IMA Conference on Mathematics of Robotics
9 – 11 September 2015, St Anne's College, University of Oxford

Kinematics of an X-Ray Mirror Tripod Isomorphic to a 6-3 Stewart Platform

By R. C. Harwin, P.M. Sharkey, W. S. Harwin, S. M. Scott, D. M. Hawkins and S. P. Collins

Abstract

This paper examines the kinematics of parallel linkage arrangement referred to as a tripod. The mechanism has 6 degrees of freedom and is isomorphic to a 6-3 Stewart platform. A practical and fast forward kinematics solution is given, based on mapping to the 6-3 Stewart platform, and has been implemented on a prototype platform in both Python and Matlab. The platform is designed for a set of Kirkpatrick-Baez X-ray mirrors and the results show that an accuracy of the order of tens of nanometres is possible.

1. Introduction

A parallel linkage structure - here referred to as a tripod - was chosen to move a system of X-ray mirrors on beamline I16 [1] at Diamond Light Source, a synchrotron research facility in Oxfordshire, England. The X-ray mirror system is used to reduce the focal spot size of the X-rays to reduce blurring in the images obtained. Since experiments typically run for 6 days the tripod is required to be placed in a particular alignment and hold its position with very little slippage for the duration of an experiment. In contrast to common Stewart Platform mechanisms, the tripod design features legs of fixed length. The motion of the platform is achieved through the translation motion of the X-Y plane of each leg base.

Because the X-ray beam used in the experiment is effectively cylindrically symmetric, the tripod only needs high accuracy for translation in two directions and rotation about the corresponding two axes. It was therefore designed for maximum accuracy of these movements at the expense of the accuracy of the remaining two possible movements. The platform is longer in the direction parallel to the beam, so although all the tripod's legs are the same length, it is not rotationally symmetric. The tripod investigated in this paper was constructed as a prototype as shown in Figure 1, with the legs hinged at the bases and mounted on individual motorised stages. These stages were attached to a base plate, defined as the $x_B - y_B$ plane in the base (world) coordinate frame. The legs attach to the platform via ball joints. The tripod can be moved by changing the positions of the three leg bases, each of which can move in two orthogonal directions referred to as x_B and y_B. The motorised stages used to move the prototype tripod are capable of being moved distances of the order of 10 nm.

In order to be suitable for purpose, the tripod needs to be operated remotely, via Python code that runs on the control system for the experimental station. This control system runs a version of Jython with basic Python capabilities, but cannot support numerical solvers such as those found in the SciPy module.

This paper describes a method of solving the mathematics of the tripod system ana-

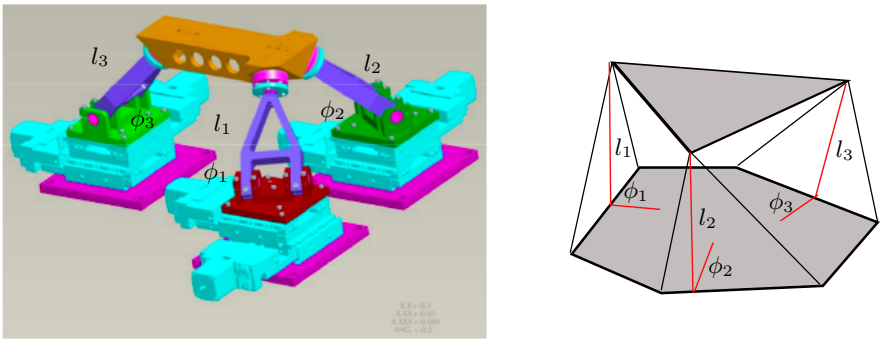

FIGURE 1. Left: A 3D model of the tripod structure. The x-y stages are shown in blue, with the hinged bases of the legs in green and red. The legs are coloured purple, with the ball joints in pink and blue, and the platform is shown in orange. The pink sections at the bottom represent sections of the plate that the stages are attached to. The leg lengths and angles shown correspond to the leg lengths and angles marked in the diagram to the right. Right: The model applied to the 6-3 Stewart platform to convert it into a tripod-like structure. These modifications enable its kinematics to be solved in both directions.

lytically. A program was created and tested to control the tripod using only the basic Python module NumPy. Matlab was used to confirm the validity of checking the results.

