ISSN 2007-9737

Modeling and Pose Control of Robotic Manipulators and Legs using Conformal Geometric Algebra

Oscar Carbajal-Espinosa[1], Luis González-Jiménez[2], Jose Oviedo-Barriga[3], Bernardino Castillo-Toledo[4], Alexander Loukianov [4], Eduardo Bayro-Corrochano[4]

[1]Instituto Tecnológico y de Estudios Superiores de Monterrey, Guadalajara, Mexico

[2]Instituto Tecnológico y de Estudios Superiores de Occidente, Guadalajara, Mexico

[3]Universidad Veracruzana, Ciudad Mendoza, Mexico

[4]Centro de Investigación y De Estudios Avanzados del I.P.N., Guadalajara, Mexico

oscar.carbajal@itesm.mx, luis.gonzalez@iteso.mx, luoviedo@uv.mx, {toledo, louk, edb}@gdl.cinvestav.mx

Abstract. Controlling the pose of a manipulator involves finding the correct configuration of the robot's elements to move the end effector to a desired position and orientation. In order to find the geometric relationships between the elements of a robot manipulator, it is necessary to define the kinematics of the robot. We present a synthesis of the kinematical model of the pose for this type of robot using the conformal geometric algebra framework. In addition, two controllers are developed, one for the position tracking problem and another for the orientation tracking problem, both using an error feedback controller. The stability analysis is carried out for both controllers, and their application to a 6-DOF serial manipulator and the legs of a biped robot are presented. By proposing the error feedback and Lyapunov functions in terms of geometric algebra, we are opening a new venue of research in control of manipulators and robot legs that involves the use of geometric primitives, such as lines, circles, planes, spheres.

Keywords. Serial manipulators, pose control, motors, conformal geometric algebra.

1 Introduction

The combination of several areas of science resulted in a new interdisciplinary science called robotics and with it a new branch of problems to tackle and solve. The control of serial manipulators has a wide area of investigation to the development of new techniques of modeling and control for the pose of the robot; one of them is differential kinematics. In this work, a novel method for kinematic modeling and control of the pose of robotic manipulators will be investigated using the conformal geometric algebra (CGA) approach.

The kinematic model is obtained using **motors** which are a conformal entity that represents a rigid transformation and permit us to represent position and orientation motions. Using the same framework, error feedback controllers will be designed for the position and orientation tracking problem for an n-manipulator. The stability analysis using Lyapunov functions will be derived. The objective of this work is to develop the kinematic model and control laws for the pose of serial manipulators using CGA. This methodology is applied to a 6-DOF manipulator and a biped humanoid.

The CGA framework provides an easier and more intuitive way to tackle the kinematics problem due to its algebra properties. Furthermore, using this framework to define an error signal, we will be able to propose new control laws using geometric primitives like planes, spheres, or lines.

2 Geometric Algebra

A Geometric Algebra G_n is a linear space of dimension 2^n, $n = p + q + r$, where p, q, and r are the numbers of bases that square $+1$, -1, and 0, respectively. As well as vector-addition and scalar multiplication, G_n has a non-commutative product which is associative and distributive over addition. The latter called the *geometric* or *Clifford product*.

The Clifford product of two vectors a and b is defined as the sum of the *inner product* and the *wedge product*

$$ab = a \cdot b + a \wedge b, \tag{1}$$

where the inner product of the two vectors is the standard *scalar* or *dot* product, which produces a scalar. The outer or wedge product is anti-commutative ($a \wedge b = -b \wedge a$) and generates a new quantity which is called a *bivector*.

Then, the outer product is generalizable to higher dimensions. For example, $(a \wedge b) \wedge c$, a *trivector*, is interpreted as an oriented volume formed by sweeping the area $a \wedge b$ along vector c. The outer product of k vectors is a k-blade, and such a quantity is said to have *grade* k. A *multivector* is defined as a linear combination of objects of different grades, and is a *homogeneous* k-vector if it contains terms of only a single grade k. Now, given two k vectors A and B, we define the *conmutator product* [1] as

$$A \bar{\times} B = \frac{1}{2}\left(AB - BA\right). \tag{2}$$

We use e_i to denote the $i-th$ basis vector, where $1 \leq i \leq n$. In geometric algebra $G_{p,q,r}$, the geometric product of two basis vectors is defined as

$$e_i e_j = \left\{ \begin{array}{rll} 1 & for & i = j \in 1, \cdots, p \\ -1 & for & i = j \in p+1, \cdots, p+q \\ 0 & for & i = j \in p+q+1, \cdots, p+q+r. \\ e_i \wedge e_j & for & i \neq j \end{array} \right.$$

This leads to a basis for the entire algebra:

$$\{1\}, \{e_i\}, \{e_i \wedge e_j\}, \{e_i \wedge e_j \wedge e_k\}, \ldots, \{e_1 \wedge \ldots \wedge e_n\}. \tag{3}$$

3 Conformal Geometric Algebra

Conformal Geometric Algebra $G_{4,1,0}$ (CGA) can be used to treat conformal geometry in a very elegant way [3]. CGA allow us to represent the Euclidean vector space $\mathbb{R}^3$ in $\mathbb{R}^{4,1}$. This space has an orthonormal vector basis given by $\{e_i\}$ and $e_{ij} = e_i \wedge e_j$ are bivectorial bases and a bivector basis e_{23}, e_{31}, and e_{12}.

The unit Euclidean pseudo-scalar $I_e := e_1 \wedge e_2 \wedge e_3$, a pseudo-scalar $I = I_e E$, and the bivector or Minkowski plane $E := e_4 \wedge e_5 = e_4 e_5$ are used for computing Euclidean and conformal duals of multivectors.

A null can be defined as

$$e_\infty = e_4 + e_5, \;\; e_0 = \frac{1}{2}(e_4 - e_5), \tag{4}$$

where e_∞ is the point at infinity and e_0 is the origin point. This two null vector satisfies

$$e_\infty^2 = e_0^2 = 0, \;\; e_\infty \cdot e_0 = 1.$$

For a more complete treatment, the reader is referred to [3, 1, 7].

