J. Appl. Comput. Mech., 11(4) (2025) 1172-1182
DOI: 10.22055/jacm.2025.48048.4945

ISSN: 2383-4536
jacm.scu.ac.ir

Shahid Chamran University of Ahvaz

Journal of Applied and Computational Mechanics

Research Paper

Analytical Inverse Kinematics-Based Method for a 6-DOF Hexa Parallel Robot

Hendriko Hendriko[1], Nur Khamdi[1], Jajang Jaenudin[1,2], Roni Novison[1], Muhammad Ikhsan[1]

[1] Mechatronics Engineering Department, Politeknik Caltex Riau, Pekanbaru 28265, Riau Province, Indonesia,
Email: hendriko@pcr.ac.id (H.H.); khamdi@pcr.ac.id (N.K.); roni@pcr.ac.id (R.N.); mikhsan@pcr.ac.id (M.I.)

[2] Graduate Institute of Automation and Control National, Taiwan University of Science and Technology, Taipei, Taiwan, Email: jajang@pcr.ac.id

Received November 29 2024; Revised January 23 2025; Accepted for publication January 25 2025.
Corresponding author: H. Hendriko (hendriko@pcr.ac.id)

Abstract. This paper presents an analytical inverse kinematic simulation method to define the kinematics aspect of a 6-degree-of-freedom (DOF) Hexa parallel robot. The robot was designed using six independent links, each constructed of upper and lower links. Every link has three joints with a 6-RSS configuration. This configuration is aimed to enhance motion flexibility. In this study, the orientation of the end effector is adjusted based on its position to larger the reach and workspace. The proposed method was implemented to create a computer program in MATLAB, which was used to calculate the motors' motion during a complex path. The results indicate that the developed algorithm can determine the rotation motion of all the motors, which are used to place the end of the arm tool to the target position and orientation. Another verification test was conducted to assess the applicability of the proposed algorithm to a physical robot. The results demonstrated that the proposed approach could perform complex tasks accurately. Moreover, the comparison test proved that the proposed method is more efficient regarding computational time than the Jacobian method. It is the main advantage of the method compared to numerical based method.

Keywords: Hexa robot, inverse kinematics, 6-DOF, analytical method, parallel robot.

1. Introduction

In industrial manufacturing, the parallel manipulator is extensively employed for precision tasks, with many designed to increase productivity and improve the applicability of products. Parallel robots offer several benefits over serial robots, like better load and weight ratio, better stiffness, accuracy, higher velocity, and lower inertia. Most researchers agree that the drawback of parallel robots is the limited movement area and the possibility of complicated dynamic singularities that may remain in the working area [1]. A comprehensive comparative analysis between serial and parallel robots has been presented by Cox and Tesar [2].

Several types of parallel manipulator structures and their implementation in industrial have been published in the literature, such as the DELTA robot [3], 3-DOF Planar robot [4], 3-DOF Spatial robots [5], 3-Universal-Prismatic-Universal robot [6], and 3-Prismatic Revolute Spherical robot [7]. All these manipulators attribute a parallel structure with three DOFs. Nunez et al. performed a comparative analysis in parallel mechanism focused on additive manufacturing [8]. They analyze several aspects, including mechanical design, kinematics analysis, dynamics analysis, and workspace. There are several parallel robots featuring more than five DOFs that have been published. Isaksson et al. [9] constructed an octahedral hexarot, a 6-DOF manipulator. The link formation was influenced by the original structure developed by the original Gough [10]. Research in the kinematic aspect of hexarot parallel manipulator was also performed by Pedrammehr et al. [11]. The 6-DOF parallel manipulator, which has now become well-accepted in numerous modern industrial applications, is a hexapod manipulator. Various research has been carried out to scrutinize the performance of this manipulator [12-14]. Tunc et al. [12] studied the dynamic aspect of a hexapod manipulator for mobile machining. They concluded that the complicated kinematic link of a hexapod manipulator affects the rigidity. It is generally accepted that the hexapod manipulator exhibits the advantage of better stiffness and load/weight capacity, but it also shows the disadvantage of shorter reach. Another structure of the 6-DOF parallel manipulator that attracts much attention is a Hexa robot [15-21]. The Hexa robot is widely developed because it contrasts with the hexapod robot. The hexa robot is less stiff but has a larger workspace and is more flexible. Moreover, the Hexa robot is normally dedicated to high-speed applications [21].

In parallel robot technology, kinematics analysis is essential for designing the manipulator's architecture, outlining the path, and analyzing the dynamics. Inverse kinematics defines the robot's individual joint angle by considering the position and orientation of the end-effector. The problems of inverse kinematics in robotics have been extensively investigated in recent years. Many concepts have been proposed, including mostly numerical and analytical-based approaches [22].

The Jacobian-based method is an extensively used numerical approach. Campa et al. [23] developed a concept to control a 3-PUS hexapod parallel manipulator for calculating the forward and inverse kinematics. Wu and Dong also used the Jacobian algorithm to define the inverse kinematics of a 6-RUU Hexa parallel robot [16]. Jacobian-based methods provide accurate and smooth outcomes, but the most significant drawback of these methods is that they are computationally expensive owing to the complex matrix calculations.

Another type of numerical approach includes the Newton method. These methods overcome the issue using a minimization approach by searching the target form given as results; hence, they create smooth movements without uncertain discontinuities. Valencia et al. [18] developed a method for the direct and differential kinematic control of a 6-DOF Hexa robot. The kinematics model was developed based on the Newton–Raphson method. The most widely used Newton models are Broyden's approach [24] and the Broyden, Fletcher, Goldfarb, and Shanno (BFGS) approach [25]. However, those methods are expensive in terms of computational time due to the iterative calculation. To solve the problem of calculation time in a numerical approach, several researchers developed the cyclic coordinate descent (CCD) algorithm [26]. The matrix calculation of CCD is straightforward; therefore, the time for the repetitive calculation process is shorter. Another recent heuristic approach, FABRIK [27], is faster and more efficient for calculating multiple end-effectors' motion. However, the motion with erratic discontinuities is the drawback of these methods. The accuracy can be corrected by intensifying the iteration process, but this action can create expensive calculation time.

Several studies [28-33] have developed analytical methods for various applications to solve the issue of long calculation times in numerical methods. The results of these studies demonstrated that analytical approaches can significantly reduce the calculation time. Kofinas et al. [29] proposed a comprehensive analytical approach to calculate the inverse and forward kinematics for the NAO humanoid manipulator. The proposed approach is more accurate and efficient than the existing one. Dehghani et al. [19] combined an analytical approach with a neural network to determine the kinematics of a Hexa-parallel robot. The method was relevant for robots with USR configuration. Hendriko et al. [32, 33] developed an inverse kinematic algorithm based analytical method called the analytical inverse kinematics simulation (AIKS). This method was developed for a 5-DOF delta robot with three DOF for linear movement using three independent links and two DOF from the rotational motion of a mobile platform. Based on a comprehensive test, the method was found to be accurate. A comparison test demonstrated that the AIKS had a significantly lower computational cost than the Jacobian method. Analytical-based approaches developed the algorithm based on the robot's architecture. Hence, it is only applicable to a particular architecture of robot.

