Design and Control for Trajectory Tracking of Three-Wheeled Mecanum Omnidirectional Kiwi-Robot

Diseño y Control para el Seguimiento de Trayectorias de un Robot Kiwi Omnidireccional de Tres Ruedas Mecanum

Angel L. Aguirre Alatorre [1], Horacio Rostro González[2,3], Higinio Juárez Rios[4], Marcelino Carrera Rodríguez[4], Rodrigo Delgadillo Gaytán[1], Miroslava Cano Lara*[1].

[1]Tecnológico Nacional de México/ITESI

[2]Universidad Ramon Llull/School of Engineering, GEPI

[3]Universidad de Guanajuato/DICIS

[4]Instituto Politécnico Nacional/UPIIG

LIS23110297@irapuato.tecnm.mx

miroslava.cl@irapuato.tecnm.mx

Abstract

This work presents the design and construction of a three-wheeled mecanum omnidirectional mobile robot. The structural components, including the wheels, were modeled in SolidWorks and fabricated using 3D printing technology. A hexagonal chassis configuration was implemented to ensure structural stability and optimal weight distribution. Regarding the electronics, a custom circuit was developed for the independent control of three DC motors. The software architecture comprises tailored algorithms designed for specific operational tasks, featuring a core program capable of executing eight distinct omnidirectional movement actions, detailed through a block diagram and pseudocode. The overall stability of the kiwi-robot is highly dependent on the real-time coordination of operational speeds, planned trajectories, and orientation angles. Additionally, proper motor torque management, coupled with digital signal conditioning, is essential to achieve a robust and stable control system. Finally, the robot was programmed to track complex trajectories guided by mathematical formulations. Consequently, geometric figures such as a circle, an astroid, and a Bernoulli lemniscate were successfully generated. The experimental results and continuous-path graphs demonstrate a high-performance evaluation of the integrated system.

Keywords: Omnidirectional robot, Mecanum wheels, 3D printing, Control.

Introduction

Omnidirectional mobile robots have a holonomic locomotion system. This means that they can move in any direction on a plane without first turning the body. There are different kinematic configurations, but the most studied are four-wheel and three-wheel systems (Barrientos *et al*., 2007). Three-wheeled Kiwi-robots, have isostatic contact that gives good balance on the floor. They also need fewer motors, which can reduce energy and computing use (Ollero Baturone, 2001). Because they can move easily, these systems have useful applications in service robotics, industrial automation, and movement in small spaces (Ollero Baturone, 2001; Siegwart *et al*., 2011). To obtain this free movement, different special wheels can be used, the most common are normal omnidirectional wheels and Mecanum wheels. Their passive rollers are placed around the wheel and add more freedom to the system. When torque is applied, the rollers help the Kiwi-robot keep the correct grip and direction by reducing friction on the transverse axis. The three wheels must be placed with a separation angle of 120° between them. This gives a symmetric kinematic map for the independent control of each actuator (Barrientos *et al*., 2007; Siegwart *et al*., 2011).

Position control of the Kiwi robot starts from a reference point in an XY plane and reaches a desired orientation (García & Kelly, 2009). The robot starts at rest and moves to a target point while changing its angle. Then, it can stop at a new orientation, for example 160°, as shown in Figure 1.

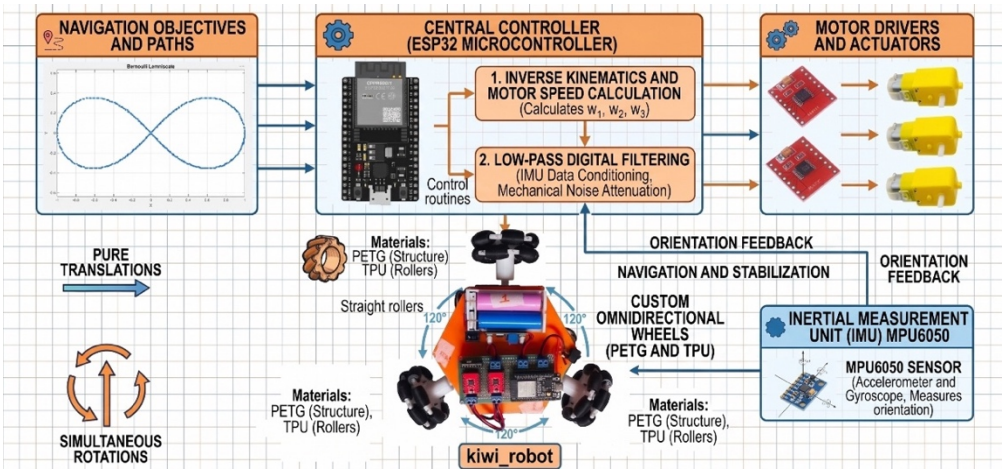


Figure 1. Technical scheme of centralized control for a three-wheeled Mecanum robot.
Source: Own work.

