Robotica (2009) volume 27, pp. 1017–1026. © Cambridge University Press 2009
doi:10.1017/S0263574709005426

A new method to solve robot inverse kinematics using Assur virtual chains

H. Simas†,‡,*, R. Guenther†,§, D. F. M. da Cruz† and D. Martins†

†*Departamento de Engenharia Mecânica, Laboratório de Robótica, UFSC Florianópolis, SC, Brazil*
‡*CTTMAR – Centro de Ciências Tecnológicas da Terra e do Mar, UNIVALI - São José, SC, Brazil*
§*in memoriam*

(Received in Final Form: January 27, 2009. First published online: March 6, 2009)

SUMMARY
This paper describes a numerical algorithm to solve the inverse kinematics of parallel robots based on numerical integration. Inverse kinematics algorithms based on numerical integration involve the drift phenomena of the solution; as a consequence, errors are generated when the end-effector location differs from that desired. The proposed algorithm associates a novel method to describe the differential kinematics with a simple numerical integration method. The methodology is presented in this paper and its exponential stability is proved. A numerical example and a real application are presented to outline its advantages.

KEYWORDS: Differential kinematics model; Error control; Closed kinematics chains; Numerical stability.

1. Introduction

A kinematic chain is a system of bodies composed of an assemblage of links connected by joints. When every link in a kinematic chain is connected to at least two other links, the kinematic chain forms one or more closed loops and it is called a closed-loop chain or, in short, a closed chain. On the other hand, if every link is connected to every other link by one and only one path, the kinematic chain is called an open-loop chain or an open chain. It is also possible for a kinematic chain to be made up of both closed and open chains. We call such a chain a hybrid kinematic chain.[8]

A robot is said to be a serial robot or open-loop manipulator if its kinematic structure takes the form of an open-loop chain. It is a parallel manipulator if it is made up of a closed-loop chain and a hybrid manipulator if it consists of both open- and closed-loop chains.

A robotic system typically consists of a mechanical manipulator, an end-effector, a microprocessor-based controller and a computer. A mechanical manipulator comprises several links connected by joints constituting a kinematic chain. Some of the joints in the manipulator are actuated; the others are passive. Typically, the number of actuated joints is equal to the degrees of freedom.

The concept of Assur virtual chains is described by Campos *et al.*[1] The virtual kinematic chains are used to monitor or impose movement in a kinematic chain. When the Assur virtual chains are used to impose movements in serial robot manipulators, the resultant mechanism is closed, or represents parallel robot.[1] Therefore, either for parallel, serial or hybrid manipulators, with virtual chains, the kinematic procedure will be that of a parallel manipulator. This article discusses this general representation.

In this paper, we focus on the kinematics of closed chains present in parallel manipulator. More specifically, we obtain the positions of the passive joints given the positions of the active joints by integrating the differential kinematics of the closed chain using numerical techniques. The classical numerical integration is based on the Euler method, better known as *resolved motion control*.[18,19]

It is well known that in open-loop kinematic chains the numerical integration of the differential kinematics involves drift phenomena of the solution and, as a consequence, the end-effector location corresponding to the computed joint variables differs from that desired (see Sciavicco and Siciliano,[7] for example).

In closed-loop kinematic chains, besides the drift phenomena the integration errors in the joint positions reflect in an opening of the closed chain, which could be measured by a closure error.[6] For parallel robots, this drift must be treated and minimized.

The main contribution of this paper is to introduce a new method to model the differential kinematics of parallel robots. The discussed method is applied to integrate numerically the differential kinematics of closed chains, guaranteeing the exponential convergence of the error, in this case a closure error.

To present the method, we first state the problem conceptually. Next, the differential kinematics based on the screw representation of the Davies Method is summarily reproduced for completeness. We then present the proposed algorithm and state its convergence properties. An example outlines the main algorithm features.

2. Closure Error in Closed Chain Mechanisms

In closed kinematic chains not all of the joints can be controlled independently. Thus, some of the joints are driven by actuators whereas others are passive. In this paper, the joints driven by actuators are called primary joints, and the passive ones are called secondary joints.

* Corresponding author. E-mail: hsimas@univali.br

Let the primary joint variables be denoted by a vector q_p and the secondary joint variables be described by a vector q_s. The kinematic constraints imposed by the limbs can then be written in the general form:[8]

$$f(q_p, q_s) = 0, \tag{1}$$

where f is an n-dimensional implicit function of q_p and q_s, and 0 is an n-dimensional zero vector. Equation (1) is sometimes referred to as the *closure equation*.

Differentiating Eq. (1) with respect to time, we obtain a relationship between the input joint rates (primary joint velocities $\dot{q}_p$) and the output joint rates (secondary joint velocities $\dot{q}_s$) as follows:

$$N_p\dot{q}_p + N_s\dot{q}_s = 0, \tag{2}$$

where $N_p = \partial f/\partial q_p$ and $N_s = \partial f/\partial q_s$ are dependent on the position vectors q_p and q_s. From Eq. (2) we have:

$$\dot{q}_s = -N_s^{-1}N_p\dot{q}_p. \tag{3}$$

The secondary joint position can be calculated by integrating Eq. (3) as follows:

$$q_s(t) - q_s(0) = \int_0^t \dot{q}_s \, \mathrm{d}t = -\int_0^t N_s^{-1}N_p\dot{q}_p \, \mathrm{d}t. \tag{4}$$

This technique for calculating the secondary joint positions is independent of the solvability of the kinematic structure. Nonetheless, the secondary matrix N_s must be square and of full rank. When this matrix has more columns than rows there are infinite solutions of Eq. (3). In this case, a viable solution method is to formulate the problem as a constrained linear optimization problem, as is usual in redundant manipulator analysis. This approach and a method to deal with singularities can be found in Sciavicco and Siciliano.[7]

The integration can be performed in discrete time by resorting to numerical techniques. The simplest technique is based on the Euler integration method; given an integration interval Δt, if the joint positions and velocities at time t_{k-1} are known, the joint positions at time $t_k = t_{k-1+}\Delta t$ can be computed as

$$q_s(t_k) = q_s\big(t_{k-1}\big) - N_s^{-1}\big(t_{k-1}\big)N_p\big(t_{k-1}\big)\dot{q}_p(t_k)\Delta t. \tag{5}$$

By calculating the secondary joint positions using Eq. (5), a cumulative error in q_s is introduced. Therefore, Eq. (1) is not satisfied and an opening in the closed-loop chain is introduced. To illustrate this, consider the four-bar mechanism in Fig. 1.