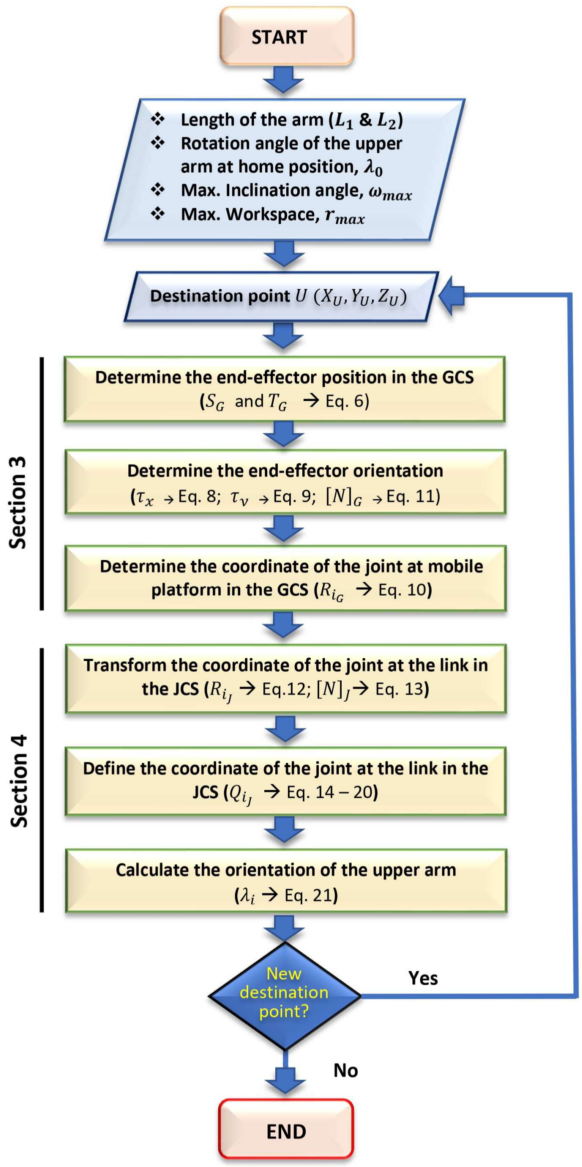

Fig. 1. Flowchart for obtaining the orientation of the upper link.

Therefore, this paper presents an analytical-based methodology to define the kinematics problems of a 6-DOF parallel robot. The AIKS, proposed for a 5-DOF delta manipulator, was modified to enable its application to a 6-DOF Hexa parallel kinematic mechanism. The AIKS was redeveloped because the configuration and construction of the 6-DOF Hexa parallel robot proposed in this study significantly differ from the 5-DOF delta robot proposed by [32, 33]. The construction of a 5-DOF delta robot has three independent links for vertical linear motion and two actuators installed on the mobile platform for rotation motion. In this study, the 6-DOF Hexa parallel robot has six independent links to move the end effector. Every independent link is actuated by a motor and constructed by two links. The motion configuration of every link was designed with a revolute, spherical, spherical (RSS). This configuration is aimed to enhance motion flexibility. The inverse kinematic model proposed in this paper was intended to define the coordinates of all the joints in every link during the end effector motion. By defining the coordinates of the joints, the motion of the motor required to move the robot link could be calculated.

Furthermore, the end effector orientation strategy was also implemented to increase the reach and workspace. The orientation of the end effector is adjusted based on its position concerning the start position (home position). The robot's architecture and motion strategy made the proposed algorithm different. In terms of computational time, the proposed method is cheaper compared to numerical based. It is another advantage of the proposed method. Figure 1 shows a flowchart summarizing a brief step to define the motor's motion. The following sections elaborate on the details. Furthermore, the developed algorithm is tested using a complex path to ensure its implementability.

2. Designing the Robot Architecture and Coordinate Systems

The construction of the Hexa-robot is an improvement of a prominent parallel manipulator, the Delta robot. Hexa robots are faster and more precise than the Delta robots. Moreover, the Hexa robot has more flexible motion due to three additional rotational DOFs [34]. In this paper, the architecture of a 6-DOF Hexa-parallel robot, as shown in Fig. 2(a), was developed. This robot consists of six independent and identical links, each consisting of the upper and lower links, as seen in Fig. 2(b). The links have a 6-RSS mechanism, where R and S stand for revolute and spherical, respectively. The developed kinematic configuration allows the end effector to move flexibly in any direction.

Inverse kinematics calculations proposed in this paper were applied to define the motion of the motor to move the end effector to the destination point. The end effector is moved by six motors installed at the fixed platform and connected to the upper link. Hence, the rotation of each motor could be defined by defining the orientation of the upper link. The orientation of the upper-link (λ) can be defined after the coordinates of point P_i and point Q_i, as depicted in Fig. 2(b), are obtained. Point P_i was installed and connected to the motor on the fixed platform. Their coordinates are fixed and known. Then, to calculate the upper-link (λ) orientation, only point Q_i must be defined. Point Q_i connects the upper-link and the lower-links. The symbol i in P_i and Q_i denotes the link order, where $i = 1, .., 6$.

An appropriate coordinate system must be established for the analytical calculation of the inverse kinematics. Therefore, three coordinate systems were established to determine the position and orientation of the moving parts, as shown in Fig. 2(c). The first is the Global Cartesian System (GCS), the main reference. The GCS is set up at the centre of the basement (point O), described by the vectors $X, \mathrm{Y}, \text{and } Z$. The second coordinate system is the Mobile Cartesian System (MCS), placed at the centre of the mobile platform (point S), as shown in Fig. 2(d). The MCS is a local coordinate system indicated by $u, v,$ and w. The coordinate transformation from MCS to JGS, and vice versa, is calculated based on the orientation of the end effector relative to GCS, and the coordinate of point S. This transformation is needed to define the coordinate of the point at the end effector, especially point R_i. The last coordinate system is the Joint Cartesian system (JCS), denoted by x_i, y_i, z_i. The JCS is located at point $P_i(P_1, P_2, P_3, P_4, P_5, P_6)$. Point P_i is located at the uppermost side of every upper link, as shown in Fig. 2(b), and there are six JCS, as shown in Figs. 2(c) and 2(e). The coordinate transformation from GCS to MCS is calculated using the orientation of the coordinate system at point P_i and the coordinate of point P_i. The transformation from GCS to MCS is required to define the coordinate of the upper link Q_i.

The coordinates of several points at the start position (home position) must be established. These points are $P_{i0}(X_{P_{i0}}, Y_{P_{i0}}, Z_{P_{i0}})$, $R_{i0}(X_{R_{i0}}, Y_{R_{i0}}, Z_{R_{i0}})$, $S_0(X_{S_0}, Y_{S_0}, Z_{S_0})$, and $T_0(X_{T_0}, Y_{Tp_0}, Z_{T_0})$. Points P_{i0}, R_{i0}, S_0, and T_0 correspond to points P_i, R_i, S, and T at the start position, subsequently.

