Some practical uses of the Lie group SE(3) in computers visualization

Jean D. Reuss
Association Euratom-C.E.A sur la Fusion.
CEA/DSM/DRFC, CEA-Cadarache,
F-13108, Saint-Paul-lez-Durance, France.
E-mail: reuss@drfc.cad.cea.fr

April, 30, 2002

Abstract

In the tremendous world of computer graphics, the evolution of scientific visualization emphasizes the role of on-line simulation for which a fundamental tool is the common concept of displacement in space. The Lie group theory supplies not only an abstract framework for the displacements properties, but also a useful formalism. Consequently, we review here the basic practical algorithms involved in perspective drawing and in simulating the dynamics of rigid bodies systems.

1 About a didactic role of numerical algorithms

Graphical representation in an interactive simulation is a problem often encountered in scientific work. Such on-line visualization is based essentially on a conceptual loop without end, which is formed, in short, by three parts:

1) Feedback from the user, who generally handles (only) a mouse with two or three buttons and a keyboard.

2) Calculus of the new state of the model which is supposed to evolve according to some mathematical law, typically a differential system.

3) Drawing on the computer screen of some conventional items which can represent the model seen by the designer. In fact, on this point, a large place is naturally devoted to the old, but irreplaceable linear perspective of geometric shapes.

Nowadays this kind of visual applications is more or less completely realizable by using numerous well-known softwares, which are called for example: Matlab, Scilab, Mathematica, OpenGL, Java3D, but they are many others. Each of these tools has its own practical characteristics, more or less 'open', for example, and its own deficiencies or strong advantages.

A Lie group is a continuous group of transformation, which is also a differentiable manifold. The space tangent at the identity element of a Lie group is called the Lie algebra for that group. This algebra is a vector space of finite or infinite dimension, which has a bracket [1] which behaves as the ordinary cross product. The reader knows that the previous sentences are far from being rigorous and mathematically complete. Really the genuine Lie group theory is fairly abstract. However, they are at least two Lie groups which are very close to the concrete experience of daily life, with its activities in our prosaic Euclidean space with three dimensions. These are $SO(3)$, the special Orthogonal group (the group of rotations) and $SE(3)$, the special Euclidean group (the group of rigid body displacements). In other words, these two groups play, or can play, a natural role in the algorithms of geometry and rigid body mechanics used in visualization.

Well, but why to choose employing effectively the (or a) formalism of the Lie group ? Disregarding the pedantic ones, they are three reasons. We take advantage of a safe and well elaborated frame to verify the correctness of several new numerical methods which sometimes could appear fairly strange. We obtain efficient and coherent notations to act on often slightly tedious formulas. The last reason is a didactic one. The progress of rational mechanics has been intimately bound to the advancement of mathematics and in the purpose of learning abstract notions, it is good to become accustomed with some more concrete examples.

Today, a natural and often possible way to make concrete mathematical notions, consists to write or read some numerical algorithms which would be, delivered to and used by a computer graphics system.

Thus the reader is advised that the following part of this text is only a kind of crib sheet which collects a series of hopefully useful and important mathematical relations, with a certain attempt to unified notations, in the areas of geometric drawing and dynamics of articulated rigid bodies systems.

2 Fundamentals of notations for the displacement group

In the ordinary Euclidean space $E\left(3\right)$, we consider two orthonormal frames of reference: $\Re_1 \triangleq O_1,\ \overrightarrow{e_{1\alpha}}$ and $\Re_2 \triangleq O_2,\ \overrightarrow{e_{2\alpha}}$ (with $\alpha = 1,2,3$). Moreover, at each of these frames will be attached later one rigid body, which will be called C_1 and C_2.

The situation of $\Re_2$ relatively to $\Re_1$ is defined by Θ : a 3×3 matrix, and b; a 3×1 vector with:

$$\Theta_{\alpha\beta} = \overrightarrow{e_{1\alpha}} \bullet \overrightarrow{e_{2\beta}} \quad (1)$$

$$b_{_\alpha} = \overrightarrow{e_{1\alpha}} \bullet \overrightarrow{O_1O_2} \quad (2)$$

(i.e. the column vectors of the matrix Θ, which is characterized by **3 parameters**, are formed by the unitary vectors of $\Re_2$ expressed in the frame of $\Re_1$, (shortly said: measured or counted in $\Re_1$) and b is the coordinates of $\overrightarrow{O_1O_2}$, counted in $\Re_1$.

Note that indeed, Θ and b define both either a **transformation** with **6 parameters** or the **change of coordinates** as a result of a change of frame of reference.

With the Lie group formalism the preceding text become approximately:

$\Theta \in SO(3)$, consequently $\Theta^{-1} = \Theta^T$ with $\det(\Theta) = +1$ (however $\det(\Theta) = -1$ is indeed useful as describing symmetric structures).

$f \triangleq f_{1,2} \triangleq f_2 \in SE(3)$, which can acting on RP^3 is written as a 4×4 matrix:

$$f = \begin{vmatrix} \Theta & b \\ 0 & 1 \end{vmatrix} \quad (3)$$

and we have:

$$f^{-1} = \begin{vmatrix} \Theta^T & -\Theta^T \cdot b \\ 0 & 1 \end{vmatrix} \tag{4}$$

The Lie algebra $se(3)$ associated with $SE(3)$ can be presented by referring to the exponential mapping $SE(3) \longrightarrow SE(3)$, which is defined by two vectors $\in R^3$: $\overrightarrow{a}$ which is the vector of finite rotation, and $\overrightarrow{\ell}$. We need first some additional notations.

