

Development of interface for kinematic analysis of a delta-type parallel robot

Desarrollo de interfaz para análisis cinemático de robot paralelo tipo delta

RODRÍGUEZ-FRANCO, Martín Eduardo†*, JARA-RUIZ, Ricardo, LÓPEZ-ÁLVAREZ, Yadira Fabiola and GARCÍA-RODRÍGUEZ, Juan Carlos

Universidad Tecnológica del Norte de Aguascalientes

ID 1st Author: *Martín Eduardo, Rodríguez-Franco* / ORC ID: 0000-0002-6804-4777, Researcher ID Thomson: T-1539-2018, CVU CONACYT ID: 660892.

ID 1st Co-author: *Ricardo, Jara-Ruiz* / ORC ID: 0000-0001-7725-4138, Researcher ID Thomson: T-1532-2018, CVU CONACYT ID: 630276

ID 2nd Co-author: *Yadira Fabiola, López-Álvarez* / ORC ID: 0000-0002-9041-1908, Researcher ID Thomson: T-1555-2018, CVU CONACYT ID: 375952

ID 3rd Co-author: *Juan Carlos, García-Rodríguez* / ORC ID: 0000-0002-3602-7809, Researcher ID Thomson: J-8291-2017, CVU CONACYT ID: 677265

DOI: 10.35429/JCSI.2021.20.7.18.27

Received July 30, 2021; Accepted October 28, 2021

Abstract

The development and implementation process of a computer interface for the kinematic analysis of a parallel robot, in delta configuration, and its application to a previously formed prototype are exposed. Being identified the associated equations, and deduced the respective geometric parameters. On the other hand, the synthesis of the direct and inverse kinematic models, with the Matlab software, guarantees the calculation of a specific Cartesian position, in the end effector of the robot used, once certain joint values have been assigned to it, or vice versa. Finally, a user-friendly graphical interface is created, whose functions are: data entry, resolution of the models described, issuance of the corresponding results, representation of the robot used and its physical manipulation. The results obtained in the real location of the end effector with respect to the values deduced by the interface, are competitive for both models analyzed, even though the prototype used operates by means of servomotors. An average position error of 0.083 cm per axis and overall of 0.006 cm is observed during the tests developed.

Graphical user interface, Kinematic analysis, Parallel robot in delta configuration

Resumen

Se expone el proceso de desarrollo e implementación de una interfaz por computadora para el análisis cinemático de un robot paralelo, en configuración delta, y su aplicación a un prototipo previamente conformado. Siendo identificadas las ecuaciones asociadas, y deducidos los parámetros geométricos respectivos. Por su parte, la sintetización de los modelos cinemáticos directo e inverso, con el software Matlab, garantizan el cálculo de una posición cartesiana específica, en el efector final del robot utilizado, una vez asignados ciertos valores articulares al mismo, o viceversa. Finalmente, se constituye una interfaz gráfica de aspecto amigable para el usuario, cuyas funciones son: el ingreso de datos, la resolución de los modelos descritos, la emisión de los resultados correspondientes, la representación del robot utilizado y su manipulación física. Los resultados obtenidos en la localización real del efector final respecto a los valores deducidos por la interfaz, son competitivos para ambos modelos analizados, aun pese a que el prototipo empleado opera mediante servomotores. Se observa un error de posición promedio de 0.083 cm por eje y global de 0.006 cm, durante las pruebas desarrolladas.

Interfaz gráfica de usuario, Análisis cinemático, Robot paralelo en configuración delta

Citation: RODRÍGUEZ-FRANCO, Martín Eduardo, JARA-RUIZ, Ricardo, LÓPEZ-ÁLVAREZ, Yadira Fabiola and GARCÍA-RODRÍGUEZ, Juan Carlos. Development of interface for kinematic analysis of a delta-type parallel robot. Journal of Computational Systems and ICTs. 2021. 7-20: 18-27

* Correspondence to Author (martin.rodriguez@utna.edu.mx)

† Researcher contributing as first Author.

Introduction

In its original conception, a parallel robot in delta configuration consists of three kinematic chains connected to both a base and a mobile platform (Clavel, 1991). Each chain, also called extremity, is made up of a set of arm and forearm joined by a ball joint, and it is driven from the movement generated by motors located on the base of the robot itself (Pierrot, Reynaud, & Fournier, 1990). Therefore, the adoption of a specific location by the mobile platform is established by combining the movement of each chain, restricting it to remain parallel to the base (Codourey, Clavel, & Burckhardt, 1992).

