Linear Algebra and Numerical Algorithms Using Dual Numbers

E. Pennestrì[†], **R. Stefanelli** [†]

[†] Dipartimento di Ingegneria Meccanica
Università di Roma tor Vergata
Via del Politecnico, 1 00174 Roma, Italy
e-mail: pennestri@mec.uniroma2.it

1 Introduction

A *dual number* $\widehat{a}$ is an ordered pair of real numbers (a, a^o) associated with a *real unit* +1, and the *dual unit*, or operator ε, where $\varepsilon^2 = \varepsilon^3 = \ldots = 0$. A dual number is usually denoted in the form

$$\widehat{a} = a + \varepsilon a^o \ . \tag{1}$$

The algebra of dual numbers has been originally conceived by W.K. Clifford (1873) [17], but its first applications to mechanics are due to A.P. Kotelnikov (1895)[1] and E. Study (1901) [59]. Because of conciseness of notation, dual algebra has been often used for the search of closed form solutions in the field of displacement analysis, kinematic synthesis and dynamic analysis of spatial mechanisms. One of the purposes of this investigation is the development and implementation of algorithms for the solution of numerical problems. All the algorithms are related to the solution of kinematic problems.

The algorithms can be splitted into three categories:

1. algebra and geometry of dual vectors;
2. dual version of basic algorithms of linear algebra;
3. numerical solution of linear and nonlinear equations.

Many of the discussed algorithms of linear algebra are very well known in real notation. However, for some of them, namely the QR and SVD decompositions, pseudoinverse and the computation of eigenvalues and eigenvectors, to the best of our knowledge, their dual counterpart has never been discussed. All the algorithms have been implemented using the `Ch` programming language[2] which efficiently handles operations with dual numbers. Some of the routines implemented have been tested on the kinematic analysis of the RCCC spatial linkage and their computational efficiency in the solution of the problem measured. As a result, the proposed strategy for the solution of systems of linear equations seems to offer a substantial reduction of CPU time.

2 Some contributions to mechanics based on dual numbers

Dual numbers and their algebra proved to be a powerful tool for the analysis of mechanical systems. In the following an attempt is made to list the contributions made in the different fields of mechanics by using dual numbers. No claim of exhaustiveness is made.

[1]The original paper of A.P. Kotelnikov, published in the Annals of Imperial University of Kazan (1895), is reputed to have been destroyed during the Russian revolution. [55]

[2]The `Ch` programming language has been developed by Prof. H.H. Cheng from University of California at Davis. More info about its main features are available at the web page `www.softintegration.com`.

Textbooks/monographies entirely dedicated to engineering applications of dual numbers have been authored to F.M Dimentberg [22], R. Beyer [8] and I.S. Fischer [28]. Books with chapters or sections dedicated to dual algebra and its applications have been authored by L. Brand [9], M.A. Yaglom [67], J.S. Beggs [7], J. Duffy [23], Gonzáles-Palacios and J. Angeles [35], J. McCarthy [43].

In the following noteworthy contributions of dual algebra to rigid body motion analysis, displacement analysis, kinematic synthesis and dynamic analysis of spatial mechanisms are listed:

- Rigid body motion: A.T. Yang [73], G.R. Veldkamp [64], K. Sugimoto and J. Duffy [60], K. Wohlhart [66]
- Displacement analysis of spatial mechanisms: J. Denavit [19], A.T. Yang [69, 71], A.T. Yang and F. Freudenstein [68], L.M. Hsia and A.T. Yang [38], J. Wittenburg [65], H.H. Cheng [13, 14, 15, 11, 12], I.S. Fischer [27, 29, 32, 28, 25, 31, 30], A.Perez and J.M. McCarthy [52],....
- Robotics: You Liang and J.Y.S. Luh.[36], G.R. Pennock and A.T. Yang [50], J. McCarthy [42], M.A. Gonzáles-Palacios et al. [34], G.R. Pennock and K.G. Mattson [48], K. Daniilidis [18];
- Surface shape analysis and computer graphics: P. Azariadis, N. Aspragathos [3], J.A. Schaaf, B. Ravani [56]
- Human body motion analysis: K.K. Teu and W. Kim [61]
- Kinematic synthesis: M.A. Gonzáles-Palacios and J. Angeles [35], J. McCarthy [43], Y.M. Moon and S. Kota [44];
- Dynamic analysis: A.T. Yang [70, 72], C. Bagci [4, 5], M. Keler [39], A.T. Yang and G. Pennock [49], I.S. Fischer and F. Freudenstein [26], C.K. Chen and F. Freudenstein [16], S.K. Agrawal [1],M. Shoham and V. Brodsky [57], E. Fasse [24], E. Pennestrì et al. [46, 45].

3 Outlines of motor and screw theory

With reference to Figure 1, a *line vector* is a vector $\overrightarrow{W}$ bound to a line L and is completely defined by the two vectors $\overrightarrow{W}$ and $\overrightarrow{V} = \overrightarrow{OP} \times \overrightarrow{W}$. The components of $\overrightarrow{W}$ and $\overrightarrow{V}$ form the Plücker or line coordinates. These obviously satisfy the condition $\overrightarrow{W} \cdot \overrightarrow{V} = 0$. The vectors $\overrightarrow{W}$ and $\overrightarrow{V}$ can be amalgamated, using matrix notation, into a dual vector

$$\left\{\widehat{W}\right\} = \{W\} + \varepsilon \{V\} \tag{2}$$

General line vectors depend upon five independent scalars. Unit line vectors $\left\{\widehat{U}\right\} = \{U\} + \varepsilon \{U^o\}$ satisfy the conditions $\left|\overrightarrow{U}\right| = 1$ and $\{U\}^T \{U^o\} = 0$.

A dual vector is named with the combined words term *motor* if both the following conditions apply: a) the vector $\overrightarrow{W}$ is independent of the choice of O; b) when the new origin is displaced in P, its moment vector $\overrightarrow{V}$ changes accordingly with the expression $\overrightarrow{V}_P = \overrightarrow{V} + \overrightarrow{PO} \times \overrightarrow{W}$. A motor $\left\{\widehat{M}\right\}$ can be interpreted as a dual multiple of a unit line vector [22, 9]

$$\left\{\widehat{M}\right\} = \{M\} + \varepsilon \{M^o\} = (a + \varepsilon a^o)\left(\{U\} + \varepsilon \{U^o\}\right) \tag{3}$$

with

$$\{M\} = a\,\{U\}\ , \tag{4a}$$
$$\{M^o\} = a\,\{U^o\} + a^o\,\{U\}\ . \tag{4b}$$

It is also easy to verify that the scalars $\overrightarrow{M} \cdot \overrightarrow{M}$ and $\overrightarrow{M} \cdot \overrightarrow{M}^o$ are invariant with respect to the origin of coordinates. In particular, from (4) one obtains

$$a = \{M\}^T\{M\}\ , \quad aa^o = \{M\}^T\{M^o\} \tag{5}$$

and the pitch μ of the motor

$$\mu = \frac{a^o}{a} = \frac{\{M\}^T\{M^o\}}{\{M\}^T\{M\}} \tag{6}$$

can then be introduced.

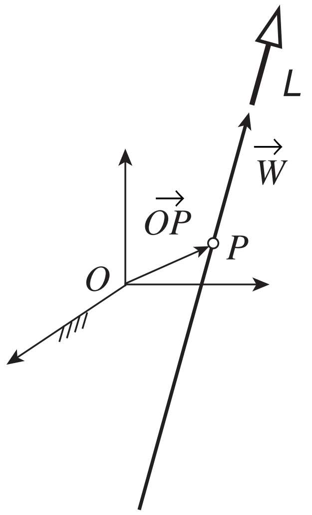

Figure 1: Line vector

Table 1: Classification of motors. (Adapted from [9]).

Motor	Conditions
Screw	$\vec{M} \neq 0$, $\vec{M} \cdot \vec{M}^o \neq 0 \Rightarrow a \neq 0$, $a^o \neq 0$
Line vector[1]	$\vec{M} \neq 0$, $\vec{M} \cdot \vec{M}^o = 0 \Rightarrow a \neq 0$, $a^o = 0$
Couple [2]	$\vec{M} = 0$, $\vec{M} \cdot \vec{M}^o = 0 \Rightarrow a = 0$, $a^o \neq 0$
Zero	$\vec{M} = 0$, $\vec{M} \cdot \vec{M}^o = 0 \Rightarrow a = 0$, $a^o = 0$

[1] A line vector is a screw of zero pitch.
[2] A couple is a screw of infinite pitch with an axis of given direction, but arbitrary position in space.

According to Table 1

From the geometric point of view:

> A screw is a straight line with which a definite linear magnitude termed the pitch associated.

In kinematics and dynamics we are interested at particular types of screws.

An infinitesimal rigid body motion can always be reduced to a twist motion about a screw axis. After this property, discovered by Giulio Giuseppe Mozzi del Garbo (1763), the following definition can be introduced:

> A body is said to receive a *twist about a screw* when it is rotated uniformly about the screw, while it is translated uniformly parallel to the screw through a distance equal to the product of the pitch and the circular measure of angle of rotation. [6]

A system of forces and torques acting on a rigid body can be always replaced by a force and a couple in a plane perpendicular to such force. After this property, discovered by L. Poinsot (1804), the following definition can be introduced:

> A *wrench on a screw* is a force directed along the screw and a couple in the plane perpendicular to the screw, the moment of the couple being equal to the product of the force and the pitch of the screw [6].

In the following sections we will consider some kinematics problems solved within the framework of dual algebra.

4 Dual numbers

Dual numbers can be represented as follows:

- Gaussian representation: $\widehat{a} \equiv a + \varepsilon a^o$.
- Polar representation: $\widehat{a} \equiv \rho\,(1 + \varepsilon t)$, where $\rho = a$ and $t = a^o/a$.
- Exponential representation: $\widehat{a} \equiv \rho \mathrm{e}^{\varepsilon t}$, where $\rho = a$, $t = a^o/a$ and $\mathrm{e}^{\varepsilon t} = 1 + \varepsilon t$.

The adoption of one representation instead of another depends on the context.

5 Dual functions

A function F of a dual variable $\widehat{x} = x + \varepsilon x^o$ can be represented in the form $F\,(\widehat{x}) = f\,(x, x^o) + \varepsilon g\,(x, x^o)$, where f and g are real functions of real variables x and x^o. The necessary and sufficient conditions in order F be analytic are [22]

$$\frac{\partial f}{\partial x^o} = 0\,, \qquad \frac{\partial f}{\partial x} = \frac{\partial g}{\partial x^o}\,. \tag{7}$$

From these follows

$$f\,(\widehat{x}) = f\,(x + \varepsilon x^o) = f(x) + x^o \frac{\partial f}{\partial x}\,. \tag{8}$$

6 Comparison between basic operations and functions with real and dual numbers

Table 2 summarizes the definitions of basic algebraic operations and functions with dual numbers and trigonometric functions. In the last three columns of this table the number of real operations equivalent to a function evaluation with dual numbers.

