

Implementasi Kinematika *Trajectory* Lingkaran pada Robot Roda *Mecanum*

Fahmizal^{1*}, Angga Priyatmoko¹, Afrizal Mayub²

¹Departemen Teknik Elektro dan Informatika, Universitas Gadjah Mada; angga.priyatmoko@mail.ugm.ac.id

²Pascasarjana Pendidikan IPA, Universitas Bengkulu; afrizalmayub@unib.ac.id

*Korespondensi: fahmizal@ugm.ac.id

Abstract - *The movement of the non-holonomic mobile robot is still not efficient because it can only move forward, backward, and turn so that it requires a wheel platform that is able to move more efficiently and more variedly. The use of mecanum wheels on wheeled robots is one of the best solutions because the mecanum wheels can move in all directions and are more stable. This paper proposes a mechanical robot trajectory kinematics design with a simple and easy mechanism to implement using an embedded system such as using Arduino. From the results of this study, it was found that the robot was able to maneuver following a circular trajectory with a radius of 50 cm in 7 seconds.*

Keywords - *mobile robot, mecanum robot, inverse kinematics, trajectory*

Intisari - Pergerakan *mobile robot non holonomic* masih belum efisien karena hanya dapat bergerak maju, mundur, berbelok sehingga membutuhkan platform roda yang mampu bergerak lebih efisien dan lebih variatif. Penggunaan roda *mecanum* pada robot beroda menjadi salah satu solusi terbaik karena roda *mecanum* dapat bergerak ke segala arah dan lebih stabil. Makalah ini mengusulkan sebuah perancangan desain kinematika *trajectory* robot *mecanum* dengan mekanisme yang sederhana dan mudah untuk diimplementasikan menggunakan *embeded system* seperti menggunakan Arduino. Dari hasil penelitian ini diperoleh bahwa robot mampu bermanuver mengikuti *trajectory* lingkaran dengan radius 50 cm dalam waktu 7 detik.

Kata kunci - *robot beroda, roda mecanum, invers kinematika, trajectory*

I. PENDAHULUAN

Roda *mecanum* masuk ke dalam kategori robot *holonomic*. Berbeda dengan roda pada umumnya harus berbelok ketika menggerakkan dan tidak dapat bergerak ke segala arah sedangkan roda *mecanum* dapat bergerak ke segala arah tanpa harus berbelok. Roda *mecanum* disebut juga "Roda Ikon", jenis roda ini ditemukan pertama oleh Bengt Ikon dari Swedia [1].

Robot roda *mecanum* merupakan jenis robot *omnidirectional mobile robot* yang berperan penting dalam transportasi dan industri modern karena fleksibilitas dan kemampuan manuvernya yang tinggi. Platform dengan roda *mecanum* memiliki keunggulan besar dibandingkan desain konvensional dalam hal mobilitas di lingkungan yang padat. Lingkungan ini biasanya ditemukan di kantor, bengkel pabrik, gudang, rumah sakit, dan fasilitas perawatan lansia.

Roda *mecanum* menggunakan desain roda yang dikelilingi oleh *roller* dengan membentuk sudut 45°. Arah dan kecepatan masing-masing roda menghasilkan resultan gaya yang menerjemahkan pergerakan mobilitas tanpa mengubah arah hadap robot (*heading*) ataupun perubahan sudut putar *yaw* roda [2]-[4]. Namun terdapat kelemahan pada platform roda *mecanum*, yaitu apabila kondisi lantai yang tidak rata, masalah slip roda akan muncul [5].

Automatic guided vehicle (AGV) biasanya menggunakan platform roda *mecanum* dalam pengoperasiannya. Robot AGV merupakan suatu pengembangan dari teknologi otomasi dibidang *mobile robot* pada *warehouse industry* [6], [7]. Penggunaan AGV dengan roda *mecanum* sangat bermanfaat untuk dunia industri karena lebih fleksibel dan daya tahan serta mobilitasnya. Agar robot dapat bergerak ke segala arah maka membutuhkan perhitungan kinematika gerak sehingga

dalam pergerakannya dapat menghasilkan pola-pola seperti *trajectory* bentuk lingkaran.

