Received March 20, 2021, accepted March 29, 2021, date of publication April 15, 2021, date of current version May 14, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3073402

Modeling in Two Configurations of a 5R 2-DoF Planar Parallel Mechanism and Solution to the Inverse Kinematic Modeling Using Artificial Neural Network

EUSEBIO JIMÉNEZ LÓPEZ[1], DANIEL SERVÍN DE LA MORA-PULIDO[2], RAÚL SERVÍN DE LA MORA-PULIDO[2], FRANCISCO JAVIER OCHOA-ESTRELLA[2], MARIO ACOSTA FLORES[3], AND GABRIEL LUNA-SANDOVAL[4]

[1]CIAAM, Universidad Tecnológica del Sur de Sonora-ULSA Noroeste-IIMM, Ciudad Obregón 85190, México
[2]CETA, Instituto Tecnológico Superior de Cajeme, Ciudad Obregón 85024, México
[3]Faculty of Chemical Sciences and Engineering, UAEM, Cuernavaca 62209, México
[4]Industrial Engineering Department, Universidad Estatal de Sonora, San Luis Río Colorado 83470, México

Corresponding authors: Eusebio Jiménez López (ejimenezl@msn.com) and Mario Acosta Flores (marioacosta8@yahoo.com.mx)

ABSTRACT This article introduces a new kinematic modeling method used to analyze coupled rigid multi-body movements. The method was applied to the study of a 5R planar parallel mechanism's kinematics and consists of analyzing two fixed configurations of the mechanism to systematize the rotational relationships between the two structures. Mathematical models were developed using complex numbers. The inverse kinematic problem was modeled as a system of eight nonlinear equations and eight unknowns, which was solved with Newton-Raphson's method. Subsequently, with the inverse problem model, a numerical database related to the mechanism configurations, including singular positions, was generated to train a multilayer neural network. The Levenberg-Marquardt algorithm was used for network training. Finally, an interpolated linear path was used to understand the efficiency of the trained network.

INDEX TERMS Parallel robots, artificial neural networks, complex numbers, kinematics, Newton–Raphson.

I. INTRODUCTION

Parallel manipulators have been studied by several researchers over the past two decades, because they have competitive advantages over open-chain robots, for example, greater accuracy, increased load capacity, more rigidity, among others [1]. These features are essential in industrial applications, such as simulators, machine tools, and CNCs, among others [2]. Due to their special characteristics, parallel robots are currently studied by various authors [3]–[8].

Parallel robots are used in several applications where high levels of accuracy and precision are required: for example, in spray paint operations [9] and machining operations [10]. To gain control of robots, the kinematic models that govern their movements are needed. These models are classified into inverse kinematics and direct kinematics. Various mathematical tools have been used to model the kinematics of parallel robots, such as homogeneous matrices [11], complex numbers [12], [13], and Quaternions [14], and methods such as Denavit-Hartenberg [15]. However, the use of these tools and methods involves a high mathematical calculation. This increased computational cost emphasizes the need to seek new alternative ways to solve a parallel robot's direct and reverse kinematic problems.

In this sense, Artificial Neural Networks (ANN) have been used to solve the inverse problem of open-chain robots, parallel robots, and other models such as the synthesis of mechanisms [16]–[19]. The use of neural networks is mainly due to the fact that the generated processing models are nonlinear. There are several methods for calculating the inverse kinematics of robots, such as geometric, algebraic, and iterative methods [20]. Artificial neural networks represent another alternative to solve the inverse problem of serial and parallel robots [21]. For example, in [22] neural networks are used to generate the workspace of a 2-DOF parallel mechanism by taking information from the solution of the reverse kinematic problem. In [23] three types of neural networks (multilayer perceptron, radial basis function, and Local Linear Model

The associate editor coordinating the review of this manuscript and approving it for publication was Yongming Li.

Trees) are used to solve inverse and direct kinematics, formulated from nonlinear mathematical models related to an Under-Constrained Cable-driven Parallel Robot.

Some combined artificial intelligence techniques are used to solve the inverse or direct problems of robots. For example, in [24] neural networks and genetic algorithms were used to design a Stewart platform. These techniques and methods solve issues such as maximum workspace and direct kinematics. A fast analytical method and a deep learning approach model were used in [25] to solve an industrial parallel robot's reverse problem. The analytical method was compared with three models of neural networks. This process decreases time for calculation and processing compared to traditional methods.

In [26] the inverse kinematics problem of a 3-PSS parallel manipulator is solved using several Machine learning approaches (Multiple Linear Regression, Multi-Variate Polynomial Regression, Support Vector Machine, Decision Tree Regression, and Random Forest Regression). The Machine Learning method, according to the authors, is easy to implement and requires fewer calculations.

Hybrid methods are also applied for the study of the kinematic problems of parallel robots. For example, in [27] a hybrid strategy consisting of a neural network and a numerical method is presented to solve the direct kinematics formulated from non-linear models of a tensegrity mechanism. In [28] a hybrid algorithm composed of neural networks and the Newton-Raphson method is presented, which solves a direct kinematic problem of a 2RPU-2SPR parallel manipulator. A mixed strategy was used in [29] to solve the immediate problem of a Gough-Stewart platform. Neural networks were used to solve the immediate problem, and the results were combined with the Newton-Raphson method.

To provide data for the training of a neural network, it is necessary to model the movements of robots or mechanisms to generate mathematical models through which it is possible to formulate inverse and direct kinematic problems. Traditionally these models are built using only one analysis configuration [27]–[31], limiting the motion relationships between the links that make up a robot. It is necessary to propose new modeling methods that consider two analysis configurations to systematize the rotations or translations that the actuators will perform and generate the mathematical models to raise and solve the kinematic problems associated with robots. Typically, the resulting models develop nonlinear equations, so it is necessary to use numerical methods such as the Newton-Raphson method to solve them [32].

On the other hand, the 5R planar parallel mechanism has been studied by various authors [33–41] as this mechanism has some advantages such as: 1) it has a simple structure, 2) it has a large workspace, 3) it has good flexibility and lack of singularities in the accessible workspace, and 4) it is easy to control [33]. This robot has been studied in various applications of Artificial Intelligence. For example, in [42] a neural application based on an inverse kinematic model was developed to generate workspaces of a parallel 2-DOF mechanism. The application was oriented to the movement of antennas and telescopes. In [43] the dynamics of the parallel 2-DOF mechanism was modeled and a non-linear pair control was designed using artificial neural networks.

In [44], control of the type of neuro-slip was developed for path tracking of a 2-DOF Parallel Mechanism. The control consists of a feeding neural network combined with an error estimator to compensate for large friction uncertainties and external disturbances. In [45], a 2- DOF Parallel Mechanism was designed to transplant flower seedlings, whose links were optimized by an objective function and using genetic algorithms. In [46], a 2- DOF Parallel Mechanism and a micro bipod were studied where the inverse kinematics of the robot and its work area are modeled.

This work applies a new methodology for modeling the movements of a 5R planar parallel mechanism. Two fixed configurations of the tool and algebra of complex numbers [47] were considered to construct the kinematic equations that govern their movements. The reverse problem was formulated with these equations, which were solved by Newton-Raphson's method [32]. Subsequently, a numerical database with robot positions in the workspace and some unique positions were generated to train an artificial neural network that can be used as a model equivalent to the reverse model. A straight path and a grade 5 polynomial speed profile [48] were used to test the neural network's efficiency. The reverse problem obtained with the Newton-Raphson method was compared to those generated by the neural network.

II. COMPLEX NUMBERS

This section presents a brief theoretical framework related to complex numbers conceived as ordered pairs of real numbers. Two binary operations are defined on the set $\Re^2$, this is, $\oplus : \Re^2 \times \Re^2 \rightarrow \Re^2$ and $\otimes : \Re^2 \times \Re^2 \rightarrow \Re^2$ through which couples $(\Re^2, \oplus)$ and $(\Re^2, \otimes)$ form two groups, one additive and one multiplicative commutative, respectively. The triad $(\Re^2, \oplus, \otimes)$ is a commutative body [47]. Operations $\oplus : \Re^2 \times \Re^2 \rightarrow \Re^2$ and $\otimes : \Re^2 \times \Re^2 \rightarrow \Re^2$ are defined by:

$$\begin{aligned} &\text{i) } \{a, b\} \oplus \{\alpha, \beta\} = \{a + \alpha, b + \beta\} \\ &\text{ii) } \{a, b\} \otimes \{\alpha, \beta\} = \{a\alpha - b\beta, a\beta + b\alpha\}, \forall\{a, \\ &\qquad b\}, \{\alpha, \beta\} \in \Re^2 \end{aligned} \tag{1}$$

On the other hand, in $\Re^2$ a scalar product is defined $\bullet : \Re \times \Re^2 \rightarrow \Re^2$, as an internal product $< \bullet, \bullet >: \Re^2 \times \Re^2 \rightarrow \Re$ and a norm $|\bullet| : \Re^2 \rightarrow \Re$ and, therefore, the structure $(\Re^2, \oplus, \otimes, < \bullet, \bullet >, |\bullet|)$ is a normed vector space with an internal product called *the vector space of complex numbers*. The transformation $\rho : \Re^2 \rightarrow \Re^2$ defined by:

$$\rho\,(\mathrm{p,q}) = \frac{1}{\|\mathrm{p}\|} \bullet \mathrm{p} \otimes \mathrm{q}; \quad \mathrm{q} \in \Re^2 \quad \text{fixed,} \tag{2}$$

is linear and with positive determinant and is characterized as a rotation. The matrix associated with transformation (2) given the canonical basis $\mathrm{B} = [(1, 0), (0, 1)] \subset \Re^2$ is:

$$\mathbf{M}_{\rho(p,\bullet)} = \frac{1}{\|\mathrm{p}\|} \begin{bmatrix} \mathrm{a} & -\mathrm{b} \\ \mathrm{b} & \mathrm{a} \end{bmatrix}$$

On the other hand, p, q $\in \Re^2$ are two complex numbers with unitary norm. This is:

$$\begin{aligned} &1)\ \mathrm{p} = \{\mathrm{p}_0, \mathrm{p}_1\}; \quad \mathrm{p}_0^2 + \mathrm{p}_1^2 = 1 \\ &2)\ \mathrm{q} = \{\mathrm{q}_0, \mathrm{q}_1\}; \quad \mathrm{q}_0^2 + \mathrm{q}_1^2 = 1 \end{aligned} \tag{3}$$

Finally, the relationships between the components of the complexes p, q $\in \Re^2$ and the components of the rotations are as follows:

$$\begin{aligned} &1)\ \mathrm{p} = \{\mathrm{p}_0, \mathrm{p}_1\}; \quad && \mathrm{p}_0 \in \Re;\ \mathrm{p}_0 = \cos\theta_1 \\ & && \mathrm{p}_1 \in \Re;\ \mathrm{p}_1 = \pm\sin\theta_1 \\ &2)\ \mathrm{q} = \{\mathrm{q}_0, \mathrm{q}_1\}; \quad && \mathrm{q}_0 \in \Re;\ \mathrm{q}_0 = \cos\theta_2 \\ & && \mathrm{q}_1 \in \Re;\ \mathrm{q}_1 = \pm\sin\theta_2 \end{aligned} \tag{4}$$

Here, $\theta_1, \theta_2 \in \Re$ are angular displacements.