The length of $\overrightarrow{a}$ is $\vartheta = \|\overrightarrow{a}\| = \sqrt{a_1^2 + a_2^2 + a_3^2}$

Its unit colinear vector of $\overrightarrow{a}$ is $\overrightarrow{u} = \frac{\overrightarrow{a}}{\vartheta}$, so $\|\overrightarrow{u}\| = 1$ and $\overrightarrow{a} = \overrightarrow{u} \cdot \vartheta$

The skew-symmetry operator ("hat map") acting on vectors such as $\overrightarrow{a}$ and $\overrightarrow{u}$ gives 3×3 matrices which are written as $\widehat{a}$ and $\widehat{u}$ with:

$$\widehat{u} = \begin{vmatrix} 0 & -u_3 & u_2 \\ u_3 & 0 & -u_1 \\ -u_2 & u_1 & 0 \end{vmatrix} \text{ ; and } \qquad \widehat{a} = \widehat{u} \cdot \vartheta$$

The following obvious properties are sometimes useful:

$\widehat{u} + \widehat{u}^T = 0$; $\widehat{a} + \widehat{a}^T = 0$; $\widehat{a}^3 = -\vartheta^2 \cdot \widehat{a}$; $\widehat{u}^3 = -\widehat{u}$

A matrix product such as $\widehat{a} \cdot b$ is corresponding to the cross product $\overrightarrow{a} \wedge \overrightarrow{b}$ and $trace\left(\widehat{a} \cdot \widehat{b}\right) = -2 \cdot \overrightarrow{a} \bullet \overrightarrow{b}$. Denoting the unity matrix by $\top \triangleq \top_{3\times3} = diag(1,1,1)$ we have also:

$$u \cdot u^T = \top + \widehat{u}^2 \tag{5}$$

With these notations the two fundamental relations between the log and exp operators take the form:

$$f = \exp\left(\begin{vmatrix} \widehat{u} \cdot \vartheta & \ell \\ 0 & 0 \end{vmatrix}\right) = \begin{vmatrix} \Theta & b \\ 0 & 1 \end{vmatrix} \tag{6}$$

$$\log(f) = \log\left(\begin{vmatrix} \Theta & b \\ 0 & 1 \end{vmatrix}\right) = \begin{vmatrix} \widehat{u} \cdot \vartheta & \ell \\ 0 & 0 \end{vmatrix} \tag{7}$$

with:

$$\Theta = \exp\left(\widehat{u} \cdot \vartheta\right) = \top + \sin\vartheta \cdot \widehat{u} + (1 - \cos\vartheta) \cdot \widehat{u}^2 \tag{8}$$

$$\widehat{u} \cdot \vartheta = \log(\Theta) = \frac{\vartheta}{2 \cdot \sin\vartheta}\left(\Theta - \Theta^T\right) \tag{9}$$

$$b = \left(\top + \frac{(1 - \cos\vartheta)}{\vartheta} \cdot \widehat{u} + \left(1 - \frac{\sin\vartheta}{\vartheta}\right) \cdot \widehat{u}^2\right) \cdot \ell \tag{10}$$

$$\ell = \left(\top - \frac{\vartheta}{2} \cdot \widehat{u} + \left(1 - \frac{\vartheta \cdot (1 + \cos\vartheta)}{2 \cdot \sin\vartheta}\right) \cdot \widehat{u}^2\right) \cdot b \tag{11}$$

(The equation (8) is the well-known $Rodrigues' \; formula$ and we have also; $trace\,(\Theta) = 1 + 2 \cdot \cos\vartheta$).

We use now a kinematic viewpoint by considering that f is time-dependent; thus $f \triangleq f(t)$ and $\overset{\bullet}{f} \equiv \frac{df}{dt}$ is corresponding to a tangent vector.

The angular velocity vector $\overrightarrow{\omega} = \overrightarrow{u} \cdot \varpi$, and $\overrightarrow{v}$ the linear velocity vector (of $\Re_2$ relatively to $\Re_1$), with $\varpi = \sqrt{\omega_1^2 + \omega_2^2 + \omega_3^2}$; $\overrightarrow{u} = \frac{\overrightarrow{\omega}}{\varpi}$; $\overrightarrow{\omega} = \overrightarrow{u} \cdot \varpi$ and $\widehat{\omega} = \widehat{u} \cdot \varpi$ are then introduced.

