

# Forward Kinematics of a Stewart Platform Mechanism

Domagoj Jakobović

Faculty of Electrical Engineering and Computing  
Unska 3, 10000 Zagreb  
Croatia  
domagoj.jakobovic@fer.hr

Leo Budin

Faculty of Electrical Engineering and Computing  
Unska 3, 10000 Zagreb  
Croatia  
leo.budin@fer.hr

**Abstract** – Parallel kinematic manipulators possess some inherent advantages over their serial counterparts, such as rigidity, precision and higher workload. However, their movement control is restricted to inverse kinematics only, because of the complexity of the forward kinematics relations. The aim of this work is to combine different mathematical representations of the forward kinematics problem with various optimization algorithms and find a suitable combination that may be utilized in real-time environment. Additionally, we note the existence of equivalent trajectories of the mobile platform and suggest an adaptation to the solving method that, having satisfied certain assumptions, is able to successfully solve the forward kinematics problem in real-time conditions with very high precision.

## I. INTRODUCTION

The Stewart platform mechanism, mainly referred to as *hexapod*, is a parallel kinematic structure that can be used as a basis for controlled motion with 6 degrees of freedom (d.o.f.), such as manufacturing processes and precise manipulative tasks. The mechanism itself consists of a stationary platform (base platform, the base) and mobile platform that are connected via six struts mounted on universal joints. The struts have an in-built mechanism that allows changing the length of each individual strut. The desired position and orientation of the mobile platform is achieved by combining the lengths of the six struts, transforming the six transitional d.o.f. into three positional and three orientational ones. The lengths of the struts cannot, of course, be changed independently, but only in such a fashion that the hexapod construction allows.

Parallel manipulators have received an increasing attention in the past two decades due to their inherent advantages over their conventional sequential counterparts. These include, among others, more robust mechanical structure and higher base frequencies. The strength – mass ratio is also in favour of parallel manipulators, which allows for relatively large workloads. While the positioning error in serial mechanisms is accumulated through a series of links, this is not the case with parallel manipulators, which are consequently capable of performing very precise tasks.

On the other hand, these mechanisms have a relatively small workspace, limited with maximum strut lengths and angle values at the joints, as well as their dimensions. However, the main difficulty with parallel manipulators is the complexity of controlling their movement. The problem of inverse kinematics for hexapod parallel manipulators can be defined as finding the strut lengths needed to position the mobile platform in a certain position with desired orientation. The solution to this problem is indeed not at all complex and can be computed in a very short time. Furthermore, the computation of length for each strut can be carried away independently in parallel, which can additionally speed up the process. This

procedure is used to guide the mobile platform in controlling its movement.

The forward kinematics of a parallel manipulator is finding the position and orientation of the mobile platform when the strut lengths are known. This problem has no known closed form solution for the most general 6-6 form of hexapod manipulator (with six joints on the base and six on the mobile platform). This procedure would be invaluable in controlling the force-feedback loop of the manipulator. It would also provide new application possibilities for a hexapod mechanism, such as a force-torque sensor, position-orientation sensor etc.

In this work several mathematical representations of the forward kinematics problem, in the form of optimization functions, are combined with various optimization algorithms and adaptation methods in order to find an efficient procedure that would allow for precise forward kinematics solving in real-time conditions. In the next section we define the most promising representations of the problem. Optimization algorithms and some adaptations regarding the problem at hand are described in section III. In section IV the experimental results are presented and closing remarks are stated in section V.

## II. THE FORWARD KINEMATICS PROBLEM

The forward kinematics relations for a hexapod machine can be mathematically formulated in several ways. Every representation of the problem can have its advantages and disadvantages which become emphasized when a different optimization algorithm is applied. In this section the five mathematical representations used as an optimization target functions are described.