On the other hand, since the mobile platform supports the end effector that allows the delta robot to perform a specific function, it is essential to know its precise position with respect to its base (Khorasani, Gholami, & Taghirad, 2015). Hence, since each extremity of the robot forms a spatial parallelogram (Vischer & Clavel, 1998) which restricts the specific location of its mobile platform (Idumudi & Javed, 2021), the models for the respective forward or inverse kinematic analysis are defined. Model solution establishes the location of the end effector from the movement of each actuator or vice versa, respectively (Kuo, 2016).

In particular, the resolution of the direct kinematic model lies in determining the location of the end effector, in Cartesian space, by assigning specific joint values to the actuators of the delta robot. Whereas, to solve the inverse model, one starts from the proposal of a specific position of the end effector, in Cartesian space; therefore, it is necessary to find the values of the joint variables that ensure said location (López, Castillo, García, & Bashir, 2006) (Abele, Weigold, & Rothenbücher, 2007) (Peña-Cortés, Martínez-Oviedo, & Cárdenas-Herrera, 2011).

From the evolution of computing, the development of software interfaces has expanded the implementation and exploration of algorithms, increasingly sophisticated, based on multiple methodologies for the efficient and precise analysis of the capabilities of a delta robot (Rivas, Galarza, Tumbaco, Quimbita, & Galarza, 2015).

Among the incorporated functions, the solution to the respective kinematic models (Poppeova, Uricek, Rejda, & Bulej, 2012), the simulation of the robot's operating conditions (Hendriko, Nurkhamdi, & Imam, 2018) and the reconfiguration of its elements for the optimization of its function are distinguished (Maya, Castillo, Lomelí, González-Galván, & Cárdenas, 2013). Likewise, the application of analytical interfaces to enable and manipulate physical prototypes is becoming more and more common (Huang, Weng, Wei, & Kamruzzaman, 2021). Such a situation is prompted by the simplicity in the use of common communication buses, such as USB (Universal Serial Bus), for the exchange of information between the computer and a peripheral device (Antal, 2018). Equally favorable has been the commercial availability of widely disseminated electronic cards, for their incorporation in academic and research projects, because of their low cost, easy programming and acceptable processing capacity (Villegas, Avizzano, Ruffaldi, & Bergamasco, 2015).

Background

Prior to this study, a prototype of a delta-type parallel robot was enabled; whose integration and commissioning have allowed ensuring correct performance in the adoption of the joint values designated by the user, from a computer. Thus, the continuity in the analysis of this implement aims to the development of a graphical interface that allows its management. Such interface will have to expose with certainty the spatial location of the end effector of the analyzed robot, since the set points previously entered corresponded to the direct movement of the actuators, without these being issued as a result of a formal calculation process.

Therefore, in order to determine the Cartesian coordinates of the end effector of the delta robot, from the manipulation of the respective joints, or vice versa, the analysis of the respective kinematic model and the development of an algorithm that synthesizes said mathematical process are essential. It is worth mentioning that, the establishment of the spatial location of the end effector of a robot based on a formal methodology, in this case, its kinematic modeling, leads to ensure with certainty that the data emitted assumes at least the restrictions of the movement that due to its mechanical structure, a particular robot is able to execute.

The addition of an interface for the management of a delta robot previously available strengthens the functionality achieved up until to now by it, in order to constitute a test equipment that allows characterizing its operation with respect to that described by a serial configuration. An effort to provide the user, in this case, Mechatronics Engineering students from the Technological University of North Aguascalientes, a friendly environment for handling the said robot, will be made. Being understandable the deduction of the results of the own kinematic models, through their graphical presentation, and possible the transfer of such values directly to the implement controller, for its adoption by it.

Kinematic analysis of the delta robot

Dr. Raymond Clavel proposed the structure of a parallel robot with three translational and one rotational degrees of freedom, which he called the delta configuration, and whose kinematic synthesis is based on the study of parallelograms (Clavel, 1991). Each robot extremity, made up of two links, forms a parallelogram; in this, the exit link remains in a fixed orientation with respect to the entry link, to which it is held together. The use of three parallelograms completely restricts the location of the robot's mobile platform, giving it only three degrees of translation freedom (Maya, Castillo, Lomelí, González-Galván, & Cárdenas, 2013).



Figure 1 Reference system and geometric parameters of the parallel robot in delta configuration

Source: Own Elaboration, 2021

ISSN-2444-5002

ECORFAN® All rights reserved

For its part, the geometric analysis of the delta robot assumes that its three extremities are identical in length (Angel & Viola, 2016) (Kuo, 2016). While the respective geometric parameters, shown in figure 5, are defined as the length of the actuated link L_A , the length of the driven link L_B , the radius of the fixed platform or base R_A , the radius of the mobile platform or end effector R_B , the fixed platform radius angles θ_i ($i = 1, 2, 3$) and the joint angles that determine each extremity configuration $\varphi_{1i}, \varphi_{2i}, \varphi_{3i}$ ($i = 1, 2, 3$) (Laribi, Romdhane, & Zeghloul, 2007) (Sánchez, Castillo, González, & Balmaceda, 2015).