Penelitian yang dilakukan oleh Zhe Sun dkk [8] membahas simulasi pelacakan *trajectory* lintasan dari robot *mecanum* dengan metode *adaptive integral terminal sliding mode* (AITSM). Serta dilakukan juga oleh peneliti Trinh dkk. [9] membahas perancangan *trajectory* lintasan dari robot *Mecanum* dengan menggunakan metode *non-uniform rational basis spline* (NURBS). Selanjutnya diteliti juga oleh Veer Alakshendra dkk. [10] yang membahas *adaptive robust second-order sliding mode control* (ARSSMC) untuk kasus pengendalian *trajectory* secara simulasi.

Dari tinjauan beberapa pustaka, umumnya para peneliti melakukan pembahasan perancangan *trajectory* robot *mecanum* dengan pengujian di level simulasi. Makalah ini mengusulkan sebuah perancangan desain *trajectory* robot *mecanum* dengan mekanisme yang sederhana dan mudah untuk diimplementasikan menggunakan *embeded system* seperti menggunakan Arduino.

Susunan pada makalah ini disusun menjadi lima bagian. Pada bab 2, konfigurasi robot *mecanum* dalam penelitian ini diusulkan. Di bab 3 perancangan desain *trajectory* pada robot *mecanum* dijelaskan. Prosedur penerapan persamaan *inverse kinematika* (IK) robot *mecanum* di bab 4 beserta hasil data eksperimen. Bab 5 menjelaskan kesimpulan dari penelitian ini.

II. KONFIGURASI ROBOT MECANUM

A. Roda Mecanum

Roda *mecanum* yang digunakan pada penelitian ini berdiameter 6 cm dengan *rollers* dipasang membentuk sudut 45° seperti yang disajikan pada Gambar 1, sehingga dapat

menyelesaikan rintangan dengan mudah dan posisi pemasangan roda *mecanum* berbeda dengan roda *Omni* seperti pada Gambar 2.

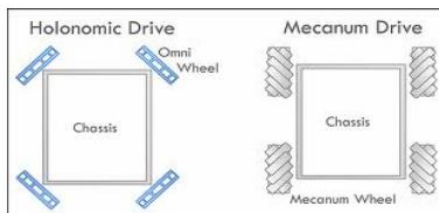
Kombinasi arah putar pada letak masing-masing roda *mecanum* akan mempengaruhi arah atau resultan gaya yang mengakibatkan perpindahan posisi robot *mecanum* seperti pada Gambar 3.

B. Desain Badan Robot Mecanum

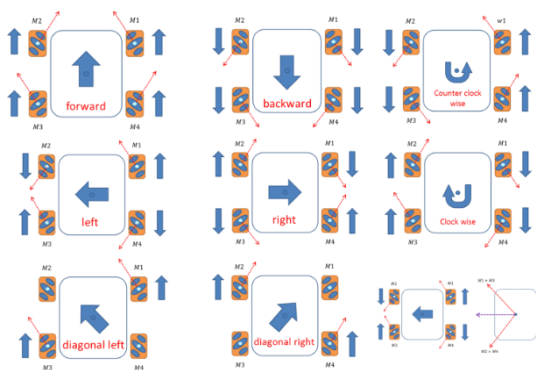
Desain mekanik sangat berpengaruh pada kinerja robot *mecanum* karena desain harus dibuat secara simetris dan sepresisi mungkin sehingga dapat mempermudah dalam menyesuaikan pengaturan perhitungan kinematika untuk pergerakannya. Pada penelitian ini badan robot dibuat menggunakan bahan akrilik dengan ketebalan 5 mm dengan bantuan perangkat lunak Corel Draw seperti yang disajikan pada Gambar 4. Hasil keseluruhan didesain badan robot *mecanum* yang telah disusun dengan perangkat elektroniknya disajikan pada Gambar 5. Badan robot *mecanum* dibuat dengan ukuran panjang antar sumbu 150 mm yang membentuk segi empat.



