Asociación Argentina de Mecánica Computacional

Mecánica Computacional Vol XXXI, págs. 2833-2848 (artículo completo)
Alberto Cardona, Paul H. Kohan, Ricardo D. Quinteros, Mario A. Storti (Eds.)
Salta, Argentina, 13-16 Noviembre 2012

A COMPARATIVE STUDY OF THE KINEMATICS OF ROBOTS MANIPULATORS BY DENAVIT-HARTENBERG AND DUAL QUATERNION

Luiz Radavelli[a,d], Roberto Simoni[a,b], Edson Roberto De Pieri[a,c] and Daniel Martins[a]

[a]*Post Graduation of Mechanical Engineering, Federal University of Santa Catarina, Campus Florianópolis, Brasil. radavelli@mtm.ufsc.br, daniel@emc.ufsc.br, http://www.posmec.ufsc.br*

[b]*Mobility Engineering Center, Federal University of Santa Catarina, Campus Joinville, Brasil. roberto.simoni@ufsc.br, http://www.joinville.ufsc.br/*

[c]*Post Graduation of Engineering of Automation and Systems, Federal University of Santa Catarina, Campus Florianópolis, Brasil. edson@das.ufsc.br, http://www.das.ufsc.br/*

[d]*Mathematics Department, Federal University of Santa Catarina, Campus Florianópolis, Brasil. radavelli@mtm.ufsc.br, http://www.mtm.ufsc.br*

Keywords: Homogeneous transformations, kinematic equations, robot manipulators, dual-quaternion

Abstract. This paper presents a comparative study of the kinematics of robot manipulators between Denavit-Hartenberg convention and the Dual Quaternion approach. The kinematics of robot manipulators can be obtained from a traditional form by *Denavit-Hartenberg* convention. In this way, the posture (position and orientation) of the end-effector is determined from a homogeneous transformation matrix. The dual-quaternion algebra is composed of elements with 8 components and under conditions it represents the posture of a rigid body by a minimal form. The operations and the properties of the dual-quaternion algebra arise from the definitions of elements of a more general algebra: the *Clifford* algebra that provides the necessary framework to the approach chosen in this paper. In this paper a dual-quaternion algebra is used to model the kinematic equations of robot manipulators in a more compact representation. A numerical robustness analysis is performed and the main characteristics of the dual-quaternion approach and its performance with respect to the *Denavit-Hartenberg* method will be illustrated in a case study of a 3R robot manipulator.

1 INTRODUCTION

Robotic kinematics is commonly divided into two problems: direct kinematics and inverse kinematics. Direct kinematics the joint variables are given and the problem is to find the position of the end-effector. In the inverse kinematics the location of the end-effector is given and the problem is to calculate the joint variables.

The coordinate transformations often have singularities. Singularities in parallel robots are extremely damaging, may even reach the integrity of the robot because in this kind of structure, in singularities positions, it wins or loses degrees of freedom (Craig, 2005; Selig, 1992).

The kinematic analysis can also be treated by means of differential kinematics. In this case, the Jacobian matrix relates base and end-effector. Singularities occur when the Jacobian matrix loses its full rank.

Singularity problem justifies the need of a robust and precise approach to perform the kinematic of robots. Most important objectives are related with avoid singularity problems, obtain convenient equations for kinematic, and to reduce the computational cost. Dual quaternion constitutes a promising tool for kinematic analysis of serial and parallel robots (Hestenes, 1999; Murray et al., 1994; Porteous, 1995; Selig, 2000b; Sommer, 2001).

Several authors have shown advantages in the context of uses for dual quaternions in robotics kinematics. Shoham and Ben-Horin (2009) utilize Grassmann-Cayley algebra to study singularities of parallel robot and apply their investigation to a general class of *Gough Stewart plataforms*.

Agrawal (1987) and Akyar (2008) represent screw motion of a rigid body by dual quaternions and study the *Hamilton operator*, which arises from dual quaternion multiplication.

Walker and Shao (1991) present a new form to solve object locating. They use a formulation based on optimization by means of dual quaternions. The Hamilton operator also are studied and plays a central role in their algorithm.

Horn (1987) presents a closed-form solution to the least-square problem for points. The solution given uses unit quaternions to represent rotations. He explains how to calculate the axis and angle of rotation from unit quaternion and four solutions to the components of the quaternion are presented to ensure numerical accuracy.

Aspragathos and Dimitros (1998) present three approaches to solve the direct kinematics: homogeneous transformation via Denavit-Hartenberg, Lie algebra and screw theory modeled by dual quaternions. An important result presented by the authors is that the transformations are performed an iterative process ${}^{i+1}p = h\,{}^{i}p$ instead of the traditional conjugation ${}^{i+1}p = h\,{}^{i}p\,h^{*}$. This algorithm is applied on a five degree of freedom robot. Sahul et al. (2008) applied the same iterative process to analise a kind of $3R$ robot.

Sariyildiz et al. (2011) and Sariyildiz and Temeltas (2009) compare three inverse kinematic methods of serial manipulators, which is applied on *Stäubli RX 160L* robot. In their work they uses Paden-Kahan subproblems do derive inverse kinematic solution. Paden-Kahan subproblems can be seen on Murray et al. (1994).

Pennestrì and Valentini (2009) use dual quaternion algebra for description to the screw displacement and study the human motion. They use interpolation by means of dual quaternion on the motion captured by *OptiTrack System* (System, 2011).

Aydin and Kucuk (2006) use dual quaternion and present a closed-form analytical solution to the inverse kinematics of industrial robots with six degrees of freedom and spherical wrist. Gan et al. (2008) present a a closed-form analytical solution to the inverse kinematics of a spatial $7R$ robot.

Many other areas of applications includes quaternions and dual quaternions. For example, Kavan et al. (2008) and Vince (2008) applies dual quaternions to the computer graphic; Dooley and McCarthy (1993); Choe (2006); Campa and Camarillo (2008) applies to track planning and Chou (1992) to dynamic.

In this bibliography review several applications of quaternions and dual quaternions in robotics were observed. Quaternions and dual quaternions appear to be flexible in this way. This can be seen in the works of Walker and Shao (1991); Sariyildiz et al. (2011), which model screw theory by means of dual quaternion and solve the kinematic by iterative process and other works (Ge and McCarthy, 1991; Funda and Paul, 1990; Sommer, 2001) use dual quaternion and solve kinematic by the most traditional conjugation operation.

An iterative process is certainly a good way to perform rigid transformations, since requires only first transformation to be storage - the others will be calculate by the iteration. In Aspragathos and Dimitros (1998) and in Sahul et al. (2008) the iterative process is done.

Sahul et al. (2008) present computational analysis of homogeneous transformations and quaternions transformations in $3R$ robot manipulator and conclude that transformations by quaternions are more efficient, from a numerical operations point of view, than equivalent homogeneous matrices.