Trajectory tracking control makes the Kiwi-robot follow a curve that is described by time (García & Kelly, 2009). The robot keeps the desired position during the path and tries to keep the error very close to zero, as shown in Figure 2. An example is a continuous circular path, where the Kiwi-robot must keep smooth wheel motion to complete the geometric trajectory, as shown in Figure 3.

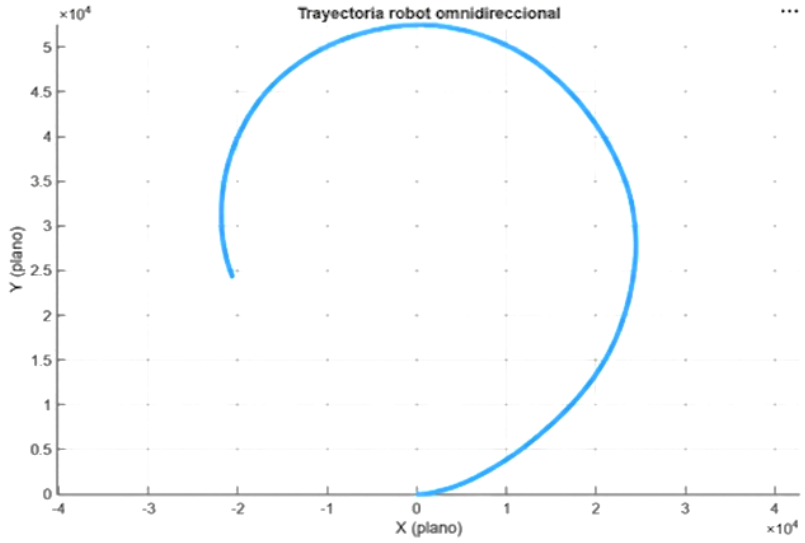


Figure 2. Curve parametrized with time
Source: Own work.

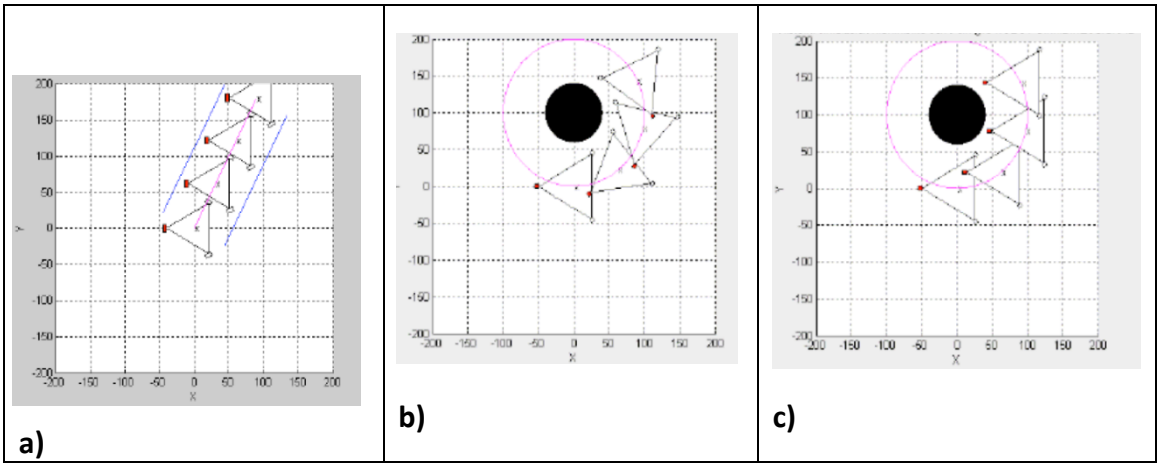


Figure 3. a) Example of linear motion, b) Linear displacement with orientation variation, c) Rotation of the robot around a point without changes in its orientation.
Source: Alfredo J, Dávila Aarón

This work starts with CAD design and 3D printing of the structural parts, from the hexagonal base to the wheels. The next step is the electronic stage, where the three motors are controlled independently. After that, eight basic movement actions are programmed. These actions include the activation of one motor to create turns and different motor combinations to create linear movement. Finally, mathematical formulas are used to draw a circle, an astroid, and a Bernoulli lemniscate. In this way, the precision and continuous trajectory tracking ability of the Kiwi robot are evaluated.

