The 2010 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 18-22, 2010, Taipei, Taiwan

Parallel Forward Dynamics: a geometric approach

Julio Zamora-Esquivel and Eduardo Bayro-Corrochano

***Abstract*— The authors present a new algorithm to compute the forward Dynamics of n degrees of freedom serial kinematic chains, which is faster than the classical approaches. This algorithm was created rewriting the Lagrange equation in terms of lines and points in the framework of conformal geometric algebra, which allows having a new equation to compute the dynamics with less number of products. This algorithm not only performs less computations but it also takes the advantages of the newest multi core architectures by computing the dynamics in parallel.**

I. INTRODUCTION

As demand of more computation power has grown, companies which designs processor has chosen to develop multicore architectures to satisfy such needs. With each new design of multicore processors it is usual that the number of cores is increased and with this, the need of algorithms which take full advantage of this highly parallel environment also grows rapidly in importance.

On the other hand, in the field of robotics there is needed to calculate the dynamics of serial robots for controlling, identifying and simulating of robots. Currently there are a large number of algorithms to calculate the dynamics but most of them work sequentially. For example, the solutions presented in [8] have a similar complexity of $O(n^3)$. In this paper the authors introduce a new algorithm that allows the computation of each matrix evolved in the computation of robot dynamics in an independent way, reducing the complexity to $O(n)$ with the advantage that only a reduced number of operations on each processor is performed.

This algorithm also allows the modification of the topology of the robot, that is, the position and direction of the axis, the mass centers, switch between prismatic and revoluted joints, on the fly. Also, this algorithm can be used with parallel robots following the procedure describe in [7]

II. THE GEOMETRIC ALGEBRA OF N-D SPACE

In this paper we will specify a geometric algebra $\mathcal{G}_n$ of the n dimensional space by $\mathcal{G}_{p,q,r}$, where p, q and r stand for the number of basis vector which squares to 1, -1 and 0 respectively and fulfill $n = p + q + r$.

We will use e_i to denote the vector basis i. In a Geometric algebra $G_{p,q,r}$, the geometric product of two basis vector is defined as

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

Julio Zamora-Esquivel Intel Tecnologia de Mexico, S.A de C.V.,VPG GDC julio.c.zamora.esquivel@intel.com

Eduardo Bayro-Corrochano CINVESTAV, Unit Guadalajara, Mexico, address edb@gdl.cinvestav.mx

This leads to a basis for the entire algebra:

$$\{1\}, \{e_i\}, \{e_i \wedge e_j\}, \ldots, \{e_1 \wedge e_2 \wedge \ldots \wedge e_n\} \tag{1}$$

Any multivector can be expressed in terms of this basis. In the n-D space there are multivectors of grade 0 (scalars), grade 1 (vectors), grade 2 (bivectors),... up to grade n. Any two such multivectors can be multiplied using the geometric product. Consider two multivectors $\boldsymbol{A_r}$ and $\boldsymbol{B_s}$ of grades $\boldsymbol{r}$ and $\boldsymbol{s}$ respectively. The geometric product of $\boldsymbol{A_r}$ and $\boldsymbol{B_s}$ can be written as

$$\boldsymbol{A_r B_s} = \langle \boldsymbol{AB} \rangle_{\boldsymbol{r+s}} + \langle \boldsymbol{AB} \rangle_{\boldsymbol{r+s-2}} + \ldots + \langle \boldsymbol{AB} \rangle_{\boldsymbol{|r-s|}} \tag{2}$$

where $\langle\rangle_{\boldsymbol{t}}$ is used to denote the $\boldsymbol{t}$-grade part of multivector.

$$A_r \cdot B_s = \langle AB \rangle_{|r-s|} \tag{3}$$
$$A_r \wedge B_s = \langle AB \rangle_{|r+s|} \tag{4}$$
$$A_r \bar{\times} B_s = \langle AB \rangle_{|r+s-2|}. \tag{5}$$

III. CONFORMAL GEOMETRY

The Geometric algebra $G_{4,1}$ can be used to treat conformal geometry in a very elegant way. The Euclidean vector space $\mathbb{R}^3$ is represented 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 basis where e_{23}, e_{31} and e_{12} correspond to the Hamilton basis. The pseudo-scalar $I_c = e_1 \wedge \ldots \wedge e_5$ and $E = e_4 \wedge e_5$ are used to compute duals of multivectors.

