Trajectory Planning and Motion Control Based Microcontroller

Dhiya Uddin Rijalusalam [1], Iswanto Iswanto [2]*
[1, 2] Department of Electrical Engineering, Universitas Muhammadiyah Yogyakarta, Indonesia
Email: [2] iswanto_te@umy.ac.id
*Corresponding Author

***Abstract*— The control of kinematic modeling in a four wheel omni-directional robot (FWOR) is very difficult. Because you have to adjust the speed of the four DC motors. The speed of DC motors is controlled so that the FWOR robot can be controlled. This paper will explain the application of kinematic modeling of four wheel omni directional robots as track tracking controllers and microcontroller based movement control. Kinematic is the study of robot motion based on geometric structure analysis of a stationary / moving reference coordinate frame system without considering the force, torque or certain moments that cause movement. By applying kinematic modeling and calculation of the odometric system as feedback, the control of the robot trajectory movement can be controlled with precision in accordance with the path planning that has been made. The robot track control technique is embedded in a 32-bit ARM microcontroller. The path planning system and observing robot movement are carried out using a friendly graphic interface using Processing to facilitate the robot monitoring process. The results of the experiments and tests carried out, the system is able to control the rate of movement of the robot with great precision in accordance with the path planning made.**

Keywords*— *Omni Wheel Mobile Robot, Kinematics Modeling, Odometry, Trajectory Motion Control.

I. INTRODUCTION

The rapid development of science and technology has had a significant impact on human life, one of which is in the field of mobile robots. Nowadays, mobile robot technology has been widely used in various applications in human life, such as transporting objects from one place to another. This is due to their efficiency in handling a job continuously independently. One example is shown by the Amazon company in implementing wheeled mobile robots. Holonomic wheeled robot that is often applied to transport goods [1] [2] [3]. The robot uses an omnidirectional wheel that can move in all directions [4] [5] [6] [7] [8]. In the last few decades, the increasing number of mobile robot applications has attracted the attention of many researchers. The main problems in controlling the movement of mobile robots such as point stabilization, path planning, trajectory tracking, real-time avoidance, and motion control have attracted a lot of attention from both academics and practitioners.

In general, wheeled robot navigation systems that are often encountered consist of two types, namely ackerman steering and differential steering which have low mobility. This is due to the limitations of the type of robot being unable to move in all directions, so that these two types can be categorized as non-holomonic robots [6]. To overcome this problem, a holomonic robot that has a high level of mobility was developed, that is, it can move in all directions in the x-y cartesian plane without having to rotate the direction of the robot's orientation. One type of holomonic robot is omnidirectional robot [5] [6] [7].

To be able to control the movement of the holomonic omnidirectional robot in all directions, a structural geometry analysis of the coordinate reference frame is required [8]. This is very closely related to the structure of the robot and the placement of the wheels is often called kinematic modeling. By understanding the kinematic modeling of the omniwheel robot, the motor direction and speed can be calculated easily. In addition, to determine the actual and precise robot movement system, odometric data is needed, which is data that can predict changes in the robot's position with respect to the robot's initial location.

Several previous researchers have conducted research on robot track and motion tracking control. Modeling and Motion Control of Soft Robots was studied by Fei and Xu [9]. Avoidance of Dynamic Inhibition from Restricted Cellular Manipulation Using a Predictive Control Model was investigated by Li and Xiong [3]. Fast Online Frequency Adaptation Mechanism for CPG-based Robot Movement Control was investigated by Thor and Manoonpong [10]. Powerful Control of the Trajectory of Time to New with High Order Shift Mode for Human-Robot Cooperation was investigated by Ren, Wang and Chen [11]. Feedforward Motion Control with Variable Stiffness Actuators Inspired by Muscle Cross-Bridge Kinematics was studied by Chang, Kim and Kim [12]. Control of SCARA 2-DOF Robot Hybrid Tracking via Port-Controlled Hamiltonian and Backstepping was investigated by Chi, Yu and Yu [13]. Decentralized Track Tracking Control for Soft Robots Interacting with the Environment was investigated by Angelini [14]. Movement control of maintenance robots based on Kinect motion

recognition was studied by Ge, Wang and Xing [15]. Control of the motion of robotic fish in dynamic environmental conditions using an adaptive control approach was investigated by Verma, Shen and Xu [16].

Utilizing Distance Information from Different Interaural Levels for Motion Control of Binaural Robots was investigated by Magassouba, Bertin and Chaumette [17]. Control of Vehicle Track Tracking with Less Planar Activation was studied by Ashrafiuon, Nersesov and Clayton [18]. Movement Control Tracking Safety Based on Prohibited Virtual Equipment in Robot-Assisted Nose Surgery was investigated by Zheng [19]. Interference rejection MPC for tracking wheeled mobile robots was investigated by Sun [20]. Mode-switching motion control systems for reactive and following surface interactions using industrial robots were investigated by Nakhaeinia, Payeur and Laganiere [21]. Robotic Motion Control of Wire-Based Continuous Contact Catheter was investigated by Zhang [22]. Long Stroke Hydraulic Robot Motion Control with Increased Nonlinear Dynamic Inversion was investigated by Huang [23]. Movement Control Strategy for the Dolphin Robot Repeating Leap was investigated by Yu [24]. Maintain Rollin - Whole Body Movement Control and Planning for a Wheeled Four-Legged Robot researched by Bjelonic [25]. Motion control of a floating biomimetic four-fin underwater robot was investigated by Salumae, Chemori and Kruusmaa [26]. Motion Control of UX-1 Underwater Mining Exploration Robot: Field Trials researched by Fernandez [27].