Thereby, given the global reference system (X, Y, Z) of the delta robot, represented in figure 5, the point P with coordinates (X_P, Y_P, Z_P) is established at the center of the mobile platform. Therefore, in the respective forward kinematic model, the unknowns will correspond to the location of said point P , once the joint angles $\varphi_{1i}, \varphi_{2i}, \varphi_{3i}$ ($i = 1, 2, 3$) have been assigned, through simultaneous resolution of (1); since, $i = 1, 2, 3$.

$$(X_i - X_P)^2 + (Y_i - Y_P)^2 + (Z_i - Z_P)^2 = L_B^2 \quad (1)$$

Where:

$$X_i = (r + L_A \cos \varphi_{1i}) \cos \theta_i \quad (2)$$

$$Y_i = (r + L_A \cos \varphi_{1i}) \sin \theta_i \quad (3)$$

$$Z_i = -L_A \sin \varphi_{1i} \quad (4)$$

Considering $r = r_A - r_B$, (1) in its individual form represents a sphere with radius L_B , centered at the point $S_i = (X_i, Y_i, Z_i)$. The solution of such a system of equations is represented by the point drawn at the intersection of the three spheres described by each extremity.

On the other hand, for the respective inverse kinematic model, the unknowns are the joint angles $\varphi_{1i}, \varphi_{2i}, \varphi_{3i}$ ($i = 1, 2, 3$) to be established, given the proposal of a particular spatial location for point P ; where (5) defined for the j -th extremity of the robot can be expressed as a function of $\cos \varphi_{1i}$ and $\sin \varphi_{1i}$.

$$l_i \cos \varphi_{1i} + m_i \sin \varphi_{1i} = n_i \quad (5)$$

where:

$$l_i = 2rL_A - 2L_A X_P \cos \theta_i - 2L_A Y_P \sin \theta_i \quad (6)$$

$$m_j = 2L_A Z_P \quad (7)$$

$$n_j = 2rX_P \cos \theta_j - 2rY_P \sin \theta_j + X_P^2 + Y_P^2 + Z_P^2 + r^2 + L_A^2 - L_B^2 \quad (8)$$

Moreover, (5) can have a solution if and only if:

$$\frac{n_i}{\sqrt{l_i^2 + m_i^2}} \leq 1 \leftrightarrow n_i^2 - (l_i^2 + m_i^2) \leq 0 \quad (9)$$

If a value greater than zero is established in (9), it is said that the represented point is mechanically unreachable by the end effector of the delta robot, causing a singularity condition.

Determination of geometric parameters

As previously analyzed, the geometric parameters of a robot are directly associated with the mechanical characteristics that lead to physically differentiate its structure, with respect to some other existing configuration. Hence, in order to establish the geometric parameters of the delta robot used, it was necessary to identify each of the mechanical elements that constitute it, as well as to define its particular function within the mechanism as a whole. Each element described has a specific length that represents it; these values are concentrated in table 1.

Geometric parameter	Value
Base radius (R_A)	7.5 cm
End effector radius (R_B)	4.5 cm
Actuated link length (L_A)	9.5 cm
Driven link length (L_B)	32 cm

Table 1 Definition of geometric parameters of the delta robot used

Source: Own Elaboration, 2021

The geometric parameters are incorporated into the respective kinematic models, to determine with great precision the position of the end effector or the movement that the actuators must execute, as required. However, to validate that the real movement, performance by each motor, will lead to the correct location of the end effector in the location designated by the user; the origin of the Cartesian reference system is established, at the center of the fixed platform of the robot used, just on its underside.

The proposed axes and their respective orientation can be seen in figure 1. Since a previously constituted prototype of a delta robot is being used for the present study, it is noteworthy that both the motors used to transmit movement to each end of the robot, as well as to drive the gripper, are servomotors. In this way, an easy manipulation of the position that the actuators may acquire during the request, by the user or the control system, of a specific joint movement value or the location of a point designated by Cartesian coordinates, with respect to the reference system, are guaranteed.

Algorithm programming

For a particular robotic configuration, the solution of the associated forward and inverse kinematic problems involves deducing the spatial position that the end effector must reach, by assigning certain values to its joint variables, or vice versa, as appropriate. Therefore, after the identification of the mathematical equations that lead to the solution of both kinematic models, for the analyzed delta robot, it was necessary to program such expressions, in order that the results given, regardless of the forward or inverse approach, establish the specific movement of the motor that enables each extremity of the robot itself.