By counting the coordinates ω and v in $\Re_2$, it becomes possible to represent the generic element g of $se(3)$ as a 4×4 matrix in this (not unique) way:

$$g = f^{-1} \cdot \overset{\bullet}{f} = \begin{vmatrix} \widehat{\omega} & v \\ 0 & 0 \end{vmatrix} = \begin{vmatrix} \widehat{u} \cdot \varpi & v \\ 0 & 0 \end{vmatrix} = \begin{vmatrix} \Theta^T \cdot \overset{\bullet}{\Theta} & \Theta^T \cdot \overset{\bullet}{b} \\ 0 & 0 \end{vmatrix} \tag{12}$$

i.e. we have:

$$\widehat{\omega} = \Theta^T \cdot \overset{\bullet}{\Theta} \tag{13}$$

$$v = \Theta^T \cdot \overset{\bullet}{b} \equiv \Theta^T \cdot \frac{db}{dt} \tag{14}$$

$$\varpi = \overset{\bullet}{\vartheta} \equiv \frac{d\vartheta}{dt} \tag{15}$$

3 Drawing

Pictures ultimately consist of points. In computers graphics the most important operation is surely the transformation of the point of some abstract "user world" to two or three discrete coordinates which will be used by a graphic device built in hardware.

For reason of simplicity we restrict here our description to the very common situation where the "user world" is the ordinary space, the transformation is the linear perspective and the graphic device is a screen with a limited number of (colored) pixels (typically today: 1280 x 1024).

Naturally, in this case, numerous points are gathered in straight-line segments, triangles, and even often more complicated structures, but, maybe, this fact is conceptually of little importance.

3.1 Transformations in $\mathbf{RP}^3$

A point X in the "user space" is then described by its homogeneous coordinate in the corresponding orthonormal frame $\Re$ as:

$$X = \begin{vmatrix} x \\ y \\ z \\ s \end{vmatrix}$$

The equation $s = 0$ represent the *plane at infinity* and the user space can be considered as being the projective space RP^3.

Several transformation operators T are introduced, which are written under the form of 4×4 matrices .

Given an ordered sequence of operators $T_1, T_2, T_3 \cdots T_n$ the composition rule is:

$X_0 = T_1 \cdot X_1, X_1 = T_2 \cdot X_2, \cdots X_{n-1} = T_n \cdot X_n \Longrightarrow X_0 = T_1 \cdot \cdots T_n \cdot X_n$

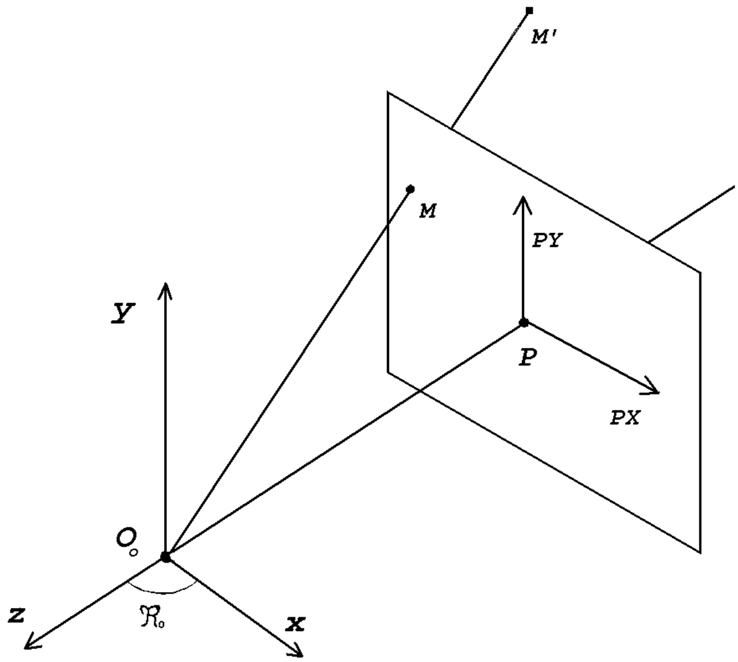

Figure 1: The architect's perspective with the eye situated at the origin of $\Re_0$

Note that this rule of "right multiplication" can be expressed in an corresponding recursive form.

Putting $U = \top_{4\times4} = diag\,(1,1,1,1)$

For $i = 1$ *to n do* $U = U \cdot T_i$; $\quad X_0 = U \cdot X_i$ end loop.

These matrix-operators are classified in three different types:

1) Authentic element of $SE\,(3)$; then $T = f$: i.e. $SE(3) \times RP^3 \longrightarrow RP^3$, according to the formulas (1) and (8).

2) Several geometric transformations out of $SE\,(3)$ such as reflection $\in O(3)$, shearing, scaling, warping $\cdots$

3) A particular and efficient first transformation T_1 is generally employed:

the *frustum transformation* . It operate the space clipping which is needed to eliminate what is outside the truncated viewing pyramid.

3.2 The architect's perspective

1) The point of view, or eye, is put at the origin O_0 of $\Re_0$ and in this frame the axis is choose as being perpendicular to the perspective screen according to the perspective terminology.

2) Some intermediate perspective coordinates PX, PY, PT in the plane of the perspective screen are obtained according to the so called perspective division:

$$PX = \frac{x}{z}\ ;\ PY = \frac{y}{z}\ ;\ PT = \frac{s}{z} \tag{16}$$

3) Afterwards a trivial similitude and a discretization are made to fit PX and PY to the position of the corresponding pixel.

3.3 The painter's perspective

In the above formalism, despite the fact that the information "on infinity", given by s, is perfectly available, it is neither elegantly nor effectively treated and used by the standard architect's perspective. One way to overcome this deficiency is to turn up oneself to the painter's perspective. Then the drawing in linear projection is obtained by appropriate geometric constructions, which employ principally the elements at infinity (with $s = 0$), in the plane of the screen.

