The International Journal of Advanced Manufacturing Technology (2023) 126:813–825
https://doi.org/10.1007/s00170-023-11130-9

ORIGINAL ARTICLE

# Forward kinematics solution for a general Stewart platform through iteration based simulation

**Sourabh Karmakar**[1] · **Cameron J. Turner**[1]

Received: 27 September 2022 / Accepted: 16 February 2023 / Published online: 27 February 2023


## Abstract

This paper presents a method to generate feasible, unique forward-kinematic solutions for a general Stewart platform. This is done by using inverse kinematics to obtain valid workspace data and corresponding actuator lengths for the moving platform. For parallel kinematic machines, such as the Stewart platform, inverse kinematics are straight forward, but the forward kinematics are complex and generates multiple solutions due to the closed loop structure of the kinematic links. In this research, a simple iterative algorithm has been used employing modified Denavit-Hartenberg convention. The outcome is encouraging as this method generates a single feasible forward kinematic solution for each valid pose with the solved DH parameters and unlike earlier forward kinematics solutions, this unique solution does not need to be manually verified. Therefore, the forward kinematic solutions can be used directly for further calculations without the need for manual pose verification. This capability is essential for the six degree of freedom materials testing system developed by the authors in their laboratory. The developed system is aimed at characterizing additively manufactured materials under complex combined multiple loading conditions. The material characterization is done by enabling high precision force control on the moving platform via in situ calibration of the as-built kinematics of the Stewart Gough platform.

**Keywords** Stewart platform · Inverse kinematics · Forward kinematics · Denavit-Hartenberg convention · Transformation matrix

## 1 Introduction

The Stewart platform is one of the most popular parallel kinematic machines (PKMs) [1]. Though there are PKMs with 3, 4, 5, and 6 parallel links, 6 parallel linked stationary PKMs are termed as Stewart platform or hexapod and are most versatile among the PKMs because of having six degrees of freedom available in a compact machine [2]. A typical hexapod consists of one fixed base plate and one movable plate or platform connected by six actuators. Each of the six parallel actuators generally provides one degree of freedom (DOF) to the machine by three translations along $x$, $y$, and $z$ axes and three rotations about $x$, $y$, and $z$ axes when a Cartesian reference frame is attached to the movable platform center [3]. The hexapod considered here has each prismatic (P) actuator connected to the base with a universal (U) joint and the other end is connected to the moving platform by a spherical (S) joint. Such a hexapod or Stewart platform is designated as 6-UPS parallel kinematic machine [4]. The movement of the platform in 6-DOF through the movement of six actuators makes the motion control complex compared to other machines.

To provide position and motion control, hexapod platforms use either inverse kinematics or forward kinematics. The position and orientation of the platform center point is known as platform pose [5]. In inverse kinematics, the actuator lengths are calculated based on the platform pose [6]. In forward kinematics, the platform pose is calculated for a given set of the actuator lengths and joint angles [7]. The mathematics and solution of inverse kinematics is much easier than forward kinematics for a hexapod. The complexity of forward kinematics, also called direct kinematics, is generated due to highly non-linear kinematic equations with multiple solutions. To solve the forward kinematics, many researchers tried different ways to solve the non-linear

✉ Cameron J. Turner
cturne9@clemson.edu

Sourabh Karmakar
skarmak@clemson.edu

[1] Mechanical Engineering, Clemson University, Clemson, SC, USA

problem. A 16th-degree univariate polynomial on the 6–3 type PKM had been formulated by Innocenti and Parenti-Castelli [8]. Huang et al. [9] presented algebraic method for a general 6–6 Stewart platform that yielded a 20th degree univariate polynomial from the determinant of the $15 \times 15$ Sylvester's matrix. Husty [10] derived a 40th-degree univariate equation for a general 6–6 Stewart platform, by finding the greatest common divisor of the intermediate polynomials. Domagoj and Jelenkovicti [7] used canonical formulation for the forward kinematics and derived 9 equations with 9 unknowns and then solved it by multiple optimization methods. Wang and Yunfeng [11] derived the direct kinematics solutions for calculating the platform pose by increasing the actuator lengths in small amount and then increasing the joint parameters also in small amount utilizing numerical methods. Cardona [12] calculated the possible poses using Newton–Raphson method for a set of joint angles and actuator lengths which includes invalid and valid poses for the platform and those need to be checked manually for acceptance. Also, other researchers [13, 14] used Newton-Rapson method to find the forward kinematic solution for parallel robots. The accuracy of convergence obtained in those works is at different levels based on the initial guesses. Zhou et al. [15] created pose error model with the help of Denavit-Hartenberg (DH) parameters and then converted it to a constrained quadratic optimization problem. In another research by Tarokh [16] used an approach to generate a lookup-table with possible solution space data. This solution space is divided into multiple clusters. For forward kinematic solution, the system looks at lookup-table clusters to get the required data directly or from the fitted curves with the available.

Other research on forward kinematics for Stewart platform used algebraic elimination [17], interval analysis, multiple optimization techniques, continuation algebraic formulations to generate solutions for a set of nonlinear equations, or high degree of polynomials. Some researchers utilized neural network algorithms [18–21] and artificial intelligence (AI) [22] for improving the accuracy of the hexapod platform solving forward kinematics problem. All these works generated algorithms to obtain a valid solution for various types of PKMs, but finding a single feasible practical solution is still a challenging problem and limited for real-time applications.

The proposed method in this paper uses inverse kinematics to solve forward kinematics using modified Denavit-Hartenberg (DH) convention. DH convention is the most popular method for forward kinematics in serial manipulators, but its application remains very limited in parallel manipulators due to the closed loop nature of PKMs. The authors adopted this convention because of its simplicity and straightforward nature of implementation. The proposed algorithm in this paper generates a single feasible orientation solution for a general Stewart platform calculated by forward kinematics from a set of input data. The solution does not need to be validated through manual inspection. This is a simple iterative method using the available information from inverse kinematics. The pose data and corresponding actuator lengths are stored in a database. Then based on the motion limits of each joint, the algorithm generates the DH parameter dataset consisting of joint angles for the platform pose through iterative forward kinematics. The authors tried to exploit the power of the latest generation of computing systems by using simple iterative method which is not significantly more time-consuming method in the present days compared to the other efficient optimization methods like Newton-Rapson method. Another advantage of this simple iterative method is that there is no initial guesses and no doubt about getting a solution (convergence). As long as a pose exists, a solution must be available.

The rest of the paper is organized into the following parts. The first part serves as an introduction. The next part elaborates the general mathematical expressions for a Stewart platform. Sections 3 and 4 explain the inverse kinematics and forward kinematics used in the calculations. Section 5 explains the method that has been used in the algorithm to simulate the desired results for a Stewart platform-based test frame "Tiger 66.1". Section 6 contains the result and discussion, and finally, the document ends with the conclusion.

## 2 Platform pose and workspace

Figure 1 shows the typical sketch of a hexapod platform. The circular plates at bottom and top are fixed base and moving platform respectively. Two Cartesian coordinate frames are attached at the center of each circular plate. The platform frame is defined by $\boldsymbol{P_x}$, $\boldsymbol{P_y}$, and $\boldsymbol{P_z}$ axes with origin $\boldsymbol{O_P}$ and the base coordinate frame is expressed by $\boldsymbol{B_x}$, $\boldsymbol{B_y}$, and $\boldsymbol{B_z}$ axes with origin $\boldsymbol{O_B}$. $\boldsymbol{P_z}$, and $\boldsymbol{B_z}$ are denoting the vertical axes of the respective frames. The orientation of the moving platform is defined by the orientation of the $\boldsymbol{O_P P_x P_y P_z}$ frame with respect to base coordinate frame $\boldsymbol{O_B B_x B_y B_z}$. The position of the platform center with respect to base center is defined by vector $\boldsymbol{h}$. In one condition, the base and platform frames remain parallel; actuators are at their smallest length and the z-axes are colinear. This orientation is called '*home pose*' [23]. In this condition, all the six actuators are of same length. Once the platform is moved by controlling the lengths of the actuators, the resultant motion at the center of the platform is a translation or rotation or a combination of both.