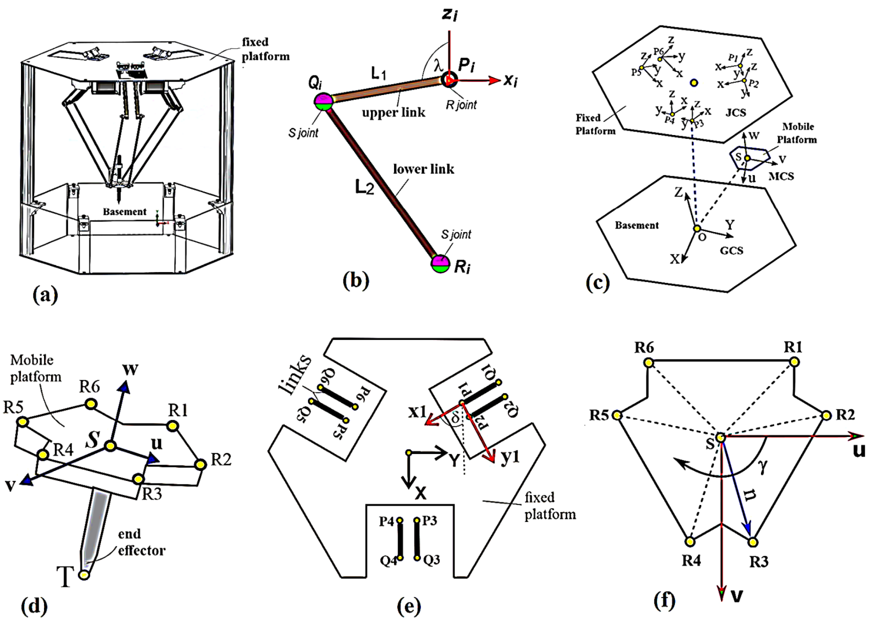

Fig. 2. (a) The architecture of the parallel robot, (b) robotic links, (c) the coordinate systems, (d) the end effector and mobile platform, (e) top view of the fixed platform, and (f) top view of the mobile platform.

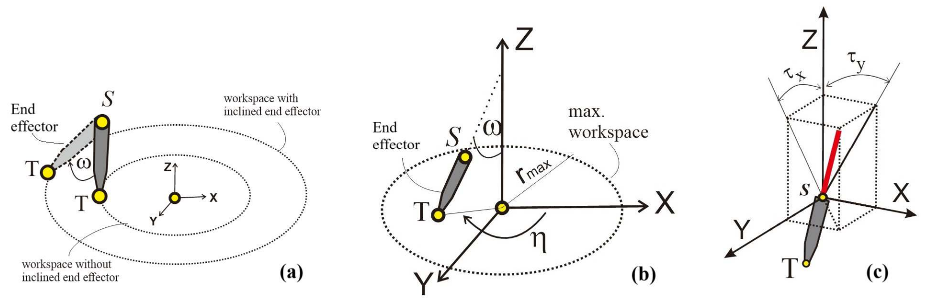

Fig. 3. Orientation of the end effector: (a) end effector reach with and without inclination angle, (b) inclination angle and (c) orientation relative to the GCS.

3. Determining the Orientation of the End Effector and the Coordinate of Points at the Mobile Platform

As shown in Fig. 1, three aspects need to be determined in this section. They are the position of the end effector, the orientation of the end effector, and the coordinate of the joint between the mobile platform and the lower link. As mentioned in the previous section, the formula proposed in this paper aimed to define the coordinates of point $Q_i(X_{Q_i}, Y_{Q_i},\ Z_{Q_i})$ for the end effector position. The position of the end effector is represented by points $T(X_T, Y_T,\ and\ Z_T)$. Point T is located at the endmost point of the end effector, as depicted in Fig. 2(d). Point T is the desired to move to the destination point $U(X_U, Y_U,\ Z_U)$. A series of destination points creates a path that should be passed through the end effector.

In this research, the orientation of the end effector was manipulated based on the coordinates of the destination point (U). This strategy was applied to enlarge the end effector's reach and increase the robot's workspace. As illustrated in Fig. 3(a), the workspace of the end effector with an inclination angle is larger than without an inclination angle. The inclination angle (ω) is set maximum at maximum reach, and then proportionally reduced based on the coordinate of point U. The inclination angle can be determined after setting the maximum reach of the end effector on the space (r_{max}), and the maximum inclination angle (ω_{max}).

The orientation of the end effector is defined by two coordinates, $T\ (X_T, Y_T,\ Z_T)$ and $S\ (X_S, Y_S,\ Z_S)$, as shown in Fig. 3(a). Therefore, point S must be calculated by referring to the coordinates of point T. The coordinates of points T and S in the MCS at the start position are $(0,0,-l)$ and $(0,0,0)$, respectively. Meanwhile, l denotes the length between points T and S. Point T in GCS is indicated by $T_G(x_{T_G}, y_{T_G}, z_{T_G})$, and it coincides with the destination point $U(X_U, Y_U,\ Z_U)$. In other word, T_G is equal to U $(T_G = U)$. The destination point is a point that the end effector should achieve, and its coordinate is given or known. Therefore, to define the orientation of the end effector, the coordinate of S in GCS (S_G) needs to be determined in this section.

Based on the given r_{max} and ω_{max}, the inclination angle of the end effector with regard to the X-axis (ω) at point U can be determined as follows:

$$\omega = \left\{ \frac{\left(\sqrt{{X_U}^2 + {Y_U}^2}\right)}{r_{max}} \right\} \omega_{max} \tag{1}$$

X_U and Y_U are the coordinates of the destination point $U(X_U, Y_U,\ Z_U)$ in GCS.

To put the endmost point of the end effector to the point U, it must be rotated with respect to the Z-axis in the GCS. η is the rotation angle, measured from the X-axis to point U, as shown in Fig. 3(a). It is defined as follows:

$$\eta = \tan^{-1}\left(\frac{Y_U}{X_U}\right) \tag{2}$$

Equation (2) is correct if point U is within the first quadrant. Otherwise, the rotation angle η is calculated as follows:

$$\begin{aligned} &\eta = 180 - \eta \text{ for the 2}^{\text{nd}} \text{ quadrant,} \\ &\eta = 180 + \eta \text{ for the 3}^{\text{rd}} \text{ quadrant,} \\ &\eta = 360 + \eta \text{ for the 4}^{\text{th}} \text{ quadrant.} \end{aligned} \tag{3}$$

After ω and η are determined, the new coordinates of T in the MCS, indicated by $T_N\ (x_{T_N}, y_{T_N}, z_{T_N})$, are calculated as follows:

$$T_N = K_N . T \tag{4}$$

where K_N is the formula used to define the new coordinates of the endmost point of the end effector in the MCS. K_N that includes the end effector rotations around the Z-axis (η) and the Y-axis (α) is presented as:

$$K_N = Rot(Z,\eta).\,Rot(Y,\omega) = \begin{vmatrix} \cos(\eta)\cos(\alpha) & -\sin(\eta) & \sin(\alpha) \\ \sin(\eta)\cos(\alpha) & \cos(\eta) & \sin(\eta)\sin(\alpha) \\ -\sin(\alpha) & 0 & \cos(\alpha) \end{vmatrix} \tag{5}$$

Then, the new point S relative to the GCS, indicated by $S_G\ (X_{S_G}, Y_{S_G}, Z_{S_G})$, is determined as:

$$S_G = T_G - T_N \tag{6}$$

where T_G represents the point T at the new destination point $U(X_U, Y_U,\ Z_U)$, then $T_G = U$.