3.1 The Point

The vector $x_e \in \mathbb{R}^3$ representing a point after a conformal mapping is rewritten as

$$x_c \;\; = \;\; x_e + \frac{1}{2} x_e^2 e_\infty + e_0. \tag{5}$$

Given two conformal points x_c and y_c, we can define

$$x_c - y_c = (y_c \wedge x_c) \cdot e_\infty \tag{6}$$

and, consequently, the following equality

$$(x_c \wedge y_c + y_c \wedge z_c) \cdot e_\infty = (x_c \wedge z_c) \cdot e_\infty \tag{7}$$

is fulfilled as well.

3.2 Spheres and Planes

The equation of a sphere of radius ρ centered at point $p_e \in \mathbb{R}^3$ can be written as $(x_e - p_e)^2 = \rho^2$. Since $x_c \cdot y_c = -\frac{1}{2}(\mathrm{x}_e - \mathrm{y}_e)^2$, where x_e and y_e are the Euclidean components, and $x_c \cdot p_c = -\frac{1}{2}\rho^2$, we can rewrite the formula above in terms of homogeneous coordinates. Since $x_c \cdot e_\infty = -1$, we can factor the expression above to

$$x_c \cdot (p_c - \frac{1}{2}\rho^2 e_\infty) = 0. \tag{8}$$

This equation corresponds to the so-called inner product null space (IPNS) representation, which finally yields the simplified equation for the sphere as $s = p_c - \frac{1}{2}\rho^2 e_\infty$. Note from this equation that a point is just a sphere with a zero radius. Alternatively, the dual of the sphere is represented as a 4-vector $s^* = sI$. The advantage of the dual form is that the sphere can be directly computed from four points as

$$s^* = x_{c_1} \wedge x_{c_2} \wedge x_{c_3} \wedge x_{c_4}. \tag{9}$$

If we replace one of these points for the point at infinity, we get the equation of a 3D plane:

$$\pi^* = x_{c_1} \wedge x_{c_2} \wedge x_{c_3} \wedge e_\infty. \tag{10}$$

We put π in the standard IPNS form as follows:

$$\pi \quad = I\pi^* = \quad n + de_\infty, \tag{11}$$

where n is the normal vector and d represents the Hesse distance for the 3D space.

3.3 Circles and Lines

A circle z can be regarded as the intersection of two spheres s_1 and s_2 as $z = (s_1 \wedge s_2)$ in IPNS. The dual form of the circle can be expressed by three points lying on the circle, namely,

$$z^* = x_{c_1} \wedge x_{c_2} \wedge x_{c_3}. \tag{12}$$

Similar to the case of planes, lines can be defined by circles passing through the point at infinity as

$$L^* = x_{c_1} \wedge x_{c_2} \wedge e_\infty. \tag{13}$$

The standard IPNS form of the line can be expressed as

$$L = \mathbf{n}I_e - e_\infty \mathbf{m} I_e, \tag{14}$$

where $\mathbf{n}$ and $\mathbf{m}$ stand for the line orientation and moment, respectively. The line in the IPNS standard form is a bivector representing the six Plücker coordinates.

All these entities are useful to represent the parts of a robotic manipulator; for example, the line is used to express the action axes of each DOF of the robot.

4 Rigid Transformations

In this section we define the two rigid transformations used in this work, also we define the *reversion* of a multivector, which is used to apply a rigid transformation to any geometric entity of the algebra.

4.1 Reversion

The reversion of an r-grade multivector $A_r = \sum_{i=0}^{r} \langle A_r \rangle_i$ is defined as

$$\widetilde{A}_r = \sum_{i=0}^{r} (-1)^{\frac{i(i-1)}{2}} \langle A_r \rangle_i . \tag{15}$$

In fact, the reversion can be obtained by simply reversing the order of basis vectors making up the blades in a multivector and then rearranging them in their original order using the anticommutativity of the Clifford product [1].

4.2 Reflection

The combination of reflections of conformal geometric entities enables us to form other transformations. The reflection of a point x with respect to the plane π is equal to x minus twice the directed distance between the point and plane (see Fig. 1(a)). That is, $ref(x) = x - 2(\pi \cdot x)\pi^{-1}$. We get this expression by using the reflection $ref(x_c) = -\pi x_c \pi^{-1}$ and the property of the Clifford product of vectors $2(b \cdot a) = ab + ba$.

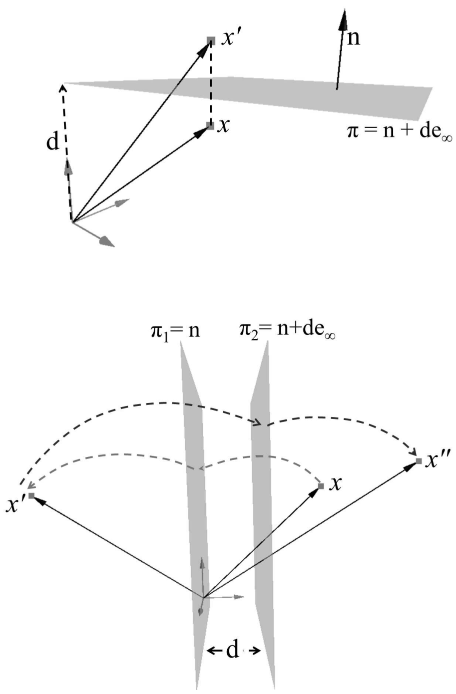

Fig. 1. (a) (top) Reflection of a point x with respect to the plane π, (b) (bottom) reflection about parallel planes

For an IPNS geometric entity Q, the reflection with respect to the plane π is given as

$$Q' = \pi Q \pi^{-1}. \quad (16)$$

4.3 Translation

The translation of conformal geometric entities can be done by carrying out two reflections in parallel planes π_1 and π_2 (see Fig. 1(b)). That is,

$$Q' = \underbrace{(\pi_2\pi_1)}_{T_a} Q \underbrace{(\pi_1^{-1}\pi_2^{-1})}_{\widetilde{T_a}} \quad (17)$$

$$T_a = (n + de_\infty)n = 1 + \frac{1}{2}ae_\infty = e^{\frac{a}{2}e_\infty} \quad (18)$$

with $a = 2dn$.

