Designing and controlling a Mecanum-wheeled mobile robot: From the concept to the implementation via Kinematic Control.

José C. Ortiz Hernández [a], *David I. Rosas Almeida* [a]

[a] Facultad de Ingeniería Mexicali, Universidad Autónoma de Baja California, jose.carlos.ortiz.hernández@uabc.edu.mx, drosas@uabc.edu.mx, Mexicali, Baja California, México.

Resumen

Este trabajo presenta el diseño, construcción y control de un robot móvil con ruedas Mecanum. En cuanto al diseño, se presenta la construcción de un prototipo de arquitectura abierta con forma rectangular de cuatro ruedas Mecanum, que puede ser útil para hacer investigación y enseñanza de robótica y control. En cuanto al control, se implementa un enfoque denominado como control cinemático, el cual se explica típicamente mediante una jerarquía de controles conocidos como alto y bajo nivel. Por una parte, el control de alto nivel se encarga de regular la posición del robot parametrizada en términos de coordenadas cartesianas, lo que es útil para tareas de navegación. Por otra parte, el control de bajo nivel consiste en ajustar la velocidad de rotación de cada rueda del robot. Este enfoque de control trata el modelo dinámico como una caja negra, considerando solo la entrada y salida como conocido. Para el control de alto nivel implementamos un control Proporcional-Integral-Derivativo, mientras que para el bajo nivel usamos controladores Proporcional-Derivativo. Los resultados experimentales que aquí se presentan se programaron en Matlab y se utilizó la tarjeta en tiempo real dSPACE-DS1103 para la adquisición de datos y generación de señales. Presentamos un procedimiento detallado para ofrecer reproducibilidad al lector.

Palabras clave—robot móvil, rueda Mecanum, control cinemático, alto nivel, bajo nivel.

Abstract

This work presents the design, construction, and control of a mobile robot equipped with Mecanum wheels. Regarding the design, we describe the development of an open-architecture prototype with a rectangular four-wheel Mecanum configuration, which can be useful for research and teaching in robotics and control. In terms of control, we implement an approach known as kinematic control, which is typically explained through a hierarchy of high-level and low-level controllers. The high-level control is responsible for regulating the robot's position, parameterized in Cartesian coordinates, which is particularly useful for navigation tasks. Meanwhile, the low-level control adjusts the rotational speed of each wheel. This control approach treats the dynamic model as a black box, considering only the input and output as known variables. For high-level control, we implement a Proportional-Integral-Derivative (PID) controller, while for low-level control, we use Proportional-Derivative (PD) controllers. The experimental results presented in this work were programmed in MATLAB, utilizing the dSPACE DS1103 real-time board for data acquisition and signal generation. We provide a detailed procedure to ensure reproducibility for the reader.

Keywords— *Mobile robot, Mecanum wheel, kinematic control, high-level, low-level.*

1. Introduction

Mobile robots have been a topic of interest to the scientific community in the past years due to the growing technologies in mobile automation. Their applications can be observed in various sectors, including but not limited to manufacturing, healthcare, warehousing, agriculture, exploration, and others, see, for example, [1] and [2]. Therefore, an urgent demand has arisen to deal with more complex tasks requiring greater maneuverability in mobile robots for navigation in indoor and outdoor environments. To this end, mobile robots equipped with Mecanum wheels may be the new paradigm shift in mobile robotics for the challenges demanded by the modern world. The Mecanum wheel, invented by Bengt Illon in 1973 [3], is composed of passive rollers made of a rubber alloy, fixed onto the wheel, mounted at a 45-degree angle concerning the wheel's axis of rotation [4], [5]. The Mecanum wheel has been proven superior among the variety of other wheel types in terms of high-load capacity and easy installation, while still achieving omnidirectional movements [6]. The omnidirectional movements are forward, backward, left and right sideways, diagonal at any angle, and rotate around its axis, all while maintaining their same pose or angle of orientation [7], [8]. Despite other wheels that provide omnidirectional locomotion, the Mecanum wheel has been raised as an option for large-scale applications due to its ability to support heavy loads and easy assembly to the mobile robot platform, which commonly is directly attached to a DC motor and the chassis [9]. This class of omnidirectional mobile robots have been being implemented in many areas. For instance, in the healthcare field, as medication delivery and distribution mobile robots [10] or electric wheelchairs for people with disabilities [11], [12]. Industrial fields, in the heavy-duty material handling robots in manufacturing industries [13], among others.