The position and orientation of the platform center depends on the values of roll, pitch, yaw, and the translation motion along $x$, $y$, and $z$ axes as per the Euler angle representations [24]. The rotations are expressed by vector $\Phi$ and

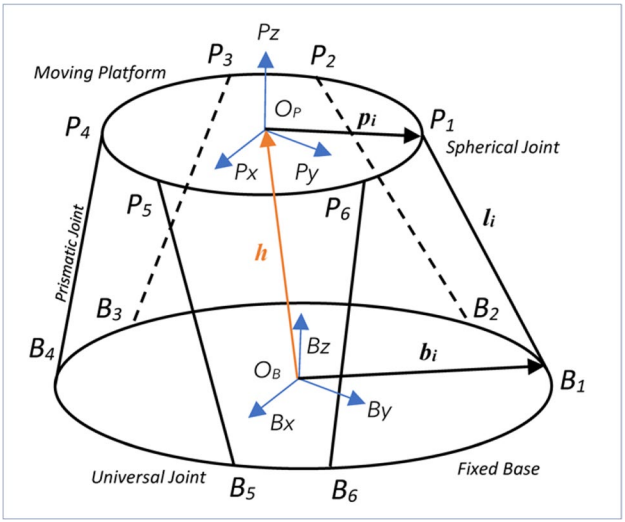


**Fig. 1** A typical hexapod configuration

$$\Phi = (\alpha\ \beta\ \gamma)^T \tag{1}$$

where $\alpha$ (roll), $\beta$ (pitch), $\gamma$ (yaw) denotes the rotation angles about $x$, $y$ & $z$ axes respectively.

The translations are expressed by vector $d$ where

$$d = (dx\ dy\ dz)^T \tag{2}$$

$dx$, $dy$, and $dz$ are the translation values along $x$, $y$, and $z$ axes respectively.

The relationships between different platform poses and actuator variables are expressed by forward kinematics and inverse kinematics. In forward kinematics, the platform poses are calculated by the length and orientations of the six actuators. It can be expressed by Eq. (3):

$$[dx, dy, dz, \alpha, \beta, \gamma]^T = f(q_1, q_2, q_3, \ldots\ldots, q_n) \tag{3}$$

where $dx, dy, dz, \alpha, \beta, \gamma$ are platform pose and $q_1, q_2, q_3 \ldots q_n$ are link variables that includes joint angles and actuator lengths.

In inverse kinematics, actuator lengths are calculated for a platform pose by Eq. (4).

$$q_i = f_i(dx, dy, dz, \alpha, \beta, \gamma) \tag{4}$$

where $i = 1 \ldots n$ are link numbers. For a Stewart platform, the value for $i = 1$ to 6.

Mathematically, the rotations about each axis are represented by the following equations as per Euler angle representation:

$$R_{x,\alpha} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\alpha & -s\alpha \\ 0 & s\alpha & c\alpha \end{bmatrix} \tag{5a}$$

$$R_{y,\beta} = \begin{bmatrix} c\beta & 0 & s\beta \\ 0 & 1 & 0 \\ -s\beta & 0 & c\beta \end{bmatrix} \tag{5b}$$

$$R_{z,\gamma} = \begin{bmatrix} c\gamma & -s\gamma & 0 \\ s\gamma & c\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{5c}$$

here ***s*** and ***c*** represent ***sine*** and ***cosine*** functions respectively.

Combining all these rotations, the complete rotation of the platform center with respect to its original fixed frame axes is calculated. The combined rotation denoted by $\boldsymbol{R}$ is calculated by pre-multiplying each subsequent rotation.

$$\begin{aligned} \boldsymbol{R} &= R_{z,\gamma}.R_{y,\beta}.R_{x,\alpha} \\ &= \begin{bmatrix} c\beta c\gamma & s\alpha s\beta c\gamma - c\alpha s\gamma & c\alpha s\beta c\gamma + s\alpha s\gamma \\ c\beta s\gamma & s\alpha s\beta s\gamma + c\alpha c\gamma & c\alpha s\beta s\gamma - s\alpha c\gamma \\ -s\beta & s\alpha c\beta & c\alpha c\beta \end{bmatrix} \end{aligned} \tag{6}$$

Equation (6) represents the final rotation matrix. This is further combined with the $x$, $y$, $z$ translations, and the homogeneous transformation matrix (HTM) [25] in Eq. (7) is obtained. The complete HTM is expressed as

$$\begin{aligned} \boldsymbol{H} &= \begin{pmatrix} \boldsymbol{R} & d \\ 0 & 1 \end{pmatrix} \\ \therefore \boldsymbol{H} &= \begin{bmatrix} c\beta c\gamma & s\alpha s\beta c\gamma - c\alpha s\gamma & c\alpha s\beta c\gamma + s\alpha s\gamma & dx \\ c\beta s\gamma & s\alpha s\beta s\gamma + c\alpha c\gamma & c\alpha s\beta s\gamma - s\alpha c\gamma & dy \\ -s\beta & s\alpha c\beta & c\alpha c\beta & dz \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned} \tag{7}$$

## 3 Inverse kinematics and workspace

The platform pose with respect to the base center defines the state of the system. In inverse kinematics, the actuator lengths $\boldsymbol{l_i}$ for a particular platform pose are calculated. With the change of vectors $\Phi$ and $d$ the coordinate of the platform center $\boldsymbol{O_P}$, platform joint vectors $\boldsymbol{p_i}$, and vectors $\boldsymbol{h}$, $\boldsymbol{l_i}$ change. As the base is fixed, the base center vector $\boldsymbol{O_B}$ and base joint vectors $\boldsymbol{b_i}$ remain unchanged. The new values of $\boldsymbol{l_i}$ and $\boldsymbol{h}$ are calculated by calculating new values of $\boldsymbol{p_i}$ and $\boldsymbol{O_P}$ by finding value of $\boldsymbol{H}$ in Eq. (7) and pre-multiply to the respective coordinate values before the change. Then the length of each actuator is obtained by using Eq. (8) [26]

$$\boldsymbol{l_i} = ||\boldsymbol{h} + {}^{\boldsymbol{B}}\boldsymbol{R_P}.{}^{\boldsymbol{B}}\boldsymbol{p_i} - \boldsymbol{b_i}|| \tag{8}$$

where ${}^{B}\boldsymbol{R}_{P}$ denotes the rotation vector to express the rotation of the platform coordinate frame with respect to the base coordinate frame, and ${}^{B}\boldsymbol{p}_{i}$ is the platform joint vectors expressed with respect to the base coordinate frame.

For each PKM, the actuators operate between fixed length limits. Conforming to the values of $\boldsymbol{l}_{i}$ for each new pose obtained by changing vectors $\Phi$ and $d$ define the feasible movable points for the platform center. All these feasible points form the working space for the selected hexapod platform.

Figure 2 shows the flowchart of the inverse kinematics calculations used by the authors to generate the valid workspace points. An important aspect of the platform pose is singularity. In singular condition, the end-effector gains one or more unwanted instantaneous degrees of freedom (DOF), and the platform pose cannot be determined by unique actuator lengths. In such situation, the PKM becomes out of control and can transitorily make the drive force go infinity [27]. A pose with singularity is not considered as a valid workspace point and discarded from further calculations. The mathematical check for singularity is included inside the calculation code by checking if the determinant of the force Jacobian matrix in that pose is zero or not [28].

## 4 Forward kinematics and valid poses

The Denavit-Hartenberg (DH) convention [29] is one of the most popular and earliest ways for solving forward kinematics for any serial manipulator in 3D space. For each pair of links, there are 4 DH parameters that are used in DH matrix to transfer the coordinate of a point from one coordinate frame to another.

The initial DH convention was introduced in 1955 by Jacques Denavit and Richard Hartenberg. In due course of time, it has been revisited by researchers, and a modified DH convention [30] has been introduced. In this paper, modified DH convention has been used. The 4 DH parameters as per modified conventions are (Table 1)

Here $\boldsymbol{i}$ denotes the joint in consideration, and $(\boldsymbol{i}-1)$ is the previous joint.$\boldsymbol{i}$ is always a positive integer. ${}^{i-1}\boldsymbol{T}_{i}$ describes the transformation matrix for frame $\boldsymbol{i}$ relative to frame $(\boldsymbol{i}-1)$. Using these DH parameters, the frame transformation matrix is calculated by Eqs. (9) and (10) [31].

$$ {}^{i-1}T_{i} = R_{x}(\alpha_{i-1}).T_{x}(a_{i-1}).R_{z}(\theta_{i}).T_{z}(r_{i}) \tag{9} $$

$$ \textbf{Or}, {}^{i-1}T_{i} = \begin{bmatrix} cos\theta_{i} & -sin\theta_{i} & 0 & a_{i-1} \\ sin\theta_{i}cos\alpha_{i-1} & cos\theta_{i}cos\alpha_{i-1} & -sin\alpha_{i-1} & -r_{i}sin\alpha_{i-1} \\ sin\theta_{i}sin\alpha_{i-1} & cos\theta_{i}sin\alpha_{i-1} & cos\alpha_{i-1} & r_{i}cos\alpha_{i-1} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{10} $$

