Implementation of a Generic Constraint Function to Solve the Direct Kinematics of Parallel Manipulators Using Newton-Raphson Approach

Lisandro J. Puglisi, Roque J. Saltaren Pazmiño, Cecilia E. Garcia Cena, * Pedro F. Cárdenas, ** Hector A. Moreno ***

* *Centro de Automática y Robótica, UPM-CSIC, Madrid, 28006, Spain (e-mail: lisandro.puglisi@alumnos.upm.es, rsaltaren@etsii.upm.es, cecilia.garcia@upm.es).*
** *Universidad Nacional de Colombia (e-mail: pfcardenash@unal.edu.co).*
*** *Universidad Autónoma de Coahuila (e-mail: h_moreno@uadec.edu.mx.*

AbstractNewton-Raphson (NR) methods have been implemented to find the solution of the Direct Kinematics (DK) problem of Parallel Mechanism (PM) for a long time. However, all the objective functions presented so far are topology-dependent and can not be used for every PM. In this work this topic is addressed by introducing a generic constraint function that can be adapted effortlessly to other PMs.
In order to demonstrate this capability the formulation is implemented for the most known PMs: the planar 3-$\underline{\text{R}}$RR, the spherical 3-$\underline{\text{R}}$RR, the Delta robot, and the Stewart-Gough manipulator. The rate of convergence, the accuracy and the velocity of the numerical method are analysed. Results show that the implementation of this generic constraint function within the NR algorithm provides a robust and accurate solution for the DK for suitable initial estimation. It is also shown that the simplicity of this constraint function may lead to a generic formulation for the DK of PMs.

Keywords: direct kinematic, parallel mechanism, distance constraint, Newton-Raphson

1. INTRODUCTION

To find the solution of the Direct Kinematic (DK) problem of a Parallel Mechanism (PM) is rather complex principally due to the highly non-linear and coupled relations between its elements. There are two main approaches implemented to tackle this problem and they can be classified into analytical solutions and based on numerical methods.

An analytical solution defines the geometric expressions that establish the relation between the end-effector coordinates with the joint coordinates. In general, these expressions are complex polynomial equations that rarely provide an unique solution (see Tsai (1999) and Merlet (2006)). For instance, the solution of the DK of a planar mechanism may have up to 6 different poses as stated in Merlet (1996) and Pennock and Kassner (1993). The spherical 3$\underline{\text{R}}$RR has four different assembly modes (Bonev et al. (2006)), and the special offset-3U$\underline{\text{P}}$U translational PM presented in Ji and Wu (2003) has 16 possible solutions. If the PM has more degrees of freedom (DoF) the problem is even worse: the decahedral PM described in Jin and Hai-rong (1995) has at least 48 different solutions, the 5-R$\underline{\text{P}}$UR PM (5 DoFs) has 208 real solutions (Masouleh et al. (2011)), and Stewart-Gough-type platforms (6 DoFs) has 40 possible solutions as discussed in Raghavan (1993) and Dietmaier (1998), that under certain geometrical conditions they can be reduced to 16 different poses (see Nanua et al. (1990) and Mutlu et al. (2005)).

The number of possible solutions can be reduced by collecting additional information from the state of passive joints and location of specific links, which leads to the implementation of additional sensors (see Bonev and Ryu (1999), and Baron and Angeles (2000)). Although this alternative may reduce the unknown variables other issues arise, such as establishing the optimal location and quantity of sensors that must be used.

Numerical methods appear like a strong alternative to find the solution of the DK. In general, these methods rely on a search algorithm governed by an optimization criterion. Many authors use neural network (see Boudreau et al. (1998), Li et al. (2007), Parikh and Lam (2008)); genetic algorithm (as presented in Chandra and Rolland (2011), Wang et al. (2008), Boudreau and Turkkan (1996) and Omran et al. (2009)); Newton-Raphson (NR) (Dunlop (1997), Song and Kwon (2002)); Taylor series (Sadjadian and Taghirad (2006)); fuzzy logic (Jamwal et al. (2010)) or interval analysis (Merlet (2004)) among others.

⋆ The authors would like to thank the financial support of the Spanish Government through the CICYT Project Ref. DPI2009-08778 and also to Comunidad de Madrid who supports the project ROBOCITY2030-II Ref. P2009/DPI-1559.

In particular, the NR method is widely implemented for finding the solution of the DK problem of a PM. Even though there are several variants of its implementation (some of these variants are described in Merlet (2006)), they basically consist on the search of the solution of a given set of constraint functions of the mechanism by successive approximations governed by the Jacobian matrix derived from the set of constraint functions.

As the literature reveals, there exists several alternatives to formulate this constraint function. In Almonacid et al. (2003) the DK of a $6U\underline{P}S$ PM is solved implementing a function vector with 49 elements that includes the holonomic constraints imposed by all the joints, the displacement of each actuator and the normalization of the Euler parameters. Even though this formulation presents a complete description of the kinematics of the PM, operating with the resulting Jacobian matrix is time-consuming. Another example is the scalar function derived from the vector loop-closure equation for each leg implemented in Hopkins and II (2002) for a $6\underline{P}SU$ PM. In Der-Ming (1999), the DK of an octahedral Stewart-Gough type platform is obtained with the NR method and the definition of a kinematically equivalent 3-legged mechanism. Another alternative is presented in Zubizarreta et al. (2012), where the position problem of a planar $3\underline{R}RR$, is based on a constraint vector that depends on which joint is sensorized and whether extra sensors are used.

All these functions are specifically formulated for the mechanism under analysis.

The main contribution of this article is the definition of a constraint function that could be extended to other PMs regardless of their topology. This requirement forces that the constraint function does not depend on the kinematic formulation of the PM.

In order to evaluate the convenience of the generic function proposed, it is implemented on the most popular parallel robots and its performance is analyzed in terms of accuracy, velocity and robustness to initial estimations. For this purpose a performance evaluation methodology is introduced and a set of performance indexes are defined. The latter are independent of the hardware where the simulations are executed.

This paper is organized as follows: Section 2 describes the implementation of the constraint function within the NR method. In Section 3, it is presented the customized DK model for each PM. In Section 4, the procedure used for the evaluation of its performance is described. The results of the evaluations are presented in Section 5. Finally, conclusions and discussions are stated in Section 6.

2. DEFINITION AND IMPLEMENTATION OF THE CONSTRAINT FUNCTION

2.1 Direct Kinematics Approach

A mechanism with n holonomic kinematic constraints can be expressed as Haugh (1989):

$$\boldsymbol{\Phi}(\boldsymbol{q}) = \begin{bmatrix} \phi_1(\boldsymbol{q}) \\ \vdots \\ \phi_n(\boldsymbol{q}) \end{bmatrix} = \boldsymbol{0}, \tag{1}$$

where $\boldsymbol{q}$ is the generalized coordinate vector of the mechanism.

It is important to remark that the definition of the kinematic constraints $\phi_i(\boldsymbol{q})$ that compose the constraint vector $\boldsymbol{\Phi}(\boldsymbol{q})$ are not unique and they will define the size of the Jacobian matrix. In this work a generic distance constraint function is used for each leg of the PM.

The NR method with the constraint function proposed proceeds as follows (see Fig. 1):

(1) The constraint vector of the mechanism is calculated.
(2) If the constraint vector satisfies the error condition (i.e. $\|\boldsymbol{\Phi}(\boldsymbol{q}_k)\| < \epsilon$), then the configuration of the mechanism is given by $\boldsymbol{q}_k$. If not, the method proceeds with the iterative calculation. It must be recalled that the definition of ϵ is associated with the dimensions of the mechanism. In this work it is assumed that $\epsilon = 10^{-6} mm$ for all PMs.
(3) The Jacobian matrix $\boldsymbol{\Phi}_{\boldsymbol{q}}(\boldsymbol{q}_k)$ is calculated.
(4) A new estimation for the configuration of the mechanism $\boldsymbol{q}_{k+1}$ is approximated implementing the Newton-Raphson method with the Jacobian matrix $\boldsymbol{\Phi}_{\boldsymbol{q}}(\boldsymbol{q}_k)$.
(5) The new estimation is employed to calculate the constraint vector in the following iteration.