### A. The position and orientation of the mobile platform

In order to define a forward kinematics problem we have to represent the actual hexapod configuration, i.e. the actual position and orientation of the mobile platform. The most common approach utilizes the three positional coordinates of the center of the mobile platform and three angles that define its orientation. The coordinates are represented by vector  $\vec{t}$  :

$$\vec{t} = \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} \quad (1)$$

and the three rotational angles are defined as *roll-pitch-yaw* angles  $\alpha$ ,  $\beta$  and  $\gamma$ . The angle values represent the consecutive rotation about the  $x$ ,  $y$  and  $z$  axis, respectively. The hexapod geometry is defined with six vectors for base and six vectors for mobile platform, which define the six joint coordinates on each platform:

$$\vec{b}_i = \begin{bmatrix} b_{ix} \\ b_{iy} \\ 0 \end{bmatrix}, \quad \vec{p}_i = \begin{bmatrix} p_{ix} \\ p_{iy} \\ 0 \end{bmatrix}, \quad i = 1, \dots, 6. \quad (2)$$

The above vectors are represented in local coordinate systems of the base and mobile platform and are of constant value. The base and mobile platform are presumed to be planar, which can be perceived from the  $z$  coordinate of the joint vectors. The strut vectors  $\vec{l}_i$  can then be expressed as

$$\vec{l}_i = -\vec{b}_i + \vec{t} + \underline{R} \cdot \vec{p}_i, \quad i = 1, \dots, 6, \quad (3)$$

where  $\underline{R}$  is the rotational matrix, calculated from three rotational angles. If the position and orientation of the mobile platform is known, the length of each strut is

$$l_i = D(\vec{b}_i, \vec{t} + \underline{R} \cdot \vec{p}_i), \quad i = 1, \dots, 6, \quad (4)$$

where  $D$  represents the Euclidean distance between the vector pairs. For an arbitrary solution to a forward kinematics problem, i.e. arbitrary position and orientation of the mobile, the error can be expressed as the sum of squares of differences between the calculated and actual length values:

$$f = \sum_{i=1}^6 \left( D(\vec{b}_i, \vec{t} + \underline{R} \cdot \vec{p}_i)^2 - l_i^2 \right)^2. \quad (5)$$

Having stated the above relations, we can define the first optimization function and the related unknowns as

$$F_1 = \sum_{i=1}^6 \left[ D(\vec{b}_i, \vec{t} + \underline{R} \cdot \vec{p}_i)^2 - l_i^2 \right]^2. \quad (6)$$

$$\vec{X}_1 = [t_x \quad t_y \quad t_z \quad \alpha \quad \beta \quad \gamma]^T$$

This function is highly non-linear due to the trigonometric functions included in the calculation of the rotation matrix. On the other hand, it is derivable and is the most commonly used one for the representation of the forward kinematics problem.

### B. The canonical formulation of the forward kinematics

The idea behind this approach [1] is to use the elements of the rotation matrix, rather than the angle values, to represent orientation:

$${}^B \underline{R} = [\vec{n} \quad \vec{o} \quad \vec{a}] = \begin{bmatrix} n_x & o_x & a_x \\ n_y & o_y & a_y \\ n_z & o_z & a_z \end{bmatrix}. \quad (7)$$

The elements of the rotation matrix are interdependent in the following fashion:

$$\vec{n} \cdot \vec{n} = n_x^2 + n_y^2 + n_z^2 = 1, \quad (8)$$

$$\vec{o} \cdot \vec{o} = o_x^2 + o_y^2 + o_z^2 = 1, \quad (9)$$

$$\vec{n} \cdot \vec{o} = n_x o_x + n_y o_y + n_z o_z = 0, \quad (10)$$

$$\vec{a} = \vec{n} \times \vec{o}. \quad (11)$$

Without loss of generality we can position the origins of the local coordinate systems of the base and mobile platform at the strut joints with index one, as shown in Fig. 1, which gives us the following parameter values:

$$b_{1x} = b_{1y} = p_{1x} = p_{1y} = b_{2y} = p_{2y} = 0. \quad (12)$$

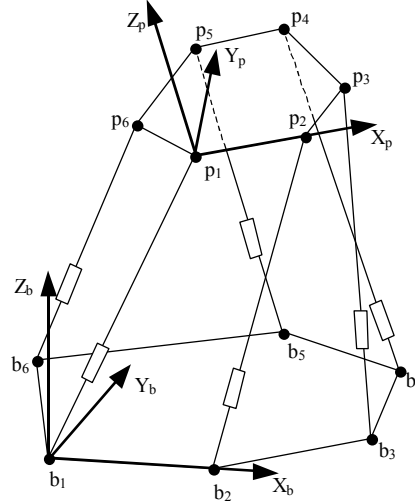


Fig. 1 Positioning of coordinate systems for base and mobile platform

After extensive simplifications, the forward kinematics can be expressed as a system of 9 equations with 9 unknowns:

$$n_x^2 + n_y^2 + n_z^2 = 1, \quad (13)$$

$$o_x^2 + o_y^2 + o_z^2 = 1, \quad (14)$$

$$t_x^2 + t_y^2 + t_z^2 = l_1^2, \quad (15)$$

$$n_x o_x + n_y o_y + n_z o_z = 0, \quad (16)$$

$$n_x t_x + n_y t_y + n_z t_z = A_1 t_x + A_2 n_x + A, \quad (17)$$

$$o_x t_x + o_y t_y + o_z t_z = B_1 t_x + B_2 n_x + B_3 o_x + B_4 t_y + B_5 n_y + B_6 o_y + B, \quad (18)$$

$$C_{41} t_x + C_{42} n_x + C_{43} o_x + C_{44} t_y + C_{45} n_y + C_{46} o_y = C_4, \quad (19)$$

$$C_{51} t_x + C_{52} n_x + C_{53} o_x + C_{54} t_y + C_{55} n_y + C_{56} o_y = C_5, \quad (20)$$

$$C_{61} t_x + C_{62} n_x + C_{63} o_x + C_{64} t_y + C_{65} n_y + C_{66} o_y = C_6, \quad (21)$$

where the constants are

$$A_1 = b_{2x} / p_{2x}, \quad A_2 = b_{2x}, \quad A = K_2 / p_{2x}. \quad (22)$$

$$B_1 = \frac{b_{3x} p_{2x} - b_{2x} p_{3x}}{p_{2x} p_{3y}}, \quad B_2 = \frac{(b_{3x} - b_{2x}) p_{3x}}{p_{3y}},$$

$$B_3 = b_{3x}, \quad B_4 = b_{3y} / p_{3y},$$

$$B_5 = \frac{b_{3y} p_{3x}}{p_{3y}}, \quad B_6 = b_{3y}, \quad (23)$$

$$B = \frac{K_3 p_{2x} - K_2 p_{3x}}{p_{2x} p_{3y}}.$$

$$\begin{aligned}
C_{i1} &= A_1 p_{ix} + B_1 p_{iy} - b_{ix}, \\
C_{i2} &= A_2 p_{ix} + B_2 p_{iy} - b_{ix} p_{ix}, \\
C_{i3} &= B_3 p_{iy} - b_{ix} p_{iy}, \\
C_{i4} &= B_4 p_{iy} - b_{iy}, \\
C_{i5} &= B_5 p_{iy} - b_{iy} p_{ix}, \\
C_{i6} &= B_6 p_{iy} - b_{iy} p_{iy}, \\
C_i &= K_i - \frac{K_2}{p_{2x}} p_{ix} - \frac{K_3 p_{2x} - K_2 p_{3x}}{p_{2x} p_{3y}} p_{iy}. \\
K_i &= \frac{(l_i^2 - b_{ix}^2 - b_{iy}^2 - p_{ix}^2 - p_{iy}^2 - l_1^2)}{2}.
\end{aligned} \tag{24}$$

The optimization function  $F_2$  and the related variables vector  $X_2$  are then defined as

$$\begin{aligned}
F_2 &= (n_x^2 + n_y^2 + n_z^2 - 1)^2 + (o_x^2 + o_y^2 + o_z^2 - 1)^2 \\
&+ (t_x^2 + t_y^2 + t_z^2 - l_1^2)^2 \\
&+ (n_x o_x + n_y o_y + n_z o_z)^2 \\
&+ (n_x t_x + n_y t_y + n_z t_z - A_1 t_x - A_2 n_x - A)^2 \\
&+ \left( \begin{array}{c} o_x t_x + o_y t_y + o_z t_z \\ -B_1 t_x - B_2 n_x - B_3 o_x \\ -B_4 t_y - B_5 n_y - B_6 o_y - B \end{array} \right)^2 \\
&+ \left( \begin{array}{c} C_{41} t_x + C_{42} n_x + C_{43} o_x + C_{44} t_y \\ + C_{45} n_y + C_{46} o_y - C_4 \end{array} \right)^2 \\
&+ \left( \begin{array}{c} C_{51} t_x + C_{52} n_x + C_{53} o_x + C_{54} t_y \\ + C_{55} n_y + C_{56} o_y - C_5 \end{array} \right)^2 \\
&+ \left( \begin{array}{c} C_{61} t_x + C_{62} n_x + C_{63} o_x + C_{64} t_y \\ + C_{65} n_y + C_{66} o_y - C_6 \end{array} \right)^2, \\
\bar{X}_2 &= [n_x \quad n_y \quad n_z \quad o_x \quad o_y \quad o_z \quad t_x \quad t_y \quad t_z]^T.
\end{aligned} \tag{26}$$

