International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Modeling, Simulation and Control of An Omnidirectional Mobile Manipulator Robot

Modelado, Simulación Y Control De Un Robot Manipulador Móvil Omnidireccional

María José Mendoza Salazar[1], Martha Ximena Dávalos Villegas[2], María de Lourdes Palacios Robalino[3]

Submitted: 24/01/2023 **Accepted:** 05/04/2023

Abstract

Manipulator robots are the most representative robots in the industrial, medical, service and security sectors; together with mobile robots, they constitute mixed robots that facilitate various tasks in these fields. The points of interest of these robots are kinematic modeling, control, planning, and simulation, hence the importance of their study. The present study aims to obtain the mathematical model for controlling an omnidirectional mobile manipulator robot and its implementation in simulation software. The agile methodology is used for the development of the control algorithms. In order to validate the obtained model. First, the basic behavioral tests are performed, then its movement is simulated by following predefined trajectories representing its behavior. In the simulation of movement in displacements, the mobile manipulator robot finds the position in the order z, y, x with the inverse kinematic model Newton method in a few iterations; on the other hand, with the inverse kinematic model Gradient method, it finds the position in the same order, but with more iterations than the other method; it is also shown the errors of the operating end before reaching the initial position, with results similar to those of the displacement. The results of this research open the possibility of performing different simulations, which allow evaluating the robot's initial conditions and selecting the best ones to optimize resources.

Keywords: Inverse Kinematics, Newton's Method, Gradient Method, Modeling, Mobile Manipulator Robot, Omnidirectional.

1. Introduction

The new industrial revolution combines advanced production and operations techniques with intelligent technologies, demanding greater interest in studying robotics, analytics, artificial intelligence, nanotechnology and IoT. Organizations are looking for technology that fits their needs, with lower costs and generating higher production.

The use of robots in the industry has increased significantly, and now, repetitive, difficult or dangerous tasks are carried out by robotics [1]. Other sectors in which robots are used are military procedures, civilian agriculture, mining, medicine, and education. In this way, robots are used in the automated exploration of hostile terrain, cargo transportation for logistics and distribution of materials, and others for mobilizing people with physical movement difficulties [2].

Manipulator robots are used for handling objects, and their tool design allows them to manipulate objects as small as a thread to as large as a heavy box. In addition, a control system provides these types of robots' intelligence [3].

The points of interest of omnidirectional mobile manipulator robots are kinematic modeling, control, planning, and simulation [4]. Therefore, it is necessary to work with robots so that they are autonomous and accurate systems or at least their error is as close to zero. Taking this into account, the importance of developing mathematical models for omnidirectional mobile manipulator robots that meet the requirements of accuracy, autonomy and stability is verified.

The following works are currently presented around the modeling of mobile manipulator robots.

Bilateral Tele-Operation of Mobile Manipulators is a work developed by the authors [4] that presents a stable

[1]Escuela Superior Politécnica de Chimborazo (ESPOCH), Facultad de Ciencias, Carrera de Matemática, Address: Panamericana Sur Km 1 ½, Riobamba - Ecuador, mmendoza@espoch.edu.ec
[2]Escuela Superior Politécnica de Chimborazo (ESPOCH), Facultad de Ciencias, Carrera de Matemática, Address: Panamericana Sur Km 1 ½, Riobamba - Ecuador, mdavalos@espoch.edu.ec
[3]Escuela Superior Politécnica de Chimborazo (ESPOCH), Facultad de Ciencias, Carrera de Matemática, Address: Panamericana Sur Km 1 ½, Riobamba - Ecuador, mpalacios@espoch.edu.ec

control structure; the image and robot-medium interaction forces are fed back to the operator in order to improve the performance in performing small tasks that require both locomotion and manipulation capabilities, through a wireless LAN network. The system comprises two subsystems a kinematic controller for motion control of the mobile manipulator and a dynamics compensator to minimize velocity tracking errors; through redundancy control of the mobile manipulator, obstacle avoidance and control of the robotic arm configuration is achieved.

Coordinated Control of an Omnidirectional Double Mobile Manipulator, a proprietary paper by [2], proposes a coordinated, cooperative control algorithm for tracking trajectories applied on two anthropomorphic robotic arms mounted on an omnidirectional platform for the transport of a common object. They use kinematic modeling of the whole coupled system, considering as the position of interest the midpoint of the operating ends of each manipulator and their training characteristics, defining reference velocities for the omnidirectional platform and the robotic arms. Stability and robustness are tested using Lyapunov's method, obtaining that the control is asymptotically stable. The experiments were performed through a virtual reality framework confirming the controller's scope to solve different motion problems through a strong selection of control references.

Trajectory control of a mobile manipulator in the presence of base disturbance of [6] presents a manipulator's arm mounted on the base of a mobile robot, with an additional controller whose task is to deal with disturbances of the mobile robot due to dynamic iterations between the base and the arm. Dynamic models are developed for the manipulator's arm and the base, which is the omnidirectional mobile robot using Bond Graph methodology and control strategies for tip trajectory tracking. They use the Bond Graph methodology to develop the dynamic model and control the mobile manipulator.