4.4 Rotation

The rotation is the product of two reflections at nonparallel planes that pass through the origin (see Fig. 2):

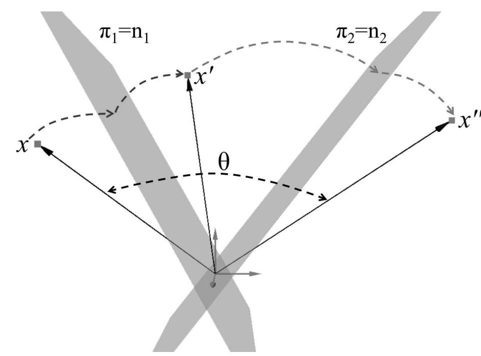

Fig. 2. Reflection about nonparallel planes

$$Q' = \underbrace{(\pi_2\pi_1)}_{R_\theta} Q \underbrace{(\pi_1^{-1}\pi_2^{-1})}_{\widetilde{R_\theta}}, \quad (19)$$

or by computing the conformal product of the normals of the planes:

$$R_\theta = n_2 n_1 = cos(\frac{\theta}{2}) - sin(\frac{\theta}{2})l = e^{-\frac{\theta}{2}l}, \quad (20)$$

with $l = n_2 \wedge n_1$, and θ twice the angle between the planes π_2 and π_1.

The screw motion, called *motor*, related to an arbitrary axis L is $M = TR\widetilde{T}$ and is applied in the same way as a rotor; that is,

$$Q' = \underbrace{(TR\widetilde{T})}_{M_\theta} Q \underbrace{(T\widetilde{R}\widetilde{T})}_{\widetilde{M_\theta}} \quad (21)$$

$$M_\theta = TR\widetilde{T} = cos(\frac{\theta}{2}) - sin(\frac{\theta}{2})L = e^{-\frac{\theta}{2}L}. \quad (22)$$

5 Kinematic Modeling of Manipulators

The *direct kinematics* for a serial robot is computed as a successive multiplication of motors given by

$$Q' = \left(\prod_{i=1}^{n} M_i Q \prod_{i=1}^{n} \widetilde{M}_{i=n-i+1} \right), \tag{23}$$

and it is valid for points (i.e. the position of the end-effector), lines (i.e. the orientation of the end-effector), planes, circles, and spheres, where the joint variable is a rotation $M_i = R_i = \exp^{\frac{-qL_r}{2}}$ for a revolutive joint, for a given angular position vector $q = [q_1 \ \dots \ q_n]^T$.

Differentiation of (23) gives the *differential kinematics* of the system for points and lines:

$$\begin{aligned} \dot{x}' &= J_x \dot{q}, \\ \dot{L}' &= J_L \dot{q}, \end{aligned} \tag{24}$$

with $\dot{q} = [\dot{q}_1 \ \dots \ \dot{q}_n]$, and

$$J_x = \left[x' \cdot L_1' \ \dots \ x' \cdot L_n' \right], \tag{25}$$

$$J_L = [\alpha_1 \ \dots \ \alpha n], \tag{26}$$

where

$$\begin{aligned} L_j' &= \left(\prod_{i=1}^{j-1} M_i \right) L_j \left(\prod_{i=1}^{j-1} \widetilde{M}_{j-i} \right), \\ \alpha_j &= L' \bar{\times} L_j', \end{aligned} \tag{27}$$

and L_j is the axis for the j^{th} joint in the initial position. Please refer to [8] for a more detailed explanation about the differentiation process.

6 Kinematic Control

Now the output tracking problem for the position x_p' and orientation L_p' of the end effector will be solved using the geometric algebra approach separately.

Fig. 3(a) shows a general scheme of the case study that we are solving, where the current orientation and position vectors for the end effector of a serial manipulator and for the target are shown. The control objective is to make the end effector and target positions, and orientations, equal by means of reconfiguring the structure of robot kinematics through the actuators of the joints.

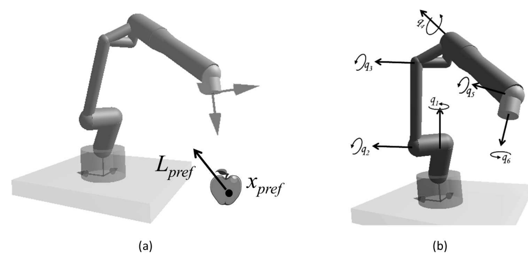

Fig. 3. (a) Current orientation and position vectors for the end effector of a serial manipulator and for the target, (b) robotic system composed by a 6-DOF manipulator

6.1 The Position Tracking Problem

A state-space model for the position of the end effector can be obtained as

$$\begin{aligned} \dot{x}_p' &= J_x u_1 \\ y_1 &= x_p', \end{aligned} \tag{28}$$

where y_1 is the output of the system, the control term is $u_1 = \dot{q}$, and the Jacobian J_x is defined as in (25).

Now, let $x_{ref}(t)$ be the reference for the position of the end effector expressed in conformal algebra. Omitting the parentheses of the reference, the tracking error is given by

$$\epsilon_p = \left(x_{ref} \wedge x_p' \right) \cdot e_\infty. \tag{29}$$

Differentiation of (29) yields

$$\begin{aligned} \dot{\epsilon}_p &= \left(\dot{x}_{ref} \wedge x_p' \right) \cdot e_\infty + \left(x_{ref} \wedge \dot{x}_p' \right) \cdot e_\infty \\ &= \left(x_{ref} \wedge (J_x u_1) + \dot{x}_{ref} \wedge x_p' \right) \cdot e_\infty. \end{aligned} \tag{30}$$

Assuming that we know the derivative $\dot{x}_{ref}$, then the control law

$$u_1 = -J_x^+ \left[\dot{x}_{ref} \wedge (k_1 \epsilon_p) \right] \cdot e_\infty \tag{31}$$

is proposed to stabilize system (30), where k_1 is a constant.