The conformal geometry is related to a stereographic projection in Euclidean space for $\mathbb{R}^1$ the conformal maping x is given by

$$x = 2\frac{x_e}{x_e^2+1}e_1 + \frac{x_e^2-1}{x_e^2+1}e_4 + e_5. \tag{6}$$

From (6) we can infer the point at infinity and the origin point as

$$e_\infty = \lim_{x_e \to \infty} \{x\} = e_4 + e_5, \tag{7}$$

$$e_o = \frac{1}{2} \lim_{x_e \to 0} \{x\} = \frac{1}{2}(-e_4 + e_5), \tag{8}$$

Note that (6) can be rewritten to

$$x \quad = \quad x_e + \frac{1}{2}x_e^2 e_\infty + e_o, \tag{9}$$

A. Line

Lines can be defined as circles passing through the point at infinity (10),this is showed in the figure 1.

$$L^* = x_1 \wedge x_2 \wedge e_\infty = m e_\infty + n E, \tag{10}$$

where n is a direction's vector and m is the momentum. The resulting line has six parameters so called Plucker coordinates, but just four degrees of freedom.

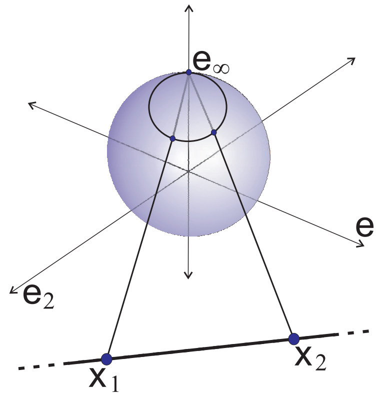

Fig. 1. Line projected to the unitary sphere.

IV. Rigid Transformations

We can express rigid transformations in conformal geometry carrying out reflections between planes.

A. Translation

The translation is the product of two reflections between parallel planes

$$T = 1 + \frac{1}{2} a e_\infty = e^{-\frac{a}{2} e_\infty} \tag{11}$$

here a represents the translation vector, any geometric entity can be translated doing $x' = Tx\widetilde{T}$

B. Rotation

The rotation is the product of two reflections between nonparallel planes

$$R = \cos\left(\frac{\theta}{2}\right) - \sin\left(\frac{\theta}{2}\right) l = e^{-\frac{\theta}{2} l} \tag{12}$$

here l denotes the rotation axis. The screw motion *called motor* $M = TR\widetilde{T}$ represents the rotation related to an arbitrary axis L defined on (10)

$$M = e^{-\frac{q}{2} L} \tag{13}$$

where q represents the rotation angle or the translation in case of L at infinity. Any geometric entity can be rotated doing $x' = Mx\widetilde{M}$

V. Kinematics

The direct kinematics for serial robot arms is a succession of motors and it is valid for points, lines, planes, circles and spheres more information on [3].

$$x'_j = \prod_{i=1}^{j} M_i x_j \prod_{i=1}^{j} \widetilde{M}_{j-i+1} \tag{14}$$

similarly L' is defined in terms of L as follows

$$L'_j = \prod_{i=1}^{j-1} M_i L_j \prod_{i=1}^{j-1} \widetilde{M}_{j-i}, \tag{15}$$

Differential kinematics equation:

$$\dot{x}'_j = \sum_{i=1}^{j} \left[x'_j \cdot L'_i \right] \dot{q}_i, \tag{16}$$

is explained on [4]

VI. Dynamics

In this section we describe the equations of kinetic and potential energy in terms of geometric algebra. Based on these equations and using the Lagrange equation we synthesize the dynamic model of any n-degrees of freedom serial robot.

A. Kinetic Energy

We introduce in our analysis the mass center in order to formulate an expression that describes the kinetic energy of a system of particles.

1) Kinetic energy of a system of particles.: We are considering a system with n particles showed in the figure 2. The total relative kinetic energy K of the system is given by

$$K = \sum_{i=1}^{n} \frac{1}{2} m_i V_i^2. \tag{17}$$

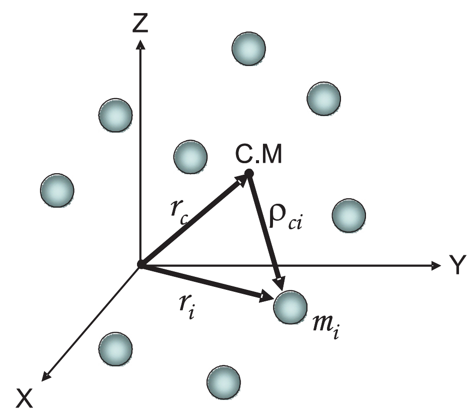

Fig. 2. System of particles with their mass center.