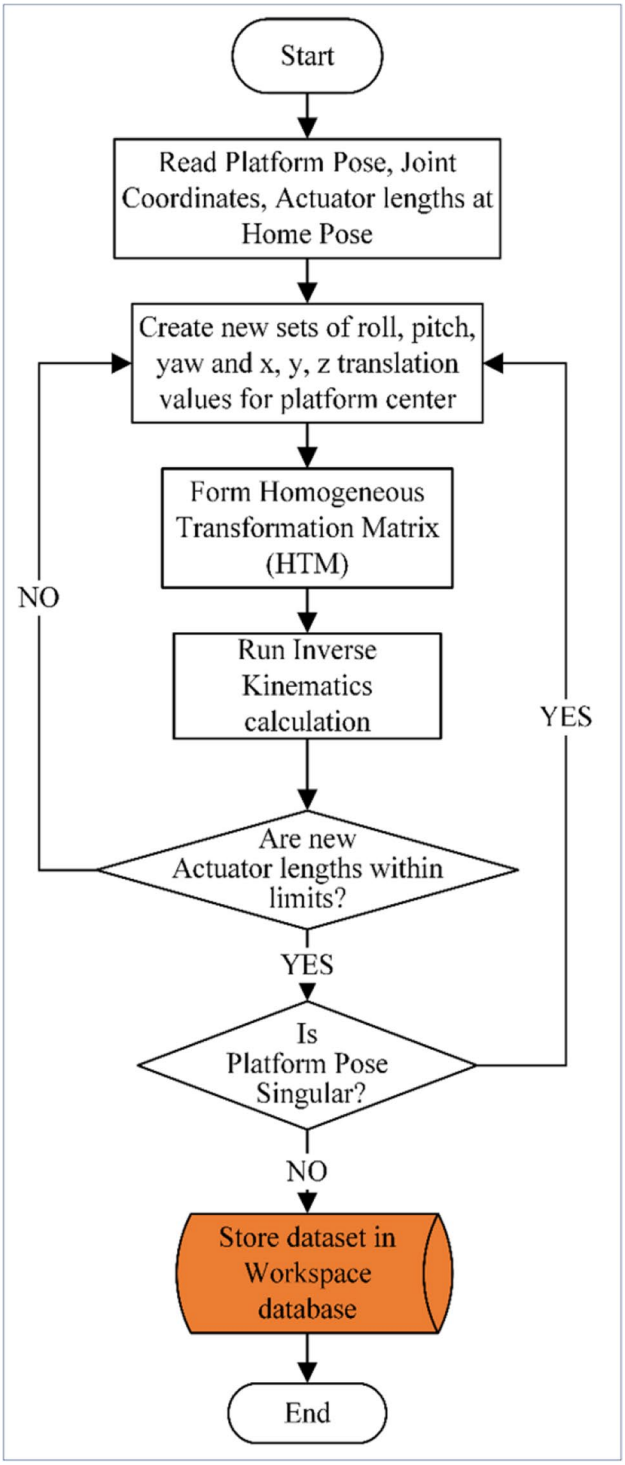


**Fig. 2** Flowchart for workspace calculation by inverse kinematics

**Table 1** DH Parameters descriptions

| Modified DH parameters | | | | |
|---|---|---|---|---|
| Link ($i$) | $a_{i-1}$ | $\alpha_{i-1}$ | $r_{i}$ | $\theta_{i}$ |
| Parameter name | Link length | Link twist | Link offset | Joint angle |

If there are, suppose, 4 joints, they are numbered from 0 to 3. A point $P$ is expressed with respect to the end coordinate frame {3} as ${}^{3}P = [P_x\ P_y\ P_z]^T$. The transformation matrix to express frame {3} with respect to the base frame {0} is calculated by Eq. (11).

$$ {}^{0}T_3 = {}^{0}T_1.{}^{1}T_2.{}^{2}T_3 \tag{11} $$

And the point ${}^{3}P$ with respect to the base frame {0} is calculated Eq. (12).

$$ {}^{0}P = {}^{0}T_3.{}^{3}P \tag{12} $$

In the workspace calculation by inverse kinematics, the valid actuator lengths are calculated for each valid pose of the platform, and these data are stored in the workspace database. For calculating the DH parameters, the MATLAB code follows the flowchart shown in Fig. 3. The code iterates different combinations of joint angles $\boldsymbol{\theta_i}$ to a reach to the pose coordinate using the corresponding actuator lengths available in the workspace database. The angles $\boldsymbol{\alpha_{i-1}}$ are defined by the Cartesian frame attached at each joints following modified DH convention rules, and these values are not going to change for the whole process. A predefined error value is set in the calculation for the pose error which measures the distance between the pose in the database and pose calculated through forward kinematics. The forward kinematics calculation is repeated several times by changing $\boldsymbol{\theta_i}$. Once the pose error is achieved below the predefined value, the DH parameter sets are stored in the DH parameter database for the pose. In this process, there are some instances when the angle combinations cannot construct an orientation with a pose error below the defined error value, that pose is recorded without a valid DH parameter set.

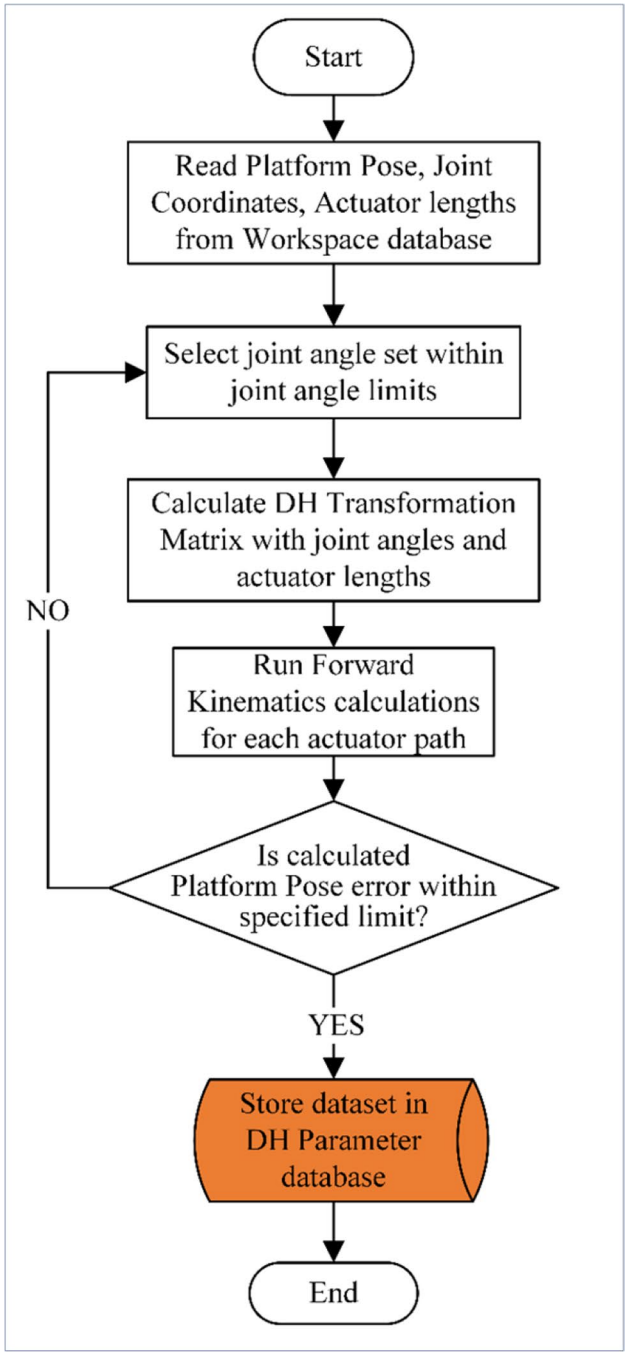


**Fig. 3** Flowchart for DH parameters calculation by forward kinematics

## 5 Simulation results for "Tiger 66.1"

The above method of DH parameter calculation has been tried out through simulation for the hexapod test frame "Tiger 66.1" developed by the authors in their lab. Tiger 66.1 is a special Stewart platform–based test frame developed for full-field characterization of additively manufactured specimens. A CAD model of the test frame is shown in Fig. 4. A partial sketch also shows a couple of critical dimensions. The test process uses photogrammetry, so there are four cameras (2) mounted to capture data from the test zone. This restricts the movements of the platform center and overall workspace. The two green blocks on the upper part of the system are the grippers for holding the material specimen (4) to be tested. The upper gripper (1) is mounted on the fixed frame and the lower gripper (3) is fixed at the center of the hexapod top moving platform. All motions and forces are applied on the test specimen by moving the lower gripper.