### C. Reduced canonical formulation

The equations (19)-(21) are of linear form which can be used to reduce the number of variables without introducing additional complexity in the system. Three of the six variables  $t_x$ ,  $n_x$ ,  $o_x$ ,  $t_y$ ,  $n_y$  and  $o_y$  can be replaced with linear combinations of the other three, which leaves us with only six unknowns. For instance, if we choose to eliminate the following variables

$$\begin{aligned}
t_x &= D_{11} n_x + D_{12} o_x + D_{13} o_y + D_1, \\
t_y &= D_{21} n_x + D_{22} o_x + D_{23} o_y + D_2, \\
n_y &= D_{31} n_x + D_{32} o_x + D_{33} o_y + D_3,
\end{aligned} \tag{27}$$

we can define another target function as

$$\begin{aligned}
F_3 &= (n_x^2 + n_y^2 + n_z^2 - 1)^2 + (o_x^2 + o_y^2 + o_z^2 - 1)^2 \\
&+ (t_x^2 + t_y^2 + t_z^2 - l_1^2)^2 \\
&+ (n_x o_x + n_y o_y + n_z o_z)^2 \\
&+ (n_x t_x + n_y t_y + n_z t_z - A_1 t_x - A_2 n_x - A)^2 \\
&+ \left( \begin{array}{c} o_x t_x + o_y t_y + o_z t_z - B_1 t_x - B_2 n_x \\ -B_3 o_x - B_4 t_y - B_5 n_y - B_6 o_y - B \end{array} \right)^2 \\
\bar{X}_3 &= [n_x \quad n_z \quad o_x \quad o_y \quad o_z \quad t_z]^T.
\end{aligned} \tag{28}$$

It is possible to further reduce the number of unknowns to as low as three, but with inevitable increase in equation complexity [1]. That way the fourth optimization function can be defined as

$$\begin{aligned}
F_4 &= \left[ (n_x o_x - n_y o_x)^2 - (n_x^2 + n_y^2 + o_x^2 + o_y^2) + 1 \right]^2 \\
&+ \left[ (n_x t_y - n_y t_x)^2 - (A_1 t_x + A_2 n_x + A)^2 \right. \\
&\quad \left. + 2(A_1 t_x + A_2 n_x + A)(n_x t_x + n_y t_y) \right. \\
&\quad \left. - (t_x^2 + t_y^2) + l_1^2 (1 - n_x^2 - n_y^2) \right]^2 \\
&+ \left[ 2(E_1 n_x + E_2 o_x + E_3 o_y + E)(o_x t_x + o_y t_y) \right. \\
&\quad \left. + (o_x t_y - o_y t_x)^2 - (E_1 n_x + E_2 o_x + E_3 o_y + E)^2 \right. \\
&\quad \left. - (t_x^2 + t_y^2) + l_1^2 (1 - o_x^2 - o_y^2) \right]^2 \\
\bar{X}_4 &= [n_x \quad o_x \quad o_y]^T,
\end{aligned} \tag{29}$$

where the constant elements are

$$\begin{aligned}
E_1 &= B_2 + B_1 D_{11} + B_4 D_{21} + B_5 D_{31}, \\
E_2 &= B_3 + B_1 D_{12} + B_4 D_{22} + B_5 D_{32}, \\
E_3 &= B_6 + B_1 D_{13} + B_4 D_{23} + B_5 D_{33}, \\
E &= B + B_1 D_1 + B_4 D_2 + B_5 D_3.
\end{aligned} \tag{30}$$