This paper present a study about rigid transformations by means of homogeneous matrices - Denavit-Hartenberg method - and by means of dual quaternions method, showing the compact form in which dual quaternions represent the kinematic equations. An analysis of computational efficiency is also performed. The necessary algebra of rigid transformations via dual quaternions comes from a specific Clifford algebra. Due to the importance of Clifford algebra, a general topic on this subject will be presented. Flexibility of dual quaternions promotes real advantages in modeling kinematic of robots. Not just points and vectors but lines, planes and kinematics pairs can be represented in Clifford algebra.

This paper is structured as follows. Section 2 presents the well-know Denavit-Hartenberg method (DH). Section 3 presents Clifford algebra, properties and main operations. The Spinors of tridimensional space, quaternions and dual quaternions, are shown. A general rigid body transformation is derived from dual quaternions method (DQ). Lower kinematic pairs, points, lines and planes are modeled in dual quaternions algebra. In section 4, the rigid transformations found apply to solve the kinematics of the robot $3R$ planar. Finally, a computational efficiency analysis is performed and conclusions are presented.

2 DENAVIT-HARTENBERG METHOD (DH)

Denavit-Hartenberg method represent each transformation by a specific convention established by a series of definitions.

Consider a robot manipulator with n kinematic pairs (rotative and prismatic). Let L_i the i-th link and j_i the i-th kinematic pair between L_{i-1} and L_i, $i = 1, 2, \ldots, n$. L_0 is the link between base and the first kinematic pair. After defining a reference coordinate system, a coordinate system must be incorporated in each joint of robot manipulator, also in the end-effector, in order to establish the coordinate transformation between links, and solve the robotic kinematics.

In summary, Denavit-Hartenberg method defines a frame $\mathcal{F}_i$, $i = 0, \ldots, n$, by:

- z_i-axis: axis of the $i+1$ axis;
- x_i-axis: is parallel to the common normal: $x_i = z_{i-1} \times z_i$;
- y_i-axis: follows from right-hand rule;

- O_i: intersection between z_i axis and common normal;
- O_i': intersection between z_{i-1} axis and common normal;

Then, a transformation from frame $\mathcal{F}_i$ to frame $\mathcal{F}_{i-1}$ is defined by DH parameters (see Fig. 1):

- a_i: distance from O_i and O_i' measured along common normal;
- d_i: distance from O_{i-1} and $O_{i'}$ measured along z_i;
- α_i: angle between axes z_{i-1} and z_i about axis x_i to be taken positive when rotation is made counter-clockwise;
- θ_i: angle between axes x_{i-1} and x_i about axis z_{i-1} to be taken positive when rotation is made counter-clockwise.

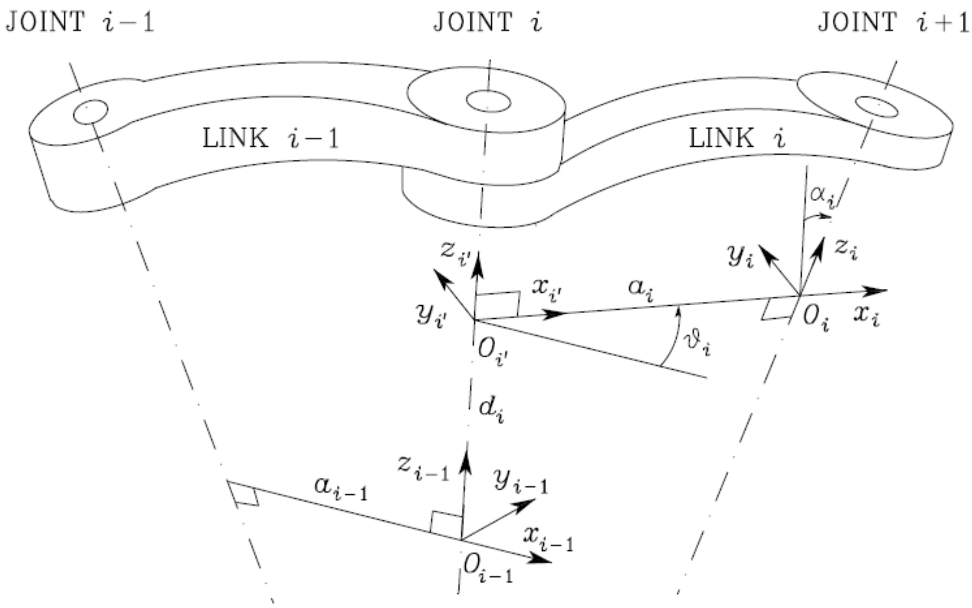

Figure 1: DH parameters. Figure by Sciavicco et al. (2009).

More about Denavit Hartenberg convention can be found in Sciavicco et al. (2009), Craig (2005), Tsai (1999) and Crane and Duffy (1998).

In general, DH parameters can be tabulated, leaving the transformations as function of the variable θ, in the revolute case, or d, in the prismatic case. Homogeneous coordinates allow to establish the relation between two adjacent links, connected by a kinematic pair, following four steps:

$$\text{Rotate } \alpha \text{ in } Ox \longrightarrow \text{Translate } a \text{ by } Ox \longrightarrow \text{Rotate } \theta \text{ in } Oz \longrightarrow \text{Translate } d \text{ in } Oz.$$

The steps above express the product of four homogeneous transformation matrix from frame $\mathcal{F}_i$ to frame $\mathcal{F}_{i-1}$, $i = 1, \ldots, n$, that is,

$$
\begin{aligned}
{}^{i-1}H_i &= T_z(d_i)R_z(\theta_i)T_x(a_i)R_x(\alpha_i) \\
&= \begin{bmatrix} \cos\theta_i & -\cos\alpha_i\sin\theta_i & \sin\alpha_i\sin\theta_i & a_i\cos\theta_i \\ \sin\theta_i & \cos\alpha_i\cos\theta_i & -\sin\alpha_i\cos\theta_i & a_i\sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}
\end{aligned}
\tag{1}
$$

This transformation matrix represents orientation and position of frame $\mathcal{F}_i$ with respect to frame $\mathcal{F}_{i-1}$. A ortogonal characteristic of left block matrix, $H(1:3, 1:3)$, simplifies the inversion transformation matrix.

If axes Oz_i, Oz_{i+1}, Ox_i and Ox_{i+1} are parallel, then $\alpha_i = 0$, $d_i = 0$, and the transformation matrix from frame $\mathcal{F}_i$ to frame $\mathcal{F}_{i-1}$ is simplified to

$$
{}^{i-1}H_i = \begin{bmatrix} \cos\theta_i & -\sin\theta_i & 0 & a_i\cos\theta_i \\ \sin\theta_i & \cos\theta_i & 0 & a_i\sin\theta_i \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{2}
$$

This is the transformation matrix from the joint i coordinate system to joint $i+1$ in the $3R$ planar robot, which will be presented in details in Section 4.1. Note that the ${}^{i-1}H_i$ matrix transformation has sparse structure.