III. MODELING MECHANISM

This section presents the modeling of the 5R planar parallel mechanism, also known as 2 GDL flat parallel robot. The configuration of the mechanism is as shown in Figure 1 where moving position vectors and bases have been defined. The purpose of modeling in the reference configuration is to determine the coordinates of *the pot* point (see Figure 1) from point "O" located in the Cartesian system (x, y).

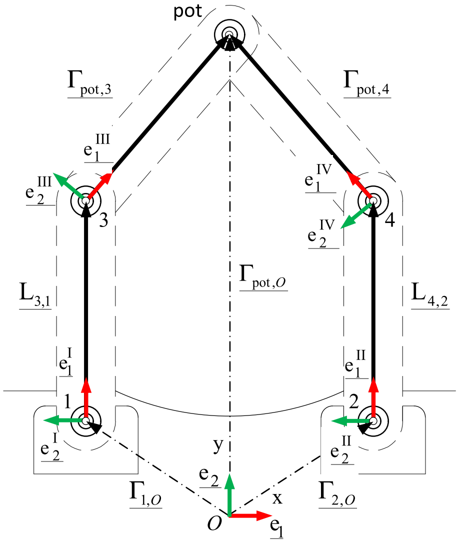

FIGURE 1. Configuration of the 5R planar parallel mechanism and bases in the initial or non-deformed configuration.

According to Figure 1 the coordinates of the *pot* point are determined by means of the following loop equations:

$$\begin{aligned} \underline{\Gamma_{pot,O}} &= \underline{\Gamma_{1,O}} \oplus \underline{L_{3,1}} \oplus \underline{L_{pot,3}} \\ \underline{\Gamma_{pot,O}} &= \underline{\Gamma_{2,O}} \oplus \underline{L_{4,2}} \oplus \underline{L_{pot,4}} \end{aligned} \tag{5}$$

Or, in equivalent form:

$$\begin{aligned} \underline{\Gamma_{pot,O}} &= \underline{\Gamma_{1,O}} \oplus L_{3,1} \bullet \underline{e_1^{I}} \oplus L_{pot,3} \bullet \underline{e_1^{III}} \\ \underline{\Gamma_{pot,O}} &= \underline{\Gamma_{2,O}} \oplus L_{4,2} \bullet \underline{e_1^{II}} \oplus L_{pot,4} \bullet \underline{e_1^{IV}} \end{aligned} \tag{6}$$

Now consider that inertial base rotations $\underline{e_j} = \{\underline{e_1}, \underline{e_2}\}$ on the mobile bases can be modeled using a linear transformation $\rho : \Re^2 \rightarrow \Re^2$. Such rotations are modeled, according to the expression (2), as follows:

$$\begin{aligned} &1)\ \underline{e_1^{I}} = \rho\left(\mathrm{p}, \underline{e_1}\right) = \frac{1}{\|\mathrm{p}\|} \bullet \mathrm{p} \otimes \underline{e_1} \\ &2)\ \underline{e_1^{III}} = \rho\left(\mathrm{r}, \underline{e_1}\right) = \frac{1}{\|\mathrm{r}\|} \bullet \mathrm{r} \otimes \underline{e_1} \\ &3)\ \underline{e_1^{II}} = \rho\left(\mathrm{q}, \underline{e_1}\right) = \frac{1}{\|\mathrm{q}\|} \bullet \mathrm{q} \otimes \underline{e_1} \\ &4)\ \underline{e_1^{IV}} = \rho\left(\mathrm{s}, \underline{e_1}\right) = \frac{1}{\|\mathrm{s}\|} \bullet \mathrm{s} \otimes \underline{e_1} \end{aligned} \tag{7}$$

Here, p, q, r, s $\in \Re^2$, are complex numbers of unit norm [44]. Expressions (6) are written in terms of the fixed inertial base and complex of the unit norm; that is, based on the expressions (7):

$$\begin{aligned} \underline{\Gamma_{pot,O}} &= \underline{\Gamma_{1,O}} \oplus L_{3,1} \bullet \left\{\mathrm{p} \otimes \underline{e_1}\right\} \oplus L_{pot,3} \bullet \left\{\mathrm{r} \otimes \underline{e_1}\right\} \\ \underline{\Gamma_{pot,O}} &= \underline{\Gamma_{2,O}} \oplus L_{4,2} \bullet \left\{\mathrm{q} \otimes \underline{e_1}\right\} \oplus L_{pot,4} \bullet \left\{\mathrm{s} \otimes \underline{e_1}\right\} \end{aligned} \tag{8}$$

Using the operations $\oplus : \Re^2 \times \Re^2 \rightarrow \Re^2$ and $\otimes : \Re^2 \times \Re^2 \rightarrow \Re^2$ defined in (1), the following expressions are obtained:

$$\begin{aligned} &1)\ \Gamma_{pot,O,x} = \Gamma_{1,O,x} + L_{3,1}\mathrm{p}_0 + L_{pot,3}\mathrm{r}_0 \\ &2)\ \Gamma_{pot,O,y} = \Gamma_{1,O,y} + L_{3,1}\mathrm{p}_1 + L_{pot,3}\mathrm{r}_1 \\ &3)\ \Gamma_{pot,O,x} = \Gamma_{2,O,x} + L_{4,2}\mathrm{q}_0 + L_{pot,4}\mathrm{s}_0 \\ &4)\ \Gamma_{pot,O,y} = \Gamma_{2,O,y} + L_{4,2}\mathrm{q}_1 + L_{pot,4}\mathrm{s}_1 \end{aligned} \tag{9}$$

The unit norms associated with complex numbers are:

$$\begin{aligned} \mathrm{p}_0^2 + \mathrm{p}_1^2 = 1; \quad \mathrm{r}_0^2 + \mathrm{r}_1^2 = 1 \\ \mathrm{q}_0^2 + \mathrm{q}_1^2 = 1; \quad \mathrm{s}_0^2 + \mathrm{s}_1^2 = 1 \end{aligned} \tag{10}$$

The geometric relationships defined between the components of the rotations and the parameters associated with the complex numbers p, q, r, s $\in \Re^2$, according to relationships (4), are as follows:

$$\begin{aligned} \mathrm{p} = \left(\mathrm{p}_0, \mathrm{p}_1\right); \quad & \mathrm{p}_0 = \cos\theta_1,\ \mathrm{p}_1 = \pm\sin\theta_1 \\ \mathrm{q} = \left(\mathrm{q}_0, \mathrm{q}_1\right); \quad & \mathrm{q}_0 = \cos\theta_2,\ \mathrm{q}_1 = \pm\sin\theta_2 \\ \mathrm{r} = \left(\mathrm{r}_0, \mathrm{r}_1\right); \quad & \mathrm{r}_0 = \cos\theta_3,\ \mathrm{r}_1 = \pm\sin\theta_3 \\ \mathrm{s} = \left(\mathrm{s}_0, \mathrm{s}_1\right); \quad & \mathrm{s}_0 = \cos\theta_4,\ \mathrm{s}_1 = \pm\sin\theta_4 \end{aligned} \tag{11}$$

Here, $\theta_1, \theta_2, \theta_3, \theta_4 \in \Re$ are the angular displacements related to the positions of the links and the Cartesian axis x, as shown in Figure 3.

The inverse kinematic problem is formulated below:

Given $\underline{\Gamma_{pot,O}}, \underline{\Gamma_{1,O}}, \underline{\Gamma_{2,O}} \in \Re^2$, $L_{3,1}, L_{4,2}, L_{pot,3}, L_{pot,4} \in \Re^+$, find $\mathrm{p} = \{\mathrm{p}_0, \mathrm{p}_1\}, \mathrm{q} = \{\mathrm{q}_0, \mathrm{q}_1\}, \mathrm{r} = \{\mathrm{r}_0, \mathrm{r}_1\}, \mathrm{s} = \{\mathrm{s}_0, \mathrm{s}_1\}$ such that expressions (9) and (10) are satisfied.

The problem of inverse kinematics in the non-deformed configuration generates a system of eight equations with eight linear algebraic unknowns.

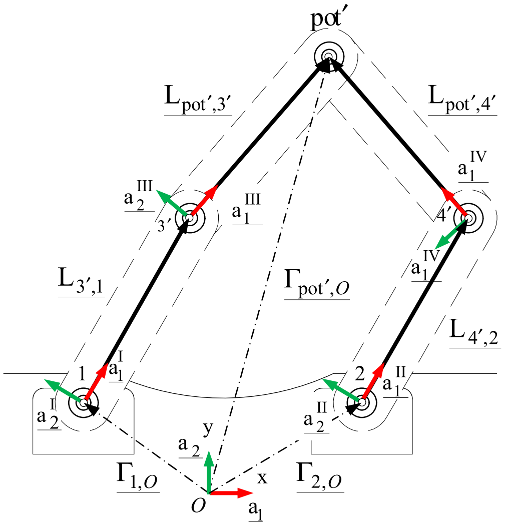

FIGURE 2. New bases in the final configuration.

A. MODELING IN THE FINAL CONFIGURATION

This section models the deformed or final position of the 5R planar parallel mechanism. Figure 2 shows the final configuration and its associated mobile bases.