2. Kinematics of the tripod

The tripod is isomorphic to a 6-3 Stewart platform since the axes of the lower three hinge joints on the three individual x-y stages define a triangle, the nodes of which define the intersections of the pairs of parallel actuators needed for a Stewart platform. (In fact the kinematics are isomorphic to a 3-3 Stewart platform). A typical solution for the forward kinematic of a Stewart platform usually then proceeds by defining circles perpendicular to the hinge axes, and in this case these can be identified with the three tripod links as shown in Figure 1. Throughout this solution, positions on the base coordinate system are denoted by the subscript B and positions on the platform coordinate system are denoted by the subscript P. Both the Python and Matlab scripts are available in a GitHub repository.

In any given position the 3-RPS parallel robot is isomorphically identical to both the x-ray mirror and the 6-3 Stewart Platform [2, 3, 4]. The only distinction is that the solution can be simplified in any application by fixing the locations of the three rotational joints prior to the control loop. Thus the position of the three rotational joints in the 3RP configuration can be matched to those of the three base rotations in the x-ray mirror kinematics, and the lengths of the parallel joints equated to the lengths $l_1 - l_3$ in Figure 1.

Table 1 gives the definitions of the key parameters used in the solution described in this section, and validated numerically in Section 3.

Forward Kinematics: Top plate position and orientation

First, the top plate vectors $\mathbf{t}_i$ are calculated from known bottom base vectors $\mathbf{b}_i$ by applying the constraint that the distances h_i are fixed. It is convenient to write each of the top plate vectors as the sum of the corresponding bottom vector and a vector linking the two, which in turn depends on the rotation angles ψ_i (known and fixed) and ϕ_i (unknown):

$$\mathbf{t}_i = \mathbf{b}_i + \hat{\mathbf{v}}_i l_i \tag{2.1}$$

Parameter	Definition
(x_B, y_B, z_B)	The coordinate system referenced to the base plate, as seen in the right hand diagram on Figure 2, where $z_B = 0$ is defined to be the base plate plane and contains the three base points of the tripod legs $\mathbf{b}_i$.
(x_P, y_P, z_P)	The coordinate system attached to the top plane. The $z_P = 0$ plane contains $\mathbf{t}_i$ and y_P is parallel to $\mathbf{t}_1 - \mathbf{t}_2$.
$\mathbf{b}_i$	The base points of the tripod legs, in the $z_B = 0$ plane, as seen on the right hand side of Figure 2, written as sums of their centred values $\mathbf{b}_i^C$ and translations along x_B and y_B provided by two slides on each leg *i.e.* $\mathbf{b}_i = \mathbf{b}_i^C + b_i^x \hat{\mathbf{x}}_\mathbf{B} + b_i^y \hat{\mathbf{y}}_\mathbf{B}$.
$\mathbf{t}_i$	The tops of the tripod legs, in the $z_P = 0$ plane, as marked on the right hand side of Figure 2.
$\mathbf{c_P}$	A tooling point, placed at a fixed point $(c_{x_P}, c_{y_P}, c_{z_P})$ in the platform coordinate frame.
ψ_i	The angle defining the orientation of the vector about which leg i is free to pivot, the angle through which the y_B axis is rotated about the z_B axis to give the pivot vector, shown on the left hand side of Figure 2. This is fixed - the tripod cannot pivot about this angle.
ϕ_i	The angle between leg i and the $x_B - y_B$ plane, giving the orientation of leg i, as seen in the left hand diagram in Figure 2.
l_i	The fixed length of leg i.
h_i	The fixed distance between points $\mathbf{t}_j$ and $\mathbf{t}_k$, where $i \neq j \neq k$, as indicated on the right hand diagram in Figure 2.

TABLE 1. A table to show the definitions of all the parameters used in this paper.