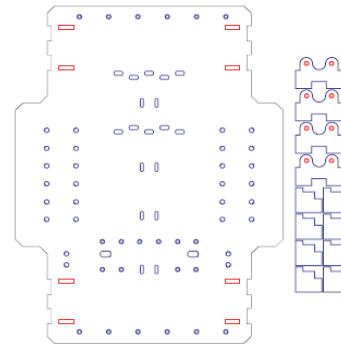
Gambar 1. Roda *mecanum* dengan rollers 45° [5]



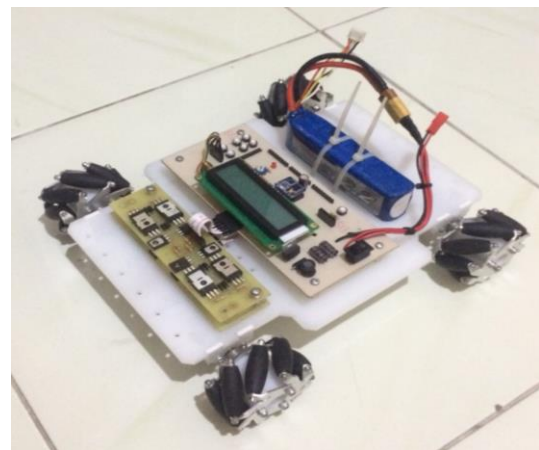
Gambar 2. Perbedaan posisi pemasangan antara roda *omni* dan roda *mecanum* [8]



Gambar 3. Resultan gaya perpindahan robot *mecanum* akibat dari kombinasi arah putar pada masing-masing roda *mecanum*



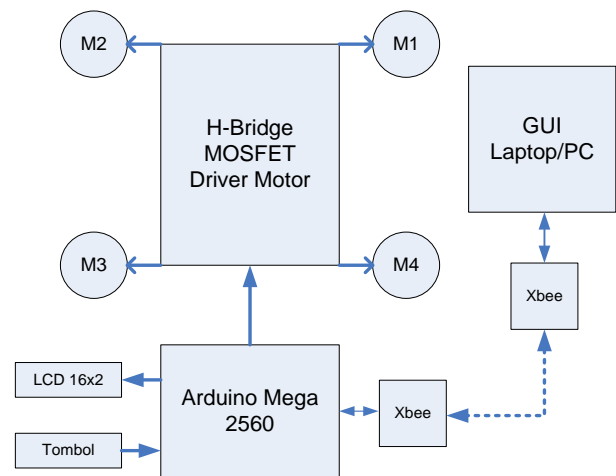
Gambar 4. Desain mekanik *body* robot *mecanum* dengan Corel Draw



Gambar 5. Realisasi desain mekanik dari robot *mecanum*

C. Desain Elektronika Robot Mecanum

Pada penelitian ini perangkat pengendali elektronis robot *mecanum* menggunakan Arduino Mega 2560 yang memiliki memori *flash* sebesar 256KB (8KB telah berisi *bootloader*), SRAM 8KB, dan EEPROM 4KB. Alasan pemilihan Arduino Mega karena nantinya memori *trajectory* lintasan robot *mecanum* membutuhkan memori SRAM di atas 1 Kb. Desain elektronika robot *mecanum* pada penelitian ini secara menyeluruh disajikan pada Gambar 6.



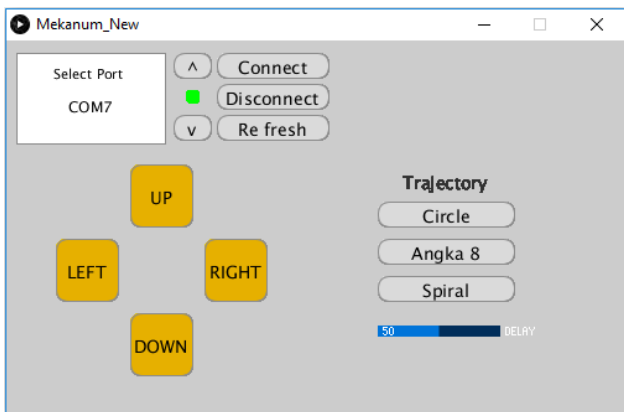
Gambar 6. Desain elektronika robot *mecanum*

Selain itu, desain komunikasi robot *mecanum* dirancang secara *wireless* (garis putus-putus), menggunakan modul X-Bee serta aplikasi Processing IDE sebagai *graphical user interface* (GUI) sebagai pembentuk antarmuka robot dengan *user*. Modul X-Bee yang digunakan adalah X-Bee S2C yang ditunjukkan seperti pada Gambar 7. X-Bee sendiri merupakan suatu modul yang dapat melakukan komunikasi serial secara *wireless* pada frekuensi 2,4 GHz. Untuk dapat mengakses X-Bee menggunakan aplikasi bernama XCTU [11].