Tiger 66.1 uses 6-UPS link combinations. As per the design and construction, the base frame center is situated 71.55 mm about the world frame origin which is located on the center of the circular plate at the lower structure and platform home pose is considered at a point 343.20 mm above the actual platform center point. This dimension does not change. The test frame lower and upper grip centers are considered at this point which is the condition of the system

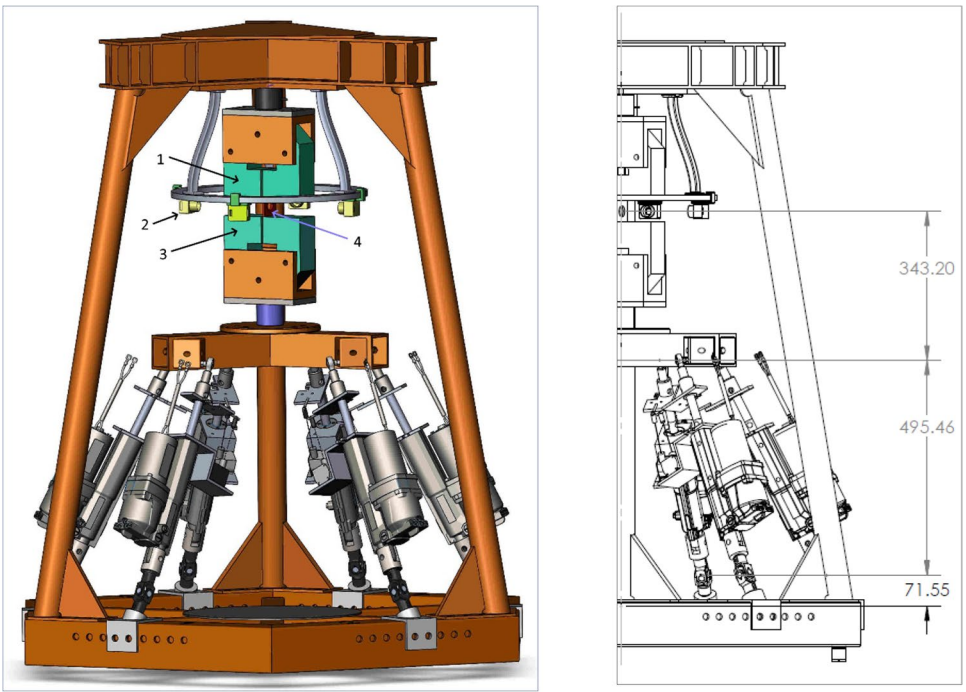


**Fig. 4** CAD model of hexapod test frame "Tiger 66.1" with partial sketch

**Table 2** Major dimensions of Tiger 66.1

| Sl no | Parameters | Dimensions |
|---|---|---|
| 1 | Base joint circle radius | 477.4 mm |
| 2 | Smaller sides of base | 377.9 mm |
| 3 | Larger sides of base | 570.4 mm |
| 4 | Platform joint circle radius | 225.1 mm |
| 5 | Smaller sides of platform | 178.8 mm |
| 6 | Larger sides of platform | 268.7 mm |
| 7 | Actuator stroke length | 203.0 mm |
| 8 | Base center to platform center height at test start | 495.46 mm |

**Table 3** Motion limits of Tiger 66.1

| Motion no | Parameters | Limits |
|---|---|---|
| 1 | Rotation about x-axis | ±30° |
| 2 | Rotation about y-axis | |
| 3 | Rotation about z-axis | ±55° |
| 4 | Translation along x-axis | ±158 mm |
| 5 | Translation along y-axis | |
| 6 | Translation along z-axis | 50 mm |

at the beginning of any test. All the calculations were done between the world frame center and this grip center pose. The major dimensions for Tiger 66.1 are shown in Table 2.

The workspace for Tiger 66.1 has been calculated based on the actuator stroke length limit, translation, and rotation limits of the platform along and about $x$, $y$, and $z$ axes respectively. The limits are given (Table 3) as per the feasible test conditions. All these motions limits are for pure translation and pure rotations.

The workspace for the test frame is plotted and shown in Fig. 5. Under given machine limits, the algorithm generated more than 180,000 valid workspace poses. This value may increase or decrease depending on the increment values used for platform's $dx$, $dy$, $dz$, $\alpha$, $\beta$, $\gamma$ parameters. The interval steps used in this case for translations along $x$ and $y$ axes = 15 mm, $z$-axis = 10 mm, and ± 10° for rotations about each axis.

In the next part of the calculation, a random 100 poses have been selected (Fig. 6) for finding the DH parameters for those poses through forward kinematics using DH transformation matrix.

For using the DH convention, coordinate frames have been assigned to each joint and a DH frame layout has been done (Fig. 7).

The layout shows the frames and variables assigned for one actuator path from world coordinate frame {0} to the moving grip center frame {8} on the platform. The coordinate frames were assigned as per the right-hand rule and modified DH convention. Frame {1} is the base center, frames {2} and {3} represent the universal joint at the bottom of each actuator, frame {4} denotes the prismatic joint on the actuator and frames {5}, {6}, and {7} represent spherical joint at the top end of each actuator. As per construction, variables $a$, $b$, $c$, and $d$ remain constant and represent height of the base center, base joint radius, platform joint radius, and grip center height from the platform center respectively.

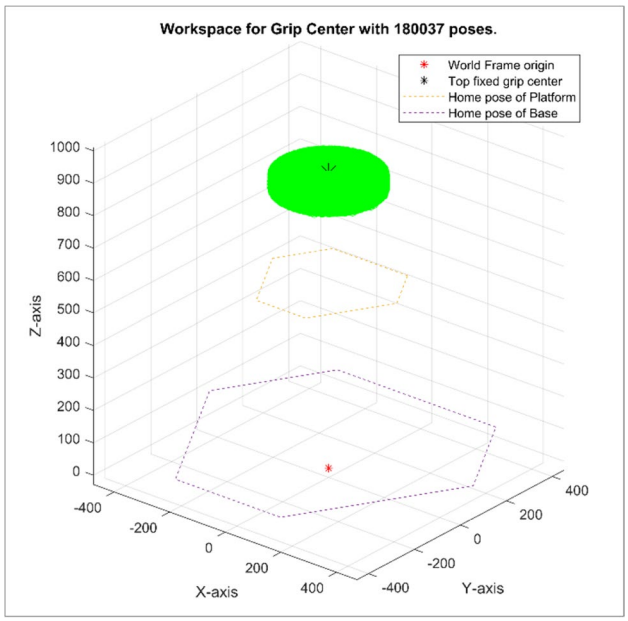


**Fig. 5** Graphical representation of workspace

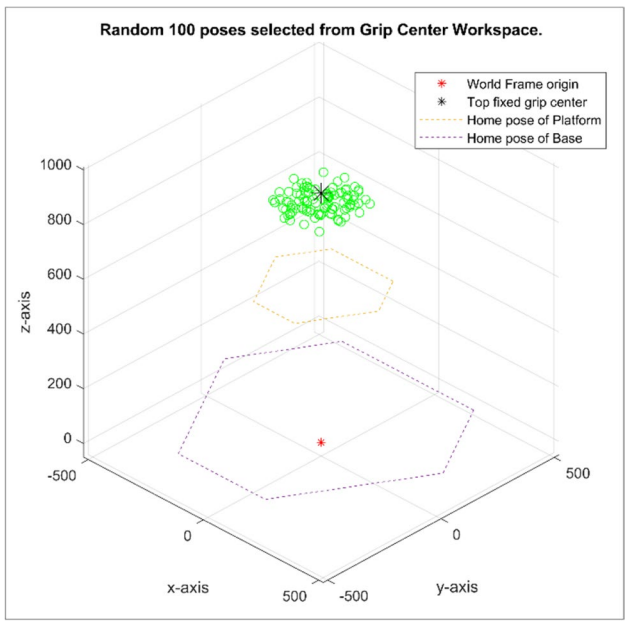


**Fig. 6** Selected 100 random poses from workspace

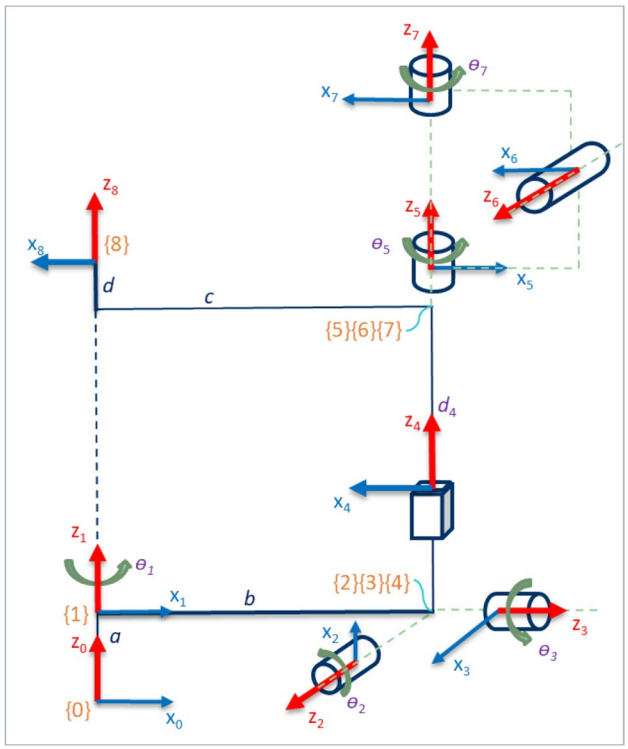


**Fig. 7** DH frame layout for each actuator path

**Table 4** Modified DH parameter table

| Link (i) | $a_{i-1}$ | $\alpha_{i-1}$ | $r_i$ | $\theta_i$ |
|---|---|---|---|---|
| 1 | 0 | 0 | a | $\theta_1$ * |
| 2 | b | π/2 | 0 | $\theta_2$ |
| 3 | 0 | π/2 | 0 | $\theta_3$ |
| 4 | 0 | π/2 | 0 | $-\pi/2$ |
| 5 | 0 | 0 | d4 | $\theta_5$ |
| 6 | 0 | π/2 | 0 | $\theta_6$ |
| 7 | 0 | π/2 | 0 | $\theta_7$ |
| 8 | c | 0 | d | 0 |