This paper is different from previous research papers. This paper describes the implementation of and kinematic modeling and odometry in path planning and controlling the movement of an omnidirectional robot based on a microcontroller. In section II, we will explain the mathematical calculations of the omnidirectional robot kinematic model, and the odometric system in determining the change in the position of the robot with respect to its initial location. In section III, we will explain the architectural control system consisting of track planning, motion control, and the overall system architecture. In section IV, we will explain some of the results of testing the robot movement. Section V contains the conclusions, notes, and research plans that will be carried out next.

II. METHODS

A. Kinematic Modeling on Omni Wheel Platform

Kinematic modeling is the study of robot motion based on geometric structure analysis of a stationary / moving reference coordinate frame system without considering the force, torque or certain moments that cause movement [28]. The position of the omni wheel greatly affects the kinematic model of the omni directional robot. This study uses 4 omni wheels, where the axes of the four wheels are joined at the center point of the robot and the angle between the wheels forms an angle of 90°.

In the kinematic modeling of robots, there are two kinematic models, namely inverse kinematic (IK) and forward kinematic (FK) shown in figure 1. IK is used to determine the linear velocity of the four omni wheels and uses forward kinematics to determine the linear velocity of the robot against global coordinates.

Figure 1 shows the display of the wheel placement frame from FWOR. The angle formed between the wheels and the robot reference point is denoted using $\boldsymbol{\alpha_1, \alpha_2, \alpha_3, \alpha_4}$. The angular velocity of each wheel is denoted using $\boldsymbol{\omega_w} = [\boldsymbol{\omega_1}\ \boldsymbol{\omega_2}\ \boldsymbol{\omega_3}\ \boldsymbol{\omega_4}]^{\boldsymbol{T}}$, and the linear velocity of each wheel is denoted using $\boldsymbol{v_w} = [\boldsymbol{v_1}\ \boldsymbol{v_2}\ \boldsymbol{v_3}\ \boldsymbol{v_4}]^{\boldsymbol{T}}$. This wheel speed value will be positive when the wheel rotates clockwise (CW), and will be negative when counter clockwise (CCW). The linear velocity and angular velocity of the robot with respect to the XOY global coordinates are notified using $\boldsymbol{V_{(g)}} = [\dot{\boldsymbol{X}}\ \ \dot{\boldsymbol{Y}}\ \boldsymbol{\omega}\ ]$. R is the distance from the wheel to the center of the robot and r is the radius of the omni wheel.

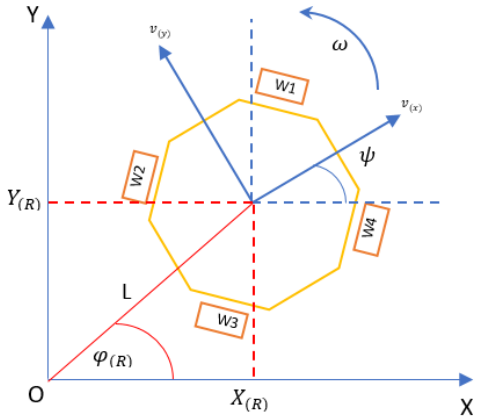

Fig. 1. Wheel placement of FWOR

The display of coordinates and orientation of FWOR to global XOY coordinates is shown in Figure 2.From this figure, it can be seen that the linear velocity of the robot relative to the robot's body frame is denoted using $\boldsymbol{v_{(R)}} = [\boldsymbol{v_{(x)}}\ \ \boldsymbol{v_{(y)}}\ ]^{\boldsymbol{T}}$ The position of the robot's coordinates and orientation to global coordinates are denoted using $\boldsymbol{X_{(g)}} = \left[\boldsymbol{x_{(R)}}\ \boldsymbol{y_{(R)}}\ \boldsymbol{\psi}\right]^{\boldsymbol{T}}$. $\boldsymbol{\omega}$ is the notation of the angular velocity of the robot to the global reference coordinates. $\boldsymbol{\varphi_{(R)}}$ is the direction of movement of the robot at global coordinates, L is the notation of the resultant linear velocity of the robot.

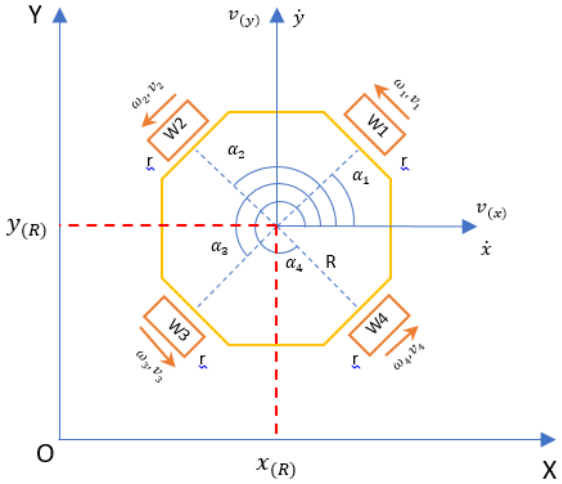

Fig. 2. Wheel placement of FWOR

Based on Figure 2, we can formulate the following equations.

$$\varphi_{(R)} = tan^{-1} \frac{Y_{(R)}}{X_{(R)}} \quad (1)$$

$$L = \sqrt{X_{(R)}^{\ 2} + Y_R^{\ 2}} \quad (2)$$

$$\begin{matrix} v_{(x)} \\ v_{(y)} \\ \omega \end{matrix} = \begin{bmatrix} \cos(\psi) & \sin(\psi) & 0 \\ -\sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} . \begin{bmatrix} \dot{X} \\ \dot{Y} \\ \omega \end{bmatrix} \quad (3)$$