Finally, if ${}^{i-1}p$ represents a point in coordinate frame $\mathcal{F}_{i-1}$, then the transformation from ${}^{i}p$ to ${}^{i-1}p$ is

$$
{}^{i-1}p = {}^{i-1}H_i\, {}^{i}p, \quad i = 1, \ldots, n.
$$

So, the kinematic of a robot, composed by n kinematic pairs, is

$$
{}^{0}p_e = {}^{0}H_e\, {}^{e}p_e \tag{3}
$$

where

$$
{}^{0}H_e = {}^{0}H_1\, {}^{1}H_2 \, \cdots \, {}^{n-1}H_e \tag{4}
$$

is the global transformation, i.e., the composition among the transformation of the frames $\mathcal{F}_0, \mathcal{F}_1, \ldots, \mathcal{F}_e$. The coordinate systems are incorporated locally, so the end-effector is ${}^{e}p_e = [0\ 0\ 0\ 1]^T$.

3 DUAL QUATERNION METHOD

This section presents a different way to describe the kinematics of robot manipulators. An alternative algebra models this end. Instead of homogeneous matrices, a specific Clifford algebra known as dual quaternions defines the kinematic of serial robots. The compact way dual quaternions represent the kinematic equations and its flexibility in modeling kinematic constitute some advantages.

The Clifford algebra (or geometric algebra) is a largest division algebra, associative, which includes several algebraic systems (like complex numbers algebra, vector algebra, matrix algebra, quaternions algebra, etc.) in a coherent and unified mathematical language. The geometric entities such point, line, plane, area, volume, and even the transformations of rotation and translation are basic members of Clifford algebra (Selig, 2000a) and, thus, can be handled by a set of algebraic operations defined in the own algebra.

3.1 Clifford algebra basics and Spinors

There are several ways to define a Clifford algebra and the definition will depend on its proposed (Lounesto, 2001). One way to deal with the Clifford algebra is to analyze it as vector space on $\mathbb{R}^n$. To this purpose, a basis of elements which define the Clifford space is necessary. Also, a quadratic bilinear form defined on a linear space $\mathbb{R}^n$ is required to allow the calculations with the Clifford elements.

A Clifford space is an extension of a Euclidean vector space that works with more general concepts - multivectors - introducing the concept of oriented component, like oriented areas and oriented volumes (Hestenes, 1999). A Clifford space becomes a Clifford algebra when a product between multivectors, called geometric product, is defined.

Considering an orthonormal basis $\{e_1, \ldots, e_n\}$ of Euclidian vector space $\mathbb{R}^n$, the corresponding n-dimensional Clifford space $\mathrm{Cl}(n)$ follows from geometric product definition. To Clifford space, e_i relates a generator element, and on two generators the geometric product produces

1. $e_i e_j + e_j e_i = 0$, if $i \neq j$;
2. $e_i^2 = \epsilon_i$

which define a Clifford algebra $\mathrm{Cl}(n)$. $\epsilon_i = +1, -1, 0$ represents the signature of an generator. Algebras with 0 signature generators are non degenerate algebras, otherwise they are degenerate algebras. Thus, a $\mathrm{Cl}(\mathrm{p,q,r})$ also represents an algebra $\mathrm{Cl}(n)$ with $n = p + q + r$: p, q and r represent the amounts of generators with signature $+1$, -1 and 0, respectively. As a vector space, a Clifford algebra $\mathrm{Cl}(n)$ has 2^n basis elements: generators and all their possible combinations. A generator conjugate is defined by $e_i^* = -e_i$.

Geometric product defines other products. For $\vec{u}$ and $\vec{v}$ in $\mathbb{R}^3$ space, $\vec{u} \cdot \vec{v} = \frac{1}{2}(u\,v + v\,u)$ and $\vec{u} \times \vec{v} = \frac{1}{2}(u\,v - v\,u)$.

To illustrate, the Clifford algebra $\mathrm{Cl}(0,0,1)$ has e as generator, and $e^2 = 0$. As vector space, Clifford algebra $\mathrm{Cl}(1)$ has basis $\{1, e\}$. So, a general element is $d = a_1 + e\,a_2$. Actually there is a isomorphism $\mathrm{Cl}(0,0,1) \simeq \mathbb{D}$ - the dual numbers. The dual conjugate of d is $d^* = a_1 - e\,a_2$ or $d^* = a_1 - \varepsilon\,a_2$ in dual numbers notation. When end-effector posture is represented by point or plane, the dual conjugate becomes fundamental for kinematics via dual quaternions.

On the Clifford algebra context, two other spaces are also important: The non degenerate algebra $\mathrm{Cl}(0,2,0)$, constituted by basis

$$\{1,\ e_1,\ e_2,\ e_1e_2\} \tag{5}$$

with generators $e_1^2 = e_2^2 = -1$, and the degenerated algebra $\mathrm{Cl}(0,3,1)$, constituted by basis

$$\{1,\ e_1,\ e_2,\ e_3,\ e,\ e_1e_2,\ e_2e_3,\ e_3e_1,\ e_1e,\ e_2e,\ e_3e,\ e_1e_2e_3,\ e_1e_2e,\ e_2e_3e,\ e_3e_1e,\ -e_1e_2e_3e\} \tag{6}$$

with generators $e_1^2 = e_2^2 = e_3^2 = -1$ and $e^2 = 0$.

In a general Clifford algebra, the elements e_i are vectors, e_ie_j are bivectors, $e_ie_je_k$ are trivectors, and so on. Thus, a multivector is a general element of a Clifford algebra and the number of generators which compose a multivector defines the degree of an element.

Clifford algebras have many isomorphisms which allow multiple representations for the group of rigid motions. One representation of the Clifford algebra is the decomposition

$$\mathrm{Cl}(p,q,r) = \mathrm{Cl}^+(p,q,r) \oplus \mathrm{Cl}^-(p,q,r)$$

where $\mathrm{Cl}^+(p,q,r)$ subalgebra consisting by elements of even degree, named *Spinors*. Spinors are rotors and translators and they perform rotations and translations. In spinors algebra, the isomorphism $\mathrm{Cl}^+(p,q,r) \simeq \mathrm{Cl}(p,q-1,r)$ produces two other important isomorphisms: $\mathrm{Cl}^+(0,3,0) \simeq \mathrm{Cl}(0,2,0) = \mathbb{H}$ - the quaternions space - and $\mathrm{Cl}(0,2,1) \simeq \mathrm{Cl}^+(0,3,1) = \mathbb{H}_2$ - the dual quaternions space. These two isomorphisms, say the spinors of $\mathbb{R}^3$, are quaternions and dual quaternions. So, there is no difference between bases in Eq. 5 and Eq. 6 and bases $\{1,i,j,k\}$, $\{1,i,j,k,\varepsilon,i\varepsilon,j\varepsilon,k\varepsilon\}$, respectively.