The autonomous control design is the hearth for a mobile robotic system and plays a key role. Here, the most elemental autonomous control process is based on the robot's kinematic model to achieve navigation. This approach is known as kinematic control and still remains as a reliability approach [14]. Kinematic control involves designing a control law that guides the robot to the desired position or along a desired path using solely the kinematic model. This method is universally applicable to any mobile robot, regardless of its configuration, as it only requires wheel speed measurement. For instance, in [15], a kinematic control for a mobile robot with a differential locomotion system is presented using Proportional-Integral-Derivative (PID) control, while in [16], Sliding-Mode control is utilized to enhance performance robustness. Regarding omnidirectional vehicles, [17], presents a kinematic control for a robot with four Omniwheels.

This paper details the design and construction of a mobile robot equipped with Mecanum wheels, along with a kinematic control implementation to solve the trajectory tracking control problem. This approach considers the robot's dynamic model as a black box, with only the input and output signals being well-known. Based on this, two levels of control are formulated —both are obtained from kinematic transformations— and are called high-level and low-level control. On one hand, high-level control refers to controlling the robot's position, parametrized in its Cartesian coordinates and its pose or orientation angle, which are assumed to be measurable. On the other hand, kinematics transformations can determine the angular velocity that the wheels require to reach a certain global velocity in the robot. In that sense, the low-level control is also formulated, which is associated with controlling the rotational speed of the wheels, which are commonly directly coupled to the shaft of a direct current motor; then, the rotational speed can be measured if the motor incorporates an encoder. In this work, the robot's position is estimated by odometry computations, and the sensitivity to errors due to the integration of velocity measurements is partially avoided by using the dSPACE ds1103 real-time data acquisition card while conducting short-time experiments and constantly resetting the integral.

The results obtained in this work are illustrated entirely experimentally on a conventional tiled floor, where the real-time data acquisition card dSPACE ds1103 is used for signal generation, data acquisition, and control algorithm execution. The experiments are discussed based on comparing the desired time-variant trajectories and the realized trajectories with different gains, each being these trajectories a boomerang-like trajectory and a clover-shaped trajectory. On the other hand, we establish the desired orientation angle as zero. Therefore, the robot's pose is assumed to be the same at the start and end of the trajectory. The procedure of this work is fully explained at a tutorial level. The results presented in subsequent sections demonstrate that kinematic control remains an economical and easy-to-implement alternative for managing mobile robots in indoor environments where navigation is sought. We address complex trajectories for tracking, and by tuning the PID controller, we obtain better reasonably good perfo

The paper's organization can be summarized as follows. Section 2 presents the design and construction of the mobile robot. In section 3, we present the kinematic model. The kinematic control is presented in Section 4. In Section 5 the experiments are conducted and illustred. Finally, the conclusions belong to Section 6.

2. Design and construction

2.1. Platform design

The Mecanum wheeled mobile robot consists of a rectangular metal platform mounted on four motors model 9.7:1 from Pololu, with transmission and gearbox with straight metal gears, operating with 12 V direct current (DC). Each motor is directly coupled to a 60 mm diameter Mecanum wheel, which provides omnidirectional locomotion to the system. We consider that all motors and wheels are completely identical. The platform was designed and machined using CNC. The contour of the robot, used to hold terminals, switches, and LED indicators, as well as to ensure a rigid chassis, was designed and manufactured using 3D printing. The first step consisted of assembling the following components: wheel, mounting hub, DC motor bracket, and DC motor. This is illustrated in Fig. 1. Next, the following step is to mount the wheels onto the rectangular platform. The remaining assembly was purely modular, relying on a screw-and-nut-based fastening system. Figure 2 illustrates the assembly drawing of the Mecanum-wheeled mobile robot, and the manual assembly is avaible via email-request. Table 1 compiles the main parameters of the geometry of the platform.

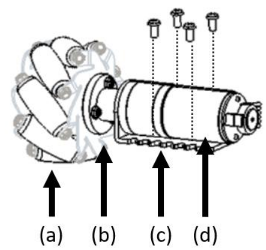

Fig. 1. Mecanum wheel (a), mounting hub (b), motor bracket (c), and direct current motor (d).