1. Theoretical Framework

1.1 Mathematical trajectory generation

For trajectory tracking in omnidirectional vehicles, the literature uses parametric equations (Maldonado Peña *et al.*, 2025; MathWorks, 2023). These equations describe the position in a two-dimensional plane with x and y coordinates, which depend on a continuous variable, usually time t. By dividing time into small steps, a continuous path becomes a sequence of ordered points. Using inverse kinematics, the robot controller changes these time-dependent coordinates, $x(t)$ and $y(t)$, into individual speed commands for each actuator (García & Kelly, 2009).

In classical analytic geometry, there are several well-known parametric curves that can be used to model trajectories:

Círcle. It describes a closed path with a constant radius r.

$$\boldsymbol{x(t) = r\cos(t)} \quad (1)$$

$$\boldsymbol{y(t) = r\sin(t)} \quad (2)$$

Astroid. It represents a hypocycloid with four cusps, where a is the maximum amplitude from the origin.

$$\boldsymbol{x(t) = a\cos^3(t)} \quad (3)$$

$$\boldsymbol{y(t) = a\,sin^3(t)} \quad (4)$$

Bernoulli lemniscate. It models a continuous figure-eight path, with a geometric amplitude defined by parameter a.

$$\boldsymbol{x(t) = \frac{a\cos(t)}{1+\sin^2(t)}}\ ,\ \boldsymbol{y(t) = \frac{a\sin(t)\cos(t)}{1+\sin^2(t)}} \quad (5)$$

1.2 Digital signal filtering

During motor operation in a Kiwi-robot, the continuous contact of the passive rollers with the floor creates high-frequency mechanical vibration. These vibrations produce important noise in inertial sensor (IMU) readings. To reduce this effect and obtain a more precise orientation estimate, signal conditioning with digital filters is recommended, usually low-pass filters (Barrientos *et al.*, 2007). After the signal is filtered, the clean data are sent back to a closed-loop control system. The controller calculates the kinematic error between the desired orientation angle and the measured angle. Then, it can change the motor power signals to keep the direction of the omnidirectional platform.

1.3 Kinematic interpretation of trajectories

In trajectory movement, parametric equations do more than draw a shape in space and describe vehicle behavior over time. To change a mathematical equation into real Kiwi-robot motion, the model must be interpreted with inverse kinematics. The variable t works like a clock or phase step. At each instant, the equation gives a desired coordinate (x, y). The controller compares this point with the current position. Using a geometric transformation matrix at 120°, the system divides the movement vector into three different angular speeds $(\omega 1, \omega 2, \omega 3)$, one for each wheel. When these three speeds work at the same time, the resulting forces move the Kiwi-robot toward the calculated coordinate and follow the geometry given by the equation.

2. Methodology

This section describes the complete process used for the Kiwi-robot, from mechanical design to the movement control algorithm. The process starts with CAD design, additive manufacturing of the parts, mechanical assembly, electronic architecture, and finally programming of the kinematic sequences.

2.1 CAD design

The robot was designed in SolidWorks ® with a hexagonal structural base that distributes the weight and allows the actuators to be aligned at 120°. The Mecanum wheel components were also obtained and modified, the central rims were modeled, and the roller profile was checked to improve the system kinematics.

2.2 Additive manufacturing

The mechanical design, modeled in SolidWorks, features a regular hexagonal chassis for uniform weight distribution and 120° symmetrical coupling of three motors, alongside custom 90° omnidirectional wheel components. Components were exported as +.STL files and fabricated via FDM 3D printing on a Creality K1. Structural parts, including the hexagonal base and wheel supports, were printed in PETG for optimal rigidity and durability, whereas the passive wheel rollers utilized flexible TPU to absorb mechanical vibrations and ensure optimal traction.

2.3 Omnidirectional Mecanum Wheels

Standard omnidirectional wheels use passive rollers distributed around the wheel. The roller rotation axis is oriented at 90° to the main wheel traction axis, while Mecanum wheel rollers are inclined at 45° (Siegwart *et al.*, 2011).

The main effect of this design is the reduction of lateral friction. When the motor applies torque, the wheel produces a main traction force in the direction of rotation. At the same time, the passive rollers can turn freely, allowing the wheel to slide with low resistance on its axis. The advantage of using three of these wheels in a symmetric 120° configuration is that each actuator can be controlled independently in speed and direction. The system can then add the resulting forces in a coordinated way. In this manner, the Kiwi-robot can move

in any Cartesian direction, make diagonal movements, and rotate around its central axis without previous orientation maneuvers, as shown in Figure 4.