The closed-loop system defined by (30) and (31) results in

$$\dot{\epsilon}_p = \left(-x_{ref} \wedge (k_1 \epsilon_p - \dot{x}_{ref}) + \dot{x}_{ref} \wedge x_p' \right) \cdot e_\infty. \tag{32}$$

Then, using distributivity and $x \wedge x = 0$ yields

$$\begin{aligned} \dot{\epsilon}_p &= \left(-x_{ref} \wedge \left(k_1 \left(x_p' - x_{ref} \right) - \dot{x}_{ref} \right) + \dot{x}_{ref} \wedge x_p' \right) \cdot e_\infty \\ &= \left(-k_1 x_{ref} \wedge x_p' + x_{ref} \wedge \dot{x}_{ref} + \dot{x}_{ref} \wedge x_p' \right) \cdot e_\infty. \end{aligned} \tag{33}$$

Now, applying (7) results in

$$\dot{\epsilon}_p = \left(\left(-k_1+1\right) x_{ref} \wedge x'_p\right) \cdot e_\infty = \left(-k_1+1\right) \epsilon_p. \quad (34)$$

Now, consider the positive definite candidate Lyapunov function [2] given by

$$V_{\epsilon_p} = \frac{1}{2}\left[\left(\epsilon_p \wedge E\right) E\right]^2 \quad (35)$$

to prove stability of (32). Differentiation of (35) yields

$$\begin{aligned} \dot{V}_{\epsilon_p} &= \left[\left(\epsilon_p \wedge E\right) E\right]\left[\dot{\epsilon}_p \wedge E\right] E \quad (36) \\ &= \left[\left(\epsilon_p \wedge E\right) E\right]\left[\left(-\left(k_1-1\right)\epsilon_p \wedge E\right) E\right] \\ &= -2\left(k_1-1\right) V_{\epsilon_p}, \end{aligned}$$

which is a negative definite function for $k_1 > 1$. Therefore, the origin of the system (32) is a globally exponentially stable equilibrium point, that is, $\lim_{t\to\infty} x'_p = x_{ref}$, and the control objective is fulfilled. It is clear that the system is linear and it is not necessary to have a Lyapunov function to prove the stability of the system, but with this first approach in a future we will propose Lyapunov functions using geometric entities.

6.2 Orientation Tracking Problem

Similar to the position tracking problem, a state-space model for the orientation of the end effector can be obtained as

$$\begin{aligned} \dot{L}'_p &= J_L u_2 \quad (37) \\ y_2 &= L'_p, \end{aligned}$$

where y_2 is the output of the system, $u_2 = \dot{q}$, and the Jacobian J_L is defined as in (25).

Now, let $L_{ref}(t)$ be the reference line for the orientation of the end effector expressed in conformal algebra. The tracking error is defined as

$$\epsilon_L = L'_p - L_{ref}, \quad (38)$$

which represents the difference between the line of the end effector L'_p and the reference L_{ref}.

Differentiation of (38), and using (38), yields

$$\dot{\epsilon}_L = \dot{L}'_p - \dot{L}_{ref} = J_L u_2 - \dot{L}_{ref}. \quad (39)$$

Assuming that the derivative $\dot{L}_{ref}$ is known, the control law

$$u_2 = -J_L^+\left[k_2\epsilon_L - \dot{L}_{ref}\right] \quad (40)$$

is proposed to stabilize system (39), where k_2 is a constant.

The closed-loop system is obtained using (39)-(40) as

$$\dot{\epsilon}_L = -k_2\epsilon_L \quad (41)$$

and using the following positive definite candidate Lyapunov function

$$V_{\epsilon_L} = \frac{1}{2}\epsilon_L^2 \quad (42)$$

to prove stability of (41).

Differentiation of (42) results in $\dot{V}_{\epsilon_L} = \epsilon_L\dot{\epsilon}_L = -k_2\epsilon_L^2$, which is a negative definite function for $k_2 > 1$. Therefore, the origin of the system (39) is a globally exponentially stable equilibrium point, that is, $\lim_{t\to\infty} \epsilon_L = 0$, and the control objective is fulfilled.

To avoid singularities in computing J_x^+ and J_L^+, we use the robust damped least-square (DLS) method proposed in [5], where the pseudo-inverse of a matrix J is defined by

$$J^+ = J^T\left(JJ^T + \alpha I_m\right)^{-1}, \quad (43)$$

where I_m is an identity matrix with the same dimension as JJ^T and α is a positive damping factor given by

$$\alpha = \begin{cases} \alpha_0\left(1-\left(h/h^s\right)\right), & \text{if } h < h^s, \\ 0 & \text{otherwise,} \end{cases}$$

where h^s denotes the threshold value, α_0 is the value of damping factor at singular points, and h is defined as

$$h(\theta) = \sqrt{det\left(JJ^T\right)}.$$

With this adaptive α, it is possible to avoid singularities, without affecting the solved $\dot{\theta}$, because α is effective only when the configuration is near a singularity.

ISSN 2007-9737

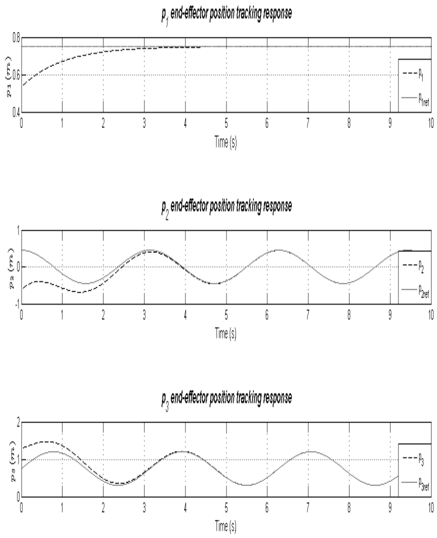

Fig. 4. Euclidean components for the position of the end effector of the 6-DOF manipulator and their references

7 Simulations

Consider the system shown in Fig. 3(b), which is composed of a serial manipulator of 6 DOFs. For a given target, the end effector of the 6-DOF manipulator must realize position tracking of the target. First, the kinematic model of the manipulator will be defined. Then the parameters of the proposed controllers are determined. Finally, a simulation of the performance of the closed-loop system is presented.