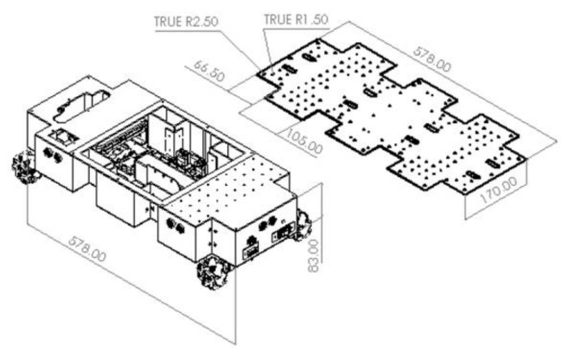

Fig. 2. Assembly during of the mobile robot. Isometric model (left) and platform dimension (right). The units given above are in milimeters.

Table 1. Main parameters of the mobile robot platform.

Parameter	Value
Length of the robot platform [mm]	578
Width of the robot platform [mm]	170
Number of Mecanum wheels equipped	4
Radius of Mecanum wheel [mm]	30
Mecanum wheel thickness [mm]	32
Angle of the passive rollers [degrees]	45
Width of the robot platform [mm]	45
Number of passive rollers per wheel	8
Distance l_x [mm]	255
Distance l_y [mm]	85

2.2. Electronic architecture

The electronic stage is crucial as it manages power distribution to the various components of the system. The mobile robot is equipped with a mounted four-cell Li-Po battery with a capacity of 15V and 6800mAh, which supplies power to the robot's components. Inside its chassis, there is a PCB board for managing the charging direction through a three-position switch, which includes the following states: 1) turning the robot ON, 2) turning the robot OFF, and 3) recharging the battery, allowing for recharging without removing it from the platform. See Fig. 3. A PCB voltage regulation circuit also provides outputs of 3.3V, 5V, 9V, 12V, and 15V to power components operating at different nominal voltages. The motor driver is the commercial L298N card, which controls both the direction and speed of each motor. Fig. 4 illustrates the CAD view of the robot with its main components.

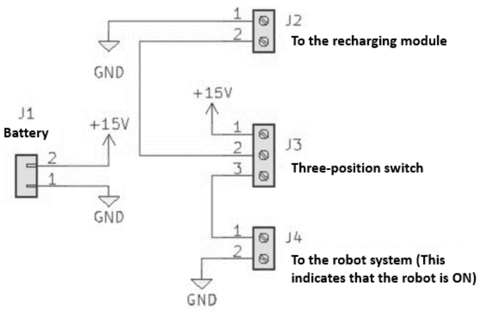

Fig. 3. Design of the three-position button for powering on, powering off, and charging the robot

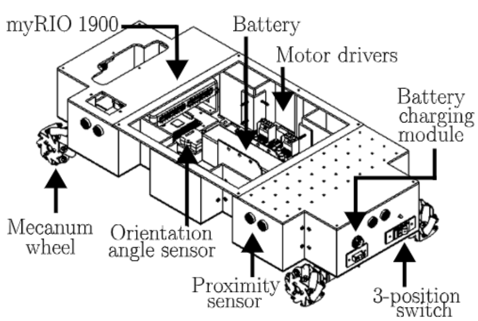

Fig. 4. Isometric CAD view of the main components of the mobile robot.

2.3. Data acquisition and control platforms

The vehicle features a two-level data acquisition and control architecture, explained as follows. The first level is associated with the measurement of the orientation sensor, using the myRIO 1900 card from National Instruments, programmed with LabVIEW 2014. The second level operates through the DSPACE-DS1103 platform and is responsable for measuring the rotational speed of each Mecanum wheel by measuring and characterizing the pulses from the encoder lectures to angular position and next to rotational speed, dividing its position by the sample time. Additionally, the DSPACE-DS1103 also executes the high and low-level control algorithms. The programming and data management were performed using Matlab and Simulink for programming, while ControlDesk for data management.