4 Dynamics of the single rigid body

4.1 Spatial velocity and wrench

Later on the elements of $se\,(3)$ will be called spatial velocities (or velocity twists), and will be written as 6×1 vectors, $V = \begin{vmatrix} \omega \\ v \end{vmatrix}$. The elements of the dual algebra $se\,(3)^\star$ are called torque-force wrenches and are written as a 6×1 vectors, $F = \begin{vmatrix} N \\ \digamma \end{vmatrix}$.

Note that $\digamma^T \cdot V$ is a power.

4.2 Kinematics

Spatial velocities and wrenches, which are called " *torseurs* " in french have their own specific well-known properties. With the use of the adjoint representation and of the operators "Ad" : $SE(3) \times se(3) \longrightarrow se(3)$ and "$Ad^\star$" : $SE(3) \times se(3)^\star \longrightarrow se(3)^\star$, the changes of coordinates of spatial velocities and wrenches is governed by the following rules, (in which it can be noted that $Ad^\star \equiv Ad^T$):

$$V_1 = Ad_{f_{1,2}}(V_2) = \begin{vmatrix} \omega_1 \\ v_1 \end{vmatrix} = \begin{vmatrix} \Theta & 0 \\ \widehat{b} \cdot \Theta & \Theta \end{vmatrix} \begin{vmatrix} \omega_2 \\ v_2 \end{vmatrix} \tag{17}$$

$$V_2 = Ad_{f_{1,2}^{-1}}(V_1) = \begin{vmatrix} \omega_2 \\ v_2 \end{vmatrix} = \begin{vmatrix} \Theta^T & 0 \\ \Theta^T \cdot \widehat{b}^T & \Theta^T \end{vmatrix} \begin{vmatrix} \omega_1 \\ v_1 \end{vmatrix} \tag{18}$$

$$F_1 = Ad^\star_{f_{1,2}^{-1}}(F_2) = \begin{vmatrix} N_1 \\ F_1 \end{vmatrix} = \begin{vmatrix} \Theta & \widehat{b} \cdot \Theta \\ 0 & \Theta \end{vmatrix} \begin{vmatrix} N_2 \\ \digamma_2 \end{vmatrix} \tag{19}$$

$$F_2 = Ad^\star_{f_{1,2}}(F_1) = \begin{vmatrix} N_2 \\ \digamma_2 \end{vmatrix} = \begin{vmatrix} \Theta^T & \Theta^T \cdot \widehat{b}^T \\ 0 & \Theta^T \end{vmatrix} \begin{vmatrix} N_1 \\ F_1 \end{vmatrix} \tag{20}$$

Moreover, the operator "ad": $se(3) \times se(3) \longrightarrow se(3)$, which corresponds to the Lie bracket, and thus to the cross-product, is needed for the calculus of the accelerations. We have, with $V = \begin{vmatrix} \omega \\ v \end{vmatrix} \in se(3)$, $\tilde{\tilde{V}} = \begin{vmatrix} \tilde{\tilde{\omega}} \\ \tilde{\tilde{v}} \end{vmatrix} \in se(3)$, and $F = \begin{vmatrix} N \\ \digamma \end{vmatrix} \in se\,(3)^\star$:

$$ad_V\left(\tilde{\tilde{V}}\right) = \begin{vmatrix} \widehat{\omega} & 0 \\ \widehat{v} & \widehat{\omega} \end{vmatrix} \cdot \begin{vmatrix} \tilde{\tilde{\omega}} \\ \tilde{\tilde{v}} \end{vmatrix} \tag{21}$$

The dual operator of "ad" is "$ad^{\bigstar}$": $se(3)\times se(3)^{\star}\longrightarrow se(3)^{\star}$ and is defined by:

$$ad_V^{\bigstar}(F)=\begin{vmatrix}\widehat{\omega}^T & \widehat{v}^T\\ 0 & \widehat{\omega}^T\end{vmatrix}\cdot\begin{vmatrix}N\\ F\end{vmatrix}=-\begin{vmatrix}\widehat{\omega} & \widehat{v}\\ 0 & \widehat{\omega}\end{vmatrix}\cdot\begin{vmatrix}N\\ F\end{vmatrix}\tag{22}$$

4.3 Inertial frame

In a gravitational field of intensity $\overrightarrow{g}$, instead of considering that the absolute reference frame $\Re_1$ is a Galilean frame, it is suitable (by referring to General Relativity) and frequent to define $\Re_1$ as being in an uniform accelerated motion with the acceleration $-\overrightarrow{g}$.

4.4 Inertia characteristics

For a rigid body, we need 10 parameters, the mass m, the position of the center of mass G which is defined by the vector $\overrightarrow{r}=\overrightarrow{OG}$, and I_G the inertia tensor about G, which is represented by a 3×3 symmetric matrix. These parameters are gathered in a 6×6 matrix J called the spatial inertia:

$$J=\begin{vmatrix}I_G-m\cdot\widehat{r}^2 & m\cdot\widehat{r}\\ -m\cdot\widehat{r} & m\cdot\top\end{vmatrix}\tag{23}$$

4.5 Newton Law for the rigid body

Considering that a (moving) rigid body C_2 is attached to $\Re_2$, the 6×1 vector F represents the torque and the force at any origin O, counted in $\Re_2$. Similarly, the spatial velocity V, counted in $\Re_2$, describes the motion of C_2 relatively to the inertial frame $\Re_1$. Then the Newton-Euler equation takes the form:

$$F=J\cdot\overset{\bullet}{V}-ad_V^{\bigstar}(J\cdot V)\tag{24}$$

5 The free, open and unbranched chain of bodies

This generic system is a key for the simulation of elaborated behaviors such as those of biological beings or moderns mechanisms, despite the noticeable approximations involved by the use of rational mechanics (cf. rigidity).