7.1 6-DOF Manipulator

The position kinematic model for the 6-DOF manipulator is defined entirely by the following axes of rotation and lengths of the links:

$$\begin{aligned} &L_1 = e_{12}, && L_2 = e_{31} + e_\infty \left(x_{1e} \cdot e_{31} \right), \\ &L_3 = e_{13} + e_\infty \left(x_{2e} \cdot e_{13} \right), && L_4 = e_{23} + e_\infty \left(x_{3e} \cdot e_{23} \right), \\ &L_5 = e_{13} + e_\infty \left(x_{4e} \cdot e_{13} \right), && L_6 = e_{12} + e_\infty \left(x_{4e} \cdot e_{12} \right), \end{aligned} \quad (44)$$

where x_{ie}, $i = 1, ...6$, are the vectors that define the initial position of each joint of the manipulator.

The differential kinematics are defined by the Jacobians defined as

$$\begin{aligned} J_x &= \left[x'_p \cdot L'_1, x'_p \cdot L'_2, \cdots, x'_p \cdot L'_5, x'_p, \cdot L'_6 \right], \\ J_L &= \left[\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5, \alpha_6 \right], \end{aligned} \quad (45)$$

where x'_p and L'_p are defined by Eq. (23) and α_i, $i = 1...6$ are given by Eq. (27).

7.1.1 Applied Controllers

The pose control term for the 6-DOF manipulator is defined as

$$u_1 = \left[\dot{q}_1, \dot{q}_2, \dot{q}_3, \dot{q}_4, \dot{q}_5, \dot{q}_6 \right]^T$$

and is obtained via Eqs. (29) and (31). The control gain for the position was selected as $k_1 = 1$. The position reference vector used was

$$\begin{aligned} x_{ref} &= \left[0.75, 0.45 \cos\left(2t\right), 0.75 + 0.45 \sin\left(2t\right) \right]^T, \\ \dot{x}_{ref} &= \left[0, -0.9 \sin\left(2t\right), 0.9 \cos\left(2t\right) \right]^T. \end{aligned} \quad (46)$$

On the other hand, the control gain for the orientation control term was selected as $k_2 = 2$, and the reference vector for the orientation used was

$$L_{ref} = \left[0, 0, -1 \right]^T, \quad \dot{L}_{ref} = \left[0, 0, 0 \right]^T.$$

7.1.2 Simulation Results

The simulation processes was developed in two steps. First, the differential kinematic model and the controllers for the robotic system were programmed in Matlab [4] using our own conformal geometric libraries, and the response for the closed-loop system was obtained. Then the data on the joint variables obtained from Matlab were used in a 3D model of the robotic system developed in CLUCalc [6], in order to obtain a better visual appreciation of the closed-loop system's behavior.

The following figures show the simulation response of the position and orientation tracking for the 6-DOF manipulator using the control laws proposed. Fig. 4 shows the tracking response for the position of the end effector in each Euclidean component of the work space. One can note that

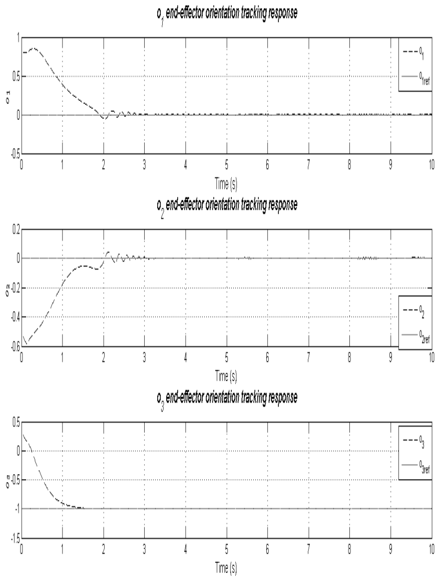

Fig. 5. Euclidean components for the orientation of the end effector of the manipulator and their references

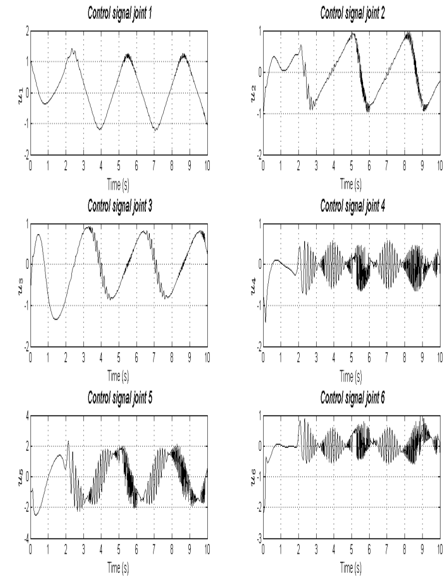

Fig. 6. Control values for position and orientation tracking of the end effector of the 6-DOF manipulator

the objective of control is fulfilled. Similarly, in Fig. 5, one can observe the orientation tracking performance for each component that defines the orientation of the end effector. Also, in Fig. 6, the values of the control signal are shown. Note that there are some high-frequency components on the control signals; these components are due to the use of the pseudo-inverse matrix defined in (43).

Finally, a sequence of images from the simulation realized in CLUCalc is shown in Fig. 7. Here the position tracking realized by the 6-DOF manipulator can be appreciated.