2.4. Signals measurement

2.4.1. Wheel rotational speed measurement

A two-channel Hall effect encoder, which is attached to the DC motor, is used to sense the rotation of a magnetic disk on a rear protrusion of the motor shaft. Then, after characterizing the sensor to convert the encoder counts into angular position [rad] using the dSPACE ds1103 platform, the angular speed [rad/s] was estimated by calculating the difference between the previous position and the current position, divided by the sampling time set in the real-time processor. This estimation is often called the M method, according to the authors [18], [19], and [20], which establish

$$\omega_s = \frac{3.1415\Delta_c}{E_c T_s}, \quad [1]$$

where ω_s is the angular speed estimation applied for each Mecanum wheel, the term Δ_c is the encoder delta position, $\Delta_c = current_{pos} - previous_{pos}$, E_c is the encoder resolution in terms of its counts per revolution, and T_s is the sampling time.

2.4.2. Mobile robot position measurement

For position measurement, the global velocities of the robot were numerically integrated using the dSPACE ds1103 platform, with Euler 1 selected as the ordinary differential equation (ODE) solver and a fixed step of 0.001 seconds. Since the experimental test was conducted in a completely indoor environment in a short-time exposition, and by constantly resetting the integral for each test, the well-known effect of noise was not presented. However, it is important to say that for a large-scale application and long-term operation exposure, a more accurate and reliable method for position measurement needs to be employed, such as multi-sensor fusion [21], IMU and Kalmar filter techniques [22], and the ultra-wideband (UWB) positioning system [23], among others [24]. Fig. 5 illustrates the integration method.

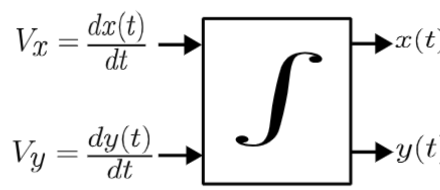

Fig. 5. Position estimation by numeric integration using real-time dSPACE ds1103, Euler as Ordinary Differential Equation (ODE) solver, and fixed step as 0.001 seconds.

3. Kinematic modeling

The kinematic model of mobile robots can be equivalently explained through forward and inverse kinematics. Forward kinematics aims to define the global velocity of the mobile robot's platform based on specific angular velocities of the robot's wheels. Conversely, inverse kinematics aims to define the angular velocities of each wheel based on specific global velocities of the mobile robot's platform. The kinematic model presented has been sufficiently explored by the authors [25] and [26] and is presented in

$$\begin{bmatrix}\dot{x}\\ \dot{y}\\ \theta_z\end{bmatrix}=\frac{r}{4}\begin{bmatrix}1 & 1 & 1 & 1\\ -1 & 1 & 1 & -1\\ \frac{-1}{l} & \frac{1}{l} & \frac{-1}{l} & \frac{1}{l}\end{bmatrix}\begin{bmatrix}\dot{\theta}_1\\ \dot{\theta}_2\\ \dot{\theta}_3\\ \dot{\theta}_4\end{bmatrix}, \quad [2]$$

where r is the radius of the Mecanum wheel given in [mm], and $l=(l_x+l_y)$ is the distance between the geometric centers of the wheels along the width l_x and length l_y [mm] of the platform, respectively. Similarly, the inverse kinematics is described by

$$\begin{bmatrix}\dot{\theta}_1\\ \dot{\theta}_2\\ \dot{\theta}_3\\ \dot{\theta}_4\end{bmatrix}=\frac{1}{r}\begin{bmatrix}1 & -1 & -l\\ 1 & 1 & l\\ 1 & 1 & -l\\ 1 & -1 & l\end{bmatrix}\begin{bmatrix}\dot{x}\\ \dot{y}\\ \theta_z\end{bmatrix}. \quad [3]$$

4. Kinematic control

Definition. Kinematic control deals with the planning and control of the movement of a mechanical system, such as a mobile robot, without considering the forces or dynamics that move [28]. Kinematic control is divided into high-level and low-level. In subsequent subsections, both are presented.

4.1. High-level control

Problem statement. Let's denote $q_d=[x_d \quad y_d \quad \theta_z]^T$ the desired position vector coordinates of the mobile robot, and $q=[x \quad y \quad \theta]^T$ the arbitrary initial position vector coordinates of the mobile robot. The problem statement of the high-level control aims to define a control law, denoted as V such that $\lim_{q\to\infty}|q_d-q\,|=0$. Then, the general error vector is defined as $e=[e_x \quad e_y \quad e_\theta]^T$, where $e_x=x_d-x$, $e_y=y_d-y$, and $e_\theta=\theta_d-\theta$. The goal of e is to converge it to zero, with a small enough bound.