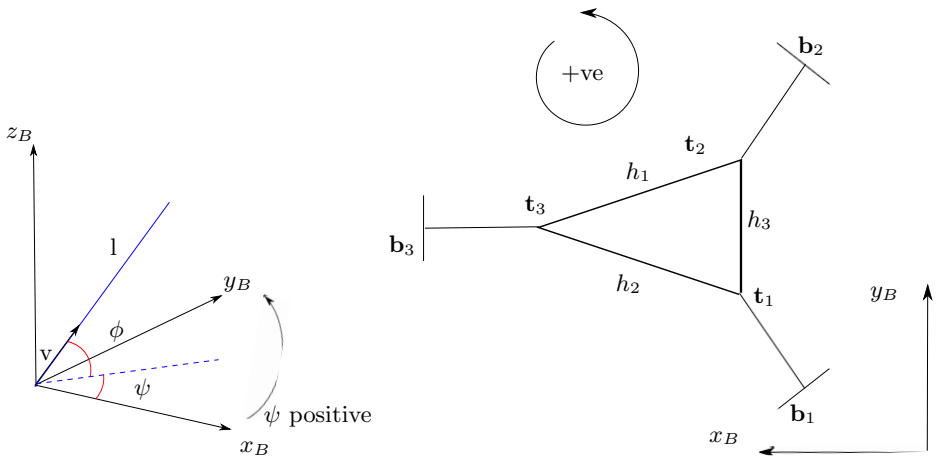

FIGURE 2. The diagram to the left shows the position of one of the tripod legs (in blue) to show the relationship between the angles ϕ and ψ, and their definitions relative to the base coordinate system. The diagram to the right shows the tripod assembly from the top, with the dimensions of the platform marked, as well as the vectors defining the positions of the bases of the legs and the positions of the tops of the legs.

where

$$\hat{\mathbf{v}}_i = \begin{pmatrix} \cos\psi_i \cos\phi_i \\ \sin\psi_i \cos\phi_i \\ \sin\phi_i \end{pmatrix} \tag{2.2}$$

Applying the constraint that h_i are fixed gives a set of three quadratic equations,

$$\begin{aligned} |\mathbf{t}_2 - \mathbf{t}_1|^2 - h_3^2 &= 0 \\ |\mathbf{t}_3 - \mathbf{t}_2|^2 - h_1^2 &= 0 \\ |\mathbf{t}_1 - \mathbf{t}_3|^2 - h_2^2 &= 0 \end{aligned} \tag{2.3}$$

the solutions of which give ϕ_i, which in turn can be used to calculate $\hat{\mathbf{v}}_i$ and $\mathbf{t}_i$. It is evident that without mechanical constraints there would be two solutions for each leg, corresponding to $\pm(\phi_1, \phi_2, \phi_3)$, suggesting 2^3 configurations, so it is essential to ensure that the correct solution is selected. The operational Jython code includes additional range checks to ensure that the only computed solution has all tripod link angles $0 < \phi_i < \pi$ to ensure there is no accidental damage to the mechanism.

As the tripod is isomorphic to the 6-3 Stewart platform, the equations can be solved using existing methods for solving the 6-3 Stewart platform [4, 5].

Expanding and simplifying (using the identity $\cos^2\phi_i + \sin^2\phi_i = 1$) Eqns.(2.3) gives the following equations:

$$\begin{aligned} D_1\cos\phi_1 + D_2\cos\phi_2 + D_3\cos\phi_1\cos\phi_2 + D_4\sin\phi_1\sin\phi_2 + D_5 &= 0 \\ E_1\cos\phi_2 + E_2\cos\phi_3 + E_3\cos\phi_2\cos\phi_3 + E_4\sin\phi_2\sin\phi_3 + E_5 &= 0 \\ F_1\cos\phi_3 + F_2\cos\phi_1 + F_3\cos\phi_3\cos\phi_1 + F_4\sin\phi_3\sin\phi_1 + F_5 &= 0 \end{aligned} \tag{2.4}$$

with coefficients:

$$\begin{aligned} D_1 &= -2l_1((b_{2x} - b_{1x})\cos\psi_1 + (b_{2y} - b_{1y})\sin\psi_1) \\ D_2 &= 2l_2((b_{2x} - b_{1x})\cos\psi_2 + (b_{2y} - b_{1y})\sin\psi_2) \\ D_3 &= -2l_1l_2(\cos\psi_1\cos\psi_2 + \sin\psi_1\sin\psi_2) \\ D_4 &= -2l_1l_2 \\ D_5 &= (b_{2x} - b_{1x})^2 + (b_{2y} - b_{1y})^2 + l_1^2 + l_2^2 - h_3^2 \\ E_1 &= -2l_2((b_{3x} - b_{2x})\cos\psi_2 + (b_{3y} - b_{2y})\sin\psi_2) \\ E_2 &= 2l_3((b_{3x} - b_{2x})\cos\psi_3 + (b_{3y} - b_{2y})\sin\psi_3) \\ E_3 &= -2l_2l_3(\cos\psi_2\cos\psi_3 + \sin\psi_2\sin\psi_3) \\ E_4 &= -2l_2l_3 \\ E_5 &= (b_{3x} - b_{2x})^2 + (b_{3y} - b_{2y})^2 + l_2^2 + l_3^2 - h_1^2 \\ F_1 &= -2l_3((b_{1x} - b_{3x})\cos\psi_3 + (b_{1y} - b_{3y})\sin\psi_3) \\ F_2 &= 2l_1((b_{1x} - b_{3x})\cos\psi_1 + (b_{1y} - b_{3y})\sin\psi_1) \\ F_3 &= -2l_1l_3(\cos\psi_1\cos\psi_3 + \sin\psi_1\sin\psi_3) \\ F_4 &= -2l_1l_3 \\ F_5 &= (b_{1x} - b_{3x})^2 + (b_{1y} - b_{3y})^2 + l_1^2 + l_3^2 - h_2^2 \end{aligned} \tag{2.5}$$

These equations can be solved by making the tan substitutions,$\cos\phi_i = \frac{1-x_i^2}{1+x_i^2}$ and $\sin\phi_i = \frac{2x_i}{1+x_i^2}$, with $x_i = \tan(\frac{\phi_i}{2})$. leaving only three variables to solve for (x_1, x_2 and x_3).
Collecting terms of the same order in x_1, x_2 or x_3 gives the following set of equations:

$$(G_1x_1^2 + G_2)x_2^2 + G_3x_1x_2 + (G_4x_1^2 + G_5) = 0 \tag{2.6a}$$

$$(H_1x_3^2 + H_4)x_2^2 + H_3x_3x_2 + (H_2x_3^2 + H_5) = 0 \tag{2.6b}$$

$$(I_1x_1^2 + I_4)x_3^2 + I_3x_1x_3 + (I_2x_1^2 + I_5) = 0 \tag{2.6c}$$

where the G_i, H_i and I_i are linear functions of the D_i, E_i and F_i respectively.

We compute the resultant of the equations (2.6a) and (2.6b) with respect to the unknown x_2 [6] to give an equation that only contains x_1 and x_3 terms. After expansion and simplification of the determinant, the solution of this system is a 4th order polynomial

$$J_1 x_3^4 + J_2 x_3^3 + J_3 x_3^2 + J_4 x_3 + J_5 = 0 \tag{2.7}$$

with

$$\begin{aligned} J_1 &= K_1 x_1^4 + K_2 x_1^2 + K_3 \\ J_2 &= K_4 x_1^3 + K_5 x_1 \\ J_3 &= K_6 x_1^4 + K_7 x_1^2 + K_8 \\ J_4 &= K_9 x_1^3 + K_{10} x_1 \\ J_5 &= K_{11} x_1^4 + K_{12} x_1^2 + K_{13} \end{aligned} \tag{2.8}$$

where the K_i are functions of the G_i and H_i:

$$\begin{aligned} K_1 &= G_1^2 H_2^2 - 2G_1 G_4 H_1 H_2 + G_4^2 H_1^2 \\ K_2 &= 2G_1 G_2 H_2^2 - 2G_1 G_5 H_1 H_2 - 2G_2 G_4 H_1 H_2 + 2G_4 G_5 H_1^2 + G_3^2 H_1 H_2 \\ K_3 &= G_2^2 H_2^2 - 2G_2 G_5 H_1 H_2 + G_5^2 H_1^2 \\ K_4 &= -G_1 G_3 H_2 H_3 - G_3 G_4 H_1 H_3 \\ K_5 &= -G_2 G_3 H_2 H_3 - G_3 G_5 H_1 H_3 \\ K_6 &= 2G_1^2 H_2 H_5 - 2G_1 G_4 H_2 H_4 - 2G_1 G_4 H_1 H_5 + 2G_4^2 H_1 H_4 + G_1 G_4 H_3^2 \\ K_7 &= 4G_1 G_2 H_2 H_5 - 2G_1 G_5 H_2 H_4 - 2G_1 G_5 H_1 H_5 - 2G_2 G_4 H_2 H_4 - 2G_2 G_4 H_1 H_5 \\ &\quad + 4G_4 G_5 H_1 H_4 + G_3^2 H_2 H_4 + G_3^2 H_1 H_5 + G_2 G_4 H_3^2 + G_1 G_5 H_3^2 \\ K_8 &= 2G_2^2 H_2 H_5 - 2G_2 G_5 H_2 H_4 - 2G_2 G_5 H_1 H_5 + 2G_5^2 H_1 H_4 + G_2 G_5 H_3^2 \\ K_9 &= -G_1 G_3 H_3 H_5 - G_3 G_4 H_3 H_4 \\ K_{10} &= -G_2 G_3 H_3 H_5 - G_3 G_5 H_3 H_4 \\ K_{11} &= G_1^2 H_5^2 - 2G_1 G_4 H_4 H_5 + G_4^2 H_4^2 \\ K_{12} &= 2G_1 G_2 H_5^2 - 2G_1 G_5 H_4 H_5 - 2G_2 G_4 H_4 H_5 + 2G_4 G_5 H_4^2 + G_3^2 H_4 H_5 \\ K_{13} &= G_2^2 H_5^2 - 2G_2 G_5 H_4 H_5 + G_5^2 H_4^2 \end{aligned} \tag{2.9}$$

The equation (2.6c) can be rewritten as $M_1 x_3^2 + M_2 x_3 + M_3 = 0$, where $M_1 = I_1 x_1^2 + I_4$, $M_2 = I_3 x_1$ and $M_3 = I_2 x_1^2 + I_5$. Again by means of the resultant, x_3 can be eliminated. On simplification a 16th order polynomial biquadratic equation (which reduces to an 8th order polynomial via the substitution $y = x_1^2$) is obtained:

$$P_{16} x_1^{16} + P_{14} x_1^{14} + P_{12} x_1^{12} + P_{10} x_1^{10} + P_8 x_1^8 + P_6 x_1^6 + P_4 x_1^4 + P_2 x_1^2 + P_0 = 0 \tag{2.10}$$

The coefficients P_i are functions of the K_i and the I_i.

This equation can then be solved in principle to give possible values of x_1, although this is not a straightforward process. The Abel-Ruffini theorem states that there is no general, finite, algebraic solution for a polynomial of degree 5 or higher [7] so the equation must be solved by using a polynomial solver. Most computer programming languages have basic polynomial solving capabilities, using various different algorithms. In this case, the polynomial solver used was NumPy's roots solver, which uses an algorithm that relies on computing the eigenvalues of the companion matrix [8].

On using the numerical solver, with the parameter values listed in Section 3, 16 possible x-values are obtained. From mechanical constraints of the tripod, it is known that $\phi_1 \in [\frac{\pi}{4} - \epsilon, \frac{\pi}{4} + \epsilon]$ so the desired value of x_1 can be determined, as $x_1 \in [\tan\frac{\pi}{8} - \frac{\epsilon}{2}, \tan\frac{\pi}{8} + \epsilon]$.