Desain GUI yang dipergunakan pada penelitian ini disajikan pada Gambar 8. GUI ini dirancang untuk memberikan perintah *trajectory* yang telah ditanamkan pada Arduino Mega 2560. Adapun *trajectory* yang dirancang pada penelitian ini meliputi bentuk lingkaran (*circle*), bentuk angka 8 dan bentuk spiral.



Gambar 7. X-Bee S2C

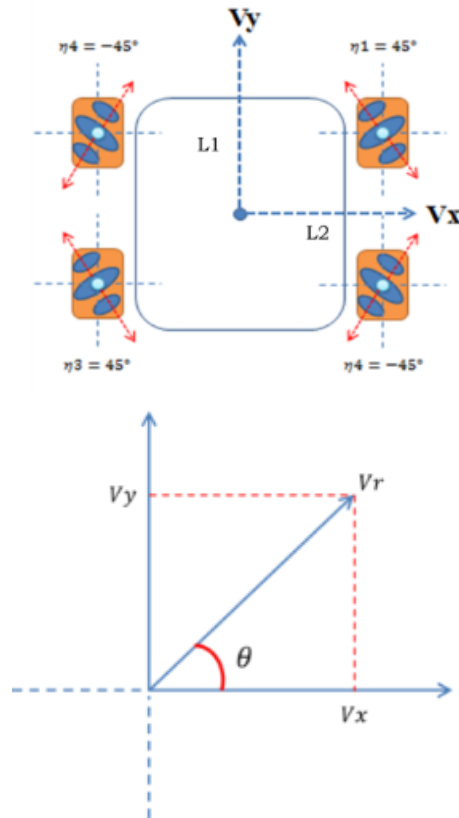


Gambar 8. Hasil rancangan GUI *trajectory* robot *mecanum* menggunakan Processing IDE

III. DESAIN TRAJECTORY ROBOT MECANUM

A. Kinematika pada Robot Mecanum

Penurunan kinematika robot *mecanum* merupakan tahap awal dalam perancangan desain *trajectory*. Pada penelitian ini diperlukan formula persamaan kinematika terbalik/*inverse* (IK). Persamaan ini akan digunakan sebagai dasar pergerakan robot agar mampu bergerak ke segala arah. Persamaan IK digunakan saat menentukan kecepatan putaran motor yang diperlukan untuk memenuhi gerakan yang diinginkan dari platform roda *mecanum*. Demikian pula, persamaan kinematika maju/*forward* (FK) dapat digunakan untuk menentukan *trajectory* robot *mecanum* dari kecepatan rotasi motor saat ini. Persamaan FK dan IK dari konfigurasi robot *mecanum* disajikan pada (1) dan (2).



Gambar 9. Konfigurasi dan resultan gaya robot *mecanum*

Dengan merujuk pada konfigurasi desain robot *mecanum* pada Gambar 9. Di mana R adalah jari-jari roda; ω_i adalah kecepatan sudut roda ke- i ($i = 1 \dots 4$); l_1, l_2 adalah jarak antara sumbu roda dan pusat badan robot *mecanum* dan $L=l_1+l_2$.

$$\begin{bmatrix} V_x \\ V_y \\ \omega_z \end{bmatrix} = \frac{R}{4} \begin{bmatrix} -1 & 1 & -1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & -1 & -1 & 1 \\ l_1 + l_2 & -l_1 + l_2 & -l_1 + l_2 & l_1 + l_2 \end{bmatrix} \begin{bmatrix} \omega_{M1} \\ \omega_{M2} \\ \omega_{M3} \\ \omega_{M4} \end{bmatrix} \quad (1)$$

$$\begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \end{bmatrix} = \begin{bmatrix} R\omega_1 \\ R\omega_2 \\ R\omega_3 \\ R\omega_4 \end{bmatrix} = \begin{bmatrix} 1 & 1 & -(l_1 + l_2) \\ 1 & -1 & (l_1 + l_2) \\ 1 & -1 & -(l_1 + l_2) \\ 1 & 1 & (l_1 + l_2) \end{bmatrix} \begin{bmatrix} V_x \\ V_y \\ \omega_z \end{bmatrix} \quad (2)$$