Based on Figure 1 and Figure 2, we can formulate the inverse kinematic equation and kinematic forward. In the inverse kinematic the input is the linear velocity of the robot $\boldsymbol{V_{(g)}}$ to determine the linear velocity of each wheel. The solution of the inverse kinematic equation can be seen in equation (4).

$$v_w = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix} = T(\theta) . V_{(g)} \quad (4)$$

$$V_{(g)} = [\dot{X} \quad \dot{Y} \quad \omega]^T \quad (5)$$

$$T(\theta) = \begin{bmatrix} -\sin(\psi + \alpha_1) & \cos(\psi + \alpha_1) & R \\ -\sin(\psi + \alpha_2) & \cos(\psi + \alpha_2) & R \\ -\sin(\psi + \alpha_3) & \cos(\psi + \alpha_3) & R \\ -\sin(\psi + \alpha_4) & \cos(\psi + \alpha_4) & R \end{bmatrix} \quad (6)$$

In the forward kinematic input is the linear velocity of the wheel to get the linear velocity of the robot $\boldsymbol{V_{(g)}}$. By referring to the inverse kinematic equation (4) we can get the value of $\boldsymbol{V_{(g)}}$ by multiplying $\boldsymbol{V_{(g)} = V_w . T(\theta)^{-1}}$, Because $\boldsymbol{T(\theta)}$ is a 4x3 matrix, so $\boldsymbol{T(\theta)^{-1}}$ is a pseudo-inverse. If the formula is used, a calculation error will occur, so we try to propose a new equation which can be seen in the following equation Eq (7).

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \\ \omega \end{bmatrix} = D . \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix} \quad (7)$$

$$D = \frac{1}{2} \begin{bmatrix} -\sin(\psi + \alpha_1) & \cos(\psi + \alpha_1) & 2R^{-1} \\ -\sin(\psi + \alpha_2) & \cos(\psi + \alpha_2) & 2R^{-1} \\ -\sin(\psi + \alpha_3) & \cos(\psi + \alpha_3) & 2R^{-1} \\ -\sin(\psi + \alpha_4) & \cos(\psi + \alpha_4) & 2R^{-1} \end{bmatrix}^T \quad (8)$$

B. Odometry System on Omni Wheel Platform

Odometry is a method used to obtain data from various motion sensors to estimate changes in position over certain time intervals. In this study, the odometry sensor used a rotary encoder on each driving wheel. To obtain data on changes in position with time, it is necessary to read the linear velocity of the wheel $\boldsymbol{v_{w(n)}}$ as in equation number (9).

$$v_{w(n)} = \frac{\dot{e}_{(n)}}{PPR} \text{ x } K \quad (9)$$

$$\dot{e}_{(n)} = \frac{d_{enc_{(n)}}}{d_t} \quad (10)$$

where $K = 314.28\ mm$ and n is the notation of the wheel number. After getting the linear velocity of each wheel, this value is then entered into the FK equation in equation (7) to get the linear velocity value for the robot $\boldsymbol{V_{(g)}}$. To get the global coordinates and orientation of the $\boldsymbol{X_{(g)}}$ robot, equations (11), (12), and (13) can be used.

$$X_{(R)} = \sum_{k=1}^{n} (\dot{X}_{(n)}) \quad (11)$$

$$Y_{(R)} = \sum_{k=1}^{n} (\dot{Y}_{(n)}) \quad (12)$$

$$\Psi = \sum_{k=1}^{n} (\omega_{(n)}) \quad (13)$$

III. CONTROL SYSTEM ARCHITECTURES

A. Overall System Architectures

In this explanation, we will explain about robot architecture systems and robot movement control systems using kinematic modeling and odometry based on microcontrollers. Figure 3 is a detailed block diagram of the system being made.

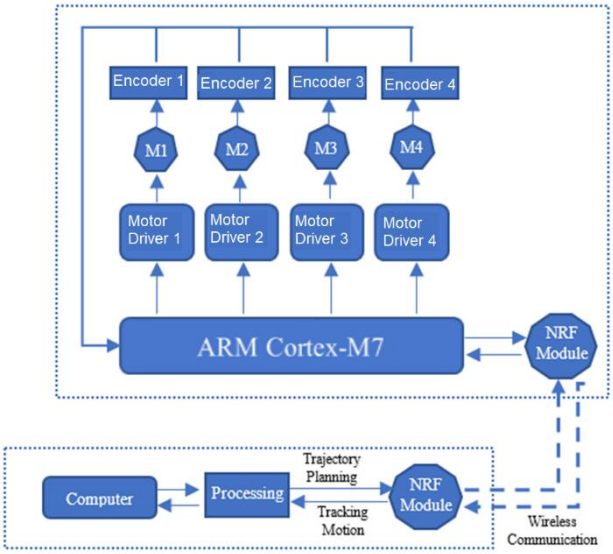

Fig. 3. Detailed Block Diagram System

The FWOR system is controlled using an ARM Cortex M7 microcontroller. Motion control is based on information obtained through the rotary encoder of the DC motor. The rotary encoder used has a number of pulses per revolution (PPR) of 320. The linear and angular speed is estimated by counting the number of pulses obtained from the encoder per one unit of time as in equations (10) and (11). The microcontroller generates 6 Pulse Width Modulation (PWM) signals which are expressed in values of 0-255 (8 bits) and are applied to 3 H-bridge motor drivers. Each motor driver

consists of 2 inputs PWM1 and PWM2. When the motor rotates CW, PWM1 will be equal to the pulse value PWM, and PWM2 is equal to zero, and vice versa when the direction of CCW. To plan the trajectory and the process of monitoring the movement of the robot, a graphical user interface (GUI) is made using a processing application, the process of sending and retrieving robot data using the NRF24l01 wirelessly. Figure 4 is a FWOR physical construction image.