7 Algebra of dual vectors and matrices

7.1 Scalar and cross products of dual vectors

With reference to Figure 2,

$$\widehat{A} = \vec{a} + \varepsilon\,(\vec{r}_1 \times \vec{a})\ ,$$
$$\widehat{B} = \vec{b} + \varepsilon\left(\vec{r}_2 \times \vec{b}\right)\ ,$$

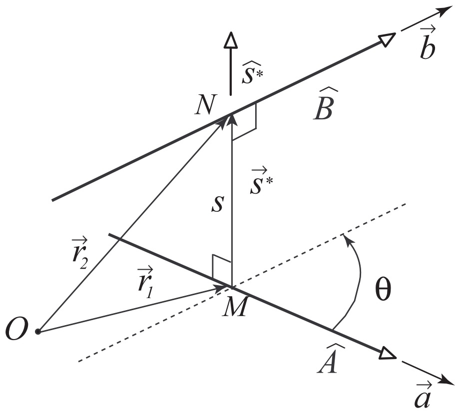

Figure 2: Product of dual vectors:Nomenclature

be two dual vectors representing two distinct line vectors and let $\vec{s}^*$ the direction versor of the minimum distance between these line vectors directed from $\vec{a}$ to $\vec{b}$.

In such a context, it is necessary to introduce the concept of dual angle [59] $\widehat{\theta} = \theta + \varepsilon s$ as a variable required to characterize the relative position and orientation of line vectors $\widehat{A}$ and $\widehat{B}$. The angle θ is measured counterclockwise about $\vec{s}^*$.

Table 2: Computational cost of operations with dual numbers

Dual operation	**Mathematical expression**	Mult. and Div.	Sums	Trig. evaluations
Sum	$\widehat{a} \pm \widehat{b} = (a \pm b) + \varepsilon (a^o \pm b^o)$	-	2	-
Product	$\widehat{a}\widehat{b} = ab + \varepsilon (a^o b + ab^o)$	3	1	-
Division[1]	$\frac{\widehat{a}}{\widehat{b}} = \frac{a}{b} + \varepsilon \frac{a^o b - ab^o}{b^2}$	5	1	-
Vector scaling	$\widehat{a}\{\widehat{n}\}$	9	3	-
Dot product[2]	$\left\{\widehat{A}\right\}^T \left\{\widehat{B}\right\} = \{A\}^T \{B\} + \varepsilon \left(\{A^o\}^T \{B\} + \{B^o\}^T \{A\}\right)$	9	7	-
Cross product	$\left[\widetilde{\widehat{A}}\right] \left\{\widehat{B}\right\} = \left[\widetilde{A}\right] \{B\} + \varepsilon \left(\{A\}^T \{B^o\} + \{A^o\}^T \{B\}\right)$	18	12	-
Dual sin	$\sin\widehat{\theta} = \sin\theta + \varepsilon d \cos\theta$	1	0	2
Dual cos	$\sin\widehat{\theta} = \sin\theta + \varepsilon d \cos\theta$	1	0	2
Dual tan	$\tan\widehat{\theta} = \tan\theta + \varepsilon \frac{d}{\cos^2\theta}$	2	0	2
Dual tan	$\tan\widehat{\theta} = \tan\theta + \varepsilon \frac{d}{\cos^2\theta}$	2	0	2
Dual asin	$\arcsin\widehat{a} = \arcsin a + \varepsilon \frac{a^o}{\sqrt{1-a^2}}$	3[3]	1	1
Dual acos	$\arccos\widehat{a} = \arccos a - \varepsilon \frac{a^o}{\sqrt{1-a^2}}$	3[3]	1	1
Dual atan	$\arctan\widehat{a} = \arctan a + \varepsilon \frac{a^o}{1+a^2}$	2	1	1

[1] Division by a pure dual number εb^o is not defined.
[2] We assume vectors of 3 elements.
[3] The computational cost of a square root operation on real numbers not considered in this table is usually considered equivalent to about 8 multiplications/divisions.

The scalar and cross products of two dual vectors are sepectively defined as follows [22]

$$
\begin{aligned}
\widehat{A} \cdot \widehat{B} &= \vec{a} \cdot \vec{b} + \varepsilon \left[\vec{a} \cdot \left(\vec{r}_2 \times \vec{b} \right) + \vec{b} \cdot \left(\vec{r}_1 \times \vec{a} \right) \right] \\
&= \vec{a} \cdot \vec{b} + \varepsilon \left[\left(\vec{r}_1 - \vec{r}_2 \right) \cdot \left(\vec{a} \times \vec{b} \right) \right] \\
&= ab \cos\theta - \varepsilon \left[(s\vec{s}^{\star}) \cdot (ab \sin\theta \vec{s}^{\star}) \right] \\
&= ab \left[\cos\theta - \varepsilon s \sin\theta \right] = ab \cos\widehat{\theta} \, . \qquad (9)
\end{aligned}
$$

$$
\begin{aligned}
\widehat{A} \times \widehat{B} &= \vec{a} \times \vec{b} + \varepsilon \left[\vec{a} \times \left(\vec{r}_2 \times \vec{b} \right) + \left(\vec{r}_1 \times \vec{a} \right) \times \vec{b} \right] \\
&= \vec{a} \times \vec{b} + \varepsilon \left[\left(\vec{a} \cdot \vec{b} \right) \left(\vec{r}_2 - \vec{r}_1 \right) + \vec{r}_1 \times \left(\vec{a} \times \vec{b} \right) \right] \qquad (10) \\
&= ab \left\{ \vec{s}^{\star} \sin\theta + \varepsilon \left[s \cos\theta \vec{s}^{\star} + \sin\theta \left(\vec{r}_1 \times \vec{s}^{\star} \right) \right] \right\} \\
&= ab\widehat{S}^{\star} \left(\sin\theta + \varepsilon s \cos\theta \right) = ab\widehat{S}^{\star} \sin\widehat{\theta} \, , \qquad (11)
\end{aligned}
$$

Table 3 summarizes the result of different dual vectors products for different cases of relative position of line vectors.

Table 3: Noteworthy cases of dual vectors products (Adapted from [22])

Line vector	$\widehat{A} \cdot \widehat{B}$	$\widehat{A} \times \widehat{B}$
Skew	$ab \cos\widehat{\theta}$	$ab\widehat{S}^{\star} \sin\widehat{\theta}$
Incident ($s = 0$)	$ab \cos\theta$	$ab\widehat{S}^{\star}$
Parallel ($\theta = 0$)	ab	$\varepsilon ab\vec{s}^{\star}$
Coaxial ($\theta = s = 0$)	ab	0

7.2 Dual angle between line vectors

The dual angle between the two line vectors

$$
\widehat{A}_i = \vec{a}_i + \varepsilon \left(\vec{s}_i \times \vec{a}_i \right) \, . \quad (i = 1, 2)
$$

must be computed. The computational steps are described in the following and are justified by the geometry depicted in Figure 3.

1. Compute the dual vectors

$$
\widehat{E}_i = \frac{\widehat{A}_i}{\left\| \widehat{A}_i \right\|} \quad (i = 1, 2) \qquad (12)
$$

2. Compute their cross product

$$
\widehat{E}_3 = \frac{\widehat{E}_1 \times \widehat{E}_2}{\left\| \widehat{E}_1 \times \widehat{E}_2 \right\|} \, . \qquad (13)
$$

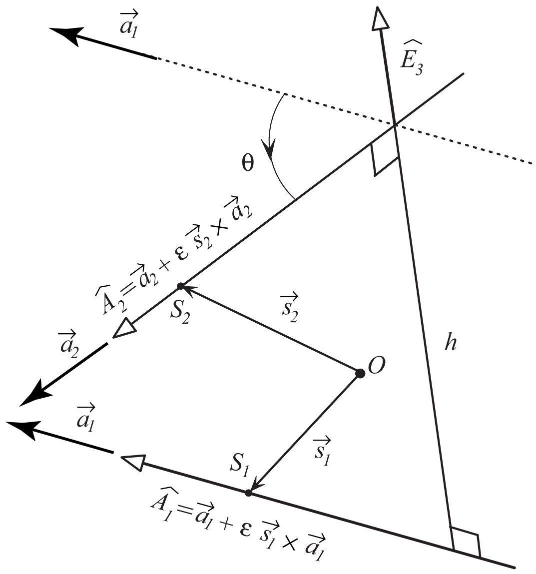

Figure 3: Computation of dual angle between two line vectors

3. Compute cosine and sine of dual angle $\widehat{\theta}$

$$\cos\widehat{\theta} = \widehat{E}_1 \cdot \widehat{E}_2 \ , \tag{14a}$$

$$\sin\widehat{\theta} = \widehat{E}_1 \times \widehat{E}_2 \cdot \widehat{E}_3 \ . \tag{14b}$$

4. Compute dual angle

$$\widehat{\theta} = \text{atan2}\left(\sin\widehat{\theta}, \cos\widehat{\theta}\right) = \theta + \varepsilon h \ . \tag{15}$$

The procedure is not valid if line vectors are parallel. In this case, there is an infinite set of dual vectors $\widehat{E}_3$.

7.3 Dual vector perpendicular to two dual vectors

In many applications it is necessary the definition of line vector $\widehat{E}_3$, which intersects orthogonally in P_i the line vectors $\widehat{A}_i = \vec{a}_i + \varepsilon\,(\vec{s}_i \times \vec{a}_i)$, $(i = 1, 2)$. In order to have a unique solution, it is required that the line vectors are not coincident or parallel.

The algorithm herein described is due to M. A. Gonzáles-Palacios and J. Angeles [35].

With reference to Figure 4, let

- $\widehat{Q}_i = \vec{q}_i + \varepsilon 0$ be the minimum distance line vectors of line vectors $\widehat{A}_i$ from origin O $(i = 1, 2)$;
- Q_i be the intersections of orthogonal lines to $\widehat{A}_i$ $(i = 1, 2)$ from O;

and

$$h_i = Q_i P_i \tag{16}$$

the minimum distance between $\widehat{Q}_i$ and $\widehat{E}_3$. Observing the geometry of Figure 4 and taking into account the discussion of the previous algorithm, the following computational steps can be stated:

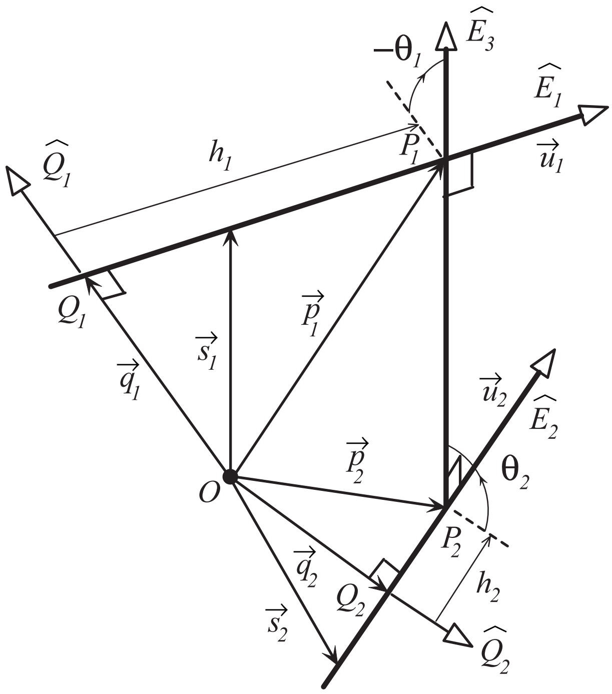

Figure 4: Nomenclature. Computation of points P_1 e P_2 on the minimum distance line

1. Compute dual vectors

$$\left\{\widehat{E}_i\right\} = \frac{\left\{\widehat{A}_i\right\}}{\left\|\widehat{A}_i\right\|} = \vec{u}_i + \varepsilon\left(\vec{s}_i \times \vec{u}_i\right) \quad (i = 1, 2)\ . \tag{17}$$

where $\vec{u}_i = veca_i / \|\vec{a}_i\|$.