In order to avoid infinite loops, a forced exit is generated if the NR method finds no solution after 100 iterations.

It must be noticed that after each iteration it is forced that $\|\boldsymbol{p}\| = 1$ by performing $\boldsymbol{p} = \boldsymbol{p}/\|\boldsymbol{p}\|$ accordingly with the quaternion representation for rotation.

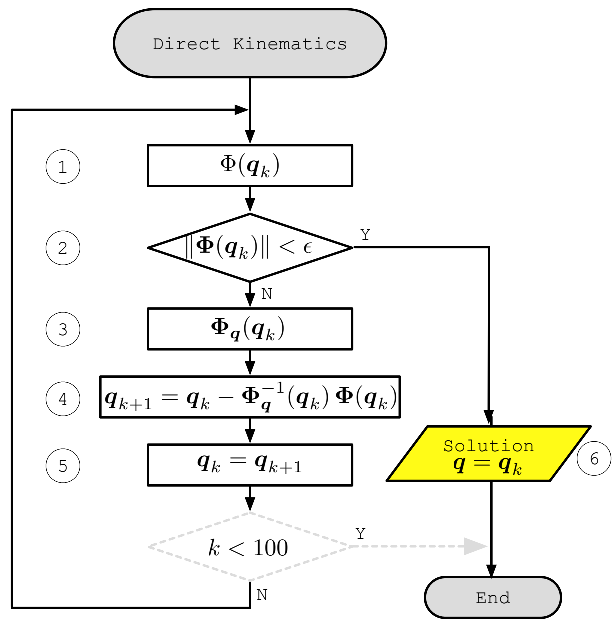


Figure 1. Direct Kinematics Algorithm.

2.2 Constraint vector definition

Taking into consideration a generic PM of with n legs composed by two links as presented in Fig. 2. The location and orientation of the moving frame P_{uvw} (attached to the moving platform) referred to the fixed frame O_{xyz} can be given by the generalized coordinate vector: $\boldsymbol{q} = [\boldsymbol{r}, \boldsymbol{p}]$, where $\boldsymbol{r} = [x, y, z]^T$ defines its position and

$\boldsymbol{p} = [e_0, e_1, e_2, e_3]^T$ defines its orientation in terms of the Euler parameters (i.e. quaternion representation for rotation). Euler parameters are used in this approach since they not only provide a compact and non-singular representation for orientation, but also they provide useful properties and identities for the manipulation of rotations.

Independently of the topology of the PM, the following vectorial equation must be met for the n legs of the mechanism:

$$\overrightarrow{OA_i} + \overrightarrow{A_iB_i} + \overrightarrow{B_iC_i} = \overrightarrow{O_iP_i} + \overrightarrow{P_iC_i}. \qquad (2)$$

Without loosing generality, it is assumed that the length of $\overrightarrow{B_iC_i}$ remains invariant for all the configurations of the mechanism, i.e. $\|\overrightarrow{B_iC_i}\| = l_{0i}$. Therefore, for all the postures of the mechanism (i.e. $\forall\, \boldsymbol{q}$) a distance constraint function for the ith leg can be defined as follows:

$$\phi_i(\boldsymbol{q}) = \|\boldsymbol{l}_i(\boldsymbol{q})\| - l_{0i} = 0, \qquad (3)$$

where $\boldsymbol{l}_i(\boldsymbol{q}) = \overrightarrow{OA_i} + \overrightarrow{A_iB_i} - \overrightarrow{O_iP_i} - \overrightarrow{P_iC_i}$. The latter one, can be expressed as:

$$\boldsymbol{l}_i(\boldsymbol{q}) = {}^O\boldsymbol{a}_i + \boldsymbol{b}_i - \left({}^O\boldsymbol{r} + {}^O\boldsymbol{R}_P(\boldsymbol{p})\,{}^P\boldsymbol{c}_i\right), \qquad (4)$$

where ${}^O\boldsymbol{R}_P(\boldsymbol{p})$ is the orientation of the end effector expressed as a rotation matrix (hereafter ${}^O\boldsymbol{R}_P$ for simplification).

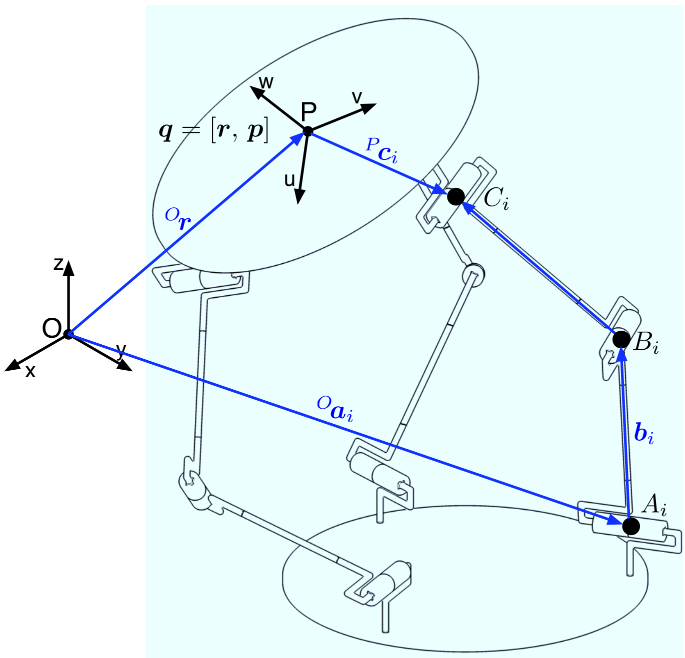


Figure 2. Schematic diagram of a two-links leg of a generic PM.

Considering the constraint distance function given by (3) for the n legs and arranging them into a single vector, the constraint vector (1) for a generic PM can be stated as,

$$\boldsymbol{\Phi}(\boldsymbol{q}) = \begin{bmatrix} \|\boldsymbol{l}_1(\boldsymbol{q})\| - l_{01} \\ \vdots \\ \|\boldsymbol{l}_n(\boldsymbol{q})\| - l_{0n} \end{bmatrix} = \boldsymbol{0}. \qquad (5)$$

2.3 Jacobian matrix of the constraint vector

Let us remember that the generalized coordinate vector of the end effector was defined as $\boldsymbol{q} = [\boldsymbol{r}, \boldsymbol{p}]$. Therefore, the Jacobian matrix $\boldsymbol{\Phi}_{\boldsymbol{q}}(\boldsymbol{q})$ of the constraint vector $\boldsymbol{\Phi}(\boldsymbol{q})$ can be found as follows,

$$\boldsymbol{\Phi}_{\boldsymbol{q}}(\boldsymbol{r},\, \boldsymbol{p}) = \begin{bmatrix} \dfrac{\partial}{\partial \boldsymbol{r}}[\phi_1(\boldsymbol{r},\boldsymbol{p})] & \dfrac{\partial}{\partial \boldsymbol{p}}[\phi_1(\boldsymbol{r},\boldsymbol{p})] \\ \vdots & \vdots \\ \dfrac{\partial}{\partial \boldsymbol{r}}[\phi_n(\boldsymbol{r},\boldsymbol{p})] & \dfrac{\partial}{\partial \boldsymbol{p}}[\phi_n(\boldsymbol{r},\boldsymbol{p})] \end{bmatrix}. \qquad (6)$$

In order to simplify the notation, it is defined $\boldsymbol{u}_i$ as follows:

$$\boldsymbol{u}_i = {}^O\boldsymbol{a}_i + \boldsymbol{b}_i - \left({}^O\boldsymbol{r} + {}^O\boldsymbol{R}_P\,{}^P\boldsymbol{c}_i\right). \qquad (7)$$