The authors of the article in [7] describe a mobile, two-handed manipulator with right-handed manipulation capability called MADAR (Mobile Anthropomorphic Dual-Arm Robot) dividing the system into two parts, a circular mobile base with three wheels of novel design that allow omnidirectional mobility and an upper structure carrying two arms in anthropomorphic configuration with right-handed mechanical hands equipped with tactile sensors. The MADAR robot is intended to be an open experimentation platform for testing new planning algorithms for mobile manipulation, as well as the combination with dexterous manipulation algorithms for sensorized hands.

The contribution of the present work is to obtain the mathematical model of a mobile manipulator robot with 4 degrees of freedom and use the algorithms' simulation to validate the motion control, called the system RMMO.

2. Structure of the problem

Modeling and control are topics of research and interest in mobile and manipulator robots, as they are the foundation of robot design. Although robots may be physically the same, each design's mathematical model that represents it varies. Furthermore, the various kinematic and dynamic characteristics of the mathematical model provide different utilities in terms of the properties and behavior that are needed from the robot and are requirements of the user [9].

One of the main problems in the construction of robots is to obtain a geometric model of its structure that allows relating the generalized variables with the Cartesian coordinates of every one of the points that constitute it; this case is known as a direct kinematic problem. Another case arises when the input values are the positions and what must be found are the values of the generalized variables, a problem that arises when positioning a robotic arm or a humanoid leg, which is called an inverse kinematic problem, solved analytically and with infinite solutions. A Jacobian matrix is a relative approach that contributes to solving these problems. However, the kinematic model is not feasible when it is required to manipulate moving objects, for which dynamic models are used with methods such as Lagrange-Euler, Newton-Euler, and generalized d'Alambert equations, among others [8].

3. Mathematical model

This section describes the obtaining of the mathematical and control model of the system. These are conceived by obtaining the models of the robotic arm and the platform, combining them in the omnidirectional mobile manipulator, and subsequently defining the control algorithms.

The study of the kinematic model is a prerequisite for dynamic modeling and other important characteristics of robots, such as stability and control. By performing a correct geometrical structure, kinematic equations that best approximate the real system can be obtained, which is why new and specialized structures are continuously developing [3].

For the above reasons, it is decided to obtain the kinematic modeling of the robot for the control of the omnidirectional mobile manipulator robot and its implementation in simulation software.

3.1. Direct kinematic model of the omnidirectional mobile robot (platform)

Figure 1 shows the four-wheeled omnidirectional mobile robot, $\{\Re\}$ which represents the global coordinate system, with $h(t) = [h_x \quad h_y \quad h_\psi]$ as the position in

$\{\Re\}$ and the angle of rotation h_ψ for the point of interest O (geometric center of the wheels). Where (x_o, y_o) is the position of the point O y ψ the vehicle's orientation related to $\{\Re\}$.

The geometric representation of the omnidirectional robot allows determining the position and orientation of the point of interest (control point) $\boldsymbol{h}(t)$. The model allows locating the point of interest $\boldsymbol{h}(t)$ in any position on the robotic system,

$$\begin{cases} h_x = x + l\,cos(\psi) - m\,sin(\psi) \\ h_y = y + l\,sin(\psi) + m\,cos(\psi) \\ h_\psi = \psi \end{cases} \quad (1)$$

The kinematic model of the omnidirectional robot (2) can be obtained by deriving (1) for time, therefore,

$$\begin{bmatrix} \dot{h}_x \\ \dot{h}_y \\ \dot{h}_\psi \end{bmatrix} = \begin{bmatrix} C_\psi & -S_\psi & -lS_\psi - mC_\psi \\ S_\psi & C_\psi & lC_\psi - mS_\psi \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_l \\ u_m \\ \omega \end{bmatrix} \quad (2)$$

where, $C_\psi = \cos(\psi)$, $S_\psi = \sin(\psi)$.

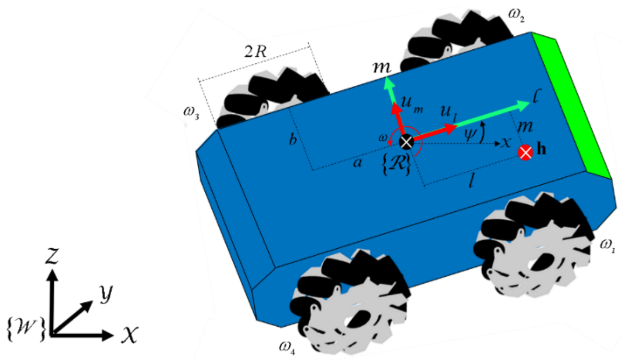

Figure 1. Mobile robot with omnidirectional drive.
Source: Mendoza, María.

The omnidirectional mobile robot is characterized by being holonomic, as it has no restrictions on movement.

The basic equations of motion of the mobile robot help to represent the u_l and u_m that are the linear velocities of front and side of the mobile; and with them to find the ω that is the angular velocity of the mobile robot; being a , b , and R the physical parameters of the robot where R is the radius of the omnidirectional wheels, a is the distance of the wheel with respect to the center of the robot on the axis L and b is the distance of the wheel with respect to the center of the robot on the axis M . l and m are the position distances with respect to the center of the platform where the manipulator robot will be located with respect to the axis L and M , respectively.