Now we will rewrite (17) to introduce the mass center. Here r_i represents the distance to the particle, r_c the distance to the mass center and ρ_{ci} the distance from the mass center to the particle.

$$r_i = r_c + \rho_{ci}. \tag{18}$$

The time derivative of (18) is

$$\dot{r}_i = \dot{r}_c + \dot{\rho}_{ci}. \tag{19}$$

Therefore, the velocity equation of the i-th particle ($\dot{\rho}_{ci}$) with respect to the mass center is given by

$$V_i = V_c + \dot{\rho}_{ci}. \tag{20}$$

By substitution of the equation 20 in the expression of kinetic energy 17, we obtain.

$$\begin{aligned} K &= \sum_{i=1}^{n} \frac{1}{2} m_i (V_c + \dot{\rho}_{ci})^2 \qquad (21) \\ &= \frac{1}{2} \sum_{i=1}^{n} m_i (V_c)^2 + \sum_{i=1}^{n} m_i V_c \dot{\rho}_{ci} + \frac{1}{2} \sum_{i=1}^{n} m_i (\dot{\rho}_{ci})^2. \end{aligned}$$

As V_c is not related to the sum index i we can extract it.

$$K = \frac{1}{2}V_c^2(\sum_{i=1}^{n} m_i) + V_c \frac{d}{dt}\sum_{i=1}^{n} m_i \rho_{ci} + \frac{1}{2}\sum_{i=1}^{n} m_i \dot{\rho}_{ci}^2. \quad (22)$$

Being $m = \sum_{i=1}^{n} m_i$ the total mass of the system and considering that $\sum_{i=1}^{n} m_i \rho_{ci}$ is by definition equal to zero

$$K = \frac{1}{2}mV_c^2 + \frac{1}{2}\sum_{i=1}^{n} m_i \dot{\rho}_{ci}^2. \quad (23)$$

As conclusion, we see *Kinetic energy* with respect to a reference system could be considered as the sum of two parts: *(1) the kinetic energy of total mass moving respect to this reference system at the same velocity*, plus *(2) the kinetic energy of the particles moving respect to the mass center (momentum of inertia)*.

2) The kinetic energy of a robot arm: We denote with x_i to the mass center i in its initial position and with x_i' the mass center in function of joints variables. Similarly we will denote the joints axis i as L_i and the joints axis i in function of the joints variables as L_i'.

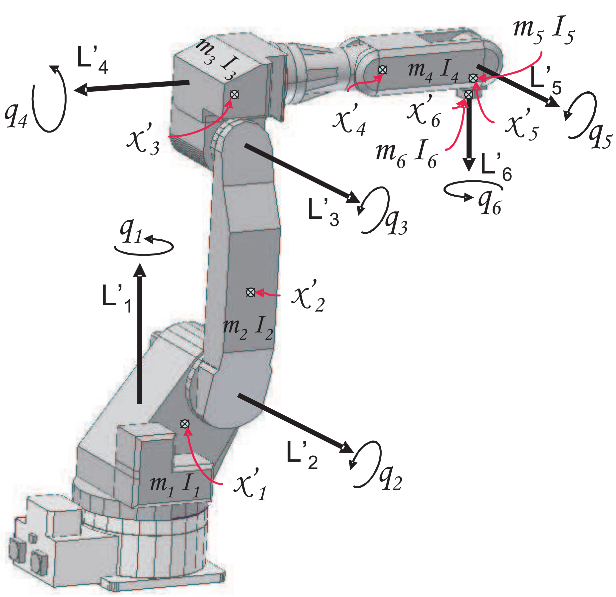

Fig. 3. Brazo AdepdtSix600.

Recalling the direct kinematics equation 14 that relates x_i with x_i' and L with L_i' and are written using conformal geometric algebras as.

$$x_i' = M_1 M_2 \cdots M_i x_i \tilde{M}_i \cdots \tilde{M}_2 \tilde{M}_1, \quad (24)$$
$$L_i' = M_1 M_2 \cdots M_{i-1} L_i \tilde{M}_{i-1} \cdots \tilde{M}_2 \tilde{M}_1. \quad (25)$$