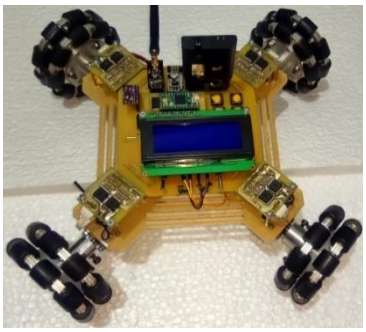

Fig. 4. FWOR Physical Construction

Figure 5 is a graphical user interface of the application being made.

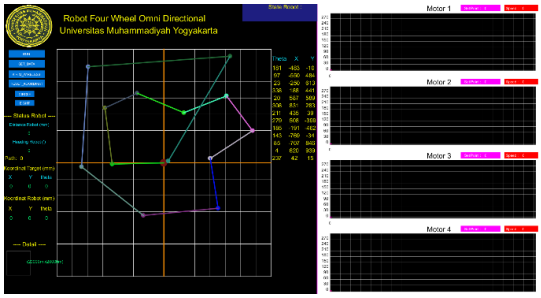

Fig. 5. Graphical User Interface (GUI)

B. *Trajectory Planning*

The basis for determining trajectory planning from FWOR can be seen in Figure 6. This figure illustrates the mapping of robot positions and trajectories that have been made. $\boldsymbol{X_{(R)}}, \boldsymbol{Y_{(R)}}$ shows FWOR coordinates to global coordinates, $\boldsymbol{X_{p(n)}}, \boldsymbol{Y_{p(n)}}$ are the coordinates of the trajectory planning symbol n shows the sequence notation of the path point, $\boldsymbol{\varphi_{(n)}}$ is a notation of the direction of movement of the robot to global coordinates.

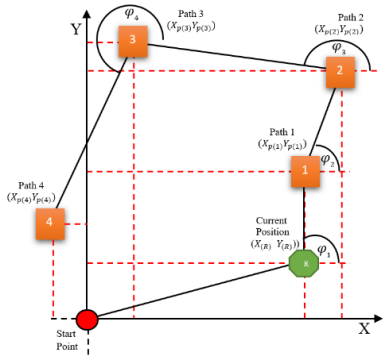

Fig. 6. Trajectory Planning Algorithm

By referring to this figure, several equations can be formulated as follows.

$$\boldsymbol{X_{t(n)} = X_{p(n)} - X_{p(n-1)}} \quad (14)$$

$$\boldsymbol{Y_{t(n)} = Y_{p(n)} - Y_{p(n-1)}} \quad (15)$$

$$\Psi = \sum_{k=1}^{n} (\omega_{(n)}) \quad (16)$$

To calculate the linear velocity value for each $\boldsymbol{v_w}$ wheel, you can use the IK equation as in equation (3). However, to be able to control the PWM value into a precise linear velocity value, controls such as PID, fuzzy and etc. So that we cannot use equation (3) in this system, but we can still run FWOR in the direction according to the point of destination coordinates $\boldsymbol{X_{p(n)}}, \boldsymbol{Y_{p(n)}}$ using equation (17) as follows.

$$\begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix} = \begin{bmatrix} -\sin(\psi + \alpha_1) & \cos(\psi + \alpha_1) & R \\ -\sin(\psi + \alpha_2) & \cos(\psi + \alpha_2) & R \\ -\sin(\psi + \alpha_3) & \cos(\psi + \alpha_3) & R \\ -\sin(\psi + \alpha_4) & \cos(\psi + \alpha_4) & R \end{bmatrix} \cdot \begin{bmatrix} \cos(\varphi_{(n)} \\ \sin(\varphi_{(n)} \\ \omega \end{bmatrix} . \mathrm{MP} \quad (17)$$

MP is a notation of the maximum PWM variable value, this value has a value between 0-255, used as the maximum PWM value limit. The process of changing the direction of the robot's movement can be estimated using equation (18)

$$Res \leq R_{Circle};\ X_{p(n)} = X_{p(n+1)};\ Y_{p(n)} = Y_{p(n+1)} \quad (18)$$

where $\boldsymbol{R_{Circle}}$ is the Radius Circle of Target. Changes in the direction of movement of the robot that occur suddenly when the motor rotates rapidly when the trajectory point is touched, can cause the system to become unstable, and cause wheel slip, so a smooth change in speed direction is needed. To overcome this problem, it can be overcome by reducing the value of $\boldsymbol{pwm_{ref}}$ by

$$pwm_{ref} = pwm_{ref} . 0.3 \quad (19)$$

With this equation, it causes a decrease in the value of PWM according to equation (19) and anticipates a sudden change. The following is a block diagram image of the control of the FWOR system.