Gambar 9 menunjukkan persamaan vektor roda *mecanum*. Persamaan 3 menjadi acuan dalam menurunkan Persamaan IK. Hasil dari persamaan IK yang diturunkan akan diimplementasikan pada setiap roda *mecanum*. Dengan memahami resultan kecepatan (V_r) dan arah (θ), diperoleh kecepatan pada sumbu y (V_y) dan sumbu x (V_x), nilai-nilai ini akan digunakan dalam solusi persamaan IK pada robot *mecanum*. Langkah-langkah menurunkan persamaan tersebut dapat dijelaskan sebagai berikut:

$$V_r = \sqrt{V_x^2 + V_y^2} \quad (3)$$

$$\theta = \tan^{-1} \frac{V_y}{V_x} \quad (4)$$

$$V_r^2 = V_x^2 + V_y^2 \quad (5)$$

$$V_y = \sqrt{V_r^2 - V_x^2} \quad (6)$$

Dengan mensubstitusi (6) ke (4), diperoleh:

$$\theta = \tan^{-1} \frac{V_y}{V_x}$$

$$\theta = \tan^{-1} \left(\frac{V_y^2}{V_x^2} \right)^{\frac{1}{2}}$$

$$\theta = \tan^{-1} \left(\frac{V_r^2 - V_x^2}{V_x^2} \right)^{\frac{1}{2}} \quad (7)$$

$$(\tan \theta)^2 = \frac{V_r^2 - V_x^2}{V_x^2}$$

$$V_x^2 = \frac{V_r^2}{(\tan \theta)^2 + 1}$$

$$V_x = \sqrt{\frac{V_r^2}{(\tan \theta)^2 + 1}}$$

Dengan menggunakan (6) dan (7), nilai V_y dan V_x akan digunakan dalam solusi persamaan IK pada robot *mecanum*. Data robot dan roda yang digunakan dalam penelitian ini dideskripsikan pada Tabel 1.

B. Desain Trajectory

Setelah formula persamaan IK diperoleh maka selanjutnya dapat menerapkan *trajectory* yang akan diterapkan pada robot *mecanum*. Pada penelitian ini mencoba merancang tiga jenis *trajectory* yaitu lingkaran, angka 8 dan spiral dengan mengacu pada (8)-(10). Dimana nilai t adalah $0 \leq t \leq 6\pi$.

$$\begin{bmatrix} V_x \\ V_y \\ \theta \end{bmatrix} = \begin{bmatrix} \cos t \\ \sin t \\ 0 \end{bmatrix} \quad (8)$$

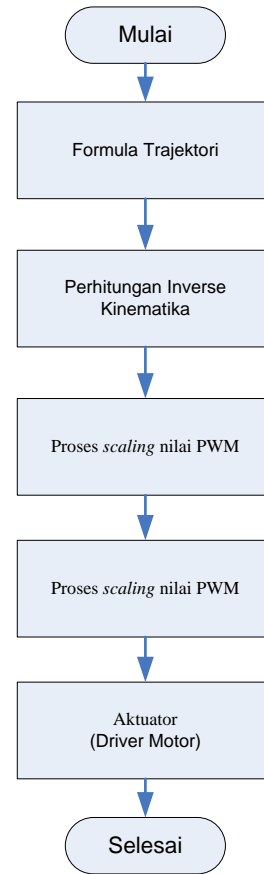
$$\begin{bmatrix} V_x \\ V_y \\ \theta \end{bmatrix} = \begin{bmatrix} \cos t \\ \frac{\sin(2t)}{2} \\ 0 \end{bmatrix} \quad (9)$$

$$\begin{bmatrix} V_x \\ V_y \\ \theta \end{bmatrix} = \begin{bmatrix} t \cos \omega t \\ t \sin \omega t \\ 0 \end{bmatrix} \quad (10)$$

Selanjutnya agar aktuator dapat bergerak maka memerlukan proses *scaling* seperti yang disajikan pada Gambar 10 kedalam *range* nilai *pulse width modulation* (PWM). Pada penelitian ini menggunakan PWM 8-bit sehingga *range* nilai PWM berada pada (0-255).