We have seen that the kinetic energy is equal to the sum of the energy related to the velocity of mas center and the energy related to the momentum of inertia. So the kinetic energy of the link i is computed as

$$K_i = \frac{1}{2}m_i \dot{x}_i'^2 + \frac{1}{2}I_i\left(\sum_{j=1}^{i}\dot{q}_j\right)^2. \quad (26)$$

where I_i is the inertia of the link i and $\dot{x}_i'$ represents the velocity of the mass center x_i'. The velocity of the mass center is computed using the equation of differential kinematics 16 explained in the past section

$$\dot{x}'_i = x_i' \cdot \left(\sum_{j=1}^{i} L_j' \dot{q}_j\right). \quad (27)$$

Replacing the equation 27 in 26 we have the expression of kinetic energy in conformal geometric algebra

$$K_i = \frac{1}{2}m_i\left[x_i' \cdot \left(\sum_{j=1}^{i} L_j' \dot{q}_j\right)\right]^2 + \frac{1}{2}I_i\left(\sum_{j=1}^{i}\dot{q}_j\right)^2. \quad (28)$$

The total kinetic energy on the arm is given by the expression $\sum_{i=1}^{n} K_i$, where n is the number of degrees of freedom. In order to simplify the explanation we will separate the kinetic energy $K = K_v + K_I$ in two components K_v and K_I defined as

$$K_v = \frac{1}{2}\sum_{i=1}^{n} m_i\left[x_i' \cdot \left(\sum_{j=1}^{i} L_j' \dot{q}_j\right)\right]^2, \quad (29)$$

$$K_I = \frac{1}{2}\sum_{i=1}^{n} I_i\left(\sum_{j=1}^{i}\dot{q}_j\right)^2. \quad (30)$$

We will attend firstly K_v and later K_I. The objective is simplify the expression of of total kinetic energy in the arm

$$K_v = \frac{1}{2}\sum_{i=1}^{n} m_i\left[x_i' \cdot \left(\sum_{j=1}^{i} L_j' \dot{q}_j\right)\right]^2. \quad (31)$$

The square of the velocity's magnitude is equal to the dot product of the vector with itself

$$K_v = \frac{1}{2}\sum_{i=1}^{n} m_i\left(\sum_{j=1}^{i}(x_i' \cdot L_j')\dot{q}_j\right) \cdot \left(\sum_{j=1}^{i}(x_i' \cdot L_j')\dot{q}_j\right). \quad (32)$$

Evaluating the sums for j from 1 to i,

$$K_v = \tfrac{1}{2}\sum_{i=1}^{n} m_i \left(x_i' \cdot L_1' \dot{q}_1 + \cdots + x_i' \cdot L_i' \dot{q}_i\right) \cdot \left(x_i' \cdot L_1' \dot{q}_1 + \cdots + x_i' \cdot L_i' \dot{q}_i\right). \quad (33)$$

Evaluating the sum for i from 1 to n, reorganizing the terms and extracting $\dot{q}$ it is possible by symmetry of the terms rewrite (33) in matrix form to get a better compression of this equation

$$K_v = \frac{1}{2}\begin{pmatrix}\dot{q}_1 & \cdots & \dot{q}_n\end{pmatrix}\mathcal{M}_v\begin{pmatrix}\dot{q}_1 \\ \vdots \\ \dot{q}_n\end{pmatrix}. \quad (34)$$

where each element $\mathcal{M}_{v_{ij}}$ of the matrix $\mathcal{M}_v$ is computed doing

$$\mathcal{M}_{v_{ij}} = \sum_{k=Max(i,j)}^{n} m_k(x_k' \cdot L_i') \cdot (x_k' \cdot L_j'). \quad (35)$$

The symmetric matrix $\mathcal{M}_v$ cold be separated in the product of three matrix, two triangular and one diagonal matrix

$$\mathcal{M}_v = V^T m V. \tag{36}$$

Where the elements of the matrix V are vectors and the elements of m are scalars those matrix are given by

$$m = \begin{pmatrix} m_1 & 0 & \cdots & 0 \\ 0 & m_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & m_n \end{pmatrix}, \tag{37}$$

$$V = \begin{pmatrix} x_1' \cdot L_1' & 0 & \cdots & 0 \\ x_2' \cdot L_1' & x_2' \cdot L_2' & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ x_n' \cdot L_1' & x_n' \cdot L_2' & \cdots & x_n' \cdot L_n' \end{pmatrix}. \tag{38}$$

