Nonlinear Dynamics **5**: 209–231, 1994.

Singularity-Free Augmented Lagrangian Algorithms for Constrained Multibody Dynamics

E. BAYO
Department of Mechanical Engineering, University of California, Santa Barbara, CA 93106, U. S. A.

and

A. AVELLO
Dept. of Applied Mechanics, University of Navarra and CEIT, Manuel de Lardizábal 15, 20009 San Sebastián, Spain

(Received: 13 March 1992; accepted: 4 August 1992)

Abstract. After a general review of the methods currently available for the dynamics of constrained multibody systems in the context of numerical efficiency and ability to solve the differential equations of motion in singular positions, we examine the acceleration based augmented Lagrangian formulations, and propose a new one for holonomic and non-holonomic systems that is based on the canonical equations of Hamilton. This new one proves to be more stable and accurate that the acceleration based counterpart under repetitive singular positions. The proposed algorithms are numerically efficient, can use standard conditionally stable numerical integrators and do not fail in singular positions, as the classical formulations do. The reason for the numerical efficiency and better behavior under singularities relies on the fact that the leading matrix of the resultant system of ODEs is sparse, symmetric, positive definite, and its rank is independent of that of the Jacobian of the constraint equations. The latter fact makes the proposed method particularly suitable for singular configurations.

Key words: Constrained multibody systems, penalty and augmented Lagrangian method, holonomic and non-holonomic constraints, canonical equations of Hamilton.

1. Review of Current Approaches for Multibody Dynamics

Computer systems, while increasing tremendously in power in recent years, are so affordable nowadays, that their use have become widely spread in many different fields and for a number of applications. The computer kinematic and dynamic analysis of multibody systems is increasingly being used in fields such as the automobile industry, aerospace, robotics, machinery, biomechanics, etc., and it has been receiving considerable attention recently, as seen by the amount of literature on multibody simulation and computer aided analysis programs being sold in the market of engineering software. Nevertheless, there is an increasing demand for faster and more reliable simulations that must be based on more efficient and robust algorithms for multibody dynamics.

The dynamic analysis of multibody systems is a process which is most appropriately performed using interactive (rather than batch) type of analysis. The analyst is interested in visualizing a whole set of successive responses of the multibody, a simulation of its behavior and operation over all the mechanism workspace and over a certain period of time. In certain cases it may be even necessary to introduce the engineer as an additional element in the simulation, called "man-in-the-loop", who may act by introducing external forces or control over specific degrees of freedom. In any case, each response over a time step needs to be calculated and displayed at the highest speed possible in order to give a picture that will possibly resemble the actual motion of the system in both time and space: *the real-time behavior.*

While it is important for multibody dynamic simulations to have fast and accurate interactive graphical interfaces, it is essential that the computer software relies on good numerical algorithms that will permit a fast and reliable solution of the resulting algebraically constrained differential equations of motion. Consequently, it becomes very important that numerical efficiency and stability be combined with robustness so that the simulation does not reach dead-lock situations due to singularity positions in the multibody motion.

In recent years, some important advances have been made in the development of new formulations for multibody dynamics. Some formulations, stemming mainly from the robotics field, have been especially conceived for real-time simulation and are based on the recursive computation of some or all of the terms in the equations of motion [1-4]. Some of these algorithms are $O(N)$ meaning that the number of floating point arithmetic operations grows linearly with the number of degrees of freedom. Others require the solution of a system of N linear equations and, therefore, are of order $O(N^3)$ (if Gaussian elimination is used). Although it has been demonstrated by Featherstone [3] that the best $O(N^3)$ algorithms are faster than the best $O(N)$ algorithms for $N < 10$, the elegance and attractiveness of the $O(N)$ Featherstone's formulation has exerted a strong influence on later developments that have generalized these ideas for non-serial (tree-configuration) and closed-loop systems [5-7]. A limitation arises when closed-chain multibodies are analyzed, since for these cases special provisions must be made to account for the reaction forces between the different loops [6].