The cumulative error in $\dot{q}_s$ opens the closed chain, as depicted in Fig. 2. To solve this problem, we present in Section 4 a new method to model the robot differential kinematic equation, where the position is obtained using numerical techniques in which the closure error converges exponentially to zero. To describe it, first we present the fundamental kinematics tools used in this study.

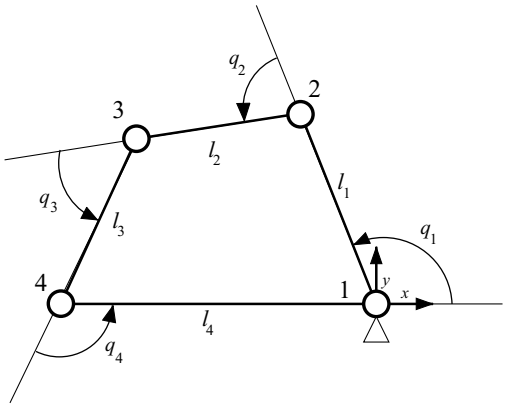

Fig. 1. Four bar planar closed chain.

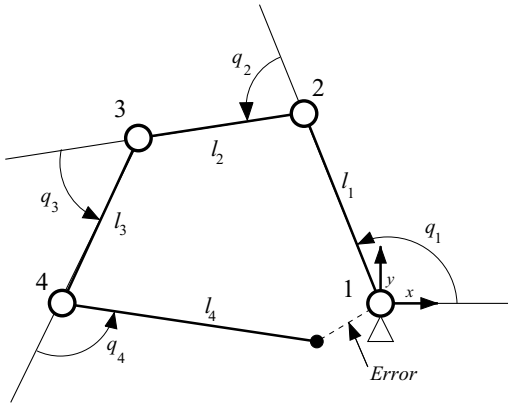

Fig. 2. Closure error in a four bar chain.

3. Fundamental Kinematics Tools

Our approach is based on the method of successive screw displacement,[8] on the screw representation of differential kinematics, on the Davies method and on the Assur virtual chain concept, which is briefly presented in this section.

3.1. Method of successive screw displacements

In this subsection, we describe a method to represent the location of a rigid body in a kinematic chain with respect to a coordinate frame, based on the successive screw displacement concept. First, we present the transformation matrix associated with a screw displacement, and then describe the concept of the resultant screw of two successive screw displacements.

3.1.1. Homogeneous transformation screw displacement representation. The Chasles theorem states that the general spatial displacements of a rigid body are a rotation about and a translation along some axis. Such a combination of translation and rotation is called a screw displacement.[10] Below, we derive a homogeneous transformation that represents a screw displacement.[8]

Figure 3 shows a point P of a rigid body, which is displaced from a first position P_1 to a second position P_2 by a rotation θ about a screw axis followed by a translation of $\boldsymbol{t}$ along the same axis. The rotation brings P from P_1 to P_2^r, and the translation brings P from P_2^r to P_2. In the figure, $s = [s_x \;\; s_y \;\; s_z]^T$ denotes a unit vector along the direction of the screw axis, and $s_o = [s_{ox} \;\; s_{oy} \;\; s_{oz}]^T$ denotes the position vector of

https://doi.org/10.1017/S0263574709005426 Published online by Cambridge University Press

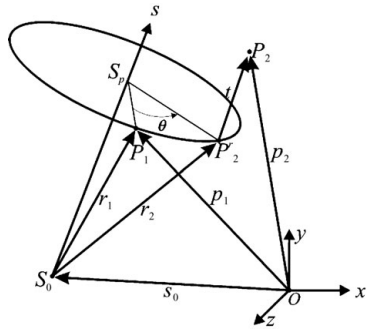

Fig. 3. Vector diagram of a spatial displacement.

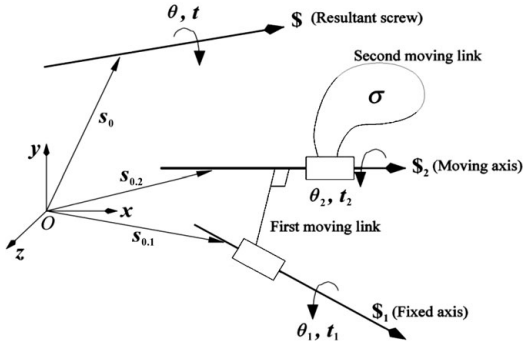

Fig. 4. Two-link chain and its associated screw displacements.

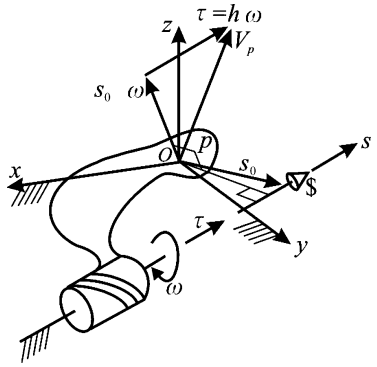

Fig. 5. Screw movement or twist.

a point lying on the screw axis. The rotation angle θ and the translation t are called the screw parameters. These screw parameters together with the screw axis completely define the general displacement of a point attached to a rigid body and, consequently, define the displacement of the rigid body itself.

Representing the first position P_1 by the vector $p_1 = [p_{1x}\ \ p_{1y}\ \ p_{1z}]^T$ and the second position P_2 by the vector $p_2 = [p_{2x}\ \ p_{2y}\ \ p_{2z}]^T$, the general screw displacement for a rigid body can be given by the Rodrigues formula as

$$p_2 = R(\theta)p_1 + d(t), \tag{6}$$

where $R(\theta)$ is the rotation matrix corresponding to the rotation θ about the screw axis and $d(t)$ is the displacement vector corresponding to the translation $\boldsymbol{t}$ along the screw axis.

