

KINEMATIC MODELING OF STEWART-GOUGH PLATFORMS

Pedro Cruz, Ricardo Ferreira
Institute for Systems and Robotics
Av. Rovisco Pais 1, 1049-001 Lisboa, Portugal
{pcruz,ricardo}@isr.ist.utl.pt

João Silva Sequeira
Instituto Superior Técnico
Institute for Systems and Robotics
Av. Rovisco Pais 1, 1049-001 Lisboa, Portugal
jseq@isr.ist.utl.pt

Keywords: Parallel robots, Direct kinematics, Inverse kinematics.

Abstract: This paper describes a method to solve the direct kinematics of a generic Stewart-Gough manipulators. The method is formulated in terms of a search in the space of rigid body transformations. The underlying idea is that the solutions of the direct kinematics can be obtained by moving the end-effector body according rotations and translations and accounting for the rigidity conditions.
The paper presents simulation results for a 6-3 Stewart-Gough robot.

1 INTRODUCTION

This paper presents a method to solve the direct kinematics of Stewart-Gough robots. This work aims at assessing the use of numerical algorithms for a real time application to the control of such robots.

The use of parallel robots undergoes a renewed interest nowadays due to the variety of possible practical applications. Parallel robot structures can be designed which are specially tailored to handle heavy loads with accurate positioning. Roughly, positioning errors in serial kinematic chains tends to propagate additively throughout the chain links. This is not the case with parallel manipulators, which are consequently capable of performing positioning tasks to a high degree of accuracy. Furthermore, the parallel structure inherently distributes the forces/torques by the actuators giving this class of robots high bandwidth dynamic characteristics. Typical applications include flight simulators, shaking tables (used in simulation of the effects of earthquakes in building structures), support structures for the accuracy positioning of instrumentation, medical instrumentation and even entertainment devices (see for instance (Merlet, 2000)).

The generic Stewart-Gough platform is composed of two rigid bodies connected through a number of prismatic actuators as in a parallel arrangement of kinematic chains. Usually six actuators are used, pairing arbitrary points in the two bodies. The linkages between the actuators and the bodies are

made through non-actuated universal or ball-in-socket joints.

Figure 1 presents a Lego-based physical model of a Stewart-Gough platform with six connecting rods of fixed length in place of the linear actuators.

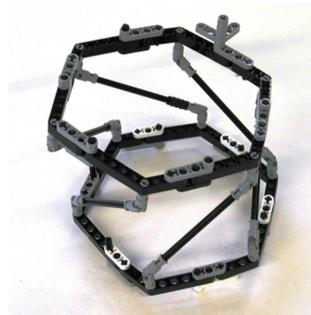


Figure 1: A Stewart-Gough platform model constructed using Lego

Figure 2 illustrates the terminology used in the paper. The two bodies are denoted by B and P and the linear actuators connect them at the anchor points denoted by $\{\mathbf{b}_1, \dots, \mathbf{b}_6\}$ and $\{\mathbf{p}_1, \dots, \mathbf{p}_6\}$, respectively in B and P . For the sake of simplicity, the polygonal line joining the \mathbf{b}_i points and the polygonal line joining the \mathbf{p}_i points, are assumed to form simple polygons. In the sequel B is assumed to be rigidly connected

to the ground, as the base of the robot. The body P moves according to the joint values and the kinematic constraints. The actuator lengths are denoted by l_i , $i = 1, \dots, 6$. The i -th actuator connects the points \mathbf{b}_i and \mathbf{p}_i and has length l_i .

Unlike serial manipulators, the direct kinematics of generic parallel manipulators can not be easily written in closed form. Numerical methods are often used to determine the position and attitude of the P body from the set of joint variables (the lengths of the actuators).