The second group of methods encompasses those that reduce the equations of motion in dependent coordinates to a minimum set of independent ones via a transformation matrix obtained from the nullspace of the Jacobian of the constraint equations. Different methods of choosing the independent set of coordinates and generating the transformation matrix have been proposed [8-14]. The concept of *velocity transformations,* initially introduced by Jerkovsky [15], has been subsequently extended into efficient algorithms [16-19] that avoid the Jacobian factorization, and allow for an efficient and simple way of generating the equations of motion in independent coordinates in a way that can be fully parallelized [19].

The classical way to generate the equations of motion is to use dependent (or absolute) coordinates to generate and solve the equations of motion [14]. These algorithms are based on the classical Lagrange's formulation which leads to a set of differential and algebraic equations (DAE) of motion with the coordinates and multipliers as unknowns. The solution of these equations require special techniques [20] whose merit has not been thoroughly calibrated yet for the integration of multibody systems. A way to avoid the DAE is by differentiating the constraints. The resulting constraint violations are commonly stabilized using the method proposed by Baumgarte [21]. An extension for violation stabilization of holonomic systems based on the use of the canonical momenta has been proposed in [22].

Although the methods described above are well established (some of them very efficient numerically), they can not directly handle redundant constraints. In fact these have to be eliminated prior to the dynamic analysis. In addition they all fail to give successful solutions when the multibody undergoes a singular position. A partial solution to the problem of singular positions was provided in [23] where a method is developed that detects the ill-conditioning of the Jacobian matrix so that the integrator can step over it. In [24-25] *a regularization method* is proposed to cope with singularities. The main idea consists in adding to the vanishing and the linearly dependent constraints their third derivatives, and this turns the Jacobian nonsingular. A staggered stabilization approach was presented by Park and Chiou [26] which was later refined by means of an explicit-implicit integration procedure [27]. This method integrates two different sets of equations one for the coordinates and another for the Lagrange multipliers, and avoids the singular position problem of the equations of motion. A small limitation, however, is that it requires the inversion of the mass matrix, which is in general semi-positive definite and may not have an inverse in certain instances (in particular when redundant dependent coordinates are used).

Bayo, Garcia de Jalon and Serna proposed a penalty method [28] by which the acceleration, velocity and position constraint conditions are added to the equations of motion as a "dynamical penalty system" to obtain a simple and efficient formulation for the dynamic equations. The appeal of this formulation lies in two main points. Firstly, it leads to a reduced set of equations in the form $\dot{\mathrm{y}} = \mathrm{g}(\mathrm{y}, t)$ that can be integrated by standard conditionally stable numerical algorithms, without the need of further stabilization techniques to control the violation of the constraints during the integration process. Secondly, unlike the classical methods which rely on the Jacobian, this penalty formulation leads to matrices that can be inverted even in singular positions, and in the presence of redundant (linearly dependent) constraints and coordinates. Important theoretical studies of its convergence and stability have been carried out in [29] and [30]. The penalty method of [28] has also been successfully extended to real time dynamics within the context of fully Cartesian coordinates in [31]. There, it has been shown that the penalty method requires the factorization of a symmetric matrix that is dominated by the terms in the main diagonal (no pivoting is required), and is strongly banded, feature that makes it an order n method, where n is the number of coordinates. In addition, the different steps of the algorithm can be parallelized, making this method suitable for very large systems.

It was also proposed in [28] a more complete and accurate *augmented Lagrangian* method (combination of the penalty formulation and Lagrange's multipliers), which allows for convergence independently of the penalty values and which yields the constraint forces (Lagrange multipliers) as a by-product without having to integrate additional equations. In this paper we examine this augmented Lagrangian formulation within the context of singular positions and, in addition, propose a new one based on the use of the canonical equations of Hamilton that is even more stable and numerically efficient than the previous one.