2. Compute the dual vector orthogonal to the previous ones

$$\widehat{E}_3 = \frac{\widehat{E}_1 \times \widehat{E}_2}{\left\|\widehat{E}_1 \times \widehat{E}_2\right\|}\ . \tag{18}$$

3. Compute vectors $\vec{q}_i$

$$\vec{q}_i = \vec{u}_i \times (\vec{s}_i \times \vec{u}_i) = \vec{s}_i - (\vec{u}_i \cdot \vec{s}_i)\,\vec{u}_i\ . \tag{19}$$

4. If $\|\vec{q}_i\| \neq 0$, then let

$$\widehat{Q}_i = \frac{\vec{q}_i}{\|\vec{q}_i\|} + \varepsilon 0\ . \tag{20}$$

5. Compute the dual angles $\widehat{\theta}_i$ formed by dual vectors $\widehat{Q}_i$ and $\widehat{E}_3$ by means of the relationships[3]

$$\cos\widehat{\theta}_i = \widehat{Q}_i \cdot \widehat{E}_3\ , \tag{21a}$$

$$\sin\widehat{\theta}_i = \widehat{Q}_i \times \widehat{E}_3 \cdot \widehat{E}_i\ ; \tag{21b}$$

$$\widehat{\theta}_i = \operatorname{atan2}\left(\sin\widehat{\theta}_i, \cos\widehat{\theta}_i\right)\ . \tag{21c}$$

[3]Reference [35] does not use the `atan2` function.

6. The distances h_i can be now computed from

$$\widehat{\theta}_i = \theta_i + \varepsilon h_i \; . \tag{22}$$

If $\|\vec{q}_i\| = 0$, then $h_i = 0$.

7. Compute coordinates of P_i:

$$\vec{p}_i = \vec{q}_i + h_i \vec{u}_i \; . \tag{23}$$

The procedure is not valid if dual vectors are parallel.

7.4 Sum of two dual vectors

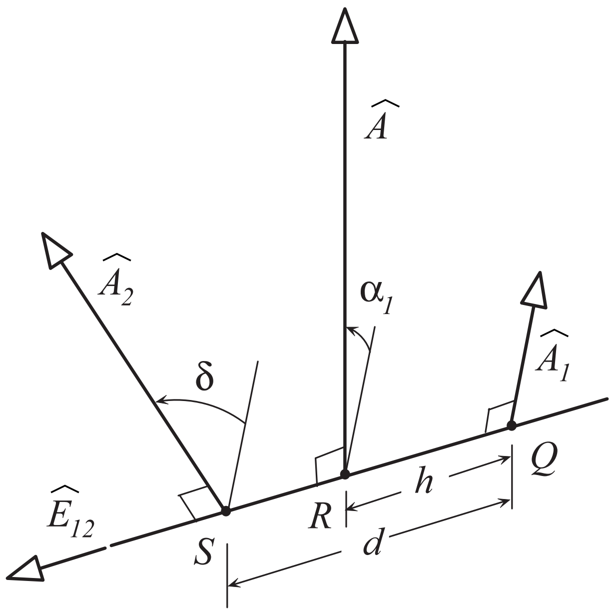

Figure 5: Sum of dual vectors

With reference to the geometry of Figure 5, we wish compute the sum

$$\widehat{A} = \widehat{A}_1 + \widehat{A}_2 \tag{24}$$

One can observe that the direction of $\widehat{A}$ is obtained by prescribing a screw motion to $\widehat{A}_1$ defined by the screw axis $\widehat{E}_{12}$ and dual angle $\widehat{\alpha_1}$. On the basis of this observation, the following algorithm can be stated:

1. Compute the dual vectors

$$\widehat{E}_1 = \frac{\widehat{A}_1}{\left\|\widehat{A}_1\right\|} \; ,$$

$$\widehat{E}_2 = \frac{\widehat{A}_2}{\left\|\widehat{A}_2\right\|} \; .$$

2. Compute the dual angle $\widehat{\theta}$ and the dual vector $\widehat{E}_{12}$ perpendicular to both $\widehat{A}_1$ and $\widehat{A}_2$ (see previous section).

3. Compute the module of the dual vector sum

$$A = \sqrt{\widehat{A}_1 \cdot \widehat{A}_1 + \widehat{A}_2 \cdot \widehat{A}_2 + 2\widehat{A}_1 \cdot \widehat{A}_2}$$

4. Compute sine and cosine of dual angle $\widehat{\alpha}_1$

$$\begin{aligned}
\sin\widehat{\alpha}_1 &= \frac{\sin\widehat{\theta}\sqrt{\widehat{A}_2 \cdot \widehat{A}_2}}{A} \ , \\
\cos\widehat{\alpha}_1 &= \frac{\sqrt{\widehat{A}_1 \cdot \widehat{A}_1}}{A} + \sin\widehat{\alpha}_1 \frac{\cos\widehat{\theta}}{\sin\widehat{\theta}} \ , \\
\widehat{\alpha}_1 &= \operatorname{atan2}\left(\sin\widehat{\alpha}_1, \cos\widehat{\alpha}_1\right)
\end{aligned}$$

5. Let

$$\widehat{t} = \tan\frac{\widehat{\alpha}_1}{2}$$

and compute

$$\widehat{T} = \widehat{t}\widehat{E}_{12} \ . \tag{25}$$

6. Apply formula

$$\widehat{E} = \widehat{E}_1 + \frac{2\widehat{T}}{1+\widehat{t}^2} \times \left(\widehat{E}_1 + \widehat{T} \times \widehat{E}_1\right)$$

for the definition of $\widehat{E}_1$ after the screw motion

7. Finally compute

$$\widehat{A} = A\widehat{E} \ .$$

Example*:* If we assume

$$\widehat{A}_1 = \{\begin{array}{ccc} 0 & 0 & 1 \end{array}\}^T \ , \quad \widehat{A}_2 = \{\begin{array}{ccc} 0 & 1 & \varepsilon \end{array}\}^T \ ,$$

the previous algorithm give the following numerical values

$$\begin{aligned}
\widehat{\theta} &= \frac{\pi}{2} - \varepsilon \ , \\
\widehat{E}_{12} &= \{\begin{array}{ccc} -1 & 0 & 0 \end{array}\}^T \ , \\
\widehat{\alpha}_1 &= \frac{\pi}{4} - \frac{\varepsilon}{2} \ , \\
\widehat{A} &= \{\begin{array}{ccc} 0 & 1 & 1+\varepsilon \end{array}\}^T \ .
\end{aligned}$$

The loci formed by all possible $\widehat{A}$ form a ruled surface named cylindroid. Ready to use computer-aided solutions of this problem have been also proposed by A.Perez [53].

7.5 Product of two dual matrices

Assuming that $\left[\widehat{A}\right] = [A] + \varepsilon\,[A^o]$ and $\left[\widehat{B}\right] = [B] + \varepsilon\,[B^o]$ have the appropriate dimensions, then their dual product is defined as follows:

$$\left[\widehat{A}\right]\left[\widehat{B}\right] = [A]\,[B] + \varepsilon\left([A]\,[B^o] + [A^o]\,[B]\right) \tag{26}$$

7.6 Inverse of a dual matrix

The inverse of a square dual matrix is defined as

$$\left[\widehat{A}\right]\left[\widehat{A}\right]^{-1} = [I] \ . \tag{27}$$

By letting $\left[\widehat{B}\right] = \left[\widehat{A}\right]^{-1}$, from (26) one obtains [36, 2]

$$\left[\widehat{A}\right]^{-1} = [A]^{-1} - \varepsilon\left([A]^{-1}[A^o][A]^{-1}\right) \ . \tag{28}$$

7.7 Solution of a system of dual linear equations

Let us denote with

$$\left[\widehat{A}\right]\{\widehat{x}\} = \left\{\widehat{b}\right\} \tag{29}$$

a system of linear dual equations [12] where $\left[\widehat{A}\right] = [A] + \varepsilon[A^o]$ and $\left\{\widehat{b}\right\} = \{b\} + \varepsilon\{b^o\}$. Assuming $[A]$ nonsingular. The solution $\{\widehat{x}\} = \{x\} + \varepsilon\{x^o\}$ is computed by solving the systems

$$[A]\{x\} = \{b\} \ , \tag{30a}$$

$$[A]\{x^o\} = \{b^o\} - [A^o]\{x\} \ . \tag{30b}$$

To improve the overall computational efficiency it is convenient to factor matrix $[A]$ only once. Thus, we suggest to proceed as follows:

I) Apply QR decomposition to matrix $[A]$ so that $[A] = [Q][R]$.

II) Solve system $[R]\{x\} = [Q]^T\{b\}$.

III) Solve system $[R]\{x^o\} = [Q]^T(\{b^o\} - [A^o]\{x\})$.

IV) Form the dual vector $\{\widehat{x}\} = \{x\} + \varepsilon\{x^o\}$

Since $[R]$ is an upper triangular matrix, at steps II and III only simple procedures of back substitution are executed.

7.8 QR decomposition of a dual matrix

A dual matrix $\left[\widehat{A}\right]$ can be decomposed as follows

$$\left[\widehat{A}\right] = \left[\widehat{Q}\right]\left[\widehat{R}\right] \tag{31}$$

where $\left[\widehat{Q}\right] = [Q] + \varepsilon[Q^o]$ is an orthogonal matrix and $\left[\widehat{R}\right] = [R] + \varepsilon[R^o]$ is an upper triangular matrix. To obtain the QR decomposition one can adapt to dual numbers the modified Gram-Schmidt orthogonalization procedure [33].

Example: The matrix

$$\left[\widehat{A}\right] = \begin{bmatrix} 1+\varepsilon & 2+\varepsilon 3 \\ 3+\varepsilon 9 & 3+\varepsilon \end{bmatrix}$$

can be decomposed in the following matrices[4]:

$$\left[\widehat{Q}\right] = \begin{bmatrix} 0.316-\varepsilon 0.569 & 0.949+\varepsilon 0.190 \\ 0.949+\varepsilon 0.190 & -0.316+\varepsilon 0.569 \end{bmatrix} , \quad \left[\widehat{R}\right] = \begin{bmatrix} 3.162+\varepsilon 8.854 & 3.478+\varepsilon 1.328 \\ 0.000+\varepsilon 0.000 & 0.948+\varepsilon 4.617 \end{bmatrix} .$$

[4]Numerical results are displayed with three decimal digits only.