According to Figure 2, the new coordinates of the *pot* point are obtained by the following expressions:

$$\begin{aligned}\underline{\Gamma_{pot',O}} &= \underline{\Gamma_{1,O}} \oplus \underline{L_{3',1}} \oplus \underline{L_{pot',3'}}\\ \underline{\Gamma_{pot',O}} &= \underline{\Gamma_{2,O}} \oplus \underline{L_{4',2}} \oplus \underline{L_{pot',4'}}\end{aligned} \tag{12}$$

Expressions (12) can be written in terms of the new bases shown in Figure 2, as follows:

$$\begin{aligned}\underline{\Gamma_{pot',O}} &= \underline{\Gamma_{1,O}} \oplus L_{3',1} \bullet \underline{a_1^{I}} \oplus L_{pot',3'} \bullet \underline{a_1^{III}}\\ \underline{\Gamma_{pot',O}} &= \underline{\Gamma_{2,O}} \oplus L_{4',2} \bullet \underline{a_1^{II}} \oplus L_{pot',4'} \bullet \underline{a_1^{IV}}\end{aligned} \tag{13}$$

The rotational relationships between the initial configuration bases and the new bases of the final configuration (see Figures 1 and 2) can be represented in terms of unit complex numbers as follows:

$$\begin{aligned}\underline{a_1^{I}} &= \rho\left(\mathrm{P},\underline{e_1^{I}}\right) = \rho\left(\mathrm{P}, \rho\left(\mathrm{p}, \underline{e_1}\right)\right) = \frac{1}{\|\mathrm{P}\|}\frac{1}{\|\mathrm{p}\|} \bullet \mathrm{P} \otimes \mathrm{p} \otimes \underline{e_1}\\ &= \mathrm{P} \otimes \mathrm{p} \otimes \underline{e_1}\\ \underline{a_1^{III}} &= \rho\left(\mathrm{R},\underline{e_1^{III}}\right) = \rho\left(\mathrm{R}, \rho\left(\mathrm{r}, \underline{e_1}\right)\right) = \frac{1}{\|\mathrm{R}\|}\frac{1}{\|\mathrm{r}\|} \bullet \mathrm{R} \otimes \mathrm{r} \otimes \underline{e_1}\\ &= \mathrm{R} \otimes \mathrm{r} \otimes \underline{e_1}\\ \underline{a_1^{II}} &= \rho\left(\mathrm{Q},\underline{e_1^{II}}\right) = \rho\left(\mathrm{Q}, \rho\left(\mathrm{q}, \underline{e_1}\right)\right) = \frac{1}{\|\mathrm{Q}\|}\frac{1}{\|\mathrm{q}\|} \bullet \mathrm{Q} \otimes \mathrm{q} \otimes \underline{e_1}\\ &= \mathrm{Q} \otimes \mathrm{q} \otimes \underline{e_1}\\ \underline{a_1^{IV}} &= \rho\left(\mathrm{S},\underline{e_1^{IV}}\right) = \rho\left(\mathrm{S}, \rho\left(\mathrm{s}, \underline{e_1}\right)\right) = \frac{1}{\|\mathrm{S}\|}\frac{1}{\|\mathrm{s}\|} \bullet \mathrm{S} \otimes \mathrm{s} \otimes \underline{e_1}\\ &= \mathrm{S} \otimes \mathrm{s} \otimes \underline{e_1}\end{aligned} \tag{14}$$

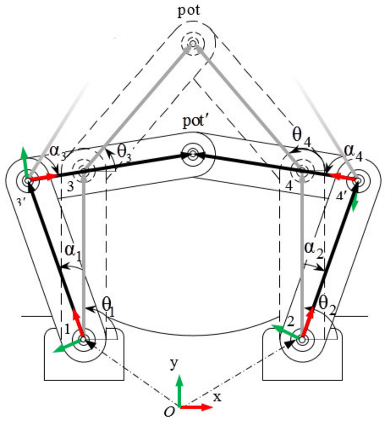

FIGURE 3. Angular offsets in the start and end configuration.

Therefore, loop equations (13) are written in terms of unit complexes. This is:

$$\begin{aligned}&\underline{\Gamma_{pot',O}}\\ &= \underline{\Gamma_{1,O}} \oplus L_{3',1} \bullet \left\{\mathrm{P} \otimes \mathrm{p} \otimes \underline{e_1}\right\} \oplus L_{pot',3'} \bullet \left\{\mathrm{R} \otimes \mathrm{r} \otimes \underline{e_1}\right\}\\ &\underline{\Gamma_{pot',O}}\\ &= \underline{\Gamma_{2,O}} \oplus L_{4',2} \bullet \left\{\mathrm{Q} \otimes \mathrm{q} \otimes \underline{e_1}\right\} \oplus L_{pot',4'} \bullet \left\{\mathrm{S} \otimes \mathrm{s} \otimes \underline{e_1}\right\}\end{aligned} \tag{15}$$

Here, P, Q, R, S $\in \Re^2$ are complex numbers related to the deformed or final configuration. The explicit form of expressions (15) are as follows:

$$\begin{aligned}\Gamma_{pot',O,x} &= \Gamma_{1,O,x} + L_{3',1}\left(P_0p_0 - P_1p_1\right)\\ &\quad + L_{pot',3'}\left(R_0r_0 - R_1r_1\right)\\ \Gamma_{pot',O,y} &= \Gamma_{1,O,y} + L_{3',1}\left(P_0p_1 + P_1p_0\right)\\ &\quad + L_{pot',3'}\left(R_0r_1 + R_1r_0\right)\\ \Gamma_{pot',O,x} &= \Gamma_{2,O,x} + L_{4',2}\left(Q_0q_0 - Q_1q_1\right)\\ &\quad + L_{pot',4'}\left(S_0s_0 - S_1s_1\right)\\ \Gamma_{pot',O,y} &= \Gamma_{2,O,y} + L_{4',2}\left(Q_0q_1 + Q_1q_0\right)\\ &\quad + L_{pot',4'}\left(S_0s_1 + S_1s_0\right)\end{aligned} \tag{16}$$

Unitary norm expressions are:

$$\begin{aligned}\|\mathrm{P}\| &= P_0^2 + P_1^2 = 1; \quad \|\mathrm{Q}\| = Q_0^2 + Q_1^2 = 1\\ \|\mathrm{R}\| &= R_0^2 + R_1^2 = 1; \quad \|\mathrm{S}\| = S_0^2 + S_1^2 = 1\end{aligned} \tag{17}$$

On the other hand, the geometric relationships defined between the components of the rotations and the parameters associated with the complex numbers of the deformed or final configuration are as follows:

$$\begin{aligned}\mathrm{P} &= \left(P_0, P_1\right); \quad P_0 = \cos\alpha_1;\ P_1 = \pm\sin\alpha_1\\ \mathrm{Q} &= \left(Q_0, Q_1\right); \quad Q_0 = \cos\alpha_2;\ Q_1 = \pm\sin\alpha_2\\ \mathrm{R} &= \left(R_0, R_1\right); \quad R_0 = \cos\alpha_3;\ R_1 = \pm\sin\alpha_3\\ \mathrm{S} &= \left(S_0, S_1\right); \quad S_0 = \cos\alpha_4;\ S_1 = \pm\sin\alpha_4\end{aligned} \tag{18}$$

Here, $\alpha_1, \alpha_2, \alpha_3, \alpha_4 \in \Re$ are the angular displacements defined between the initial configuration and the final configuration, as shown in Figure 3. The inverse kinematic problem is formulated as follows:

Given $\underline{\Gamma_{\mathrm{pot}',O}}, \underline{\Gamma_{1,O}}, \underline{\Gamma_{2,O}} \in \Re^2$, $L_{3,1}, L_{4,2}, L_{\mathrm{pot},3}, L_{\mathrm{pot},4} \in \Re^+$, $p = \{p_0, p_1\}$, $q = \{q_0, q_1\}$, $r = \{r_0, r_1\}$, $s = \{s_0, s_1\}$ with, $\|p\| = \|q\| = \|r\| = \|s\| = 1$, find: $P = \{P_0, P_1\}$, $Q = \{Q_0, Q_1\}$, $R = \{R_0, R_1\}$ y $S = \{S_0, S_1\}$ such that expressions (16) and (17) are satisfied.

The problem of inverse kinematics in the deformed configuration generates a system of eight equations with eight linear algebraic unknowns.

The new features presented in this work in the modeling part are summarized below: Two-configuration modeling allows a clear and correct analysis of the transition of movements and relationships between local and mobile bases. This modeling process is traditionally carried out in a single configuration presented in [33]–[38], [49] which prevents us from observing the physics of the movement and in particular the relationships between mobile systems or bases. The mathematical modeling of the 5R planar parallel mechanism generated by the systematization of complex numbers presented in [47] developed a nonlinear equation system. However, this nonlinearity does not affect the application of the modeling methodology presented in this work, as it can in fact be applied to model mechanical systems that move in space, since Quaternions [50] are a formal extension of complex numbers.

IV. TRAINING A NEURAL NETWORK

As mentioned above, the position kinematic models, both in the reference configuration and in the deformed configuration, are associated with systems of nonlinear equations of the polynomial type. These models require the use of iterative algorithms such as the Newton–Raphson method to provide numerical solutions to these problems. However, there are four disadvantages: 1) incorrect initial conditions, 2) no guarantee as to the correct solution before executing the algorithm convergence, 3) multiple solutions, and 4) no solution (if the Jacobian matrix is singular). Another disadvantage of numerical approaches is that they require heavy computational analysis and time. This article introduces an unconventional technique to solve the displacement problem associated with the 5R mechanism under study using artificial neural networks (ANN).

A. PROCEDURE FOR THE DESIGN OF NEURAL NETWORKS

This section systematically describes the procedure to obtain the inverse kinematic model related to the 5R mechanism using artificial neural networks. The neural network design process can be simple when you have the experience; however, 4 fundamental stages are distinguished when neural networks are used as a regression or classification technique:

1) A set of data must be obtained to identify the input and output parameters that the neural network will have to train.
2) The type and architecture of the neural network to train must be designated.
3) The algorithm and training of the neural network are selected.
4) Testing and validation of the performance of the trained neural network are performed.

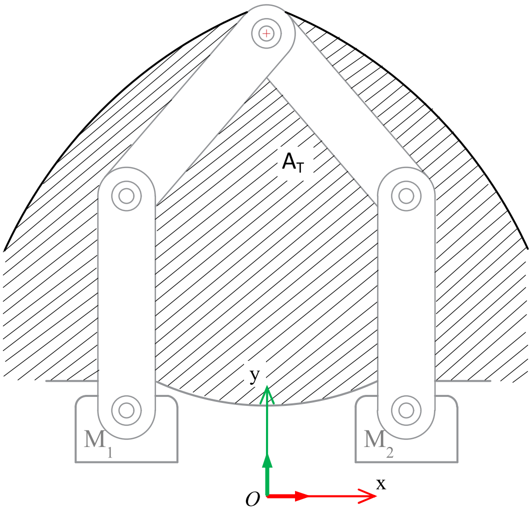

FIGURE 4. Operational workspace.