The derivative of the constraint function with respect to the position of the end effector is given by:

$$\begin{aligned} \frac{\partial}{\partial \boldsymbol{r}}\left[\phi_i(\boldsymbol{r},\, \boldsymbol{p})\right] &= \frac{\partial}{\partial \boldsymbol{r}}\left[\|\boldsymbol{l}_i(\boldsymbol{r},\, \boldsymbol{p})\| - l_{0i}\right] \\ &= \frac{\partial}{\partial \boldsymbol{r}}\left[\sqrt{\boldsymbol{u}_i^T\boldsymbol{u}_i} - l_{0i}\right] \\ &= \underbrace{\frac{1}{2\sqrt{\boldsymbol{u}_i^T\boldsymbol{u}_i}}2\boldsymbol{u}_i}_{\hat{\boldsymbol{u}}_i}\ \underbrace{\frac{\partial}{\partial \boldsymbol{r}}\left[\boldsymbol{u}_i\right]}_{\boldsymbol{I}_{3x3}} \\ &= \hat{\boldsymbol{u}}_i. \end{aligned} \qquad (8)$$

In the same way, the derivative of the constraint function with respect to the orientation of the end effector is obtained according to the following operations,

$$\begin{aligned} \frac{\partial}{\partial \boldsymbol{p}}\left[\phi_i(\boldsymbol{r},\, \boldsymbol{p})\right] &= \frac{\partial}{\partial \boldsymbol{p}}\left[\|\boldsymbol{l}_i(\boldsymbol{r},\, \boldsymbol{p})\| - l_{0i}\right] \\ &= \frac{\partial}{\partial \boldsymbol{p}}\left[\sqrt{\boldsymbol{u}_i^T\boldsymbol{u}_i} - l_{0i}\right] \\ &= \underbrace{\frac{1}{2\sqrt{\boldsymbol{u}_i^T\boldsymbol{u}_i}}2\boldsymbol{u}_i}_{\hat{\boldsymbol{u}}_i}\ \frac{\partial}{\partial \boldsymbol{p}}\left[\boldsymbol{u}_i\right] \\ &= \hat{\boldsymbol{u}}_i\left(-\frac{\partial}{\partial \boldsymbol{p}}\left[{}^O\boldsymbol{R}_P\,{}^P\boldsymbol{c}_i\right]\right) \\ &= \hat{\boldsymbol{u}}_i\left(-2\,{}^O\boldsymbol{R}_P\,{}^P\tilde{\boldsymbol{c}}_i\,\boldsymbol{G}\right), \end{aligned} \qquad (9)$$

where ${}^P\tilde{\boldsymbol{c}}_i$ is the skew antisymmetric matrix of ${}^P\boldsymbol{c}_i = [c_{ix},\, c_{iy},\, c_{iz}]^T$, given by:

$${}^P\tilde{\boldsymbol{c}}_i = \begin{bmatrix} 0 & -c_{iz} & c_{iy} \\ c_{iz} & 0 & -c_{ix} \\ -c_{iy} & c_{ix} & 0 \end{bmatrix}, \qquad (10)$$

$\boldsymbol{G} = [\boldsymbol{e}, -\tilde{\boldsymbol{e}} + e_0\boldsymbol{I}]$, $\boldsymbol{e} = [e_1, e_2, e_3]$ and $\tilde{\boldsymbol{e}}$ is the skew antisymmetric matrix of $\boldsymbol{e}$ (see Appendix A).

Then, the Jacobian matrix of the constraint vector for a generic PM can be stated as the following $n \times 7$ matrix:

$$\boldsymbol{\Phi}_{\boldsymbol{q}}(\boldsymbol{q}) = \begin{bmatrix} \hat{\boldsymbol{u}}_1^T & -2\,\hat{\boldsymbol{u}}_1^T\,{}^O\boldsymbol{R}_P\,{}^P\tilde{\boldsymbol{c}}_1\,\boldsymbol{G} \\ \vdots & \vdots \\ \hat{\boldsymbol{u}}_n^T & -2\,\hat{\boldsymbol{u}}_n^T\,{}^O\boldsymbol{R}_P\,{}^P\tilde{\boldsymbol{c}}_n\,\boldsymbol{G} \end{bmatrix}. \qquad (11)$$

As it was mentioned above, the method is governed by the inverse of the Jacobian matrix ($\boldsymbol{\Phi}_{\boldsymbol{q}}$). However, since $\boldsymbol{\Phi}_{\boldsymbol{q}}$

is not square, its inverse can not be directly calculated and the Moore-Penrose pseudo-inverse is implemented instead.

3. PARALLEL MANIPULATORS DESCRIPTION

In the following paragraphs a brief description of the PMs under evaluation is presented. It is also introduced their constraint vector and their Jacobian matrices. Even though some expressions are quite similar, they are explicitly written to highlight the adaptability of the constraint function.

3.1 3-RRR planar PM

The 3-*RRR* planar PM is composed by three identical kinematic chains, each one has two links and three rotational joints with all their axes parallel Tsai (1999).

For the purposes presented in this work, it is considered a symmetric PM whose attachment of the legs in the mobile and the fixed platform describes equilateral triangles (see Fig. 3). Therefore, the PM can be fully described by the following four parameters: the radius (R_b and R_m) of the circles that circumscribes the equilateral triangle of the base and moving platform, and the length of the first and second links (l_1 and l_2, respectively).

By simple observation of Fig. 3, it can be found that the constraint distance function is given by:

$$\phi_i(\boldsymbol{q}) = \|{}^O\boldsymbol{a}_i + \boldsymbol{R}_{\theta_i}\boldsymbol{b}_i - ({}^O\boldsymbol{r} + {}^O\boldsymbol{R}_p\,{}^P\boldsymbol{c}_i)\| - l_2, \qquad (12)$$

where $\boldsymbol{R}_{\theta_i}\boldsymbol{b}_i$ defines the location of the distal extreme of link L_{1i} (B_i) referred to A_i according to the state of the *ith*-joint (θ_i) of the PM.

The Jacobian matrix for the constraint vector of the PM is given by:

$$\boldsymbol{\Phi}_{\boldsymbol{q}}(\boldsymbol{q}) = \begin{bmatrix} \hat{\boldsymbol{u}}_1^T & -2\,\hat{\boldsymbol{u}}_1^T\,{}^O\boldsymbol{R}_p\,{}^P\tilde{\boldsymbol{c}}_1\boldsymbol{G} \\ \hat{\boldsymbol{u}}_2^T & -2\,\hat{\boldsymbol{u}}_2^T\,{}^O\boldsymbol{R}_p\,{}^P\tilde{\boldsymbol{c}}_2\boldsymbol{G} \\ \hat{\boldsymbol{u}}_3^T & -2\,\hat{\boldsymbol{u}}_3^T\,{}^O\boldsymbol{R}_p\,{}^P\tilde{\boldsymbol{c}}_3\boldsymbol{G} \end{bmatrix}, \qquad (13)$$

where $\hat{\boldsymbol{u}}_i$ is the unitary vector of $\boldsymbol{u}_i = {}^O\boldsymbol{a}_i + \boldsymbol{R}_{\theta_i}\boldsymbol{b}_i - ({}^O\boldsymbol{r} + {}^O\boldsymbol{R}_p\,{}^P\boldsymbol{c}_i)$.

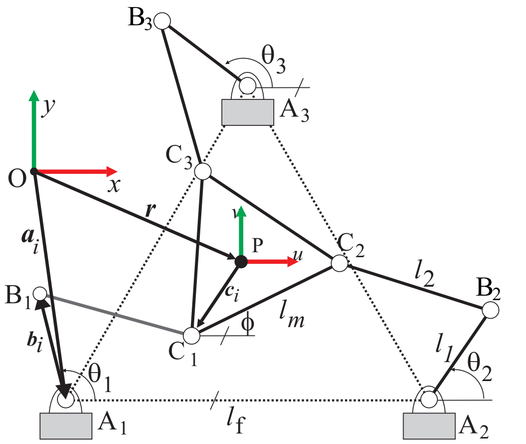


Figure 3. Schematic diagram of the Planar-3RRR PM.