7.2 Biped Humanoid

In this section, we use the same methodology of modeling presented above using the conformal geometric algebra framework. We obtain a model for the legs of a biped robot, dividing the problem in two models of manipulators, where the end effector for each manipulator is located at the center of mass (center of the hip). First, the kinematic

Fig. 7. Sequence of images of the simulation for the robotic system using CLUCalc

models of the legs will be defined. Then the parameters of the proposed controller are determined. Finally, the performance of the simulation of the closed-loop system is presented. The kinematic

ISSN 2007-9737

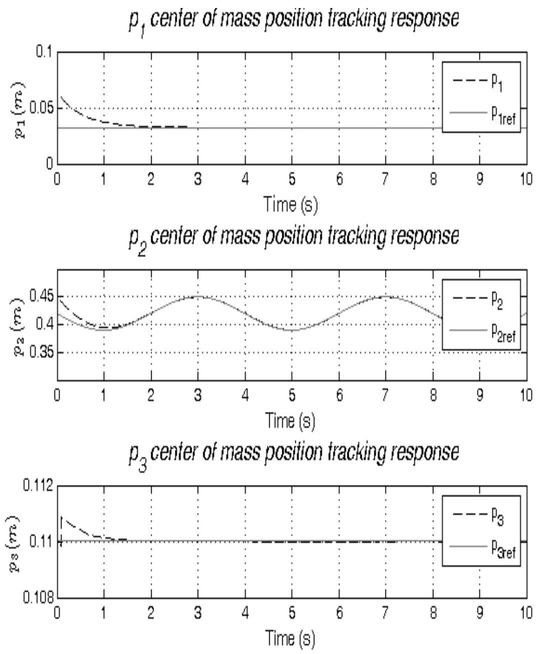

Fig. 8. Euclidean components of the center of mass and their references

model for the legs of a biped robot (6 DOFs per leg) is defined by the following axes of rotation and lengths of each link:

$$\begin{aligned} L_1 &= e_{32} + e_\infty(x_1 \cdot e_{32}), & L_4 &= e_{13} + e_\infty(x_4 \cdot e_{13}), \\ L_2 &= e_{12} + e_\infty(x_2 \cdot e_{12}), & L_5 &= e_{32} + e_\infty(x_5 \cdot e_{32}), \\ L_3 &= e_{21} + e_\infty(x_3 \cdot e_{21}), & L_6 &= e_{12} + e_\infty(x_6 \cdot e_{12}), \end{aligned} \tag{47}$$

where x_i, $i = 1, ..., 6$ are the vectors that define the initial position of each joint of the right leg of a biped robot.

7.2.1 Applied Controllers

The pose control term for a leg of the biped robot is defined as $u_1 = [q_1, \ldots, q_6]^T$ and is obtained via Eqs. (29) and (31). The state-space model is obtained by the differential kinematics and is defined in Eqs. (46). The control gain was selected as $K_1 = [1.9, 1.9, 1.9, 3.2, 3.2, 3.2]^T$. The reference position vectors used to move the center position mass of the biped robot was

$$x_{ref} = [0.3238, 4.2 - 0.3\sin(0.5\pi t), 1.1]^T,$$

where x_{ref} is the reference vector and the reference vector for the orientation was

$$L_{ref} = [0, 0, 1]^T.$$

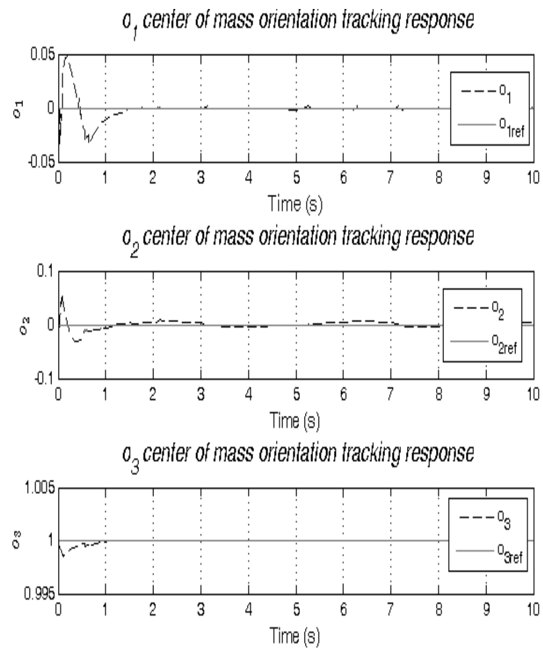

Fig. 9. Euclidean components of the orientation of the biped robot and their references

7.2.2 Simulation Results

Next, we present the same simulation using differential kinematic, MatLab, conformal geometric algebra, and CLUCalc [6] for the 6-DOF manipulator robot. We used the model of a 6-DOF manipulator for the kinematic model of the legs of the biped robot. The first goal is to raise and lower the center of mass of the biped robot repeatedly.

The following figures show the simulation response of the position and orientation tracking for the legs of a biped robot using the control laws proposed. Fig. 8 shows the tracking response of the position of the end effector (center of mass) in each Euclidean component of the work space. One can note that the objective of control is fulfilled. Similarly in Fig. 9, one can observe the orientation