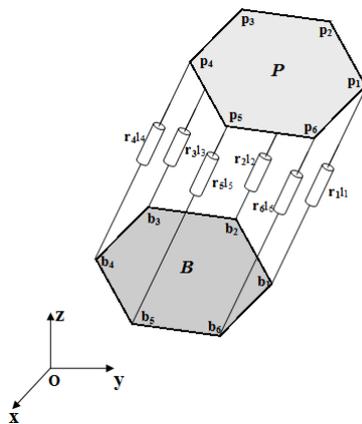


Figure 2: A generic Stewart-Gough platform

The direct kinematics of a generic Stewart-Gough platform has often multiple solutions. For instance, the general Stewart-Gough platform can have 40 real solutions, (Dietmaier, 1998). Practical applications of direct kinematics, e.g. in robot control architectures, often require a one to one correspondence between subsets in the spaces of joint variables and positions and attitudes. Therefore, a relevant aspect in the study of the kinematics of parallel robots is related with the ability of the kinematics solution methods to converge to a particular solution.

Various approaches to the computation of the kinematics of Stewart-Gough robots have been presented in the literature. In (Jakobović and Budin, 2002) the direct kinematics problem is addressed by solving six optimization problems, one for each actuator. Algorithms like Powell's method, Hooke-Jeeves', steepest descent with constant update steps and Fletcher-Powell's were used to solve those problems. The direct kinematics is also addressed as an optimization problem in (Hopkins, 2002), solved using a Newton-Raphson method. The work in (Hopkins, 2002) is

integrated in a control architecture. A hybrid strategy using neural networks and Newton-Raphson techniques is proposed in (Parikh and Lam, 2005). In this strategy the neural network stage is used to obtain the initial estimate for the Newton-Raphson method. Dynamic modeling, a fundamental aspect for high performance control, has been addressed in (Khalil and S., 2004).

This paper presents an algorithm that describes the motion of the P body as a rigid body transformation, i.e., through rotations and translations. Newton's method is used to compute the rigid body transformation matrix that corresponds to the desired solution of the direct kinematics of Stewart-Gough platforms. Simulation results are obtained from one special class of Stewart-Gough platform 6-3/ \wedge^3 configuration (the notation in (Merlet, 2000) is followed hereafter to describe the organisation of the actuators in the robot). The direct kinematics problem is formalized as an optimization problem and simulation results obtained are presented.

The paper is organized as follows. Section 2 describes the direct kinematics problem (the inverse kinematics is trivial) in the space of rigid body transformations. Section 3 describes a set of simulation experiments. Section 4 presents the conclusions of the paper and ongoing work.

2 KINEMATICS MODELING

Table 1 details the notation used in the paper to formulate the kinematics of the robot.

Table 1: Notation used in the paper

l_i	a scalar standing for the i -th actuator length
$\mathbf{p}_i = (p_i^x, p_i^y, p_i^z)$	point in \mathbb{R}^3 belonging to P
$\mathbf{b}_i = (b_i^x, b_i^y, b_i^z)$	point in \mathbb{R}^3 belonging B
\mathbf{r}_i	unit vector in \mathbb{R}^3
$\bar{\mathbf{p}}_i$	the "usual" inclusion of \mathbf{p}_i in the projective space \mathbb{P}^3
\mathbf{q}_i	representation of point \mathbf{p}_i described in a local reference frame (see Figure 3)

Figure 3 illustrates relations between the coordinate reference frames assigned to the rigid bodies that form the robot.

Assuming the rigidity of the two bodies B and P , the kinematics of the robot is completely described by the set of equations,

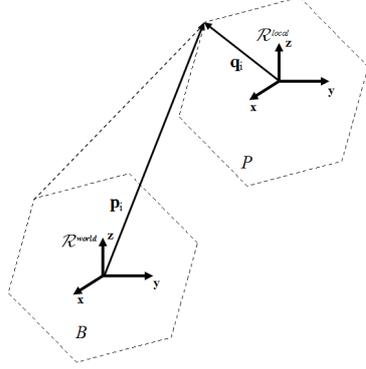


Figure 3: Relations between reference frames

$$(p_i^x - b_i^x)^2 + (p_i^y - b_i^y)^2 + (p_i^z - b_i^z)^2 = l_i^2, \quad i = 1, \dots, 6 \quad (1)$$

meaning that, for any actuator, any achievable configuration must lie in a 2D ball embedded in \mathbb{R}^3 .