Considering the augmented vectors $\hat{p}_1 = [p_1^{\mathrm{T}}\ 1]^{\mathrm{T}}$ and $\hat{p}_2 = [p_2^{\mathrm{T}}\ 1]^{\mathrm{T}}$ the general displacement of a rigid body (Eq. (6)) can be represented by a homogeneous transformation given by

$$\hat{p}_2 = A(\theta, t)\hat{p}_1, \tag{7}$$

where

$$A(\theta, t) = \begin{bmatrix} R(\theta) & d(t) \\ 0 & 1 \end{bmatrix}, \tag{8}$$

and the elements of $R(\theta)$ and of $d(t)$, according to Tsai,[8] are given by

$$R(\theta) = \begin{bmatrix} \cos\theta + s_x^2(1-\cos\theta) & s_y s_x(1-\cos\theta) - s_z\sin\theta & s_z s_x(1-\cos\theta) + s_y\sin\theta \\ s_x s_y(1-\cos\theta) + s_z\sin\theta & \cos\theta + s_y^2(1-\cos\theta) & s_z s_y(1-\cos\theta) - s_x\sin\theta \\ s_x s_z(1-\cos\theta) - s_y\sin\theta & s_y s_z(1-\cos\theta) + s_x\sin\theta & \cos\theta + s_z^2(1-\cos\theta) \end{bmatrix},$$

and

$$d(t) = ts + [I - R(\theta)]s_0.$$

3.1.2. Successive screw displacements. We now use the homogeneous transformation screw representation to express the composition of two or more screw displacements applied successively to a rigid body.[8]

Figure 4 shows a rigid body σ which is guided to a fixed base by a dyad made up of two kinematic pairs, denoted by $\$_1(q_1)$ and $\$_2(q_2)$, respectively. The first kinematic pair connects the first moving link to the base, and the second kinematic pair connects the second link (σ) to the first. We call the axis of the first kinematic pair the *fixed joint axis* and the axis of the second kinematic pair the *moving joint axis*.

As the rigid body is rotated about and/or translated along these two joint axes, the best way to obtain its resultant displacement is to displace the rigid body σ about/along the fixed axis and, subsequently, displace the body about/along the moving axis. In this way, the initial location of the moving joint axis can be used to derive the transformation matrix $A_2(q_2)$, which represents the $\$_2(q_2)$ screw displacement while the fixed joint axis is used for the derivation of matrix $A_1(q_1)$, which represents the $\$_1(q_1)$ screw displacement (see ref. [8]).

Consequently, the resulting transformation matrix is given by a premultiplication of the two successive screw displacements, i.e.

$$A_r(q_1, q_2) = A_1(q_1), A_2(q_2). \tag{9}$$

3.2. Screw representation of differential kinematics

The general spatial differential movement of a rigid body consists of a differential rotation about, and a differential rotation along an axis named the instantaneous screw axis. In this way, the velocities of the points of a rigid body with respect to an inertial reference frame *O-xyz* may be represented by a differential rotation ω about the instantaneous screw axis and a simultaneously differential translation τ about this axis. The complete movement of the rigid body, combining rotation and translation, is called screw movement or twist and is here denoted by$. Fig. 5 shows a body 'twisting' around the instantaneous screw axis. The

https://doi.org/10.1017/S0263574709005426 Published online by Cambridge University Press

ratio of the linear velocity to the angular velocity is called the pitch of the screw $h = \| \tau \| / \| \omega \|$.

The twist may be expressed by a pair of vectors, i.e. $\$ = [\omega^{\mathrm{T}}; V_p^{\mathrm{T}}]^{\mathrm{T}}$, where ω represents the angular velocity of the body with respect to the inertial frame, and V_p represents the linear velocity of a point P attached to the body which is instantaneously coincident with the origin O of the reference frame. A twist may be decomposed into its magnitude and its corresponding normalized screw. The twist magnitude, denoted as $\dot{q}$ in this study, is either the magnitude of the angular velocity of the body $\|\omega|$, if the kinematic pair is rotative or helical, or the magnitude of the linear velocity $\|V_p\|$, if the kinematic pair is prismatic. The normalized screw $\hat{\$}$ is a twist in which the magnitude is factored out, i.e.

$$\$ = \hat{\$}\dot{q}. \tag{10}$$

The normalized screw coordinates[4] may be given by

$$\hat{\$} = \begin{bmatrix} s \\ s_0 \times s + hs \end{bmatrix}, \tag{11}$$

where, as above, the vector $s = [s_x \;\; s_y \;\; s_z]^T$ denotes a unit vector along the direction of the screw axis, and the vector $s_o = [s_{ox} \;\; s_{oy} \;\; s_{oz}]^T$ denotes the position vector of a point lying on the screw axis.

Thus, the twist given in Eq. (10) expresses the general spatial differential movement (velocity) of a rigid body with respect to an inertial reference frame O-xyz. The twist can also represent the movement between two adjacent links of a kinematic chain. In this case, the twist $\$_i$ represents the movement of link i with respect to link $(i-1)$.

3.3. Davies' method

Davies' method is a systematic way to relate the joint velocities in closed kinematic chains. Davies[1] derives a solution to the differential kinematics of closed kinematic chains from the Kirchhoff circulation law for electrical circuits. The resulting Kirchhoff–Davies circulation law states that "The algebraic sum of relative velocities of kinematic pairs along any closed kinematic chain is zero".[5]

We use this law to obtain the relationship between the velocities of a closed kinematic chain.[1] Thus, considering that the velocity of a link with respect to itself is null, the circulation law can be expressed as

$$\sum_{i=1}^{n} \$_i = 0, \tag{12}$$

where 0 is a vector the dimension of which corresponds to the dimension of the twist $\$_i$.

According to the normalized screw definition introduced above, Eq. (12) may be rewritten as

$$\sum_{i=1}^{n} \hat{\$}_i \dot{q}_i = 0, \tag{13}$$

where $\hat{\$}_i$ and $\dot{q}_i$ represent the normalized screw and the magnitude of twist $\$_i$, respectively.

Equation (13) is the constraint equation which, in general can be written as

$$N\dot{q} = 0, \tag{14}$$

where $N = [\hat{\$}_1 \;\; \hat{\$}_2 \;\; \ldots \;\; \hat{\$}_n]$ is the network matrix containing the normalized screws, the signs of which are dependent on the screw definition in the circuit orientation, and $\dot{q} = [\dot{q}_1 \;\; \dot{q}_2 \;\; \ldots \;\; \dot{q}_n]$ is the magnitude vector.

A closed kinematic chain has actuated joints, here named primary joints, and passive joints, named secondary joints. The constraint equation, Eq. (14), allows the calculation of the secondary joint velocities as functions of the primary joint velocities. To this end, the constraint equation is rearranged highlighting the primary and secondary joint velocities and Eq. (14) is rewritten as follows:

$$[\, N_p \vdots N_s \,] \begin{bmatrix} \dot{q}_p \\ \cdots \\ \dot{q}_s \end{bmatrix} = 0, \tag{15}$$

where N_p and N_s are the primary and secondary network matrices, respectively, and $\dot{q}_p$ and $\dot{q}_s$ are the corresponding primary and secondary magnitude vectors, respectively.

Equation (15) can be rewritten as

$$N_p\dot{q}_p + N_s\dot{q}_s = 0, \tag{16}$$

which is Eq. (2) derived in another way.