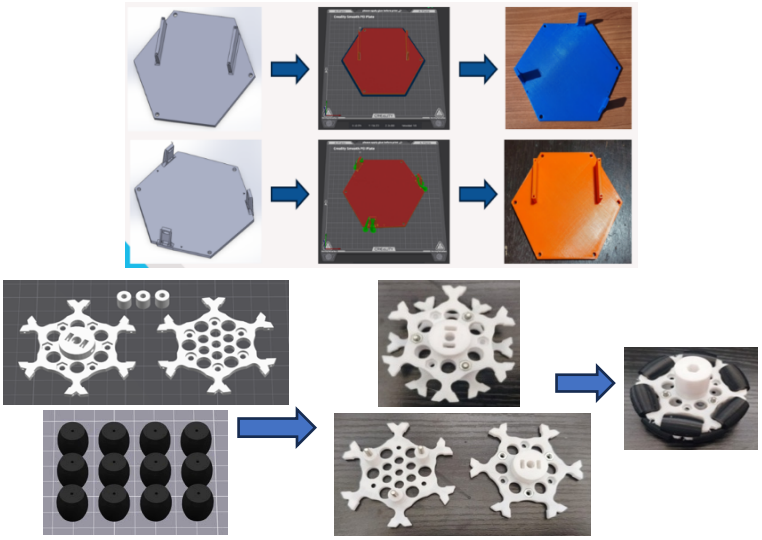


Figure 4. Design and additive manufacturing of the structural base and traditional omnidirectional wheels.
Source: Own work.

2.4 Electronics Stage

The electronic architecture is centered on ESP32. It processes inverse kinematics and sends pulse-width modulation (PWM) and direction signals to the actuator stage. Three 48:1 DC motors were selected for traction, with a nominal voltage of 5 V and a no-load speed of 200 rpm (Rashid, 2015). Two power driver modules based on the DRV8833 integrated circuit are used to drive the motors. To make the motor response smoother during sudden trajectory changes (InvenSense Inc., 2013), an electronic solution was implemented. Acceleration and deceleration ramps are generated by changing the PWM duty cycle from the ESP32, and this limit current peaks caused by motor torque and helps avoid unwanted sliding of the 90° wheel rollers. The detailed wiring for this architecture is shown in Figure 5.

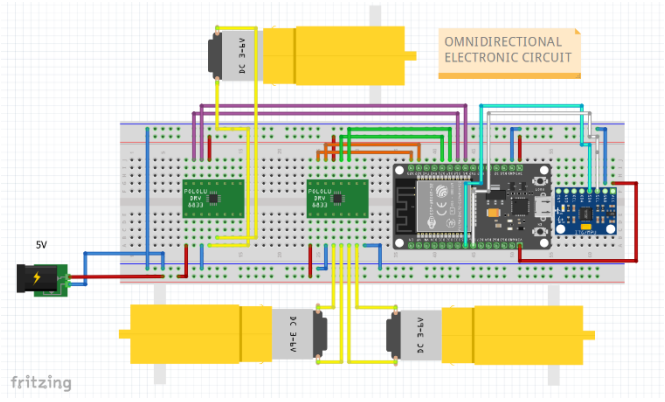


Figure 5. Electronic diagram required for the omnidirectional Kiwi robot.
Source: Own work.

2.5 Programming

The Kiwi-robot navigation algorithm features eight basic movement actions. First, individual control states are programmed for each actuator, allowing the system to set forward, reverse, or stop. From the point of view of robot kinematics, this individual control is important because it shows the basic traction vector of the system. After that, the code combines the individual states to add or cancel forces and create the required rotations and translations. The activation logic is summarized in the pseudocode in Table 1.

***Table 1**. Motor activation pseudocode*

Step	Action	Code Logic
1	Start	Read the desired movement instructions
2	Condition	Which of the 8 available actions was selected?
3	Execution	If Action 1: Motor 1 forward, Motor 2 off, Motor 3 off.
4	Execution	If Action N: Set the direction (forward) for motors 1, 2, and 3 as required.
5	End	Keep the motors running until the stop command is received.

To create more complex movement, three geometric figures are used: a circle (Eqs. 1 and 2), an astroid (Eqs. 3 and 4), and a Bernoulli lemniscate (Eq. 5). Each formula is changed into programming code so that the motors move only as much as needed to draw the complete figure.