Expanding the square terms in the lefthand side of (1) the expression can be written as

$$\bar{\mathbf{p}}_i^T \mathbf{C}_i(\mathbf{b}_i, l_i) \bar{\mathbf{p}}_i = 0, \quad i = 1, \dots, 6 \quad (2)$$

where \mathbf{C}_i is a matrix relating the i -th actuator and the B body defined as

$$\mathbf{C}_i = \begin{bmatrix} 1 & 0 & 0 & -b_i^x \\ 0 & 1 & 0 & -b_i^y \\ 0 & 0 & 1 & -b_i^z \\ -b_i^x & -b_i^y & -b_i^z & \mathbf{b}_i^T \mathbf{b}_i - l_i^2 \end{bmatrix} \quad (3)$$

The solution of the direct kinematics problem expressed by (2) can be obtained by moving the P body in the 3D space. This amounts to finding the adequate rotation and translation such that (2) has a solution. Alternatively, this can also be seen as finding a new reference frame where to describe the \mathbf{p}_i solutions of (2), that is,

$$\bar{\mathbf{p}}_i = \mathbf{T} \bar{\mathbf{q}}_i \quad (4)$$

where $\bar{\mathbf{q}}_i$ describes $\bar{\mathbf{p}}_i$ in a local reference frame and $\mathbf{T} \in SE(3)$ is the homogeneous transformation matrix given by

$$\mathbf{T} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{bmatrix} \quad (5)$$

with \mathbf{R} standing for a rotation matrix and \mathbf{t} standing for the displacement between the origins of the two reference frames. Equation (2) can then be written as

$$\bar{\mathbf{q}}_i^T \mathbf{T}^T \mathbf{C}_i(\mathbf{b}_i, l_i) \mathbf{T} \bar{\mathbf{q}}_i = 0 \quad i = 1, \dots, 6 \quad (6)$$

In addition, it is also necessary to ensure that the structure of the elements in \mathbf{T} is preserved, namely that $\mathbf{R} \in SO(3)$. Given the homogeneous structure of (6) this condition can be relaxed to $\mathbf{R} \in O(3)$ yielding the following 9 additional constraints,

$$\begin{aligned} \mathbf{R}^T \mathbf{R} &= \mathbf{I}_3 \\ \iff [\mathbf{I} \quad \mathbf{0}] \mathbf{T}^T \begin{bmatrix} \mathbf{I} \\ \mathbf{0} \end{bmatrix} [\mathbf{I} \quad \mathbf{0}] \mathbf{T} \begin{bmatrix} \mathbf{I} \\ \mathbf{0} \end{bmatrix} &= \mathbf{I}_3 \\ \iff \mathbf{I}_{3 \times 4} \mathbf{T}^T \mathbf{D} \mathbf{T} \mathbf{I}_{4 \times 3} &= \mathbf{I}_3 \end{aligned}$$

All the 15 restrictions have the general form

$$g_i(\mathbf{T}) = \mathbf{a}_i^T \mathbf{T}^T \mathbf{C}_i \mathbf{T} \mathbf{b}_i - d_i = 0 \quad (7)$$

and hence the solution of the forward kinematics amounts to find a solution for a non-linear system of 15 equations and 12 unknowns.

Expression (7) is thus the formulation of the direct kinematics problem in the space of homogeneous transformations.