It consist of $n+1$ rigid bodies labeled from 0 *to* n, whose inertia properties are characterized by $J_0, J_1, \cdots J_n$. We define the two integers: .

$$\sigma=6+n\quad and\quad \xi=6+6\cdot n\tag{25}$$

The body label 0 (body C_0 with frame $\Re_0$ appended) is **free** relatively to an inertial frame $\Re_{-1}$ (therefore generalizing slightly [3] and [4] in which $\Re_{-1}=\Re_0$). Its position, according to formulas (1), (2), (3), is of course, defined by 6 parameters that we call the six first configuration variables $q_{-5}, q_{-4}, q_{-3}, q_{-2}, q_{-1}, q_0$.

(For reason of coherence in notations, the body C_0 often gets the range -5 to 0 for its indices).

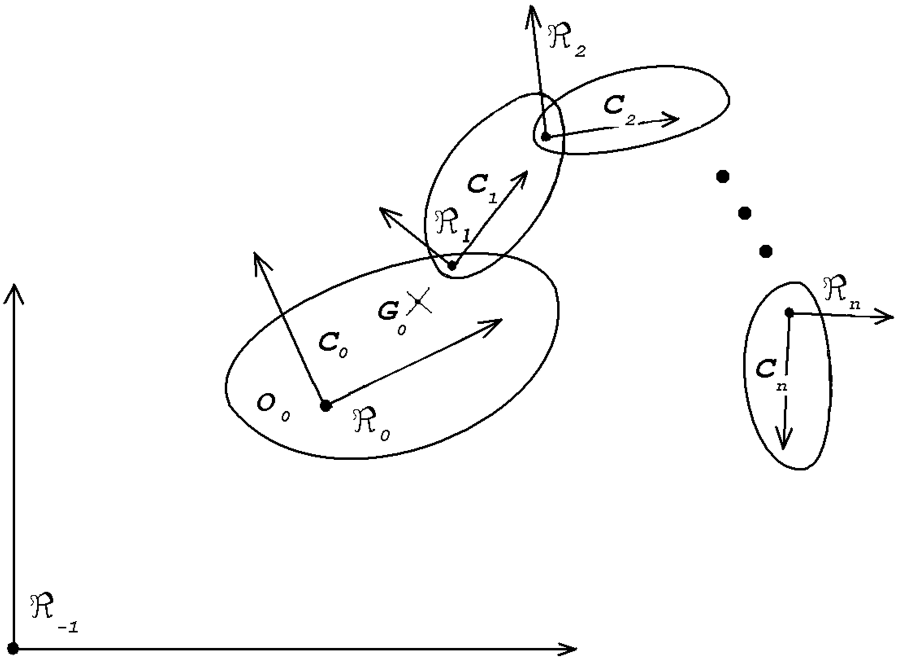

Figure 2: A free and open chain of $n+1$ bodies

Each of the bodies label $j = 1 \cdots n$, is attached to the preceding link, by a mechanical joint which has one degree of freedom, i.e. that is thus either a revolute or a prismatic or (seldom) a twist joint. The state of each of these articulations is described by one parameter θ_j and we define n additional configuration variables $q_j = \theta_j$, so $q \in R^{\sigma \times 1}$. Moreover each of these articulation is equipped with a device, precisely an actuator or a brake, which generates either a known torque or a known force, τ_j, along its axis. (Here $\tau_j = \tau_j\left(q, w, t\right),\ j = 1 \cdots n$; denote known scalar functions. Moreover $\tau \in R^{\sigma \times 1}$ *and* $\tau_0 \in R^{6 \times 1}$ *with* $\tau_0 = 0$).

For completeness, remarking again that each articulation is characterized by its type (revolute or prismatic) and by a single axis (usually the Z-axis), only $3 \cdot n$ additional constitutive parameters are needed to describe the geometrical structure of the entire chain.

(Generally the Denavit-Hartenberg or preferably the Khalil-Kleinfinger (denoted KK) constant parameters are used).

In the continuation, we employ also $n+1$ constant matrices (with j or $k = 1 \cdots n$):

$$S_j = \begin{vmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{vmatrix} ; S_k = \begin{vmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{vmatrix} ; S_0 = \top_{6 \times 6} = \begin{vmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{vmatrix} \tag{26}$$

j corresponds to a revolute joint around the Z axis, and k to a prismatic one.

According to the power of exponential concept, we have also: $f_{i,j} \triangleq f_j = KK_i \cdot$

$\exp(S_j \cdot q_j)$ and let's remind that $f_{-1,0} \triangleq f_0$ is depending of $q_{-5}, q_{-4}, q_{-3}, q_{-2}, q_{-1}, q_0$.