B. GENERATION OF TRAINING DATA SET TO SOLVE KINEMATICS

One of the most complicated stages in the process of designing a neural network is to obtain a set of input data since these play a fundamental role in the quality of the model obtained when training the neural network, making the design process a task that is not trivial and often requires experts and ingenuity to obtain it. In this sense, obtaining a data set to train the kinematics of robots associated with nonlinear models will depend on the physical characteristics of the same robot, its geometry, and the space in which it can work. The purpose of this section is to express a training data set as examples of appropriate network behavior made up of vector pairs as shown in (19), where $\mathbf{p}_q$ is the input of the network and $\mathbf{t}_q$ is the corresponding (target) output, according to the nomenclature used in [51].

$$\{\mathbf{p}_1, \mathbf{t}_1\}, \{\mathbf{p}_2, \mathbf{t}_2\}, \ldots, \{\mathbf{p}_Q, \mathbf{t}_Q\} \tag{19}$$

To train a network that learns the inverse kinematic problem, the position vector $\underline{\Gamma_{\mathrm{pot}',O}} \in \Re^2$ should be the network input and the parameters of the complexes $P, Q, R, S \in \Re^2$ will be the output. The geometry associated with the working area (A_T) of the 5R mechanism must be known and narrowed in its entirety or in a section of interest, as shown in Figure 4. Within the portion of the selected workspace, single points originating unique robot configurations were considered, to study the behavior of the network at these points. The study area should be discretized into samples, forming an array of network input examples:

$$\mathbf{P} = \{\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_Q\} \tag{20}$$

Here, $\mathbf{p} = \left[\Gamma_{\mathrm{pot}',O,\mathrm{x}}\ \Gamma_{\mathrm{pot}',O,\mathrm{y}}\right]^T \in Z^+$ and $Q \in Z^+$. Q is defined within the workspace and by a resolution parameter μ that symmetrically separates the plane (x, y) point to point. Figure 5 shows the workspace's discretization for a value of $\mu = 4$ and Q = 4968 (number of data).

To obtain the angular parameters, the formulated Newton–Raphson method was used, solving the inverse problem for each element at (20), thus forming the array of objective examples of the network:

$$\mathbf{T} = \{\mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_Q\} \tag{21}$$

being $\mathbf{t} = \left[\, \mathrm{P}_0\ \mathrm{P}_1\ \mathrm{Q}_0\ \mathrm{Q}_1\ \mathrm{R}_0\ \mathrm{R}_1\ \mathrm{S}_0\ \mathrm{S}_1 \,\right]^{\mathrm{T}}$ and $Q \in \mathrm{Z}^+$

The explicit form of the set of examples (the training set) to train a neural network with the inverse kinematic problem is:

$$\left\{ \mathbf{p}_1 = \begin{bmatrix} \Gamma_{\mathrm{pot}',O,\mathrm{x}} \\ \Gamma_{\mathrm{pot}',O,\mathrm{y}} \end{bmatrix}_1, \mathbf{t}_1 = \begin{bmatrix} \mathrm{P}_0 \\ \mathrm{P}_1 \\ \mathrm{Q}_0 \\ \mathrm{Q}_1 \\ \mathrm{R}_0 \\ \mathrm{R}_1 \\ \mathrm{S}_0 \\ \mathrm{S}_1 \end{bmatrix}_1 \right\}, \quad \left\{ \mathbf{p}_2 = \begin{bmatrix} \Gamma_{\mathrm{pot}',O,\mathrm{x}} \\ \Gamma_{\mathrm{pot}',O,\mathrm{y}} \end{bmatrix}_2, \mathbf{t}_2 = \begin{bmatrix} \mathrm{P}_0 \\ \mathrm{P}_1 \\ \mathrm{Q}_0 \\ \mathrm{Q}_1 \\ \mathrm{R}_0 \\ \mathrm{R}_1 \\ \mathrm{S}_0 \\ \mathrm{S}_1 \end{bmatrix}_2 \right\}, \quad \ldots, \left\{ \mathbf{p}_Q = \begin{bmatrix} \Gamma_{\mathrm{pot}',O,\mathrm{x}} \\ \Gamma_{\mathrm{pot}',O,\mathrm{y}} \end{bmatrix}_Q, \mathbf{t}_Q = \begin{bmatrix} \mathrm{P}_0 \\ \mathrm{P}_1 \\ \mathrm{Q}_0 \\ \mathrm{Q}_1 \\ \mathrm{R}_0 \\ \mathrm{R}_1 \\ \mathrm{S}_0 \\ \mathrm{S}_1 \end{bmatrix}_Q \right\} \tag{22}$$

Expression (22) forms a set of data that was used to train a neural network to solve the inverse kinematic problem. It is important to mention that the parameters of the complex P, Q, R, S $\in \Re^2$ take values in a range of -1 to 1, so it is necessary to normalize the vector $\Gamma_{\mathrm{pot}',O} \in \Re^2$ in the same range, which is commonly referred to as pre-processing and post-processing.

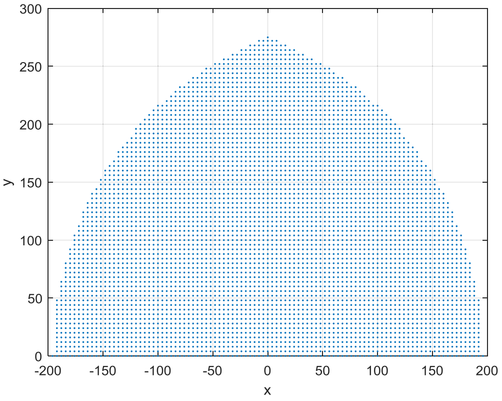

FIGURE 5. Discretized workspace.

C. NEURAL NETWORK ARCHITECTURE

The topology or architecture of a neural network consists of the organization and arrangement of neurons in the network. Neurons are grouped into layers, and the feedforward multilayer Artificial Neural Network (ANN) architecture is one of the most used due to the flexibility in the good results it delivers and its robustness in the presence of disturbances. This section proposes the structure of the artificial neural network to solve the inverse kinematics of the 5R planar parallel mechanism.

Choosing the correct neural network architecture for a regression or classification problem remains a difficult issue, and despite the existence of various proposals to solve it, there is no agreement on the strategy to follow to select the appropriate neural network architecture [52], [53]. Some authors have used genetic algorithms to generate the neural network topology for the study of the movements of a PUMA 560 robot [54].

For the network used in this work, the topology was selected by performing different tests with different configurations, and the one that generated the best results was taken. Figure 6 shows the proposed network architecture to solve the inverse kinematic problem related to the 5R planar parallel mechanism, and the focus of this study. It should be noted that this article does not focus on neural network optimization.

For multilayer networks, the output of one layer becomes the input of the next. The equation that describes this operation is a recursive equation represented as follows [51]:

$$\mathbf{a}^{m+1} = \mathbf{f}^{m+1}(\mathbf{W}^{m+1}\mathbf{a}^m + \mathbf{b}^{m+1}); \ \text{for } m = 0, 1, \ldots, M-1 \tag{23}$$

where $M \in \mathrm{Z}^+$ is the number of layers in the network, $\mathbf{W}$ and $\mathbf{b}$ contain the weights and biases, respectively, of the current layer, and $\mathbf{f}$ represents the transfer function. Neurons in the first layer receive external inputs as follows:

$$\mathbf{a}^0 = \mathbf{p}_q \tag{24}$$

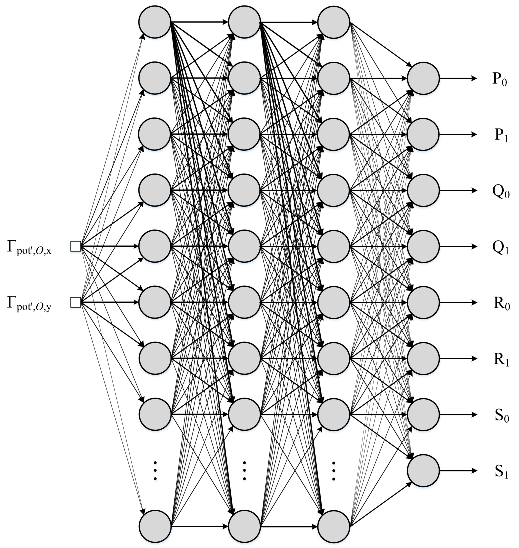

FIGURE 6. Network architecture for the inverse problem.

Equation (24) provides the starting point for (23). The neural network shown in Figure 6 consists of 2 inputs, three hidden layers with 15 neurons per layer, and an output layer of 8 neurons and 8 outputs, according to (22). This network has associated in the neurons of the hidden layer transfer functions (also called activation function) hyperbolic tangent sigmoid and in the neurons of the output layer linear transfer functions, and so, maintains the properties of nonlinearity and linearity between the input and output data of the network.

D. NEURAL NETWORK TRAINING

In multilayer feedforward neural networks there are several training algorithms available; all these algorithms use the gradient of the performance function to determine how to adjust weights and biases to obtain the best performance from the network. The default performance function for feedforward networks is the mean quadratic error (*mse*):

$$mse = \frac{1}{Q}\sum_{k=1}^{Q}\mathbf{e}(k)^2 = \frac{1}{Q}\sum_{k=1}^{Q}(\mathbf{t}(k) - \mathbf{a}(k))^2 \tag{25}$$

Here, **a** is a vector with network-generated outputs, **t**, which are the target outputs that the network should have in the k iteration. On the other hand, the gradient is determined using the technique called retro-propagation, which involves performing backward calculations through the network. This technique updates network weights and biases in the direction of performance that decreases faster. You can write an iteration of this algorithm as follows:

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k \mathbf{g}_k \tag{26}$$

Here, $\mathbf{x}_k$ represents the current vector of weights and biases, $\mathbf{g}_k$ is the current gradient, and α_k is the reason for learning. Among the algorithms available to train neural networks are several high-performance algorithms that can converge ten to one hundred times faster than other algorithms, such as the Levenberg–Marquardt algorithm which is considered the fastest method to train moderately sized feedforward neural networks (up to several hundred weights) [55]; it also has an efficient implementation in the MATLAB software®.

For neural network training, this work used the Levenberg–Marquardt algorithm; the data used are those described in (22). The dataset was not divided for testing and validation because the network is intended to fully learn all the data from the area (A_T) portion considered; the *target mse* that the network must achieve in its performance was set to $1e^{-6}$ and the maximum number of epochs was 3000.