The position and orientation of the end effector, represented by points S_G and T_G in GCS, have been determined. Another point in the mobile platform that should be calculated is point R_i $(R_1, \ldots, R_6)$. Point R_i is a joint that connects the mobile platform and the lower link. Point R_i in GCS, which is denoted by R_{i_G}, is determined based on the orientation and position of the end effector. Before calculating Point R_i in GCS, the coordinates of R_i in the MCS can be defined by referring to Fig. 2(f) as follows:

$$R_i = (u_{R_i}, v_{R_i}, w_{R_i}) = (n\cos(\gamma_i), n\sin(\gamma_i), 0) \tag{7}$$

where n denotes the distance between point S and the joint at the mobile platform. Meanwhile, γ_i is the angle of R_i relative to the u-axis. The orientation of the mobile platform in the GCS, owing to the inclination of the end effector, is defined by determining the orientation of the end effector with respect to the x-axis and y-axis. The angles are denoted by τ_x and τ_y, as shown in Fig. 3(b). They are defined as:

$$\tau_x = \tan^{-1}\left(\frac{Y_{S_G} - Y_{T_G}}{Z_{S_G} - Z_{T_G}}\right) \tag{8}$$

$$\tau_y = \tan^{-1}\left(\frac{(X_{S_G} - X_{T_G})\cos(\tau_y)}{Z_{S_G} - Z_{T_G}}\right) \tag{9}$$

After obtaining τ_x and τ_y, the coordinates of the joints at the mobile platform (R_{i_G}) in MCS are defined by:

$$R_{i_G} = [N]_G . R_i; \tag{10}$$

where $[N]_G$ is the formula for transforming the coordinate system of a coordinate at the mobile platform from the MCS to the GCS. It incorporates the rotations around the Y-axis (τ_y) and X-axis (τ_x), and a translation at S_G. It is expressed as follows:

$$[N]_G = Rot(Y,\tau_y).\,Rot(X,\tau_x) + S_G = \begin{bmatrix} \cos(\tau_x) & 0 & \sin(\tau_x) & X_{S_G} \\ -\sin(\tau_x)\sin(\tau_y) & \cos(\tau_x) & -\cos(\tau_x)\sin(\tau_y) & Y_{S_G} \\ -\cos(\tau_x)\sin(\tau_y) & \sin(\tau_x) & \cos(\tau_x)\cos(\tau_y) & Z_{S_G} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{11}$$

4. Calculating the Orientations of the Links

As shown in Fig. 1, the objective of this section is to transform the coordinate of the joint at the upper and lower link from GCS to JCS and finally calculate the orientation of the upper link. The orientation of the upper link is then used to define the rotation angle of the motor at every link. The orientations of the lower and upper links are calculated using the coordinates of points P_i, Q_i, and R_i, as depicted in Fig. 2(b). The coordinates of P_i concerning the GCS are fixed because they are located on the fixed platform. Therefore, their coordinates in the GCS are similar to those at the start position (P_{i0}). Meanwhile, the coordinates of R_i in the GCS (R_{i_G}) have been determined in the previous section (Eq. (10)). Hence, only the coordinates of Q_i remain to be calculated. Point $Q_i(x_{B_i}, y_{B_i}, z_{B_i})$ is calculated in the JCS to simplify the calculation. Then, the coordinates of $R_{i_G}(x_{R_{i_G}}, y_{R_{i_G}}, z_{R_{i_G}})$ and $P_i(x_{P_i}, y_{P_i}, z_{P_i})$ are mapped to the JCS. The centre of the JCS was set at $P_{i_J}(0,0,0)$. The coordinates of R_{i_G} in the JCS (R_{i_J}) are defined as:

$$R_{i_J} = [N]_J . R_{iG} \tag{12}$$

where $[N]_J$ denotes the formula for transforming the cartesian coordinate system from the GCS to the JCS and includes the rotational around the z-axis (δ_i) and the centre movement from coordinate O to P_{i0}. The transformation formula is composed as follows:

$$[N]_J = Rot(Z,\delta_i) - P_i = \begin{bmatrix} \cos(\delta_i) & -\sin(\delta_i) & 0 & -X_{P_i} \\ \sin(\delta_i) & \cos(\delta_i) & 0 & -Y_{P_i} \\ 0 & 0 & 1 & -Z_{P_i} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{13}$$

The coordinates of Q_{i_J} are defined by analyzing the linkages, as well as the upper and lower links. The distance between P_{i_J} and Q_{i_J} from the perspective of the upper-link is given by:

$$L_1 = \vec{Q}_{i_J} - \vec{P}_{i_J} \tag{14}$$

$$L_1^2 = \left(x_{Q_{i_J}} - x_{P_{i_J}}\right)^2 + \left(y_{Q_{i_J}} - y_{P_{i_J}}\right)^2 + \left(z_{Q_{i_J}} - z_{P_{i_J}}\right)^2 \tag{15}$$

where L_1 is the length of the upper-link. Due to the centre point of the JCS placed at P_{i_J}, the coordinates of $P_{i_J}(x_{P_{i_J}}, y_{P_{i_J}}, z_{P_{i_J}})$ are $(0,0,0)$. In this case, the x-axis of the JCS at each joint is aligned with the upper link, as shown in Fig. 2(e), and then, the y-axis of Q_{i_J} $(y_{Q_{i_J}})$ is zero. Hence, Eq. (15) yields the following:

$$L_1^2 = {x_{Q_{i_J}}}^2 + {z_{Q_{i_J}}}^2 \tag{16}$$

Two unidentified variables that remain in Eq. (16) must be defined; they are $x_{Q_{i_J}}$ and $z_{Q_{i_J}}$. An additional formula must be used to define these variables. Then, a formula is composed from the viewpoint of the lower-link, which is given by:

$$L_2 = \vec{R}_{i_J} - \vec{Q}_{i_J} \tag{17}$$

$$L_2^2 = \left(x_{R_{i_J}} - x_{Q_{i_J}}\right)^2 + \left(y_{R_{i_J}} - y_{Q_{i_J}}\right)^2 + \left(z_{R_{i_J}} - z_{Q_{i_J}}\right)^2, \tag{18}$$

where L_2 is the length of the lower-link. By replacing $x_{Q_{i_J}}$ in Eq. (18) with Eq. (16), we obtain:

$$H.z_{Q_{i_J}}{}^2 + V.z_{Q_{i_J}} + W = 0 \tag{19}$$

where,

$$\begin{gathered} H = 4.x_{R_{i_J}}{}^2 + 4.z_{R_{i_J}}{}^2, \\ V = -4\,z_{R_{i_J}}\left(L_2^2 - x_{R_{i_J}}{}^2 - y_{R_{i_J}}{}^2 - z_{R_{i_J}}{}^2 - L_1^2\right), \\ W = \left(L_2^2 - x_{R_{i_J}}{}^2 - y_{R_{i_J}}{}^2 - z_{R_{i_J}}{}^2 - L_1^2\right)^2 \end{gathered} \tag{20}$$

$z_{Q_{i_J}}$ in Eq. (19) can be calculated using a quadratic equation. While Eq. (19) offers two possibilities for $z_{Q_{i_J}}$, however, only one $z_{Q_{i_J}}$ is correct. Based on a set of tests, the smaller value of $z_{Q_{i_J}}$ was correct. After obtaining $z_{Q_{i_J}}$, $x_{Q_{i_J}}$ in Eq. 16 can be defined. At last, the components of Q_{i_J} are completely identified; they are $x_{Q_i}, y_{Q_i}, and\ z_{Q_i}$. Then, the orientation of the upper-link with respect to the JCS, denoted by λ, is given based on Fig. 2(b) by:

$$\lambda_i = \cos^{-1}\left(\frac{z_{Qi}}{L_1}\right) \tag{21}$$

Finally, λ_i is also used to rotate the motor to move the end effector to the desired position and orientation.