2. Preliminaries on the Classical Formulations

2.1. Acceleration Based Lagrange's Multiplier Formulation

Let us consider a multibody system whose configuration is characterized by n generalized coordinates q that are interrelated through the m holonomic kinematic constraint conditions

$$\mathbf{\Phi}(\mathrm{q}, t) = \mathbf{0} \tag{1}$$

Let L be the system Lagrangian, defined by $L = T - V$, where T and V are the kinetic and potential energy, respectively; and let Q be the vector of external and non-conservative forces. The Lagrange equations of such a system can be written as [32]

$$\frac{\mathrm{d}}{\mathrm{dt}}\left(\frac{\partial L}{\partial \dot{\mathrm{q}}}\right) - \frac{\partial L}{\partial \mathrm{q}} + \mathbf{\Phi}_{\mathrm{q}}^{T}\lambda = \mathbf{Q}, \tag{2}$$

which for a general multibody system leads to:

$$\mathrm{M}\ddot{\mathrm{q}} + \Phi_{\mathrm{q}}^{T}\lambda = \mathrm{Q} + L_{\mathrm{q}} - \dot{\mathrm{M}}\dot{\mathrm{q}}, \tag{3}$$

where M is the mass matrix, L_{q} is the partial derivative of the Lagrangian with respect to the coordinates, Φ_{q} is the Jacobian of the constraint equations, Q is the vector of external and non-conservative forces, and λ is the vector that contains the Lagrange's multipliers. Equations (1) and (3) constitute a set of $n + m$ mixed differential algebraic equations (DAE) of index three [20], with q and λ as unknowns. In order to avoid the direct integration of DAEs, a double differentiation of the constraints equations may be carried out, which along with the Baumgarte's stabilization [21] yields:

$$\begin{bmatrix} \mathrm{M} & \Phi_{\mathrm{q}}^{T} \\ \Phi_{\mathrm{q}} & 0 \end{bmatrix} \begin{Bmatrix} \ddot{\mathrm{q}} \\ \lambda \end{Bmatrix} = \begin{Bmatrix} \mathrm{Q} + L_{\mathrm{q}} - \dot{\mathrm{M}}\dot{\mathrm{q}} \\ -\dot{\Phi}_{\mathrm{q}}\dot{\mathrm{q}} - \dot{\Phi}_{t} - a\dot{\Phi} - b\Phi \end{Bmatrix}, \tag{4}$$

where a and b are the stabilization constants. These equations can now be integrated using standard numerical integrators [33] with each function evaluation performed using equation (4).

2.2. Lagrange's Multiplier Formulation in Canonical Form

The definition of the *conjugate or canonical momenta* can be taken from classical mechanics [32]

$$\mathrm{p} = \frac{\partial L}{\partial \dot{\mathrm{q}}} \tag{5}$$

along with the *Hamiltonian*

$$H = \mathrm{p}^{T}\dot{\mathrm{q}} - L. \tag{6}$$

The *canonical equations of Hamilton* for a constrained system are formulated as

$$\dot{\mathrm{q}} = \frac{\partial H}{\partial \mathrm{p}} \tag{7a}$$

$$-\dot{\mathrm{p}} = \frac{\partial H}{\partial \mathrm{q}} - \mathrm{Q} + \Phi_{\mathrm{q}}^{T}\lambda. \tag{7b}$$

In the case of multibody systems the Lagrangian L is defined in terms of q, $\dot{\mathrm{q}}$ and t, and rather than following a lengthy process to form the Hamiltonian as an explicit function of q, p and t, and then differentiate as in (7a), the canonical equations can be directly obtained from (5) and (7b). Since the system kinetic energy is a quadratic function of the generalized velocities, (5) and (7b) directly lead to the following set of equations in matrix form

$$\mathrm{p} = \mathrm{M}\dot{\mathrm{q}} \tag{8a}$$

$$\dot{\mathrm{p}} = L_{\mathrm{q}} + \mathrm{Q} - \Phi_{\mathrm{q}}^{T}\lambda. \tag{8b}$$