3.4. Assur virtual chain

The Assur virtual kinematic chain concept, virtual chain for short, is essentially a tool to obtain information on the movement of a kinematic chain or to impose movements on a kinematic chain.[1]

This concept was first introduced by Campos *et al.*,[1] which defines the virtual chain as a kinematic chain composed of links (virtual links) and joints (virtual joints) and which possesses the following three properties: (a) the virtual chain is open, (b) it has joints whose normalized screws are linearly independent and (c) it does not change the mobility of the real kinematic chain, in other words, it is an Assur group.[2,20]

From the property (c) the virtual chain proposed by Campos *et al.*[1] is in fact an Assur group,[2] i.e. a kinematic subchain with null mobility that when connected to another kinematic chain preserves mobility.

Due to the analogy with Assur groups and to avoid possible confusions with other unrelated virtual chains in the literature, we coined the name Assur virtual chain.

3.4.1. The orthogonal PPR Assur virtual chain. The PPR virtual chain is composed of two virtual links (C_1, C_2) connected by two prismatic joints, whose movements are in the x and y orthogonal directions, and a rotational joint, whose the movement is in the z direction (see Fig. 6). The prismatic joints are called px and py, and the rotative joint is called rz.

The first prismatic joint (px) and the rotative joint (rz) are attached to the chain to be analysed (real chain). The joint px connects the link R_1 with the virtual link C_1, the joint py

https://doi.org/10.1017/S0263574709005426 Published online by Cambridge University Press

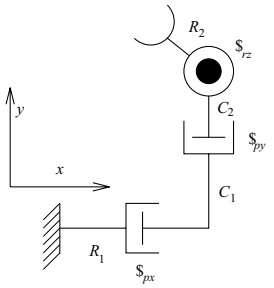

Fig. 6. PPR Assur virtual chain.

connects the virtual link C_1 with the virtual link C_2 and the joint *rz* connect the virtual link C_3 with the real link R_2.

Let the twist $\$_{px}$ represent the movement of the link C_1 in relation to the link R_1, twist $\$_{py}$ represent the movement of the link C_2 in relation to the link C_1 and twist $\$_{rz}$ represent the movement of the link R_2 in relation to the link C_2. Therefore, the movement of the link R_2 in relation to the real link R_1 may be expressed by $\$_{px} + \$_{py} + \$_{rz}$.

Consider the C-reference system (*C-system*) attached to the virtual link C_2 at the joint *rz*. Therefore, there is no rotation between the *C-system* and the *B-system* (attached to the inertial base), and the joint *rz* is aligned with the *z* direction. Thus, the normalized screws corresponding to the virtual joints represented in the *C-system* are

$$ {}^{C}\hat{\$}_{rz} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}; \; {}^{C}\hat{\$}_{px} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}; \; {}^{C}\hat{\$}_{py} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}. \qquad (17) $$

It can be observed that the orthogonal PPR Assur virtual chain represents the movements in a planar Cartesian system. Other Assur virtual chains can be found in refs. [1–3].

3.4.2. The orthogonal 3P3R Assur virtual chain. To represent the movements in the tridimensional Cartesian system the 3P3R Assur virtual chain is used. The 3P3R virtual chain is composed of three orthogonal prismatic joints (in the *x*, *y* and *z* directions), and a spherical wrist, composed of three rotative joints[1] (in the *x*, *y* and *z* directions). Fig. 7 shows the 3P3R Assur virtual chain with the virtual links C_i labelled.

3.5. The direct graph notation

Consider a kinematic pair composed of two links E_i and E_{i+1}. This kinematic pair has the relative velocity defined for a screw ${}^{R}\$_j$ (joint *j*) in relation to a reference frame *R*. The joint *j* represents the relative movement of the link E_i with respect to the link E_{i+1}. This relation can be represented as a graph,[15] as shown Fig. 8.

Where vertices represent links and arcs represent joints. The relative movement is also indicated by the arcs directions. For instance in Fig. 8, the link E_{i+1} moves with respect to link E_i via the joint *j*.

Now consider the following example, where the joint *j* is part of two closed chains. For each closed chain the circuit direction is defined.[1] Fig. 9 shows an example.

In a direct mechanism graph, if the joint has the same direction as the circuit, the twist associated with the joint has a positive sign in the circuit equation (Eq. (13)), and if the joint has the opposite direction to the circuit, the sign will be negative.

In the example, the twist ${}^{R}\$_j$, associated with the joint *j*, will have a positive sign in the *circuit a* equation and a negative sign in the *circuit b* equation.

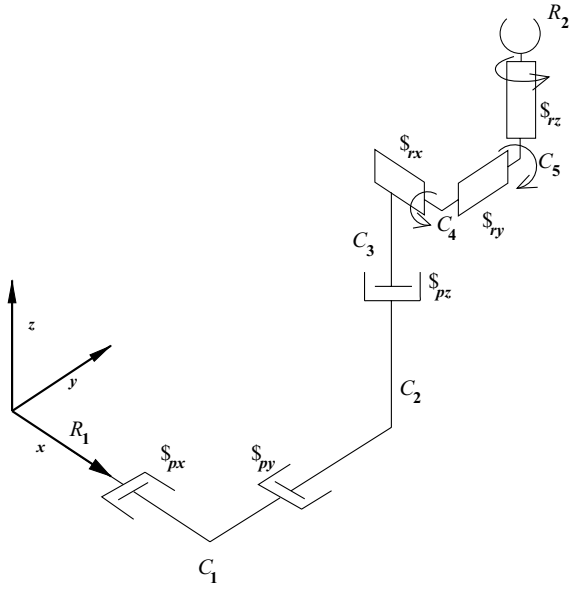

Fig. 7. 3P3R Assur virtual chain.

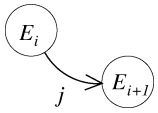

Fig. 8. Relative movement of link E_i with respect to link E_{i+1} represented graphically.

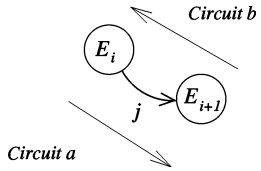

Fig. 9. Relation between joint *j* and the *circuits a* and *b*.

4. Integration Algorithm Using Assur Virtual Chains

An integration algorithm is necessary to integrate the differential kinematics equation to obtain the joint positions.[19] The algorithm proposed in this paper has two steps. The first step is to introduce a virtual chain to represent the closure error resulting from the integration error as shown in Fig. 2. For the same example of a four bar planar mechanism, the resulting closed chain is shown in Fig. 10.