From this, the value of ϕ_1 can be determined, as $\phi_1 = 2\tan^{-1} x_1$, and x_1 can then be substituted back into equations (2.6a) and (2.6c) to determine the values of ϕ_2 and ϕ_3, using the knowledge that $\phi_2 \in [\frac{\pi}{4} - \epsilon, \frac{\pi}{4} + \epsilon]$ and $\phi_3 \in [-\frac{\pi}{4} - \epsilon, -\frac{\pi}{4} + \epsilon]$. The values of ϕ_2 and ϕ_3 obtained can be checked by substitution back into Equation (2.6b).

The values of ϕ_i can be substituted into Equation (2.2) to find $\mathbf{v}_i$ and the $\mathbf{v}_i$ can then be substituted back into Equation (2.1) to find the vectors $\mathbf{t}_i$. Using $\mathbf{t}_i$, it is now possible to calculate $(c_{x_B}, c_{y_B}, c_{z_B})$. As $\mathbf{c}$ is already known in the platform frame, finding $\mathbf{c}$ in the base frame simply requires a knowledge of $\hat{\mathbf{x}}_\mathbf{P}$, $\hat{\mathbf{y}}_\mathbf{P}$ and $\hat{\mathbf{z}}_\mathbf{P}$ since

$$\mathbf{c}_{x_B,y_B,z_B} = \mathbf{t}_2 + c_{x_P}\hat{\mathbf{x}}_\mathbf{P} + c_{y_P}\hat{\mathbf{y}}_\mathbf{P} + c_{z_P}\hat{\mathbf{z}}_\mathbf{P} \tag{2.11}$$

where $\hat{\mathbf{y}}_\mathbf{P} = \widehat{(\mathbf{t}_2 - \mathbf{t}_1)}$, $\hat{\mathbf{z}}_\mathbf{P} = \{\widehat{(\mathbf{t}_3 - \mathbf{t}_1) \times (\mathbf{t}_2 - \mathbf{t}_1)}\}$ and $\hat{\mathbf{x}}_\mathbf{P} = \hat{\mathbf{y}}_\mathbf{P} \times \hat{\mathbf{z}}_\mathbf{P}$.

Since $[\hat{\mathbf{x}}_\mathbf{B}, \hat{\mathbf{y}}_\mathbf{B}, \hat{\mathbf{z}}_\mathbf{B}]$ represents a rotation matrix, the orientation of the mirror can be calculated in a convenient form, for example Euler angles.

Inverse Kinematics: Base translations

The reverse calculations, *i.e.* calculation of the base translations from known tooling point coordinates and angles, are more straightforward. The vectors $\hat{\mathbf{x}}_\mathbf{P}$, $\hat{\mathbf{y}}_\mathbf{P}$, $\hat{\mathbf{z}}_\mathbf{P}$ can be calculated from the rotation angles of the platform. With these unit vectors, it is possible to calculate $\mathbf{t}_i$. Eqn. (2.11) provides $\mathbf{t}_2$. Next, $\mathbf{t}_1 = \mathbf{t}_2 + h_3\,\hat{\mathbf{y}}_\mathbf{P}$, and the remaining vector can be calculated via the cosine-rule:

$$\begin{aligned} \cos\chi_2 &= (h_1^2 + h_3^2 - h_2^2)/(2h_1h_3) \\ \sin\chi_2 &= \sqrt{1 - \cos\chi_2^2} \\ \mathbf{t}_3 &= \mathbf{t}_2 + \hat{\mathbf{x}}_\mathbf{P}\, h_1 \sin\chi_2 + \hat{\mathbf{y}}_\mathbf{P}\, h_1 \cos\chi_2 \end{aligned} \tag{2.12}$$

with χ_2 being the angle between the sides h_1 and h_3. The tilt angles, ϕ_i, of the legs can be calculated using $\sin\phi_i = \mathbf{t}_i \cdot \hat{\mathbf{z}}_\mathbf{B}/l_i$ Here, ϕ_i can be chosen to be either positive or negative, depending on which solution is required. Finally, Eqn. (2.2) gives the vectors $\mathbf{v}_i$ which can be inserted in to Eqn. (2.1) to give the base vectors $\mathbf{b}_i$.