*$\theta_1$ for each actuator path is predefined by construction for Tiger 66.1 and the values are 53.4°, 126.6°, 173.4°, 246.6°, 293.4°, and 366.6°

Table 4 lists the DH parameters for each actuator path. All joint angles are varying except $\boldsymbol{\theta_1}$. The value of $\boldsymbol{\theta_1}$ for each actuator path remains fixed due to the construction of Tiger 66.1.

The iterations found valid DH parameter sets for 100 poses out of 100 valid poses. To check the success rate of the algorithm, a random 100 poses were selected for 10 calculation processes, and success rate of finding DH parameter sets is 100% for all processes. Table 5 shows the DH parameters value ranges as plotted in Fig. 8, and the success of finding DH parameters for the number of poses considered.

The DH parameter angles and actuator lengths variations for all 6 actuators for 100 poses are shown in Fig. 8.

The orientations of Tiger 66.1 have been drawn based on the DH parameters obtained through the simulation. Figure 9 shows random 6 numbers of such orientation layout, and it is observed that all the orientations are valid and feasible. It has been verified that all DH parameters sets generate

**Table 5** DH parameters value ranges and results obtained

| Variable | From | To |
|---|---|---|
| $\theta_2$ | −83.2° | −38.0° |
| $\theta_3$ | −118.8° | −63.4° |
| $\theta_5$ | 90.0° | 270.0° |
| $\theta_6$ | 92.0° | 231.0° |
| $\theta_7$ | −90.0° | 90.0° |
| $d_4$ | 465.68 mm | 664.68 mm |
| Number of poses evaluated | | 100 |
| Number of poses with valid DH parameters | | 100 |
| Number of poses with grip center deviation < 1 mm | | 100 |

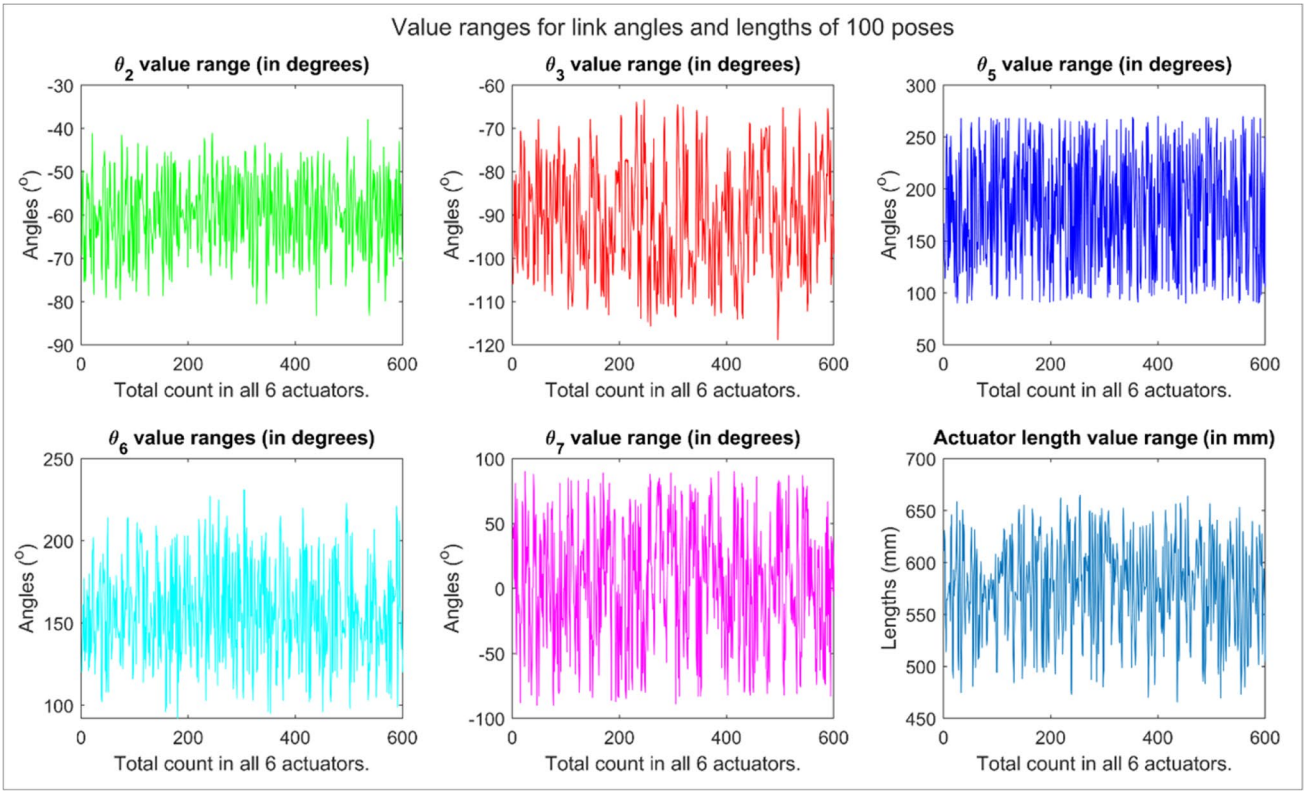


**Fig. 8** Angles and actuator length ranges

unique valid feasible poses every time, only some of them are shown here due to space limitations.

## 6 Results and discussion

The inverse kinematics and forward kinematics calculation run for Tiger 66.1 in this simulation shows that the unique solutions are achievable through the used algorithm. This algorithm does not use any complex calculations and found solutions through iterations. The pose data points were obtained for the workspace by changing the values of roll, pitch, yaw, and translations along $x$, $y$, and $z$ in each iteration. The number of data points is dependent on the increment step value of each variable. A smaller increment in value will generate more number of poses for the workspace. But the total time of calculation will increase because of an increased number of iterations.

The frame assignments from world frame to grip center frame at the joints through each actuator path can be done in different ways and the initial parameters values may be different. This depends on whether the modified or standard