The combination of (8a-b) and (1) constitutes a system of $2n + m$ differential and algebraic equations (DAE), of index two. Note, that although equations (8a-b) have n more equations than (3), $\dot{\mathrm{p}}$ can be obtained explicitly from (8b). In addition, index two DAEs are better behaved than index three DAEs [20], and therefore the consideration of equations (8a-b) may be numerically more advantageous than (3), when using algorithms for the solution of the mixed differential algebraic equations.

In order to avoid the mixed differential and algebraic equations, the system Lagrangian is modified in [22] and [34] to include the kinematic velocity constraints as

$$L^* = L + \dot{\mathbf{\Phi}}^T \sigma, \tag{9}$$

where σ are the new Lagrange multipliers. The new Hamiltonian is $H = \mathrm{p}^T \dot{\mathrm{q}} - L^*$ and the application of (3) and (5b) leads to

$$\mathrm{p}_{new} = \mathbf{M}\, \dot{\mathrm{q}} + \mathbf{\Phi}_{\mathrm{q}}^T \sigma \tag{10a}$$

$$\dot{\mathrm{p}}_{new} = L_{\mathrm{q}} + \mathbf{Q} + \dot{\mathbf{\Phi}}_{\mathrm{q}}^T \sigma \tag{10b}$$

that along with

$$\dot{\mathbf{\Phi}} + \gamma \mathbf{\Phi} = \mathbf{\Phi}_{\mathrm{q}} \dot{\mathrm{q}} + \mathbf{\Phi}_t + \gamma \mathbf{\Phi} = \mathbf{0} \tag{11}$$

constitutes a set of $2n+m$ ordinary differential equations (ODE), with p, q and σ as unknowns. The real constant γ provides asymptotic stability of the stabilization scheme. It can be very easily verified by differentiation of (10a) and substitution in (10b) that $\dot{\sigma} = \lambda$.

It is worth mentioning that only the following $n + m$ equations need be solved at each time step in the numerical implementation of the algorithm:

$$\begin{bmatrix} \mathbf{M} & \mathbf{\Phi}_{\mathrm{q}}^{\mathrm{T}} \\ \mathbf{\Phi}_{\mathrm{q}} & 0 \end{bmatrix} \begin{Bmatrix} \dot{\mathrm{q}} \\ \sigma \end{Bmatrix} = \begin{Bmatrix} \mathrm{p}_{new} \\ -\mathbf{\Phi}_t - \gamma \mathbf{\Phi} \end{Bmatrix}. \tag{12}$$

The numerical simulations of [34] show that since only the first time derivative of the constraints is used, the integration of this equations is more efficient and more stable, than the acceleration based counterparts.

2.3. Reduction to an Independent Set of Coordinates

The other widely accepted group of methods for multibody dynamics is based on the use of a transformation matrix $\mathbf{R}$ that will reduce the equations of motion to a minimum set of coordinates. The matrix $\mathbf{R}$ is obtained from the concept of the *nullspace* of the Jacobian, and allows one to express equation (3) in terms of an independent set of coordinates. The procedure starts by differentiating the constraint equations $\mathbf{\Phi}(\mathrm{q}) = \mathbf{0}$, that for simplification purposes we assume are sclerenomous, to obtain

$$\mathbf{\Phi}_{\mathrm{q}} \dot{\mathrm{q}} = \mathbf{0}. \tag{13}$$

It may be seen from equation (13) that $\dot{\mathrm{q}}$ belongs to the nullspace of the Jacobian $\mathbf{\Phi}_{\mathrm{q}}$. The dimension of the nullspace is equal to f, where f is the number of degrees of freedom of