5. Implementation and Discussion

All algorithms presented in this paper were simulated in MATLAB. The program was developed following the flow chart shown in Fig. 1. This part tests the formula's performance for defining all the coordinates needed to actuate the end effector to the target position and orientation. The movement of the end effector was performed by six sets of linkages, each of which was driven by a motor. The motors were installed at the uppermost side of the upper-link. The method's reliability in calculating the motors' motion was examined twice, the first using the CAD Software and the second using the physical 6-DOF parallel robot. The detailed procedure for every method is elaborated on in the following subsections.

To calculate the motion of the motors, several coordinates on the part of the robot at the start condition must be defined initially. In addition, other variables describing the robot's dimensions should be calculated. The coordinates of points P_i (P_1,..., P_6), R_i (R_1,..., R_6), S, and T at the start position were obtained based on the design of the robot, as shown in Table 1. The dimensions of the developed robot, including the lengths of the upper and lower links, and the rotation angle of point P_i with respect to the x-axis in the GCS, are also provided in Table 1. In this research, the maximum reach of the end effector (r_{max}) was set at 250 mm. Meanwhile, the maximum inclination angle (ω_{max}) was designed to be 60°.

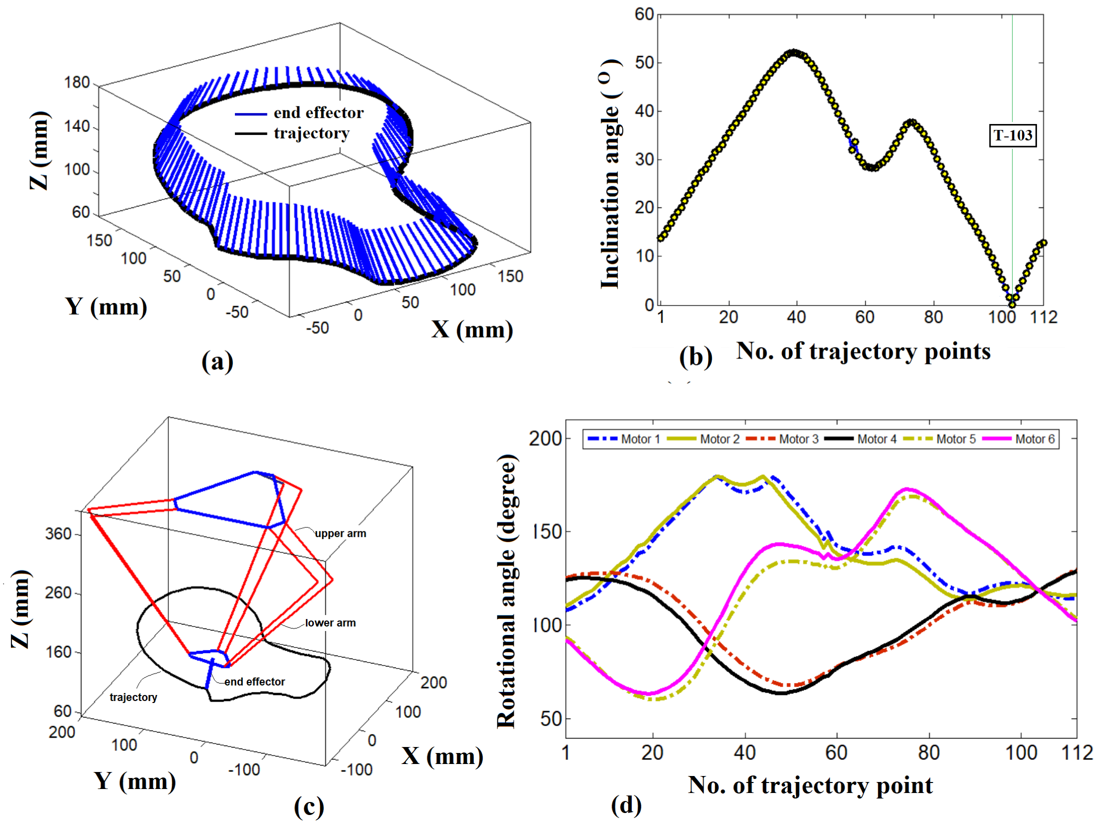

Fig. 4. Model testing: (a) robot trajectory and orientation of the end effector, (b) inclination angle of the end effector concerning the GCS, (c) example of the orientation of the links on the trajectory, and (d) the rotation angle of each motor.

Table 1. Dimension of the robot construction and the coordinates of all the joints at the start position (home position) in GCS.

Coordinates of the joints and the end effector (mm)		Dimensions of the robot
P_1 $(-23.11;\ -27.00;\ 400)$	Q_2 $(-34.51;\ -89.77;\ 250)$	Length of the upper link (L_1): $130\ mm$
P_2 $(-11.87;\ -33.56;\ 400)$	Q_3 $(95.00;\ -15.00;\ 250)$	Length of the lower link (L_2): $280\ mm$
P_3 $(35.00;\ -6.50;\ 400)$	Q_4 $(95.00;\ 15.00;\ 250)$	δ_1 and δ_1: -60^o
P_4 $(35.00;\ 6.50;\ 400)$	Q_5 $(-34.51;\ 89.77;\ 250)$	δ_3 and δ_4: 180^o
P_5 $(-11.87;\ 33.56;\ 400)$	Q_6 $(-60.49;\ 73.23;\ 250)$	δ_5 and δ_6: 60^o
P_6 $(-23.13;\ 27.06;\ 400)$	S $(0;\ 0;\ 250)$	r_{max}: $250\ mm$
Q_1 $(-60.59;-74.77;\ 250)$	T $(0;\ 0;\ 200)$	ω_{max}: 60

5.1. Calculating the rotation of the motor

All mathematical formulations presented in this paper have been used to determine the rotational motion of the motor. A test was carried out to assess the effectiveness and precision of the proposed method. In this test, the end effector was designed to follow a complex path, as presented in Fig. 4(a). This path consisted of 112 destination points. Then, the orientation and position of the end effector when getting through the path, as seen in Fig. 4(a), was computed using the proposed algorithm. The orientation of the end effector changed continuously due to the complexity of the path (trajectory), as presented in Fig. 4(b). The inclination angle of the end effector was calculated regarding the coordinate of the destination point and the coordinate of the start position, as presented in Eq. (1). The graph shows that the farther the destination point to the start position, the larger the inclination angle and vice versa.

When traversing the end effector path, all motors rotate continuously to move the end effector to its destination position and orientation. The orientation of the links and the mobile platform when the end effector at the destination point is illustrated in Fig. 4(c). This illustration was obtained from the computer program developed in MATLAB. The program can also produce the rotation motion of each motor at every point on the design path, as shown in Fig. 4(d). This graph shows that the complexity of the end effector trajectory produces the dynamic change of the rotation angles. Three pairs of motors have almost the same rotation angles (motor-1 and motor-2, motor-3 and motor-4, motor-5 and motor-6). This condition occurred due to the pair of motors and the corresponding links installed near each other, as shown in Fig. 2(c) and Fig. 2(e).