Despite the advances in control theory, the PID controller and its variants are the most widely explored for practitioners. For this reason, a PID controller was employed.

$$V=K_pe+K_i\int e\ dt+K_d\dot{e}, \quad [4]$$

where $V\in R^{3}$ is the input vector of control position signals, $K_p, K_i, K_d\in R^{3x3}$ are gain matrices.

4.2. Low-level control

Problem statement. The high-level control law, which can be considered a force or torque acting over the system, can be kinematically transformed to represent torque acting on each wheel and generating motion in the robot platform. Then, by relating equation [3] with [4] we obtain

$$\begin{bmatrix}\dot{\theta}_{1d}\\ \dot{\theta}_{2d}\\ \dot{\theta}_{3d}\\ \dot{\theta}_{4d}\end{bmatrix}=\frac{1}{r}\begin{bmatrix}1 & -1 & -l\\ 1 & 1 & l\\ 1 & 1 & -l\\ 1 & -1 & l\end{bmatrix}\begin{bmatrix}V_x\\ V_y\\ V_\theta\end{bmatrix}. \quad [5]$$

This implies assuming that the high-level control input is the mobile robot velocity. Then, trough computing equation [5], the terms $\dot{\theta}_{1d}-\dot{\theta}_{4d}$ are now a set of rotational speed **desired** to each Mecanum wheel. Thus, a simple P controller is developed through a control law u

$$u=K_{p\omega}e_v, \quad [6]$$

where $K_{p\omega}$ is the proportional gain, e_v is the low-level error signal given by $e_v=\dot{\theta}_d-\dot{\theta}$. Please note that the subscript i is not expressed to avoid abuse of notation. However, the low-level control shown in equation [6] is applied for each wheel.

5. Real-world experiments

Two experiments were conducted to illustrate the results, highlighting the omnidirectional advantage of the vehicle. The reference signals chosen were rare and complex shapes to follow in vehicles with holonomic constraints. The first experiment was a boomerang-like-shape trajectory composed by the following set of desired time-variant references $x_d(t)=Acos(2\pi ft)+Acos(2\pi ft)$ and $y_d(t)=Asin(2\pi ft)+Asin(2\pi ft)$. The second experiment involved forming a clover-shaped figure on the surface, which is obtained by the following pair of equations $x_d(t)=Asin(3\pi ft)cos(\pi ft)$ and $y_d(t)=Asin(3\pi ft)sin(\pi ft)$. The values for the first experiment were $f=0.1$ and $A=0.25$. The duration of both experiments was 20 seconds. Fig. 6 shows the real robot used for conduct this experiment.

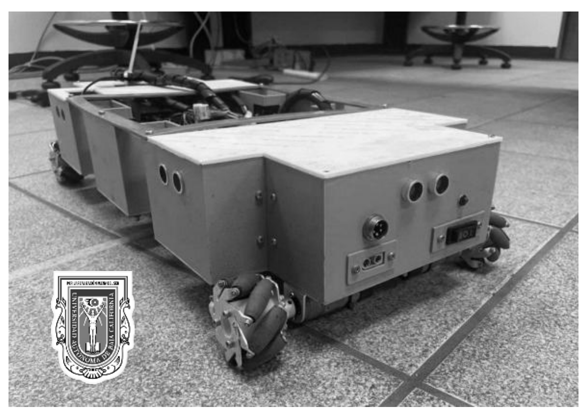

Fig. 6. Real robot constructed for experimentation.

Regarding the first experiment, we repeated the same test twice, where the control gains were tuned through trial and error until a more effective performance in trajectory tracking was observed. In the first trial, observed in Fig. 7 the gains were set to $K_p=4$, $K_i=3$, and $K_d=0.5$, resulting in a noticeable overshoot in the system response. In the second trial, shown in Fig. 8, the gains were adjusted to $K\_p = 3$, $K_i = 1.15$, and $K_d = 0.6$, leading to better vehicle performance than the first trial. The error signals history are observed in Fig. 8 and illustrates the comparison error in terms of x and y axis for both experiments, in which it is noticeable that the error signal remains around a bound close to zero.