2.1 Solution by Newton's method

To obtain a solution of (7) the $g_i(\mathbf{T})$ is expanded in Taylor series, retaining only the first order terms,

$$\begin{aligned} g_i(\mathbf{T} + \mathbf{H}) &= g_i(\mathbf{T}) + \\ &(\nabla g_i)^T|_{\mathbf{T}} \text{vec}(\mathbf{H}) + O(\|\text{vec}(\mathbf{H})\|^2) \end{aligned} \quad (8)$$

where ∇ denotes the gradient, vec denotes the vector operator that stacks the columns of \mathbf{H} to form a vector and

$$\nabla g_i|_{\mathbf{T}} = \text{vec}(\mathbf{C}_i \mathbf{T} \mathbf{b}_i \mathbf{a}_i^T + \mathbf{C}_i^T \mathbf{T} \mathbf{a}_i \mathbf{b}_i^T) \quad (9)$$

Note that not all entries of \mathbf{T} are variables, so 4 entries of the gradient vector need be removed.

Discarding the terms of higher order results in a linear system of 15 equations, compactly written as

$$\mathbf{g}(\mathbf{T} + \mathbf{H}) \approx \mathbf{g}(\mathbf{T}) + \mathbf{J} \mathbf{g}|_{\mathbf{T}} \text{vec}(\mathbf{H}) \quad (10)$$

where $\mathbf{J} \mathbf{g}|_{\mathbf{T}} = [\nabla g_1|_{\mathbf{T}}, \dots, \nabla g_{15}|_{\mathbf{T}}]^T$ denotes the jacobian matrix of \mathbf{g} which depends on the current value of the argument being searched, that is the rigid body transformation \mathbf{T} .

The solution of the direct kinematics problems in this space of transformations is obtained when a motion in this space, expressed by the Jacobian matrix,

verifies the constraints (7), that is, $\mathbf{g}(\mathbf{T} + \mathbf{H}) = \mathbf{0}$, which yields the system

$$\mathbf{J}\mathbf{g}|_{\mathbf{T}} \text{vec}(\mathbf{H}) = -\mathbf{g}(\mathbf{T}) \quad (11)$$

Expression (11) indicates that, starting at some initial configuration the motion of the P body to $\mathbf{T} + \mathbf{H}$ results in an approximate for the solution of the system (recall that the higher order terms in (8) were discarded). By repeating this process for the obtained approximate solution the sequence of transformation converges to the solution of the direct kinematics. Note that the Jacobian is overdetermined and thus a pseudo-inverse must be used.

3 SIMULATION RESULTS

This section presents simulation experiments that illustrate the solution of the direct kinematics of a 6-3 (\wedge^3) robot using the proposed algorithm. In all simulations the initial transformation matrix is

$$\mathbf{T}^{\text{initial}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

3.1 Experiment 1

Table 2 presents the values used as initial conditions by the algorithm. For this experiment, the actuator values were set at $l_1 = 2, l_2 = 2, l_3 = 2.5, l_4 = 2.5, l_5 = 2$ and $l_6 = 2$.

Table 2: Experiment 1: Initial values

Act.	l_i	p_i	b_i	r_i
1	1.1180	(0.7500	(0.5000	(0.2236
		-0.4330	-0.8660	0.3873
		1.0000)	0.0000)	0.8944)
2	1.1180	(0.7500	(1.0000	(-0.2236
		-0.4330	0.0000	-0.3873
		1.0000)	0.0000)	0.8944)
3	1.1180	(0.0000	(0.5000	(-0.4472
		0.8660	0.8660	-0.0000
		1.0000)	0.0000)	0.8944)
4	1.1180	(0.0000	(-0.5000	(0.4472
		0.8660	0.8660	-0.0000
		1.0000)	0.0000)	0.8944)
5	1.1180	(-0.7500	(-1.0000	(0.2236
		-0.4330	0.0000	-0.3873
		1.0000)	0.0000)	0.8944)
6	1.1180	(-0.7500	(-0.5000	(-0.2236
		-0.4330	-0.8660	0.3873
		1.0000)	0.0000)	0.8944)

Figure 4 displays the robot at the final configuration obtained.