Regarding path number 103 (T-103), as shown in Fig. 4(d), all motors had nearly similar rotation angles. The end effector of T-103 on the path is highlighted by the red line in Fig. 4(a). In this case, the destination points were close to the centre point (X0, Y0), or home position. The closer the destination point is to the centre point, the smaller the difference between the rotation angles of the motors. The inclination angle of the end effector decreases as the destination point approaches the centre position. The inclination angle of T-103 was 0.13, as shown in Fig. 4(b).

5.2. Model verification

The rotation motion of the motors was calculated with respect to the expected motion of the end effector based on the design path using the simulation program. In this study, the accuracy of the developed method was examined using SolidWorks. The rotation angle of the motors, which was obtained by the simulation program, was compared with the rotation angle measured using SolidWorks. In CAD software, the motion of robot design can be simulated. For this verification process, the end effector of the robot design was moved to the destination point, which automatically adjusted the link's position and orientation, as shown in Fig. 5(a). Then, the angle of all upper link relative to the Joint Coordinate System (λ_i) could be measured. Because the motor drives the movement of the upper link, hence, the rotation angles of the motors could be represented by the changing of the upper link angle.

The rotation motion of the motors shown in Fig. 4(d) was used to validate the accuracy of the developed method. The verification was conducted using 23 points selected from the 112 points on the end effector path shown in Fig. 4(a). The corresponding results are presented in Fig. 5(b). The deviations plotted in Fig. 5(b) indicate the differences between the rotation angles measured using Solidworks and those calculated using the computer program. The graph shows that the deviations for all the motors are between –1.2° and 1.3°. The mean deviation is 0.22°, and the standard deviation is 0.485°. The standar deviation is relatively high compared to the mean valued, indicates the errors are relatively spread out. Valencia et al. [18] used the Jacobian-based method to calculate the inverse kinematics of 6 DOF Hexa robots; they found that the average deviation from two simulations is 0.893°. Meanwhile Dehghani et al. [34] found the angle error is up to 4.5° before calibrated and change to less than 4° after calibrated using vision-based calibration. Thus, the verification results show that the deviation of the proposed method is relatively small and proves its accuracy.

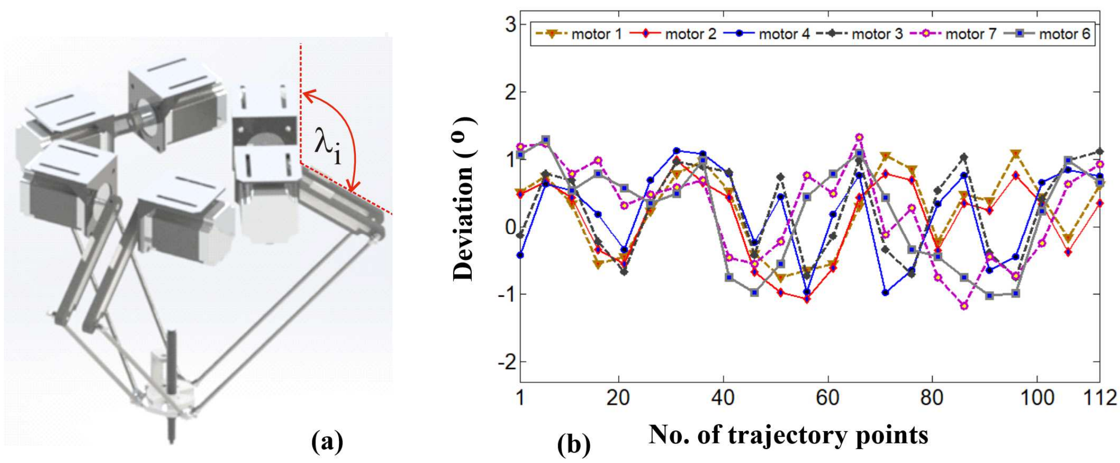

Fig. 5. (a) Model verification using CAD software, (b) Deviations of the rotation angles.

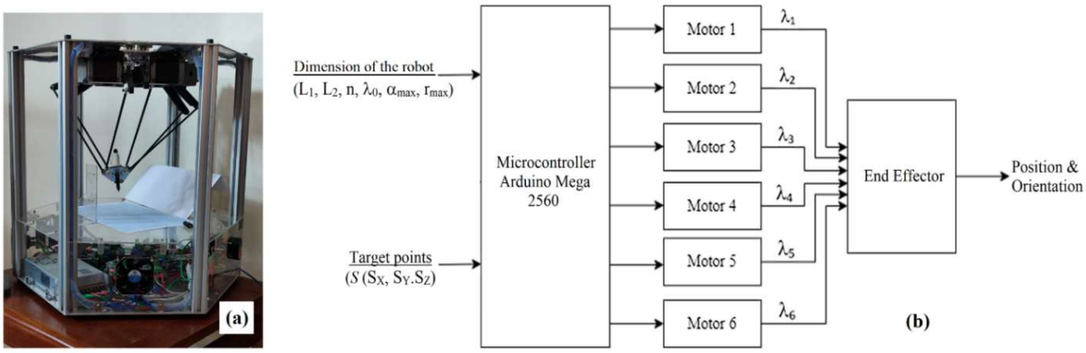

Fig. 6. (a) Construction of a 6-DOF Hexa robot, and (b) block diagram of the control system of the robot.

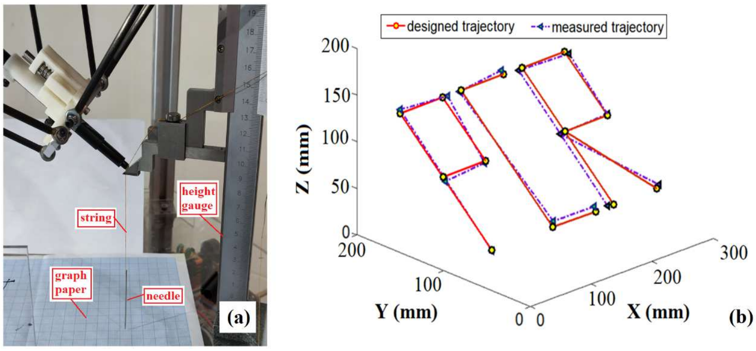

Fig. 7. (a) Experimental setup and (b) test results.

5.3. Application of the proposed method in a robot

The robot design depicted in Fig. 2(a) was constructed as shown in Fig. 6(a). The robot has six upper-links driven using a stepper motor, NEMA 23. Fisheye bearings were used to join the lower link to the upper link and the mobile platform. The robot's control system was developed based on the block diagram shown in Fig. 6(b). The program for calculating the rotation motors was implemented using an Arduino Mega 2560 microcontroller. The program was developed according to the flowchart shown in Fig. 1. Initially, two input parameters must be defined before running the program: the robot's dimensions and destination points.

The accuracy of the 6 DOF Hexa robot in moving to the expected point was verified. The verification tests examined the end effector's accuracy in following the designed path. The end effector coordinates were traditionally measured in this verification test, as shown in Fig. 7(a). The z-axis of the end effector was measured using a Krisbow height gauge. The jaw of the height gauge was set using a string with a needle at the end. They used a string with a needle to depict the x-axis and y-axis. The x-axis- and y-axis were measured using graph paper placed on the fixed platform.