7.9 Pseudoinverse of a dual matrix

Let $\left[\widehat{A}\right]$ be a matrix with m rows and n columns. Its dual Moore-Penrose pseudoinverse is defined as follows:

$$\left[\widehat{A}\right]^{+} = \begin{cases} \left(\left[\widehat{A}\right]^{T}\left[\widehat{A}\right]\right)^{-1}\left[\widehat{A}\right]^{T} & \text{if } m \geq n \\ \left[\widehat{A}\right]^{T}\left(\left[\widehat{A}\right]\left[\widehat{A}\right]^{T}\right)^{-1} & \text{if } m < n \end{cases} \tag{32}$$

If $m > n$, then the left pseudoinverse exists such that $\left[\widehat{A}\right]^{+}\left[\widehat{A}\right] = [I]$. If $m < n$ then the right pseudoinverse exists such that $\left[\widehat{A}\right]\left[\widehat{A}\right]^{+} = [I]$. Adopting a reasoning similar to the one used for the definition of the inverse matrix, one can demonstrate that

$$\left[\widehat{A}\right]^{+} = [A]^{+} - \varepsilon\left([A]^{+}[A^{o}][A]^{+}\right) \tag{33}$$

The definition of the Moore-Penrose matrix is often associated with the least squares solution of a linear system, such as (29), but with the number m of equations different from the number n of unknowns. It is well known that $[A]^{T}[A]$ (or $[A][A]^{T}$) may not have an inverse and, even when it is invertible, one usually obtains large numerical errors from using directly (32). Thus, to obtain significant numerical answers one must use more sophisticated techniques. A summary of efficient algorithms for computing the Moore-Penrose pseudoinverse is reported in [21, 20]. Alternatively, the pseudoinverse matrix can be directly computed from the SVD decomposition.

Example: The Moore-Penrose pseudoinverses of

$$\left[\widehat{A}_1\right] = \begin{bmatrix} 1+\varepsilon 4 & 3+\varepsilon 0 \\ 9+\varepsilon 2 & 22+\varepsilon 4 \\ 4+\varepsilon 4 & 4+\varepsilon 1 \end{bmatrix}, \quad \left[\widehat{A}_2\right] = \begin{bmatrix} 1+\varepsilon 4 & 3+\varepsilon 0 & 4+\varepsilon \\ 9+\varepsilon 2 & 22+\varepsilon 4 & 4+\varepsilon 4 \end{bmatrix}.$$

are, respectively,

$$\left[\widehat{A}_1\right]^{+} = \begin{bmatrix} -0.051+\varepsilon 0.064 & -0.069+\varepsilon 0.082 & 0.418-\varepsilon 0.533 \\ 0.028-\varepsilon 0.025 & 0.073-\varepsilon 0.038 & -0.170+\varepsilon 0.199 \end{bmatrix}, \quad \left[\widehat{A}_2\right]^{+} = \begin{bmatrix} -0.035\varepsilon-0.014 & 0.021+\varepsilon 0.000 \\ -0.038-\varepsilon 0.035 & 0.044-\varepsilon 0.001 \\ 0.287-\varepsilon 0.007 & -0.038-\varepsilon 0.011 \end{bmatrix}.$$

7.10 Eigenvalues and eigenvectors of a dual matrix

Let us assume that $\left[\widehat{A}\right] = [A] + \varepsilon[A^{o}]$, with $[A]$ and $[A^{o}]$ non singular square matrices. The search of eigenvalues $\widehat{\lambda} = \lambda + \varepsilon\lambda^{o}$ and eigenvectors $\{\widehat{v}\} = \{v\} + \varepsilon\{v^{o}\}$ requires the solution of the following equation

$$\left[\widehat{A}\right]\{\widehat{v}\} = \widehat{\lambda}\{\widehat{v}\}\ . \tag{34}$$

Expanding the previous equation and splitting into the primary and dual parts we obtain

$$[A - I\lambda]\{v\} = \{0\}\ , \tag{35a}$$

$$\left[\begin{array}{c:c} A - I\lambda & -v \\ \hdashline v^{T} & 0 \end{array}\right]\left\{\begin{array}{c} v^{o} \\ \hdashline \lambda^{o} \end{array}\right\} = \left\{\begin{array}{c} -[A^{o}]\{v\} \\ \hdashline 0 \end{array}\right\}\ , \tag{35b}$$

after the normalization condition $\{\widehat{v}\}^{T}\{\widehat{v}\} = 1$ is imposed.

7.11 SVD decomposition of a dual matrix

The dual $m \times n$ matrix $\left[\widehat{A}\right]$ can be decomposed as follows

$$\left[\widehat{A}\right] = \left[\widehat{U}\right]\left[\widehat{\Lambda}\right]\left[\widehat{V}\right]^T , \tag{36}$$

where $\left[\widehat{U}\right]^T\left[\widehat{U}\right] = \left[\widehat{V}\right]^T\left[\widehat{V}\right] = \left[\widehat{U}\right]\left[\widehat{U}\right]^T = [I]$ and

$$\left[\widehat{\Lambda}\right] = \begin{bmatrix} \widehat{\lambda}_1 & 0 & 0 & 0 \\ 0 & \widehat{\lambda}_2 & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & \widehat{\lambda}_p \end{bmatrix} , \tag{37}$$

with $0 \leq \lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_p$, $p = \min(m, n)$. There are many computationally efficient procedures to obtain an SVD decomposition of a matrix [33]. In the case of this investigation, the following has been implemented[5]:

I) Solve the eigenvalue problem

$$\left[\widehat{A}\right]\left[\widehat{A}\right]^T \{\widehat{u}\} = \widehat{\lambda}^2 \{\widehat{u}\} . \tag{38}$$

II) For each eigenvalue $\widehat{\lambda}_i$ and eigenvector $\{\widehat{u}_i\}$ $i = 1, 2, \ldots, m$ compute

$$\{\widehat{v}_i\} = \frac{1}{\widehat{\lambda}_i}\left[\widehat{A}\right]^T \{\widehat{u}_i\} \tag{39}$$

III) Form matrices $\left[\widehat{V}\right]$ and $\left[\widehat{U}\right]$ using $\{\widehat{u}_i\}$ and $\{\widehat{v}_i\}$ as columns, respectively.

Example:

Given the matrix

$$\left[\widehat{A}\right] = \begin{bmatrix} 1+\varepsilon & 2+\varepsilon 5 \\ 2+\varepsilon 0 & 1+\varepsilon \\ 6+\varepsilon 4 & 8+\varepsilon 2 \end{bmatrix}$$

the application of the previous computational procedure gives:

$$\left[\widehat{U}\right] = \begin{bmatrix} -0.690-\varepsilon 1.128 & 0.283+\varepsilon 0.288 & 0 \\ 0.721-\varepsilon 1.148 & 0.187-\varepsilon 0.008 & 0 \\ 0.064+\varepsilon 0.791 & 0.941-\varepsilon 0.085 & 0 \end{bmatrix} , \quad \left[\widehat{V}\right] = \begin{bmatrix} 0.805-\varepsilon 0.050 & 0.593+\varepsilon 0.068 & 0 \\ -0.593-\varepsilon 0.068 & 0.805-\varepsilon 0.050 & 0 \end{bmatrix} ,$$

and

$$\left[\widehat{\Lambda}\right] = \begin{bmatrix} 1.411+\varepsilon 1.191 & 0 & 0 \\ 0 & 10.631+\varepsilon 5.204 & 0 \\ 0 & 0 & 0 \end{bmatrix} .$$

7.12 Least squares solution of a dual system of linear equations

In kinematics is often required the solution of systems with redundant dual equations. In this case, the least squares solution can be applied. Making use of the SVD decomposition such solution is readily available [33].

[5] If $m > n$ one should apply the procedure to $\left[\widehat{A}\right]^T$ and the matrices $\left[\widehat{U}\right]$ and $\left[\widehat{V}\right]$ reciprocally exchanged.

8 The Principle of Transference

The Principle of Transference proved to be a powerful tool for kinematic analysis of spatial linkages. The Principle has been declared by A.P Kotelnikov for the first time, but the correspondence between equivalent spherical and spatial configurations is due to V.V. Dobrovolski (1947).

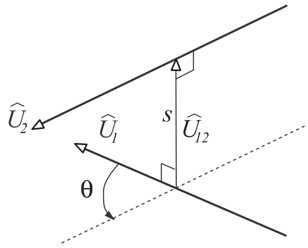

Figure 6: Nomenclature

A thoughtful discussion of the Principle of Transference is given by J. Rooney [55], L.M. Hsia and A.T. Yang [38].

The Principle of Transference can be stated as follows [22, 55, 38]:

> All valid laws and formulae relating to a system of intersecting unit line vectors (and hence involving real variables) are equally valid when applied to an equivalent system of skew vectors, if each variable a, in the original formulae is replaced by the corresponding dual variable $\widehat{a} = a + \varepsilon a^o$.

In virtue of the Principle of Transference, formulas for the composition of spherical motions can be extended to the general helicoidal motion case by simply substituting the angle of rotation θ with the *dual angle* $\widehat{\theta} = \theta + \varepsilon s$, where s is the displacem

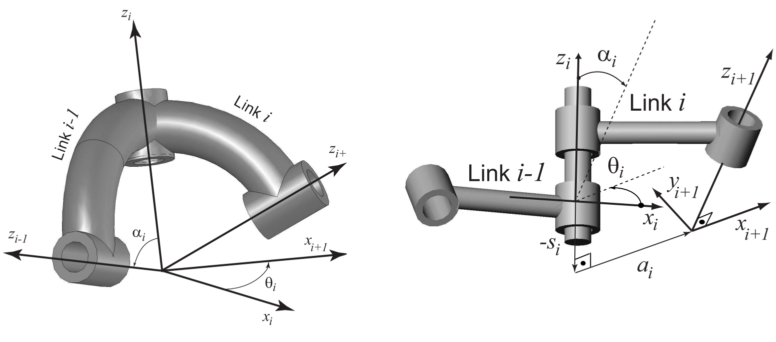

Figure 7: Adjacent links in a spherical mechanism (left) and spatial mechanism (right)

Similarly, with reference to Figure 8, the formula obtained for the kinematic analysis of spherical mechanisms can be extended to the general spatial mechanism with skew links by substituting:

- the relative position angles θ_i between adjacent links with the dual angles $\widehat{\theta}_i = \theta_i + \varepsilon s_i$;

- the angular length α_i with the dual angle $\widehat{\alpha}_i = \alpha_i + \varepsilon a_i$.