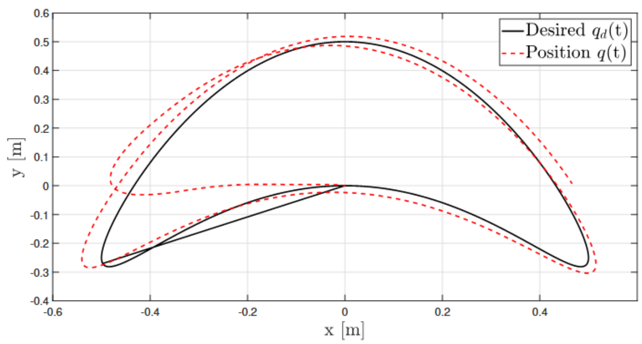

Fig. 7. First test for boomerang-like-shape trajectory tracking, with $K_p = 4, K_i = 3, K_d = 0.5, K_{p\omega} = 3$.

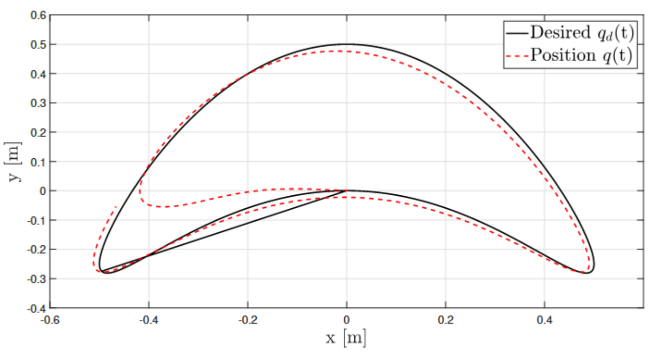

Fig. 8. Second test for boomerang-like shape trajectory tracking, with $K_p = 3, K_i = 1.15, K_d = 0.6, K_{p\omega} = 3$.

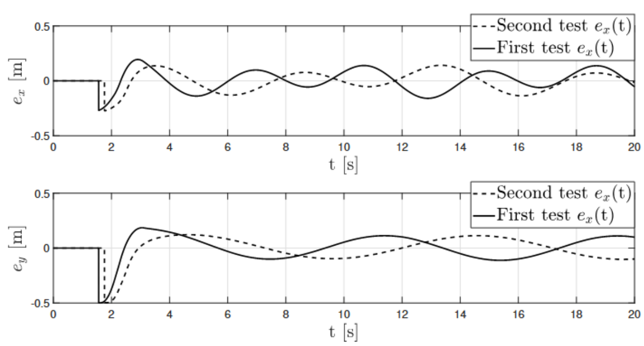

Fig. 9. Error signals history comparisons for both tests in the boomerang-like shape trajectory tracking

Regarding the second experiment, the parameter values for the reference signal were $f = 0.05$ and $A = 4$. The results are shown in Fig. 10 and Fig. 11. For the first and second tests conducted, respectively. During the first test, the system exhibited considerable overshoot due to the high computed gains in the PID controller. For that purpose, in the second test, the gains K_p, K_i and K_d were retuned by trial and error. The results illustrate that tuning is still a key factor in achieving good performance in robotics position control.

Concerning the error signals, Fig. 12 illustrates the comparison for x and y axis. The error signal remains around a bound close to zero for both experiments. However, due to the complexity of the shape trajectory tracking, oscillations are presented.

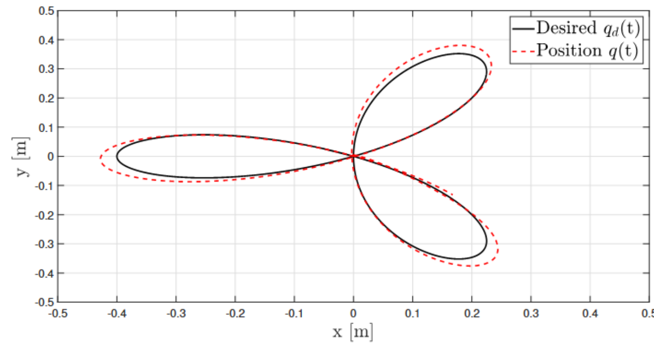

Fig. 10. First test for clover-like shape trajectory tracking, with $K_p = 4.25, K_i = 2.250, K_d = 0.5, K_{p\omega} = 3$.