Two elements $q_1 = a_0 + a_1 i + a_2 j + a_3 k = a + \vec{u}$ and $q_2 = b_0 + b_1 i + b_2 j + b_3 k = b + \vec{v}$ are quaternions, and

$$q_1\, q_2 = xy - \vec{u}\cdot\vec{v} + x\vec{v} + y\vec{u} + \vec{u}\times\vec{v} \tag{7}$$

defines a quaternion product which states the quaternion algebra. The quaternion conjugate is $q^* = a - \vec{u}$. Also, $h_i = q_{i1} + \varepsilon q_{i2}$ defines a dual quaternion and

$$h_1\, h_2 = q_{11}q_{21} + \varepsilon(q_{11}q_{22} + q_{12}q_{21}) \tag{8}$$

is the dual quaternion product, $h^* = q_1^* + \varepsilon\, q_2^*$ is the dual quaternion conjugate, and $\overline{h^*} = q_1 - \varepsilon\, q_2^*$ is dual conjugate of dual quaternion conjugate.

Quaternions and dual quaternions can be used to perform rotations followed by translations. This is done by

$${}^{j}p = {}^{j}R_i\, {}^{i}p\, {}^{j}R_i^{-1}$$

where R is the spinor operator. The R^{-1} is the inverse of spinor R and it is defined by $R^{-1} = R^*/\|R\|$.

Spinor operator R which perform rotation from θ about an unit vector axis $\vec{s}$ is the quaternion

$$q = \cos\frac{\theta}{2} + \vec{s}\sin\frac{\theta}{2}$$

Moreover, if $\|\vec{s}\| = 1$ then $\|q\| = 1$ and $q^{-1} = q^*$, which simplifies the operation of rotation considerably.

Also, the spinor operator R which defines rotation followed by translation (along some vector $\vec{t}$) is the dual quaternion product of translational spinor h_T (translator) and rotational spinor h_R (rotor)

$$h = h_T\, h_R = \left(1 + \varepsilon\frac{\vec{t}}{2}\right) q = q + \varepsilon\left(\frac{\vec{t}}{2}\, q\right).$$

This dual quaternion spinor appears in (Selig, 2000b). If $\|h\| = 1$ then dual quaternion represents the posture of a rigid body by minimal form.

In the helical movement, $\vec{t} = d\vec{s}$, where d is the distance of translation along parallel axis of rotation - screw axis actually. So the dual quaternion of a general transformation from frame $\mathcal{F}_i$ to frame $\mathcal{F}_{i-1}$ is

$$h^{O\vec{s}}_{\hat{\theta}_i} = \begin{bmatrix} \cos\frac{\theta}{2} \\ (\sin\frac{\theta}{2})s_x \\ (\sin\frac{\theta}{2})s_y \\ (\sin\frac{\theta}{2})s_z \\ -\frac{d}{2}\sin\frac{\theta}{2} \\ (\frac{d}{2}\cos\frac{\theta}{2})s_x \\ (\frac{d}{2}\cos\frac{\theta}{2})s_y \\ (\frac{d}{2}\cos\frac{\theta}{2})s_z \end{bmatrix} \tag{9}$$

This is the dual quaternion of the kinematic transformation with arbitrary axis $\vec{s} = [s_x\ s_y\ s_z]^T$.

Quaternion and dual quaternion product perform n successive transformations by

$$^{i}p = (^{i}R_{i+1} \cdots {}^{i+n-1}R_{i+n})\ {}^{i+n}p\ (^{i}R_{i+1} \cdots {}^{i+n-1}R_{i+n})^*.$$

Following the Denavit-Hartenberg method, transformation between frames $\mathcal{F}_i$ and $\mathcal{F}_{i-1}$ is a dual quaternion

$$^{i-1}h_i = h^{Oz}(d_i, \theta_i)\ h^{Ox}(a_i, \alpha_i) = h^z_{\hat{\theta}_i}\ h^x_{\hat{\alpha}_i}$$

where $\hat{\theta}_i$ and $\hat{\alpha}_i$ are the dual representations of the four parameters of Denavit-Hartenberg, i.e.,

$$\hat{\theta}_i = \theta_i + \varepsilon\, d_i \quad \text{and}$$
$$\hat{\alpha}_i = \alpha_i + \varepsilon\, a_i.$$

The dual quaternion $h^z_{\hat{\theta}_i}$ performs the transformation of the frame $\mathcal{F}_{i-1}$ about Oz_{i-1} axis, leaving Ox_{i-1} and Ox_i coincident. Likewise, $h^x_{\hat{\alpha}_i}$ is performed on the frame $\mathcal{F}_{i'}$ aligning the two axis Oz_{i-1} and Oz_i, bringing the frames $\mathcal{F}_{i-1}$ and $\mathcal{F}_i$ coincident.

Defining the auxiliary equations

$$\begin{aligned} \tilde{a}_i &= a_i/2, & A_i &= \cos(\alpha_i/2)\cos(\theta_i/2), & B_i &= \sin(\alpha_i/2)\cos(\theta_i/2), \\ \tilde{d}_i &= d_i/2, & C_i &= \sin(\alpha_i/2)\sin(\theta_i/2), & D_i &= \cos(\alpha_i/2)\sin(\theta_i/2), \end{aligned} \tag{10}$$

a general displacement transformation, in dual quaternions algebra, reduces to

$$^{i-1}h_i = \begin{bmatrix} \cos(\alpha_i/2)\cos(\theta_i/2) \\ \sin(\alpha_i/2)\cos(\theta_i/2) \\ \sin(\alpha_i/2)\sin(\theta_i/2) \\ \cos(\alpha_i/2)\sin(\theta_i/2) \\ -\frac{a_i}{2}\sin(\alpha_i/2)\cos(\theta_i/2) - \frac{d_i}{2}\cos(\alpha_i/2)\sin(\theta_i/2) \\ \frac{a_i}{2}\cos(\alpha_i/2)\cos(\theta_i/2) - \frac{d_i}{2}\sin(\alpha_i/2)\sin(\theta_i/2) \\ \frac{a_i}{2}\cos(\alpha_i/2)\sin(\theta_i/2) + \frac{d_i}{2}\sin(\alpha_i/2)\cos(\theta_i/2) \\ -\frac{a_i}{2}\sin(\alpha_i/2)\sin(\theta_i/2) + \frac{d_i}{2}\cos(\alpha_i/2)\cos(\theta_i/2) \end{bmatrix} = \begin{bmatrix} A_i \\ B_i \\ C_i \\ D_i \\ -\tilde{a}_iB_i - \tilde{d}_iD_i \\ \tilde{a}_iA_i - \tilde{d}_iC_i \\ \tilde{a}_iD_i + \tilde{d}_iB_i \\ -\tilde{a}_iC_i + \tilde{d}_iA_i \end{bmatrix}. \tag{11}$$

This is the dual quaternion version of the rigid transformation from Denavit-Hartenberg presented in Eq. 1. The same DH parameters table establishes a serial robot kinematic by dual quaternions, and Eq. 11 defines the dual quaternion transformations necessary to end-effector posture.