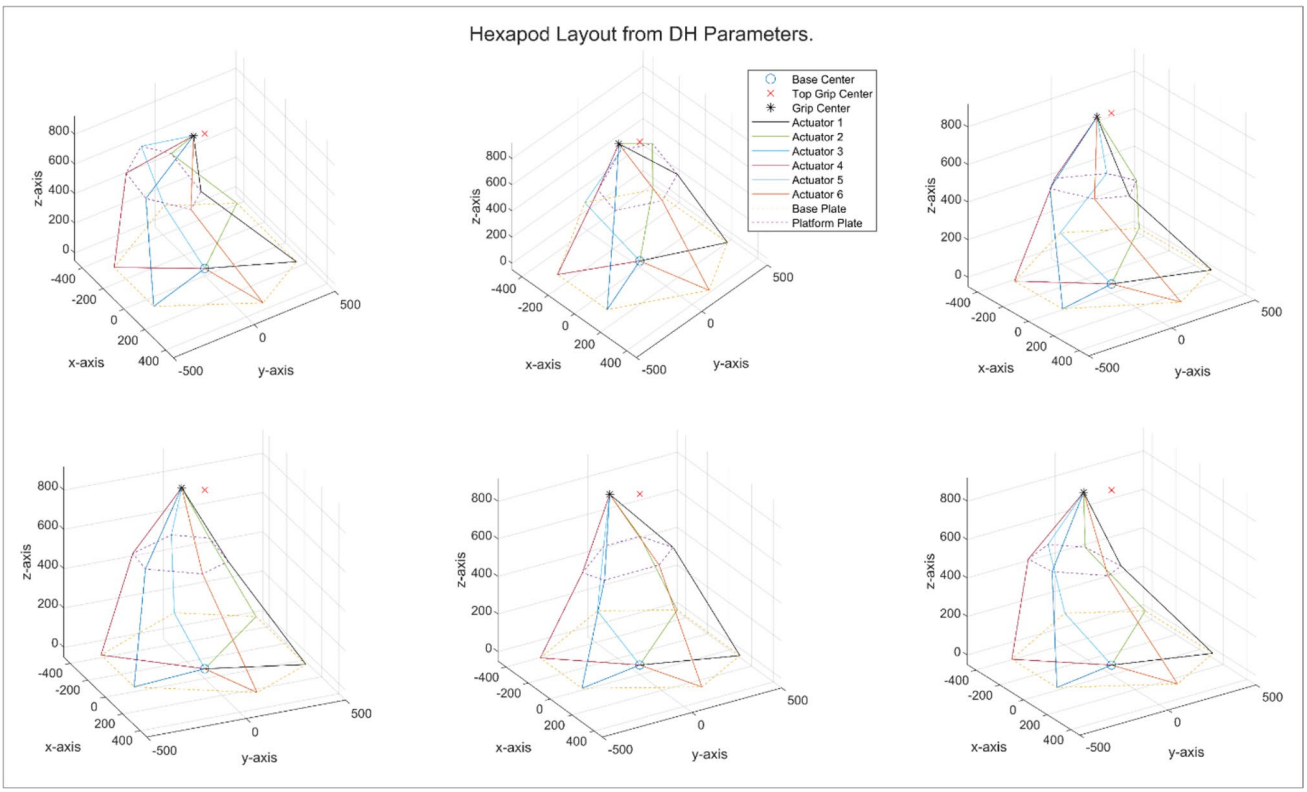


**Fig. 9** Tiger 66.1 orientation layouts through the calculated DH parameters

DH conventions are used, but the result will be the same because both the DH transformation matrices generate the same unique solutions. For finding the DH parameters for a pose, the variable values are changed by predefined steps in each iteration. The finer the increment steps are, the more accuracy level is achieved, but the calculation time will increase. In the current simulation, DH parameters for 100% valid poses have been found with pose accuracy level < 1 mm. For these iterations, the error limit for moving grip center pose was set as < 1 mm, and the angle finding steps used between 0.1 and 1°. In a standard standalone laptop with Intel(R) Core(TM) i7-8550U CPU @ 1.80 GHz and 16 GB physical memory, it took on an average 32.09 s to solve the DH parameters for one pose. In any multicore modern server, this solution time can be reduced significantly. Trial showed that in a 56-core server, the execution time comes down to 1.28 s to get the same solution.

The orientations sketched in Fig. 9 from the solutions found through forward kinematics indicate that they are feasible and unique. The forward kinematics solutions done earlier by the researchers yielded more than one solution for one pose. Those solutions need to be inspected one by one and validated for the acceptable one. The forward kinematic results from the method explained in this paper do not require any manual review to check their feasibility. This gives an option to use the DH parameters for any further calculation without any intermediate stop and manual intervention for the selection of correct results. In Sect. 6.1, we execute one such exercise which is important for the use of Tiger 66.1 in characterization of additively manufactured materials.

## 6.1 Pose deviation due to tolerances and sensitivity

A real-world Stewart platform is not free from manufacturing and assembly tolerances. These errors cause the actual platform-pose to deviate from the theoretical platform pose for a set of DH parameters. The actual measurement of those DH parameter deviations is not only difficult and time consuming, but also expensive due to proper instrumentation. With various combinations of DH parameter tolerances, a tentative pose deviation can be calculated with this algorithm.

Five tolerance values were considered for calculating the pose error. These values are $\pm 0.1, \pm 0.2, \pm 0.3, \pm 0.4$, and $\pm 0.5$ in degrees for angles and in mm for the actuator lengths. These tolerances were applied to the DH parameters for 100 poses found through the calculations as discussed in the previous section. For $\pm 0.5$ deg and

mm tolerances on the DH parameters, the platform-pose errors in terms of absolute $x$, $y$, and $z$ values and absolute distance values from the theoretical poses are shown in Fig. 10. The 0 marked horizontal lines in both the plots indicate the theoretical values.

Similar calculations were done with four other tolerance values. From calculated data, the maximum and minimum deviations were plotted to check the sensitivity of the platform poses due to the DH parameters tolerances (Fig. 11). The maximum and average deviations for each tolerance were shown here. In all these cases, the grip pose denotes the lower moving grip center mounted on the moving platform.

As seen from the plots, the maximum grip center or grip pose deviation occurs when tolerance value is $\pm 0.5$ for the DH parameters. The pose deviation distance may go more than 20 mm. This is of course a worst-case scenario where all the tolerance errors constructively combine to magnify the positional error in the end-effector position/orientation. It is far more likely that some tolerance increases the error value, while others decrease the error values. The deviations are appearing as linearly changing with the change of the tolerances, through that has not be verified in this work; but one more point has been found that the maximum and minimum values for the $x$, $y$, and $z$ coordinates as well as for the grip center deviations are not coincident. A summary of the observed data has been shown in Table 6.

## 7 Conclusion

The simulation has been successfully executed for the test-frame Tiger 66.1. Both inverse and forward kinematics are completed with the iteration-based algorithms discussed here. The results indicate that the implementation of the algorithm for real time calculations is sensible. Once the valid workspace data and corresponding DH parameters for the poses are calculated, they can be stored in a database, and those data can be used for real time applications. If a pose-data and related DH parameters are not available in the database, it can be calculated and added to the database to enrich it during the operation and can cover the whole workspace with more precise data. Also, the algorithm may be refined by reducing the iteration step values to generate valid DH parameter data sets for 100% valid poses with stricter error limits.

In this work, the authors ran the simulations with randomly selected 100 pose points. As the size of the database can grow bigger, finding the DH parameter data runtime