### D. Vector defined rotation

The arbitrary orientation can be described with a single rotation about a universal rotation vector [3]. If the rotation vector, whose length equals one, is denoted with  $\bar{k}$  and the rotation angle with  $\theta$ , then we can use the components of their product

$$\bar{r} = \theta \cdot \bar{k} \tag{31}$$

to represent a potential solution to the forward kinematics problem [2]:

$$\bar{X}_5 = [t_x \quad t_y \quad t_z \quad r_x \quad r_y \quad r_z]^T. \tag{32}$$

The conversion from *roll-pitch-yaw* angles to rotation vector and vice versa can be found in [3]. The optimization function  $F_5$  is defined in the same way as the function  $F_1$ , only the elements of the rotation matrix have to be calculated from vector  $\bar{r}$  prior to function evaluation.

## III. THE OPTIMIZATION ALGORITHMS

The forward kinematics problem is presented as five optimization functions for which the optimization algorithm has to find the minimum, the value of the functions being the error of the estimated solution. Several optimization methods have been applied to each of the functions in order to find an effective combination which would allow for real-time application. The algorithms applied in this work are Powell's method, Hooke-Jeeves', steepest descent search, Newton-Raphson's (NR) method, NR method with constant Jacobian and Fletcher-Powell algorithm. Each of the algorithms is found to be more suitable for some types of functions than for others, which can be perceived from experimental results later in this paper.

### A. Estimating the initial solution by variable elimination

The most reduced canonical formulation (function  $F_4$ ) of forward kinematics problem contains only three independent variables. The choice of those variables can be altered in the reduction process, so it is possible to have

different variable triplets in three-dimensional reduced form. However, a single solution in three-dimensional space represents not one, but a whole family of hexapod configurations, because a single configuration is fully described only in six-dimensional space. The form of the set of hexapod configurations included in a single solution is determined by the variables *not* present in the remaining three. For example, if the reduced form solution is represented with variables  $n_x$ ,  $o_x$  and  $o_y$ , then none of the variables which define the mobile *position*, i.e. variables  $t_x$ ,  $t_y$  and  $t_z$ , are not present in the solution. Hence, every hexapod configuration with mobile platform orientation matching the three values of  $n_x$ ,  $o_x$  and  $o_y$ , but with any arbitrary *position*, will be represented with the same solution in three-dimensional space. The exact position values are restored through constants' values once the transformation to six-dimensional space is undertaken. This property is used to effectively estimate the initial forward kinematics solution, presuming an orientation of the mobile platform from the previous last known position.

#### IV. EXPERIMENTAL RESULTS

Solving of forward kinematics was simulated in static and dynamic conditions. The static experiments included all the combinations of five optimization functions and six optimization algorithms applied to 20 test configurations. The goal was to find the combination which would yield the best results considering the convergence, speed and accuracy. The most promising combinations were then tested in dynamic conditions, where the algorithm had to track a preset trajectory of the mobile platform with as small error and as large sampling frequency as possible. Those combinations include Hooke-Jeeves' algorithm with function  $F_1$  and Fletcher-Powell method with functions  $F_2$  and  $F_3$ , but the most successful optimization method was Newton-Raphson's algorithm applied to function  $F_3$ . That combination has achieved the best results in dynamical testing as well.

In dynamic simulation, the starting hexapod configuration is known and serves as an initial solution. During the sampling period  $T$  the algorithm has to find the new solution, which will become the initial solution in the next cycle. Several hexapod movements were defined as time dependant functions of the position and orientation of mobile platform. One of those trajectories, hereafter denoted as  $A$ , is defined with

$$\begin{aligned} x(t) &= 2 \cdot \sin\left(t \frac{\pi}{2}\right), \\ y(t) &= 2.2 \cdot \cos\left(t \frac{\pi}{2}\right), \\ z(t) &= 7 + 1.5 \cdot \sin(2t), \\ \alpha(t) &= 25 \cdot \sin(1.8t), \\ \beta(t) &= 20 \cdot \sin\left(\frac{t}{2}\right) + 5 \cdot \cos(4t), \\ \gamma(t) &= 15 \cdot \arctan(2t - 4), \\ 0 &\leq t \leq 4. \end{aligned} \quad (33)$$