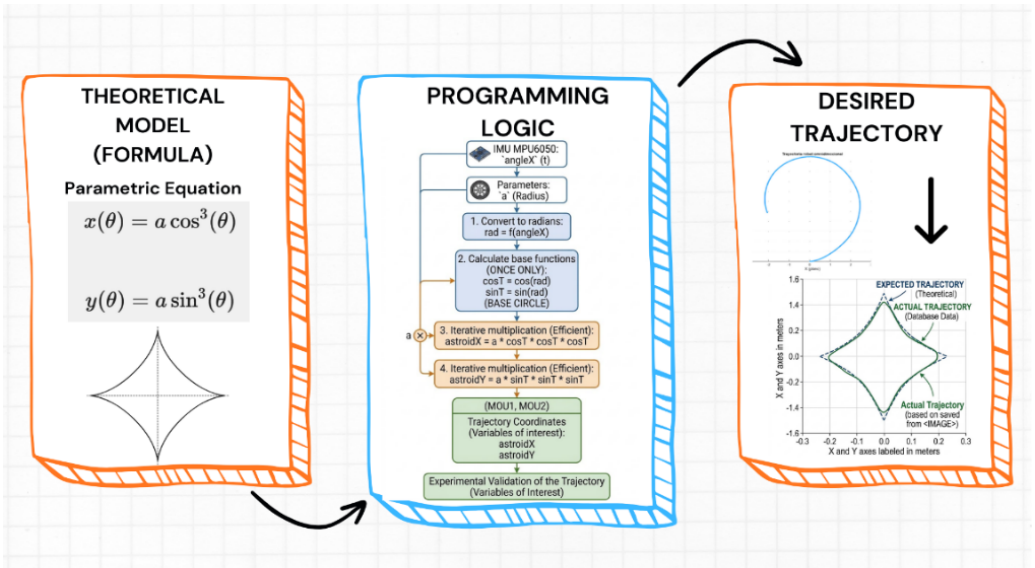


Figure 6. Graphic explanation of the programming
Source: Own work

Prior to continuous maneuver evaluation, an isolated hardware validation stage assessed each omnidirectional wheel independently. This preliminary test verified homogeneous dynamic responses and actuator integrity, ensuring uniform traction across all channels. Subsequently, eight discrete locomotion sequences were executed (Figure 6). The first seven sequences comprised distinct translations and rotations, whereas the eighth evaluated pure rotation around the central axis. In robotic systems, these discrete states serve as a primary validation phase to confirm that the $120°$ kinematic mapping and force vector combinations

effectively minimize roller lateral friction. Inertial data were synchronously acquired via an IMU during the first seven sequences. The chronological synthesis of these isolated displacements yielded the cumulative system behavior and the resulting continuous trajectory illustrated in Figure 7.

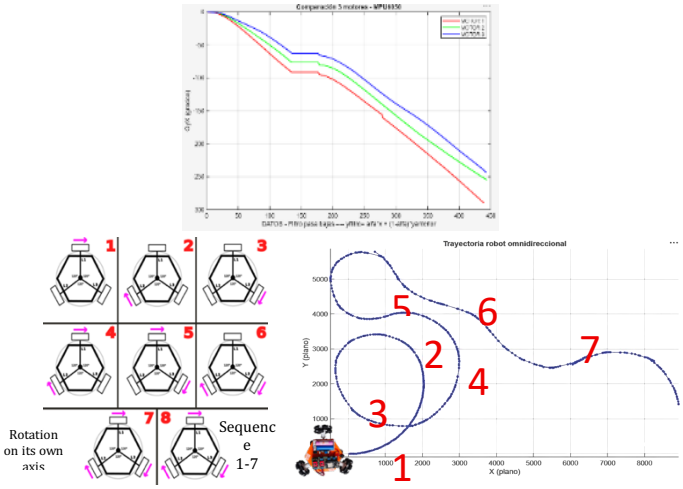


Figure 7. Motor condition check, trajectory, and movement sequences of the omnidirectional mobile robot. Source: Own work.

2.6 Real-time parametric trajectory

To bridge parametric modeling and physical locomotion, the primary control algorithm leverages the circle parametric equation as its mathematical foundation, utilizing the fundamental trigonometric oscillations, . Mapping these functions generates the primary state vectors, which are analytically scaled to derive the spatial coordinates for the astroid and Bernoulli lemniscate trajectories. Within the real-time execution environment, the parameter t. serves as the temporal variable incremented at each control cycle to continuously solve for the target coordinates

Trajectory tracking is achieved via inverse kinematics and deterministic synchronization; the microcontroller maps these spatial coordinates into discrete angular velocity references for each wheel. To ensure geometric smoothness, the algorithm couples these velocities and concurrently dispatches synchronized PWM signals to the three actuators.