Which means that the contribution of kinetic energy produced due to mass displacements with respects to the reference frame could be easily computed as

$$K_v = \frac{1}{2}\dot{q}^T V^T m V \dot{q}. \tag{39}$$

Now we follow a similar procedure for the component of the kinetic energy K_I

$$K_I = \frac{1}{2}\sum_{i=1}^{n} I_i \left(\sum_{j=1}^{i} \dot{q}_j\right)^2. \tag{40}$$

Evaluating the sums for i and j form 1 to n we get

$$K_I = \frac{1}{2}\left[I_1(\dot{q}_1)^2 + I_2(\dot{q}_1 + \dot{q}_2)^2 + \cdots + I_n(\dot{q}_1 + \cdots + \dot{q}_n)^2\right]. \tag{41}$$

Expanding the expression, extracting $\dot{q}$ and writing in matrix form.

$$K_I = \frac{1}{2}\begin{pmatrix}\dot{q}_1 & \cdots & \dot{q}_n\end{pmatrix} \mathcal{M}_I \begin{pmatrix}\dot{q}_1 \\ \vdots \\ \dot{q}_n\end{pmatrix}. \tag{42}$$

Where

$$\mathcal{M}_I = \begin{pmatrix} \sum_{i=1}^n I_i & \sum_{i=2}^n I_i & \cdots & \sum_{i=n}^n I_i \\ \sum_{i=2}^n I_i & \sum_{i=2}^n I_i & \cdots & \sum_{i=n}^n I_i \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{i=n}^n I_i & \sum_{i=n}^n I_i & \cdots & \sum_{i=n}^n I_i \end{pmatrix}. \tag{43}$$

The matrix $\mathcal{M}_I$ can be written as the product of two matrix δ and I if we define them as

$$M_I = \delta I = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ 0 & 1 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix} \begin{pmatrix} I_1 & 0 & \cdots & 0 \\ I_2 & I_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ I_n & I_n & \cdots & I_n \end{pmatrix}. \tag{44}$$

In such a way the component of kinetic energy due to the movement of links around their mass center is given by

$$K_I = \frac{1}{2}\dot{q}^T \delta I \dot{q}. \tag{45}$$

As conclusion, we have an expression to compute the total kinetic energy of the serial robot using the axes of the robot and the mass centers in conformal geometric algebra

$$K = \frac{1}{2}\dot{q}^T (V^T m V + \delta I)\dot{q}. \tag{46}$$

Note that this expression allows us to compute the kinetic energy without the derivatives.

B. Potential Energy

As opposite to kinetic energy the potential energy does not depends of the velocity, but it depends of the position of each link of the serial robot. Thanks to the equations of direct kinematics 14 we can compute the position x_i' of each link. In order to know the potential energy U_i, we compute the dot product of this points and the force applied to each point.

$$U_i = x_i' \cdot F_i, \tag{47}$$

here the potential energy is due to conservative forces such as the gravity forces, then $F_i = m_i g e_2$. Also the total potential energy of the system is equal to the sum of all U_i.

$$U = \sum_{i=1}^{n} x_i' \cdot F_i. \tag{48}$$

C. Lagrange's Equations

The dynamic equations of a robot could be computed based in the Newton equations, but the formulation becomes complicated when the number of degrees of freedom increases. For this reason we will use the Lagranges equations of movement.

The Lagrangian $\pounds$ is defined as the difference between the kinetic and potential energy of the system.

$$\pounds = K - U. \tag{49}$$

the Lagrange's movement equation is given by

$$\frac{d}{dt}\left[\frac{\partial \pounds}{\partial \dot{q}}\right] - \frac{\partial \pounds}{\partial q} = \tau. \tag{50}$$

Starting with the partial derivative of $\pounds$ respect to $\dot{q}$

$$\frac{\partial \pounds}{\partial \dot{q}} = \frac{\partial K}{\partial \dot{q}} - \frac{\partial U}{\partial \dot{q}} = \frac{\partial K}{\partial \dot{q}}. \tag{51}$$

Note that the partial derivative of U respect to $\dot{q}$ is always zero since U does not depend of the joint's velocity $\dot{q}$. Replacing K given by (46) in (51).

$$\frac{\partial \pounds}{\partial \dot{q}} = \frac{\partial}{\partial \dot{q}}\left(\frac{1}{2}\dot{q}^T (V^T m V + \delta I)\dot{q}\right) = (V^T m V + \delta I)\dot{q}. \tag{52}$$