The dual quaternion version for rotative (R), prismatic (P), cylindrical (C), helical (H), spherical (S) and planar (F) kinematic pairs are

$$h_R = \begin{bmatrix} \cos\frac{\theta}{2} \\ (\sin\frac{\theta}{2})s_x \\ (\sin\frac{\theta}{2})s_y \\ (\sin\frac{\theta}{2})s_z \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad h_P = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ (\frac{d}{2})s_x \\ (\frac{d}{2})s_y \\ (\frac{d}{2})s_z \end{bmatrix}, \quad h_S = \begin{bmatrix} \sin\frac{\theta_x}{2}\sin\frac{\theta_y}{2}\sin\frac{\theta_z}{2} + \cos\frac{\theta_x}{2}\cos\frac{\theta_y}{2}\cos\frac{\theta_z}{2} \\ \sin\frac{\theta_x}{2}\cos\frac{\theta_y}{2}\cos\frac{\theta_z}{2} - \cos\frac{\theta_x}{2}\sin\frac{\theta_y}{2}\sin\frac{\theta_z}{2} \\ \cos\frac{\theta_x}{2}\sin\frac{\theta_y}{2}\cos\frac{\theta_z}{2} + \sin\frac{\theta_x}{2}\cos\frac{\theta_y}{2}\sin\frac{\theta_z}{2} \\ \cos\frac{\theta_x}{2}\cos\frac{\theta_y}{2}\sin\frac{\theta_z}{2} - \sin\frac{\theta_x}{2}\sin\frac{\theta_y}{2}\cos\frac{\theta_z}{2} \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix},$$

$$h_C = \begin{bmatrix} \cos\frac{\theta}{2} \\ (\sin\frac{\theta}{2})s_x \\ (\sin\frac{\theta}{2})s_y \\ (\sin\frac{\theta}{2})s_z \\ -\frac{d}{2}\sin\frac{\theta}{2} \\ (\frac{d}{2}\cos\frac{\theta}{2})s_x \\ (\frac{d}{2}\cos\frac{\theta}{2})s_y \\ (\frac{d}{2}\cos\frac{\theta}{2})s_z \end{bmatrix}, \quad h_H = \begin{bmatrix} \cos\frac{\theta}{2} \\ (\sin\frac{\theta}{2})s_x \\ (\sin\frac{\theta}{2})s_y \\ (\sin\frac{\theta}{2})s_z \\ -\frac{d}{2}\sin\frac{\theta}{2} \\ (\frac{d}{2}\cos\frac{\theta}{2})s_x \\ (\frac{d}{2}\cos\frac{\theta}{2})s_y \\ (\frac{d}{2}\cos\frac{\theta}{2})s_z \end{bmatrix}, \quad h_F = \begin{bmatrix} \cos\frac{\theta}{2} \\ 0 \\ 0 \\ \sin\frac{\theta}{2} \\ 0 \\ \frac{dx}{2}\cos\frac{\theta}{2} - \frac{dy}{2}\sin\frac{\theta}{2} \\ \frac{dx}{2}\sin\frac{\theta}{2} + \frac{dy}{2}\cos\frac{\theta}{2} \\ 0 \end{bmatrix}.$$

Using the dual quaternions algebra the kinematics is established by

$$^0p_e = {}^0h_e \, {}^ep_e \, {}^0\widetilde{h}_e, \tag{12}$$

where ^0p_e is the dual quaternion of the end-effector position according to the parameters of rotation and translation, represented in the reference frame, ^ep_e is the dual quaternion of the initial position of the end-effector, represented in the end-effector frame, i.e., $^ep_e = [1\ 0\ 0\ 0\ 0\ 0\ 0\ 0]^T$.

The definition of $\widetilde{h}$ depends on the element used to represent the displacement. For example, if only the position $P = (x, y, z)$ of the end-effector is under control, then a point has sufficient degrees of freedom and can be used. In other words, to position control, $p_e = 1 + P\varepsilon$ and $\widetilde{h} = \overline{h^*} = q_1^* - q_2^*\varepsilon$. If the posture (position and orientation) of the end-effector is requested, then line and plane have sufficient degrees of freedom and may be used. In the first case $\widetilde{h} = h^*$ while in the second one, $\widetilde{h} = \overline{h^*}$.

3.2 Representation of geometric linear elements

In geometric algebra as much points as lines and planes are basic elements. Based on linear algebra the following dual quaternion representation for points, lines and planes can be stated:

$$\begin{aligned} P &= 1 + xi\varepsilon + yj\varepsilon + zk\varepsilon \\ &= \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & x & y & z \end{bmatrix}^T \end{aligned}$$

represent the point in dual quaternion coordinates. (x, y, z) are the components of the point in the cartesian coordinate system. To lines,

$$\begin{aligned} L &= s + m\varepsilon \\ &= 0 + s_x i + s_y j + s_z k + 0\varepsilon + m_x i\varepsilon + m_y j\varepsilon + m_z k\varepsilon \\ &= \begin{bmatrix} 0 & s_x & s_y & s_z & 0 & m_x & m_y & m_z \end{bmatrix}^T, \end{aligned}$$

where s is the unit vector of the line, m is the moment of the line given by $m = s \times s_0$, and s_0 is the position vector of an arbitrary point on the line. Planes are described by its normal vector $\vec{n}$ and the distance from the origin. In dual quaternion coordinates,

$$\begin{aligned} \pi &= n + d\varepsilon \\ &= n_x i + n_y j + n_z k + d\varepsilon \\ &= \begin{bmatrix} 0 & n_x & n_y & n_z & d & 0 & 0 & 0 \end{bmatrix}^T. \end{aligned}$$

The dual quaternion representation of basic geometrical elements models rigid body transformations and, therefore, the kinematics of robot manipulator.

4 CASE STUDY

In this section the methods studied in Section 2 and 3 are applied in the $3R$ robot manipulator shown in Figure 2. The links L_0 (base), L_1, L_2 and L_3 are connected by rotative joints J_1, J_2 and J_3. The end-effector e is attached to end of link L_3. There are three frames: $\mathcal{F}_0$, $\mathcal{F}_1$, $\mathcal{F}_2$ and $\mathcal{F}_3$=$\mathcal{F}_e$.