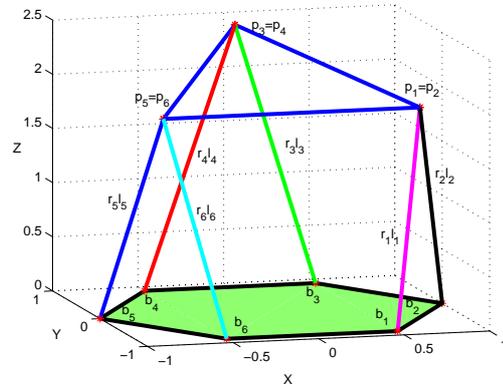


Figure 4: Experiment 1: Final configuration

The algorithm was stopped after 5 iterations, with a final absolute error of 7.5675×10^{-11} . Table 3 displays the lengths of the actuator rods, along with the corresponding positions of the connecting points in the P body.

Table 3: Experiment 1: final results

Act.	l_i	p_i	b_i	r_i
1	2.0000	(0.7500	(0.5000	(0.1250
		-0.4330	-0.8660	0.2165
		1.9365)	0.0000)	0.9682)
2	2.0000	(0.7500	(1.0000	(-0.1250
		-0.4330	0.0000	-0.2165
		1.9365)	0.0000)	0.9682)
3	2.5000	(0.0000	(0.5000	(-0.2000
		0.7614	0.8660	-0.0419
		2.4473)	0.0000)	0.9789)
4	2.5000	(0.0000	(-0.5000	(0.2000
		0.7614	0.8660	-0.0419
		2.4473)	0.0000)	0.9789)
5	2.0000	(-0.7500	(-1.0000	(0.1250
		-0.4330	0.0000	-0.2165
		1.9365)	0.0000)	0.9682)
6	2.0000	(-0.7500	(-0.5000	(-0.1250
		-0.4330	-0.8660	0.2165
		1.9365)	0.0000)	0.9682)

The key point in this experiment is the fast convergence of the algorithm. Figure 5 illustrates, in logarithmic scale, the evolution of the absolute error.

The solution for the optimal rigid body transformation is given by the matrix

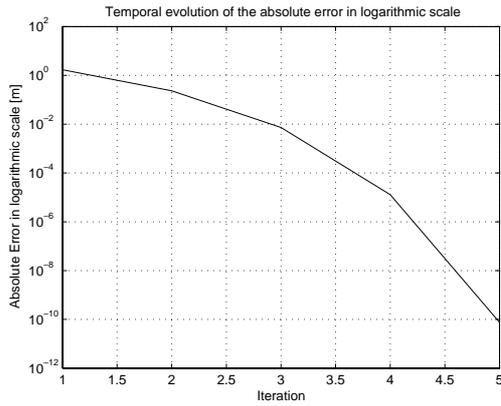


Figure 5: Experiment 1: Temporal evolution of the quadratic error

$$\mathbf{T}^{\text{final}} = \begin{bmatrix} 1.0000 & 0.0000 & 0.0000 & 0.0000 \\ 0.0000 & 0.9195 & -0.3932 & -0.0349 \\ 0.0000 & 0.3932 & 0.9195 & 2.1067 \\ 0.0000 & 0.0000 & 0.0000 & 1.0000 \end{bmatrix}$$

3.2 Experiment 2

Table 4 presents the values used as initial conditions by the algorithm for this second experiment. The goal values for the actuators were set at $l_{1,\dots,6} = 2$.

Table 4: Experiment 2: Initial values

Act.	l_i	p_i	b_i	r_i
1	2.0615	(0.0000 0.8660 1.0000)	(0.5000 -0.8660 0.0000)	(-0.2425 0.8402 0.4851)
2	1.6583	(0.0000 0.8660 1.0000)	(1.0000 0.0000 0.0000)	(-0.6030 0.5222 0.6030)
3	1.6583	(0.7500 -0.4330 1.0000)	(0.5000 0.8660 0.0000)	(0.1508 -0.7833 0.6030)
4	2.0615	(0.7500 -0.4330 1.0000)	(-0.5000 0.8660 0.0000)	(0.6063 -0.6301 0.4851)
5	1.1180	(-0.7500 -0.4330 1.0000)	(-1.0000 0.0000 0.0000)	(0.2236 -0.3873 0.8944)
6	1.1180	(-0.7500 -0.4330 1.0000)	(-0.5000 -0.8660 0.0000)	(-0.2236 0.3873 0.8944)

The final robot configuration obtained from this simulation is presented in Figure 6.