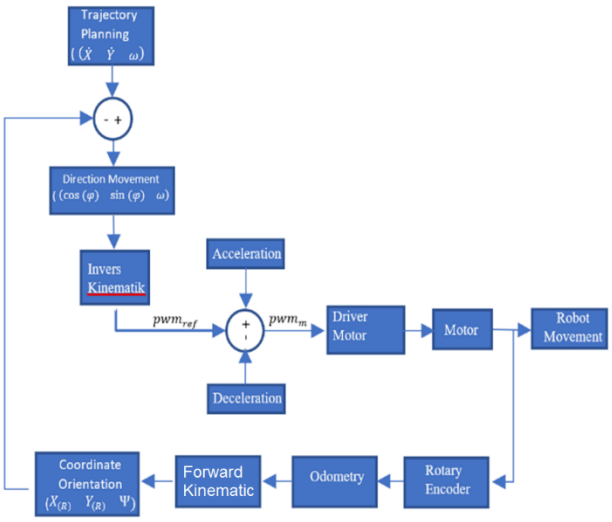

Fig. 7. Block Diagram of Control System

IV. VALIDATION, EXPERIMENT, DISCUSSION

A. Kinematic Experiment and Validation Forward

In this first test, it was carried out to test how precise the robot coordinates readings were by using the forward kinematic equation using a rotary encoder sensor on the four drive wheels. The method used is by running the robot to a certain coordinate point, then comparing the reading of the coordinates with existing measuring instruments. Table I below is the results of the tests that have been carried out.

TABLE I. KINEMATIC FORWARD TESTING DATA

Data Forward Kinematic			Real Position		
X (mm)	Y (mm)	Heading (°)	X (mm)	Y (mm)	Heading (°)
20	20	0	19	21	0
40	40	10	41	40	11
80	80	45	81	81	46
160	160	90	162	159	90
300	300	180	302	302	181
500	500	270	501	502	271
1000	1000	310	1002	1002	309
2000	2000	30	2004	2005	29
5000	5000	50	5001	5002	50

Nine experimental data on forward kinematic and real position data are listed in the table. The first data in the table shows that the robot is moved with forward kinematic data of (20,20) at a facing angle of 0 °. The condition of the robot in real position is at the coordinate point (19.21) with a facing angle of 0 °. The second data in the table shows that the robot is moved with forward kinematic data of (40.40) at a facing angle of 10 °. The condition of the robot in real position is at the coordinate point (41.40) with a face angle of 11 °.

The third data in the table shows that the robot is moved with forward kinematic data of (80.80) at a 45 ° angle. The condition of the robot in real position is at the coordinate point (81.81) with a facing angle of 46 °. The fourth data in the table shows that the robot is moved with forward kinematic data of (160,160) at a facing angle of 90 °. The condition of the robot in real position is at the coordinate point (162.159) with a facing angle of 90 °.

The fifth data in the table shows that the robot is moved with forward kinematic data of (300,300) at a facing angle of 180 °. The condition of the robot in real position is at the coordinate point (302,302) with a face angle of 181 °. The six data in the table shows that the robot is moved with forward kinematic data of (500,500) at a facing angle of 270 °. The condition of the robot in real position is at the coordinate point (501,502) with a face angle of 271 °.

The seventh data in the table shows that the robot is moved with forward kinematic data of (1000.1000) at a facing angle of 310 °. The condition of the robot in real position is at the coordinate point (1002,1002) with a face angle of 309 °. The eighth data in the table shows that the robot is moved with forward kinematic data of (2000,2000) at a facing angle of 30 °. The condition of the robot in real position is at the coordinate point (2004,2005) with a facing angle of 29 °.

The first data in the table shows that the robot is moved with forward kinematic data of (5000,5000) at a facing angle of 50 °. The condition of the robot in real position is at the coordinate point (5001,5002) with a facing angle of 50 °. From the table, there are nine data, it can be seen that the results of reading the robot coordinates using the forward kinematic equation have an average error of 1.67 mm at the x coordinate and 1.77 mm at the y coordinate and 0.67 ° on the robot heading reading. From these data we can say that reading the coordinates of the robot using the forward kinematic equation using 4 rotary encoders on all four wheels can be implemented very well.

B. Experiment and Validation of Inverse Kinematic Robot

In this second test, it is conducted to test how precise the robot's movement is to the inverse kinematic input value. The following are the results of the tests that have been carried out, can be seen in Figure 8 and Table 2. Based on the figures and tables, it can be seen that the results of the tests that have been carried out can be seen that using the inverse kinematic equation the robot can be moved in a certain direction on global coordinates. From several tests, the biggest error is 65 mm at the X coordinate and 55 mm at the Y coordinate, with an average error of 33 mm at the X coordinate and 28.4 mm at the Y coordinate.

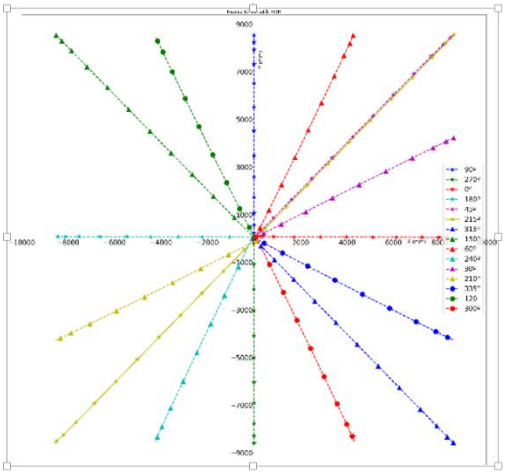

Fig. 8. Block Diagram of Control System

TABLE II. KINEMATIC FORWARD TESTING DATA

Target Coordinates (mm)		Robot Movement (mm)		Error (mm)	
X	Y	X	Y	X	Y
0	8545	0	8538	0	7
0	-8650	0	-8629	0	55
8567	53	8534	53	65	0
-8567	79	08550	77	52	1
8595	8545	8580	8537	41	27
-8559	8571	-8572	-8557	52	35
8595	-8571	8581	-8564	42	28
-8539	8545	-8529	8540	37	25
4569	8545	4264	8527	18	50
-4269	-8571	-4266	-8559	15	44
8623	4233	8594	4223	61	22
-8595	-4285	-8574	-4278	55	19
8595	-4259	8590	-4257	39	13
-4241	8518	-4235	8497	18	54
4325	-8544	4322	-8532	15	46