The results of the dynamic simulation are presented in the form of a graph where errors in the three rotation angles and three position coordinates of the mobile are pictured. The sampling period  $T$  was set to 2 ms, which equals to a 500 Hz sampling frequency. The errors shown

represent the absolute difference between the calculated and the actual hexapod configuration. Due to the large number of cycles, the error is defined as the biggest absolute error value in the last 100 ms, so the graphs in each point show the worst case in the last 100 ms of simulation. The errors are presented separately for angles, in degrees, and position coordinates. The errors for movement  $A$  and Newton-Raphson algorithm with function  $F_3$  are shown in Fig. 2 and Fig. 3.

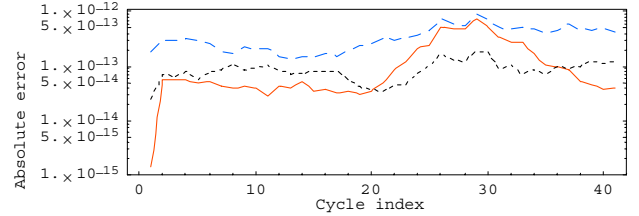


Fig. 2 Absolute angle error ( $\alpha =$  —,  $\beta =$  - - -,  $\gamma =$  ·····), NR algorithm with  $F_3$ , movement  $A$

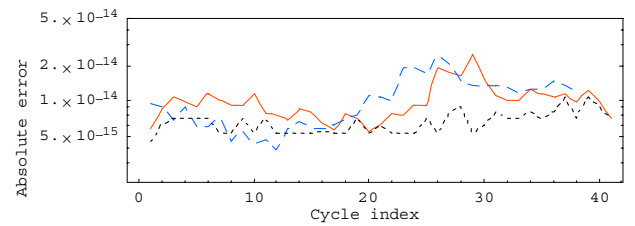


Fig. 3 Absolute coordinate error ( $x =$  —,  $y =$  - - -,  $z =$  ·····), NR algorithm with  $F_3$ , movement  $A$

The achieved level of accuracy is very high as the absolute error is not higher than  $10^{-12}$  both for angles and coordinates.

Another trajectory is derived from the described one by enlarging some of the amplitudes in (33), which is denoted as movement  $B$  (the altered values are in boldface):

$$\begin{aligned} x(t) &= 2 \cdot \sin\left(t \frac{\pi}{2}\right), \\ y(t) &= \mathbf{2.2} \cdot \cos\left(t \frac{\pi}{2}\right), \\ z(t) &= \mathbf{8} + \mathbf{3} \cdot \sin(2t), \\ \alpha(t) &= \mathbf{55} \cdot \sin(1.8t), \\ \beta(t) &= \mathbf{30} \cdot \sin\left(\frac{t}{2}\right) + 5 \cdot \cos(4t), \\ \gamma(t) &= 15 \cdot \arctan(2t - 4), \\ 0 &\leq t \leq 4. \end{aligned} \quad (34)$$

The movement  $B$  errors are shown in Fig. 4 and Fig. 5.

While still low, the error for movement  $B$  has two distinctive peaks at certain points in simulated motion. What is the cause of those peaks? Mathematical analysis has shown ([4], [5], [6]) that there may exist up to 40 distinctive solutions for forward kinematics problem for Stewart platform with planar base and mobile platform. Those 40 solutions are divided in two groups of 20 between which the only difference is the sign of the  $z$  coordinate. In another words, only one half of those mathematical solutions have a physical meaning, i.e. those with positive  $z$  axis value. But the existence of multiple solutions for the same set of strut lengths may prove as a problem for the solving method.