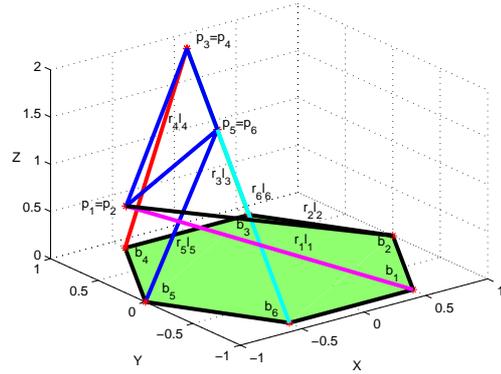


Figure 6: Experiment 2: Final configuration

The algorithm was stopped after 20 iterations, with a final absolute error of 2.0054×10^{-8} . Table 5 displays the lengths of the actuator rods, along with the corresponding positions of the connecting points in the P body.

Table 5: Experiment 2: Final values

Act.	l_i	p_i	b_i	r_i
1	2.0000	(-0.8017 0.4629 0.7346)	(0.5000 -0.8660 0.0000)	(-0.6509 0.6644 0.3673)
2	2.0000	(-0.8017 0.4629 0.7346)	(1.0000 0.0000 0.0000)	(-0.9009 0.2314 0.3673)
3	2.0000	(0.0000 0.8660 1.9365)	(0.5000 0.8660 0.0000)	(-0.2500 0.0000 0.9682)
4	2.0000	(0.0000 0.8660 1.9365)	(-0.5000 0.8660 0.0000)	(0.2500 0.0000 0.9682)
5	2.0000	(-0.7500 -0.4330 1.9365)	(-1.0000 0.0000 0.0000)	(0.1250 -0.2165 0.9682)
6	2.0000	(-0.7500 -0.4330 1.9365)	(-0.5000 -0.8660 0.0000)	(-0.1250 0.2165 0.9682)

As in the previous experiment, the algorithm converges in an acceptable number of iterations. Figure 7 shows the temporal evolution of the error.

For this experiment, the matrix that corresponds to the solution of the optimal rigid body transformation is

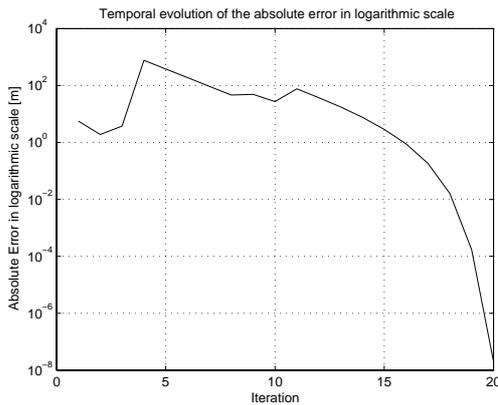


Figure 7: Experiment 2: Temporal evolution of the quadratic error

$$\mathbf{T}^{\text{final}} = \begin{bmatrix} 0.5000 & -0.3285 & -0.8013 & -0.5172 \\ 0.8660 & 0.1897 & 0.4626 & 0.2986 \\ 0.0000 & -0.9253 & 0.3793 & 1.5358 \\ 0.0000 & 0.0000 & 0.0000 & 1.0000 \end{bmatrix}$$

3.3 Experiment 3

The $6 - 3 (\wedge^3)$ structure considered in the experiments is known to have 16 real solutions. In general, these solutions differ significantly from each other. Therefore, when targeting control applications it is important to determine the sensitivity of a particular solution to the initial condition. This is illustrated by comparing this experiment with experiment 2.