The constraint equation of this closed-loop chain results in

$$ N_p\dot{q}_p + N_s\dot{q}_s + N_e\dot{q}_e = 0, \qquad (18) $$

where N_p and N_s are the primary and secondary network matrices obtained by integration, $\dot{q}_p$ and $\dot{q}_s$ are the primary and secondary magnitude vectors, respectively, N_e is the error network matrix and $\dot{q}_e$ is the error magnitude vector.

https://doi.org/10.1017/S0263574709005426 Published online by Cambridge University Press

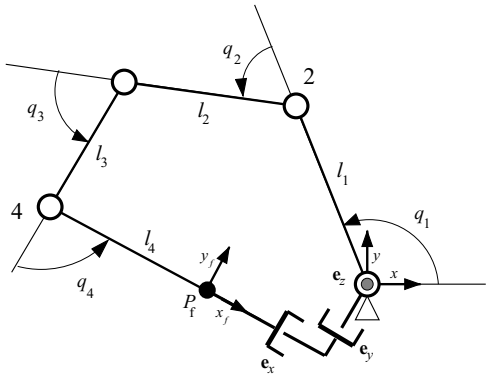

Fig. 10. Closure error represented by a virtual chain in a four bar mechanism.

The second step is to replace Eq. (3) by

$$\dot{q}_s = -N_s^{-1} N_p \dot{q}_p + N_s^{-1} N_e K_e q_e, \tag{19}$$

where the gain matrix K_e is chosen to be positive definite and q_e is the position error vector.

4.1. Stability

The algorithm stability can be verified substituting Eq. (19) into Eq. (18), obtaining

$$N_e \dot{q}_e + N_e K_e q_e = 0. \tag{20}$$

Multiplying all terms by N_e^{-1} (which always exists because the virtual chain joint normalized screws are always linearly independent), results in

$$\dot{q}_e + K_e q_e = 0. \tag{21}$$

As the gain matrix is positive definite, Eq. (21) states that the position error vector $q_e \to 0$ as $t \to \infty$ exponentially.

Theoretically, the gains can be as great as desired, however, in practice they are limited by the possibility of introducing numerical problems, such as numerical oscillations and, in some limit cases, instabilities. This gain limit can result in undesirable errors. This difficulty is overcome by using gains that do not introduce numerical problems combined with various iterations which allow control over the error, as is shown in the next subsection.

4.2. Error control

The exponential convergence is guaranteed in the general case in which the primary positions and velocities varying over time. It has the same property when the primary positions and velocities are constant, as in a second iteration. If the error is greater than the desired value, new iterations can be performed until the error is within the admissible tolerance. Consequently, the proposed algorithm allows control over the error.

4.3. Position error vector

The screw displacement of a link in a kinematic chain can be expressed by a homogeneous matrix, and the resultant screw displacement in a link j can be calculated using the successive screw displacement method (see Section 3.1) by premultiplying the homogeneous matrices corresponding to the preceding joint motions, i.e.

$$A_j = \prod_{i=1}^{j-1} A_i. \tag{22}$$

As in a closed-loop chain, the first and the last links are the same, and the orientation and position of a link with respect to itself are given by a homogeneous matrix equal to the fourth-order identity matrix. In a closed-loop chain with *np* primary joints and *ns* secondary joints (Eq. (22)), the closed-loop equation results in

$$\prod_{i=1}^{np} [A_p]_i \prod_{i=1}^{ns} [A_s]_i = I, \tag{23}$$

where $[A_p]_i$, $i = 1 \ldots np$ are the homogeneous matrices corresponding to the primary joints, and $[A_s]_i, i = 1 \ldots ns$ are the homogeneous matrices corresponding to the secondary joints.

Consider a closed-loop that has an error chain as shown in Fig. 2. As in ref. [6], we represent the closure error with a homogeneous matrix E, and the closed-loop equation becomes

$$\left\{ \prod_{i=1}^{np} [A_p]_i \prod_{i=1}^{ns} [A_s]_i \right\} E = I. \tag{24}$$

The closure error is calculated by

$$E = \left\{ \prod_{i=1}^{np} [A_p]_i \prod_{i=1}^{ns} [A_s]_i \right\}^{-1} = \begin{bmatrix} R_e & p_e \\ 0 & 1 \end{bmatrix}, \tag{25}$$

where $p_e = [p_{ex} \;\; p_{ey} \;\; p_{ez}]^T$ is the position error vector and R_e is the rotation matrix error. The matrix R_e corresponds to errors measured in r_{ex}, r_{ey} and r_{ez} virtual rotative joints considering their structural conception.

The 'position' error (which is a posture error involving position and orientation) is given by the position error vector $q_e = [r_{ex} \;\; r_{ey} \;\; r_{ez} \;\; p_{ex} \;\; p_{ey} \;\; p_{ez}]^T$.

4.4. Numerical implementation

Applying the Euler integration method in Eq. (20) we obtain

$$q_s(t_k) = q_s(t_{k-1}) - N_s^{-1}(t_{k-1}) N_p(t_{k-1}) \dot{q}_p \Delta t + \cdots + N_s^{-1}(t_{k-1}) N_e(t_{k-1}) K_e q_e \Delta t. \tag{26}$$

5. Example

The method presented is illustrated by solving the position kinematic of a planar four bar mechanism (Fig. 1.)

In this example, joint 1 is considered primary while the others are secondary. Joint 1 moves from the initial position $\pi/4$ to the final position $\pi/2$ according to $q_p(t) = \pi/4 + \pi/4 \sin(\pi\, t/8)$ from $t = 0$ to 4 s.

The magnitude of $\dot{q}_p(t)$ can be obtained by differentiating

$$qp(t) : \dot{q}_p(t) = \frac{\pi^2}{32} \cos\left(\frac{\pi t}{8}\right).$$

https://doi.org/10.1017/S0263574709005426 Published online by Cambridge University Press

The kinematic parameters are $l_1 = 0.5\,\text{m}, l_2 = 1.0\,\text{m}, l_3 = 0.5\,\text{m}$ and $l_4 = 1.0\,\text{m}$. The initial position vector is $q_s(0) = [\frac{-\pi}{4} \;\; \frac{-3\pi}{4} \;\; \frac{-\pi}{4}]^{\text{T}}$ rad. The integration interval is $\Delta t = 0.01\,\text{s}$.