3.2 Spherical 3-RRR PM

The Spherical 3-*RRR* PM (SPM) is composed by three identical legs with three rotational joints, with the particularity that all the axes of the rotational joints are intersected at an invariant common point P (see Fig. 4) which corresponds with the center of rotation of the spherical movement Gosselin and Hamel (1994), Bonev et al. (2006). The geometry of the PM can be described by the following parameters: the distance $R = \|\overline{A_i P}\|$; the elevation β of the axis of the first joints, and the angles α_1 and α_2 defined by the relative elevation of the second and third axes of the rotational joints.

Considering Fig. 4 and customizing (5) for this PM, the constraint distance function is given by (14),

$$\phi_i(\boldsymbol{q}) = \|{}^O\boldsymbol{a}_i + \boldsymbol{R}_{\theta_i}\,\boldsymbol{b}_i - ({}^O\boldsymbol{r} + {}^O\boldsymbol{R}_p\,{}^P\boldsymbol{c}_i)\| - l_2, \qquad (14)$$

and its Jacobian matrix is given by,

$$\boldsymbol{\Phi}_{\boldsymbol{q}}(\boldsymbol{q}) = \begin{bmatrix} \boldsymbol{0} & -2\,\hat{\boldsymbol{u}}_1^T\,{}^O\boldsymbol{R}_p\,\tilde{\boldsymbol{b}}_1\boldsymbol{G} \\ \boldsymbol{0} & -2\,\hat{\boldsymbol{u}}_2^T\,{}^O\boldsymbol{R}_p\,\tilde{\boldsymbol{b}}_2\boldsymbol{G} \\ \boldsymbol{0} & -2\,\hat{\boldsymbol{u}}_3^T\,{}^O\boldsymbol{R}_p\,\tilde{\boldsymbol{b}}_3\boldsymbol{G} \end{bmatrix}, \qquad (15)$$

where $\boldsymbol{u}_i = {}^O\boldsymbol{a}_i + \boldsymbol{R}_{\theta_i}\,\boldsymbol{b}_i - ({}^O\boldsymbol{r} + {}^O\boldsymbol{R}_p\,{}^P\boldsymbol{c}_i)$. It is important to remark that the first three columns of $\boldsymbol{\Phi}_{\boldsymbol{q}}(\boldsymbol{q})$ in (15) are null since the PM presents an spherical motion pattern.

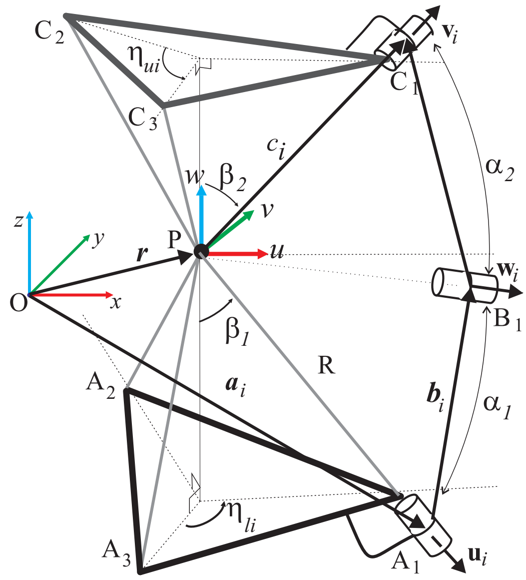


Figure 4. Schematic diagram of the Spheric-3RRR PM.

3.3 Delta PM

This PM presents a pure translational movement pattern, i.e. the relative orientation between the platforms is invariant. Each leg is composed by a four-bar parallelogram in series with a second link (see Fig.5) Clavel (1991) and Tsai and Stamper (1996).

Considering a symmetric PM, the geometry of the Delta robot can be described by: R_b, R_m, l_1 and l_2 . The constraint distance function is given by:

$$\phi_i(\boldsymbol{q}) = \|{}^O\boldsymbol{a}_i + \boldsymbol{R}_{\theta_i}\boldsymbol{b}_i - ({}^O\boldsymbol{r} + {}^O\boldsymbol{R}_p\,{}^P\boldsymbol{c}_i)\| - l_2, \qquad (16)$$

and the Jacobian matrix is given by:

$$\boldsymbol{\Phi_q}(\boldsymbol{q}) = \begin{bmatrix} \hat{\boldsymbol{u}}_1^T & \boldsymbol{0} \\ \hat{\boldsymbol{u}}_2^T & \boldsymbol{0} \\ \hat{\boldsymbol{u}}_3^T & \boldsymbol{0} \end{bmatrix}, \quad (17)$$

where $\boldsymbol{u}_i = {}^O\boldsymbol{a}_i + \boldsymbol{R}_{\theta_i}\boldsymbol{b}_i - ({}^O\boldsymbol{r} + {}^O\boldsymbol{R}_p {}^P\boldsymbol{c}_i)$. It must be noticed in (17) that the terms related to the orientation of the mechanism are null. This assumption is taken since the Delta PM is a pure translation PM.

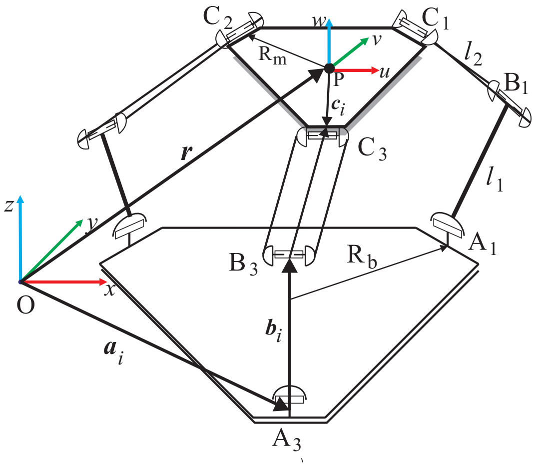


Figure 5. Schematic diagram of the Delta PM.

3.4 Stewart-Gough o 6-UPS

The Stewart-Gough PM is perhaps the most known PM Stewart (1965), Gough and Whitehall (1962). This 6 DoF mechanism posses six legs, each of them presents a universal - prismatic - spherical joint topology, being the prismatic the actuated joint.

Considering a symmetric PM, its geometry can be described by the following parameters: R_b, R_m, l_1, l_2, η_{li}, η_{ui} and η_O (see Fig. 6), and the constraint function is given by:

$$\phi_i(\boldsymbol{q}) = \|{}^O\boldsymbol{r} + {}^O\boldsymbol{R}_p {}^P\boldsymbol{c}_i - {}^O\boldsymbol{a}_i\| - (l_{2i} + l_1). \quad (18)$$

It must be noticed in (18), that l_{2i} is the state of the *ith*-prismatic actuator.

The Jacobian matrix for (18) is given by:

$$\boldsymbol{\Phi_q}(\boldsymbol{q}) = \begin{bmatrix} \hat{\boldsymbol{u}}_1^T & -2\,\hat{\boldsymbol{u}}_1^T {}^O\boldsymbol{R}_p {}^P\tilde{\boldsymbol{c}}_1\,\boldsymbol{G} \\ \vdots & \vdots \\ \hat{\boldsymbol{u}}_6^T & -2\,\hat{\boldsymbol{u}}_6^T {}^O\boldsymbol{R}_p {}^P\tilde{\boldsymbol{c}}_6\,\boldsymbol{G} \end{bmatrix}, \quad (19)$$

where $\hat{\boldsymbol{u}}_i$ is the unitary vector of $\boldsymbol{u}_i = {}^O\boldsymbol{r} + {}^O\boldsymbol{R}_p {}^P\boldsymbol{c}_i - {}^O\boldsymbol{a}_i$.