Table 1. Data robot dan parameter roda robot *mecanum*

| Parameter | Panjang (mm) |
|---------------|--------------|
| Diameter roda | 60 |
| Panjang $l1$ | 150 |
| Panjang $l2$ | 150 |



Gambar 10. Tahapan scaling PWM aktuator

IV. HASIL DAN PEMBAHASAN

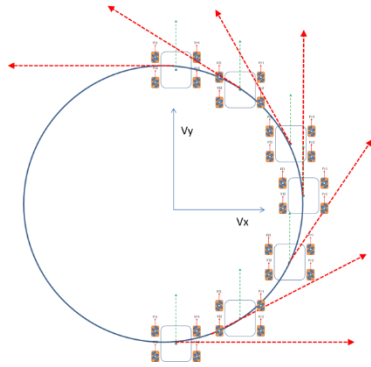
Pada tahapan pengujian, diuji coba dengan cara memberi masukan nilai $\begin{bmatrix} V_x \\ V_y \\ \theta \end{bmatrix}$ sesuai *trajectory* yang diinginkan.

Percobaan pertama adalah *trajectory* lingkaran dengan menggunakan formula pada (8). Tabel 2 menyajikan hasil data masukan untuk proses perhitungan persamaan IK pada kasus *trajectory* lingkaran.

Dari data V_x dan V_y apabila disajikan dalam bentuk grafis dua dimensi maka dapat dilihat seperti pada Gambar 11. Dari data Tabel 2 maka dapat diperoleh hasil perhitungan persamaan IK pada Tabel 3.

Tabel 2. Data masukan untuk proses hitung IK pada *trajectory* lingkaran

| θ (radians) | V_x | V_y |
|-----------------------|-------------|--------------|
| 0 | 1 | 0 |
| 0.1 | 0.995004165 | 0.099833417 |
| 0.2 | 0.980066578 | 0.198669331 |
| 0.3 | 0.955336489 | 0.295520207 |
| 0.4 | 0.921060994 | 0.389418342 |
| 0.5 | 0.877582562 | 0.479425539 |
| ... | ... | ... |
| 6.2 | 0.996542097 | -0.083089403 |



Gambar 11. Ilustrasi hasil plot dua dimensi data dari Tabel 2

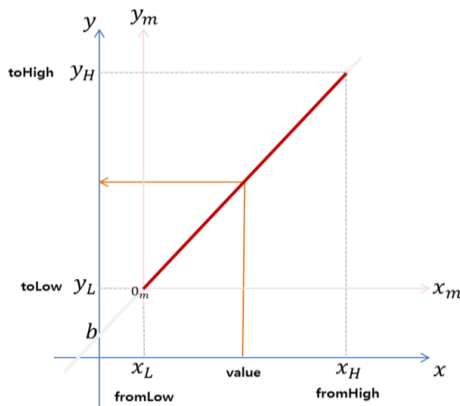
Tabel 3. Data hasil persamaan IK pada (8)

| θ (radians) | $\omega 1$ | $\omega 2$ | $\omega 3$ | $\omega 4$ |
|-----------------------|------------|------------|------------|------------|
| 0 | 1 | 1 | 1 | 1 |
| 0.1 | 1.0948 | 0.8952 | 0.8952 | 1.0948 |
| 0.2 | 1.1787 | 0.7814 | 0.7814 | 1.1787 |
| 0.3 | 1.2509 | 0.6598 | 0.6598 | 1.2509 |
| 0.4 | 1.3105 | 0.5316 | 0.5316 | 1.3105 |
| 0.5 | 1.3570 | 0.3982 | 0.3982 | 1.3570 |
| ... | ... | ... | ... | ... |
| 6.2 | 0.9135 | 1.0796 | 1.0796 | 0.9135 |

Hasil IK dan *trajectory* dari Tabel 3 nilainya masih kecil sehingga tidak dapat langsung dimasukkan ke dalam fungsi PWM pada perangkat Arduino, oleh karena itu perlu dilakukan *scaling* agar nilai tersebut sesuai dengan nilai PWM pada *range* (0-255). Metode *scaling* ini dijelaskan pada Gambar 12 dengan (11).

$$y = \left(\frac{y_H - y_L}{x_H - x_L}\right)(x - x_L) + y_L \tag{11}$$

Setelah tahapan *scaling* ini dilewati dengan hasil pada Tabel 4 maka nilai kecepatan pada masing-masing roda *mecanum* sudah dalam range nilai PWM antara 0-255. Nilai dari Tabel 4 ini membentuk pola *trajectory* lingkaran. Dari hasil percobaan diperoleh bahwa robot mampu bermanuver mengikuti *trajectory* lingkaran dengan radius 50 cm dalam waktu 7 detik.