Considering the reference frame attached to the last link of the error chain, the network matrices result in:

$$N_p = \begin{bmatrix} 1 \\ l_3 s_4 + l_2 s_{34} + l_1 s_{123} \\ l_4 + l_3 c_4 + l_2 c_{34} + l_1 c_{123} \end{bmatrix},$$

$$N_s = \begin{bmatrix} 1 & 1 & 1 \\ l_3 s_4 + l_2 s_{34} & l_3 s_4 & 0 \\ l_4 + l_3 c_4 + l_2 c_{34} & l_4 + l_3 c_4 & l_4 \end{bmatrix},$$

$$N_e = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & e_y \\ 1 & 0 & -e_x \end{bmatrix},$$

where: e_x, e_y are the prismatic displacement of the Assur virtual error chain; s_{ij} and c_{ij} are the *sine* and *cosine* of $q_i + q_j + \cdots$.

The secondary position vector is calculated using Eq. (26) and the resulting vector components are compared with the analytical solution presented by Waldron and Kinzel.[9]

The solution of the planar four bar mechanism is obtained choosing $K_e = K\, I_3$, where K is a positive scalar and I_3 is the 3×3 identity matrix. Varying K from 10 to 190 and the number of iterations from 1 to 4, we obtain the results shown in Fig. 11.

Figure 11 shows the maximum error between the numerical solution and the analytical solution as a function of K. To maintain the error magnitude within an admissible error we can choose the gain value combined with iterations in order to obtain a result that is within a given tolerance ($\|q_e\| < \xi$). It can be observed that the best result is obtained for $K = 100$, with any number of iterations.

6. Experimental Result

To evaluate the performance of the proposed method a real application was implemented.

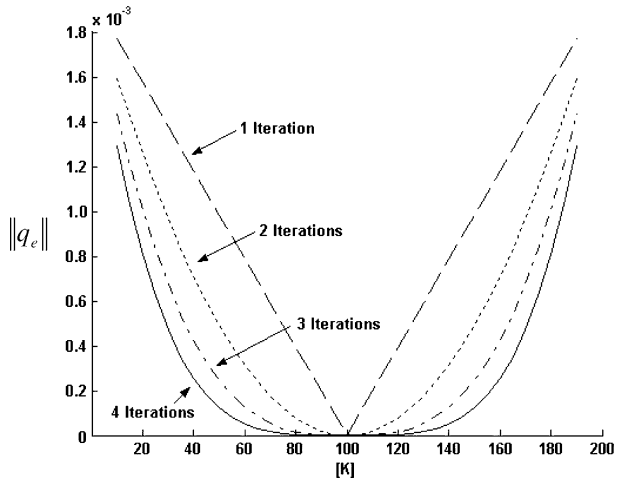

Fig. 11. Numerical solution results.

Fig. 12. The Roboturb prototype.

Fig. 13. Real experimental environment.

In this experiment a 7-DOF manipulator called the Roboturb manipulator was used. The Roboturb manipulator was conceived to recover blades with surfaces eroded by cavitations, a phenomenon found in hydroelectric power turbines.[12] This manipulator is composed of one prismatic joint, which comprises a mobile platform that moves on a flexible rail and six revolute joint (including a spherical wrist). Figure 12 depicts the Roboturb manipulator.

The Roboturb manipulator carries out the recovery through an automated welding process, where the material is deposited in parallel chords in interposed layers.[13]

The trajectories have to follow the recovery welding process specifications[14] and an eroded model was constructed to represent the real eroded surface. Figure 13 depicts the real environment and Fig. 14 the simulated environment.

To apply the proposed differential kinematics method, the closed chains are firstly defined. To impose a desired trajectory, a 3P3R Assur virtual chain is attached to the

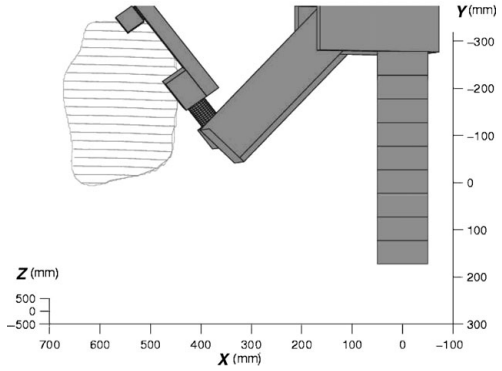

Fig. 14. Virtual environment to calculate the weld trajectories.

https://doi.org/10.1017/S0263574709005426 Published online by Cambridge University Press

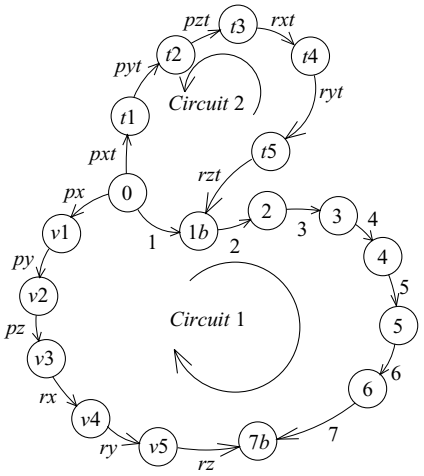

Fig. 15. Graph of two kinematic circuits.

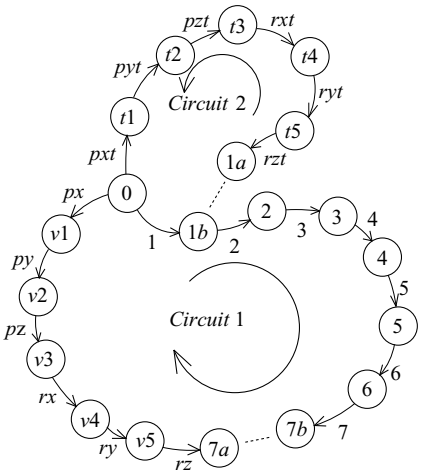

Fig. 16. Opening the circuits.

end-effector with the base attached to the trajectory reference frame, this closed chain defines *circuit* 1. To solve the redundancy, a classical proposed solution is to define the prismatic velocity through the projection of the end-effector velocity.[12] Adding a second 3P3R Assur virtual chain attached to the mobile platform with the base attached to the reference frame, a velocity can be imposed activating some of the prismatic virtual joints to specify the mobile platform (or prismatic) velocity, and this closed chain defines *circuit* 2.

It can be seen that the resultant Roboturb solution is a parallel robot with two closed chains defined by *circuits* 1 and 2.