4. PERFORMANCE EVALUATION

The performance of the methodology is quantified in terms of the following key performance indexes (KPI):

(1) **Convergence**: it measures the ratio of convergence independently of the result, and it is defined as:

$$C\% = \frac{N_C}{N_{WS}} \times 100, \quad (20)$$

where N_{WS} is the number of nodes of the workspace evaluated, and N_C is the number of nodes of the workspace where the method finds a solution in less than 100 iteration.

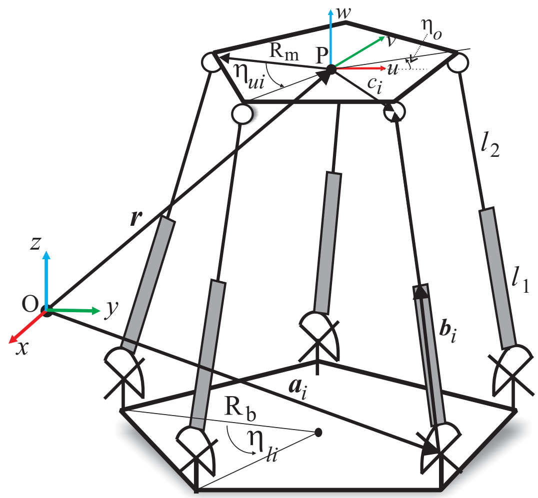


Figure 6. Schematic diagram of the 6UPS PM (Stewart-Gough).

(2) **Accuracy**: it evaluates the overall error of the method. It is defined as the ratio of the solutions that satisfy a predefined error tolerance (δ_i) and the number of nodes where it converges (N_C), in other words:

$$A_{cci}\% = \frac{N_{\delta_i}}{N_C} \times 100, \quad (21)$$

where N_{δ_i} is the number of nodes of the workspace where the method converges and satisfies one of the following error tolerance criterion:

$$\delta_1: \quad \delta_P < 1 \times 10^{-6} mm,\ \delta_O < 0.01°, \quad (22)$$

$$\delta_2: \quad \delta_P < 1 \times 10^{-3} mm,\ \delta_O < 0.1°, \quad (23)$$

where $\delta_P$and δ_O are the errors in position and orientation respectively (the orientation error is defined in the next paragraphs). During the evaluation, it is also obtained the maximum (δ_{OM} and δ_{PM}), the mean ($\overline{\delta_O}$ and $\overline{\delta_P}$) and the standard deviation (σ_{δ_O} and σ_{δ_P}) of the error in position and orientation, respectively.

(3) **Velocity**: the number of iterations needed until the method finds a solution is considered as a measure of velocity. This measure eliminates the dependability on the hardware where the simulation are carried on. The maximum (i_M), average ($\bar{i}$) and standard deviation (σ_i) of the number of iterations are taken as performance indexes.

The constraint function proposed for the DK is evaluated for all the configurations of the workspace of each PM under analysis, following the procedure depicted in Fig. 7 and detailed in the following paragraphs:

(1) For a given configuration $\boldsymbol{q} = [\boldsymbol{r}, \boldsymbol{p}]$ of the PM that belongs to its workspace, the joint states $\boldsymbol{Q}$ is obtained by means of the calculation of the inverse kinematics (IK) of the PM.
(2) An initial estimation $\boldsymbol{q}_0 = [\boldsymbol{r}_0, \boldsymbol{p}_0]$ for the end effector is randomly generated (detailed bellow).
(3) The state of the end effector $\boldsymbol{q}_D = [\boldsymbol{r}_D, \boldsymbol{p}_D]$ is obtained using the numerical method presented in section 2, considering the initial estimation generated $\boldsymbol{q}_0$ and the joint states $\boldsymbol{Q}$.
(4) If the method converges (i.e. $\exists\boldsymbol{q}_D$), the error of the solution found is calculated as follows:

$$\| \boldsymbol{r} - \boldsymbol{r}_D \| < \delta_P, \tag{24}$$

$$\| [\phi,\, \theta,\, \psi] - [\phi_D,\, \theta_D,\, \psi_D] \| < \delta_O, \tag{25}$$

where the coordinate vector $[\phi,\, \theta,\, \psi]$ is the orientation in *Roll-Pitch-Yaw* representation.

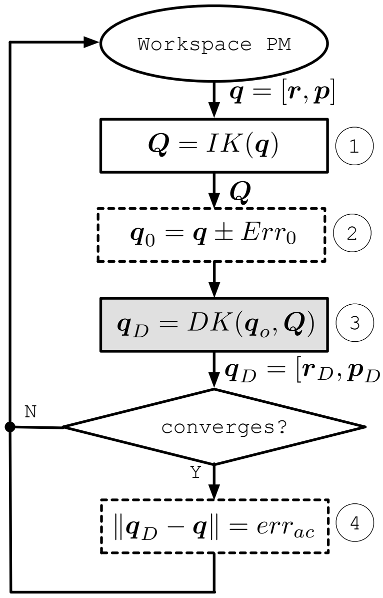


Figure 7. Evaluation Procedure (2 and 4 are symbolic expression).

The initial estimation $\boldsymbol{q}_0 = [\boldsymbol{r}_0,\, \boldsymbol{p}_0]$ is generated by considering initial distances of $L_{Err_0} = 1$, 10, 25, 50 mm and orientation of $\theta_{err_0} = 1^o$, 10^o, 25^o, 50^o, of the real value randomly placed. This approach of considering different distances from the real value should reveal the sensibility of the method towards the existence of local minimum, which is one of the main drawbacks of NR method. Despite this fact, as it was mentioned in the introductory section, NR methods are widely used because of its simplicity.

Thus, the initial estimation for the position $\boldsymbol{r} = [r_x,\, r_y,\, r_z]$ is defined as follows:

$$\boldsymbol{r}_0 = [r_x \pm L_{err_0},\, r_y \pm L_{err_0},\, r_z \pm L_{err_0}]\,, \tag{26}$$

where the sign $\pm$ is randomly selected.

The generation of the initial estimation for the orientation is more complex. The real orientation $\boldsymbol{p}$ is expressed in the angle-axis representation $(\theta,\, \boldsymbol{v})$, where $\boldsymbol{v}$ is a unit vector associated with the Euler axis of rotation and θ is the magnitude of rotation. These elements are modified with a θ_{err_0} distance as follows:

$$\theta_0 = \theta \pm \theta_{err_0}{}^o, \tag{27}$$

$$\boldsymbol{v}_0 = \boldsymbol{R}_{Err0x}\, \boldsymbol{R}_{Err0y}\, \boldsymbol{v}, \tag{28}$$

where $\boldsymbol{R}_{Err0x}$ and $\boldsymbol{R}_{Err0y}$ are pure rotations of $\pm\theta_{err_0}{}^o$ along the $\boldsymbol{x}$, $\boldsymbol{y}$ axes, respectively. The sign $\pm$ is randomly selected. Then, the new orientation $(\theta_0,\, \boldsymbol{v}_0)$ is expressed again as a quaternion (i.e: $(\theta_0,\, \boldsymbol{v}_0) \rightarrow \boldsymbol{p}_0$).

Since the home posture $\boldsymbol{q}_H$ is always a reference configuration of any manipulator, it is also considered as an initial estimation (see Table 1 for the home posture considered for each PM).

The dimensions of the PMs analysed and the exploration of their workspace are summed up in Table 1. The exploration is defined by discretized intervals for each coordinate that describes the state of the end effector (i.e., X, Y, Z, ψ, e_i), and their corresponding increment (i.e., ΔX, ΔY, ΔZ, $\Delta\psi$, Δe_i).

In order to discern whether a given configuration $\boldsymbol{q}$ belongs to the workspace of the PM it is only verified the range of work of their active joints (ρ_i). It must be stressed that it is not the objective of this work to generate an accurate workspace of the mechanism, and that the workspace is only needed to provide configurations where to test the formulation proposed to solve the DK problem. Based on this consideration and in order to reduce the simulation time (during the verification of the workspace), the collisions between the element of the PM and the constraints of the passive joints are not checked since they are not needed for the purpose of this work. In order to see how it is found the complete workspace of a mechanism see Puglisi et al. (2012) and Serracín et al. (2012).