Now, the matrix (3) relating the maneuverability velocities to the angular velocities of each wheel is represented as follows:

$$\begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \end{bmatrix} = \frac{1}{R} \begin{bmatrix} 1 & -1 & -(a+b) \\ 1 & 1 & (a+b) \\ 1 & 1 & -(a+b) \\ 1 & -1 & (a+b) \end{bmatrix} \begin{bmatrix} u_l \\ u_m \\ \omega \end{bmatrix} \quad (3)$$

The kinematic model of the platform found allows the robotic arm to be located in any platform quadrant to arrange the space according to the designer's preferences and maneuverability; however, in this work, the manipulator will be located in the upper center of the platform.

3.2. Modeling of the robotic manipulator arm

In order to adequately simulate the movement of the robot, it is proposed that the task to be performed by the robot will be the manipulation of small objects (beakers, test tubes, among other objects belonging to a laboratory). For this purpose, an anthropomorphic arm with 4 degrees of freedom will be used.

3.2.1. Direct Kinematic Model

Figure 3 shows the configuration of the robotic manipulator to be modeled, with $h_b(t)$ as the position of the point of interest (operating end), where $\{\Re_B\}$ represents the local coordinate reference system for the robotic arm, with (h_{bx} , h_{by} , h_{bz}) being coordinates of the point $h_b(t)$ with respect to $\{\Re_B\}$.

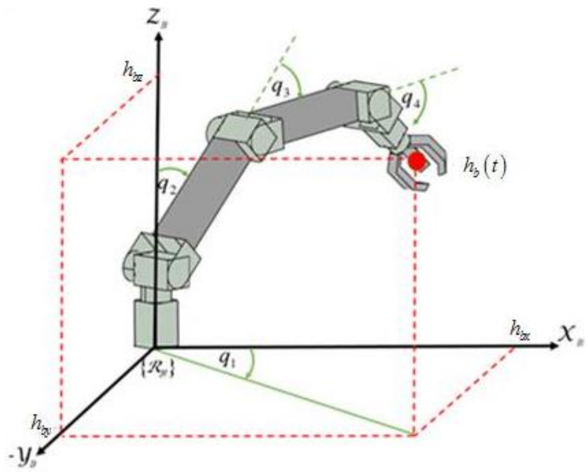

Figure 2. Anthropomorphic Robotic Arm 4 g.d.l.
Source: Mendoza, María.

The geometrical representation of the robotic arm 4 g.d.l. allows determining the position of the end-effector

$h_b(t)$ through the *direct kinematics* of the robot, whose matrix representation is (4):

$$\begin{cases} h_{bx} = l_2S_2C_1 + l_3S_{23}C_1 + l_4S_{234}C_1 \\ h_{by} = l_2S_2S_1 + l_3S_{23}S_1 + l_4S_{234}S_1 \\ h_{bz} = l_1 + l_2C_2 + l_3C_{23} + l_4C_{234} \end{cases} \quad (4)$$

where, $C_\alpha = \cos(\alpha)$, $C_{\alpha\beta} = \cos(\alpha+\beta)$, $S_\alpha = \sin(\alpha)$, $S_{\alpha\beta} = \sin(\alpha+\beta)$; l_1, l_2, l_3, and l_4 represent the lengths of each rigid link of the robotic arm; and q_1, q_2, q_3, q_4 are the rotation angles for each degree of freedom of the manipulator robot. Therefore, the *kinematic model* of the robotic arm is defined in matrix form as (5):

$$\begin{cases} \dot{h}_{bx} = -(l_2S_1C_2 + l_3S_1C_{23} + l_4S_1C_{234})\dot{q}_1 + (l_2C_1C_2 + l_3C_1C_{23} + l_4C_1C_{234})\dot{q}_2 + \ldots \\ \quad (l_3C_1C_{23} + l_4C_1C_{234})\dot{q}_3 + (l_4C_1C_{234})\dot{q}_4 \\ \dot{h}_{by} = (l_2C_1C_2 + l_3C_1C_{23} + l_4C_1C_{234})\dot{q}_1 + (l_2S_1C_2 + l_3S_1C_{23} + l_4S_1C_{234})\dot{q}_2 + \ldots \\ \quad (l_3S_1C_{23} + l_4S_1C_{234})\dot{q}_3 + (l_4S_1C_{234})\dot{q}_4 \\ \dot{h}_{bz} = -(l_2S_2 + l_3S_{23} + l_4S_{234})\dot{q}_2 - (l_3S_{23} + l_4S_{234})\dot{q}_3 - l_4S_{234}\dot{q}_4 \end{cases} \quad (5)$$

The robot manipulator is redundant since the number of degrees of mobility is greater than the number of variables necessary to describe a specific task.

3.2.2. Inverse Kinematic Model of the Manipulator

To solve the inverse kinematic problem, the values of the joint variables are set to achieve a desired position and movement. It was mentioned in the previous topic that the manipulator is redundant, so $n > m$, being n the number of degrees of mobility and m the number of variables; in this case, it is better to use a numerical solution iteratively. The study will consider the solutions with Newton's and Gradient methods.