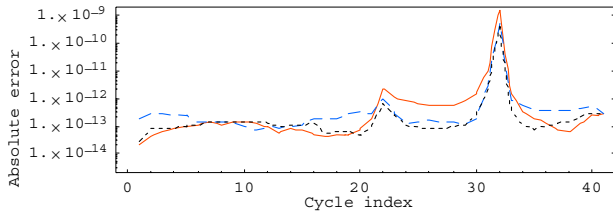


Fig. 4 Absolute angle error ( $\alpha =$  —,  $\beta =$  - - -,  $\gamma =$  ·····), NR algorithm with  $F_3$ , movement  $B$

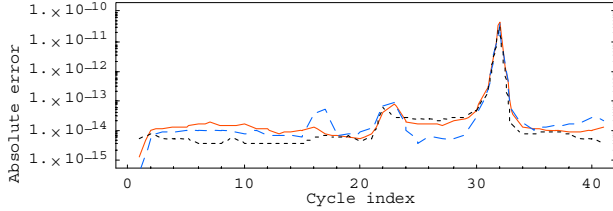


Fig. 5 Absolute coordinate error ( $x =$  —,  $y =$  - - -,  $z =$  ·····), NR algorithm with  $F_3$ , movement  $B$

Let us suppose that in one hexapod configuration there exists no other forward kinematics solution for actual set of strut lengths, but that in some other configuration there exist several of them. If hexapod in its movement passes through those two configurations, then at a certain point in between there has to be a *division* point where the number of solutions increases. In those division points the solving algorithm may, unfortunately, begin to follow any of the possible paths, because any of them represents a valid forward kinematics solution! That is exactly the problem that occurs in movement  $B$ : the algorithm may or may not follow the correct trajectory. If the latter is the case, than the absolute error looks like in Fig. 6 and Fig. 7.

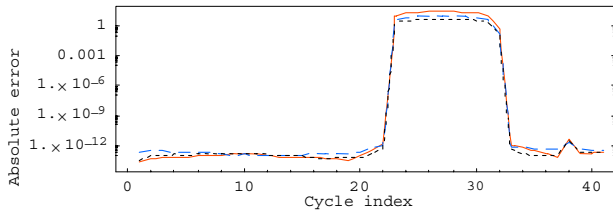


Fig. 6 Absolute angle error ( $\alpha =$  —,  $\beta =$  - - -,  $\gamma =$  ·····), NR algorithm with  $F_3$ , movement  $B$  - division

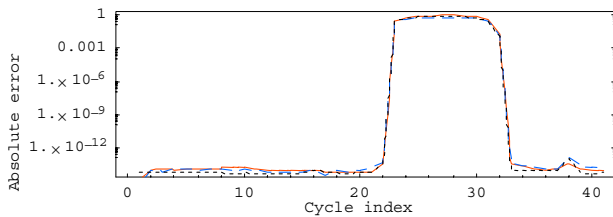


Fig. 7 Absolute coordinate error ( $x =$  —,  $y =$  - - -,  $z =$  ·····), NR algorithm with  $F_3$ , movement  $B$  - division

The algorithm will randomly follow either the correct trajectory or the equivalent one. It is important to note that in both cases the optimization function remains very low (app.  $10^{-30}$  to  $10^{-20}$ ) during the whole process because both trajectories depict a valid solution to the forward kinematics problem. The problem is, only one of them represents the actual hexapod configuration in each point of time. The error between junction points in Fig. 6 and Fig. 7 actually shows the distance between the two equivalent trajectories.

### A. The adaptation method

Without any additional information about the hexapod configuration, such as may be obtained from extra transitional displacement sensors, there is unfortunately no way to determine which of the existent solutions to the forward kinematics problem for the same set of strut lengths describes the actual hexapod configuration. Nevertheless, with some assumptions we may devise a strategy that should keep the solving method on the right track.

If the change of the direction of movement is relatively small during a single period, which is in this case only 2 ms, then we can try to predict the position of the mobile platform in the next cycle. We can use the solutions from the past cycles to construct a straight line and estimate the initial solution in the next iteration. Let the solution in the current iteration be  $\vec{P}_0$  and the solutions from the last two cycles  $\vec{P}_1$  and  $\vec{P}_2$ . Then we can calculate the new initial solution using one of the following methods:

$$\vec{X}_0 = 2 \cdot \vec{P}_0 - \vec{P}_1, \quad (35)$$

$$\vec{X}_0 = 1.5 \cdot \vec{P}_0 - 0.5 \cdot \vec{P}_2, \quad (36)$$

$$\left. \begin{aligned} \vec{T}_1 &= 0.5 \cdot (\vec{P}_0 + \vec{P}_1), \\ \vec{T}_2 &= 0.5 \cdot (\vec{P}_1 + \vec{P}_2), \\ \vec{X}_0 &= 2.5 \cdot \vec{T}_1 - 1.5 \cdot \vec{T}_2 \end{aligned} \right\} \quad (37)$$

The above methods were tested in conjunction with NR algorithm and function  $F_3$  for all the simulated trajectories. The results are very good: the solving method was now able to track the correct solution during the whole simulation process for all three estimation methods. The number of conducted experiments was several hundred times and every time the algorithm's error margin was below  $10^{-11}$  both for angles and coordinates.