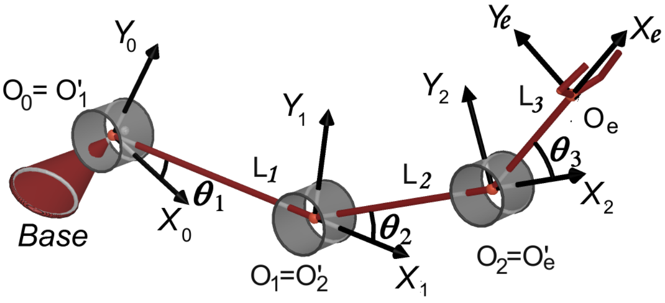

Figure 2: 3R robot.

All joints are rotative, so $q_i(\theta_i, d_i) = (\theta_i, 0)$, $i = 1, 2, 3$. For $3R$ robot the DH parameters are

	${}^0H_1$	${}^1H_2$	${}^2H_e$
a_i	L_1	L_2	L_3
α_i	0	0	0
d_i	0	0	0
θ_i	θ_1	θ_2	θ_3

Table 1: 3R robot parameters.

4.1 Kinematics via Denavit-Hartenberg

Denavit-Hartenberg method requires a coordinate system for each joint, a coordinate system to the end-effector and a reference coordinate system. If $\mathcal{F}_i$, $i = 0, 1, \ldots, n$ are these frames, where $\mathcal{F}_0$ is the base frame and $\mathcal{F}_n = \mathcal{F}_e$ is the end-effector frame, then Eq. 2 gives the transformation between frames $\mathcal{F}_i$ and $\mathcal{F}_{i-1}$ by

$$ {}^{i-1}H_i = \begin{bmatrix} \cos\theta_i & -\sin\theta_i & 0 & a_i\cos\theta_i \\ \sin\theta_i & \cos\theta_i & 0 & a_i\sin\theta_i \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}; \quad i = 1, 2, 3. \tag{13} $$

Thus, the global transformation explained in Eq. 4 is

$$ {}^0H_e = \begin{bmatrix} \cos(\theta_1+\theta_2+\theta_3) & -\sin(\theta_1+\theta_2+\theta_3) & 0 & a_{14} \\ \sin(\theta_1+\theta_2+\theta_3) & \cos(\theta_1+\theta_2+\theta_3) & 0 & a_{24} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, $$

with

$$a_{14} = L_1 \cos\theta_1 + L_2 \cos(\theta_1+\theta_2) + L_3 \cos(\theta_1+\theta_2+\theta_3), \text{ and}$$
$$a_{24} = L_1 \sin\theta_1 + L_2 \sin(\theta_1+\theta_2) + L_3 \sin(\theta_1+\theta_2+\theta_3).$$

Therefore, the end-effector position (Eq. 3) is

$$^0p_e = \begin{bmatrix} L_1 \cos\theta_1 + L_2 \cos(\theta_1+\theta_2) + L_3 \cos(\theta_1+\theta_2+\theta_3) \\ L_1 \sin\theta_1 + L_2 \sin(\theta_1+\theta_2) + L_3 \sin(\theta_1+\theta_2+\theta_3) \\ 0 \\ 1 \end{bmatrix}. \tag{14}$$

4.2 Kinematics via dual quaternion

The $3R$ planar robot axis are $\vec{s}_i = [0 \;\; 0 \;\; 1]^T$, with $\alpha_i = 0$ and $d_i = 0$, $i = 1, 2, 3$. Therefore, from Eq. 11 and DH parameters in Table 1, the transformations via dual quaternion are

$$^0h_1 = \begin{bmatrix} \cos\frac{\theta_1}{2} \\ 0 \\ 0 \\ \sin\frac{\theta_1}{2} \\ 0 \\ \frac{L_1}{2}\cos\frac{\theta_1}{2} \\ \frac{L_1}{2}\sin\frac{\theta_1}{2} \\ 0 \end{bmatrix}, \quad ^1h_2 = \begin{bmatrix} \cos\frac{\theta_2}{2} \\ 0 \\ 0 \\ \sin\frac{\theta_2}{2} \\ 0 \\ \frac{L_2}{2}\cos\frac{\theta_2}{2} \\ \frac{L_2}{2}\sin\frac{\theta_2}{2} \\ 0 \end{bmatrix}, \quad ^2h_e = \begin{bmatrix} \cos\frac{\theta_3}{2} \\ 0 \\ 0 \\ \sin\frac{\theta_3}{2} \\ 0 \\ \frac{L_3}{2}\cos\frac{\theta_3}{2} \\ \frac{L_3}{2}\sin\frac{\theta_3}{2} \\ 0 \end{bmatrix}. \tag{15}$$

The dual quaternion product determines the global transformation:

$$^0h_e = \begin{bmatrix} \cos\left(\frac{\theta_1+\theta_2+\theta_3}{2}\right) \\ 0 \\ 0 \\ \sin\left(\frac{\theta_1+\theta_2+\theta_3}{2}\right) \\ 0 \\ \frac{L_1}{2}\cos\left(\frac{\theta_1-\theta_2-\theta_3}{2}\right) + \frac{L_2}{2}\cos\left(\frac{\theta_1-\theta_2-\theta_3}{2}\right) + \frac{L_3}{2}\cos\left(\frac{\theta_1+\theta_2+\theta_3}{2}\right) \\ \frac{L_2}{2}\sin\left(\frac{\theta_1+\theta_2-\theta_3}{2}\right) + \frac{L_1}{2}\sin\left(\frac{\theta_1-\theta_2-\theta_3}{2}\right) + \frac{L_3}{2}\sin\left(\frac{\theta_1+\theta_2+\theta_3}{2}\right) \\ 0 \end{bmatrix}. \tag{16}$$

Using point representation to describe the end-effector position, Eq. 16 into Eq. 12 gives

$$^0p_e = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ L_1 \cos\theta_1 + L_2 \cos(\theta_1+\theta_2) + L_3 \cos(\theta_1+\theta_2+\theta_3) \\ L_1 \sin\theta_1 + L_2 \sin(\theta_1+\theta_2) + L_3 \sin(\theta_1+\theta_2+\theta_3) \\ 0 \end{bmatrix}, \tag{17}$$

which agree with the Denavit-Hartenberg calculations (14).

5 COMPUTATIONAL ANALYSIS

A comparative on the storage and numerical robustness may help you to choose the most convenient method for the application. This section presents storages and computational cost of the methods which been studied in most literatures. The best way to do this is using sparse matrix computations (Tewarson, 1973; Duff et al., 1989; Davis, 2006).

In sparse matrix study, only the values of the nonzeros elements and the index information telling where each nonzero belongs in the regular array are stored.

When storing and manipulating sparse matrices on a computer, it is beneficial and often necessary to use specialized algorithms and data structures that take advantage of the sparse structure of the matrix. Operations using standard dense matrix structures and algorithms are slow, and consume large amounts of memory when applied to large sparse matrices. Sparse data is by nature easily compressed, and this compression almost always results in significantly less computer data storage usage.