Thereby, the algorithm for the solution of forward kinematics requires by parameters the position, in Cartesian coordinates, of the point where the end effector will be established in a workspace of the respective robot. Thus, it will be outputted as a result the amount of radial movement, in degrees, that each of the three actuators used must adopt, in order to ensure the requested position. On the other hand, for the solution of the inverse kinematics, the parameters entered into the algorithm must be the values assigned to each motor, to deduce the position, in Cartesian coordinates, of the point that the end effector of the delta robot will reach.

It should be highlighted the need for the values of the respective geometric parameters to be incorporated into the algorithm for the solution of the given kinematic models. For this case, such values are included as constants in the algorithm itself. Hence, the algorithm was programmed using Matlab software, through generating a script for the solution of each proposed kinematic model.

The software uses each generated files as a function, which the user can call it, if it is accompanied by the indicated parameters. Since otherwise, such functions could emit an error on the way of their use.

Development of the graphical interface

Through a properly constituted user panel, whose elements are distributed and identified in the graphical environment provided, the entry of the parameters for the required calculation process, the selection of the operation to be performed and the subsequent appreciation of the results obtained, are facilitated. It is by means of this type of synoptic that the exposed values usually provide a better visual appreciation than in the case of their simile in text mode. Consequently, from a GUI (Graphical User Interface) file of the Matlab software, the environment for the solution of the kinematic models of the analyzed delta robot was developed. The created screen is presented in figure 2 and its functions are described below.

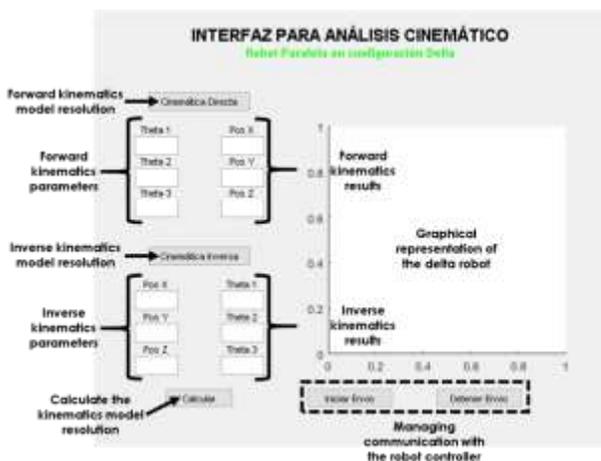


Figure 2 Graphical interface for kinematic analysis of the delta robot

Source: Own Elaboration, 2021

The interface is made up of the following sections:

1. *Forward kinematics*. It allows the input of the value of each joint variable for calculation of the coordinates (X, Y, Z) of the position of the robot's end effector.
2. *Inverse kinematics*. It admits the Cartesian coordinates of the position that the robot's end effector will adopt, for the deduction of the respective values of the joint variables.
3. *Calculate*. By pressing this button, and having previously done so with the one corresponding to the model to be calculated, the results of the Cartesian coordinates of the end effector position or the respective joint variables of each motor are output in the specified places, according to the data entered by the user.
4. *Graphical representation of the delta robot*. It provides a visualization of the location of the delta robot elements, according to the specified geometric parameters and the location that the end effector must reach, after the execution of a calculation process.

It is worth mentioning that, to ensure the operation of the integrated graphical interface, the algorithms of the previously developed kinematic models are called from its programming. In addition, the said interface have been configured to run only one kinematic model solution at a time. For which, the parameters of the model to be solved must be entered, the associated button must be pressed and pressed the Calculate button either. So, if, after solving a specific kinematic model, the user decides to execute another calculation, the results obtained from the first execution will be eliminated, to present the most recent values.

Implementation of joint control

Since the servomotors used to move each robot extremity operate exclusively by assigning integer values between 0° and 180° , a setting was required in the joint values provided by the graphical interface. Such joint coupling process was incorporated into the programming of the interface itself, in order to generate data that the robot controller could interpret properly, and then transmit the voltage signal associated with the correspondent motor. Therefore, in turn, the functions for data communication between both described systems were added to the same interface, as shown in figure 2 itself.

The interaction between the computer and the robot controller, in this case an Arduino UNO board, is ensured by enabling the USB port that allows data transfer between both of them, from the interface used. Hence, once the joint values have been designated, through the graphical interface, and then transmitted to the controller, the location that the end effector of the robot must reach will be manipulated.

So, The voltage signal sent to each motor, from the controller, may then correspond both to the values entered by the user when solving the forward kinematic model, and to the results obtained from the solution of the inverse kinematic model.