A) Inverse Kinematic Model, Newton's Method

Newton's method is also known as the tangent method. For a differentiable function $f(x)$, find a root $x*$ of $f(x*) = 0$, iterating as $x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$, an approximate sequence $\{x1, x2, x3, x4, x5.,..\} \to x*$.

Newton's method is used when a closed solution q is needed for $r_d = f_r(q)$, i.e., it does not exist or is too difficult to find, for which the Jacobian matrix $J_r(q) = \frac{\partial f_r(q)}{\partial q}$ is used. Equations (6) and (7) describe the method to use

$$r_d = f_r(q) = f_r(q^k) + J_r(q^k)(q - q^k) + o(\|q - q^k\|) \quad (6)$$

$$q^{k+1} = q^k + J_r(q^k)[r_d - f_r(q^k)] \quad (7)$$

Convergence for q^0 (initial guess), close enough to some $q*$: $f_r(q*) = r_d$ problems near the singularity of the Jacobian matrix $J_r(q)$.

In the case of a manipulator robot with redundancy $m < n$, a pseudoinverse $J_r^{\#}(q)$ is used and has a quadratic convergence rate when it is close to a solution.

B) Inverse Kinematic Model, Gradient Method

The Gradient method is used to obtain the minimum of a function, considering the error it can be used to obtain the inverse kinematics.

Minimizes the error of function (8) and (9).

$$H(q) = \frac{1}{2}\|r_d - f_r(q)\|^2 = \frac{1}{2}(r_d - f_r(q))^T(r_d - f_r(q)) \quad (8)$$

$$q^{k+1} = q^k - \alpha\nabla_q H(q^k) \quad (9)$$

from

$$\nabla_q H(q) = \left(\frac{\partial H(q)}{\partial q}\right)^T = -\left((r_d - f_r(q))^T\left(\frac{\partial f_r(q)}{\partial q}\right)\right)^T = -J_r^T(q)(r_d - f_r(q)) \quad (10)$$

is obtained

$$q^{k+1} = q^k + \alpha J_r^T(q^k)(r_d - f_r(q^k)) \quad (11)$$

The scalar step size must be chosen to ensure a decrease of the error function at each iteration too large. The values can lead the method to miss the minimum. On the other hand, when the size is too small, the convergence is extremely slow.

The Gradient method is computationally simpler and uses the transposed Jacobian matrix rather than its pseudoinverse. As a result, it may not converge to a solution, but it never diverges. The discrete-time evolution of the continuous scheme is represented in (12).

$$q^{k+1} = q^k + \Delta T J_r^T(q^k)(r_d - f_r(q^k)), \alpha = \Delta T \quad (12)$$

3.3.3. RMMO Modeling

The nomenclatures developed in the independent models of the robotic arm (manipulator) and the mobile robot (platform) are used to obtain the mathematical model of the mobile manipulator robot. Figure 3 presents the omnidirectional mobile manipulator robot, where h is the position of the operating end for $\{W\}$

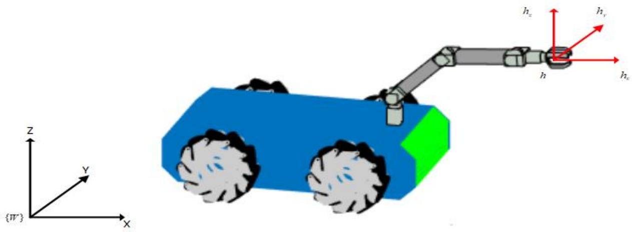

Figure 3. RMMO omnidirectional mobile manipulator.
Author: Mendoza, María.

The geometrical representation of the omnidirectional mobile manipulator robot consists of a 4 g.d.l. The robotic arm (Figure 2) on an omnidirectional platform (Figure 3) allows to determine the position of the end effector $\mathbf{h}(t)$ (Figure 4) through its *direct kinematics:*

$$\begin{cases} h_x = x + lC_\psi - mS_\psi + l_2S_2C_{1\psi} + l_3S_{23}C_{1\psi} + l_4S_{234}C_{1\psi} \\ h_y = y + lS_\psi + mC_\psi + l_2S_2S_{1\psi} + l_3S_{23}S_{1\psi} + l_4S_{234}S_{1\psi} \\ h_z = h + l_1 + l_2C_2 + l_3C_{23} + l_4C_{234} \end{cases} \tag{13}$$

Where $\mathrm{C}_\alpha = \cos(\alpha), \mathrm{C}_{\alpha\beta} = \cos(\alpha+\beta)$, $S_\alpha = \sin(\alpha)$, $S_{\alpha\beta} = \sin(\alpha+\beta)$; l and m represents the position of the robotic arm with respect to the reference frame $\{R\}$; x and y denotes the position of the moving platform with respect to the inertial reference frame $\{W\}$; and h is the height of the omnidirectional robot. While l_1, l_2, and l_3 l_4 represent the lengths of each rigid link of the robotic arm. Therefore, the *kinematic model* of the omnidirectional robot is defined as system (14):