However, the described algorithm adaptation will only be successful if the assumption of a small direction change during a few iterations is valid. To test the algorithms behaviour, simulated movement  $B$  was accelerated by factor 2, 4 and 8, while maintaining the same cycle duration of 2 ms. Only by reaching the 8-fold acceleration, when the total movement time equals a very unrealistic half a second, did the algorithm produce significant errors, while still holding to the correct solution. In this case only some differences can be perceived between the three proposed estimation methods. If we set an error margin of  $10^{-3}$  for angles (in degrees!) and  $10^{-5}$  for coordinates, then the percentage of breaking those margins for the three adaptation methods will be approximately equal to those shown in Table I.

TABLE I  
THE PERCENTAGE OF BREACHING ERROR MARGINS

Adaptation method	As (35)	As (36)	As (37)
Error breaching percentage	6 %	11 %	3.6 %

There is no way of telling, without extensive mathematical analysis for every planned mobile platform movement, whether hexapod will pass any division points or hyperplains. That is why the sampling period must be

kept as low as possible, to assure relatively small direction changes during a few cycles.

## V. CONCLUSION

Combining several representations of the forward kinematics problem with optimization techniques, an efficient method for solving the problem was found. For the purposes of real-time simulation, several mobile platform trajectories were mathematically predefined. The task of a forward kinematics solving method was then to track the position and orientation of the mobile platform, knowing the starting position. The solving method was able to determine the exact position and orientation of the mobile platform within insignificant error margins (less than 10 to the power of  $-12$  of the minimum hexapod dimension) and with 500 Hz sampling frequency.

The problem of equivalent trajectories was noted: because of the existence of multiple solutions to forward kinematics, there may exist more than one path that mobile platform can follow while having exactly the same strut lengths in every point of the way. The solving algorithm may, under some circumstances, skip to an equivalent trajectory at certain division points. It has to be said that every such path represents an equal correct solution of the forward kinematics, but only one of them represents the true mobile platform trajectory.

An empirical algorithm was devised which can increase the probability of finding the right solution, and it proved itself successful in every test case. Unfortunately, it cannot be proven that it will do so in every imaginable movement of the mobile platform. The solving method will always find the right solution if the change in the position or moving direction of the mobile platform is relatively small during a few sampling periods. If that condition is fulfilled,

the described method can be used in real-time forward kinematics solving.

## VI. ACKNOWLEDGMENT

This work was carried out within the research project "*Problem-Solving Environments in Engineering*", supported by the Ministry of Science and Technology of the Republic of Croatia.

## VII. REFERENCES

- [1] B. Dasgupta, T.S. Mruthyunjaya, "A Canonical Formulation of the Direct Position Kinematics Problem for a General 6-6 Stewart Platform", *Mech. Mach. Theory*, Vol. 29, No. 6, pp. 819-827, 1994.
- [2] J. -P. Merlet, "Direct Kinematics of Parallel Manipulators", *IEEE Transactions on Robotics and Automation*, Vol. 9, No. 6, pp. 842-845, 1993.
- [3] R. P. Paul, *Robot Manipulators, The MIT Press*, Cambridge, 1981.
- [4] M. Raghavan, "The Stewart Platform of General Geometry has 40 Configurations", *Journal of Mechanical Design*, Vol. 115, pp. 277-282, June 1993.
- [5] M. Husty, "An Algorithm for Solving the Direct Kinematic Of Stewart-Gough-Type Platforms", <ftp://ftp.mrcim.mcgill.edu/pub/techrep/1994/CIM-94-01.pdf>, 1994.
- [6] F. Wen, C. Liang, "Displacement Analysis of the 6-6 Stewart Platform Mechanisms", *Mechanism and Machine Theory*, Vol. 29, No. 4, pp. 547-557, 1994.