Algorithms operation results

The tests to validate the operation of the algorithms developed were executed from the proposal of different values, for the management of the delta robot used. In this way, it was possible to verify the effectiveness in the programming of each kinematic model described, by obtaining a solution associated with the entered data. In this particular case, both the call of the function to be used, the input of the required parameters and the issuance of the respective results, were executed through the Matlab Command Window. Next, the sequence performed for the verification of both algorithms is stated, as well as the results established during the execution of a test with specific values.

1. The function calling to solve the forward kinematic model, using the values given to each motor of the delta robot as parameters. In Figure 3, the entry of the values 45° , -45° and 5° , respectively, is shown.
2. The exposition of the solution of the forward model, in the Cartesian coordinates, relative to the location of the end effector of the robot, for the joint values proposed. In Figure 3, the results for the values assigned in the previous step are observed.

```
>> [x,y,z]=delta_FwdKin(45,-45,5)

x =

    11.8292

y =

    18.3279

z =

   -21.3006
```

Figure 3 Parameters and results of forward kinematic model in Matlab software

Source: Own Elaboration, 2021

3. The function calling to solve the inverse kinematic model, whose parameters correspond to the data obtained in the preceding step, as shown in Figure 4.
4. The emission of the solution results for the inverse kinematic model, respective to the joint values of each robot motor. Thus, the results shown in figure 4 coincide with those entered for the forward kinematics function.

```
>> [q1,q2,q3]=delta_InvKin(11.8292,18.3279,-21.3006)

q1 =

    44.9999

q2 =

   -45.0001

q3 =

    4.9997
```

Figure 4 Parameters and results of inverse kinematic model in Matlab software

Source: Own Elaboration, 2021

Graphical interface operation results

In a similar way to how the operation of the algorithms previously formulated in Matlab were corroborated, it was also necessary to verify the effectiveness of the graphical interface function developed from the GUI environment. Consequently, the parameters used to validate the functionality of the constituted programming were used again, only this time, starting from their entry in the respective spaces within the integrated graphical interface. The process performed for such validation is detailed below.

1. For the solution of the forward kinematic model, the proposed values for the joint variables are entered in the spaces assigned within the graphical interface. In this case, 45° , -45° and 5° were used respectively, as shown in figure 5.
2. Press the button associated with the application of the forward kinematic model, and then press the Calculate button to output the Cartesian coordinates corresponding to the location of the end effector of the delta robot, as shown in figure 5.

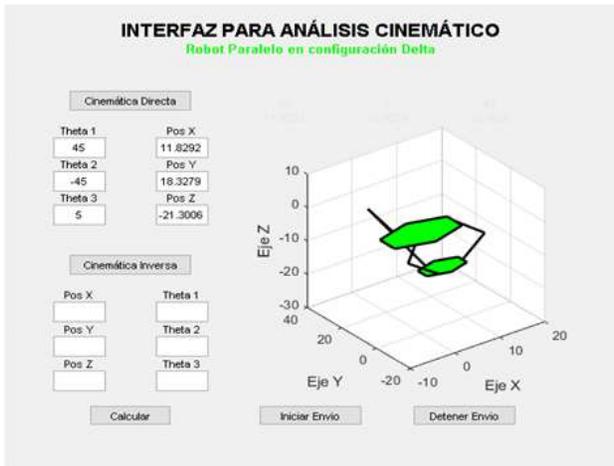


Figure 5 Parameters and results of the graphical interface for solution of forward kinematic model
Source: Own Elaboration, 2021

3. For the solution of the inverse kinematic model, the values obtained from the previously executed calculation process are placed in the spaces assigned for the application of the respective model, as shown in figure 6.
4. Press the button corresponding to the application of the inverse kinematic model, and again the Calculate button, to return the joint values that the delta robot motors must perform, as shown in figure 6.

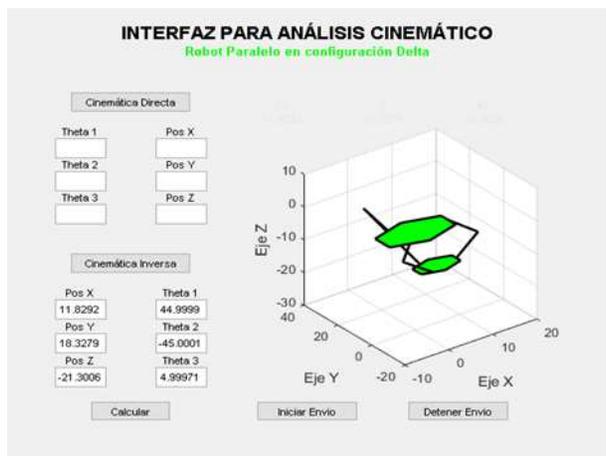


Figure 6 Parameters and results of the graphical interface for solution of inverse kinematic model
Source: Own Elaboration, 2021

In this way, it can be verified the results issued during the execution of the tests with the algorithms for the solution of the forward and inverse kinematic models. Hence, the values shown in figures 3 and 4, respectively, can also be obtained from the use of the graphical interface developed, as corroborated in figures 5 and 6.