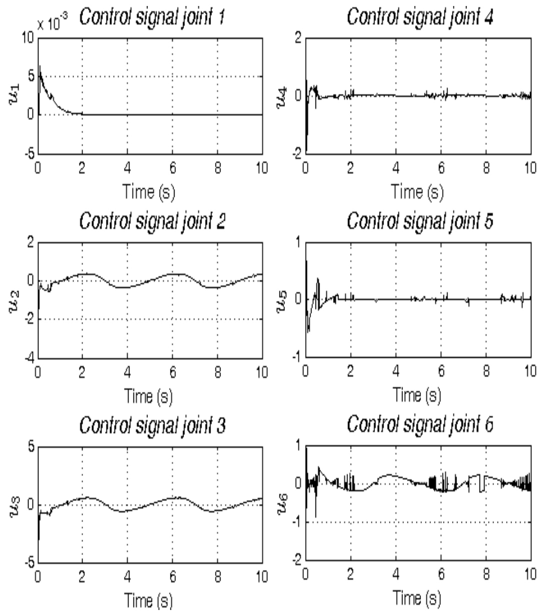

Fig. 10. Control values of position and orientation of the biped robot

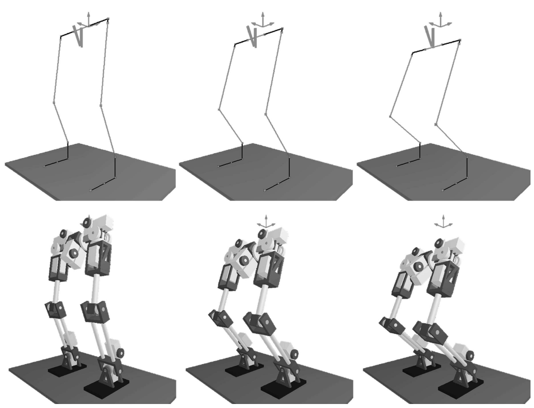

Fig. 11. Sequence of images of the simulation of a vertical motion for the biped robot using CLUCalc (x_{ref})

tracking performance for each component that defines the orientation of the center of mass of the biped robot. Also, in Fig. 10, the values of the control signal are shown. Also, simulations for the reference vectors $x_{ref_1} = [0.32, 4, 1.1 - .4\sin(2t)]^T$ and $x_{ref_2} = [0.32, 4 - 0.2\cos(2t), 1.1 - 0.2\sin(2t)]^T$ was developed and are showed in Figs. 12 and 13, respectively.

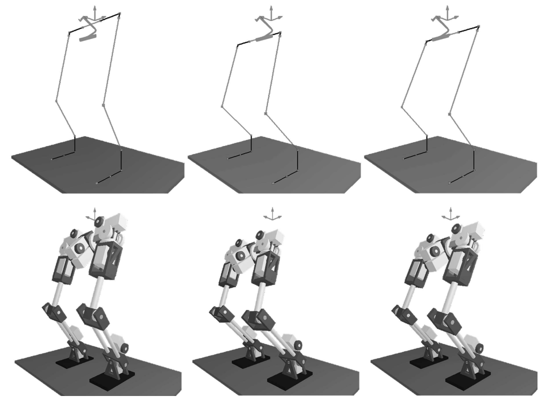

Fig. 12. Sequence of images of the simulation of a lateral motion for the biped robot using CLUCalc (x_{ref_1})

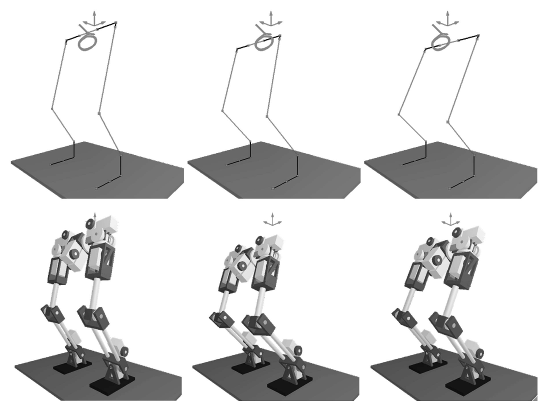

Fig. 13. Sequence of images of the simulation of an elliptical motion for the biped robot using CLUCalc (x_{ref_2})

8 Conclusions

This paper presents the modeling and control of a 6-DOF manipulator and legs of biped robot using conformal geometric algebra (CGA). The orientation tracking and position tracking problem is defined entirely in this mathematical framework, as well as the stability analysis in both cases. The use of CGA to define the tracking error opens a new area on the control of the manipulator because we are able to define the error between geometric primitives. CGA provides a descriptive language to represent geometric primitives and their rigid transformations, facilitating the procedure of kinematics chain calculation in serial manipulators. These features are provided exclusively by the use of CGA. This method of control can be easily extended to advanced nonlinear control strategies such as *sliding modes*, *adaptive control*, and so on. Due to the use of the adaptive pseudo-inverse of the Jacobian, the control law presents high-frequency components. For this reason, in our future work we will make a new stability analysis, to propose a control law that is a smooth signal and rejects these high frequencies. The simulation obtained using CLUCalc provides a better visualization for the response of the closed-loop system. Thus, we can conclude that CGA is a good framework for kinematic modeling, control, and visualization of robotic manipulators.

Acknowledgements

This work was supported in part by Project SEP-CONACYT "Métodos Geométricos y Cognitivos para la percepción, aprendizaje, control y acción de humanoides" under Grant 82084 and PhD scholarships No. 219316 and No. 28824.

References

1. **Bayro-Corrochano, E.** (**2010**). *Geometric Computing: for Wavelet Transforms, Robot Vision, Learning, Control and Action*. Springer Verlag, London.
2. **Khalil, H.** (**1996**). *Nonlinear Systems*. Prentice-Hall.
3. **Li, H., Hestenes, D., & Rockwood, A.** (**2001**). Generalized homogeneous coordinates for computational geometry. *Geometric Computing with Clifford Algebras*, Springer-Verlag Heidelberg, pp. 27–52.
4. **MathWorks** (**2008**). *Matlab, Release Notes*.
5. **Nakamura, Y. & Hanafusa, H.** (**1986**). Inverse kinematics solution with singularity robustness for robot manipulator control. *ASME Journal of Dynamic Systems, Measurement and Control*, Vol. 108, pp. 163 – 171.
6. **Perwass, C.** (**2010**). *CLUCalc, Interactive Visualization*.
7. **Perwass, C. & Hildenbrand, D.** (**2004**). *Aspects of Geometric Algebra in Euclidean, Projective and Conformal Space*.
8. **Zamora-Esquivel, J. & Bayro-Corrochano, E.** (**2006**). Kinematics and diferential kinematics of binocular robot head. *IEEE International Conference on Robotics and Automation*, pp. 4130 – 4135.