In this experiment the values for the actuator are set identical to those in experiment 2 but with the initial position of the P body slightly changed. Table 6 displays the data for this experiment.

Figure 8 illustrates the final configuration of the robot which is clearly different from that obtained in experiment 2.

The algorithm was stopped after 4 iterations, with a final absolute error of 1.052×10^{-7} . Table 7 displays final data obtained in this experiment.

Figure 9 shows the time evolution, in logarithmic scale, of the error.

For this experiment, optimal rigid body transformation matrix is given by

$$\mathbf{T}^{\text{final}} = \begin{bmatrix} 1.0000 & 0.0000 & 0.0000 & 0.0000 \\ 0.0000 & 1.0000 & 0.0000 & 0.0000 \\ 0.0000 & 0.0000 & 1.0000 & 1.9365 \\ 0.0000 & 0.0000 & 0.0000 & 1.0000 \end{bmatrix}$$

Table 6: Experiment 3: Initial values

Act.	l_i	p_i	b_i	r_i
1	1.1180	(0.7500 -0.4330 1.0000)	(0.5000 -0.8660 0.0000)	(0.2236 0.3873 0.8944)
2	1.1180	(0.7500 -0.4330 1.0000)	(1.0000 0.0000 0.0000)	(-0.2236 -0.3873 0.8944)
3	1.1180	(0.0000 0.8660 1.0000)	(0.5000 0.8660 0.0000)	(-0.4472 -0.0000 0.8944)
4	1.1180	(0.0000 0.8660 1.0000)	(-0.5000 0.8660 0.0000)	(0.4472 -0.0000 0.8944)
5	1.1180	(-0.7500 -0.4330 1.0000)	(-1.0000 0.0000 0.0000)	(0.2236 -0.3873 0.8944)
6	1.1180	(-0.7500 -0.4330 1.0000)	(-0.5000 -0.8660 0.0000)	(-0.2236 0.3873 0.8944)

4 CONCLUSIONS

The paper presents a solution for the direct kinematics of a Stewart-Gough robot supported on a Newton method for the search of a rigid body transformation that places the moving body (the P body in the terminology used in the paper) in a configuration compatible with the desired actuator lengths. The experimental results show fast convergence to the solution.

4.1 Ongoing work

Experiments 2 and 3 show the importance of the initial conditions to the final solution. Future work includes the analysis of solutions sensitivity to initial conditions.

High performance applications, namely those requiring the manipulation of heavy loads at large bandwidths, require the knowledge of the dynamics model in order to design the adequate control strategy.

For a large class of applications, e.g., shaking tables for seismic simulations, the typical frequency band goes from 0 up to 20-40 Hz (the typical bandwidth of a real earthquake). Despite the heavy loads considered in such applications, the bandwidth requirements do not impose detailed knowledge on the dynamics of the robot. Thus, the use of control schemes using direct and inverse kinematics solutions obtained by methods such as the one proposed in this paper are acceptable.

For the experiments in Section 3, the algorithm requires less than 20 ms to reach a solution within the aforementioned errors (Matlab implementation).