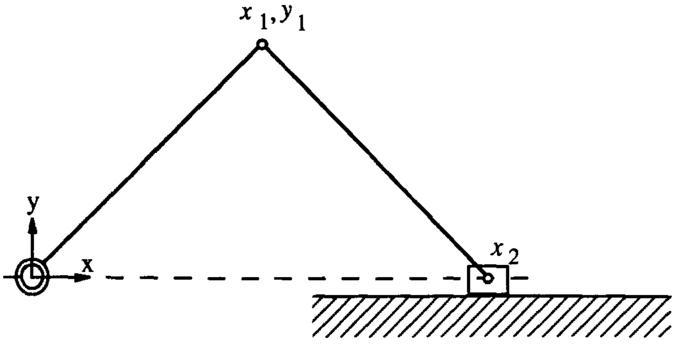

Fig. 1. Slider-crank mechanism.

the multibody system. We can always express $\dot{\mathbf{q}}$ as a linear combination of the vectors of a nullspace's basis, in the form

$$\dot{\mathbf{q}} = \mathbf{R}\dot{\mathbf{z}}, \tag{14}$$

where $\mathbf{R}$ is an $n \times f$ matrix whose columns constitute a basis of the nullspace, and $\dot{\mathbf{z}}$ are the f independent velocities. Since $\mathbf{R}$ constitutes a basis of the nullspace of the Jacobian, it satisfies the relationship $\mathbf{\Phi}_\mathbf{q}\mathbf{R} = \mathbf{0}$. The matrix $\mathbf{R}$ may be obtained from the Jacobian by projection methods using Gauss factorization [8], the singular value decomposition [9] or the QR method [10]. It can also be obtained more efficiently by velocity transformations [15-19]. The substitution of (14) into (3) and premultiplication by $\mathbf{R}^T$ yields:

$$\mathbf{R}^T\mathbf{M}\,\mathbf{R}\,\ddot{\mathbf{z}} = \mathbf{R}^T(\mathbf{Q} + L_\mathbf{q} - \dot{\mathbf{M}}\dot{\mathbf{q}}) - \mathbf{R}^T\mathbf{M}\,\dot{\mathbf{R}}\,\dot{\mathbf{z}} \tag{15}$$

from which $\ddot{\mathbf{z}}$ can be calculated. An extension of this method within the setting of canonical equations has been proposed in [35], where the same leading matrix is obtained.

2.4. Why the Classical Formulations Fail in Singular Positions

As mentioned before, a singular position is encountered when the multibody reaches a configuration in which there is a sudden change in the number of degrees of freedom. For instance, a slider-crank mechanism as the one shown in Figure 1, reaches a singular position when the two links are in vertical position. In that configuration, both links are coincident and the mechanism has not one but two degrees of freedom. These two degrees of freedom correspond to the two possible motions (bifurcations) that the mechanism can undergo, and which are illustrated in Figure 2. Figure 2a shows the first possible motion that corresponds to a slider-crank mechanism, Figure 2b, shows the second motion corresponding to a rotating bar (in fact two coincident rotating bars). As may be seen, a singular position implies a bifurcation point, in which the mechanism can, at least theoretically, undergo different paths.

The existence of a singular position with both, the classical Lagrange's multipliers approach and the reduction to a set of independent coordinates, is invariably detected when the Jacobian matrix of the constraints becomes rank deficient. These formulations are based on the decomposition of the Jacobian and since its rank suddenly falls at a singular position, the

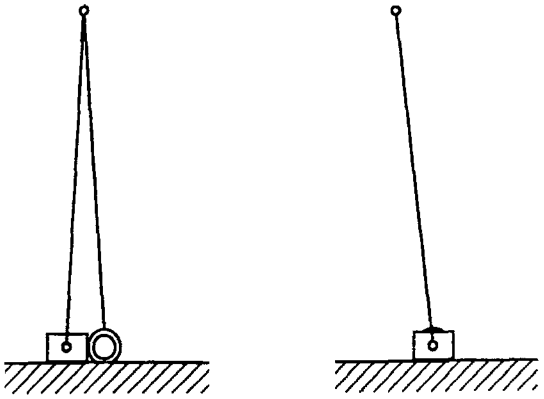

Fig. 2. a. Slider-crank mechanism motion. b. Rotating bar motion.

decomposition fails and therefore no solution can be found. The simulation then crashes not because of the physics of the problem, but because of the inability of the dynamic formulation to overcome the sudden change in the rank of the Jacobian.