The DH method uses 4×4 homogeneous matrices with a specific structure, with zeros in specific index locations which requires 12 memory locations (Aspragathos and Dimitros, 1998). There is 6 multiplications and 4 trigonometric functions to define the matrix transformations ${}^{i-1}H_i$, $i = 1, \ldots, n$ in Eq. 1. To evaluate the number of numerical products from the first transformation to the successive ones, is necessary to establish the order in which the multiplications are performed. The best performance is achieved when computational operations are done from right to left. The first transformation is ${}^{n-2}H_{n-1}{}^{n-1}H_e$, and the computations returns the following numbers:

Fase		
Definition	1st Transformation	Next
0+	20+	23+
6$*$	29$*$	32$*$

Table 2: Computational performance of general DH transformations - Eq. 1.

"$*$" is used to state the multiplication and "$+$" to the addition operations. In n-link robot arm there are n matrix transformations of the form given in Eq. 1, so $29+6n+32(n-2) = 38n-35$ multiplications and $20 + 23(n-2) = 23n - 26$ additions is required to determine the global transformation and hence the end effector posture. So, a general robot with $n = 3$ requires 79$*$ and 43$+$.

In dual quaternion method 8 memory locations are required. A general dual quaternion transformation ${}^{i-1}h_i$ is given in the Eq. 11 and requires 20 multiplications, four additions and four trigonometric function calculations for its complete definition, but from the auxiliary equations (see Eq. 10) the multiplications are reduced to 14 (A similar fashion was not found to simplifying the computations in DH method).

Quaternion and dual quaternion multiplications play a central role in rigid body transformations then the computational cost study requires an analysis on these operations. The numerical cost of general quaternion multiplication are 16 multiplications and 12 additions, and to dual quaternion this numbers increase for 48 multiplications and 40 additions - see Eq.'s 7 and 8. So, defining and performing the transformation ${}^{i-1}h_i$ require the following numbers:

In this way, a n-link robot has $14n + 48(n-1) = 2(31n-24)$ multiplications and $40(n-1)$ additions to describe the global transformation by means of dual quaternions. A dual quaternion multiplication more is necessary to complete the kinematic equation 12.

Fase	
Definition	Transformations
0+	40+
14∗	48∗

Table 3: Computational performance of general DQ transformations - Eq. 11.

The numbers are summarized in Table 4 and Figure 3:

	DH Method	DQ Method
$+$	$23n - 26$	$40(n-1)$
$*$	$38n - 35$	$2(31n - 24)$

Table 4: Computational performance of n-link robot.

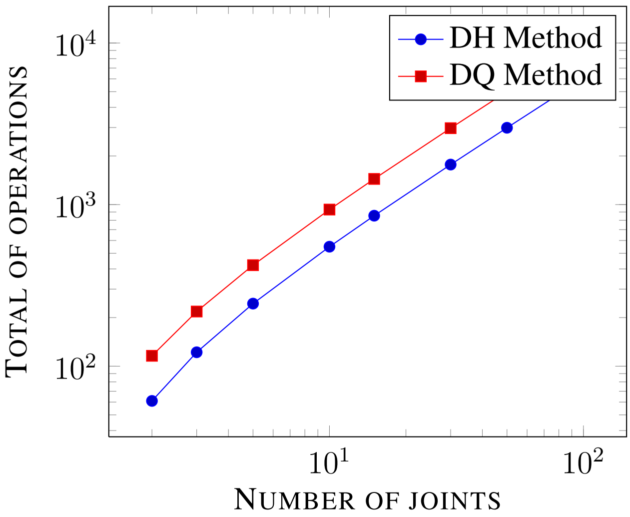

Figure 3: Computational performance of n-link robot.

6 RESULTS

A $3R$ planar robot has all joint axes parallel, then $\alpha_i = 0$ and $d_i = 0$, $i = 1, 2, 3$. Just two multiplications and two trigonometric functions defines the matrix transformation ${}^{i-1}H_i$, and all matrices the same sparse structure as in Eq. 13. the numbers are presented in table 5:

Fase	
Definition	Transformations
0+	8+
2∗	12∗

Table 5: Computational performance of DH transformations with $\alpha_i = 0$ and $d_i = 0$ - Eq. 13.

In n-link robot, $2n + 12(n-1) = 2(7n-6)$ multiplications and $8(n-1)$ additions are necessary. To the $3R$ planar robot manipulator, $n = 3$, then 30 multiplications and 16 additions are performed.

In dual quaternions perspective, if the robot has $\alpha_i = 0$ and $d_i = 0$, then each transformation requires four memory locations, two multiplications and two trigonometric function calculations to define the rigid transformation by means of dual quaternions - see Eq.15. The kinematics exibe the following numbers:

Fase	
Definition	Transformations
0+	8+
4$*$	12$*$

Table 6: Computational performance of DQ transformations with $\alpha_i = 0$ and $d_i = 0$ - Eq. 15.

To n-link robot, the global transformation requires $4n + 12(n-1) = 4(4n-3)$ multiplications and $8(n-1)$ additions. In a $3R$ planar robot manipulator, $n = 3$, then there are 36 multiplications and 16 additions. To describe the final position of the end-effector one more dual quaternion multiplication is necessary. So, 48 multiplications and 24 additions are required to complete the operation.

The numbers are summarized in Table 7.

	DH Method	DQ Method
$+$	16	24
$*$	30	48

Table 7: Computational performance of $3R$ planar robot.

7 DISCUSSION

The present work agree with Aspragathos and Dimitros (1998) in the homogeneous matrices storage cost point of view, but not for the computational cost to the end-effector posture. Also our analysis disagree with Sahul et al. (2008), that calculates the kinematic of a kind of $3R$ robot through 114 multiplications and 72 additions. To general $3R$ robot, the real numbers seems to be quite different: 79 multiplications and 43 additions (see Table 4, with $n = 3$).

It probably the differences occurred due to the rigid transformation structure. In the homogeneous transformation matrix, the sparsity is an inherent characteristic of the method and it reduces the computational consuming. So it must be considered.

To dual quaternion point of view, the authors above equates direct kinematic by an iterative process which a simple dual quaternion product, ${}^j p_i = {}^j h_i {}^i p$, plays a central role.

8 CONCLUSION

From the main references on the subject it can be stated that spinors algebra allows to model tridimensional displacements. In this context the dual quaternions algebra form a flexible algebra in the description of rigid body transformations, geometrical elements and kinematic pairs. Also from dual quaternions algebra, a cylindrical kinematic pair is exactly the dual quaternion transformation operator of the helical motion, suggesting a non-decomposition of a cylindrical joint to rotative and prismatic joints, which has been done by most researches, increasing the number of transformations, then the number of operations.

Problems arise from numeric inconsistences in the rotational matrix, where it will be necessary to renormalize its columns. Calculations are not carried out with infinite precision then the product of many orthogonal matrices may no longer be orthogonal, just as the product of many quaternions many no longer be an unit quaternion (Taylor, 1979). It is not difficult to find the nearest unit quaternion and the nearest orthogonal matrix (Taylor, 1979). The major problem in the homogeneous matrices method due to singularities which occurs in the inverse kinematic also to the nonlinear expressions to for the joints relations (Aydin and Kucuk, 2006).