To keep the robot stable under the mechanical disturbances created by the wheel rollers, a signal conditioning stage was implemented in the ESP32. The raw acceleration and angular velocity values from the MPU6050 sensor were processed with a digital low-pass filter programmed in C++. This conditioning reduced noise and provided a stable and reliable orientation angle.

2.7 Trajectory Execution

The system worked with discrete control cycles. In each time step, the algorithm continuously processed the navigation variables: the current orientation angle of the Kiwi robot ($\theta actual$) measured by the IMU, the desired spatial coordinates $x(t)\ and\ y(t)$ from the mathematical model, and the angular speed required for each independent wheel ($\omega 1, \omega 2, \omega 3$).

From these variables, parametric trajectories were generated to create three physical figures: a circle, an astroid, and a Bernoulli lemniscate. During the paths, the robot state information was recorded and stored in a database. This structured data was used later to build the kinematic validation graphs.

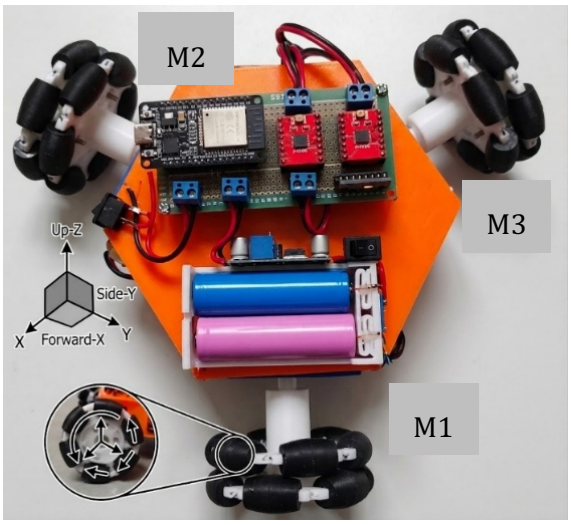
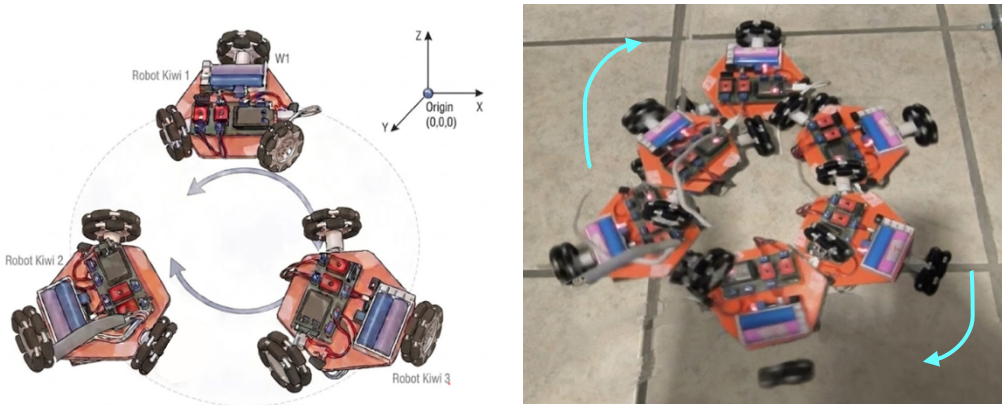


Figure 8. Mechatronic Omnidirectional Kiwi-robotic.

Source: Own work.

3. Results

As the first experimental validation, the physical behavior of the system during movement was analyzed. Figure 9 shows a chronological sequence of video frames from the Kiwi-robot during a trajectory. The traction vectors of each motor and the activation angles at the captured moment were added to the images. This shows how the sum of the forces produces the desired displacement without losing chassis orientation. Later, using the database generated during the control cycles, graphs of expected trajectories and real trajectories were built. These graphs showed good agreement with the continuous movement and with the circular geometric shape physically drawn by the Kiwi-robot.



a) b)

Figure 9. a) Theoretical representation. b) Experimental results of the circular trajectory.
Source: Own work.

3.1 Circle

To draw the circle, motors M1 and M2 of the Kiwi robot work at the same time with a constant speed profile. This needs only small changes in the wheel traction angles and gives continuous and smooth movement. The inertial data showed mostly constant acceleration, which is related to a stable speed profile. The control system needed only small adjustments in the traction angles, achieving continuous and smooth movement around the complete circle, as shown in Figure 10.