It is worth observing that θ_i, s_i, α_i and a_i form a set of Denavit-Hartenberg (DH) parameters. The dual transform matrix from coordinates system $O_{i+1} - X_{i+1}Y_{i+1}Z_{i+1}$ to $O_i - X_iY_iZ_i$ has the following expression [12]

$$\left[\widehat{A}_i\right] = \left[\widehat{\Theta}_i\right]\left[\widehat{\Lambda}_i\right] \tag{40}$$

where

$$\left[\widehat{\Theta}_i\right] = \begin{bmatrix} \cos\widehat{\theta}_i & -\sin\widehat{\theta}_i & 0 \\ \sin\widehat{\theta}_i & \cos\widehat{\theta}_i & 0 \\ 0 & 0 & 1 \end{bmatrix} , \quad \left[\widehat{\Lambda}_i\right] = \begin{bmatrix} 0 & 0 & 1 \\ \cos\widehat{\alpha}_i & -\sin\widehat{\alpha}_i & 0 \\ \sin\widehat{\alpha}_i & \cos\widehat{\alpha}_i & 0 \end{bmatrix} . \tag{41}$$

9 Some basic algorithms involving dual vectors

9.1 Computation of finite screw motion parameters from dual transform matrix

Making use of the Principle of Transference, the algorithm proposed for the spherical case [47] can be extended to the more general screw motion.

Let us denote with

$$\left[\widehat{A}\right] = \begin{bmatrix} \widehat{a}_{11} & \widehat{a}_{12} & \widehat{a}_{13} \\ \widehat{a}_{21} & \widehat{a}_{22} & \widehat{a}_{23} \\ \widehat{a}_{31} & \widehat{a}_{32} & \widehat{a}_{33} \end{bmatrix} \tag{42}$$

the dual transform matrix associated with a finite screw motion.

We will first compute from the matrix elements the dual Euler parameters and then the corresponding screw motion parameters.

The dual Euler parameters can be computed by choosing the most appropriate of the four formulas set presented in Table 4. In this choice division by a pure dual number must be avoided.

Once the dual Euler parameters are computed, the dual angle $\widehat{\theta}$ and the dual components of the screw axis $\left\{\widehat{h}\right\}$ are obtained as follows:

$$\widehat{\theta} = 2\ \mathrm{acos}\widehat{e}_0 ,$$

$$\widehat{h}_x = \frac{\widehat{e}_1}{\sin\frac{\widehat{\theta}}{2}} , \quad \widehat{h}_y = \frac{\widehat{e}_2}{\sin\frac{\widehat{\theta}}{2}} , \quad \widehat{h}_z = \frac{\widehat{e}_3}{\sin\frac{\widehat{\theta}}{2}} .$$

9.2 Composition of finite screw motions

Given two spherical rotations, the first one of an angle θ_1 about the axis $\vec{u}_1$, and the second of an angle θ_2 abuot the axis $\vec{u}_2$. Let us introduce the vector

$$\vec{\tau}_i = \tan\frac{\theta_i}{2}\vec{u}_i \quad (i = 1, 2, 3) , \tag{48}$$

It can be demonstrated [41, 43] that the resultant spherical motion is defined by the following vector

$$\vec{\tau}_3 = \frac{\vec{\tau}_1 + \vec{\tau}_2 - \vec{\tau}_1 \times \vec{\tau}_2}{1 - \vec{\tau}_1 \cdot \vec{\tau}_2} . \tag{49}$$

Consider two finite screw motions about the axes located by the unit line dual vector $\widehat{E}_i$ $(i = 1, 2)$. The rotation angle and the displacement along the axis are denoted with θ_i and h_i $(i = 1, 2)$, respectively.

Table 4: Set of formulas for computing Euler parameters

Case No.1

$$\widehat{e}_1 = \pm\frac{\sqrt{1+\widehat{a}_{11}-\widehat{a}_{22}-\widehat{a}_{33}}}{2}\,, \tag{43a}$$

$$\widehat{e}_2 = \frac{\widehat{a}_{12}+\widehat{a}_{21}}{4\widehat{e}_1}\,, \tag{43b}$$

$$\widehat{e}_3 = \frac{\widehat{a}_{13}+\widehat{a}_{31}}{4\widehat{e}_1}\,, \tag{43c}$$

$$\widehat{e}_0 = \frac{\widehat{a}_{32}-\widehat{a}_{23}}{4\widehat{e}_1}\,. \tag{43d}$$

Case No.2

$$\widehat{e}_2 = \pm\frac{\sqrt{1-\widehat{a}_{11}+\widehat{a}_{22}-\widehat{a}_{33}}}{2}\,, \tag{44a}$$

$$\widehat{e}_3 = \frac{\widehat{a}_{23}+\widehat{a}_{32}}{4\widehat{e}_2}\,, \tag{44b}$$

$$\widehat{e}_0 = \frac{\widehat{a}_{13}-\widehat{a}_{31}}{4\widehat{e}_2}\,. \tag{44c}$$

$$\widehat{e}_1 = \frac{\widehat{a}_{12}+\widehat{a}_{21}}{4\widehat{e}_2}\,, \tag{44d}$$

Case No.3

$$\widehat{e}_3 = \pm\frac{\sqrt{1-\widehat{a}_{11}-\widehat{a}_{22}+\widehat{a}_{33}}}{2}\,, \tag{45a}$$

$$\widehat{e}_2 = \frac{\widehat{a}_{32}+\widehat{a}_{23}}{4\widehat{e}_3}\,, \tag{45b}$$

$$\widehat{e}_1 = \frac{\widehat{a}_{13}+\widehat{a}_{31}}{4\widehat{e}_3}\,, \tag{45c}$$

$$\widehat{e}_0 = \frac{\widehat{a}_{21}-\widehat{a}_{12}}{4\widehat{e}_3}\,. \tag{45d}$$

Case No.4

$$\widehat{e}_0 = \pm\frac{\sqrt{1+\widehat{a}_{11}+\widehat{a}_{22}+\widehat{a}_{33}}}{2}\,, \tag{46a}$$

$$\widehat{e}_1 = \frac{\widehat{a}_{32}-\widehat{a}_{23}}{4\widehat{e}_0}\,, \tag{46b}$$

$$\widehat{e}_2 = \frac{\widehat{a}_{13}-\widehat{a}_{31}}{4\widehat{e}_0}\,, \tag{46c}$$

$$\widehat{e}_3 = \frac{\widehat{a}_{21}-\widehat{a}_{12}}{4\widehat{e}_0}\,. \tag{46d}$$

The formula for the composition of these finite motions is obtained by changing the vectors into dual vectors

$$\widehat{T}_3 = \frac{\widehat{T}_1+\widehat{T}_2-\widehat{T}_1\times\widehat{T}_2}{1-\widehat{T}_1\cdot\widehat{T}_2}\,, \tag{50}$$

where

$$\widehat{T}_i = \widehat{E}_i \tan\frac{\widehat{\theta}_i}{2}\,, (i=1,2,\ldots) \tag{51}$$

10 Nonlinear equations

10.1 Polynomial equations

An n^{th} degree dual polynomial has the form

$$\widehat{P}_n\left(\widehat{x}\right) \equiv \widehat{c}_n\widehat{x}^n + \widehat{c}_{n-1}\widehat{x}^{n-1} + \cdots \widehat{c}_1\widehat{x} + \widehat{c}_0 = 0\,. \tag{52}$$

Since $\widehat{x} = x + \varepsilon x^o$ and $\widehat{c}_i = c_i + \varepsilon c_{o_i}$ $(i = 0, 1, \ldots, n)$. After expansion and separation of (52) into primary and dual parts one obtains, respectively [22, 11]:

$$P_n(x) \equiv c_n x^n + c_{n-1}x^{n-1} + \cdots + c_1 x + c_0 = 0\,, \tag{53}$$

$$x^o = -\frac{\sum_{k=0}^{n} c_{o_k} x^k}{\sum_{k=0}^{n} k c_k x^{k-1}}\,. \tag{54}$$

Since the solutions of (52) are not necessarily real, the *complex dual numbers* need to be introduced [19, 11]. A complex dual number can be written as follows:

$$\breve{w} = (x + iy) + \varepsilon\,(x^o + iy^o) \tag{55}$$

where:

- $i = \sqrt{-1}$ is the imaginary unit;
- ε is the pure dual unit, such that $\varepsilon^2 = \varepsilon^3 = \ldots = 0$;
- x and y are the primary real and imaginary parts, and x^o and y^o are the dual real and imaginary parts, respectively.

When (53) has coincident roots, the following cases apply[11]:

- If $\sum_{k=0}^{n} c_{o_k} x^k = 0$, then there is an infinite number of solutions for the dual part x^o of $\widehat{x}$;
- If $\sum_{k=0}^{n} c_{o_k} x^k \neq 0$, then the corresponding dual part of the solution is $x^o = \infty$.

10.2 Iterative solution of a nonlinear dual equation

Let $F\left(\widehat{x}\right) = 0$ be a non linear equation with dual unknown $\widehat{x}$. The monodimensional Newton-Raphson iteration can be written in the form

$$\widehat{x}^{(k+1)} = \widehat{x}^{(k)} - \frac{F\left(\widehat{x}^{(k)}\right)}{F'\left(\widehat{x}^{(k)}\right)}\,, \tag{56}$$

where $F'\left(\widehat{x}^{(k)}\right)$ denotes the derivative $\frac{\partial F}{\partial \widehat{x}}\big|_{\widehat{x}=\widehat{x}^{(k)}}$ and the upperscript k the iteration counter, respectively.

10.3 Iterative solution of a system of nonlinear dual equations

Let $\left\{\widehat{f}\right\} = \left\{ \begin{array}{cccc} \widehat{f}_1 & \widehat{f}_2 & \ldots & \widehat{f}_m \end{array} \right\}^T$ be a vector of dual equations in the dual unknowns $\widehat{x}_i$, $(i = 1, 2, \ldots, m)$. The Newton-Raphson iteration for the multidimensional case takes the form

$$\{\widehat{x}\}^{(k+1)} = \{\widehat{x}\}^{(k)} - \left[\widehat{J}\right]^{-1}\left\{\widehat{f}\right\} \tag{57}$$

where $\left[\widehat{J}\right]$ is the dual Jacobian matrix evaluated at $\{\widehat{x}\} = \{\widehat{x}\}^k$. If the vector of unknowns is formed by both dual and real variables, then the dual pseudoinverse matrix of the jacobian $\left[\widehat{J}\right]^+$ must be substituted to $\left[\widehat{J}\right]^{-1}$.

11 Applications

Many applications of dual algebra in kinematics require the numerical solution of the system of dual equations of the form

$$\widehat{f}_j\left(\widehat{x}_1, \widehat{x}_2, \ldots, \widehat{x}_n\right) = 0 \quad (j = 1, 2, \ldots, m) \tag{58}$$

where $\widehat{x}_i$ $(i = 1, 2, \ldots, n)$ are the unknowns.

A common approach (*e.g.* [29, 27, 32]) is to split the equations in the primary and dual parts and then form a system of real equations. In this investigation the possibility of solving the equations *directly* in dual form is explored. H.H. Cheng and S. Thompson [12] followed similar lines. However, during Newton-Raphson iteration, the system of dual equations is splitted into two different linear systems of real equations. The first one composed by the real part of closure equations and the second by their dual part. As demonstrated in this paper, solution schemes can be devised such that the separation in real and dual parts is not required.

In order to test the feasibility of our approach, programs have been written in the Ch programming language [15] for different kinematic analysis applications.