The graph in Figure 7 shows the mean quadratic error for neural network training. The error starts at a large value and as the training times increase the error decreases to a small value, in other words, the network can be said to have learned. The training ended up reaching the proposed goal *mse,* the best performance is for a $9.98e^{-7}$ error.

This article presents novelties in neural networks since traditionally the training is done only by taking the draft angles and not the axis of rotation of each joint. The neural network handles the parameters of the inverse kinematic problem, i.e. 2 inputs (x, y) and 8 outputs (four angular displacements and four axes of rotation). This fact has an advantage since it is possible to have information about each angular displacement parameter, both the rotation angles and the axes. While the studied mechanism has flat movements and the axes of rotation only characterize the direction of rotation on the z-axis, spatial motion does require accurate information from the rotation axes since they are updated in each sequence of movements.

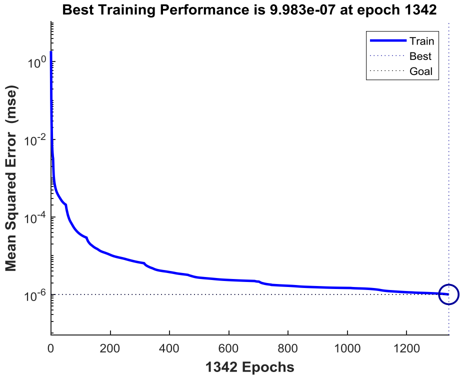

FIGURE 7. **Performance graph.**

E. NEURAL NETWORK VALIDATION

In this section, we proceed to validate the trained neural network with the inverse kinematic problem. The performance of a trained neural network can be measured to some extent by an error during training, but it is good practice to investigate the network response in more detail. One option is to perform a regression analysis between the network response and the corresponding target data in the training set. The graph in Figure 8 shows the regression of the network's fit to the training set data. The correlation coefficient (R-value) is a measure of how well the objectives explain the variation in the output; if this number equals 1, there is a perfect correlation between the target data and the network outputs. A continuous line indicates the best fit, the network outputs are plotted against the target data as open circles and a continuous line indicates the perfect fit (network output equal to target data).

With regression analysis, it can be inferred that the network will perform well to unknown data during training and the following is to test the network with unknown data. To do this, two cases were considered: 1) a study of the singularities and 2) the use of a straight path. The singularities related to the 5R planar mechanism studied in this work have been analyzed in [56]. For the case of the singularities related to the 5R mechanism in this study, the cases shown in Figures 8 and 9 were considered.

According to Tables 1 and 2, the trained network gave very close results, both at the singular points considered and at nearby points, so it is possible to affirm that it had a satisfactory training. Another novelty and advantage offered by the modeling proposed in this work and its solution are that

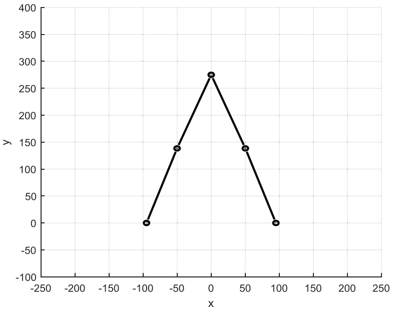

FIGURE 8. Singular configuration 1.

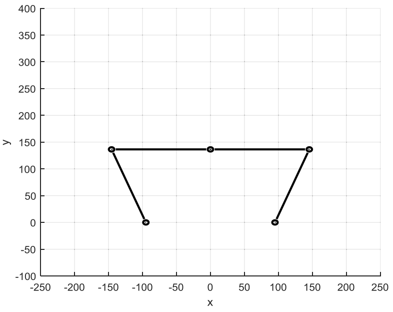

FIGURE 9. Singular configuration 2.

TABLE 1. Network results from a singular point.

	$\Gamma_{pot',O}$		ANN			
	$\Gamma_{pot',O,x}$	$\Gamma_{pot',O,y}$	α_1	α_2	α_3	α_4
Singular point	0	275.056	72.390	108.036	69.539	110.129
Nearby point 1	-1	274.000	75.969	105.937	66.270	112.731
Nearby point 2	0	274.000	75.035	104.956	66.815	113.318
Nearby point 3	1	274.000	74.026	104.069	67.430	113.811

TABLE 2. Network results for another singular point.

	$\Gamma_{pot',O}$		ANN			
	$\Gamma_{pot',O,x}$	$\Gamma_{pot',O,y}$	α_1	α_2	α_3	α_4
Singular point	0	136.455	110.285	69.694	-1.531	179.270
Nearby point 1	-1	136.000	110.707	70.116	-1.520	-179.192
Nearby point 2	0	136.000	110.285	69.694	-1.518	-179.260
Nearby point 3	1	136.000	109.864	69.272	-1.524	-179.303

it does not present mathematical singularities, so it is possible to reach the unique points in the working area, considering that the singular mechanics (physical) are different from the mathematics. Mathematically, it is possible to obtain data in a physical singularity; however, it is impossible to pass into reality.

It should be mentioned that this work does not present a comprehensive and in-depth study of the problem of the singularities associated with the 2-DOF Parallel Mechanism. Only a few unique positions were considered to know the results delivered by the neural network.

On the other hand, to test the network elsewhere in its workspace, the path shown in Figure 10 was considered. The mathematical relationships of the rectilinear trajectory are as follows:

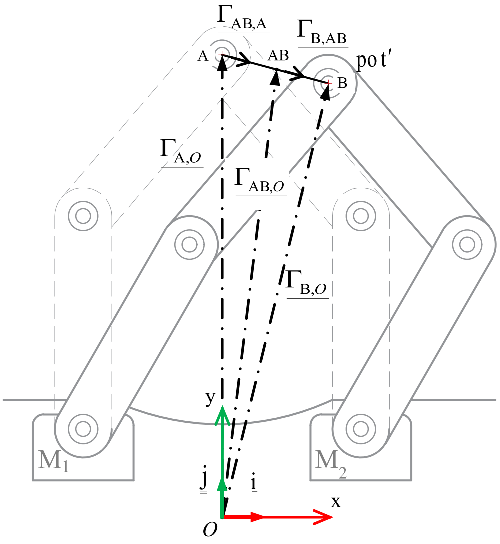

FIGURE 10. Straight-line trajectory.

For the stretch A–AB.

$$\underline{\Gamma_{AB,O}}(t) = \underline{\Gamma_{A,O}} \oplus \Gamma_{AB,A}(t) \bullet \frac{\begin{bmatrix} x_{AB,A} - x_A \\ y_{AB,A} - y_A \end{bmatrix}^T}{d_{AB,A}}$$

$$\dot{\underline{\Gamma_{AB,O}}}(t) = \dot{\Gamma}_{AB,A}(t) \bullet \frac{\begin{bmatrix} x_{AB,A} - x_A \\ y_{AB,A} - y_A \end{bmatrix}^T}{d_{AB,A}}$$

$$\ddot{\underline{\Gamma_{AB,O}}}(t) = \ddot{\Gamma}_{AB,A}(t) \bullet \frac{\begin{bmatrix} x_{AB,A} - x_A \\ y_{AB,A} - y_A \end{bmatrix}^T}{d_{AB,A}} \tag{27}$$

For the stretch A-AB.

$$\underline{\Gamma_{B,O}}(t) = \underline{\Gamma_{AB,O}}(t) \oplus \Gamma_{B,AB}(t) \bullet \frac{\begin{bmatrix} x_B - x_{B,AB} \\ y_B - y_{B,AB} \end{bmatrix}^T}{d_{B,AB}}$$

$$\dot{\underline{\Gamma_{B,O}}}(t) = \dot{\Gamma}_{B,AB}(t) \bullet \frac{\begin{bmatrix} x_B - x_{B,AB} \\ y_B - y_{B,AB} \end{bmatrix}^T}{d_{B,AB}}$$

$$\ddot{\underline{\Gamma_{B,O}}}(t) = \ddot{\Gamma}_{B,AB}(t) \bullet \frac{\begin{bmatrix} x_B - x_{B,AB} \\ y_B - y_{B,AB} \end{bmatrix}^T}{d_{B,AB}} \tag{28}$$

Expressions (25) and (26) represent displacement, speed, and acceleration related to a moving point traveling through space-time. The functions $\Gamma_{AB,A}(t)$, $\dot{\Gamma}_{AB,A}(t)$, $\ddot{\Gamma}_{AB,A}(t)$, $\Gamma_{B,AB}(t)$, $\dot{\Gamma}_{B,AB}(t)$ and $\ddot{\Gamma}_{B,AB}(t)$ are movement intensities and can be represented by equivalent interpolated functions ($\Leftrightarrow$), such as degree 5 polynomial functions [57].

For the A-AB stretch, the following expressions are available:

$$\begin{aligned} \Gamma_{AB,A}(t) &\Leftrightarrow P_{AB,A}(t) = \alpha_0 + \alpha_1 t + \alpha_2 t^2 + \alpha_3 t^3 + \alpha_4 t^4 + \alpha_5 t^5 \\ \dot{\Gamma}_{AB,A}(t) &\Leftrightarrow \dot{P}_{AB,A}(t) = \alpha_1 + 2\alpha_2 t + 3\alpha_3 t^2 + 4\alpha_4 t^3 + 5\alpha_5 t^4 \\ \ddot{\Gamma}_{AB,A}(t) &\Leftrightarrow \ddot{P}_{AB,A}(t) = 2\alpha_2 + 6\alpha_3 t + 12\alpha_4 t^2 + 20\alpha_5 t^3 \end{aligned} \tag{29}$$

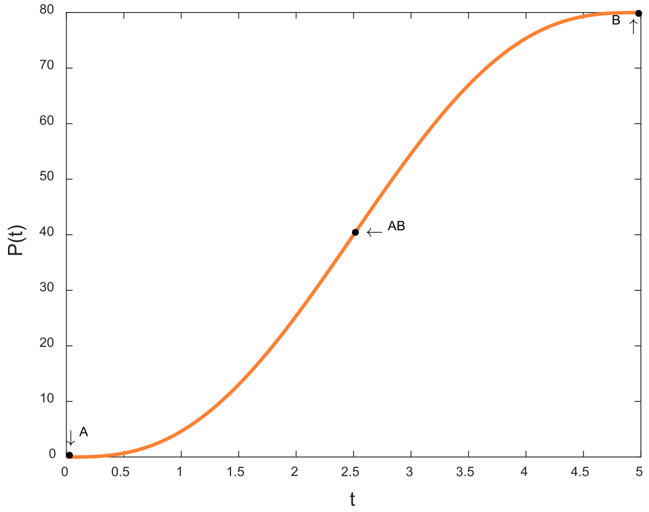

FIGURE 11. Smoothed scroll profile.