3. Numerical testing of the tripod

The Python class containing the calculations from Section 2 was implemented and tested, to see if certain movements (namely translation in the z_B direction and rotation about the y_B and z_B axes) produced the expected changes in $\mathbf{c}$. These movements were selected because they are the most important and mechanically complex movements needed to position the tripod for use on the beamline. The tripod is set up with the x_B axis parallel to the incoming x-ray beam, so it is important that the sample can be moved into and out of the beam (the z_B direction or the y_B direction, but translation in the y_B direction is straightforward) and rotated about the two axes that are not parallel to the beamline (the y_B and z_B axes). The beam is cylindrically symmetric so rotations about the x_B axis are unimportant. The tests simulated the tripod being moved and then returned to its initial position, as theoretically, the perfect tripod being simulated should return to exactly the same position in which it started. This allowed the accuracy of both the forward calculations and the inverse calculations to be checked.

4. Analysis of results of numerical testing

The numerical tests show very accurate results - reversing the calculations to reproduce the base translations produces near-identical results to the initial calculations, accurate

to 10nm, the accuracy of IEEE arithmetic. The slight discrepancy may be due to the fact that the analytic solution to Equations (2.3) requires the parameters to be accurate to the same degree of accuracy as the solutions. The impact of inaccuracies in the parameters was checked by adding the error in each initial value onto each of the initial value and seeing how this changed the final values.

The uncertainties of 0.001mm in the tripod leg positions, 0.1mm in the tripod leg lengths and 0.001mm in the tripod tooling point are large enough to explain the zero errors seen in the values of $\mathbf{X}$ and $\mathbf{Y}$ when the reverse calculations are carried out as checks.

The issue may also be due to the choice of parameters - there are two solutions very close together and they are close to a maximum on the graph of Equation (2.10) which makes the correct solution harder to find exactly - some polynomial solvers slow down or become less precise when looking for solutions near a maximum. Having one plane of reflection symmetry seems to give two solutions very close together at a maximum, but having no planes or two planes does not seem to produce the same pattern of solutions. Additionally, the tripod has been designed so that the platform is least stable and motion is least precise in the direction parallel to the direction of the beam, x_P, as motion in this direction is not relevant. This may have an effect on the motion in the other two directions as well.

From Figure 3, a plot of the three original simultaneous equations, it can be seen that one of the visible intersection points, the point at the top of the plot, is on a line of intersection of Equations (2.6b) and (2.6c) - this corresponds to a repeated root. The plot has rotational symmetry of order 2 so it can be determined that there are in total 6 distinct real intersection points, two of which correspond to repeated roots. Figure 4 shows the two equations remaining after one of the Bezout elimination steps has taken place. The equations can be displayed in a 2D plot which can be interpreted more easily than the 3D plot in Figure 3. The plot shows 6 distinct real intersection points. Theoretically, there should be 8 real intersection points, so this also suggests that there are two repeated roots. In Figure 5 it can be seen that the solution for ϕ_1 corresponding to the position of the tripod are at a repeated root, which may impact the accuracy that the numerical solver can achieve. Figure 6 shows that when there is no reflectional symmetry, there are no repeated roots, suggesting that the repeated roots may be connected to the symmetry of the tripod.

REFERENCES

1. Collins, S. P., Bombardi, A., Marshall, A. R., Williams, J. H., Barlow, G., Day, A. G., Pearson, M. R., Woolliscroft, R. J., Walton, R. D., Beutier, G. & Nisbet, G. 2009. AIP Conf. Proc. **1234**, 303 306
2. Schadlbauer, J., Husty, M., Caro, S. & Wenger, P. 2013. Self Motions of 3-RPS Manipulators. *Frontiers of Mechanical Engineering.* **Vol. 8, No. 1**, 62-69.
3. Rao, N. 2010. Synthesis of a spatial 3-RPS parallel manipulator based on physical constraints. *Sādhanā.* **Vol. 35, Part 6**, 739-746
4. Ibrahim, O. & Khalil, W. 2007. Kinematics and Dynamic Modelling of the 3-RPS Parallel Manipulator. *IFToMM Conf. Proc.*
4. Nanua, P., Waldron, K. J. & Murthy, V. 1990. Direct Kinematic Solution of a Stewart Platform. *IEEE Transactions on Robotics and Automation.* **Vol. 6, No. 4**, 438-444.
5. Ben-Horin, R., Shosham, M. & Djerassi, S. 1999. Kinematics, Dynamics and Construction of a Planarly Actuated Parallel robot. Israel Institue of Technology.
6. Salmon, G. 1964. Lessons Introductory to the Modern Higher Algebra (5th ed.). *New York, Chelsea.* p76-77; p82-83.