12 Loop closure equations using dual matrices

Loop closure equations are often used to solve displacement analysis of spatial mechanisms [62, 37, 54]. By resorting to dual algebra, instead of the classical 4×4 Denavit-Hartenberg transform matrix, a 3×3 matrix of dual elements [13, 14] can be used [62, 37].

12.1 Kinematic analysis of the RCCC spatial mechanism

The following different numerical methods of solution are presented:

- Method A: Iterative numerical solution of a dual nonlinear equation.
- Method B: Iterative solution of system of dual nonlinear equations.
- Method C: Iterative solution of a system of redundant nonlinear equations.

Method A
The following example is aimed to give a practical application of (56).

In bibliographical reference [12] it is demonstrated that position analysis of the RCCC linkage reduces to the solution of the following nonlinear equation[6]:

$$\widehat{F}\left(\widehat{\theta}_4\right) \equiv \widehat{A}\sin\widehat{\theta}_4 + \widehat{B}\cos\widehat{\theta}_4 + \widehat{C} = 0\,, \tag{59}$$

where

$$\widehat{A} = \sin\widehat{\alpha}_1 \sin\widehat{\alpha}_3 \sin\widehat{\theta}_1\,, \tag{60}$$

$$\widehat{B} = -\sin\widehat{\alpha}_3\left(\cos\widehat{\alpha}_1 \sin\widehat{\alpha}_4 + \sin\widehat{\alpha}_1 \cos\widehat{\alpha}_4 \cos\widehat{\theta}_1\right)\,, \tag{61}$$

$$\widehat{C} = \cos\widehat{\alpha}_3\left(\cos\widehat{\alpha}_1 \cos\widehat{\alpha}_4 - \sin\widehat{\alpha}_1 \sin\widehat{\alpha}_4 \cos\widehat{\theta}_1\right) - \cos\widehat{\alpha}_2\,. \tag{62}$$

[6] The nomenclature adopted in this section follows the one of reference [12].

Table 5: Results of dual monodimensional Newton-Raphson iteration applied to equation (59).

k	$\widehat{\theta}_4^{(k)}$	$\widehat{F}\left(\widehat{\theta}_4^{(k)}\right)$
0	1.745329 -ε 1.300000	-0.162529 -ε 0.403280
1	2.009102, -ε1.657790	-0.013617 - ε 0.003539
2	2.035995 -ε 1.767060	-0.000178 + ε 0.000931
3	2.036356 -ε 1.770564	-0.000000 + ε 0.000001

We acknowledge that (59) can be solved by substituting the $\sin\widehat{\theta}_4$ and $\cos\widehat{\theta}_4$ functions with their $\tan\frac{\widehat{\theta}_4}{2}$ correspondent. However, the numerical solution of such equation is herein reported for demonstration purposes only. If the link dimensions, in term of DH parameters, are as follows: $\alpha_1 = 30°$, $\alpha_2 = 55°$, $\alpha_3 = 45°$, $\alpha_4 = 60°$, a_1 = 2, a_2 = 4, a_3 = 3, a_5 = 5, then we have

$$\widehat{A} = 0.227260 + \varepsilon 1.469030\ , \quad \widehat{B} = -0.665749 - \varepsilon 2.212148\ , \quad \widehat{C} = -0.501943 - \varepsilon 1.433104\ .$$

Assuming $\widehat{\theta}_4^{(0)} = 1.745329 - \varepsilon 1.300000$ as solution guess, Table 5 reports the results of the Newton-Raphson iteration.

Method B

The mechanism matrix closure equation can be written in the form

$$\left[\widehat{\Lambda}_3\right]\left[\widehat{\Theta}_4\right]\left[\widehat{\Lambda}_4\right]\left[\widehat{\Theta}_1\right]\left[\widehat{\Lambda}_1\right] - \left[\widehat{\Theta}_3\right]^T\left[\widehat{\Lambda}_2\right]^T\left[\widehat{\Theta}_2\right]^T = [0]\ . \tag{63}$$

Only three of the nine elements of the above matrix form a set of independent equations[7]. If we choose the elements (3,3), (3,1), (1,1), the system of nonlinear equations is the following $\left\{\widehat{f}\right\} \equiv \left\{\ \widehat{f}_1 \quad \widehat{f}_2 \quad \widehat{f}_3\ \right\}^T = \{0\}$, where

$$\begin{aligned}\widehat{f}_1 = &\left[\sin\widehat{\alpha}_3\sin\widehat{\theta}_4\sin\widehat{\theta}_1 - \left(\sin\widehat{\alpha}_3\cos\widehat{\alpha}_4\cos\widehat{\theta}_4 - \cos\widehat{\alpha}_3\sin\widehat{\alpha}_4\right)\cos\widehat{\theta}_1\right]\sin\widehat{\alpha}_1 \\ &+ \left(\cos\widehat{\alpha}_3\cos\widehat{\alpha}_4 - \sin\widehat{\alpha}_3\cos\widehat{\theta}_4\sin\widehat{\alpha}_4\right)\cos\widehat{\alpha}_1 - \cos\widehat{\alpha}_2\ ,\end{aligned} \tag{64a}$$

$$\widehat{f}_2 = \sin\widehat{\alpha}_3\sin\widehat{\theta}_4\cos\widehat{\theta}_1 + \left(\sin\widehat{\alpha}_3\cos\widehat{\alpha}_4\cos\widehat{\theta}_4 + \cos\widehat{\alpha}_3\sin\widehat{\alpha}_4\right)\sin\widehat{\theta}_1 - \sin\widehat{\alpha}_2\sin\widehat{\theta}_2\ , \tag{64b}$$

$$\widehat{f}_3 = \cos\widehat{\theta}_4\cos\widehat{\theta}_1 - \cos\widehat{\alpha}_4\sin\widehat{\theta}_4\sin\widehat{\theta}_1 - \cos\widehat{\theta}_3\cos\widehat{\theta}_2 + \sin\widehat{\theta}_3\cos\widehat{\alpha}_2\sin\widehat{\theta}_2\ . \tag{64c}$$

Assuming the same dimensions specified in the previous example, the results of iteration (57) are summarized in Table 6.

Method C

The solution method of matrix loop closure equations, originally proposed by J.J. Uicker *al.* [63], can

[7]The choice of independent equations must obey the rules stated by the following theorem ([7], p.5):

> The orientation or attitude of a Cartesian system of coordinates 2 relative to system 1 is uniquely specified by stating the values of three elements of the transform matrix which lie in any two rows, and one of four possible values for a fourth element, chosen so that the four elements do not lie in the same minor, and less than three elements in a row. The word *row* may be replaced by the word *column* throughout the theorem.

Table 6: Results of dual multidimensional Newton-Raphson iteration applied to system of equations (64).

k	$\widehat{\theta}_4^{(k)}$	$\sum_{i=1}^{3} (f_i)^2 + (f_i^o)^2$
0	1.745329 -ε 1.300000	0.552520
1	2.009102 -ε 1.657790	0.002172
2	2.035995 - ε 1.767061	0.205704
3	2.036356 - ε 1.770565	0.002170
4	2.036356 - ε 1.770568	0.000049
5	2.036356 -ε 1.770567	0.000003
6	2.036356 - ε 1.770567	0.000000

be adapted to dual equations (*e.g.* [12, 32]). For completeness the method is summarized with reference to the position analysis of the RCCC linkage. If $\left\{\widehat{\theta}^{(k)}\right\}$ is a solution guess vector and $\left\{\Delta\widehat{\theta}^{(k)}\right\}$ the corresponding correction vector, the following equality can be established:

$$\left[\widehat{A}_i\left(\widehat{\theta}_i^{(k)} + \Delta\widehat{\theta}_i^{(k)}\right)\right] = \left[\widehat{A}_i^{(k)}\right] + \left[\left.\frac{\partial A_i^{(k)}}{\partial \theta_i}\right|_{\widehat{\theta}_i = \widehat{\theta}_i^{(k)}}\right] \Delta\widehat{\theta}_i^{(k)} \tag{65}$$

The partial derivative is also expressed by the product

$$\left[\left.\frac{\partial A_i^{(k)}}{\partial \theta_i}\right|_{\widehat{\theta}_i = \widehat{\theta}_i^{(k)}}\right] = [Q]\left[\widehat{A}_i^{(k)}\right] \quad (i = 2, 3, 4) \tag{66}$$

where

$$[Q] = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} . \tag{67}$$

Hence, when in the the matrix loop equation

$$\left[\widehat{A}_1\left(\widehat{\theta}_1\right)\right]\left[\widehat{A}_2\left(\widehat{\theta}_2\right)\right]\left[\widehat{A}_3\left(\widehat{\theta}_3\right)\right]\left[\widehat{A}_4\left(\widehat{\theta}_4\right)\right] = [I] \ . \tag{68}$$

we substitute the equalities

$$\widehat{\theta}_i = \widehat{\theta}_i^{(k)} + \Delta\widehat{\theta}_i^{(k)} \quad (i = 2, 3, 4) \tag{69}$$

we obtain, neglecting higher order terms,

$$[B_2]\,\Delta\widehat{\theta}_2^{(k)} + [B_3]\,\Delta\widehat{\theta}_3^{(k)} + [B_4]\,\Delta\widehat{\theta}_4^{(k)} = [I] - [B_1] \tag{70}$$

where

$$\left[\widehat{B}_1\right] = \left[\widehat{A}_1\right]\left[\widehat{A}_2^{(k)}\right]\left[\widehat{A}_3^{(k)}\right]\left[\widehat{A}_4^{(k)}\right] \tag{71}$$

and

$$\left[\widehat{B}_i\right] = \left[\widehat{A}_1\right]\ldots\left[\widehat{A}_{i-1}^{(k)}\right][Q]\left[\widehat{A}_{i+1}^{(k)}\right]\ldots\left[\widehat{A}_4^{(k)}\right] \ , (i = 2, 3, 4) \ . \tag{72}$$

The system of redundant but consistent equations (70) can be rewritten in the following ready for computation form

$$\begin{bmatrix} \widehat{B}_{2_{11}} & \widehat{B}_{3_{11}} & \widehat{B}_{4_{11}} \\ \widehat{B}_{2_{21}} & \widehat{B}_{3_{21}} & \widehat{B}_{4_{21}} \\ \widehat{B}_{2_{31}} & \widehat{B}_{3_{31}} & \widehat{B}_{4_{31}} \\ \widehat{B}_{2_{22}} & \widehat{B}_{3_{22}} & \widehat{B}_{4_{22}} \\ \widehat{B}_{2_{32}} & \widehat{B}_{3_{32}} & \widehat{B}_{4_{32}} \\ \widehat{B}_{2_{33}} & \widehat{B}_{3_{33}} & \widehat{B}_{4_{33}} \end{bmatrix} \begin{Bmatrix} \Delta\widehat{\theta}_2^{(k)} \\ \Delta\widehat{\theta}_3^{(k)} \\ \Delta\widehat{\theta}_4^{(k)} \end{Bmatrix} = \begin{Bmatrix} 1-\widehat{B}_{1_{21}} \\ -\widehat{B}_{1_{21}} \\ -\widehat{B}_{1_{31}} \\ 1-\widehat{B}_{1_{22}} \\ -\widehat{B}_{1_{32}} \\ 1-\widehat{B}_{1_{33}} \end{Bmatrix} \tag{73}$$