$$\begin{cases} \dot{h}_x = C_\psi u_l - S_\psi u_m - (lS_\psi + mC_\psi)\omega - ... \\ \qquad (l_2S_{1\psi}C_2 + l_3S_{1\psi}C_{23} + l_4S_{1\psi}C_{234})\dot{q}_1 + ... \\ \qquad (l_2C_{1\psi}C_2 + l_3C_{1\psi}C_{23} + l_4C_{1\psi}C_{234})\dot{q}_2 + ... \\ \qquad (l_3C_{1\psi}C_{23} + l_4C_{1\psi}C_{234})\dot{q}_3 + (l_4C_{1\psi}C_{234})\dot{q}_4 \\ \dot{h}_y = S_\psi u_l + C_\psi u_m + (lC_\psi - mS_\psi)\omega + ... \\ \qquad (l_2C_{1\psi}C_2 + l_3C_{1\psi}C_{23} + l_4C_{1\psi}C_{234})\dot{q}_1 + ... \\ \qquad (l_2S_{1\psi}C_2 + l_3S_{1\psi}C_{23} + l_4S_{1\psi}C_{234})\dot{q}_2 + ... \\ \qquad (l_3S_{1\psi}C_{23} + l_4S_{1\psi}C_{234})\dot{q}_3 + (l_4S_{1\psi}C_{234})\dot{q}_4 \\ \dot{h}_z = -(l_2S_2 + l_3S_{23} + l_4S_{234})\dot{q}_2 - (l_3S_{23} + l_4S_{234})\dot{q}_3 - l_4S_{234}\dot{q}_4 \end{cases} \tag{14}$$

For the inverse kinematic modeling, the same criteria mentioned above are applied, now with the Jacobian matrix of the whole system.

4. Simulation algorithms

The process to implement the model obtained is described through the four phases of the agile methodology.

Matlab will be used for the development of the control algorithms. Matlab is a high-performance language that integrates calculation, visualization and programming in a user-friendly environment where problems and solutions are expressed in mathematical notation [10].

4.1. Software requirements analysis

The following conditions are necessary for the development of the kinematic modeling software of the system:

The robot has a rigid physical structure, i.e., no flexible parts.

The movement space is free of obstacles.

The path along which the mobile robot will move is free of friction and irregularities, completely smooth and level.

Control algorithms are implemented to solve two motion problems: positioning and trajectory tracking.

These are based on the models found, for each of which parameters such as initial positions and orientation and physical values of the robot must be established according to the characteristics desired by the robot designer.

4.2. Design and Implementation

The system's conception is presented through block diagrams of the controls implemented according to the models found.

4.2.1. Direct kinematic control

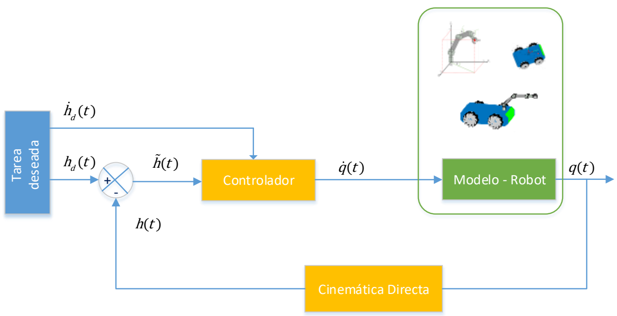

Figure 3. Block diagram of the Direct Kinematic Model.
Source: Mendoza, María.

Figure 4 shows four blocks corresponding to the direct kinematic model:

Desired task is the block where you find the specifications such as the path that you want the robot to perform, for example a circular path or a line that leads to a point somewhere.

Controller is the block containing the control errors; the controller by means of the Jacobian matrix, the weight or gain matrix, the desired task speeds and the control law.

Model that refers to the type of robot (manipulator, mobile, mobile manipulator) and the characteristics that differentiate it from the rest.

Direct kinematic model whose matrix is the one found with the analysis developed in previous sections.

4.2.2. Inverse kinematic control

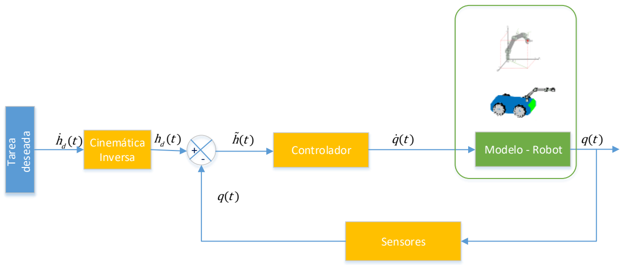

Figure 4. Block diagram of the Inverse Kinematic Model.
Source: Mendoza, María

The blocks representing the inverse kinematic model can be seen in Figure 5 and are described below:

The desired task is the block where you find the specifications, such as the path you want the robot to perform, for example, a circular path or a line that leads to a point somewhere.

Inverse kinematics whose matrix is conceived by the final positions of the robot.

Controller is the block containing the control errors; the Newton or Gradient algorithm that will be taken as the method of solving the inverse kinematic problem.

Model that refers to the type of robot (manipulator, mobile, mobile manipulator) and the characteristics that differentiate it from the rest.

4.3. Verification and Maintenance

Once the control algorithms have been built, they are executed using the trial and error method, i.e., executed, and the errors are corrected; the most frequent errors are syntax and infinite loops.

5. Results

The simulation results evaluate the system performance, and basic motion simulation tests are performed using the inverse kinematic model and tests for tracking predefined trajectories using the direct kinematic model.