Table 1. Parameters of the PM evaluated

PM	Parameters	Considerations
(3RRR)p	$R_m = 100$	$X \in [-300, 300], \Delta X = 5$
	$R_b = 400$	$Y \in [-300, 300], \Delta Y = 5$
	$l_1 = 250$	$\psi \in [-180^\circ, 180^\circ], \Delta\psi = 1^\circ$
	$l_2 = 250$	$-\pi \leq \rho_i \leq \pi,\ i = 1, 2, 3$
	$\boldsymbol{q}_H$	$[0,\ 0,\ 0,\ 1,\ 0,\ 0,\ 0]$
(3RRR)s	$R = 100$	$e_i \in [-1, 1]$
	$\eta_{ui} = \eta_{li} = 120$	$\Delta e_i = 0.01,$
	$\beta_1 = \beta_2 = 54.73$	$i = 1, 2, 3$
	$\alpha_1 = 90$	$-\pi/2 \leq \rho_i \leq \pi/2$
	$\alpha_2 = 90$	
	$\boldsymbol{q}_H$	$[0,\ 0,\ 0,\ 1,\ 0,\ 0,\ 0]$
Delta	$l_1 = 250$	$X \in [-300, 300], \Delta X = 2$
	$l_2 = 250$	$Y \in [-300, 300], \Delta Y = 2$
	$R_m = 150$	$Z \in [-500, 0], \Delta Z = 2$
	$R_b = 300$	$-\pi \leq \rho_i \leq \pi,\ i = 1, 2, 3.$
	$\boldsymbol{q}_H$	$[0,\ 0,\ -490,\ 1,\ 0,\ 0,\ 0]$
6UPS	$R_m = 100$	$X \in [-200, 200], \Delta X = 5$
	$R_b = 100$	$Y \in [-200, 200], \Delta Y = 5$
	$l_0 = 600$	$Z \in [600, 800], \Delta Z = 5$
	$l_2 = 250$	$e_i \in [-0.3, 0.3],\ \Delta e_i = 0.1,\ i = 1, 2, 3$
		$0.3 l_0 \leq \rho_i \leq 1.3 l_0,\ i = 1, 2, \cdots, 6.$
	$\boldsymbol{q}_H$	$[0,\ 0,\ 600,\ 1,\ 0,\ 0,\ 0]$

Note: lengths are in mm.

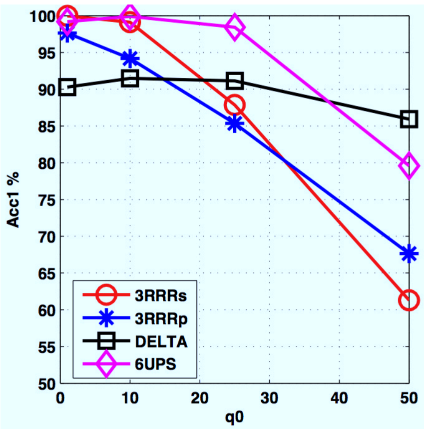
5. SIMULATION RESULTS

The results obtained during simulations are summed up in Table 2. The columns labeled as $\boldsymbol{q}_*$ presents the results for each initial estimation.

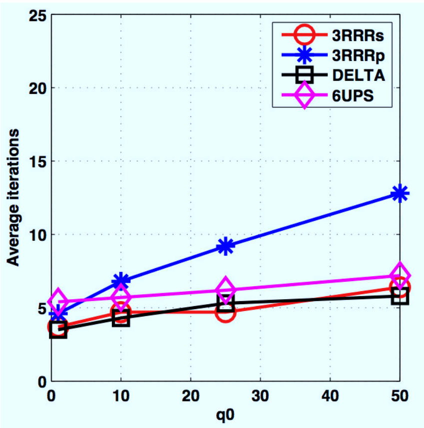
As it can be seen in Table 2, the worst rate of convergence ($C\%$) found is 84.45%, which corresponds with a considerably poor initial estimation (i.e. $\boldsymbol{q}_{50}$ for the 6UPS PM). However, for better initial estimations (i.e. $\boldsymbol{q}_1$, $\boldsymbol{q}_{10}$, $\boldsymbol{q}_{25}$), the rate of convergence is above the 98% for all the PMs.

It can be observed from the maximum errors found (i.e. δ_{OM} and δ_{PM}) that the outcome of the calculation may provide an erroneous solution. However, by examining the mean value and the standard deviation of the errors, it can be seen that it does not occur frequently, and they are highly dependent on the initial estimation (a typical characteristic of numerical methods). This is clearly shown in the results obtained for the initial estimations $\boldsymbol{q}_1$ and $\boldsymbol{q}_{10}$, where the index Acc_1 reveals that the method provides an accuracy of $1 \times 10^{-6} mm$ and $1^\circ \times 10^{-2}$ above the 90% of all the simulation for all the PM (see Fig.8(a) for clarity). Even more, if the performance index $Acc2$ is considered, it can be seen that 97% of convergences guarantee an accuracy of $1 \times 10^{-3} mm$ and $0.1°$.

From Table 2 it is also observed that in a worst case scenario a solution is found in 100 iterations ($i_M = 100$). However, the average of the iterations and their standard deviations demonstrate that this is an unlikely case. Indeed, it can be seen that the method can provide a solution in less than 10 iterations (see columns $\boldsymbol{q}_1$ and $\boldsymbol{q}_{10}$ for index $\bar{i}$ and Fig. 8(b)).



(a)



(b)

Figure 8. Performance evaluation results. (a): percentage of convergence for an accuracy given by Acc1.(b): average of iteration needed to find an accurate solution.

6. CONCLUSIONS AND DISCUSSIONS

The constraint functions used so far within the NR method to solve the DK problem of PM are mainly derived from kinematics constraints which make them specific for each PM and almost impossible to adapt to other PMs.

In this work, it was defined a generic distance constraint function that can be easily adapted to different PM since it is independent of the topology of the mechanism.

The methodology proposed was implemented in four PM with different topologies and different workspaces. The small differences between the distance function and the Jacobian matrices for each PM remark its adaptability. This characteristic is highly desired providing a step further into the definition of a generic method.

Results demonstrate the highly dependence of the method with the initial estimation, which is typical in all numerical methods. It has been proved that the distance constraint vector proposed within the NR method provides a fast, robust and accurate solution for the DK of PM if suitable initial estimation are used. In practice this is the most common scenario, since in a real control application the DK is calculated in every control loop, where the initial estimation is the result of the calculation in the previous control loop. Given the results found (in particular the average number of iteration needed to find the solution) and its simplicity, the formulation is ideal to be implemented in embedded system with reduced capabilities.

As a future work, the convergence of the method will be analysed implementing Kantorovitch's theorem and the performance of the methodology proposed will be compared with other approaches commonly used to solve the DK problem of PMs such as the *Interval Analysis* and *neural networks*, in order to provide a global view of the method.

Appendix A. VIRTUAL ROTATIONS AND EULER PARAMETERS

Let us suppose a point S attached to a moving frame $O'_{x'y'z'}$ whose origin is coincident with the origin of a fixed frame O_{xyz}. The orientation of the moving frame expressed in the fixed frame is given by the rotation matrix $\boldsymbol{R}$. The location of point S expressed in the moving frame is given by the vector $\boldsymbol{s'}$, and it can be expressed on the fixed frame as follows:

$$\boldsymbol{s} = \boldsymbol{R}\,\boldsymbol{s'} \tag{A.1}$$

Let us now consider that $O'_{x'y'z'}$ is slightly perturbed, which traduces into a change in the location of point S as follows:

$$\delta\boldsymbol{s} = \delta\boldsymbol{R}\,\boldsymbol{s'}, \tag{A.2}$$

the operator δ, called *infinitesimals*, may be interpreted as a partial differentials operator, which allows to define the following identities Haugh (1989).