C. Experiment of Trajectory Tracking Control

1) Trajectory Circle

Based on the kinematic model of the ROD. The kinematic model for trajectory circles is described in the nonlinear function of the system as follows:

$$\begin{aligned} V_x &= \sin \omega t\ (radian) \\ V_y &= \cos \omega t\ (radian) \\ \theta &= 0 \end{aligned} \quad (20)$$

The results of the trajectory circle testing that have been carried out are shown in Figure 9. From this figure, it can be seen that there are three experiments on the trajectory circle by changing the speed variable. From the figure, it can be seen that the PWM 80 speed is shown in blue, the PWM 160 speed is shown in red and the PWM 240 speed is shown in the green graph. From the three experiments, it can be seen that by using different speeds, the circle trajectory image shows no errors.

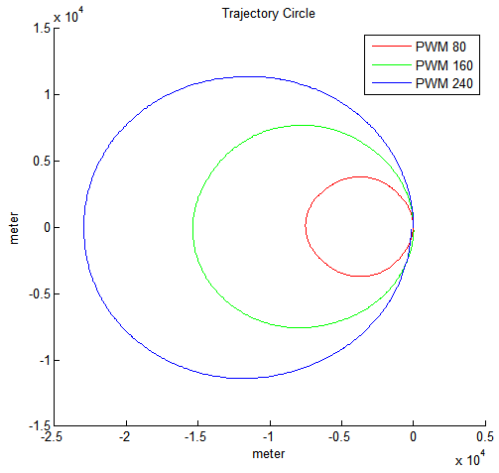

Fig. 9. Trajectory Circle Test Results

The PWM test graph for the trajectory circle is shown in Figure 10. From this figure, it can be seen that there are 3 PWM test data for the trajectory circle, each of which is 80, 160 and 240. From the graph, it can be seen that the four motors are given the same PWM value to be able to form the trajectory graph circle.

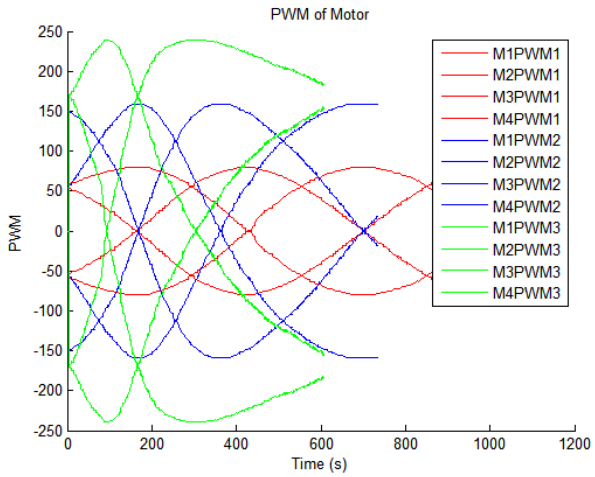

Fig. 10. PWM of Motor Test Results

The face-to-face test graph for the trajectory circle is shown in Figure 10. From this figure, it can be seen that there are 3 face-to-speed test data, each of which has a value of 80, 160 and 240. From the graph, it can be seen that the greater the speed, the greater the error in the direction of the robot.

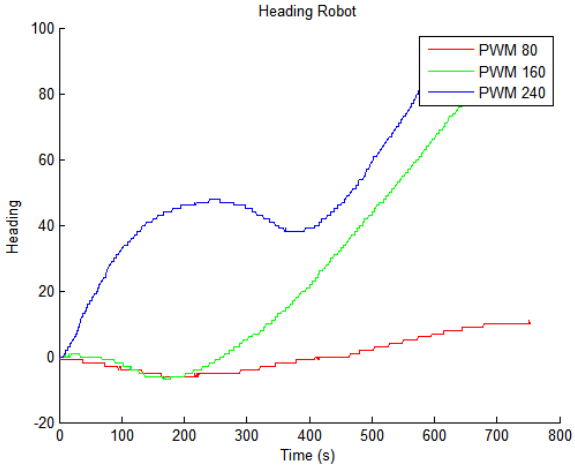

Fig. 11. Heading Robot Test Results

2) Trajectory Lemniscate

Based on the kinematic model of the ROD. The kinematic model for trajectory circles is described in the nonlinear function of the system as follows:

$$\begin{aligned} V_x &= \sin 2\ x\ \omega t\ (radian) \\ V_y &= \cos \omega t\ (radian) \\ \theta &= 0 \end{aligned} \quad (21)$$

The results of the Lemniscate trajectory test that have been carried out are shown in Figure 12. From this figure, it can be seen that there are three experiments on the Lemniscate trajectory by changing the speed variable. From the figure, it can be seen that the PWM 80 speed is shown in blue, the PWM 160 speed is shown in red and the PWM 240 speed is shown in the green graph. From the three experiments, it can be seen that using different speeds, the Lemniscate trajectory image shows no errors.