In view of preparing the formulation of the motion equation, we define now the velocities $w \in R^{\sigma\times 1}$ precisely by:

$$\begin{vmatrix} \dot{q}_{-5} \\ \vdots \\ \dot{q}_0 \end{vmatrix} = \widetilde{P}(q) \cdot \begin{vmatrix} w_{-5} \\ \vdots \\ w_0 \end{vmatrix} = \widetilde{P}(q) \cdot \begin{vmatrix} \omega_0 \\ v_0 \end{vmatrix}; \ \widetilde{P}(q) \in R^{6\times 6} \quad and \ \dot{q}_j = w_j = \dot{\theta}_j$$

Thus the equation of motion will be not exactly the (matrix) Lagrange equation, but a little more general equation, sometimes called the Boltzmann-Hamel form. Thus we put also: $P = \begin{vmatrix} \widetilde{P} & 0_{6\times n} \\ 0_{n\times 6} & \top_{n\times n} \end{vmatrix} \in R^{\sigma\times\sigma}$.

The model is enriched by r (conditional) constraints like $\breve{G}(q) = 0,\ \breve{G} \in R^{r\times 1}$, so with two successive time derivations we have:

$$\Psi(q) \cdot w = 0; \quad \Psi \in R^{r\times\sigma} \tag{27}$$

$$\Psi(q) \cdot \dot{w} + \dot{\Psi}(q) \cdot w = 0 \tag{28}$$

These r constraints provoke reaction forces which appear in the equation of motion as r Lagrange multipliers, $\lambda \in R^{r\times 1}$. Note that in typical problems implying support and mechanical contacts, the λ must be explicitly known because they must follow the Coulomb friction's laws.

Finally, the equation of motion take the general compact form (which is a DAE=differential algebraic equation):

$$\dot{q} = P(q) \cdot w \tag{29}$$

$$D(q) \cdot \dot{w} = \tau + H(q, w, \breve{g}) + \Psi^T \cdot \lambda \tag{30}$$

$$\Psi(q) \cdot w = 0 \tag{31}$$

with $D(q) \in R^{\sigma\times\sigma}$; $H(q, w, \breve{g}) \in R^{\sigma\times 1}$; $\Psi =\in R^{r\times\sigma}$.

$D(q)$, sometimes called the mass matrix, is positive definite and symmetric, $D = D^T$.

$H(q, w, \breve{g})$ represents the Coriolis and gravitational effects and has the essential property:

$w = 0 \quad and \quad \breve{g} = 0 \qquad \Longrightarrow \qquad H(q, w, \breve{g}) = 0.$

By eliminating $\dot{w}$ between (28) and (30) it is possible to isolate the r Lagrange (rather Kuhn-Tucker) multipliers λ with the relation:

$$\left(\Psi \cdot D^{-1} \cdot \Psi^T\right) \cdot \lambda = -\ \Psi \cdot D^{-1} \cdot (\tau + H) -\ \dot{\Psi} \cdot w \tag{32}$$

For determining the coefficients of equation (30) the traditional and symbolic methods lead to lengthy calculations.

Recent (1980 -2002) procedures [2],[3],[4], based on kinematics and on the Newton-Euler equation (plus some clever considerations) allows us now to calculate numerically these coefficients more efficiently, because they have recursive (i.e. in french, iterative) forms.

6 Methods and algorithms

In [3], S.R. Ploen , "using techniques and notation from the theory of Lie group" develop a formalism for the dynamics of multibody systems. With the help of [4], I review here a part of his results, written for the case, (yet not tested by the author), of open and unbranched chain where the body C_0 is free relatively to $\Re_{-1}$.

It may be pointed out that a symbol such as V_j describes the spatial velocity of the body C_j, counted in $\Re_j$.

6.1 The Newton-Euler couple of recursions

Resolve the backward dynamic problem (i.e. given $q(t)$ calculate τ).

1) Outward recursion for velocities and accelerations.

Given $V_0 = \begin{vmatrix} \omega_0 \\ v_0 \end{vmatrix} = S_0 \cdot w_0; \quad \overset{\bullet}{V}_0 = \begin{vmatrix} \overset{\bullet}{\omega}_0 \\ \overset{\bullet}{v}_0 - \Theta_0^{-1} \cdot \breve{g} \end{vmatrix};$

w_j and $\overset{\bullet}{w}_j$ *with* $j = 1 \cdots n.$

For $j = 1$ *to* n *do with* $i = j - 1$

$$V_j = Ad_{f_{i,j}^{-1}} (V_i) + S_j \cdot w_j \tag{33}$$

$$\gamma_j = -ad_{S_j \cdot w_j} (V_j) \tag{34}$$

$$\overset{\bullet}{V}_j = Ad_{f_{i,j}^{-1}} \left(\overset{\bullet}{V}_i \right) + S_j \cdot \overset{\bullet}{w}_j + \gamma_j \tag{35}$$

2) Inward recursion for wrenches.