From the orthogonal properties of a rotation matrix $\boldsymbol{R}\boldsymbol{R}^T = \boldsymbol{I}$, it is derived the definition of the *virtual rotation* of the $O'_{x'y'z'}$ frame relative to the O_{xyz} with component in the latter frame as follows:

$$\widetilde{\delta\boldsymbol{\pi}} = \delta\boldsymbol{R}\,\boldsymbol{R}^T. \tag{A.3}$$

After some algebraic manipulation, it can be shown that:

$$\delta\boldsymbol{R} = \boldsymbol{R}\,\widetilde{\delta\boldsymbol{\pi}'}, \tag{A.4}$$

where $\delta\boldsymbol{\pi}'$ is the virtual rotation expressed in $O'_{x'y'z'}$.

Therefore the changes on the generalized coordinates (i.e. $\delta\boldsymbol{s}$) can be expressed in terms of the virtual displacements as follows:

$$\delta\boldsymbol{s} = \boldsymbol{R}\,\widetilde{\delta\boldsymbol{\pi}'}\,\boldsymbol{s'}, \tag{A.5}$$

by properties of skew matrices:

$$\delta\boldsymbol{s} = -\boldsymbol{R}\,\widetilde{\boldsymbol{s'}}\,\delta\boldsymbol{\pi}', \tag{A.6}$$

On the other hand, it can be proved that $\boldsymbol{R} = \boldsymbol{E}\boldsymbol{G}^T$ and $\delta\boldsymbol{R} = 2\boldsymbol{E}\,\delta\boldsymbol{G}^T$, where $\boldsymbol{E} = [\boldsymbol{e}, \tilde{\boldsymbol{e}} + e_0\boldsymbol{I}]$ and $\boldsymbol{G} = [\boldsymbol{e}, -\tilde{\boldsymbol{e}} + e_0\boldsymbol{I}]$, are identities matrices, where $\boldsymbol{e} = [e_1, e_2, e_3]$.

Table 2. Performance evaluation of the method.

PM	index	$\boldsymbol{q}_H$	$\boldsymbol{q}_1$	$\boldsymbol{q}_{10}$	$\boldsymbol{q}_{25}$	$\boldsymbol{q}_{50}$
3$\underline{\text{R}}$RR(p) (819569*)	$C\%$	86.74	99.99	99.78	98.59	91.72
	i_M	100	93	100	100	100
	$\bar{i} \pm \sigma_i$	11.5 ± 6.8	4.6 ± 0.8	6.8 ± 2.6	9.2 ± 5.0	12.8 ± 7.7
	δ_{OM}	180	4.5	57.9	142.8	179.8
	$\overline{\delta_O} \pm \sigma_{\delta_O}$	16.7 ± 35	$(1 \pm 15)10^{-2}$	0.5 ± 2.4	2.9 ± 9	10.6 ± 22
	δ_{PM}	341.6	7.3	79.8	330.8	329.7
	$\overline{\delta_P} \pm \sigma_{\delta P}$	14.2 ± 38	$(1.1 \pm 18)10^{-2}$	0.5 ± 3	3.2 ± 11	11.2 ± 27
	$Acc_1\%$	61.18	97.64	94.18	85.36	67.63
	$Acc_2\%$	61.18	99.40	94.22	85.36	67.63
3$\underline{\text{R}}$RR(s) (1351169*)	$C\%$	100	100	100	100	100
	i_M	21	5	17	17	32
	$\bar{i} \pm \sigma_i$	6.2 ± 1.2	3.7 ± 0.4	4.7 ± 0.6	4.7 ± 0.6	6.4 ± 1.4
	δ_{OM}	313.6	$9.2\,10^{-6}$	293.28	324.5	332.5
	$\overline{\delta_O} \pm \sigma_{\delta_O}$	15.5 ± 53	$(1 \pm 2.8)10^{-7}$	1.5 ± 18	20.4 ± 60	67.2 ± 89
	$Acc_1\%$	90.85	100	99.13	87.86	61.29
	$Acc_2\%$	90.85	100	99.13	87.86	61.29
DELTA (9696131*)	$C\%$	100	100	100	100	100
	i_M	13	16	17	17	17
	$\bar{i} \pm \sigma_i$	2.0 ± 1.2	3.5 ± 0.6	4.3 ± 0.9	5.3 ± 1	5.8 ± 1.2
	δ_{PM}	3.98	6.61	36.96	88.2	17
	$\overline{\delta_P} \pm \sigma_{\delta P}$	0.01 ± 0.1	0.03 ± 0.3	0.35 ± 2.4	1.6 ± 8.1	1.2 ± 5.8
	$Acc_1\%$	89.52	90.29	91.48	91.14	85.94
	$Acc_2\%$	98.74	98.70	97.41	94.83	90.14
6U$\underline{\text{P}}$S (5838767*)	$C\%$	100	99.98	99.93	98.89	84.45
	i_M	8	32	40	100	100
	$\bar{i} \pm \sigma_i$	6.2 ± 0.4	5.4 ± 0.7	5.7 ± 0.7	6.2 ± 0.8	7.2 ± 2.7
	δ_{OM}	$1.1\,10^{-6}$	130.7	137.7	276.5	283.4
	$\overline{\delta_O} \pm \sigma_{\delta_O}$	$(2.0 \pm 5)10^{-8}$	0.006 ± 0.8	0.01 ± 1.2	0.23 ± 5	6.6 ± 29
	δ_{PM}	$6.55\,10^{-6}$	125	1390	1444	1520
	$\overline{\delta_P} \pm \sigma_{\delta P}$	$(6.2 \pm 15)10^{-7}$	0.006 ± 1.2	0.01 ± 3.1	0.8 ± 28	15.8 ± 116
	$Acc_1\%$	99.62	99.20	99.92	98.44	79.58
	$Acc_2\%$	100	99.97	99.67	98.66	79.76

* Nodes of the workspace evaluated.

Therefore,

$$\widetilde{\delta\boldsymbol{\pi}'} = \boldsymbol{R}^T\,\delta\boldsymbol{R}, \tag{A.7}$$

$$= \boldsymbol{E}\,\boldsymbol{G}^T\,2\boldsymbol{E}\,\delta\boldsymbol{G}^T, \tag{A.8}$$

$$= 2\boldsymbol{E}\,\boldsymbol{E}^T\,\boldsymbol{G}\,\delta\boldsymbol{G}^T, \tag{A.9}$$

$$= 2\,\boldsymbol{G}\,\delta\boldsymbol{G}^T. \tag{A.10}$$

Considering the definition of $\boldsymbol{G}$ and $\boldsymbol{p}$ it can also be demonstrated that $\boldsymbol{G}\,\delta\boldsymbol{G}^T = \widetilde{\boldsymbol{G}\,\delta\boldsymbol{p}}$, and thus $\delta\boldsymbol{\pi}' = 2\boldsymbol{G}\,\delta\boldsymbol{p}$.

Taking this last result into (A.6), the relation between the changes of the generalize coordinate of vector $\boldsymbol{s}$ due to small perturbations of the euler parameters can be stated as follows:

$$\delta\boldsymbol{s} = -2\boldsymbol{R}\,\widetilde{\boldsymbol{s}'}\boldsymbol{G}\,\delta\boldsymbol{p}. \tag{A.11}$$

REFERENCES

Almonacid, M., Saltaren, R.J., Aracil Santonja, R., and Reinoso, O. (2003). *IEEE Transactions on Robotics and Automation*, 19(3), 485–489. doi: 10.1109/TRA.2003.810238.

Baron, L. and Angeles, J. (2000). *IEEE Transactions on Robotics and Automation*, 16(1), 12–19. doi: 10.1109/70.833183.

Bonev, I., Chablat, D., and Wenger, P. (2006). Working and assembly modes of the agile eye. *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, (May), 2317–2322. doi: 10.1109/ROBOT.2006.1642048.