The values of the coefficients of the polynomials are obtained by knowing the distance values in the stretch under consideration, the start and end speeds of the run, and the start and end values of the acceleration (or the values where the function is required to work). Figure 11 shows the displacement profile on both sections interpolated with a polynomial of degree 5.

To verify that the neural network infers with good accuracy, the results are compared with results using the Newton–Raphson method, considering the following relationships:

$$\begin{array}{ll}\text{For the stretch A-AB.} & \text{For the stretch AB-B.}\\ \underline{\Gamma}_{\mathrm{pot}',\mathcal{O}}(t) = \underline{\Gamma}_{\mathrm{AB},\mathcal{O}}(t) & \underline{\Gamma}_{\mathrm{pot}',\mathcal{O}}(t) = \underline{\Gamma}_{\mathrm{B},\mathcal{O}}(t)\end{array} \tag{30}$$

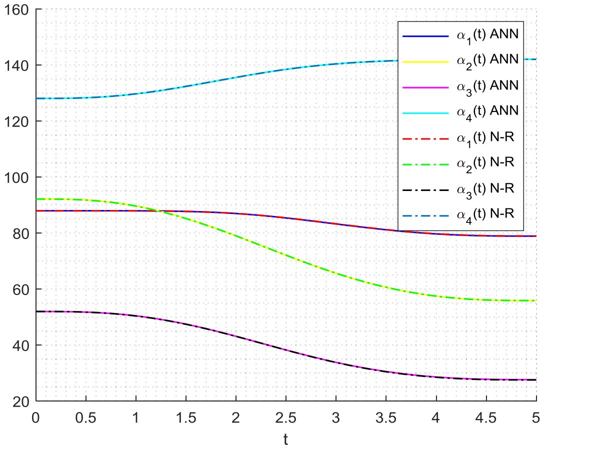

FIGURE 12. Comparison of the trained network and the Newton-Raphson method for rectilinear trajectory in terms of rotation angles.

TABLE 3. Examples of validation of ann results for inverse kinematics.

$\Gamma_{\mathrm{pot}',\mathcal{O}}$		ANN				Newton-Raphson			
$\Gamma_{\mathrm{pot}',\mathcal{O},x}$	$\Gamma_{\mathrm{pot}',\mathcal{O},y}$	α_1	α_2	α_3	α_4	α_1	α_2	α_3	α_4
0.000	260.000	87.935	92.139	51.958	128.042	87.894	92.106	51.963	128.037
2.059	258.340	87.927	90.478	50.929	129.127	87.903	90.430	50.914	129.144
16.165	246.964	87.216	80.713	44.329	134.693	87.224	80.686	44.308	134.722
40.231	227.555	83.747	67.022	34.772	139.985	83.822	67.076	34.734	139.989
57.802	213.385	79.922	57.983	28.907	141.777	79.985	57.991	28.803	141.786
62.000	210.000	78.868	55.852	27.631	142.013	78.923	55.859	27.512	142.002

Explicitly for the A-AB section, we have:

$$\begin{aligned}
\Gamma_{\mathrm{pot}',\mathcal{O},x}(t) &= \Gamma_{1,\mathcal{O},x} + L_{3',1}\left(P_0(t)p_0 - P_1(t)p_1\right)\\
&\quad + + L_{\mathrm{pot}',3'}\left(R_0(t)r_0 - R_1(t)r_1\right)\\
&= \Gamma_{A,\mathcal{O},x} + P_{AB,A}(t) \bullet \frac{x_{AB,A} - x_A}{d_{AB,A}}\\
\Gamma_{\mathrm{pot}',\mathcal{O},y}(t) &= \Gamma_{1,\mathcal{O},y} + L_{3',1}\left(P_0(t)p_0 - P_1(t)p_1\right)\\
&\quad + + L_{\mathrm{pot}',3'}\left(R_0(t)r_0 - R_1(t)r_1\right)\\
&= \Gamma_{A,\mathcal{O},y} + P_{AB,A}(t) \bullet \frac{y_{AB,A} - y_A}{d_{AB,A}}\\
\Gamma_{\mathrm{pot}',\mathcal{O},x}(t) &= \Gamma_{2,\mathcal{O},x} + L_{4',2}\left(Q_0(t)q_0 - Q_1(t)q_1\right)\\
&\quad + + L_{\mathrm{pot}',4'}\left(S_0(t)s_0 - S_1(t)s_1\right)\\
&= \Gamma_{A,\mathcal{O},x} + P_{AB,A}(t) \bullet \frac{x_{AB,A} - x_A}{d_{AB,A}}\\
\Gamma_{\mathrm{pot}',\mathcal{O},y}(t) &= \Gamma_{2,\mathcal{O},y} + L_{4',2}\left(Q_0(t)q_0 - Q_1(t)q_1\right)\\
&\quad + + L_{\mathrm{pot}',4'}\left(S_0(t)s_0 - S_1(t)s_1\right)\\
&= \Gamma_{A,\mathcal{O},y} + P_{AB,A}(t) \bullet \frac{y_{AB,A} - y_A}{d_{AB,A}}
\end{aligned} \tag{31}$$

Also, to solve the inverse kinematic problem with the Newton-Raphson method, the following relationships are required:

$$\begin{aligned}
&P_0(t)^2 + P_1(t)^2 = 1; \quad Q_0(t)^2 + Q_1(t)^2 = 1;\\
&R_0(t)^2 + R_1(t)^2 = 1; \quad S_0(t)^2 + S_1(t)^2 = 1
\end{aligned} \tag{32}$$

The relationships between rotation parameters and complex number components are:

$$\begin{aligned}
P(t) &= (P_0(t), P_1(t))\,;\; P_0(t) = \cos\alpha_1(t);\, P_1(t) = \pm\sin\alpha_1(t)\\
Q(t) &= (Q_0(t), Q_1(t))\,;\; Q_0(t) = \cos\alpha_2(t);\, Q_1(t) = \pm\sin\alpha_2(t)\\
R(t) &= (R_0(t), R_1(t))\,;\; R_0(t) = \cos\alpha_3(t);\, R_1(t) = \pm\sin\alpha_3(t)\\
S(t) &= (S_0(t), S_1(t))\,;\; S_0(t) = \cos\alpha_4(t);\, S_1(t) = \pm\sin\alpha_4(t)
\end{aligned} \tag{33}$$

V. DISCUSSION

The coordinates, $\underline{\Gamma}_{\mathrm{pot}',\mathcal{O},x}$, $\underline{\Gamma}_{\mathrm{pot}',\mathcal{O},y} \in \Re$ shown in Table 3, were generated using equations (29) and (31). This table shows three columns of interest: the linear path with 100 points (only 6 shown), the response of the trained neural network for the inverse problem, and the response of the Newton–Raphson method (see Figure 12). To get a better idea of neural network performance and how proximate their results are, they were calculated: absolute error (Ea), quadratic error (MSE), and the root of the mean quadratic error (RMSE) for each angular displacement parameter; the results are shown in the following Tables:

According to the tables above, the results obtained from the neural network model show a failure rate (mse) of 4.87e-4 and 4.44e-3 for the best and worst cases, respectively, being a fairly accurate model and with a numerical precision of one digit. For the study of singularities, Tables 8 and 9 show the comparative results between the trained network and the

TABLE 4. Examples of ann results and error calculation for α_1.

α_1		E_a	MSE	RMSE
ANN	Newton–Raphson			
87.935	87.894	4.11E-02	2.49E-03	4.99E-02
87.91	87.895	1.52E-02		
86.895	86.908	1.29E-02		
83.075	83.152	7.69E-02		
79.578	79.639	6.07E-02		
78.868	78.923	5.57E-02		

TABLE 5. Examples of ann results and error calculation for α_2.

α_2		E_a	MSE	RMSE
ANN	Newton–Raphson			
92.139	92.106	3.26E-02	1.47E-03	3.84E-02
89.570	89.514	5.57E-02		
78.682	78.678	3.93E-03		
65.222	65.268	4.54E-02		
57.277	57.284	6.94E-03		
55.852	55.859	7.49E-03		

TABLE 6. ExampleS of ann results and error calculation α_3.

α_3		E_a	MSE	RMSE
ANN	Newton–Raphson			
51.958	51.963	5.30E-03	4.44E-03	6.66E-02
50.35	50.327	2.38E-02		
42.9	42.888	1.15E-02		
33.556	33.507	4.90E-02		
28.479	28.37	1.09E-01		
27.631	27.512	1.19E-01		

TABLE 7. Examples of ann results and error calculation α_4.

α_4		E_a	MSE	RMSE
ANN	Newton–Raphson			
128.042	128.037	5.40E-03	4.87E-04	2.21E-02
129.702	129.731	2.89E-02		
135.674	135.687	1.28E-02		
140.446	140.459	1.27E-02		
141.863	141.866	3.18E-03		
142.013	142.002	1.10E-02		

TABLE 8. Results at a singular point.

$\Gamma_{pot',O}$		ANN				Newton-Raphson			
$\Gamma_{pot',O,x}$	$\Gamma_{pot',O,y}$	α_1	α_2	α_3	α_4	α_1	α_2	α_3	α_4
0	275.056	72.39	108.036	69.539	110.128	70.946	109.054	70.946	109.054
-1	274	75.969	105.937	66.27	112.731	76.533	105.422	65.596	113.195
0	274	75.035	104.956	66.815	113.318	75.625	104.375	66.131	113.869
1	274	74.026	104.069	67.43	113.811	74.578	103.467	103.467	114.404

Newton–Raphson method. As can be interpreted, the results are close, although not accurate. It is possible to study the accuracy in detail by increasing the amount of data to train.

It is worth mentioning that there are other methods for the validation of neural networks, for example, the cross-validation method [58], [59]. This method was not considered in this work because there is little data and, on the other hand, it is required to split the available dataset randomly into a training sample and a test dataset. This would mean that the neural network could not recognize a significant part of the available data needed for the cross-validation method, which would mean that the neural network could not completely learn the data considered from the workspace.

TABLE 9. Result for another singular point.