5.1. Basic Motion Simulation Tests

The basic motion simulation tests evaluate the motion of the omnidirectional mobile manipulator with the inverse kinematic model of Newton's method and the Gradient method, respectively.

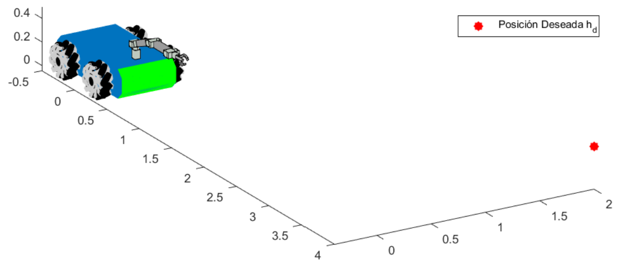

Figure 5. Position of the RMMO
Source: Mendoza, María

Figure 6 shows where the RMMO is located, and the red dot is the desired position to be reached.

5.1.1. Basic test of the inverse kinematic model Newton's method

The motion control by inverse kinematic modeling with Newton's method allows to bring the RMMO to reach the given desired position; in Figure 7 the realized motion can be observed.

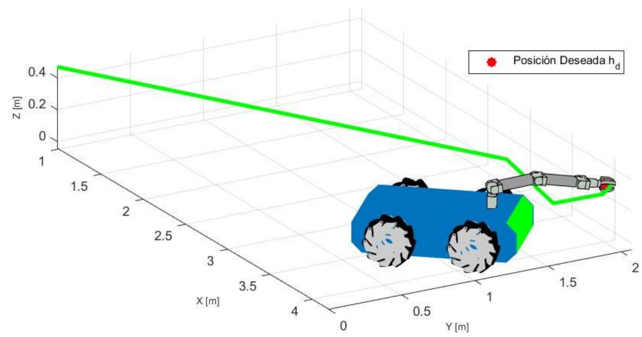

Figure 6. Inverse kinematic motion of the RMMO. Newton's method
Source: Mendoza, María

The RMMO reaches the position z with 2 iterations, then x in 4.4 iterations and finally y with 4.6 iterations performed (displacements) before reaching the desired position, as shown in Figure 8.

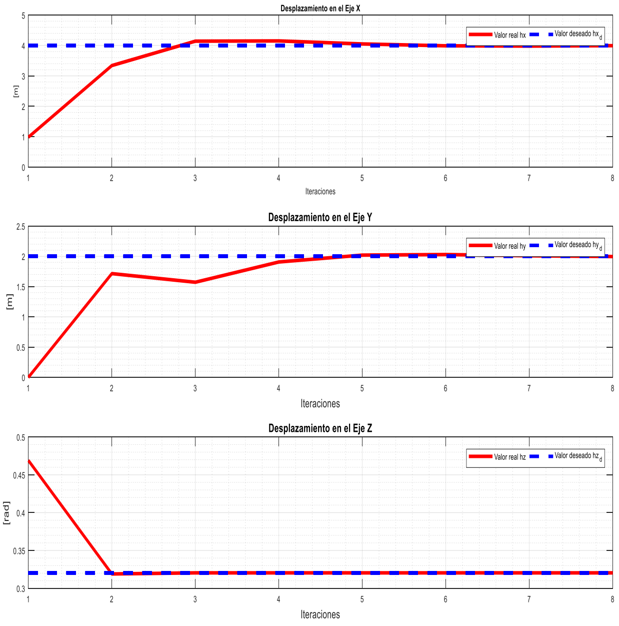

Figure 7. Displacements realized kinematic inverse motion of the RMMO. Newton's method
Source: Mendoza, María

Figure 9 shows the errors of the RMMO operating end before reaching the desired position. First, it reaches the position z with 2 iterations, secondly y with 4.6 iterations and finally x in 4.9 iterations performed.

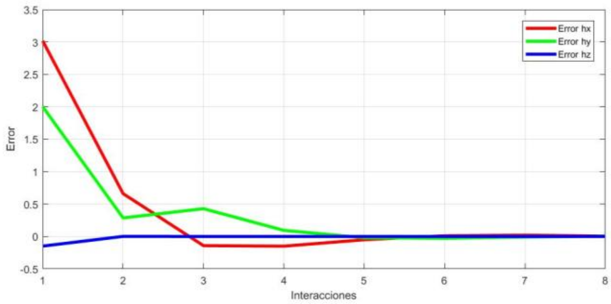

Figure 8. Inverse kinematic motion of the RMMO. Newton's method. Operating end error
Source: Mendoza, María

5.1.2. Basic test of the inverse kinematic model Gradient method

The motion performed by the RMMO using inverse kinematic modeling with the Gradient method can be seen in Figure 10.

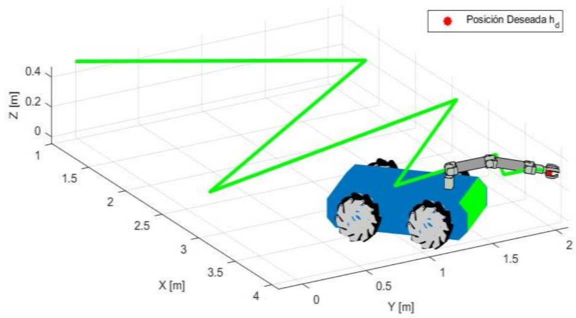

Figure 9. Inverse kinematic motion of the RMMO. Gradient Method
Source: Mendoza, María

Figure 11 shows the displacements of the movement performed by the RMMO, reaching first the position x with 7 iterations, then y in 9 iterations and finally z with 13 iterations performed before reaching the desired position.