To represent these closed chains, we enumerate each link and joint and use graph notation.[15] Figure 15 shows the resultant graph obtained by the Davies method.
where:

- edges 1, 2, 3, 4, 5, 6 and 7 represent the Roboturb links;
- vertices 1, 2, 3, 4, 5, 6 and 7 represent the Roboturb joints;
- edges $v1$, $v2$, $v3$, $v4$ and $v5$ represent the virtual trajectory links;
- vertices px, py, pz, rx, ry and rz represent the virtual trajectories joints;
- edges $t1$, $t2$, $t3$, $t4$, $t5$ represent the virtual trajectory chain links;
- vertices pxt, pyt, pzt, rxt, ryt and rzt represent the virtual trajectory chain joints.

The direction of each edge indicates the joint direction.

To represent the numerical integration error the closed kinematics must be opened at a convenient point. This point must be selected in such way that each circuit possesses only one opening: *circuit* 1 opened at link 7, forming links 7a and 7b; and *circuit* 2 opened at link 1, forming links 1a and 1b.

Figure 16 shows the direct graph opened at links 1 and 7.

In tridimensional space, the error chain must also be a 3P3R Assur virtual chain. Adding the error Assur virtual chains to these two circuits, the following complement to each circuit is obtained as depicted in Figs. 17 and 18.

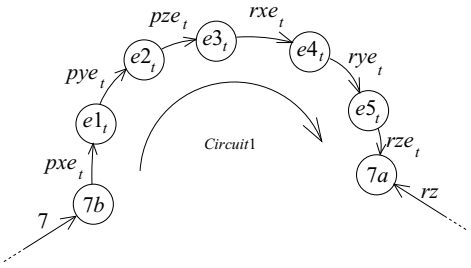

Fig. 17. Error Assur virtual chain for *circuit* 1.

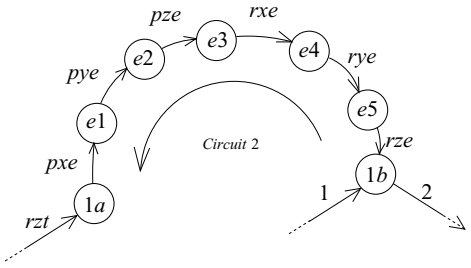

Fig. 18. Error Assur virtual chain for *circuit* 2.

where:

- edges $e1_t$, $e2_t$, $e3_t$, $e4_t$ and $e5_t$ represent the links of the virtual error trajectory;
- vertices pxe_t, pye_t, pze_t, rxe_t, rye_t and rze_t represent the joints of the virtual error trajectory.

where:

- edges $e1$, $e2$, $e3$, $e4$ and $e5$ represent the links of the virtual error chain for the redundancy solution;
- vertices pxe, pye, pze, rxe, rye and rze represent the joints of the virtual error chain for the redundancy solution.

Considering the final closed chain according to the Davies method, the differential kinematics can be represented by the following matrices, to give Eq. (26).

https://doi.org/10.1017/S0263574709005426 Published online by Cambridge University Press

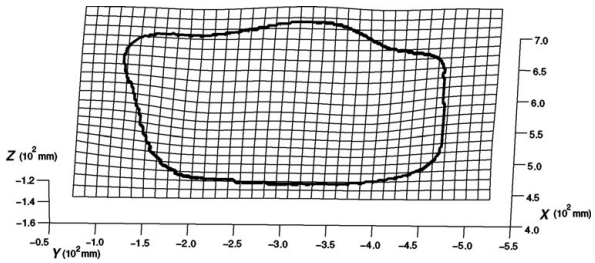

Fig. 19. Calculated contour trajectory.

$$N_p = \begin{bmatrix} 0 & -\hat{\$}_{px} & -\hat{\$}_{py} & -\hat{\$}_{pz} & -\hat{\$}_{rx} & -\hat{\$}_{ry} & -\hat{\$}_{rz} \\ -\hat{\$}_{pyt} & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$N_s = \begin{bmatrix} \hat{\$}_1 & \hat{\$}_2 & \hat{\$}_3 & \hat{\$}_4 & \hat{\$}_5 & \hat{\$}_6 & \hat{\$}_7 & 0 & 0 & 0 & 0 & 0 \\ \hat{\$}_1 & 0 & 0 & 0 & 0 & 0 & 0 & -\hat{\$}_{pxt} & -\hat{\$}_{pzt} & -\hat{\$}_{rxt} & -\hat{\$}_{ryt} & -\hat{\$}_{rzt} \end{bmatrix}$$

$$N_e = \begin{bmatrix} \hat{\$}_{pxe_t} & \hat{\$}_{pye_t} & \hat{\$}_{pze_t} & \hat{\$}_{rxe_t} & \hat{\$}_{rye_t} & \hat{\$}_{rze_t} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -\hat{\$}_{pxe} & -\hat{\$}_{pye} & -\hat{\$}_{pze} & -\hat{\$}_{rxe} & -\hat{\$}_{rye} & -\hat{\$}_{rze} \end{bmatrix}$$

The desired trajectory was calculated considering the contour of the delimitation path in Fig. 13. The delimitation path is composed of a set of 493 nodes as shown in Fig. 19.

The contour trajectory was obtained from the results of the laser sensor analysis.[16] Using the Bezier formulation the mathematic model of the eroded surface was obtained and using the Coons surface approximation,[17] the final surface was obtained. Thus, the delimitation path was computed from the intersection of the surfaces using the cube algorithm.[11]

Applying the proposed differential kinematics model, and using the Euler numerical integration method, the inverse kinematics can be calculated. For this application, and as a particular case of the modelling method, we approximate the primary joint velocities $\dot{q}_p(t)$ as follows:

$$\dot{q}_p(t) \approx \frac{q_p(t_k) - q_p(t_{k-1})}{\Delta t}. \tag{27}$$

The control variable is $\|q_e\|$, i.e. the norm of the error displacements q_e, at each instant t_k. Setting the admissible limit error ξ to 10^{-6} the joint trajectories were calculated using the following algorithm:

Algorithm 1. Inverse kinematics using the proposed integration method.

1. *Set parameters* K_e, ξ, $q_s(t_0)$ *and* $q_p(t_0)$
2. ***for*** *each instant* t_k ***do***
   a. *Set* $q_p = q_p(t_k)$
   b. *Calculate* $\|q_e\|$, *Eq.* (25)
   c. ***while*** $\|q_e\| > \xi$ ***do***
      i. *Calculate* N_s, N_p and N_e
      ii. *Calculate* q_s, *Eq.* (26)
      iii. *Calculate* $\|q_e\|$, *Eq.* (25)
   d. ***end while***
   e. *save* $q_s(t_k)$
3. ***end for***

The convergence is shown for a set t_k instants (t_{128}, t_{129} and t_{130}) in Fig. 20.