$\Gamma_{pot',O}$		ANN				Newton-Raphson			
$\Gamma_{pot',O,x}$	$\Gamma_{pot',O,y}$	α_1	α_2	α_3	α_4	α_1	α_2	α_3	α_4
0	136.455	110.285	69.694	-1.531	179.27	110.309	69.691	0	-180
-1	136	110.707	70.116	-1.52	-179.192	110.729	70.112	-0.032	-179.676
0	136	110.285	69.694	-1.518	-179.26	110.308	69.692	-0.179	-179.821
1	136	109.864	69.272	-1.524	-179.303	109.888	69.271	-0.324	-179.968

Finally, the 5R planar parallel mechanism has been studied using geometric methods and whose modeling and solution are easy to solve [60]. However, the main idea of this work is to generalize the modeling of the mechanism to spatial models, where nonlinear models are presented more frequently, and the center of the study is not in itself the mechanism but the modeling methodology, training, and validation of the network.

VI. CONCLUSION

This article has modeled a 5R planar parallel mechanism using algebra of complex number. The Newton–Raphson method and artificial neural networks were used to solve the inverse problem. The main conclusions are summarized in the following points:

- The use of two configurations in mechanism modeling makes it possible to clarify and systematize the relationships that exist between the reference position and the deformed position. This clarification is not found in the traditional modeling studies of the 5R planar parallel mechanism [36], [37]. The systems of equations generated during the analysis were eight equations and eight nonlinear unknowns of the polynomial type for the inverse kinematic problem case in both configurations.
- The results obtained from the neural network model show a failure rate (mse) of 4.87e-4 and 4.44e-3 for the best and worst cases, respectively, being a fairly acceptable model in numerical precision and accuracy to solve the inverse problem. The input data to validate the inverse problem considering the neural network were the coordinates of the points that travel a straight path in the plane, and the output data were eight parameters, four related to angular displacements and four associated with the axis of rotation. This proposal is new because traditionally only angular displacements are trained and not rotation axes [61]. For the study of motion problems in space knowing the rotation, axes are an advantage, since these are updated in each rotation sequence.
- Another advantage that represents solving inverse problem with the neural network, is the training data set, since these are careful and studied so that there is a continuity in the solutions of the 5R planar parallel mechanism, therefore, having a model that generalizes unique solutions rather than multiple solutions as does the analytical model using the Newton–Raphson method.
- The two-configuration modeling method and neural network training used to solve the reverse problem of the 5R parallel planar mechanism can be widespread for the modeling of movements in the plane and space of robots

and mechanisms, considering other mathematical tools such as quaternion algebra [50].

ACKNOWLEDGMENT

The authors are would like to thank: Universidad Tecnológica del Sur de Sonora, Universidad La Salle Noroeste, Universidad Estatal de Sonora, Universidad Autónoma de Morelos, Instituto Tecnológico Superior de Cajeme and the IIMM company, for supporting the development of this article.

REFERENCES

[1] R. Zavala, R. Ramírez, and D. Chaparro, ''Kinematic and dynamical modelling for control of a parallel robot-based surveillance/sentry device,'' *Adv. Mil. Technol.*, vol. 10, no. 1, pp. 15–30, Jun. 2015.

[2] I. Kardan and A. Akbarzadeh, ''An improved hybrid method for forward kinematics analysis of parallel robots,'' *Adv. Robot.*, vol. 29, no. 6, pp. 401–411, Mar. 2015.

[3] G. Wang, ''Dynamics analysis of parallel mechanism with flexible moving platform based on floating frame of reference formulation,'' *J. Mech. Robot.*, vol. 11, no. 4, pp. 1–22, Aug. 2019, doi: 10.1115/1.4043045.

[4] C. Tian, Y. Fang, and Q. J. Ge, ''Design and analysis of a partially decoupled generalized parallel mechanism for 3T1R motion,'' *Mechanism Mach. Theory*, vol. 140, pp. 211–232, Oct. 2019.

[5] L. Wang, Z. Zhang, and Z. Shao, ''Kinematic performance analysis and promotion of a spatial 3-RPaS parallel manipulator with multiple actuation modes,'' *J. Mech. Sci. Technol.*, vol. 33, no. 2, pp. 889–902, Feb. 2019.

[6] T. Rasheed, P. Long, and S. Caro, ''Wrench-feasible workspace of mobile cable-driven parallel robots,'' *J. Mech. Robot.*, vol. 12, no. 3, Jun. 2020, Art. no. 031009, doi: 10.1115/1.4045423.

[7] S. Staicu, Z. Shao, Z. Zhang, X. Tang, and L. Wang, ''Kinematic analysis of the X4 translational-rotational parallel robot,'' *Int. J. Adv. Robot. Syst.*, vol. 15, no. 5, pp. 1–12, Oct. 2018.

[8] J. Wu, T. Li, J. Wang, and L. Wang, ''Performance analysis and comparison of planar 3-DOF parallel manipulators with one and two additional branches,'' *J. Intell. Robot. Syst.*, vol. 72, no. 1, pp. 73–82, Oct. 2013, doi: 10.1007/s10846-013-9824-8.

[9] J. Wu, Y. Gao, B. Zhang, and L. Wang, ''Workspace and dynamic performance evaluation of the parallel manipulators in a spray-painting equipment,'' *Robot. Comput.-Integr. Manuf.*, vol. 44, pp. 199–207, Apr. 2017.

[10] R. Kelaiaia, ''Improving the pose accuracy of the delta robot in machining operations,'' *Int. J. Adv. Manuf. Technol.*, vol. 91, nos. 5–8, pp. 2205–2215, Jul. 2017.

[11] Y. Li, D. Shang, and Y. Liu, ''Kinematic modeling and error analysis of delta robot considering parallelism error,'' *Int. J. Adv. Robot. Syst.*, vol. 16, no. 5, pp. 1–9, Sep. 2019, doi: 10.1177/1729881419878927.

[12] M. Masal, S. Ersoy, and M. A. Güngör, ''Euler–Savary formula for the homothetic motion in the complex plane C,'' *Ain Shams Eng. J.*, vol. 5, no. 1, pp. 305–308, Mar. 2014, doi: 10.1016/j.asej.2013.09.006.

[13] D. Myszka, *Machine and Mechanisms Applied Kinematic Analysis*, 4th ed. Upper Saddle River, NJ, USA: Prentice-Hall, 2012.

[14] X. Yang, H. Wu, Y. Li, and B. Chen, ''A dual quaternion solution to the forward kinematics of a class of six-DOF parallel robots with full or reductant actuation,'' *Mechanism Mach. Theory*, vol. 107, pp. 27–36, Jan. 2017.

[15] S. Singh, A. Singla, A. Singh, S. Soni, and S. Verma, ''Kinematic modelling of a five-DOFs spatial manipulator used in robot-assisted surgery,'' *Perspect. Sci.*, vol. 8, pp. 550–553, Sep. 2016.

[16] H. Fadhil, ''Proposed algorithm to solve inverse kinematics problem of the robot,'' *Eur. J. Sci. Res.*, vol. 149, no. 4, pp. 376–384. Jul. 2018.

[17] A. Elsheikh, E. Showaib, and A. Asar, ''Artificial neural network based forward kinematics solution for planar parallel manipulators passing through singular configuration,'' *Adv. Robot Autom.*, vol. 2, no. 2, pp. 1–6, Sep. 2013.

[18] N. Khan, I. Ullah, and M. Al-Grafi, ''Dimensional synthesis of mechanical linkages using artificial neural networks and Fourier descriptors,'' *Mech. Sci.*, vol. 6, no. 1, pp. 29–34, Apr. 2015.

[19] J. Ghasemi, R. Moradinezhad, and M. Hosseini, ''Kinematic synthesis of parallel manipulator via neural network approach,'' *Int. J. Eng.*, vol. 30, no. 9, pp. 1319–1325, Sep. 2017.

[20] J. U. Korein and N. I. Badler, ''Techniques for generating the goal-directed motion of articulated structures,'' *IEEE Comput. Graph. Appl.*, vol. 2, no. 9, pp. 71–81, Nov. 1982.

[21] A. Gasparetto and V. Zanotto, ''Optimal trajectory planning for industrial robots,'' *Adv. Eng. Softw.*, vol. 41, no. 4, pp. 548–556, Apr. 2010.

[22] I. Tanase, T. Itul, E. Campean, and A. Pisla, ''Workspace identification using neural network for an optimal designed 2-DOF orientation parallel device,'' in *New Trends in Mechanism and Machine Science* (Mechanisms and Machine Science), vol. 7, F. Viadero and M. Ceccarelli, Eds. Dordrecht, The Netherlands: Springer, 2013, doi: 10.1007/978-94-007-4902-3_17.

[23] S. Zare, M. S. Haghighi, M. R. H. Yazdi, A. Kalhor, and M. T. Masouleh, ''Kinematic analysis of an under-constrained cable-driven robot using neural networks,'' in *Proc. 28th Iranian Conf. Electr. Eng. (ICEE)*, Tabriz, Iran, Aug. 2020, pp. 1–6, doi: 10.1109/ICEE50131.2020.9260982.

[24] E. G. López, W. Yu, and X. Li, ''Optimum design of a parallel robot using neuro-genetic algorithm,'' *J. Mech. Sci. Technol.*, vol. 35, no. 1, pp. 293–305, Jan. 2021.

[25] J. S. Toquica, P. S. Oliveira, W. S. R. Souza, J. M. S. T. Motta, and D. L. Borges, ''An analytical and a deep learning model for solving the inverse kinematic problem of an industrial parallel robot,'' *Comput. Ind. Eng.*, vol. 151, Jan. 2021, Art. no. 106682, doi: 10.1016/j.cie.2020.106682.

[26] M. M. Sanjeev, M. J. Thomas, T. K. S. Kumar, A. P. Sudheer, and M. L. Joy, ''Determination of inverse kinematic solutions for a 3 degree of freedom parallel manipulator using machine learning,'' in *Proc. IEEE Students Conf. Eng. Syst. (SCES)*, Prayagraj, India, Jul. 2020, pp. 1–6.

[27] M. Sheikhpour and B. Mahdi, ''Forward kinematic analysis of tensegrity mechanism using hybrid method,'' in *Proc. 5th RSI Int. Conf. Robot. Mechatronics (IcRoM)*, Tehran, Iran, Oct. 2017, pp. 564–569.

[28] H. Zhang, H. Fang, B. Jiang, F. Zhao, and T. Zhu, ''A Newton-Raphson and BP neural network hybrid algorithm for forward kinematics of parallel manipulator,'' in *Proc. 2nd WRC Symp. Adv. Robot. Automat.*, Beijing, China, Aug. 2019, pp. 122–127.