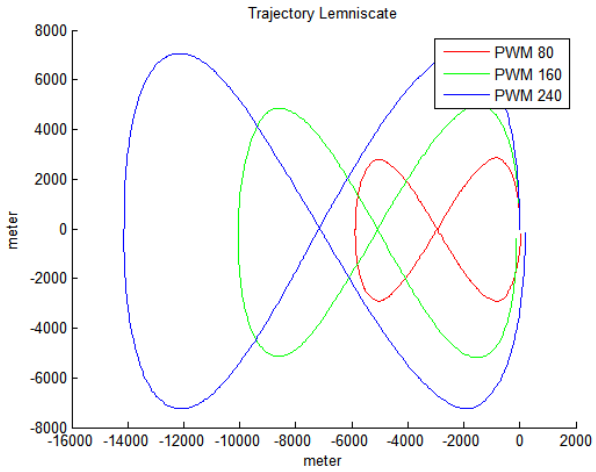

Fig. 12. Trajectory Lemniscate Test Results

The PWM test graph for the Lemniscate trajectory is shown in Figure 13. From this figure, it can be seen that there are 3 PWM test data for the Lemniscate trajectory, each of which is 80, 160 and 240. From this graph, it can be seen that the four motors are given the same PWM value to be able to form the trajectory graph Lemniscate.

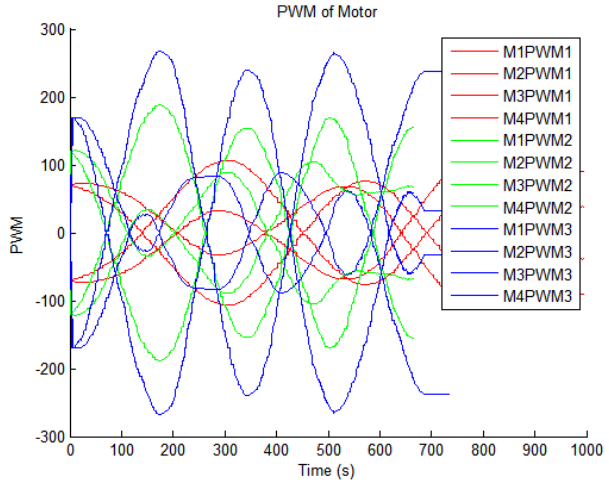

Fig. 13. PWM of Motor Test Results Lemniscate

The face-to-face test graph for Lemniscate trajectory is shown in Figure 14. From this figure, it can be seen that there are 3 test data facing the speed, each of which has a value of 80, 160 and 240. From this graph, it can be seen that the greater the speed, the greater the error in the direction of the robot.

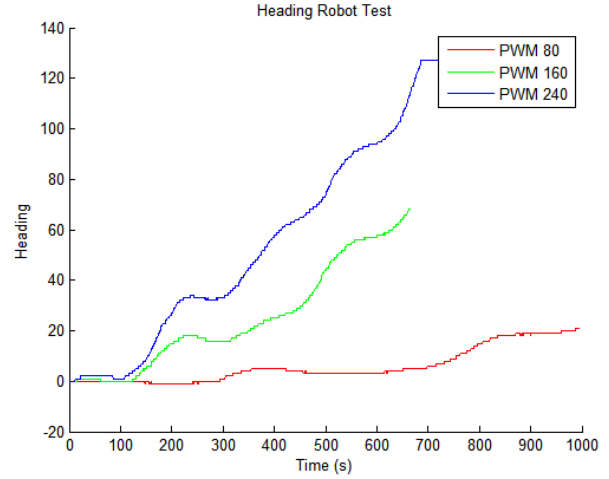

Fig. 14. Heading Robot Test Results Lemniscate

V. CONCLUSION

In this study, the implementation of kinematic modeling and odometry in the trajectory planning of the omnidirectional robot is proposed to control the movement of the omniwheel robot to follow the given path to reduce the robot's movement errors. From the results of the research that has been done, it can be seen that the use of odometric sensors produces better movement results when compared to using only kinematic control. In addition, using the robot odometry sensor can detect the value of the robot coordinates, robot headings, velocity and other related parameters. The drawbacks obtained in this study are the absence of a control such as PID or Fuzzy which is used to control some of these parameters, resulting in a change in the robot heading from the initial value of the robot movement. For further research, a control such as fuzzy or PID will be added with the addition of an IMU sensor to increase control of the movement of the robot in order to control the motor speed value and the heading of the robot.

REFERENCES

[1] P. Van Lam and Y. Fujimoto, "A Robotic Cane for Balance Maintenance Assistance," IEEE Trans. Ind. Informatics, vol. 15, no. 7, pp. 3998–4009, Jul. 2019.

[2] C. Cai, J. Lu, and Z. Li, "Kinematic Analysis and Control Algorithm for the Ballbot," IEEE Access, vol. 7, pp. 38314–38321, 2019.

[3] W. Li and R. Xiong, "Dynamical Obstacle Avoidance of Task-Constrained Mobile Manipulation Using Model Predictive Control," IEEE Access, vol. 7, pp. 88301–88311, 2019.

[4] M. Ferro, A. Paolillo, A. Cherubini, and M. Vendittelli, "Vision-Based Navigation of Omnidirectional Mobile Robots," IEEE Robot. Autom. Lett., vol. 4, no. 3, pp. 2691–2698, Jul. 2019.

[5] T. Terakawa, M. Komori, K. Matsuda, and S. Mikami, "A Novel Omnidirectional Mobile Robot With Wheels Connected by Passive Sliding Joints," IEEE/ASME Trans. Mechatronics, vol. 23, no. 4, pp. 1716–1727, Aug. 2018.

[6] B. A. Gebre and K. V. Pochiraju, "Machine Learning Aided Design and Analysis of a Novel Magnetically Coupled Ball Drive," IEEE/ASME Trans. Mechatronics, vol. 24, no. 5, pp. 1942–1953, Oct. 2019.