The dual quaternions take advantages from homogeneous matrices in the storage point of view since homogeneous matrices requires 12 numbers to represent just six degree of freedom whereas dual quaternions just eight.

To the dual quaternions method, the results found above will not be efficient from a computational point of view - see Table 7 and Figure 3 for summary. The great advantages of the quaternions and dual quaternions due to the fact that not only points but lines and planes can be used to represents positions and orientations in that algebra (Selig, 2000b) which provides robot kinematics avoiding singularities (Oliveira et al., 2010; Sariyildiz et al., 2011).

REFERENCES

Agrawal O.P. Hamilton operators and dual-number-quaternions in spatial kinematics. *Mechanism and Machine Theory*, 22(6):569 – 575, 1987.

Akyar B. Dual quaternions in spatial kinematics in an algebraic sense. *Turk J Math*, 2008.

Aspragathos N.A. and Dimitros J.K. A comparative study of three methods for robot kinematics. *IEEE*, 28(2):135–145, 1998.

Aydin Y. and Kucuk S. Quaternion based inverse kinematics for industrial robot manipulators with euler wrist. *IEEE transactions on robotics and automation*, (1-4244-9713-4/06):581–586, 2006.

Campa R. and Camarillo K. Unit quaternions: A mathematical tool for modeling, path planning and control of robot manipulators. 2008.

Choe S.B. *Statistical analysis of orientation trajectories via quaternions with applications to human motion*. Ph.D. thesis, The University of Michigan, 2006.

Chou J.C.K. Quaternion kinematic and dynamic differential equations. *IEEE transactions on robotics and automation*, 8(1):53–64, 1992.

Craig J.J. *Introduction to Robotics: Mechanics and control*. Pearson Prentice Hall, 3rd edition, 2005. ISBN 0131236296.

Crane III C.D. and Duffy J. *Kinematic Analysis of Robot Manipulators*. Cambridge University Press, New York, NY, USA, 1998. ISBN 0521570638.

Davis T.A. *Direct methods for sparse linear systems*. Fundamentals of algorithms. SIAM, 2006. ISBN 0898716136.

Dooley J.R. and McCarthy J.M. On the geometric analysis of optimum trajectories for cooperating robots using dual quaternion coordinates. *IEEE transactions on robotics and automation*, pages 1031–1036, 1993.

Duff I.S., Erisman A.M., and Rcid J.K. *Direct methods for sparse matrices*. Oxford University press, 1989. ISBN 0198534213.

Funda J. and Paul R.P. A computational analysis os screw transformations in robotics. *IEEE Transactions on robotics and automation*, 6(3):382–388, 1990.

Gan D., Liao Q., Wei S., Dai J.S., and Qiao S. Dual quaternion-based inverse kinematics of the general spatial 7r mechanism. *Journal of Mechanical Engineering Science*, (Part C), 2008.

Ge Q.J. and McCarthy J.M. Functional constraints as algebraic manifold in a clifford algebra. *Transactions on robotics and automation*, 7(5), 1991.

Hestenes D. *New Foundations for Classical Mechanics*, volume 99. Kluwer Academic Publishers, Tempe, Arizona, 2nd edition, 1999. ISBN 0-306-471 22-1.

Horn B.K.P. Closed-form solution of absolute orientation using quaternions. *Optical Society of America*, 4(4):629–642, 1987.

Kavan L., Collins S., and O'Sullivan C. Geometric skinning with approximate dual quaternion blending. *ACM Trans. Graph.*, 27:105:1–105:23, 2008.

Lounesto P. *Clifford Algebras and Spinors*. Cambridge University Press, 2nd edition, 2001. ISBN 0521005515.

Murray R., Li Z., and Sastry S. *A Mathematical Introduction to Robotic Manipulation*. CRC Press, 1994. ISBN 0849379814.

Oliveira A.S.D., Pieri E.R.D., and Moreno U.F. Análise cinemática via quatérnios duais. *Congresso Brasileiro de Automática*, 2010.

Pennestrì E. and Valentini P. Dual quaternions as tool for rigid body motion analysis: A tutorial with an application to biomechanics. pages 1–17. Eccomas Thematic Conference, Warsaw, Poland, 2009.

Porteous I. *Clifford algebra and the classical groups*. Cambridge University Press, Cambridge, 1995. ISBN 0 521 551773.

Sahul S., Biswall B.B., and Subudhi B. A novel method for representing robot kinematics using quaternion theory. *IEEE sponsored conference on computational intelligence, control and computer vision in robotics & automation*, 2008.

Sariyildiz E., Cakiray E., and Temeltas H. A comparative study of three inverse kinematic methods of serial industrial robot manipulators in the screw theory framework. *International Journal of Advanced Robotic Systems*, 8(5):9–24, 2011.

Sariyildiz E. and Temeltas H. Solution of inverse kinematic problem for serial robot using dual quaternions and plucker coordinates. *IEEE*, pages 338–343, 2009.

Sciavicco L., Villani L., Oriolo G., and Siciliano B. *Robotics: Modelling, planning and control*. Springer-Verlag, London, 2009. ISBN 978-1-84628-641-4.

Selig J.M. *Introductory Robotics*. Pretice Hall Internatinal, 1992. ISBN 0134888758.

Selig J.M. Clifford algebra of points, lines and planes. *Robotica*, 18, 2000a.

Selig J.M. *Geometric fundamentals of robotics*. Springer-Verlag, New York, 2000b. ISBN 0-387-20874-7.

Shoham M. and Ben-Horin P. Application of grassmann-cayley algebra to geometrical interpretation of parallel robot singularities. *Int. J. Rob. Res.*, 28:127–141, 2009.

Sommer G. *Geometric computations with Clifford Algebras: Theoretical fundations and applications in computer vision and robotics*. Springer-Verlag, New York, 2001. ISBN 3-540-41198-4.

System O. Optitrack - optical motion capture system and tracking software. 2011. Acessado em: 24-10-2011.

Taylor R.H. Planning and execution of straight line manipulator trajectories. *IBM J. Res. Develop.*, 23(4):424–436, 1979.

Tewarson R.P. *Sparse matrices*, volume 99 of *Mathematics in science and engineering*. Academic press, 1973.

Tsai L.W. *Robot analysis: The mechanics of serial and parallel manipulators*. Jonh Wiley e Sons, New York, 1999.

Vince J. *Geometric Algebra for Computer Graphics*. Springer-Verlag, 2008. ISBN 978-1-84628-996-5.

Walker M.W. and Shao L. Estimating 3d location parameters using dual number quaternions. *CVGIP: Image Understanding*, 54(3):358–367, 1991.