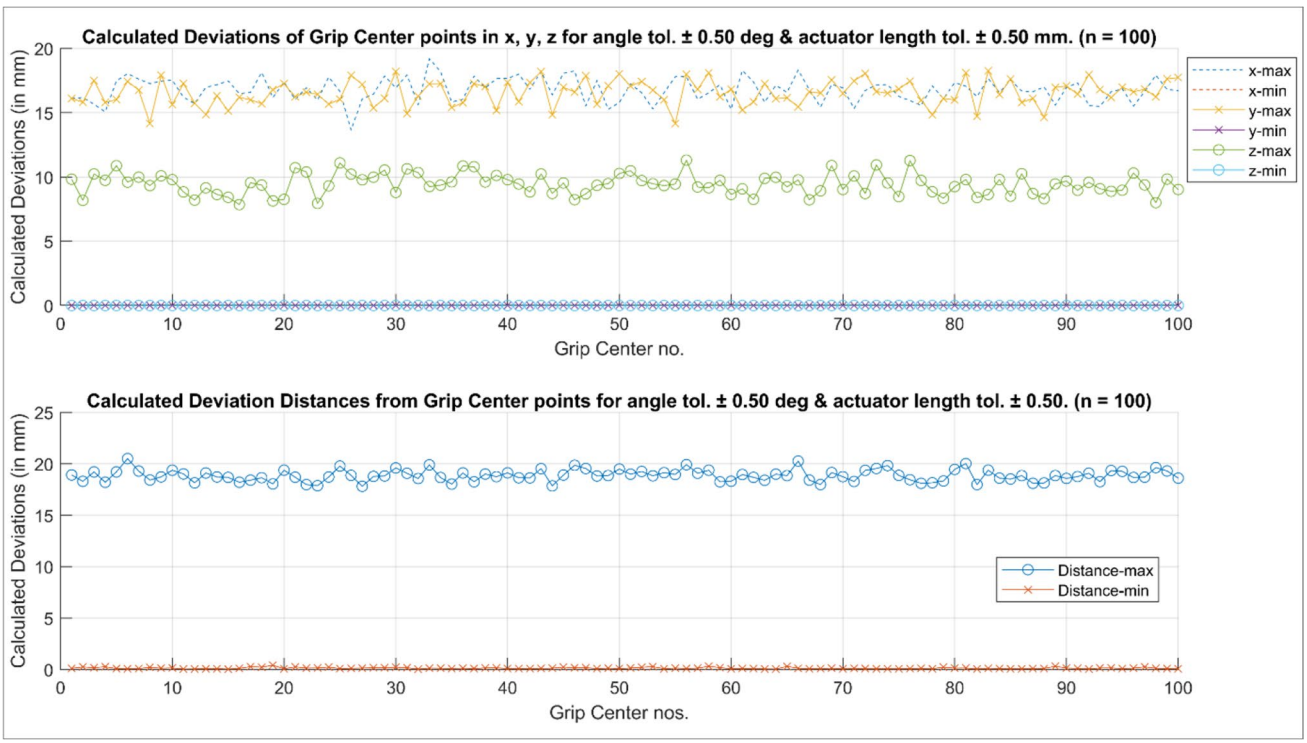


**Fig. 10** Platform pose deviations for DH parameters tolerances of $\pm 0.5$ deg and mm

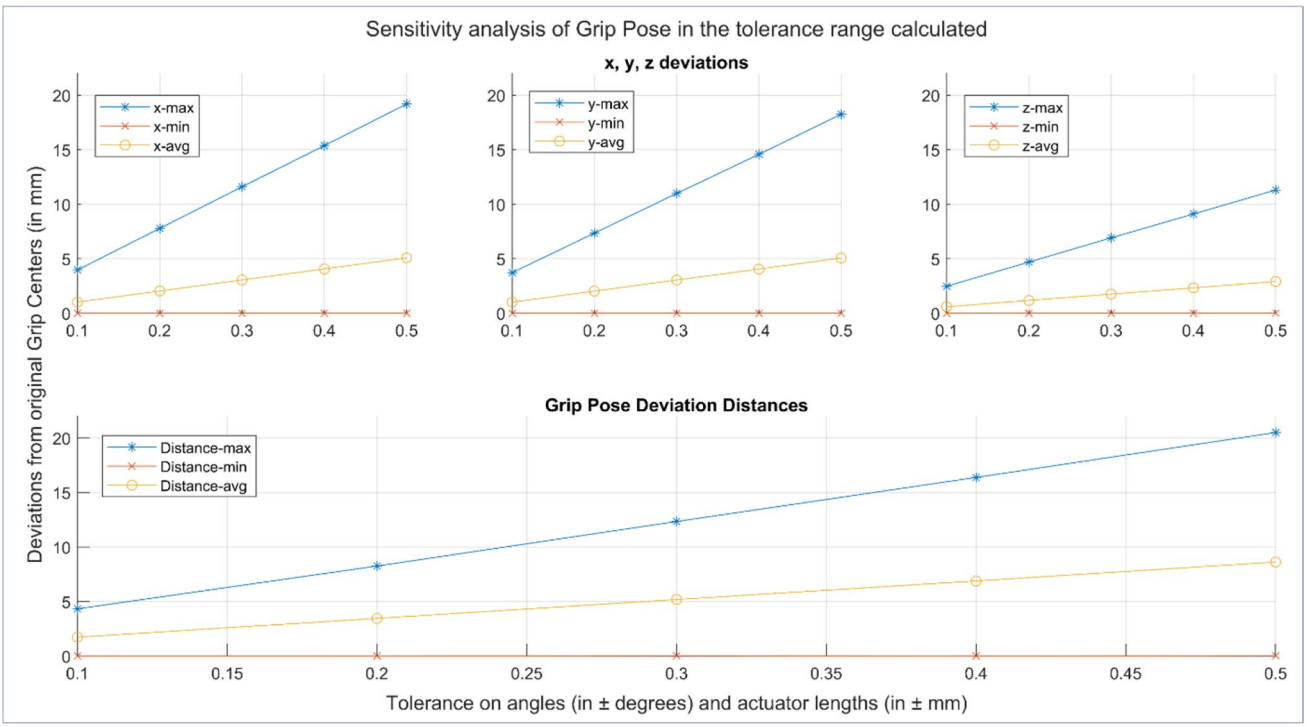


**Fig. 11** Sensitivity of platform poses due to DH parameters tolerances

**Table 6** Pose deviations corresponding to $x$, $y$, $z$ maximum and minimum absolute values

| Tol. value | At | x-deviation | y-deviation | z-deviation | Grip center deviation | At | x-deviation | y-deviation | z-deviation | Grip center deviation |
|---|---|---|---|---|---|---|---|---|---|---|
| ±0.1 | x-max | **3.96** | 0.32 | 1.01 | **4.10** | x-min | **0.00** | 2.23 | 1.10 | 2.49 |
| | y-max | 0.35 | **3.72** | 0.42 | 3.76 | y-min | 0.43 | **0.00** | 0.14 | **0.45** |
| | z-max | 1.75 | 1.99 | **2.47** | 3.62 | z-min | 0.63 | 1.96 | **0.00** | 2.06 |
| ±0.2 | x-max | **7.78** | 0.79 | 1.95 | **8.06** | x-min | **0.00** | 1.18 | 0.60 | 1.33 |
| | y-max | 1.59 | **7.35** | 1.40 | 7.65 | y-min | 1.82 | **0.00** | 0.52 | 1.89 |
| | z-max | 4.80 | 3.32 | **4.69** | 7.49 | z-min | 0.96 | 0.38 | **0.00** | **1.03** |
| ±0.3 | x-max | **11.60** | 1.28 | 2.88 | **12.02** | x-min | **0.00** | 6.59 | 1.68 | 6.81 |
| | y-max | 2.39 | **10.99** | 2.05 | 11.43 | y-min | 1.82 | **0.00** | 0.52 | 1.89 |
| | z-max | 4.80 | 3.32 | **4.69** | 7.49 | z-min | 0.96 | 0.38 | **0.00** | **1.03** |
| ±0.4 | x-max | **15.37** | 1.79 | 3.79 | **15.93** | x-min | **0.00** | 1.69 | 0.27 | **1.71** |
| | y-max | 2.55 | **14.57** | 4.60 | 15.49 | y-min | 4.18 | **0.00** | 1.79 | 4.54 |
| | z-max | 9.56 | 6.65 | **9.10** | 14.78 | z-min | 4.10 | 4.47 | **0.00** | 6.07 |
| ±0.5 | x-max | **19.19** | 2.33 | 4.69 | **19.89** | x-min | **0.00** | 7.78 | 4.03 | 8.76 |
| | y-max | 3.11 | **18.25** | 5.71 | 19.38 | y-min | 4.55 | **0.00** | 2.01 | 4.97 |
| | z-max | 11.95 | 8.34 | **11.30** | 18.44 | z-min | 0.60 | 0.15 | **0.00** | **0.62** |

The bold lettered values in the above table represent the maximum and minimum absolute values

for a required pose becomes trivial. The algorithm can be refined by adding efficient data searching methods.

There is further scope to validate these data with measurements done on the physical system for practical purposes. The authors intend to do this validation in their Tiger 66.1 hexapod platform in the next phase of this research through non-invasive methods like photogrammetry. In photogrammetry, the final platform pose can be measured directly without instrumenting and measuring each joint. By doing multiple actual measurements and comparing the

values with the calculated values, the in-built construction or fabrication errors of system can be evaluated and applying proper compensation factors for those in-built deviations, the hexapod platform can be guided to the desired pose more accurately. Such a calibration process of a Stewart platform would make practical application of the machine more useful.

**Acknowledgements** The authors would like to acknowledge the support of the Clemson University. All statements within are those of the authors and may or may not represent the views of these institutions.

**Author contribution** All authors contributed to the conception and design of the research. Software coding, data collection and analysis, and preparation of the first draft were performed by Sourabh Karmakar. All authors commented on and contributed to previous manuscript versions. All authors read and approved the final manuscript.

**Funding** The authors express their appreciation to the Clemson University who financially supported this work and to the United States Naval Research Laboratory in Washington, D.C. who provided technical insight and support for the design and operation of the TIGER 66.1 system through NCRADA-NRL-20–719.