As shown in Fig. 7(b), the end effector was set to follow a tilted letter-shaped trajectory. In this test, the trajectory consists of points. The end effector's coordinates were recorded at each point, and the results were compared with the expected coordinate, as shown in Fig. 7(b). The results demonstrated that the difference was relatively small. The mean squared error (MSE) was used to calculate the deviation, and the result for the two verification tests was 2.28%. The error is believed to be the effect of the small deviation in rotation angle calculation, as shown in Fig. 5(b). Moreover, inaccuracies in the mechanical construction of the robot are also contributed to the error. Valencia et al. [18] performed the test on the Hexa robot and found that the error on the position of the robot (x-axis, y-axis, z-axis) is lower, about 0.8%. Meanwhile, Dehghani et al. [34] performed a series of tests, and the errors for every axis (x-axis, y-axis, and z-axis) are relatively high, up to 20 mm. Based on the tests' results, it can be concluded that the method developed was successfully applied to the physical robot and relatively accurate. Nevertheless, the current study's limitation is that the robot's dynamic aspects have not been considered yet. This might be another factor that causes the errors.

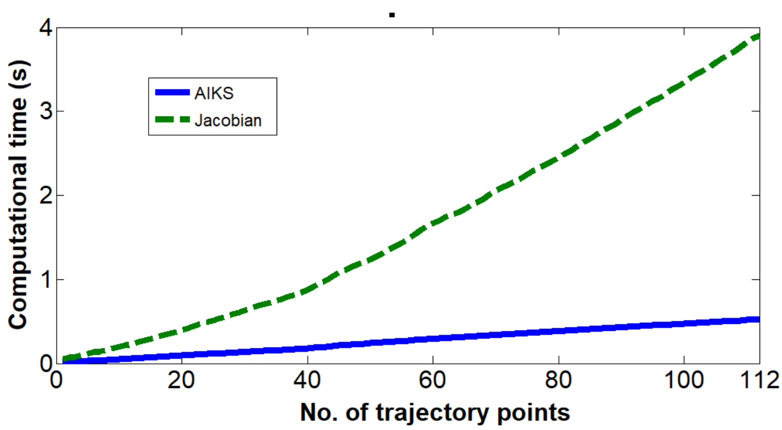

Fig. 8. Computational time of AIKS and Jacobian method.

5.4. Comparison test on computational time

As mentioned in the introduction, the advantage of the proposed method, which is called AIKS, compared to the numerical-based approach is the efficiency of computational time. Therefore, the computational time of the developed method was tested and compared to the Jacobian method. In this test, the computational time to define the kinematics aspect of the robot in moving through 112 points, as discussed in Section 5.1, was measured. The test was performed using MATLAB in an Intel Core i7 GHz laptop with 8 GB RAM and repeated three times. The average results are presented in Fig. 8. The result shows that the AIKS took a total time of 0.52 seconds. Meanwhile, the Jacobian tool took 3.9 seconds. The comparison demonstrated that the proposed method was computationally more efficient than the numerical method. In this study, the AIKS method proved applicable in supporting inverse kinematics calculations. It means that the proposed method simplifies the work needed to calculate the inverse kinematics of a parallel manipulator.

6. Conclusion

This research proposed a method to compute the inverse kinematics of a 6-DOF Hexa parallel robot. The robot was developed using six independent links, each constructed by two links: upper-link and lower-link. An analytical-based algorithm proposed in this paper was applied to calculate the rotation angles of the motors to actuate the end effector for the design orientation and position. The end effector was inclined based on the coordinates of the destination point. The aim was to enlarge the robot's reach and increase the manipulator's flexibility. Several tests were carried out to validate the applicability and accuracy of the proposed approach, and the main results of this study are summarized as follows:

1. A simulation program was created in MATLAB using the proposed algorithm. The program then defined the motor rotation needed to actuate the robot's moving component while it followed a complex path. The tests demonstrated that the program could provide all the motors' rotation angles. The program can also be used to generate the inclination angle of the end effector.
2. Two methods were used to assess the accuracy of the proposed algorithm. The first method showed that the differences between the rotation angles measured with CAD software and those produced by the simulation program were minimal. It indicates the accuracy of the developed method.
3. The second accuracy test was performed by implementing the proposed algorithm in a physical robot. The test revealed that the robot could accurately move to the destination points, demonstrating that the developed algorithm can be applied to a real robot.
4. The comparison test proved that the proposed method shortens the computational time. This is the main advantage of the method compared to the numerical methods.

Implementation tests and validation showed that the proposed method could be used accurately to define the kinematics aspects of 6 of hexa parallel robots. For future work, this method will be extended to develop a method that considers the robot's dynamics.

Author Contributions

H. Hendriko developed the mathematical algorithm of the proposed method; N. Khamdi developed the program in MATLAB and the electronic aspect of the robot; J. Jaenudin developed the robot's control system and tested the method on the real robot; R. Novison developed the robot's mechanical construction and tested the method on the real robot; M. Ikhsan designed the robot using CAD software and tested the proposed method in CAD software. All authors discussed the results and reviewed and approved the final version of the manuscript.

Acknowledgements

The authors wish to thank the Indonesian Ministry of Research Technology and Higher Education for the financial support provided through the fundamental research grant scheme.

Conflict of Interest

The authors declared no potential conflicts of interest concerning this article's research, authorship, and publication.

Funding

The authors received financial support from the Indonesian Ministry of Research Technology and Higher Education and Politeknik Caltex Riau for the research and publication of this article.

Data Availability Statements

The datasets generated or analyzed during the current study are available from the corresponding author upon reasonable request.

Nomenclature

DOF	Degrees of freedom
RSS	Revolute, spherical, spherical
CCD	Cyclic coordinate descent
GCS	Global Cartesian System
MCS	Moving Cartesian System
JCS	Joint Cartesian System
K_N	Matrix operator to transform MCS to CGS around Z-axis (η) and Y-axis (ω)
$[N]_J$	Matrix operator to transform GCS to JCS
$[N]_G$	Matrix operator to transform MCS to CGS
L_1	Length of the upper-link
L_2	Length of the lower-link
r	The reach of the end effector
ω	Inclination angle about the X-axis
η	Rotation angle measured from the x-axis to point U
γ_i	Angle of R_i