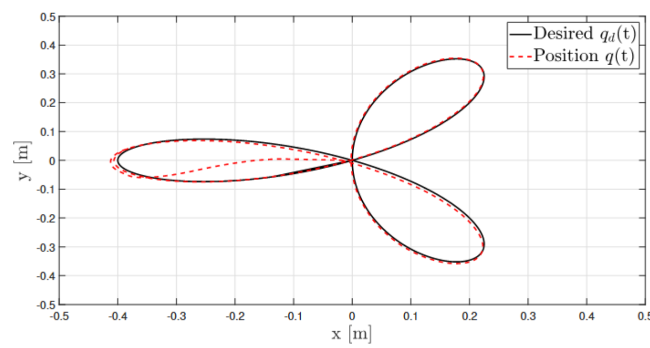

Fig. 11. Second test for clover-like shape trajectory tracking, with $K_p = 8, K_i = 1.75,\ K_d = 0.5, K_{p\omega} = 1.5$.

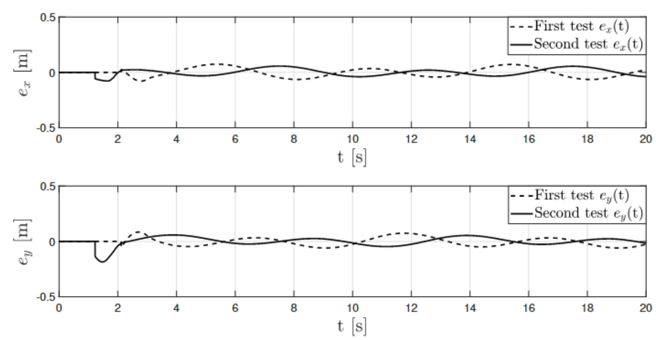

Fig. 12. Error signals history comparisons for both tests in the clover-like shape trajectory tracking

5. Conclusion

The design, construction, and control of a mobile robot with omnidirectional locomotion provided by the Mecanum wheel were presented. An approach based on kinematic control was formulated and implemented, where only the measurement of wheel velocities is considered, and the vehicle's position is estimated through kinematics and odometry computations. It was experimentally demonstrated that proper gain tuning was sufficient to address the tracking of complex trajectories. In this case, boomerang and clover-shaped figures were used to exemplify an unconventional trajectory while highlighting the vehicle's omnidirectional advantage and construction. One of the advantages of implementing kinematic control is its conceptual simplicity and ease of implementation, making it a fundamental technique for advanced autonomous navigation tasks. However, robust controller techniques may be required to correct steady-state errors and mitigate the effects of external disturbances that may arise on the surface due to irregularities. In contrast, a dynamic model contributes greater precision and robustness to the system.

4. References

[1] Rubio, F., Valero, F., and Llopis-Albert, C. (2019). A review of mobile robots: Concepts, methods, theoretical framework, and applications. International Journal of Advanced Robotic Systems, 16(2), 1729881419839596.

[2] Gon¸calves, R.S. and Carvalho, J.C.M. (2013). Review and latest trends in mobile robots used on power transmission lines. International Journal of Advanced Robotic Systems, 10(12), 408.

[3] lon, B. (1973). Directionally stable self propelled vehicle. US Patent 3,746,112

[5] Gfrerrer, A. (2008). Geometry and kinematics of the mecanum wheel. Computer Aided Geometric Design, 25(9), 784–791.

[6] Dhengre, N., Mogra, A., Verma, S., and Gupta, A. (2018). Design and manufacturing of mecanum wheel for omnidirectional robot. International Journal of Mechanical and Production Engineering Research and Development, 2018.

[7] Shabalina, K., Sagitov, A., and Magid, E. (2018). Com-parative analysis of mobile robot wheels design. In 2018 11th International Conference on Developments in esystems Engineering (dese), 175–179. IEEE.

[8] Tatar, M.O., Popovici, C., Mandru, D., Ardelean, I., and Plesa, A. (2014). Design and development of an autonomous omni-directional mobile robot with Mecanum wheels. In 2014 IEEE International Conference on Automation, Quality and Testing, Robotics, 1–6. IEEE.