## Declarations

**Competing interests** The authors declare no competing interests.

## References

1. Dasgupta B, Mruthyunjaya TS (1996) A constructive predictor-corrector algorithm for the direct position kinematics problem for a general 6–6 Stewart platform. Mech Mach Theory 31(6):799–811. https://doi.org/10.1016/0094-114X(95)00106-9
2. Tanaka W, Arai T, Inoue K, Mae Y, Park CS (2002) "Simplified kinematic calibration for a class of parallel mechanism. Proc IEEE Int Conf Robot Autom 1(1):483–488. https://doi.org/10.1109/ROBOT.2002.1013406
3. Daney D, Emiris IZ, Papegay Y, Tsigaridas E, Merlet JP (2006) Calibration of parallel robots : on the elimination of pose-dependent parameters. EuCoMeS 2006 - 1st Eur. Conf. Mech. Sci. Conf. Proc., pp 1–12
4. Majarena AC, Santolaria J, Samper D, Aguilar JJ (2010) An overview of kinematic and calibration models using internal/external sensors or constraints to improve the behavior of spatial parallel mechanisms. Sensors (Switzerland) 10(11):10256–10297. https://doi.org/10.3390/s101110256
5. Dasgupta B, Mruthyunjaya TS (1998) Closed-form dynamic equations of the general Stewart platform through the Newton-Euler approach. Mech Mach Theory 33(7):993–1012. https://doi.org/10.1016/S0094-114X(97)00087-6
6. Merlet J-P (2002) Still a long way to go on the road for parallel mechanisms. Asme 64(May):1–19
7. Jakobović D, Jelenković L (2002) The forward and inverse kinematics problems for stewart parallel mechanisms. CIM 2002 Comput. Integr. Manuf. High Speed Mach. - 8th Int. Sci. Conf. Prod. Eng
8. Innocenti C, Parenti-Castelli V (1990) Direct position analysis of the Stewart platform mechanism. Mech Mach Theory 25(6):611–621. https://doi.org/10.1016/0094-114X(90)90004-4
9. Huang X, Liao Q, Wei S, Qiang X, Huang S (2007) Forward kinematics of the 6–6 Stewart platform with planar base and platform using algebraic elimination. Proc IEEE Int Conf Autom Logist ICAL 2007(104043):2655–2659. https://doi.org/10.1109/ICAL.2007.4339029
10. Husty ML (1996) An algorithm for solving the direct kinematics of general Stewart-Gough platforms. Mech Mach Theory 31(4):365–379. https://doi.org/10.1016/0094-114X(95)00091-C
11. Wang Y (2007) A direct numerical solution to forward kinematics of general Stewart-Gough platforms. Robotica 25(1):121–128. https://doi.org/10.1017/S0263574706003080
12. Cardona M (2016) Kinematics and Jacobian analysis of a 6UPS Stewart-Gough platform. 2016 IEEE 36th Cent. Am. Panama Conv. CONCAPAN 2016. https://doi.org/10.1109/CONCAPAN.2016.7942377
13. Zhang H, Gao Q, Zhang M, Yao Y (2021) Forward kinematics of parallel robot based on neural network Newton-Raphson iterative algorithm. (December 2021):36. https://doi.org/10.1117/12.2625254
14. Yang C, Huang Q, Ogbobe PO, Han J (2009) Forward kinematics analysis of parallel robots using global Newton-Raphson method. 2009 2nd Int. Conf. Intell. Comput. Technol. Autom. ICICTA 2009. 3:407–410. https://doi.org/10.1109/ICICTA.2009.564
15. Zhou X, Zhou F, Wang Y (2017) Pose error modeling and analysis for 6-DOF stewart platform. 2017 Chinese Automation Congress (CAC), Jinan, China, pp. 6470–6475. https://doi.org/10.1109/CAC.2017.8243943
16. Tarokh M (2007) Real time forward kinematics solutions for general Stewart platforms. Proc. - IEEE Int. Conf. Robot. Autom. (April):901–906. https://doi.org/10.1109/ROBOT.2007.363100
17. Porta JM, Thomas F (2018) Yet another approach to the Gough-Stewart platform forward kinematics. Proc. - IEEE Int. Conf. Robot. Autom, pp 974–980. https://doi.org/10.1109/ICRA.2018.8460900
18. KumarP R, Bandyopadhyay B (2013) The forward kinematic modeling of a Stewart platform using NLARX model with wavelet network. https://doi.org/10.1109/INDIN.2013.6622907
19. Fazenda N, Lubrano E, Rossopoulos S, Clavel R (2006) Calibration of the 6 DOF high-precision flexure parallel robot 'Sigma 6. Proc. 5th Parallel Kinemat. Semin. Chemnitz, Ger., pp 379–398. https://infoscience.epfl.ch/record/117939. Accessed 27 Sept 2021
20. Ziegert JC, Kalle P (1994) Error compensation in machine tools: a neural network approach. J Intell Manuf 5(3):143–151. https://doi.org/10.1007/BF00123919
21. Schmidt V, Müller B, Pott A (2014) Solving the forward kinematics of cable-driven parallel robots with neural networks and interval arithmetic. Mech Mach Sci 15:103–110. https://doi.org/10.1007/978-94-007-7214-4_12
22. Morell A, Acosta L, Toledo J (2012) An artificial intelligence approach to forward kinematics of Stewart platforms. 2012 20th Mediterr. Conf. Control Autom. MED 2012 - Conf. Proc, pp 433–438. https://doi.org/10.1109/MED.2012.6265676
23. Yiu YK, Meng J, Li ZX (2003) Auto-calibration for a parallel manipulator with sensor redundancy. Proc. - IEEE Int. Conf. Robot. Autom., vol. 3, pp. 3660–3665. https://doi.org/10.1109/robot.2003.1242158
24. Michaloski J (1988) Coordinated joint motion for an industrial robot. NIST Interagency/Internal Report (NISTIR), National Institute of Standards and Technology, Gaithersburg, MD, [online].

https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=820236. Accessed 25 Feb 2021
25. Zhuang H, Liu L, Masory O (2000) Autonomous calibration of hexapod machine tools. J Manuf Sci Eng Trans ASME 122(1):140–148. https://doi.org/10.1115/1.538893
26. Tsai L-W (1999) Robot Analysis: The Mechanics of Serial and Parallel Manipulators. Wiley, New York
27. Charters T, Enguiça R, Freitas P (2009) Detecting singularities of stewart platforms. Ind Case Stud J 1:66–80. https://cdn.instructables.com/ORIG/FFK/LAIV/I55MRG6M/FFKLAIVI55MRG6M.pdf; accessed 21 Sept 2021
28. Li B, Cao Y, Zhang Q, Huang Z (2013) Position-singularity analysis of a special class of the Stewart parallel mechanisms with two dissimilar semi-symmetrical hexagons. Robotica 31(1):123–136. https://doi.org/10.1017/S0263574712000148
29. Denavit J, Hartenberg RS (1955) A kinematic notation for lower-pair mechanisms based on matrices. J Appl Mech 22(2):215–221. https://doi.org/10.1115/1.4011045
30. Granja M, Chang N, Granja V, Duque M, Llulluna F (2016) Comparison between standard and modified Denavit-Hartenberg methods in robotics modelling. Proc World Congr Mech Chem Mater Eng 1(1):1–10. https://doi.org/10.11159/icmie16.118
31. Craig JJ (2004) Introduction to robotics: mechanics and control. Pearson Prentice Hall(3e):408

# Terms and Conditions

Springer Nature journal content, brought to you courtesy of Springer Nature Customer Service Center GmbH (“Springer Nature”).
Springer Nature supports a reasonable amount of sharing of research papers by authors, subscribers and authorised users (“Users”), for small-scale personal, non-commercial use provided that all copyright, trade and service marks and other proprietary notices are maintained. By accessing, sharing, receiving or otherwise using the Springer Nature journal content you agree to these terms of use (“Terms”). For these purposes, Springer Nature considers academic use (by researchers and students) to be non-commercial.
These Terms are supplementary and will apply in addition to any applicable website terms and conditions, a relevant site licence or a personal subscription. These Terms will prevail over any conflict or ambiguity with regards to the relevant terms, a site licence or a personal subscription (to the extent of the conflict or ambiguity only). For Creative Commons-licensed articles, the terms of the Creative Commons license used will apply.
We collect and use personal data to provide access to the Springer Nature journal content. We may also use these personal data internally within ResearchGate and Springer Nature and as agreed share it, in an anonymised way, for purposes of tracking, analysis and reporting. We will not otherwise disclose your personal data outside the ResearchGate or the Springer Nature group of companies unless we have your permission as detailed in the Privacy Policy.
While Users may use the Springer Nature journal content for small scale, personal non-commercial use, it is important to note that Users may not:

1. use such content for the purpose of providing other users with access on a regular or large scale basis or as a means to circumvent access control;
2. use such content where to do so would be considered a criminal or statutory offence in any jurisdiction, or gives rise to civil liability, or is otherwise unlawful;
3. falsely or misleadingly imply or suggest endorsement, approval , sponsorship, or association unless explicitly agreed to by Springer Nature in writing;
4. use bots or other automated methods to access the content or redirect messages
5. override any security feature or exclusionary protocol; or
6. share the content in order to create substitute for Springer Nature products or services or a systematic database of Springer Nature journal content.

In line with the restriction against commercial use, Springer Nature does not permit the creation of a product or service that creates revenue, royalties, rent or income from our content or its inclusion as part of a paid for service or for other commercial gain. Springer Nature journal content cannot be used for inter-library loans and librarians may not upload Springer Nature journal content on a large scale into their, or any other, institutional repository.
These terms of use are reviewed regularly and may be amended at any time. Springer Nature is not obligated to publish any information or content on this website and may remove it or features or functionality at our sole discretion, at any time with or without notice. Springer Nature may revoke this licence to you at any time and remove access to any copies of the Springer Nature journal content which have been saved.
To the fullest extent permitted by law, Springer Nature makes no warranties, representations or guarantees to Users, either express or implied with respect to the Springer nature journal content and all parties disclaim and waive any implied warranties or warranties imposed by law, including merchantability or fitness for any particular purpose.
Please note that these rights do not automatically extend to content, data or other material published by Springer Nature that may be licensed from third parties.
If you would like to use or distribute our Springer Nature journal content to a wider audience or on a regular basis or in any other manner not expressly permitted by these Terms, please contact Springer Nature at

onlineservice@springernature.com