In order to simplify the notation the matrix $\mathcal{M}$ is defined as $\mathcal{M} = \mathcal{M}_v + \mathcal{M}_I = V^T m V + \delta I$ and the equation 52 is now written as

$$\frac{\partial \pounds}{\partial \dot{q}} = \mathcal{M}\dot{q}. \tag{53}$$

On the other hand the partial derivative of $\pounds$ respect to q is given by

$$\frac{\partial \pounds}{\partial q} = \frac{\partial K}{\partial q} - \frac{\partial U}{\partial q} = \frac{1}{2}\dot{q}^T \left(\frac{\partial \mathcal{M}}{\partial q}\right)\dot{q} - \frac{\partial U}{\partial q}. \quad (54)$$

Replacing 53 and 54 in the Lagrange equation (50) we get.

$$\frac{d}{dt}\left[\mathcal{M}\dot{q}\right] - \left[\frac{1}{2}\dot{q}^T \left(\frac{\partial \mathcal{M}}{\partial q}\right)\dot{q} - \frac{\partial U}{\partial q}\right] = \tau. \quad (55)$$

The time derivative of the equation 55 give us

$$\mathcal{M}\ddot{q} + \dot{\mathcal{M}}\dot{q} - \frac{1}{2}\dot{q}^T \left(\frac{\partial \mathcal{M}}{\partial q}\right)\dot{q} + \frac{\partial U}{\partial q} = \tau. \quad (56)$$

In order to simplify the expression we rename parts of the equation as follows.

$$C = \dot{\mathcal{M}} - \frac{1}{2}\dot{q}^T \left(\frac{\partial \mathcal{M}}{\partial q}\right), \quad (57)$$

$$G = \frac{\partial U}{\partial q}. \quad (58)$$

Where C is the coriolis and centrifugal matrix and G is the vector of gravitational components. Therefore we can write the dynamic equation for a serial robot with n degrees of freedom

$$\mathcal{M}\ddot{q} + C\dot{q} + G = \tau. \quad (59)$$

Now we analyze the G matrix, looking for an equation that allows us to get it without partial derivatives. Using the equation 48 we write

$$G = \frac{\partial U}{\partial q} = \frac{\partial}{\partial q}\left(\sum_{i=1}^{n} F_i \cdot x'_i\right). \quad (60)$$

Since the forces $F_i = m_i g e_2$ are produced by the gravity they does not depend of the joints positions q.

$$G = \sum_{i=1}^{n} F_i \cdot \left(\frac{\partial}{\partial q} x'_i\right). \quad (61)$$

Recalling the equation of differential kinematics 16 we know

$$\frac{\partial}{\partial q} x'_i = \begin{pmatrix} x'_i \cdot L'_1 \\ x'_i \cdot L'_2 \\ \vdots \\ x'_i \cdot L'_i \end{pmatrix}. \quad (62)$$

Expanding the sum (61) from $i = 1$ to n and replacing $\frac{\partial}{\partial q} x'_i$ by (62)

$$G = \begin{pmatrix} x'_1 \cdot L'_1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} F_1 + \begin{pmatrix} x'_2 \cdot L'_1 \\ x'_2 \cdot L'_2 \\ \vdots \\ 0 \end{pmatrix} F_2 + \cdots + \begin{pmatrix} x'_n \cdot L'_1 \\ x'_n \cdot L'_2 \\ \vdots \\ x'_n \cdot L'_n \end{pmatrix} F_n. \quad (63)$$

The equation 63 written as matrix.

$$G = \begin{pmatrix} x'_1 \cdot L'_1 & x'_2 \cdot L'_1 & \cdots & x'_n \cdot L'_1 \\ 0 & x'_2 \cdot L'_2 & \cdots & x'_n \cdot L'_2 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & x'_n \cdot L'_n \end{pmatrix} \begin{pmatrix} F_1 \\ F_2 \\ \vdots \\ F_n \end{pmatrix}. \quad (64)$$

As you can see this matrix is basically the transposed of the matrix V given in equation 38. Calling F to the vector with components F_i, we can finally write the equation.

$$G = V^T F. \quad (65)$$

Furthermore F is given by the product of two matrices

$$F = \begin{pmatrix} m_1 & 0 & \cdots & 0 \\ 0 & m_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & m_n \end{pmatrix} \begin{pmatrix} ge_2 \\ ge_2 \\ \vdots \\ ge_2 \end{pmatrix} = ma. \quad (66)$$