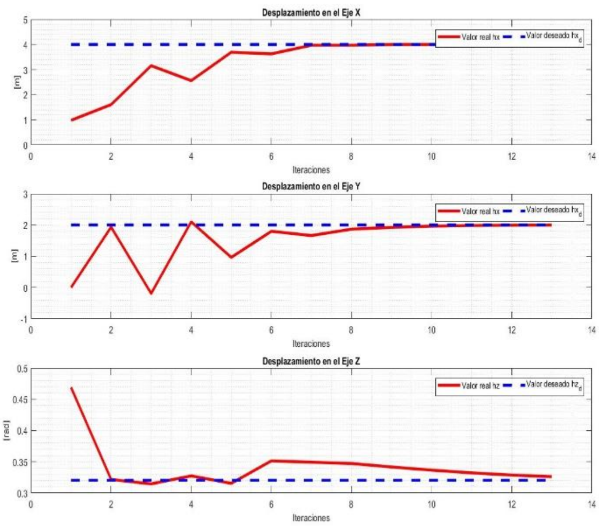

Figure 10. Displacements made by kinematic inverse motion of the RMMO. Gradient Method
Source: Mendoza, María

The operating end of the RMMO before reaching the desired position, first reaches the position z with 2 iterations, secondly x with 9 iterations and finally y in 11 iterations performed, as shown in Figure 12.

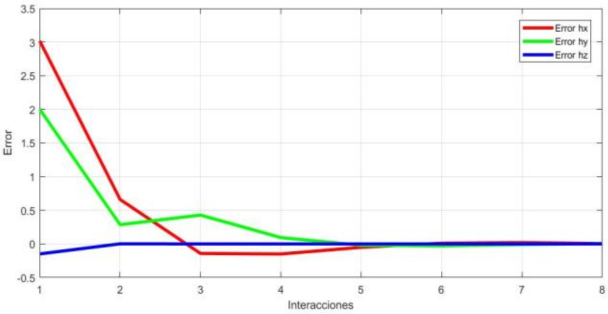

Figure 11. Inverse kinematic motion of the RMMO. Gradient method. Operating end error

Source: Mendoza, María

Table 1 shows a comparative summary of the results obtained from the basic motion test using the inverse kinematic model with Newton's method versus the model with the Gradient method. The inverse kinematic model with Newton's method obtained fewer iterations to reach the desired point.

Table 1: Comparison of inverse kinematic model Newton's method vs Gradient method

Error (Displacement)	Inverse Kinematic Modeling Control Method			
	Newton		Gradient	
	RMMO	Operating End	RMMO	Operating End
X	4.4	4.9		
Y	4.6	4.6		
Z				

Source: Mendoza, María

5.2. Predefined trajectory tracking tests

The predefined trajectory tracking simulation tests evaluate the RMMO motion using the direct kinematic model.

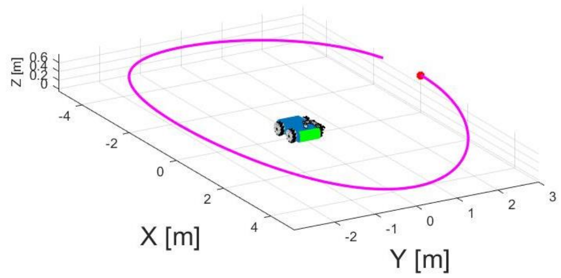

Figure 12. Initial position and trajectory of the RMMO

Source: Mendoza, María

Figure 13 shows the initial position of the RMMO and the trajectory to be followed.

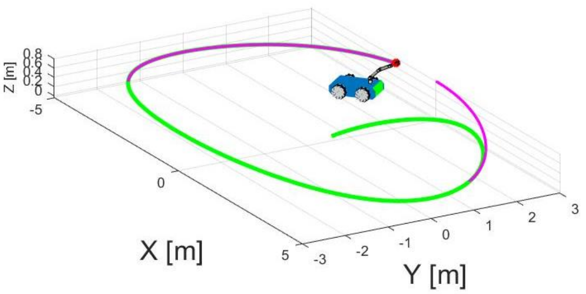

Figure 13. RMMO motion using the direct kinematic model.

Source: Mendoza, María

The RMMO using the direct kinematic model, tracks the given trajectory not completely as from its initial location it searches for the trajectory and follows it, as shown in Figure 14.