In addition, the graphical representation of the delta robot's condition is of great visual support for the user, once the calculation of the proposed kinematic models has been completed. Through which, the adoption of the respective location by the end effector is displayed.

It is worth mentioning that, up to this point in the analysis carried out, the position of the end effector has been considered at the center of the mobile platform of the robot used, as indicated in the literature. Such reference position shares the same plane and is equidistant to all the joints between the mobile platform itself and the driven link of each end of the robot. However, if later it is desired to grant an application to the analyzed robot, it will be necessary to adapt the programmed algorithms so that they consider the position of the end effector in the center of the gripper opening, as shown in the figure 7.

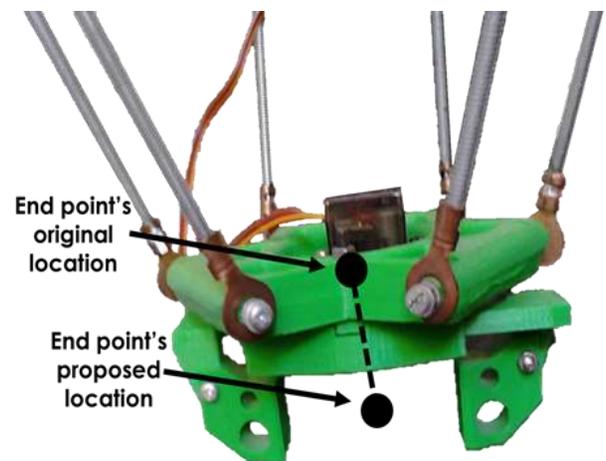


Figure 7 Reference point for end effector
Source: Own Elaboration, 2021

End effector localization results

To perform the tests of the real location of the end effector of the robot used, sets of positions deduced from the solution of both kinematic models analyzed were evaluated. However, since the position given for each servomotor can only be indicated in integer degrees, the request for a specific movement in distance units implies that the actual position reached by the actuator will differ with respect to the one requested. It is precisely from the analysis of the discrepancy between the desired and actual position of the end effector, that the effect caused by the use of servomotors as robot actuators is verified.

In the first case, the solution of the forward kinematic model is evaluated; for which, joint values: -10.30° , 14.97° and 38.45° are assigned through the developed graphical interface, as shown in figure 8a. However, the set values are sent to each servomotor rounding the proposed ones towards the nearest integers, that is, -10° , 15° and 38° , correspondingly; whose actual positioning is shown in figure 9a. Likewise, a comparison is made between the results obtained from the proposed values and those actually sent to the motors, in Cartesian and spherical coordinates, in tables 2 and 3, respectively.

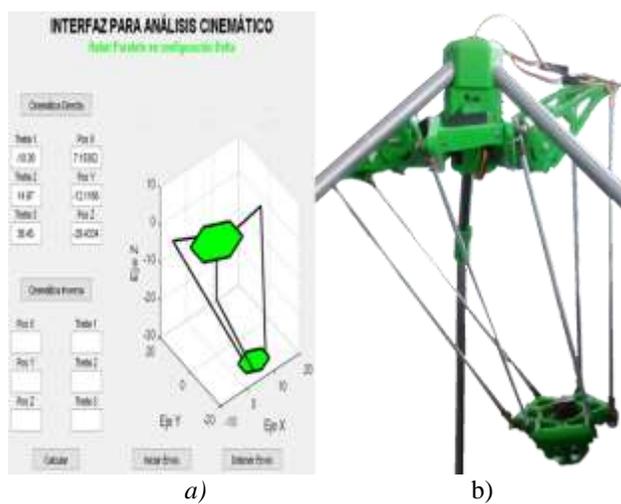


Figure 8 Test result of forward kinematic model in: a) the graphical interface and b) the delta robot
Source: Own Elaboration, 2021

Component	Desired value (cm)	Real value (cm)	Difference (cm)
X coordinate	7.154	7.010	0.143
Y coordinate	-12.117	-11.967	0.142
Z coordinate	-29.433	-29.524	0.091

Table 2 Test values of forward kinematic model, in Cartesian coordinates
Source: Own Elaboration, 2021

Component	Desired value	Real value	Difference
Magnitude r	32.624 cm	32.623 cm	0.001 cm
Angle θ	-59.442°	-59.658°	0.216°
Angle φ	154.449°	154.826°	0.377°

Table 3 Test values of forward kinematic model, in spherical coordinates
Source: Own Elaboration, 2021

In a second case, the results from the solution of the inverse kinematic model are evaluated.