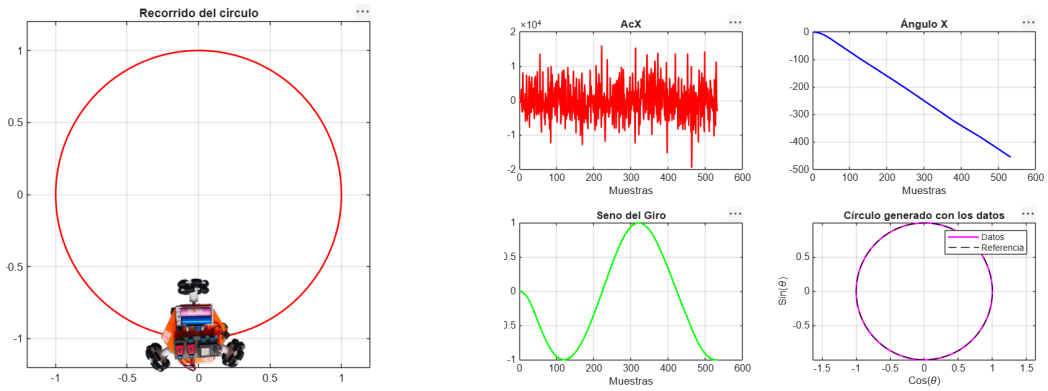


Figure 10. a) Theoretical representation of the circular trajectory (x and y coordinates). b) Experimental motion capture tracking the x and y spatial position.
Source: Own work.

3.2 Astroid

For this figure, the three motors have short reductions in speed when the robot approaches the sharp vertices of the path. Immediately after this, the actuators apply a fast-compensating acceleration to change direction. Compared with the circle, the astroid is more difficult from the kinematic point of view. The acceleration graph shows short speed reductions near the vertices. After these reductions, the controller applies a fast-compensating acceleration and corrects the orientation angles to continue toward the next vertex, as shown in Figure 11.

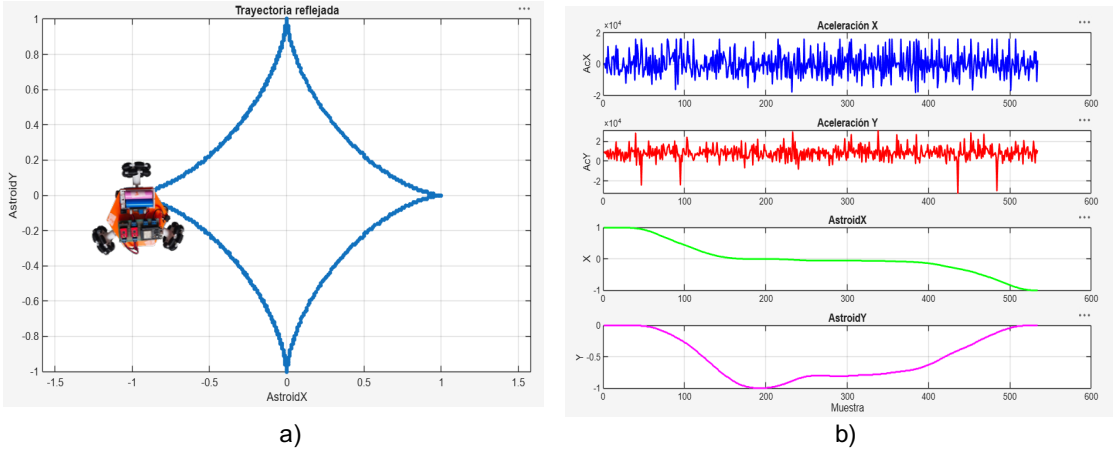
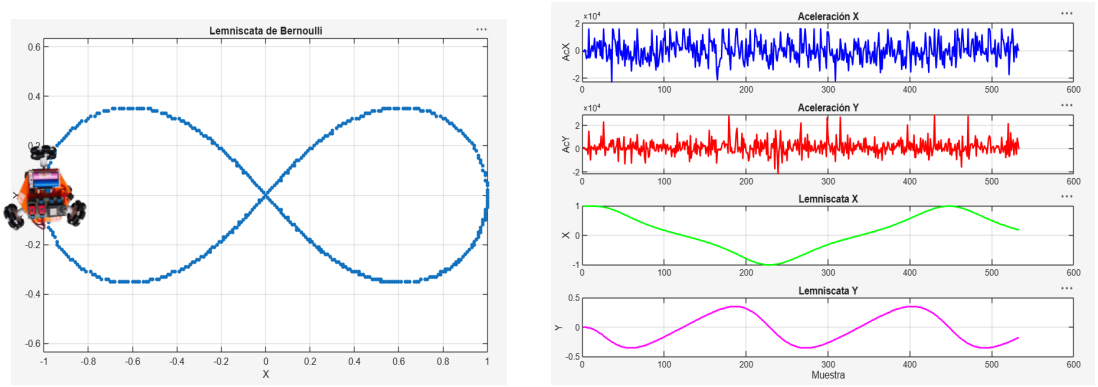