[9] Akkad, H. and Ph D, H. (2023). A comparative review of omnidirectional wheel types for mobile robotics.

[10] Najim, H.A., Kareem, I.S., and Abdul-Lateef, W.E. (2023). Design and implementation of an omnidirectional mobile robot for medicine delivery in hospitals during the covid-19 epidemic. In AIP Conference Proceedings, volume 2830. AIP Publishin

[11] Thongpance, N. and Chotikunnan, P. (2023). Design and construction of electric wheelchair with mecanum wheel. Journal of Robotics and Control (JRC), 4(1), 71–82.

[12] Chotikunnan, R., Chotikunnan, P., Thongpance, N., Put-tasakul, T., Pititheeraphab, Y., and Sangworasil, M. (2023). Application of pid control system in Mecanum wheelchair. International Journal, 10(3), 3519–3529.

[13] Chen, Z.Y., Liaw, P.R., Nguyen, V.L., and Lin, P.T. (2021). Design of a high-payload mecanum-wheel ground vehicle (mwgv). Robotic Systems and Applications, 1(1), 24 34.

[14] Hernández, J. C. O., & Almeida, D. I. R. (2024). Kinematic control in a four-wheeled Mecanum mobile robot for trajectory tracking. The Journal of Engineering, 2024(9), e70006.

[15] González, L.H.R., Valencia, J.A., and Montoya, A. (2009). Modelo cinem´atico de un robot m´ovil tipo diferencia y navegación a partir de la estimación odométrica. Scientia et technica, 1(41), 191–196.

[16] Proa˜no, P., Capito, L., Rosales, A., and Camacho, O. (2015). Sliding mode control: Implementation like pid for trajectory-tracking for mobile robots. In 2015 Asia Pacific Conference on Computer Aided System Engineering, 220–225. IEEE

[17] Sofwan, A., Mulyana, H.R., Afrisal, H., and Goni, A. (2019). Development of omni-wheeled mobile robot based-on inverse kinematics and odometry. In 2019 6th International Conference on Information Technology, Computer and Electrical Engineering (ICITACEE), 1–6. IEEE.

[18] Wang, H. and Pu, J.t. (2013). An improved variable m/t method based on speed estimation for optical incremental encoders. In Proceedings of 2012 3rd International Asia Conference on Industrial Engineering and Management Innovation (IEMI2012), 13–20. Springer.

[19] Pu, J.t. and Wang, H. (2012). A novel variable m/t method for speed measurement with high precision in a wide speed range. In 2nd International Conference on Electronic & Mechanical Engineering and Information Technology, 1855–1858. Atlantis Press.

[20] Xu, J., Yin, X., Cui, X., and Liu, Z. (2017). A realization method of rotating speed measurement based on different m/t. In AIP Conference Proceedings, volume 1829. AIP Publishing.

[21] Dobrev, Y., Flores, S., and Vossiek, M. (2016). Multimodal sensor fusion for indoor mobile robot pose estimation. In 2016 IEEE/ION Position, Location and Navigation Symposium (PLANS), 553–556. IEEE.

[22] Alatise, M.B. and Hancke, G.P. (2017). Pose estimation of a mobile robot based on fusion of imu data and visión data using an extended kalman filter. Sensors, 17(10), 2164.

[23] García, E., Poudereux, P., Hernández, Á., Ureña, J., and Gualda, D. (2015). A robust uwb indoor positioning system for highly complex environments. In 2015 IEEE International Conference on Industrial Technology (ICIT), 3386–3391. IEEE.

[24] Skrzypczyínski, P. (2017). Mobile robot localization: Where we are and what are the challenges? Automation 2017: Innovations in Automation, Robotics and Measurement Techniques 1, 249–267.

[25] Taheri, H., Qiao, B., and Ghaeminezhad, N. (2015). Kine-matic model of a four mecanum wheeled mobile robot. International journal of computer applications, 113(3), 6–9.

[26] Alhanov, D.S. and Rubtsov, V.I. (2020). Development of the laboratory work:"modeling of a mobile robot on mecanum wheels kinematics". In ITM Web of Conferences, volume 35, 04001. EDP Sciences.

[27] Siegwart, R., Nourbakhsh, I.R., and Scaramuzza, D. -- (2011). Introduction to autonomous mobile robots. MIT Press