With obvious substitutions, this matrix equation can be more concisely expressed as

$$\left[\widehat{M}\right]\left\{\Delta\widehat{\theta}^{(k)}\right\} = \{\widehat{v}\} \ . \tag{74}$$

There are different alternatives to solve (74). Adopting the least-squares method, the correction vector is obtained by solving the linear system

$$\left[\widehat{M}\right]^T\left[\widehat{M}\right]\left\{\Delta\widehat{\theta}^{(k)}\right\} = \left[\widehat{M}\right]^T\{\widehat{v}\} \ . \tag{75}$$

of three dual equations in three dual unknowns. As previously explained, there are two different computing schemes. Otherwise, one can iteratively apply the formula (69), where the pseudoinverse matrix of $\left[\widehat{M}\right]$ is used to compute the correction vector

$$\left\{\Delta\widehat{\theta}^{(k)}\right\} = \left[\widehat{M}\right]^+\{\widehat{v}\} \ . \tag{76}$$

From the numerical point of view, the solution method C can be executed in the following three different alternatives:

- Alternative C1: Solve the linear system (75) using the equations (30), as suggested in [12].
- Alternative C2: Solve the linear system (75) using the alternative procedure proposed in this paper for the solution of dual linear equations system (see end of subsection 7.7).
- Alternative C3: Solve the linear system (75) making use of the dual QR decomposition (see subsection 7.8).
- Alternative C4: Solve the redundant equations system (74) computing the pseudoinverse matrix (see subsection 7.9) and making use of equation (76).

All the alternatives listed above have been implemented in the Ch programming language and the CPU time[8] required by each of them reported in Table 7.

[8]Under Windows XP operating system it is not usually possible the precise monitoring of CPU time elapsed during the run of a task. For this reason, each alternative has been executed 10 times and the average CPU time reported.

Table 7: Comparison of CPU times required for the analysis of the RCCC spatial linkage

Alternative	CPU Time (s)
C1	1.422
C2	1.214
C3	1.175
C4	1.276

13 Conclusions

This paper presented several basic algorithms regarding vectors and matrices of dual numbers. The algorithms, arranged in a form suitable for a ready implementation into a code, should provide useful numerical tools for the development of analyses based on the use of dual numbers. In most of the cases the algorithms are accompanied by simple numerical examples to demonstrate their effectiveness. The algorithms of pseudoinverse, eigenvalue and SVD decomposition are beleived to be novel in the field of dual numbers. The availability of such algorithms should broad the field of application of dual numbers. The paper also presents some applications of these algorithms to the solution of classical kinematic problems. In particular different approaches to the numerical kinematic analysis of the RCCC spatial mechanism have been discussed and compared on the basis of their computational efficiency. The proposed approach for the solution of a redundant system of nonlinear equations offers a computational gain with respect other approaches. All Ch routines implemented are available upon request.

References

[1] S. K. Agrawal. Multibody dynamics: A formulation using Kane's method and dual vectors. *ASME Journal of Mechanical Design*, 115:833–838, 1993.

[2] J. Angeles. The application of dual algebra to kinematic analysis. In J. Angeles and E. Zakhariev, editors, *Computational Methods in Mechanical Systems*, pages 3–32. Springer Verlag, 1991.

[3] P. Azariadis and N. Aspragathos. Computer graphics representation and transformation of geometric entities using dual unit vectors and line transformations. *Computers & Graphics*, 25:195–209, 2001.

[4] C. Bagci. Static force and torque analysis using 3×3 screw matrix, and transmission criteria for space mechanisms. *ASME Journal of Engineering for Industry*, 93:90–101, 1971.

[5] C. Bagci. Dynamic force and torque analysis for mechanisms using dual vectors and 3×3 screw matrix. *ASME Journal of Engineering for Industry*, 94(738-745), 1972.

[6] R. S. Ball. *Theory of Screws*. Cambridge University Press, 1900.

[7] J. S. Beggs. *Advanced Mechanisms*. Macmillan, New York, 1966.

[8] R. Beyer. *Technische Raumkinematik*. Springer Verlag, Berlin, Gottingen, Heidelberg, 1963.

[9] L. Brand. *Vector and Tensor Analysis*. John Wiley and Sons, 1947.

[10] M. Cavacece, R. Stefanelli, P. P. Valentini, and L. Vita. A multibody dynamic model of a Cardan joint with experimental validation. In J. M. Goicolea, J. C. Cuadrado, and J. Garcia Orden, editors, *Proc. of Multibody Dynamics 2005, ECCOMAS Thematic Conference*, Madrid, Spain, 2005.

[11] H. H. Cheng and S. Thompson. Dual polynomials and complex dual numbers for analysis of spatial mechanisms. In *Proceedings of the 1996 ASME 24th Mechanisms Conference*, pages Paper 96DETC–MECH–1221. ASME, 1996.

[12] H. H. Cheng and S. Thompson. Dual iterative displacement analysis of spatial mechanisms using the Ch programming language. *Mechanism and Machine Theory*, 32(2):193–207, 1997.

[13] H. H. Cheng. Computation of dual numbers in the extended finite dual plane. In *Proc. of the 1993 ASME Design Automation Conference*, pages 73–80, Sept. 19-22 1993.

[14] H. H. Cheng. Numerical computations in the Ch programming language with applications in mechanisms and robotics. Parts I, II, III. In *Proc. of the Third National Conference on Mechanisms and Robotics*, Cincinnati, OH, Nov. 8-10 1993.

[15] H. H. Cheng. Programming with dual numbers and its applications in mechanisms design. *Engineering with Computers*, 10(4):212–229, 1994.

[16] C. K. Chen and F. Freudenstein. Dynamic analysis of a universal joint with manufacturing tolerances. *ASME Journal of Mechanisms, Transmissions and Automation in Design*, 108:524–532, December 1985.

[17] W. K. Clifford. Preliminary sketch of biquaternions. *Proc. London Mathematical Society*, 4(64):381–395, 1873.

[18] K. Daniilidis. Hand-eye calibration using dual quaternions. *International Journal of Robotics Research*, 18:286–298, 1999.

[19] J. Denavit. Displacement analysis of mechanisms on 2×2 matrices of dual numbers. *VDI Berichte*, 29:81–88, 1958.

[20] D. De Falco, E. Pennestrì, and L. Vita. Esperienze numeriche sulla formulazione di Udwadia-Kalaba. In *Atti Congresso AIMETA 2005*, 2005.

[21] D. De Falco and E. Pennestrì. The Udwadia-Kalaba formulation: A report on its numerical efficiency and on its teaching effectiveness. In *Proc. of the Multibody Dynamics 2005, ECCOMAS Thematic Conference*, 2005.

[22] F. M. Dimentberg. The screw calculus and its applications. Technical Report AD 680993, Clearinghouse for Federal and Scientific Technical Information, Virginia (USA), 1968.

[23] J. Duffy. *Analysis of Mechanisms and Robot Manipulators*. Halstead Press, 1980.

[24] E. Fasse. Some applications of screw theory to lumped parameter modeling of visco-elastically coupled rigid bodies. In *Proc. of A Symposium Commemorating the Legacy, Works, and Life of Sir Robert Stawell Ball (Ball 2000), University of Cambridge, Trinity College.* 2000. citeseer.ist.psu.edu/fasse00some.html.

[25] I. S. Fischer and T. Chu. Numerical analysis of displacements in multiloop mechanisms. *Mechanics Research Communications*, 28(2):127–137, 2001.

[26] I. S. Fischer and F. Freudensetin. Internal force and moment transmission in a cardan joint with manufacturing tolerances. *ASME Journal of Mechanisms, Transmissions and Automation in Design*, 106:301–311, December 1984.

[27] I. S. Fischer and R. N. Paul. Kinematic displacement analysis of a double-cardan-joint driveline. *ASME Journal of Mechanical Design*, 113:263–271, September 1991.

[28] I. S. Fischer. *Dual-Number Methods in Kinematics, Statics and Dynamics.* CRC Press, Boca Raton London New York Washington D.C., 1999.

[29] I. S. Fischer. Modeling of plane joint. *ASME Journal of Mechanical Design*, 121:383–386, September 1999.

[30] I. S. Fischer. Numerical analysis of displacements in spatial mechanisms with ball joints. *Mechanism and Machine Theory*, 35:1623–1640, 2000.

[31] I. S. Fischer. Velocity analysis of mechanisms with ball joints. *Mechanics Research Communications*, 30:69–78, 2003.

[32] I. Fischer. Numerical analysis of displacements in a tracta coupling. *Engineering with Computers*, 15:334–344, 1999.

[33] G. H. Golub and C. F. Van Loan. *Matrix Computations*. John Hopkins University Press, third edition, 1996.

[34] M. A. Gonzáles-Palacios, J. Angeles, and F. Ranjbaran. The kinematic synthesis of serial manipulators with prescribed jacobian. volume I, pages 450–455. IEEE International Conference on Robotics and Automation, 1993.

[35] M. A. Gonzáles-Palacios and J. Angeles. *Cam Synthesis*. Kluwer Academic Publisghers, 1993.

[36] Y.-L. Gu and J. Y. S. Luh. Dual-number transformations and its applications to robotics. *IEEE Journal of Robotics and Automation*, RA-3(615-623), December 1987.

[37] A. S. Hall, R. S. Root, and E. Sandgren. A dependable method for solving matrix loop equations for the general three-dimensional mechanism. *ASME Journal of Engineering for Industry*, pages 547–550, 1977.

[38] L. M. Hsia and A. T. Yang. On the Principle of Transference i n three dimensional kinematics. *ASME Journal of Mechanical Design*, 103:652–656, 1981.

[39] M. L. Keler. Kinematics and statics including friction in single-loop mechanisms by screw calculus and dual vectors. *ASME Journal of Engineering for Industry*, 95(2):471–480, 1973.

[40] T. Liu and T. W. Lee. Dynamics of an overconstrained shaft coupling. *ASME Journal of Mechanisms, Transmissions and Automation in Design*, 108:497–502, December 1986.

[41] R. Marcolongo. *Meccanica Razionale*. Hoepli, Milano, 1953.

[42] J. M. McCarthy. Dual orthogonal matrices in manipulator kinematics. *International Journal of Robotics Research*, 5:45–51, 1986.

[43] J. M. McCarthy. *Geometric Design of Linkages*. Springer Verlag, 2000.

[44] Y.-M. Moon and S. Kota. Automated synthesis of mechanisms using dual-vector algebra. *Mechanism and Machine Theory*, 37:143–166, 2002.

[45] E. Pennestrì, M. Cavacece, P. P. Valentini, and L. Vita. Mechanical efficiency analysis of a cardan joint. In *Proc. of 2004 ASME Design Engineering Technical Conferences*, pages DETC04/MECH–57317, Salt Lake City, Utah, 2004. ASME.