Where a is a vector of accelerations. The equation 66 allows us to separate the constant matrices and the variables of the serial robot

$$G = V^T ma. \quad (67)$$

Finally we have a short and useful equation to compute the vector G using the information of the joints axes. Now we analyze the Coriolis matrix C. In fact there are many ways to compute this matrix and there are many matrices C that satisfies the dynamic equation 59. Although we already have an equation to compute the matrix C (57), we will look for a more simple equation to avoid the necessity of derivatives. Based in the properties of the matrices $\mathcal{M}$ and C (see [1]), it is known that.

$$\dot{\mathcal{M}} = C + C^T. \quad (68)$$

Recalling that $\mathcal{M} = V^T mV + \delta I$ we will determine its derivative.

$$\dot{\mathcal{M}} = \frac{d}{dt}\mathcal{M} = \frac{d}{dt}\left(V^T mV + \delta I\right), \quad (69)$$

$$\dot{\mathcal{M}} = \frac{d}{dt} V^T mV, \quad (70)$$

$$\dot{\mathcal{M}} = V^T m\dot{V} + \dot{V}^T mV, \quad (71)$$

$$\dot{\mathcal{M}} = V^T m\dot{V} + (V^T m\dot{V})^T. \quad (72)$$

Taking into account the equations 68 and 72 we have a short and clear equation to compute the matrix C without derivatives.

$$C = V^T m\dot{V}. \quad (73)$$

The last sentence is true since we can compute the matrix $\dot{V}$ without derivatives just in function of the joint's values q, $\dot{q}$ and the axes of the robot. In order to compute the element $\dot{V}_{ij}$ the time derivative of $x'_i \cdot L'_j$ is needed, using the equation 27.

$$\dot{V}_{ij} = \frac{d}{dt}(x'_i L'_j) = \dot{x}'_i \cdot L'_j + x'_i \cdot \dot{L}'_j,$$

$$\dot{V}_{ij} = \sum_{k=1}^{i}(x'_i \cdot L'_k) \cdot L'_j \dot{q}_k + \frac{1}{2}\sum_{k=1}^{j-1} x'_i \cdot (L'_j L'_k - L'_k L'_j)\dot{q}_k. \quad (74)$$

Note that $V_{ij} = 0$ whenever $j > i$ reason why $\dot{V}_{ij} = 0$. Perhaps these equations to get $\dot{V}$ could be seen confused. We will rewrite these equations as matrix to give a more clear explanation of the method to compute $\dot{V}$. It is possible to write the matrix V as the product of two matrices.

$$V = \begin{pmatrix} x'_1 & 0 & \cdots & 0 \\ 0 & x'_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & x'_n \end{pmatrix} \begin{pmatrix} L'_1 & 0 & \cdots & 0 \\ L'_1 & L'_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ L'_1 & L'_2 & \cdots & L'_n \end{pmatrix} = XL, \quad (75)$$

then $\dot{V} = \dot{X}L + X\dot{L}$ with

$$\dot{X} = \begin{pmatrix} \dot{x}'_1 & 0 & \cdots & 0 \\ 0 & \dot{x}'_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \dot{x}'_n \end{pmatrix}, \quad \dot{L}' = \begin{pmatrix} \dot{L}'_1 & 0 & \cdots & 0 \\ \dot{L}'_1 & \dot{L}'_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \dot{L}'_1 & \dot{L}'_2 & \cdots & \dot{L}'_n \end{pmatrix} \quad (76)$$

Compute $\dot{x}'_i$ is simple using (16).

$$\begin{pmatrix} \dot{x}'_1 \\ \vdots \\ \dot{x}'_n \end{pmatrix} = XL\dot{q} = V\dot{q}. \quad (77)$$

To compute $\dot{L}'_i$ that represents the velocity of the axis i produced by the rotation around the previous axes we can do

$$\begin{pmatrix} \dot{L}'_1 \\ \dot{L}'_2 \\ \vdots \\ \dot{L}'_n \end{pmatrix} = \frac{1}{2}\left[\begin{pmatrix} L'_1L'_1 & 0 & \cdots & 0 \\ L'_2L'_1 & L'_2L'_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ L'_nL'_1 & L'_nL'_2 & \cdots & L'_nL'_n \end{pmatrix} - \begin{pmatrix} L'_1L'_1 & 0 & \cdots & 0 \\ L'_1L'_2 & L'_2L'_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ L'_1L'_n & L'_2L'_n & \cdots & L'_nL'_n \end{pmatrix}\right]\dot{q}. \quad (78)$$