So, the coordinates: 7.67, 12.89 and -25.27, in centimeters, for the spatial location of the end effector, were entered into the graphical interface; and joint values: 24.0684° , -35.0848° and 5.2276° were obtained, as shown in figure 9a. While to the servomotors were sent 24° , -35° and 5° , respectively, according is shown in figure 9b. The results of the proposed values and the data transferred to the robot controller are verified, in tables 4 and 5, corresponding to Cartesian and spherical coordinates.

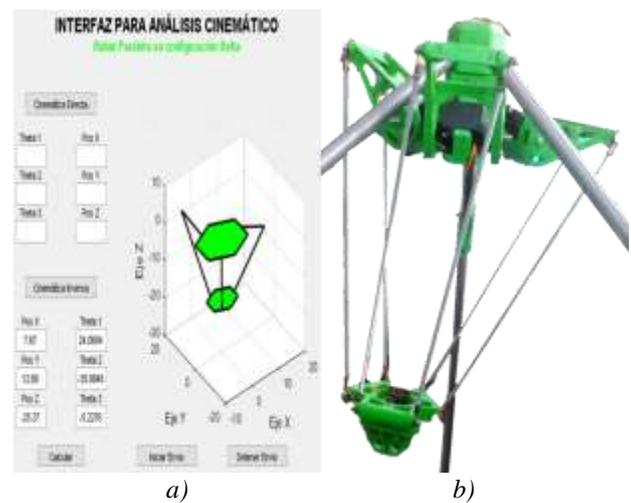


Figure 9 Test result of inverse kinematic model in: a) the graphical interface and b) the delta robot
Source: Own Elaboration, 2021

Component	Desired value (cm)	Real value (cm)	Difference (cm)
X coordinate	7.670	7.709	0.039
Y coordinate	12.890	12.836	0.054
Z coordinate	-25.370	-25.397	0.027

Table 4 Test values of inverse kinematic model, in Cartesian coordinates
Source: Own Elaboration, 2021

Component	Desired value	Real value	Difference
Magnitude r	29.472cm	29.482cm	0.010cm
Angle θ	59.246°	59.012°	0.234°
Angle φ	149.407°	149.478°	0.071°

Table 5 Test values of inverse kinematic model, in spherical coordinates
Source: Own Elaboration, 2021

The results reported from the execution of the tests with each analyzed kinematic model show an average discrepancy of 0.083 cm in the real location of the end effector with respect to the values obtained from the calculation process, by axis, when representing them in Cartesian coordinates.

So, it is verified that, despite that the movement performed by each servomotor is not exact, in reference to the results of the calculation, the approximation provided by rounding these values toward the nearest integer does not determine a large error. Of course, depending on the subsequent use given to the robot, the proven error could be ignored or limit the execution of a specific function.

Likewise, the validation of results by converting Cartesian to spherical coordinates was included. Using the latter, the absolute magnitude that relates the reference system, settled on the fixed platform of the delta robot, with the location of the end effector is examined. In this case, the average difference recorded corresponds to 0.006 cm, while in the case of the angles that complete the spatial location of the end effector, a difference of less than 0.500° is established for both cases studied. This analysis corroborates the close proximity between the results of the model and the values adopted, from a different perspective than the previous case.

Conclusions

A particularity of parallel robots and specifically, of the delta type, is the analysis of kinematic models of greater abstraction to provide a solution to the location of their end effector, than in the case of their serial similes. Thus, from the study of the respective forward and inverse kinematic models, the programming of algorithms was developed that would concentrate and ensure the functionality of both. The functionality of these algorithms was verified by requesting different locations of the robot end effector, both by proposing the Cartesian coordinates to reach and by designating a specific movement in its actuators.

On the other hand, the integration of the graphical interface for the kinematic analysis of the delta robot, allowed to provide an improved visual aspect both to the data required for the execution of a particular model, as well as to the results derived from such process. Then, such results would be sent to the robot controller itself, for the management of the associated electrical signals. The real functionality of the implemented system was ensured through Matlab software in communication with Arduino hardware. It should be noted that Arduino electronic board, despite being a simple to use and low-cost tool, presented high performance during the relevant tests.

ISSN-2444-5002

ECORFAN[®] All rights reserved

Regarding the results achieved, it stands out that although the actuators used have limited mobility, the location described by the end effector does not differ considerably from the data determined by the calculation carried out in each test performed. It is worth mentioning that the results obtained in the respective validation process were compared in Cartesian and spherical coordinates, in order to analyze the disparity between them, both by axis and globally. Thus, an average difference of slightly less than one millimeter is recorded, in the location of the end effector, by axis; while in the global space, the discrepancy is even smaller.