Bonev, I. and Ryu, J. (1999). A simple new closed-form solution of the direct kinematics using three linear extra sensors. In IEEE (ed.), *IEEE/ASME Int. Conf. on advanced intelligent mechanisms*, volume 1, 526–530.

Boudreau, R., Darenfed, S., and Gosselin, C. (1998). On the computation of the direct kinematics of parallel manipulators using polynomial networks. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, 28(2), 213–220. doi: 10.1109/3468.661148.

Boudreau, R. and Turkkan, N. (1996). Solving the forward kinematics of parallel manipulators with a genetic algorithm. *Journal of Robotic Systems*, 13(2), 111–125. doi:10.1002/(SICI)1097-4563(199602)13:2<111::AID-ROB4>3.0.CO;2-W.

Chandra, R. and Rolland, L. (2011). On solving the forward kinematics of 3RPR planar parallel manipulator using hybrid metaheuristics. *Applied Mathematics and Computation*, 217(22), 8997–9008. doi: 10.1016/j.amc.2011.03.106.

Clavel, R. (1991). *Conception d'un robot parallele rapide a 4 degres de liberte*. Ph.D. thesis, Ecole Polytechnique Federal de Lausanne.

Der-Ming, K. (1999). Direct displacement analysis of a stewart platform mechanism. *Mechanism and Machine Theory*, 34 (3), 453–465. doi:10.1016/S0094-114X(98)00043-3.

Dietmaier, P. (1998). The stewart-gough platform of general geometry can have 40 real postures. In *Advances in Robots Kinematics: Analysis and Control*, 7–16. Kluwer Academic Publishers.

Dunlop, G. (1997). Position analysis of a 3-DOF parallel manipulator. *Mechanism and Machine Theory*, 32(8), 903–920. doi:10.1016/S0094-114X(97)00011-6.

Gosselin, C.M. and Hamel, J. (1994). The agile eye: a high-performance three-degree-of-freedom camera-orienting device. In *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, 781–786. IEEE Comput. Soc. Press. doi:10.1109/ROBOT.1994.351393.

Gough, V.E. and Whitehall, S.G. (1962). Universal tyre test machine. *Proc. 9th International Technical Congress, F.I.S.I.T.A.*, 177.

Haugh, E. (1989). *Computer aided kinematics and dynamics of mechanical systems.* Allyn and Bacon.

Hopkins, B.R. and II, R.L.W. (2002). Kinematics, design and control of the 6-PSU platform. *Industrial Robot: An International Journal*, 29(5), 443–451. doi: 10.1108/01439910210440264.

Jamwal, P., Xie, S., Tsoi, Y., and Aw, K. (2010). Forward kinematics modelling of a parallel ankle rehabilitation robot using modified fuzzy inference. *Mechanism and Machine Theory*, 45, 1537–1554. doi: 10.1016/j.mechmachtheory.2010.06.017.

Ji, P. and Wu, H. (2003). Kinematics analysis of an offset 3-UPU translational parallel robotic manipulator. *Robotics and Autonomous Systems*, 42(2), 117–123. doi: 10.1016/S0921-8890(02)00324-X.

Jin, Y. and Hai-rong, F. (1995). Forward displacement analysis of the decahedral variable geometry truss manipulator. *Robotics and Autonomous Systems*, 15(3), 173–178. doi:10.1016/0921-8890(95)00025-B.

Li, L., Zhu, Q., and Xu, L. (2007). Solution for forward kinematics of 6-dof parallel robot based on particle swarm optimization. In *Proceedings of the 2007 IEEE International Conference on Mechatronics and Automation*, 2968–2973.

Masouleh, M.T., Gosselin, C.M., Husty, M., and Walter, D.R. (2011). Forward kinematic problem of 5-RPUR parallel mechanisms (3T2R) with identical limb structures. *Mechanism and Machine Theory*, 46(7), 945–959. doi:10.1016/j.mechmachtheory.2011.02.005.

Merlet, J.P. (2004). Solving the Forward Kinematics of a Gough-Type Parallel Manipulator with Interval Analysis. *The International Journal of Robotics Research*, 23(3), 221–235. doi:10.1177/0278364904039806.

Merlet, J.P. (1996). Direct kinematics of planar parallel manipulators. In *Proceedings of IEEE International Conference on Robotics and Automation*, volume 4, 3744–3749. IEEE. doi:10.1109/ROBOT.1996.509284.

Merlet, J. (2006). *Parallel Robots. 2nd.ed.*, volume 1. Springer.

Mutlu, H., Akçali, .D., and Gülsen, M. (2005). A Mathematical Model for the Use of a Gough-Stewart Platform Mechanism as a Fixator. *Journal of Engineering Mathematics*, 54(2), 119–143. doi:10.1007/s10665-005-9007-0.

Nanua, P., Waldron, K., and Murthy, V. (1990). Direct kinematic solution of a Stewart platform. *IEEE Transactions on Robotics and Automation*, 6(4), 438–444. doi: 10.1109/70.59354.

Omran, A., Bayoumi, M., Kassem, A., and El-Bayoumi, G. (2009). Optimal Forward Kinematics Modeling of Stewart Manipulator Using Genetic Algorithms. *Jordan Journal of Mechanical and Industrial Engineering*, 3(4), 280–293.

Parikh, P.J. and Lam, S.S. (2008). Solving the forward kinematics problem in parallel manipulators using an iterative artificial neural network strategy. *The International Journal of Advanced Manufacturing Technology*, 40(5-6), 595–606. doi:10.1007/s00170-007-1360-x.

Pennock, G.R. and Kassner, D.J. (1993). The workspace of a general geometry planar three-degree-of-freedom platform-type manipulator. *Journal of Mechanical Design*, 115(2), 269–276. doi:10.1115/1.2919187.

Puglisi, L.J., Saltaren, R.J., Moreno, H.A., Cárdenas, P.F., Garcia, C., and Aracil, R. (2012). Dimensional synthesis of a spherical parallel manipulator based on the evaluation of global performance indexes. *Robotics and Autonomous Systems*, 60(8), 1037–1045. doi: 10.1016/j.robot.2012.05.013.

Raghavan, M. (1993). The stewart platform of general geometry has 40 configurations. *Jounal of Mechanical Design*, 115(2), 277–282. doi:10.1115/1.2919188.

Sadjadian, H. and Taghirad, H. (2006). Comparison of different methods for computing the forward kinematics of a redundant parallel manipulator. *Journal of Intelligent and Robotic Systems*, 10.1007/s10846–005–9006–4.

Serracín, J., Puglisi, L., Saltaren, R., Ejarque, G., Sabater-Navarro, J., and Aracil, R. (2012). Kinematic analysis of a novel 2-d.o.f. orientation device. *Robotics and Autonomous Systems*, 60(6), 852–861. doi: 10.1016/j.robot.2012.01.010.

Song, S.K. and Kwon, D.S. (2002). New Direct Kinematic Formulation of 6DOF Sewart-Gough Platform Using the Thetrahedron Approach. *Transaction of Control, Automation, and Systems Engineering*, 4(3), 217–223.

Stewart, D. (1965). A platform with six dregrees of freedom. *Proc. Inst. Mech. Eng.*, 148, 371–386.

Tsai, L. (1999). *Robot Analysis, The Mechanics of Serial and Parallel Manipulators.*

Tsai, L.W. and Stamper, R. (1996). A paralle manipulator with only traslational degrees of freedom. In *96-DETC-MECH-1152, ASME 1996 Design Engineering Technical conference.* Irvine,CA.

Wang, X.S., Hao, M.L., and Cheng, Y.H. (2008). On the use of differential evolution for forward kinematics of parallel manipulators. *Applied Mathematics and Computation*, 205(2), 760–769. doi: 10.1016/j.amc.2008.05.065.

Zubizarreta, A., Marcos, M., Cabanes, I., Pinto, C., and Portillo, E. (2012). Redundant sensor based control of the 3RRR parallel robot. *Mechanism and Machine Theory*, 54, 1–17. doi: 10.1016/j.mechmachtheory.2012.03.004.