Given $F_{n+1} = \begin{vmatrix} N_{n+1} \\ \digamma_{n+1} \end{vmatrix}$

For $j = n$ *downto* 0 *do with* $k = j + 1$

$$F_j = Ad^{\star}_{f_{j,k}^{-1}} (F_k) + J_j \cdot \overset{\bullet}{V}_j - ad^{\star}_{V_j} (J_j \cdot V_j) \tag{36}$$

$$\tau_j = S_j^T \cdot F_j \tag{37}$$

This allows to calculate numerically the coefficients of (30) in an elementary but lengthy way. By putting $\overset{\bullet}{w} = 0$ and executing one couple of recursion, we obtain in (30): $\tau_j + H_j = 0$ and consequently the σ values of $H(q, w, \breve{g})$. Similarly, with $w = 0$ and $\breve{g} = 0$, (so $H = 0$), by executing n couples of recursion and putting successively $for\ j = 1 \cdots n : \overset{\bullet}{w}_j = 1\ and\ \overset{\bullet}{w}_i = 0\ if\ i \neq j$; we obtain the $\sigma \cdot (\sigma + 1) \diagup 2$ elements of $D(q)$.

6.2 Composite systems

Two adjacent and linked bodies C_j and C_k, with spatial inertia J_j and J_k, are equivalent (in some sense) to a composite system of spatial inertia $\Im = \Im(J_j, J_k)$; and the $\Im$ are determined the following inward recursion:

Given $\Im_{n+1} = 0; \ \delta_{n+1} = 0; \ B_{n+1} = F_{n+1}$

For $j = n$ *dowto* 0 *do with* $k = j+1$

$$\begin{aligned}
\Im_j &= J_j + Ad^{\bigstar}_{f^{-1}_{j,k}}\left(\top_{6\times 6} - \frac{\Im_k \cdot S_k \cdot S_k^T}{S_k^T \cdot \Im_k \cdot S_k}\right) \cdot \Im_k \cdot Ad_{f^{-1}_{j,k}} && (38)\\
\beta_j &= -\, ad^{\bigstar}_{V_j}\,(J_j \cdot V_j) && (39)\\
B_j &= Ad^{\bigstar}_{f^{-1}_{j,k}}\left(\delta_k + \frac{\Im_k \cdot S_k \cdot \left(\tau_k - S_k^T \cdot \delta_k\right)}{S_k^T \cdot \Im_k \cdot S_k}\right) + \beta_j && (40)\\
\delta_j &= \Im_j \cdot \gamma_j + B_j && (41)
\end{aligned}$$

6.3 The articulated body inertia algorithm

Resolve the forward dynamic problem, but only in the not constrained case $(r = 0; \quad \Psi = 0)$.

Given $\overset{\bullet}{V}_{-1} = \begin{vmatrix} 0_{3\times 1} \\ -\Theta_0^{-1} \cdot \breve{g} \end{vmatrix}$; (here $\overset{\bullet}{w}_0 = \begin{vmatrix} \overset{\bullet}{\omega}_0 \\ \overset{\bullet}{v}_0 \end{vmatrix}$)

For $j = 0$ *to* n *do with* $i = j-1$

$$\begin{aligned}
\overset{\bullet}{w}_j &= \left[S_j^T \cdot \Im_j \cdot S_j\right]^{-1} \cdot \left(\tau_j - S_j^T \cdot \left(\Im_j \cdot Ad_{f^{-1}_{i,j}}\left(\overset{\bullet}{V}_i\right) + \delta_j\right)\right) && (42)\\
\overset{\bullet}{V}_j &= Ad_{f^{-1}_{i,j}}\left(\overset{\bullet}{V}_i\right) + S_j \cdot \overset{\bullet}{w}_j + \gamma_j && (43)
\end{aligned}$$

6.4 Calculus of the mass matrix and of its inverse

These calculations are necessary for the general problem with friction to make use of (32).

In the following, symbols without label characterize global matrices, which are all constituted by $(n+1)\cdot(n+1)$ blocks of 6×6 matrices or 6×1 matrices (take care of that (23) and (24) does not follow this convention) and which can thus receive an auxiliary block-by-block labelling from 0 *to* n.

For example in (44) $S_0 \in R^{6\times 6}$; $S_j \in R^{6\times 1}$ *with* $j = 1\cdots n$ so $S \in R^{\xi\times\sigma}$.

$$\begin{aligned}
S &= diag(S_0, S_1, \cdots S_n) \in R^{\xi\times\sigma} && (44)\\
J &= diag(J_0, J_1, \cdots J_n) \in R^{\xi\times\xi} && (45)\\
\Im &= diag(\Im_0, \Im_1, \cdots \Im_n) \in R^{\xi\times\xi} && (46)\\
\Gamma &= \begin{vmatrix}
0_{6\times 6} & 0_{6\times 6} & 0_{6\times 6} & \cdots & 0_{6\times 6}\\
Ad_{f^{-1}_{0,1}} & 0_{6\times 6} & 0_{6\times 6} & \cdots & 0_{6\times 6}\\
0_{6\times 6} & Ad_{f^{-1}_{1,2}} & 0_{6\times 6} & \cdots & 0_{6\times 6}\\
\vdots & \vdots & \vdots & \ddots & \vdots\\
0_{6\times 6} & 0_{6\times 6} & Ad_{f^{-1}_{n-1,n}} & \cdots & 0_{6\times 6}
\end{vmatrix} \in R^{\xi\times\xi} && (47)\\
G &= \begin{vmatrix}
\top_{6\times 6} & 0_{6\times 6} & 0_{6\times 6} & \cdots & 0_{6\times 6}\\
Ad_{f^{-1}_{0,1}} & \top_{6\times 6} & 0_{6\times 6} & \cdots & 0_{6\times 6}\\
Ad_{f^{-1}_{0,2}} & Ad_{f^{-1}_{1,2}} & \top_{6\times 6} & \cdots & 0_{6\times 6}\\
\vdots & \vdots & \vdots & \ddots & \vdots\\
Ad_{f^{-1}_{0,n}} & Ad_{f^{-1}_{1,n}} & Ad_{f^{-1}_{n-1,n}} & \cdots & \top_{6\times 6}
\end{vmatrix} \in R^{\xi\times\xi} && (48)
\end{aligned}$$