[29] P. J. Parikh and S. S. Y. Lam, ''A hybrid strategy to solve the forward kinematics problem in parallel manipulators,'' *IEEE Trans. Robot.*, vol. 21, no. 1, pp. 18–25, Feb. 2005.

[30] O. Hamdoun, L. E. Bakkali, and F. Z. Baghli, ''Analysis and optimum kinematic design of a parallel robot,'' *Procedia Eng.*, vol. 181, pp. 214–220, Jan. 2017, doi: 10.1016/j.proeng.2017.02.374.

[31] D. Chablat, X. Kong, and C. Zhang, ''Kinematics, workspace, and singularity analysis of a parallel robot with five operation modes,'' *J. Mech. Robot.*, vol. 10, no. 3, pp. 1–18, Jun. 2018, doi: 10.1115/1.4039400.

[32] M. W. Jang, W. S. Jang, and S. M. Hong, ''A study on the development of a robot vision control scheme based on the Newton-Raphson method for the uncertainty of circumstance,'' *Trans. Korean Soc. Mech. Eng. A*, vol. 40, no. 3, pp. 305–315, Mar. 2016.

[33] X. J. Liu and J. S. Wang, *Parallel Kinematics*, 1st ed. Berlin, Germany: Springer-Verlag, 2014.

[34] W. Yuan, Z. Zhang, Z. Shao, L. Wang, and L. Du, ''Performance research of planar 5R parallel mechanism with variable drive configurations,'' in *Intelligent Robotics and Applications*, Y. Huang, Eds. Cham, Switzerland: Springer, 2017, pp. 455–460.

[35] L. Wang, Z. Zhang, Z. Shao, and X. Tang, ''Analysis and optimization of a novel planar 5R parallel mechanism with variable actuation modes,'' *Robot. Comput.-Integr. Manuf.*, vol. 56, pp. 178–190, Apr. 2019.

[36] E. D. Sosa, M. Arias, and E. Lugo, ''A numerical approach for the inverse and forward kinematic analysis of 5R parallel manipulator,'' presented at the 14th Int. Conf. Elect. Eng., Comput. Sci. Autom. Control (CCE), Mexico City, Mexico, Sep. 2017.

[37] F. A. Lara-Molina, E. H. Koroishi, V. Steffen, and L. A. Martins, ''Kinematic performance of planar 5R symmetrical parallel mechanism subjected to clearances and uncertainties,'' *J. Brazilian Soc. Mech. Sci. Eng.*, vol. 40, no. 4, pp. 1–15, Apr. 2018, doi: 10.1007/s40430-018-1118-4.

[38] M. Özdemir, ''High-order singularities of 5R planar parallel robots,'' *Robotica*, vol. 37, no. 2, pp. 233–245, Feb. 2019, doi: 10.1017/S0263574718000966.

[39] M. R. Vezvari, A. Nikoobin, and A. Ghoddosian, ''Perfect torque compensation of planar 5R parallel robot in point-to-point motions, optimal control approach,'' *Robotica*, pp. 1–18, Oct. 2020, doi: 10.1017/S0263574720000971.

[40] X. Kong, ''Reconfiguration analysis of a 4-DOF 3-RER parallel manipulator with equilateral triangular base and moving platform,'' *Mechanism Mach. Theory.*, vol. 98, pp. 180–189. Apr. 2016.

[41] M. Tuong, T. Tin, and C. Bang, ''Study and development of parallel robots based on 5-bar linkage,'' presented at the Nat. Conf. Mach. Mechanisms, Ho Chi Minh City, Vietnam, Oct. 2015.

[42] E. Campean, T. P. Itul, I. Tanase, and A. Pisla, ''Workspace generation for a 2–DOF parallel mechanism using neural networks,'' *Appl. Mech. Mater.*, vol. 162, pp. 121–130, Mar. 2012.

[43] T. D. Le, H.-J. Kang, Y.-S. Suh, and Y.-S. Ro, ''An online self-gain tuning method using neural networks for nonlinear PD computed torque controller of a 2-DOF parallel manipulator,'' *Neurocomputing*, vol. 116, pp. 53–61, Sep. 2013.

[44] T. D. Le, H.-J. Kang, and Y.-S. Suh, ''Chattering-free neuro-sliding mode control of 2-DOF planar parallel manipulators,'' *Int. J. Adv. Robot. Syst.*, vol. 10, no. 1, pp. 1–15, Jan. 2013.

[45] X. Zhao, J. Guo, K. Li, L. Dai, and J. Chen, ''Optimal design and experiment of 2-DoF five-bar mechanism for flower seedling transplanting,'' *Comput. Electron. Agricult.*, vol. 178, Nov. 2020, Art. no. 105746.

[46] S. Sergium, V. Maties, and R. Balan, ''Multiple-goal kinematic optimization of 2 DOF micro parallel robots,'' in *Proc. IEEE Int. Conf. Mechatronics Automat.*, Harbin, China, Aug. 2007, pp. 2535–2540.

[47] L. Reyes, ''Sobre la parametrización de las Rotaciones y Reflexiones de Multicuerpos Rígidos en el plano,'' Universidad Anáhuac del Sur, México D.F., México. Estudios ocasionales, Mexico City, Mexico, Tech. Rep. 1-98, Nov. 1998.

[48] J. Angeles, *Fundamentals of Robotic Mechanical Systems, Theory, Methods, and Algorithms*, vol. 124, 4th ed. Cham, Switzerland: Springer, 2014.

[49] E. Yime, J. Roldán, and J. Villa, ''Computed torque control of a 2-RR planar parallel robot,'' *Prospectiva*, vol. 15, no. 2, pp. 85–95, Dec. 2017.

[50] L. Reyes, ''Quaternions: Une representation parametrique systematique des rotations finies. Partie 1: Le cadre theorique,'' INRIA Rocquencourt, Le Chesnay-Rocquencourt, France, Rapport de Recherche 1303, 1990.

[51] H. Demuth, M. Beale, O. De Jess, and M. Hagan, *Neural Network Design*, 2nd ed. 2014.

[52] L. O. González, J. Gotay, M. Roodschild, A. L. Will, and S. Rodríguez, ''Optimización en la elaboración de redes neuronales artificiales adaptativas usando una metodología de algoritmo de poda,'' *Ingenio Magno*, vol. 8, no. 1, pp. 44–56, Jan. 2017.

[53] S. K. Sharma and P. Chandra, ''Constructive neural networks: A review,'' *Int. J. Eng. Sci. Technol.*, vol. 2, no. 12, pp. 7847–7855, Dec. 2010.

[54] D. Belloda, F. Trujillo, and E. Lugo, ''Topology of an RNA to calculate the inverse kinematics of the puma 560 for real-time applications,'' *Res. Comput. Sci.*, vol. 148, no. 8, pp. 397–405, Dec. 2019.

[55] M. T. Hagan and M. B. Menhaj, ''Training feedforward networks with the marquardt algorithm,'' *IEEE Trans. Neural Netw.*, vol. 5, no. 6, pp. 989–993, Nov. 1994.

[56] X.-J. Liu, J. Wang, and G. Pritschow, ''Kinematics, singularity and workspace of planar 5R symmetrical parallel mechanisms,'' *Mechanism Mach. Theory*, vol. 41, no. 2, pp. 145–169, Feb. 2006.

[57] K. Lynch and F. Park, *Modern Robotics: Mechanics, Planning, and Control*, 4th ed. Cambridge, U.K.: Cambridge Univ. Press. 2017.

[58] S. Haykin, *Neural Networks and Learning Machines*. New York, NY, USA, Prentice-Hall. 2009.

[59] M. Kearns, ''A bound on the error of cross validation using the approximation and estimation rates, with consequences for the training-test split,'' *Neural Comput.*, vol. 9, no. 5, pp. 1143–1161, Jul. 1997, doi: 10.1162/neco.1997.9.5.1143.

[60] T. D. Le, H.-J. Kang, and Q. V. Doan, ''A method for optimal kinematic design of five-bar planar parallel manipulators,'' presented at the Int. Conf. Control, Automat. Inf. Sci. (ICCAIS), Nha Trang, Vietnam, Nov. 2013.

[61] J. Demby's, Y. Gao, and G. N. DeSouza, ''A study on solving the inverse kinematics of serial robots using artificial neural network and fuzzy neural network,'' presented at the IEEE Int. Conf. Fuzzy Syst. (FUZZ-IEEE), New Orleans, LA, USA, Jun. 2019.

EUSEBIO JIMÉNEZ LÓPEZ received the bachelor's degree in industrial mechanical engineering from the Instituto Tecnológico de Saltillo, in 1994, and the master's and Ph.D. degrees in mechanical engineering from the Universidad Nacional Autónoma de México, in 1998 and 2013, respectively. He is currently a Professor with the Universidad Tecnológica del Sur de Sonora and Universidad La Salle Noroeste.

DANIEL SERVÍN DE LA MORA-PULIDO was born in Sonora, México. He received the B.S. degree in mechatronics engineering from Universidad La Salle Noroeste. His research interests include robotics systems and artificial intelligence applications.

RAÚL SERVÍN DE LA MORA-PULIDO was born in Sonora, México. He received the B.S. degree in mechatronics engineering from Universidad La Salle Noroeste. His research interests include robotics systems and integrated automation.

FRANCISCO JAVIER OCHOA-ESTRELLA received the master's degree in mechanical design from UNAM and the Ph.D. degree in electronics from UNISON. He is currently a full-time Research Professor in the Electronic Engineering Division, TNM/Instituto Tecnológico Superior de Cajeme. He is also a member of the Academic Group Design in Engineering, and collaborates in the Research Line Materials, Devices and Electromechanical Systems developing projects to solve problems through the design of electromechanical systems, opto-electronic devices and the analysis, and manufacture of nanostructured materials.

MARIO ACOSTA FLORES was born in Cuernavaca, Morelos, México. He received the M.S. and Ph.D. degrees in mechanical engineering from the Autonomous National University of México (UNAM), in 2000 and 2010, respectively. He has extensive experience in experimental mechanics, since 2002. He has worked on projects for structural redesign for International, Bombardier, and among others. He is currently a Research Professor with the Autonomous University of the State of Morelos, Mexico. His research interests include composite materials, experimental analysis of efforts in scale models, thermal efforts, and biomechanics.

GABRIEL LUNA-SANDOVAL received the master's degree in mechatronics and the Ph.D. degree in mechanical engineering. He is currently a Research Professor with the State University of Sonora, Mexico. He is also a Mechanical Engineer.