[7] P. Shen, X. Zhang, and Y. Fang, "Complete and Time-Optimal Path-Constrained Trajectory Planning With Torque and Velocity Constraints: Theory and Applications," IEEE/ASME Trans. Mechatronics, vol. 23, no. 2, pp. 735–746, Apr. 2018.

[8] M. A. Al Mamun, M. T. Nasir, and A. Khayyat, "Embedded System for Motion Control of an Omnidirectional Mobile Robot," IEEE Access, vol. 6, no. 8, pp. 6722–6739, 2018.

[9] Y. Fei and H. Xu, "Modeling and Motion Control of a Soft Robot," IEEE Trans. Ind. Electron., vol. 64, no. 2, pp. 1737–1742, Feb. 2017.

[10] M. Thor and P. Manoonpong, "A Fast Online Frequency Adaptation Mechanism for CPG-Based Robot Motion Control," IEEE Robot. Autom. Lett., vol. 4, no. 4, pp. 3324–3331, Oct. 2019.

[11] B. Ren, Y. Wang, and J. Chen, "A Novel Robust Finite-Time Trajectory Control With the High-Order Sliding Mode for Human–Robot Cooperation," IEEE Access, vol. 7, pp. 130874–130882, 2019.

[12] H. Chang, S. J. Kim, and J. Kim, "Feedforward Motion Control With a Variable Stiffness Actuator Inspired by Muscle Cross-Bridge Kinematics," IEEE Trans. Robot., vol. 35, no. 3, pp. 747–760, Jun. 2019.

[13] J. Chi, H. Yu, and J. Yu, "Hybrid Tracking Control of 2-DOF SCARA Robot via Port-Controlled Hamiltonian and Backstepping," IEEE Access, vol. 6, pp. 17354–17360, 2018.

[14] F. Angelini et al., "Decentralized Trajectory Tracking Control for Soft Robots Interacting With the Environment," IEEE Trans. Robot., vol. 34, no. 4, pp. 924–935, Aug. 2018.

[15] L. Ge, H. Wang, and J. Xing, "Maintenance robot motion control based on Kinect gesture recognition," J. Eng., vol. 2019, no. 23, pp. 8794–8796, Dec. 2019.

[16] S. Verma, D. Shen, and J. Xu, "Motion Control of Robotic Fish Under Dynamic Environmental Conditions Using Adaptive Control Approach," IEEE J. Ocean. Eng., vol. 43, no. 2, pp. 381–390, Apr. 2018.

[17] A. Magassouba, N. Bertin, and F. Chaumette, "Exploiting the Distance Information of the Interaural Level Difference for Binaural Robot Motion Control," IEEE Robot. Autom. Lett., vol. 3, no. 3, pp. 2048–2055, Jul. 2018.

[18] H. Ashrafiuon, S. Nersesov, and G. Clayton, "Trajectory Tracking Control of Planar Underactuated Vehicles," IEEE Trans. Automat. Contr., vol. 62, no. 4, pp. 1959–1965, Apr. 2017.

[19] Q. Zheng, Y. He, X. Qi, P. Zhang, Y. Hu, and B. Li, "Safety Tracking Motion Control Based on Forbidden Virtual Fixtures in Robot Assisted Nasal Surgery," IEEE Access, vol. 6, no. c, pp. 44905–44916, 2018.

[20] Z. Sun, Y. Xia, L. Dai, K. Liu, and D. Ma, "Disturbance Rejection MPC for Tracking of Wheeled Mobile Robot," IEEE/ASME Trans. Mechatronics, vol. 22, no. 6, pp. 2576–2587, Dec. 2017.
[21] D. Nakhaeinia, P. Payeur, and R. Laganiere, "A Mode-Switching Motion Control System for Reactive Interaction and Surface Following Using Industrial Robots," IEEE/CAA J. Autom. Sin., vol. 5, no. 3, pp. 670–682, May 2018.
[22] Z. Zhang, J. Dequidt, J. Back, H. Liu, and C. Duriez, "Motion Control of Cable-Driven Continuum Catheter Robot Through Contacts," IEEE Robot. Autom. Lett., vol. 4, no. 2, pp. 1852–1859, Apr. 2019.
[23] Y. Huang, D. M. Pool, O. Stroosma, and Q. Chu, "Long-Stroke Hydraulic Robot Motion Control With Incremental Nonlinear Dynamic Inversion," IEEE/ASME Trans. Mechatronics, vol. 24, no. 1, pp. 304–314, Feb. 2019.
[24] J. Yu, Z. Wu, Z. Su, T. Wang, and S. Qi, "Motion Control Strategies for a Repetitive Leaping Robotic Dolphin," IEEE/ASME Trans. Mechatronics, vol. 24, no. 3, pp. 913–923, Jun. 2019.
[25] M. Bjelonic et al., "Keep Rollin'—Whole-Body Motion Control and Planning for Wheeled Quadrupedal Robots," IEEE Robot. Autom. Lett., vol. 4, no. 2, pp. 2116–2123, Apr. 2019.
[26] T. Salumae, A. Chemori, and M. Kruusmaa, "Motion Control of a Hovering Biomimetic Four-Fin Underwater Robot," IEEE J. Ocean. Eng., vol. 44, no. 1, pp. 54–71, Jan. 2019.
[27] R. A. S. Fernandez, Z. Milosevic, S. Dominguez, and C. Rossi, "Motion Control of Underwater Mine Explorer Robot UX-1: Field Trials," IEEE Access, vol. 7, pp. 99782–99803, 2019.
[28] H. Chang, S. Wang, and P. Sun, "Omniwheel Touchdown Characteristics and Adaptive Saturated Control for a Human Support Robot," IEEE Access, vol. 6, pp. 51174–51186, 2018.