We have :

$$G = \left[\top_{\xi\times\xi} - \Gamma\right]^{-1} = \top_{\xi\times\xi} + \sum_{j=1}^{n} [\Gamma]^{j} \tag{49}$$

According to the expression of the kinetic energy, which is $w^T \cdot D \cdot w$, one obtains a first form for the mass matrix:

$$D = S^T \cdot G^T \cdot \Im \cdot G \cdot S \tag{50}$$

Three global matrices are now introduced:

$$\begin{aligned}
\Omega &= S^T \cdot \Im \cdot S \in R^{\sigma\times\sigma} \;\left(\Rightarrow \Omega_0 = \Im_0 \in R^{6\times 6} and\, \Omega_j \in R^{1\times 1}\right) && (51)\\
\Phi &= \Omega^{-1} \cdot S^T \cdot \Im \in R^{\sigma\times\xi} \;\;(\Rightarrow \Phi_0 = \top_{6\times 6}) && (52)\\
\Pi &= \Gamma^T \cdot \Phi^T \in R^{\xi\times\sigma} && (53)
\end{aligned}$$

A square factorization of the mass matrix is:

$$D = \left(\top_{\sigma\times\sigma} + S^T \cdot G^T \cdot \Pi\right) \cdot \Omega \cdot \left(\top_{\sigma\times\sigma} + S^T \cdot G^T \cdot \Pi\right)^T \tag{54}$$

Introducing again the two additional global matrices:

$$\begin{aligned}
\aleph &= (\top_{\xi\times\xi} - S \cdot \Phi) \cdot \Gamma \in R^{\xi\times\xi} && (55)\\
Y &= \left[\top_{\xi\times\xi} - \aleph^T\right]^{-1} = \top_{\xi\times\xi} + \sum_{j=1}^{n} \left[\aleph^T\right]^j \in R^{\xi\times\xi} && (56)
\end{aligned}$$

One have the relation:

$$J = \Im - \aleph^T \cdot \Im \cdot \aleph \tag{57}$$

And one obtains a square factorization of the inverse of the mass matrix as:

$$D^{-1} = \left(\top_{\sigma\times\sigma} - S^T \cdot Y \cdot \Pi\right)^T \cdot \Omega^{-1} \cdot \left(\top_{\sigma\times\sigma} - S^T \cdot Y \cdot \Pi\right) \tag{58}$$

The corresponding recursive algorithm, which contain the 6×6 or 6×1 blocks or matrices, is:

Given $\Im_{n+1} = 0$; $\aleph^T_{n+1,n} = 0$.

For $j = n \quad downto \quad 0 \quad do \quad with \quad k = j + 1 \; and \; i = j - 1$

$$\begin{aligned}
\Im_j &= J_j + \aleph_k^T \cdot \Im_k \cdot \aleph_k && (59)\\
\Omega_j &= S_j^T \cdot \Im_j \cdot S_j && (60)\\
\Phi_j^T &= \Im_j \cdot S_j \cdot \Omega_j^{-1} && (61)\\
\Pi_{i,j} &= Ad^{\star}_{f_{i,j}^{-1}} \left(\Phi_j^T\right) && (62)\\
\aleph_{j,i}^T &= Ad^{\star}_{f_{i,j}^{-1}} \left(\top_{6\times 6} - \Phi_j^T \cdot S_j^T\right) && (63)
\end{aligned}$$

Note that the labelled symbols $\Pi_{-1,0}$ and $\aleph^T_{-1,0}$ which appear formally for $j = 0$ are not used.

References

[1] Radu Balescu, *Statistical Dynamics - Matter out of Equilibrium*, Imperial College Press World Scientific, (Singapore 1977), pp 13.

[2] J. Luh, M. Walker and R. Paul, *On-line computational scheme for mechanical manipulators*, ASME Journal Dynamic Systems, Measurement and Control **102** (1980), n°. 2, 69-76. Here t*he recursive Newton-Euler algorithm is described with ordinary vectorial notations.*

[3] Scott R. Ploen, *Geometric Algorithms for the Dynamics an Control of Multibody Systems,* Ph. D. Thesis, University of California, Irvine; Department of Mechanical Engineering, 1997. Can be downloaded at URL
"http://www.eng.uci.edu/~bobrow/ ".

[4] Garett A. Sohl and James E. Bobrow, *A Recursive Multibody Dynamics and Sensitivity Algorithm for Branched Kinematics Chain* , ASME Journal Dynamic Systems, Measurement and Control **123** (2001), n°. 3, 391-405. Can be downloaded at URL "http://www.eng.uci.edu/~bobrow/ ".