References

[1] Glozman, D., Shoham, M., Novel 6-DOF parallel manipulator with large workspace, *Robotica*, 27(6), 2009, 891.
[2] Cox, D.J., *The dynamic modeling and command signal formulation for parallel multi-parameter robotic devices*, Doctoral Dissertation, University of Florida, 1981.
[3] Falezza, F., Vesentini, F., Di Flumeri, A., Leopardi, L., Fiori, G., Mistrorigo, G., Muradore, R., A novel inverse dynamic model for 3-DoF delta robots, *Mechatronics*, 83, 2022, 102752.
[4] Ed-Deryouch, A., Ech-Chhibat, M.E., Mansouri, K., The Inverse Kinematics of 3-DOF Planar Robotic Arm, *In 2023 7th IEEE Congress on Information Science and Technology (CiSt)*, 2023.
[5] Guan, J., Deng, J., Zhang, S., Liu, J., Liu, Y., A spatial 3-DOF piezoelectric robot and its speed-up trajectory based on improved stick-slip principle, *Sensors and Actuators A: Physical*, 374, 2024, 115502.
[6] Tsai, C.Y., Yu, C.P., Yeh, P.C., Lan, C.C., Parametric joint compliance analysis of a 3-UPU parallel robot, *Mechanism and Machine Theory*, 170, 2022, 104721.
[7] Chen, G., Mao, Z., Zhou, H., Yang, P., Design and control strategy of 3-prismatic-revolute-spherical ankle rehabilitation robot, *Australian Journal of Mechanical Engineering*, 21(3), 2023, 1079-1092.
[8] Nunez, D.A., Gonzalez, S., Guacheta, J., Mauledoux, M., Aviles, O., Kinematics Parallel Mechanisms Design Particularities Focused on Additive Manufacturing, *Journal of Engineering Science & Technology Review*, 14(4), 2021, 207–218.
[9] Isaksson, M., Brogårdh, T., Watson, M., Nahavandi, S., Crothers, P., The Octahedral Hexarot—A novel 6-DOF parallel manipulator, *Mechanism and Machine Theory*, 55, 2012, 91-102.
[10] Gough, V.E., Universal tyre test machine, *Proceeding FISITA 9th Int. Technical Congress*, London, 1962.
[11] Pedrammehr, S., Qazani, M.R.C., Asadi, H., Ettefagh, M.M., Nahavandi, S., Model-based control of axisymmetric hexarot parallel manipulators, *Results in Control and Optimization*, 7, 2022, 100135.
[12] Antonov, A., Fomin, A., Glazunov, V., Ceccarelli, M., Workspace and performance analysis of a 6-DOF hexapod-type manipulator with a circular guide, *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 236(18), 2022, 9951-9965.
[13] Tunc, L.T., Shaw, J., Experimental study on investigation of dynamics of hexapod robot for mobile machining, *The International Journal of Advanced Manufacturing Technology*, 84, 2016, 817-830.
[14] Zhang, G., Ma, S., Liu, J., Zeng, X., Kong, L., Li, Y., Q-Whex: A simple and highly mobile quasi-wheeled hexapod robot, *Journal of Field Robotics*, 40(6), 2023, 1444-1459.
[15] Huynh, B.P., Kuo, Y.L., Dynamic Hybrid Filter for Vision-Based Pose Estimation of a Hexa Parallel Robot, *Journal of Sensors*, 2021(1), 2021, 9990403.
[16] Wu, G., Dong, H., Kinematics of a 6-R UU Parallel Robots with Reconfigurable Platforms, *In Computational Kinematics: Proceedings of the 7th International Workshop on Computational Kinematics that was held at Futuroscope-Poitiers*, France, Springer International Publishing, in May 2017 (pp. 331-339), 2018,
[17] Alvarado, R.R., Castañeda, C.E., Type synthesis of kinematically redundant parallel manipulators based on the HEXA parallel robot, *In IFToMM Symposium on Mechanism Design for Robotics*, Springer International Publishing, 2018, 227-234.
[18] Valencia, A.J., Mauledoux, M., Castañeda, C., Inverse, Direct kinetics and differential kinematic control of parallel robot with six degrees of hexa freedom–hunt, *In 2019 IEEE 10th International Conference on Mechanical and Aerospace Engineering (ICMAE)*, July, 2019.
[19] Dehghani, M., Ahmadi, M., Khayatian, A., Eghtesad, M., Farid, M., Neural network solution for forward kinematics problem of HEXA parallel robot, *In 2008 American Control Conference*, IEEE, June, 2008.
[20] Nunez, D.A., Mauledoux, M., Aviles, O., Guacheta, J., Gonzalez, S., Optimal Design of Kinematics Parallel Manipulator Considering Workspace and Control Effort, *International Journal of Mechanical Engineering and Robotics Research*, 11(4), 2022, 234-240.
[21] Pierrot, F., Dauchez, P., Fournier, A., HEXA: a fast six-DOF fully-parallel robot, *In Fifth International Conference on Advanced Robotics' Robots in Unstructured Environments*, (pp. 1158-1163), IEEE, 1991.
[22] Kucuk, S., Bingul, Z., Inverse kinematics solutions for industrial robot manipulators with offset wrists, *Applied Mathematical Modelling*, 38(7-8), 2014, 1983-1999.
[23] Campa, R., Bernal, J., Soto, I., Kinematic modeling and control of the hexapod parallel robot, *In 2016 American Control Conference (ACC)*, IEEE, July, 2016.
[24] Gordon, R., Worrall, K., Ceriotti, M., Broyden's Method for Faster Inverse Simulation, *Journal of Guidance, Control, and Dynamics*, 2024, 1-7.
[25] Zhao, W., A Broyden–Fletcher–Goldfarb–Shanno algorithm for reliability-based design optimization, *Applied Mathematical Modelling*, 92, 2021, 447-465.
[26] Ojha, S., Leodler, K., Wu, T.H., Singularity-Free Inverse Kinematics by Cyclic Coordinate Descent of A 6 DOF Robotic Manipulator, *In 2023 20th International Conference on Ubiquitous Robots (UR)*, IEEE, June, 2023.
[27] Wu, H., Yu, J., Pan, J., Pei, X., A novel obstacle avoidance heuristic algorithm of continuum robot based on FABRIK, *Science China Technological Sciences*, 65(12), 2022, 2952-2966.
[28] Ye, H., Wang, D., Wu, J., Yue, Y., Zhou, Y., Forward and inverse kinematics of a 5-DOF hybrid robot for composite material machining, *Robotics and Computer-Integrated Manufacturing*, 65, 2020, 101961.
[29] Kofinas, N., Orfanoudakis, E., Lagoudakis, M.G., Complete analytical forward and inverse kinematics for the NAO humanoid robot, *Journal of Intelligent & Robotic Systems*, 77, 2015, 251-264.
[30] Hendriko, H., A hybrid analytical and discrete based methodology to calculate path scallop of helical toroidal cutter in five-axis milling, *FME Transactions*, 46(4), 2018, 552-559.
[31] Hendriko, H., Kiswanto, G, Istiyanto, J., Duc, E., Implementation of analytical boundary simulation method for cutting force prediction model in five-axis milling, *Machining Science and Technology*, 22(1), 2018, 163-179.
[32] Hendriko, H., Khamdi, N., Jaenudin, J., Implementation of analytical inverse kinematics simulation method for 5-dof parallel manipulator, *Journal of Engineering Science and Technology*, 14(4), 2015, 1948-1959.
[33] Hendriko, H., Khamdi, N., Jajang, J., Muthahar, I.M., Analytical based inverse kinematic method for 5-axis delta robot, *International Journal of Materials, Mechanics, and Manufacturing*, 6, 2018, 264-267,
[34] Dehghani, M., Ahmadi, M., Khayatian, A., Eghtesad, M., Yazdi, M., Vision-based calibration of a Hexa parallel robot, *Industrial Robot: An International Journal*, 41(3), 2014, 296-310.

ORCID iD

Hendriko Hendriko https://orcid.org/0000-0002-6270-1342
Nur Khamdi https://orcid.org/0000-0003-3543-0716
Jajang Jaenudin https://orcid.org/0000-0001-9602-5576
Roni Novison https://orcid.org/0009-0006-0738-8978
Muhammad Ikhsan https://orcid.org/0009-0007-8370-7130

How to cite this article: Hendriko H., et al. Analytical Inverse Kinematics-Based Method for a 6-DOF Hexa Parallel Robot, *J. Appl. Comput. Mech.*, 11(4), 2025, 1172-1182. https://doi.org/10.22055/jacm.2025.48048.4945