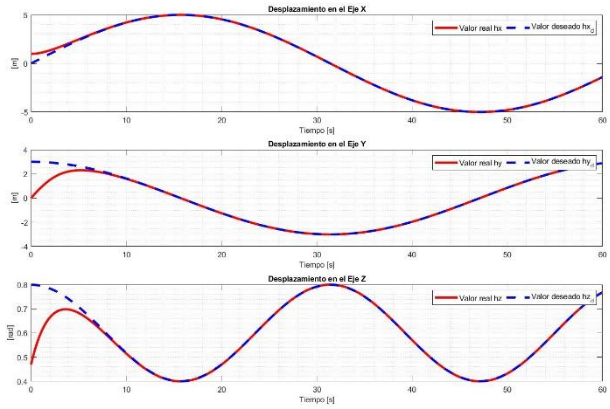

Figure 14. RMMO motion using the direct kinematic model. Displacements

Source: Mendoza, María

The RMMO reaches the position x in 3 sec, y in 6 sec and finally z in 6.4 sec, from its initial position to reach the trajectory to follow, as shown in Figure 15.

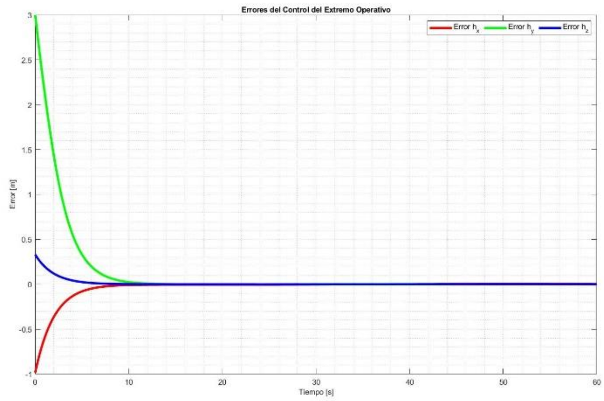

Figure 15. Direct Kinematic Motion of the RMMO. Operating end error.

Source: Mendoza, María

Figure 16 shows how the operating end of the RMMO, before reaching the desired position, first reaches the position z in 6 seconds, secondly x in 8 seconds and finally y in 10 seconds.

6. Conclusions

The system is conceived with a mobile robot as a platform, and on this, a 4 g.d.l. manipulator robot that can be located at any control point, having the manipulator in the upper center as the control point. For the control of the direct kinematic model, inverse kinematic with Newton's method and inverse kinematic with the Gradient method are implemented.

In the evaluation tests of the control algorithms, it is evident that the singularity of the RMMO must be taken into account, especially in positions with π angles, so if it is in singularity positions, an infinite loop of motion is performed.

When the manipulator tries to reach a point far away from the end effector, it makes abrupt movements in order to achieve the given task.

Newton's method to solve the inverse kinematics problem presents undesired configurations tending to infinity. However, it converges faster under optimal conditions than the inverse kinematic model with the Gradient method.

When using the Gradient method to solve the inverse kinematics problem, there is no need to worry about singularity points. However, this method takes a long time to converge.

References

[1] Isaksson, M., Nyhof, L., & Nahavandi, S. (2015). On the feasibility of utilising gearing to extend the rotational workspace of a class of parallel robots. Robotics and Computer-Integrated Manufacturing, 35, 126-136, https://doi.org/10.1016/j.rcim.2015.03.004

[2] Ortiz, J. S., Molina, F., & Andaluz, V. H. (2018). Coordinated Control of a Omnidirectional Double Mobile Manipulator, 1. https://doi.org/10.1007/978-981-10-6451-7

[3] Andaluz, G., Andaluz, V., & Rosales, A. (2013). Modelación, Identificación y Control de Robots Móviles. Escuela Politécnica Nacional, 9. Recuperado de http://bibdigital.epn.edu.ec/bitstream/15000/4912/1/Modelación%2C Identificación y Control de.pdf%0Ahttp://eelalnx01.epn.edu.ec/handle/15000/4321

[4] Gracia, L. (2000). Modelado Cinemático y Control de Robots Móviles con Ruedas. universidad Politécnica de Valencia.

[5] Ortiz, J., Morales, J., Peréz, M., & Andaluz, V. (2015). Tele-Operación Bilateral de Manipuladores Móviles, 35(2). Recuperado de https://revistapolitecnica.epn.edu.ec/images/revista/volumen35/tomo2/TeleOperacionBilateraldeManipuladoresMoviles.pdf

[6] Ram, R. V, Pathak, P. M., & Junco, S. J. (2018). Trajectory control of a mobile manipulator in the presence of base disturbance. https://doi.org/10.1177/0037549718784186

[7] Suárez, R., Palomo-Avellaneda, L., Martínez, J., Clos, D., & García, N. (2020). Manipulador móvil, bibrazo y diestro con nuevas ruedas omnidireccionales, 17, 10-21. https://doi.org/https://doi.org/10.4995/riai.2019.11422

[8] Acosta, L., Sigut, M. (2005). Matemáticas y robótica. Sociedad, Ciencia, Tecnología y Matemáticas. Obtenido de https://imarrero.webs.ull.es/sctm05/modulo2tf/4/lacosta.pdf

[9] Aracil, R., Balaguer, C., & Armada, M. (2008). Service Robots. RIAI - Revista Iberoamericana de Automática e Informática Industrial, 5(2), 6-13. https://doi.org/10.1016/s1697-7912(08)70140-7

[10] Velásquez Costa, J. A. (2014). Software para el modelamiento, simulación y programación de aplicaciones robotizadas. Industrial Data, 12(1), 032. https://doi.org/10.15381/idata.v12i1.6082