In this way, it is possible to verify that despite being a test prototype, the implement developed, and manipulated from a computer interface, provides an adequate procedure in the scope of specific positions or joint movements requested. Therefore, without a doubt, this is a motivation to continue the study, starting with the application of the robot used in the performance of a particular task. Likewise, it is highlighted the fact of having an increasingly sophisticated means of testing that allows Mechatronics Engineering students from Technological University of North Aguascalientes to understand the operation of a robot whose configuration is different from those typically valued in an Industrial Robotics course.

References

- Abele, E., Weigold, M., & Rothenbücher, S. (2007). Modeling and identification of an industrial robot for machining applications. *CIRP Annals*, 387-390.
- Angel, L., & Viola, J. (2016). Parametric identification of a delta type parallel robot. *2016 IEEE Colombian Conference on Robotics and Automation*, 1-6.
- Antal, T. A. (2018). Considerations on the serial pc-arduino uno R3 interaction, in java, using jdeveloper, for a 3r serial robot, based on the arduink library. *Acta Technica Napocensis-Series: Applied Mathematics, Mechanics and Engineering*, 7-10.
- Clavel, R. (1991). *Conception d'un robot parallèle rapide à 4 degrés de liberté (Tesis Doctoral)*. Lausanne: Ecole Polytechnique Federale de Lausanne.

- Codourey, A., Clavel, R., & Burckhardt, C. W. (1992). Control algorithm and controller for the direct drive delta robot. *Control Robot 1991*, 543-549.
- Hendriko, H., Nurkhamdi, J. J., & Imam, M. M. (2018). Analytical based inverse kinematics method for 5-axis delta robot. *International Journal of Materials, Mechanics and Manufacturing*, 264-267.
- Huang, X., Weng, F., Wei, Z., & Kamruzzaman, M. M. (2021). Application and research of robot sorting system based on LabVIEW. *Journal of Intelligent & Fuzzy Systems*, 1-8.
- Idumudi, V. S., & Javed, A. (2021). Kinematic error modeling of a parallelogram arm of the delta robot and its dimensional optimization. *Advances in Industrial Machines and Mechanisms*, 137-147.
- Khorasani, A., Gholami, S., & Taghirad, H. D. (2015). Optimization of KNTU delta robot for pick and place application. *2015 3rd RSI International Conference on Robotics and Mechatronics (ICROM)*, 127-132.
- Kuo, Y. L. (2016). Mathematical modeling and analysis of the delta robot with flexible links. *Computers & Mathematics with Applications*, 1973-1989.
- Laribi, M. A., Romdhane, L., & Zegloul, S. (2007). Analysis and dimensional synthesis of the delta robot for a prescribed workspace. *Mechanism and Machine Theory*, 859-870.
- López, M., Castillo, E., García, G., & Bashir, A. (2006). Delta robot: inverse, direct, and intermediate Jacobians. *Journal of Mechanical Engineering Science*, 103-109.
- Maya, M., Castillo, E., Lomelí, A., González-Galván, E., & Cárdenas, A. (2013). Workspace and payload-capacity of a new reconfigurable delta parallel robot. *International Journal of Advanced Robotic Systems*, 56-66.
- Peña-Cortés, C. A., Martínez-Oviedo, E., & Cárdenas-Herrera, P. F. (2011). Optimización dimensional de un robot paralelo tipo delta basado en el menor consumo de energía. *Ciencia e Ingeniería Neogranadina*, 73-88.
- Pierrot, F., Reynaud, C., & Fournier, A. (1990). DELTA: A simple and efficient parallel robot. *Robotica*, 105-109.
- Poppeova, V., Uricek, J., Rejda, R., & Bulej, V. (2012). The design and simulation of training delta robot. *Journal of Trends in the Development of Machinery and Associated Technology*, 195-198.
- Rivas, D. R., Galarza, E. E., Tumbaco, D. C., Quimbita, W. E., & Galarza, O. V. (2015). Delta robot controlled by robotic operating system. *ITECKNE*, 54-59.
- Sánchez, R. E., Castillo, E., González, J. J., & Balmaceda, A. L. (2015). Análisis del desempeño cinetostático de un robot paralelo tipo delta reconfigurable. *Ingeniería Investigación y Tecnología*, 213-224.
- Villegas, J. M., Avizzano, C. A., Ruffaldi, E., & Bergamasco, M. (2015). A low cost open-controller for interactive robotic system. *2015 IEEE European Modelling Symposium (EMS)*, 462-468.
- Vischer, P., & Clavel, R. (1998). Kinematic calibration of the parallel delta robot. *Robotica*, 207-218.