Gambar 12. Grafik proses *scaling* nilai PWM

Tabel 4. Hasil data Tabel 3 yang sudah diskalakan dengan nilai PWM

| θ (radians) | $\omega 1$ | $\omega 2$ | $\omega 3$ | $\omega 4$ |
|-----------------------|------------|------------|------------|------------|
| 0 | 180 | 180 | 180 | 180 |
| 0.1 | 183 | 177 | 177 | 183 |
| 0.2 | 186 | 173 | 173 | 186 |
| 0.3 | 189 | 171 | 171 | 189 |
| 0.4 | 192 | 167 | 167 | 192 |
| 0.5 | 195 | 164 | 164 | 195 |
| ... | ... | ... | ... | ... |
| 6.2 | 177 | 183 | 183 | 177 |

V. SIMPULAN

Makalah ini menyajikan sebuah prosedur tahapan dalam penerapan kinematika balik/inverse (IK) dan dilanjut dengan penerapan *trajectory* lingkaran pada robot *mecanum*. Proses *scaling* ke dalam nilai *range* PWM (0-255) dari hasil perhitungan IK menjadi hal yang sangat penting sebelum diterapkan pada masing-masing roda *mecanum*. Dari hasil percobaan diperoleh bahwa robot mampu bermanuver mengikuti *trajectory* lingkaran dengan radius 50 cm dalam waktu 7 detik.

UCAPAN TERIMA KASIH

Penelitian ini didukung oleh Laboratorium Instrumentasi dan Kendali, Departemen Teknik Elektro dan Informatika, Sekolah Vokasi, Universitas Gadjah Mada.

DAFTAR PUSTAKA

- [1] B. Ilon, Wheels for a course stable self propelling vehicle movable in any desired direction on the ground or some other base. U. S. A. Patent, 1975.
- [2] Yadav, P. S., Agrawal, V., Mohanta, J. C., & Ahmed, M. F. (2022). A robust sliding mode control of mecanum wheel-chair for trajectory tracking. *Materials Today: Proceedings*.
- [3] Li, S., Zhang, J., Zhao, K., Zhang, Y., Sun, Z., & Xia, Y. (2022). Trajectory tracking control for four-mecanum-wheel mobile vehicle: A variable gain active disturbance rejection control approach. *International Journal of Robust and Nonlinear Control*, 32(4).
- [4] Filomeno Amador, L. D., & Castillo Castañeda, E. (2022). Kinematic and dynamic analysis of an omnidirectional mobile platform driven by a spherical wheel. *Mechanical Sciences*, 13(1), 31-39.
- [5] Fahmizal, Kuo, C. H. (2016, August). Trajectory and heading tracking of a mecanum wheeled robot using fuzzy logic control. In *2016 International Conference on Instrumentation, Control and Automation (ICA)* (pp. 54-59). IEEE.
- [6] Le, T. L., Nghin, D. V., & Aly, M. (2022). Design and Fabrication of Mecanum Wheel for Forklift Vehicle. In *Modern Mechanics and Applications* (pp. 795-810). Springer, Singapore.
- [7] Adam, N., Aiman, M., Nafis, W. M., Irawan, A., Muaz, M., Hafiz, M., ... & Ali, S. N. S. (2017). Omnidirectional configuration and control approach on mini heavy loaded forklift autonomous guided vehicle. In *MATEC Web of Conferences* (Vol. 90, p. 01077). EDP Sciences.
- [8] Sun, Z., Hu, S., He, D., Zhu, W., Xie, H., & Zheng, J. (2021). Trajectory-tracking control of Mecanum-wheeled omnidirectional mobile robots using adaptive integral terminal sliding mode. *Computers & Electrical Engineering*, 96, 107500.
- [9] Ly, T. T. K., Thai, N. H., Dzung, L. Q., & Thanh, N. T. (2020, December). Determination of kinematic control parameters of omnidirectional AGV robot with mecanum wheels track the reference trajectory and velocity. In *International Conference on Engineering Research and Applications* (pp. 319-328). Springer, Cham.

- [10] Alakshendra, V., & Chiddarwar, S. S. (2017). Adaptive robust control of Mecanum-wheeled mobile robot with uncertainties. *Nonlinear Dynamics*, 87(4), 2147-2169.
- [11] Kumbhar, H. (2016, August). Wireless sensor network using Xbee on Arduino Platform: An experimental study. In *2016 International Conference on Computing Communication Control and automation (ICCUBEA)* (pp. 1-5). IEEE