Figure 11. Astroid trajectory tracking in the XY plane. a) Theoretical representation of position coordinates $x(t)$ and $y(t)$. b) Experimental motion capture results plotted as x vs. y.
Source: Own work

3.3 Bernoulli lemniscate

For this complex trajectory, the acceleration analysis showed an adaptive behavior. The robot kept a stable speed in the outer loops, but the large changes in direction angles, especially at the center crossing, required dynamic speed adjustments. The system detected these changes and applied precise acceleration corrections to stabilize the movement and complete the path in a smooth way. For this trajectory, the controller requires the three Kiwi robot motors to work with asynchronous angular speeds $(\omega 1, \omega 2, \omega 3)$, producing fast changes in rotation direction. Because of this, the force angles of each wheel are changed continuously at the crossing point. The needed vectors are then added to continue the movement, as shown in Figure 12.



a) b)

Figure 12. Bernoulli lemniscate trajectory tracking in the XY plane. a) Theoretical representation of position coordinates $x(t)$ and $y(t)$. b) Experimental motion capture results plotted as x vs. y.
Source: Own work

4. Conclusions

This work successfully validated the structural synthesis, electronic integration, and kinematic control of a three-wheeled mecanum omnidirectional mobile robot (Kiwi-robot) optimized for complex trajectory tracking. The implementation of a 3D-printed hexagonal chassis provided high structural rigidity and an optimal center of mass, which counteracted asymmetric loads and minimized inertial disturbances during multidirectional displacements.

From a control perspective, the deterministic synchronization of the three independent DC actuators proved highly effective. The control architecture successfully modulated individual motor speeds and distributed required torques dynamically to execute the eight fundamental locomotion sequences without modern slip or saturation. During complex trajectory tracing, the kinematic mapping accurately translated the mathematical formulations of the circle, astroid, and Bernoulli lemniscate into coordinated angular velocities. The experimental results demonstrated that real-time tracking of these highly non-linear geometries requires strict velocity coupling to handle rapid rotation and force-vector shifts at the roller interfaces. Ultimately, the continuous-path performance graphs confirm high system stability, minimal cumulative error, and robust trajectory accuracy, validating the integrated robotic platform as a high-performance system for omnidirectional automation.

5. Acknowledgment

This work was supported by the project SIRA-PVR-2026-25454 of National Technological Institute of Mexico.

References

Barrientos, A., Peñín, L. F., Balaguer, C., & Aracil, R. (2007). *Fundamentos de robótica (*2.ª ed.). McGraw-Hill Interamericana.

García, M. A., & Kelly, R. (2009). Control cinemático de una base móvil omnidireccional. *Revista Iberoamericana de Automática e Informática Industrial RIAI,* 6(4), 44–53.

Davila. A. (2007). MODELO CINEMATICO DE UN ROBOT ONMIDIRECCIONAL. Universidad de los Andes

InvenSense Inc. (2013). MPU-6000 and MPU-6050 product specification revision 3.4.

Maldonado Peña, A. A., de la Cruz Bautista, E. D., Maya Gress, K. F., Samperio Vázquez, O., & González Islas, J. C. (2025). Diseño y validación experimental de un robot omnidireccional de tres ruedas con control PID sobre ESP32. Boletín Científico INVESTIGIUM de la Escuela Superior de Tizayuca, 11(Especial), 187–200. https://doi.org/10.29057/esti.v11iEspecial.15995

MathWorks. (2023). System Identification Toolbox: User's guide. *The MathWorks, Inc.*

Ollero Baturone, A. (2001). *Robótica: Manipuladores y robots móviles.* Marcombo.

Proakis, J. G., & Manolakis, D. K. (2007). *Tratamiento digital de señales: Principios, algoritmos y aplicaciones (4.ª ed.*). Prentice Hall.

Rashid, M. H. (2015). *Electrónica de potencia: Circuitos, dispositivos y aplicaciones (4.ª ed.).* Pearson Educación.

Siegwart, R., Nourbakhsh, I. R., & Scaramuzza, D. (2011). *Introduction to autonomous mobile robots (2nd ed.).* MIT Press.