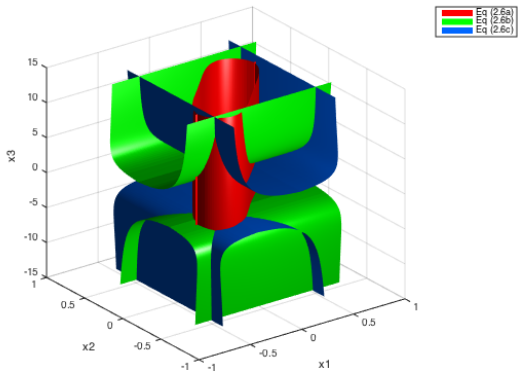

FIGURE 3. A plot of Equations (2.6*a*), (2.6*b*) and (2.6*c*) to show their intersection points, the points where there are solutions to Equations (2.3).

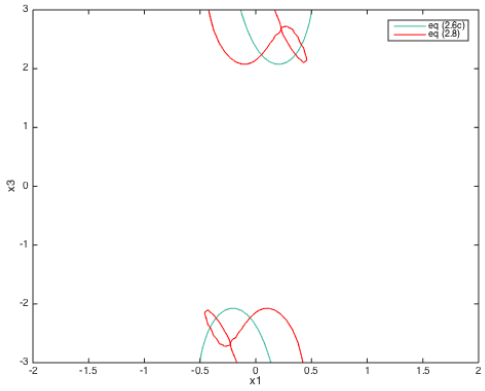

FIGURE 4. After one application of Bezout elimination, one of the variables, in this case x_2, has been eliminated. This leaves two equations in x_1 and x_3, Equation (2.6*c*) and Equation (2.7).

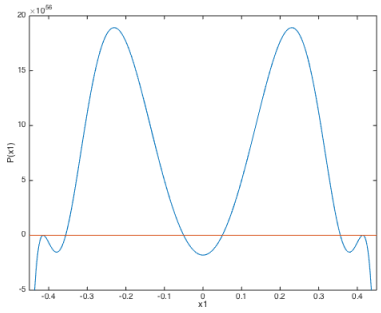

FIGURE 5. A plot to show the solutions of Equation 2.10 graphically. It can be seen that the solutions corresponding to the position of the tripod are at a maximum ($x_1 \approx 0.414$), giving a repeated root. There is another repeated root at $x_1 \approx -0.414$, showing the negative ϕ solution for x_1.

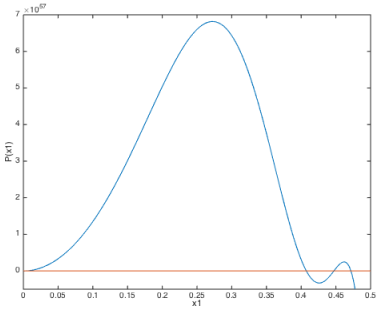

FIGURE 6. A plot to show the solutions of Equation 2.10 graphically, when $l_1 = l_2 + 20$. With no reflectional symmetry, there are no longer any repeated roots, suggesting that the repeated roots may be linked to the symmetry of the tripod.

7. PESIC, P. 2003 Abel's Proof: An Essay on the Sources and Meaning of Mathematical Unsolvability. *MIT Press.* p85-94; p155-174.
8. HORN, R. A. & JOHNSON, C. R. 1999. Matrix Analysis. *Cambridge University Press.* p146-147.