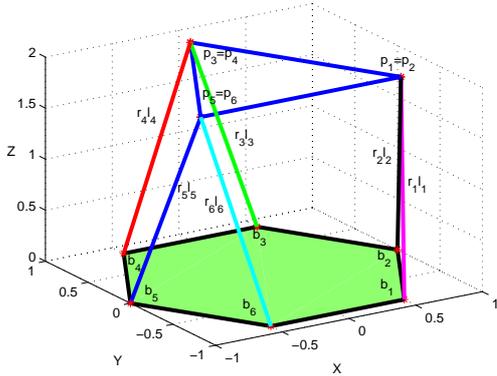


Figure 8: Experiment 3: Final Robot Solution

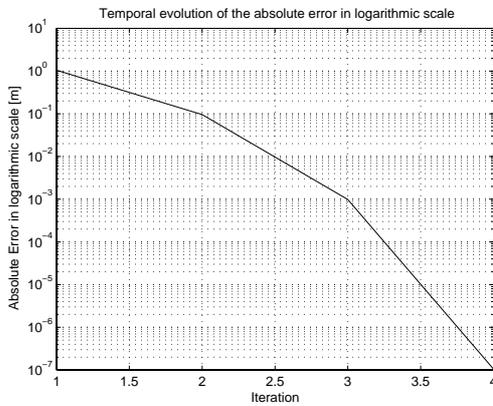


Figure 9: Experiment 3: Temporal evolution of the quadratic error

Therefore, the upper bound on the bandwidth is estimated at about 25 Hz (assuming that 20 ms corresponds to the sample time), this being compatible with real time control of a shaking table.

Figure 10 illustrates the control strategy under evaluation for the shaking table problem. Instead of having the inverse kinematics defining the trajectories for each actuator, this scheme includes both the direct and inverse kinematics in the control loop. The advantage of this procedure is that any modeling errors, e.g., those due to inaccurate positioning of the actuators anchor points in the B and P bodies, are compensated by the control loop.

ACKNOWLEDGEMENTS

This work was supported Programa Operacional Sociedade de Informação (POSI) in the frame of QCA III.

Table 7: Experiment 3: Final values

Act.	l_i	p_i	b_i	r_i
1	2.0000	(0.7500 -0.4330 1.9365)	(0.5000 -0.8660 0.0000)	(0.1250 0.2165 0.9682)
2	2.0000	(0.7500 -0.4330 1.9365)	(1.0000 0.0000 0.0000)	(-0.1250 -0.2165 0.9682)
3	2.0000	(-0.0000 0.8660 1.9365)	(0.5000 0.8660 0.0000)	(-0.2500 -0.0000 0.9682)
4	2.0000	(-0.0000 0.8660 1.9365)	(-0.5000 0.8660 0.0000)	(0.2500 -0.0000 0.9682)
5	2.0000	(-0.7500 -0.4330 1.9365)	(-1.0000 0.0000 0.0000)	(0.1250 -0.2165 0.9682)
6	2.0000	(-0.7500 -0.4330 1.9365)	(-0.5000 -0.8660 0.0000)	(-0.1250 0.2165 0.9682)

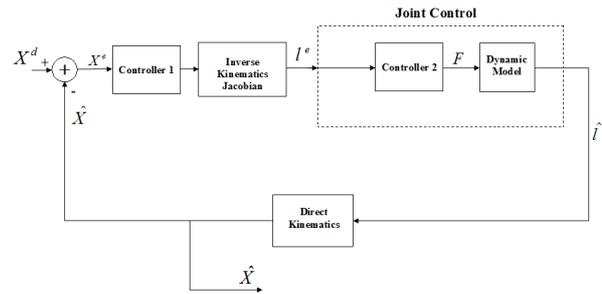


Figure 10: Control architecture

REFERENCES

- Dietmaier, P. (1998). The stewart-gough platform of general geometry can have 40 real postures. *Advances in Robot Kinematics: Analysis and Control*, pages 1–10.
- Hopkins, Brian R., W. I. R. L. (2002). Modified 6-psu platform. *Industrial Robot: An International Journal*, 29(5):443–451.
- Jakobović, D. and Budin, L. (2002). Forward kinematics of a stewart parallel mechanism. pages 149–154. Opatija, May 26-28.
- Khalil, W. and S., G. (2004). Inverse and direct dynamic modeling of gough-stewart robots. *IEEE Trans. on Robotics*, 20(4):754–762.
- Merlet, J. (2000). *Parallel Robots*. Kluwer Academic.
- Parikh, P. and Lam, S. (2005). A hybrid strategy to solve the forward kinematics problem in parallel manipulators. *IEEE Transactions on Robotics*, 21(1):18–25.