Using the definition of $\bar{\times}$ the matrix $\dot{L}'$ could be wrote as

$$\begin{pmatrix} \dot{L}'_1 \\ \dot{L}'_2 \\ \vdots \\ \dot{L}'_n \end{pmatrix} = \begin{pmatrix} L'_1\bar{\times}L'_1 & 0 & \cdots & 0 \\ L'_2\bar{\times}L'_1 & L'_2\bar{\times}L'_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ L'_n\bar{\times}L'_1 & L'_n\bar{\times}L'_2 & \cdots & L'_n\bar{\times}L'_n \end{pmatrix}\dot{q}. \quad (79)$$

In conclusion using the equation 52 and 73, we have rewritten the dynamic equation of serial robot with n degrees of freedom.

$$\delta I\ddot{q} + V^T m(V\ddot{q} + \dot{V}\dot{q} + a) = \tau. \quad (80)$$

This decomposition allows us to see the components of inertia momentum, centrifuge forces and the gravity forces and finally 80 is the dynamic equation of a n degrees of freedom serial robot where the elements of the matrices are multivectors of the geometric algebra $G_{4,1,0}$.

Summarizing δ,m, I and a are constant and known matrices. Only V and $\dot{V}$ changes trough time and they are computed in a parallel way, using a thread for each matrixs component as we show in the Fig 4, in this way and having n^2 threads it is possible to get the V in $O(Log_2(n))$

Similarly $\dot{V}$ is computed using a thread for each matrix component having this in $O(n)$, this is performed as we show in the Fig.5

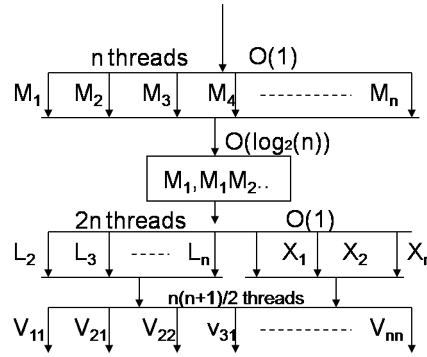

Fig. 4. Computing V matrix

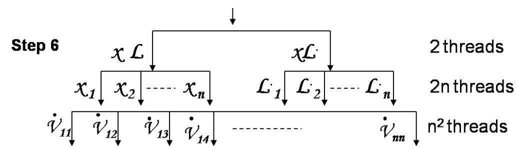

Fig. 5. Computing $\dot{V}$ matrix

VII. Conclusions

The authors have shown how the equation of dynamics can be rewritten using the conformal geometric to get a new expression in terms of the robot axis. This new equation not only is easier to implement but also it can be easily parallelized to get advantages of the new parallel architectures. This new expression could be also used for systems identification since it has only three matrices: one for the mass, one for the inertia values and one more with the geometry of the robot. Finally using our approach with parallel computing, it is possible to have a complexity $O(n)$ using n^2 threads which makes our algorithm the fastest to compute the dynamics of a serial robot having the advantage of modifying the configuration of the robot, that is, the geometry or topology on the fly without need to recompile anything.

References

[1] Kelly Rafael, Santibaez Victor, and Loria Julio, *Control of Robot Manipulators in Joint Space*. Springer, 2005.

[2] H. Li, D. Hestenes, A. Rockwood. "Generalized Homogeneous coordinates for computational geometry". G. Somer, editor, *Geometric Computing with Clifford Algebras*, pages 27-52. Springer-Verlag Heidelberg. 2001.

[3] Bayro-Corrochano E. and Kähler D. Motor Algebra Approach for Computing the kinematics of robot manipulators. *Journal of Robotics Systems*, 17(9):495-516, 2000.

[4] Julio Zamora-Esquivel Bayro-Corrochano E.*Kinematics and Differential Kinematics of Binocular Robot Heads*,Proceedings of ICRA 2006.

[5] Bayro-Corrochano E. *Geometric Computing for Perception Action Systems*. Springer Verlag, Boston, 2001.

[6] Kim, Jung-Ha.; Kumar, Vijay R.: Kinematics of Robot Manipulators Via Line Transformations, *Journal of Robotic Systems*, Vol. 7, No 4, pp. 647-674, 1990.

[7] Scott McMillan and David E. Orin, *Forward Dynamics of Multilegged Vehicles Using the Composite Rigid Body Method* International conference on robotics and Automation Leuven, Belgium - May 1998

[8] Uri M. Ascher, Dinesh K. Pai and Benoit P. Cloutier, *Forward Dynamics, Elimination Methods, and Formulation Stiffness in Robot Simulation*, May 27, 1996.