Another important result can be seen in the first iteration to each t_k instant: where the error decrease is more accentuated in comparison with the other iterations.

Figure 21 shows the final results of this experiment, where, using a pen fixed at the end-effector, the desired trajectory is correctly obtained by the Roboturb manipulator.

7. Conclusion

A new method to obtain the differential kinematics model for closed-chain robots is presented.

The differential kinematics equation is integrated numerically to obtain the joint positions. The model guaranteeing the closure error exponential convergence is presented and its stability properties are shown theoretically. The modelling method is based on the use of screw theory, Assur virtual chains and Davies method.

The error control is outlined and the influence of various iterations is analysed considering a numerical example.

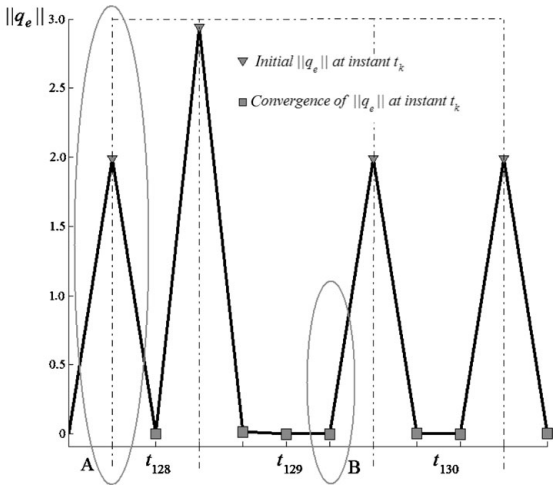

Fig. 20. Error $\|q_e\|$ convergence at instants t_k (t_{128}, t_{129} and t_{130}). Notice that after 1–3 iterations, $\|q_e\|$ converges in region B. In region A, just after the convergence, q_p assumes a new value $q_p(t_k)$; a new q_e arises, and the whole process repeats.

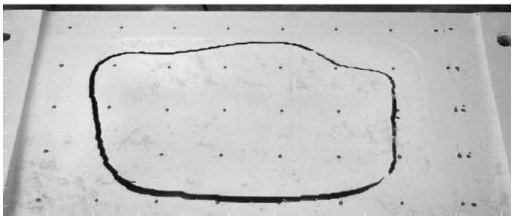

Fig. 21. Final result for the task of delimiting a contour.

https://doi.org/10.1017/S0263574709005426 Published online by Cambridge University Press

These results show that there is a compromise between error, number of iterations (computer time) and algorithm gains.

The differential model was applied to a 7-DOF manipulator in a real application know as Roboturb. In this application was validated the method for a 3D workspace was validated and the closure error convergence was within the desired limit.

The purpose of this study is to include the proposed technology in an application for the Roboturb manipulator in order to avoid collisions. This application improves the effective operation of the Roboturb manipulator.

Acknowledgements

This study was supported by 'Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPQ)' Brazil and 'Instituto de Tecnologia para o Desenvolvimento (Lactec)' Brazil.

References

1. A. Campos, R. Guenther and D. Martins, "Differential kinematics of serial manipulators using virtual chains," *J. Braz. Soc. Mech. Sci. Eng.* **27**(4), 345–356 (2005).
2. I. I. Artobolevski, *Théorie des Mécanismes et des Machines* (Mir Publishers, Moscow, Russia, 1977).
3. A. Campos, Cinemática diferencial de manipuladores empregando cadeias virtuais *Ph.D. Thesis* (Florianópolis, Brazil: Universidade Federal de Santa Catarina, 2005).
4. J. K. Davidson and K. H. Hunt, *Robots and Screw Theory: Applications of Kinematics and Statics to Robotic* (Oxford University Press, New York, USA, 2004).
5. T. H. Davies, "Kirchhoff's circulation law applied to multi-loop kinematic chain," *Mech. Mach. Theory* **16**(3), 171–183 (1981).
6. G. Pavlin, Automation Position Analysis of Spatial Kinematic Chains *Diploma Thesis* (TU-Wien, Austria: Institute für Mechanik, 1995).
7. L. Sciavicco and B. Siciliano, *Modelling and Control of Robot Manipulators* (Springer, New York, USA, 2004).
8. L. W. Tsai, *Robot Analysis: The Mechanics of Serial and Parallel Manipulators* (Wiley, New York, USA, 1999).
9. K. J. Waldron and G. L. Kinzel, *Kinematics, Dynamics, and Design of Machinery* (Wiley, New York, USA, 1999).
10. O. Bottema and B. Roth, *Theoretical Kinematics* (North-Holland Publishing Co., Amsterdam, The Netherlands 1979).
11. D. Qiulin and B. Davies, *Surface Engineering Geometry for Computer-Aided Design and Manufacture* (Ellis Horwood Limited, New York, USA, 1987).
12. H. Simas, D. Martins and R. Guenther, "A Redundant Manipulator to Operate in Confined Spaces," *Proceedings VI Induscon-Conferência Internacional de Aplicações Industriais*, Joinville- Brazil (2004).
13. K. Suresh and D. C. H. Yang, "Constant scallop-height machining of free-form surfaces," *J. Eng. Ind.* **116**(2), 253–259 (1994).
14. N. G. Bonacorso, A. A. Gonçalves Jr. and J. C. Dutra, "Automation of the processes of surface measurement and of deposition by welding for the recovery of rotors of large-scale hydraulic turbines," *J. Mater. Process. Technol.* **179**(1–3), 231–238 (2006).
15. L. W. Tsai, *Mechanism Design: Enumeration of Kinematic Structures According to Function* (CRC-Press, New York, USA, 2001).
16. T. L. F. Pinto, Avaliação de desempenho de robôs industriais utilizando um braço de medição portátil *Master's Thesis* (Florianópolis, Brazil: Universidade Federal de Santa Catarina, 2001).
17. D. F. Rogers and J. A. Adams, *Mathematical Elements for Computer Graphics* (McGraw-Hill, New York, USA, 1989).
18. D. E. Whitney, "Resolved motion rate control of manipulators and human prosteses," *IEEE Trans. Man-Mach. Syst.* **10**(2), 47–53 (1969).
19. G. Marai, "A Real-time Approach for Singularity Avoidance in Resolved Motion Rate Control on Robotic Manipulators," *Proceedings of the IEEE International Conference on Robotics and Automation* (2002) pp. 1973–1978.
20. N. I. Manolescu, "A unified method for the formation of all planar jointed kinematic chains and Baranov trusses," *Environ. Plan. B* **6**(4), 447–454 (1979).