[46] E. Pennestrì and L. Vita. Mechanical efficiency analysis of a cardan joint with manufacturing tolerances. In *Proc. of the RAAD03 12th International Workshop on Robotics in Alpe-Adria-Danube Region*, page Paper 053, Cassino, Italy, 2003.

[47] E. Pennestrì. Cinematica teorica. In F. Cheli and E. Pennestrì, editors, *Cinematica e Dinamica dei Sistemi Multibody*, chapter I, pages 21–24. Casa Editrice Ambrosiana, Milano, 2006.

[48] G. R. Pennock and K. G. Mattson. Forward position problem of two puma-type robots manipulating a planar four-bar linkage payload. Minneapolis, Minnesota, April 1996. IEEE International Conference on Robotics and Automation.

[49] G. R. Pennock and A. T. Yang. Dynamic analysis of multi-rigid-body open-chain system. *ASME Journal of Mechanisms, Transmissions and Automation in Design*, 105:28–33, 1983.

[50] G. R. Pennock and A. T. Yang. Application of dual-number matrices to the inverse kinematics problem of robot manipulators. *ASME Journal of Mechanisms, Transmissions and Automation in Design*, 107:201–208, 1985.

[51] A. Perez and J. M. McCarthy. Dimensional sinthesis of Bennet linakge. In *Proceedings of 2000 ASME Design Technical Engineering Conferences*, pages Paper DETC2000/Mech–14069. ASME, 2000.

[52] A. Perez and J. M. McCarthy. Bennet's linkage and the cylindroid. *Mechanism and Machine Theory*, 37:1245–1260, 2002.

[53] A. Perez. *Analysis and Design of Bennett Linkages*. Master thesis, University of California, 1999.

[54] P. Potts. A derivation of a minimal set of multilinear loop equations for spatial mechanisms. *ASME Journl of Engineering for Industry*, 98(4), November 1976.

[55] J. Rooney. On the Principle of Transference. In *Proc. of the IV IFToMM Congress*, pages 1089–1094, New Castle Upon Tyne (UK), 1975. I.Mech.E.

[56] J. A. Schaaf and B. Ravani. Geometry continuity of ruled surfaces. *Computer Aided Geometric Design*, 15:289–310, 1998.

[57] M. Shoham and V. Brodsky. Analysis of mechanisms by dual inertia operator. In J. A. et. al., editor, *Computational Kinematics*. Kluwer Academic Publishers, Netherlands, 1993.

[58] A. H. Soni and L. Harrisberger. Die Anwendung der 3×3 Schraubungs Matrix auf die kinematische und dynamische Analyze der raumlichen Getrieben. *VDI Berichte*, (127), 1968.

[59] E. Study. *Geometrie der Dynamen*. Verlag Teubner, Leipzig, 1903.

[60] K. Sugimoto and J. Duffy. Application of linear algebra to screw systems. *Mechanism and Machine Theory*, 17(1):73–83, 1982.

[61] K. Teu and W. Kim. Estimation of the axis of a screw motion from noisy dataa new method based on Plücker lines. *Journal of Biomechanics*, 39:2857–2862, 2006.

[62] J. J. Uicker, J. Denavit, and R. S. Hartenberg. An iterative method for the displacement analysis of spatial mechanisms. *ASME Journal of Applied Mechanics*, pages 309–314, June 1964.

[63] J. J. Uicker, J. Denavit, and R. S. Hartenberg. An iterative method for the displacement analysis of spatial mechanisms. *ASME Journal of Applied Mechanisms*, 34:309–314, 1964.

[64] G. R. Veldkamp. On the use of dual numbers, vectors and matrices in instantaneous, spatial kinematics. *Mechanism and Machine Theory,*, 11(2):141–156, 1976.

[65] J. Wittenburg. Dual quaternions in kinematics of spatial mechanisms. In E. J. Haug, editor, *Computer Aided Analysis and Optimization of Mechanical System Dynamics*, pages 129–145. Springer-Verlag, 1984.

[66] K. Wohlhart. Motor tensor calculus. In J. P. Merlet and B. Ravani, editors, *Computational Kinematics*, pages 93–102. Kluwer Academic Publishers, 1995.

[67] I. M. Yaglom. *Complex Numbers in Geometry*. Academic Press, New York, 1968.

[68] A. T. Yang and F. Freudenstein. Application of dual number quaternions algebra to the analysis of spatial mechanisms. *ASME Journal of Engineering for Industry*, 86:300–308, June 1964.

[69] A. T. Yang. *Application of Quaternion Algebra and Dual Numbers to the Analysis of Spatial Mechanisms*. PhD thesis, Columbia University, 1963.

[70] A. T. Yang. Analysis of an offset unsymmetric gyroscope with oblique rotor using (3x3) matrices with dual-number elements,. *ASME Journal of Engineering for Industry*, 91(3):535–542, 1969.

[71] A. T. Yang. Displacement analysis of spatial five-link mechanisms using 3 x 3 matrices with dual elements. *ASME Journal of Engineering for Industry*, 91(1):152–157, 1969.

[72] A. T. Yang. Inertial force analysis of spatial mechanisms. *ASME Journal of Engineering for Industry*, 93(1):27–32, 1971.

[73] A. T. Yang. Calculus of screws. In *Basic Questions of Design Theory*. North Holland Publishing Company,, Amsterdam, 1974.

14 Ch program listings

14.1 QR decomposition of a dual matrix

```
void dual_gram_schmidt(array double dual a[:][:],
array double dual q[:][:],r[:][:],int &m,int &n){
/*Purpose: QR decomposition of a dual matrix
           by means of the modified Gram-Schmidt
           orthogonalization procedure.
Input parameters:
    a: Dual matrix (m x n). Type: double dual.
    m : Number of rows; Type: int.
    n : Number of columns; Type: int.
```

```
Output parameters:
     q: Dual matrix (m x n). Type: double dual.
     r: Dual matrix (m x n). Type: double dual.
Functions required: None
     */
int i,j,k;dual z;
for(j=0;j<n;j=j+1){
for(i=0;i<n;i=i+1){r[i][j]=dual(0.,0.);}
for(i=0;i<m;i=i+1){q[i][j]=a[i][j];}
      }
for(k=0;k<n;k=k+1){
           z=dual(0.,0.);
for(i=0;i<m;i=i+1){z=z+q[i][k]*q[i][k];}
           r[k][k]=sqrt(z);
for(i=0;i<m;i=i+1){q[i][k]=q[i][k]/r[k][k];}

for(j=k+1;j<n;j=j+1){
               z=dual(0.,0.);
for(i=0;i<m;i=i+1){z=z+q[i][j]*q[i][k];}
               r[k][j]=z;
for(i=0;i<m;i=i+1){q[i][j]=q[i][j]-r[k][j]*q[i][k];}
          }
          }

#include <array.h>
#include<numeric.h>
#include<math.h>
#pragma importf <dual_library1.chf>
#define D2R M_PI/180
#define R2D 180/M_PI
#include<time.h>
main(){
    double begintime;
    begintime=clock();
     int i,j,theta1,n=4,m=6,nn=3;
     double error, epsilon=0.000001,MaxError=100000000;
     double dual theta[n],alpha[n];
     array double rd[3],dd[3];
     array double dual Q[3][3],Bi[3][3],B1[3][3],Ai[3][3],\
    M[m][3],d[3],v[m],N[3][3],u[3],Minv[3][m];
     array double dual q[3][3],r[3][3],d1[3];
     void ai(double dual dtheta, dalpha, array double dual A[3][3]){
double dual ct,st,ca,sa;
ct=cos(dtheta);st=sin(dtheta);
ca=cos(dalpha);sa=sin(dalpha);
A[0][0]=ct;  A[0][1]=-st*ca;  A[0][2]=st*sa;
A[1][0]=st;  A[1][1]=ct*ca;   A[1][2]=-ct*sa;
A[2][0]=0;   A[2][1]=sa;      A[2][2]=ca;
     }
     alpha[0]=dual(30*D2R,2);
     alpha[1]=dual(55*D2R,4);
     alpha[2]=dual(45*D2R,3);alpha[3]=dual(60*D2R,5);
/*     Solution guess values*/
     theta[0]=dual(0,0);theta[1]=dual(100*D2R,0);
```

```
    theta[2]=dual(100*D2R,0); theta[3]=dual(100*D2R,0);
    printf("  theta1       s1      theta2       s2"  \
    "        theta3      s3       theta4       s4   \n");
    Q=0;Q[0][1]=-1;Q[1][0]=1,0;

    for(theta1=0;theta1<=360;theta1+=1){
        theta[0]=theta1*D2R;
        do{
            for(i=1;i<n;i++){
                for(Bi=0,j=0;j<=2;j++) Bi[j][j]=1;
                for(j=0;j<=i-1;j++,Bi*=Ai) ai(theta[j],alpha[j],Ai);
                for(Bi*=Q,j=i;j<n;j++,Bi*=Ai) ai(theta[j],alpha[j],Ai);
                M[0][i-1]=Bi[0][0];M[1][i-1]=Bi[1][1];M[2][i-1]=Bi[2][2];
                M[3][i-1]=Bi[1][0];M[4][i-1]=Bi[2][0];M[5][i-1]=Bi[2][1];
                }
ai(theta[0],alpha[0],B1);
for(i=1;i<n;i++,B1*=Ai) ai(theta[i],alpha[i],Ai);
v[0]=1-B1[0][0];v[1]=1-B1[1][1]; v[2]=1-B1[2][2];
v[3]=-B1[1][0]; v[4]=-B1[2][0];  v[5]=-B1[2][1];
/*  Alternative C1 (Same as Cheng & Thopmson, 1997)
N=transpose(M)*M;
u=transpose(M)*v;
linsolve(rd, real(N), real(u));
linsolve(dd, real(N), dual(u)-dual(N)*real(d));
for(i=0;i<n-1;i++) d[i]=dual(rd[i],dd[i]);
*/
/* Alternative C2
dual_linsolve(d, transpose(M)*M, transpose(M)*v,3);
*/
/* Alternative C3
dual_gram_schmidt(transpose(M)*M,q,r,nn,nn);
d1=transpose(q)*transpose(M)*v;
dual_tsolve(d,r,d1,nn);
*/
/* Alternative C4 */
dual_mat_pinv1(M,Minv,m,nn);
d=Minv*v;

for(error=0,i=0;i<n-1;i++)  error+=abs(real(d[i]))+abs(dual(d[i]));
for(i=1;i<n;i++) theta[i]+=d[i-1];
} while(error>=epsilon && error < MaxError);
if(error>=MaxError)
printf("Error: the iterative algorith diverges at theta1= %d \n",theta1);
else
printf("%8.2f   %8.2f  %8.2f   %8.2f   %8.2f   %8.2f   %8.2f   %8.2f   \n",
real(theta[0])*R2D,dual(theta[0]),real(theta[1])*R2D,dual(theta[1]),\
real(theta[2])*R2D,dual(theta[2]),real(theta[3])*R2D,\
dual(theta[3]));
}
printf("CPU: %f  s \n", (clock()-begintime)/CLOCKS_PER_SEC);
}
```