Oscar Carbajal-Espinosa received his B.Sc. degree in Electrical Engineering from the Technological Institute of Merida, Yucatan, Mexico, the M.Sc. and Ph.D. degrees in Electrical Engineering from the Advanced Studies and Research Center of the National Polytechnic Institute, Campus Guadalajara, Mexico, in 2008, 2011, and 2015, respectively. His research interests include non-linear robust control design, sliding modes and its applications to robotics, modeling and control of humanoid robots, conformal geometric algebra and its applications to robotics and mechanical systems.

Luis González-Jiménez was born in Guasave, Sinaloa, Mexico. He received his B.Sc. degree in Electronic Engineering from Instituto Tecnológico de Sonora (ITSON) in 2005, his M.Sc. and Ph.D. degrees in Electrical Engineering from the Advanced Studies and Research Center of the National Polytechnic Institute (CINVESTAV-IPN), Guadalajara Campus, Mexico, in 2007 and 2011, respectively. Since 2013, he is a Research Professor at ITESO. He has published more than 10 refereed journal and conference papers. His research interests include robust automatic control, computer vision, and robotics.

Jose Oviedo-Barriga received his B.Sc. degree in Electronic Engineering from the Technological Institute of Orizaba, Veracruz, Mexico, the M.Sc. degree in Science of Electronic Engineering from the Autonomous University of Puebla, Puebla, Mexico, and Ph.D. degree in Electrical Engineering from the Advanced Studies and Research Center of the National Polytechnic Institute, Campus Guadalajara, Mexico, in 2004, 2007, and 2013, respectively. His research interests include non-linear robust control design, sliding modes and its applications to robotics, modeling control of humanoid robots. Now he works at the Veracurzana University as full time Professor at the Faculty of Engineering.

Bernardino Castillo-Toledo received his B.Sc. degree in Electrical Engineering from the National Polytechnic Institute (IPN), Mexico City, Mexico, the M.Sc. degree from the Research and Advanced Studies Center of the National Polytechnic

ISSN 2007-9737

Institute (CINVESTAV-IPN), Guadalajara, Mexico, and the Ph.D. degree from the University of Rome "La Sapienza," Rome, Italy, in 1981, 1985, and 1992, respectively. He was a Lecturer with the School of Electrical and Mechanics Engineering, IPN, from 1985 to 1989. From 1985 to 1995, he was with the Automatic Control Section, Department of Electrical Engineering, CINVESTAV-IPN, and since 1995, with CINVESTAV-IPN, Guadalajara Campus. He held several research stays with the University of Rome "La Sapienza" and University of L'Aquila, L'Aquila, Italy, and was a visiting Professor at the Laboratoire d'Automatique et d'Analyse des Systemes, French Council for Scientific Research, from 1999 to 2000, and at the University of Compiègne, Compiegne, France, during the first semester of 2002. His main research interests include nonlinear control design, the robust regulation problem, and application of artificial neural networks and fuzzy logic techniques to control and fault diagnosis of dynamical systems.

Alexander Loukianov received his B.Sc. degree from the Polytechnic Institute, Moscow, Russia, in 1975 and the Ph.D. degree in Automatic Control from the Institute of Control Sciences, Russian Academy of Sciences, Moscow, Russia, in 1985. He was with the Institute of Control Sciences in 1978 and was the Head of the Discontinuous Control Systems Laboratory from 1994 to 1995. From 1995 to 1997, he held a visiting position at the University of East London, London, U.K., and since April 1997, he has been with the Research and Advanced Studies Center of the National Polytechnic Institute, Guadalajara Campus, Mexico, as a Professor of Electrical Engineering graduate programs. From 1992 to 1995, he was in charge of an industrial project between the institute and the largest Russian car plant and also several international projects supported by the International Association and Specific International Scientific Cooperation Activities-COPERNICUS, Brussels. He has published more than 100 technical papers in international journals, books, and conferences and has been a reviewer for various international journals and conferences. His research interests include nonlinear system robust control and variable structure systems with sliding modes as applied to dynamical plants with delay, electric drives, and power systems control, robotics, spice, and automotive control.

Eduardo Bayro-Corrochano gained his Ph.D. in Cognitive Computer Science in 1993 from the University of Wales at Cardiff. From 1995 to 1999 he was Researcher and Lecturer at the Institute for Computer Science, Christian Albrechts University, Kiel, Germany, working on applications of geometric Clifford algebra to cognitive systems. At present he is a full Professor at CINVESTAV, Guadalajara campus, Mexico, Department of Electrical Engineering and Computer Science. Since 2006 he has been a fellow of the IAPR society. His current research interest focuses on geometric methods for artificial perception and action systems. It includes geometric neural networks, visually guided robotics, humanoids, color image processing, Lie bivector algebras for early vision and robot maneuvering. He developed the quaternion wavelet transform for quaternion multi-resolution analysis using the phase concept. He conducts research in robot medical image processing as well. He is associate editor of Journal Robotica, Journal of Advanced Robotic Systems, Journal of Pattern Recognition, and Journal of Mathematical Imaging. He is editor and author of the following books: Geometric Computing for Perception Action Systems, E. Bayro-Corrochano, Springer Verlag, 2001; Geometric Algebra with Applications in Science and Engineering, E. Bayro-Corrochano and G. Sobczyk (Eds.), Birkhauser 2001; Handbook of Geometric Computing for Pattern Recognition, Computer Vision, Neurocomputing and Robotics, E. Bayro-Corrochano, Springer Verlag, 2005, Geometric Computing for Wavelet Transforms, Robot Vision, Learning, Control and Action. E. Bayro-Corrochano, Springer Verlag, London, 2009; Geometric Algebra Computing for Engineering and Computer Science, E. Bayro-Corrochano and G. Sheuermann, Springer Verlag, London, 2009. He has published over 150 refereed journal and conference papers and book chapters.

Article received on 03/12/2014; accepted on 16/04/2015.
Corresponding author is Oscar Carbajal-Espinosa.