Equations (4) and (12) are the key equations for the solution of the dynamics using the Lagrange's multipliers method. Assuming that all the constraints are independent, that is if $m = n - f$, the rank of the leading matrices in those equations is $n + m$. Since the Jacobian becomes rank deficient in singular positions, this matrix becomes singular. This means that the accelerations (or velocities) may not be computed unless special care is taken to eliminate (or regularize), at that particular position of the multibody system, all the vanishing constraints. Otherwise, the dynamic simulation crashes at this point. Equation (15) is the alternate key equation for the independent coordinate method. Again, when a singular position is reached special provisions have to be made for the computation of the matrix **R**.

If a singular position is not exactly reached, the leading matrix of both classical methods will not be strictly singular, but near singular, with a very high condition number. If this situation is not correctly tracked, the integration and round-off errors will be amplified and the resulting solutions may be totally erroneous.

It is important at this stage to emphasize the difference between a singular Jacobian and a singular position. While a singular position always implies a singular Jacobian, the converse is not always true. A Jacobian can become singular when redundant constraints are present, a dead-lock position is reached, or, when the coordinate partitioning between dependent and independent coordinates is not made properly or has not been updated for a while. Contrary to the case of a singular position, these singularities can be easily avoided and the simulation may proceed smoothly. The difference between singular Jacobian and singular positions can be better understood by partitioning the columns of the Jacobian $\mathbf{\Phi}_\mathrm{q}$ into two submatrices $\mathbf{\Phi}_\mathrm{q}{}^d$ and $\mathbf{\Phi}_\mathrm{q}{}^i$, corresponding to the dependent and independent coordinates, respectively. This partition is made so that $\mathbf{\Phi}_\mathrm{q}{}^d$ has full row rank. When $\mathbf{\Phi}_\mathrm{q}{}^d$ is rank-deficient but $\mathbf{\Phi}_\mathrm{q}$ has full row rank the singularity is easily avoidable since the full rank of $\mathbf{\Phi}_\mathrm{q}{}^d$ can be recovered by a new suitable choice of independent coordinates. However, at a singular position $\mathbf{\Phi}_\mathrm{q}$ looses rank all of a sudden, and the singularity may only be avoided by eliminating the non-active constraints. As pointed out in [25] if these non-active constraints are eliminated in the neighborhood of the singular configurations the corresponding constraint forces become zero and this may result in a fast deviation of the simulation from the constrained behavior.

3. Acceleration Based Augmented Lagrangian Formulation

3.1. Description of the Method

For the sake of completeness and in order to facilitate the understanding of the methods proposed in sections 4 and 6, we present in this section the augmented Lagrangian method introduced in [28]. Later we will address its behavior in singular positions. Given a multibody system with holonomic constraint equations of the form given in (1), which represent a set of nonlinear algebraic equations in the coordinates and the time variable. The penalty-augmented Lagrangian formulation proposed in [28] is derived by adding to the Lagrangian two terms: a fictitious potential

$$V^* = \sum_k \frac{1}{2}\alpha_k\, \omega_k^2 \equiv \frac{1}{2}\mathbf{\Phi}^T \boldsymbol{\alpha}\, \mathbf{\Omega}^2 \mathbf{\Phi} \tag{16}$$

and a fictitious kinematic energy term

$$T^* = \sum_k \frac{1}{2}\alpha_k \left(\frac{d\mathbf{\Phi}_k}{dt}\right)^2 \equiv \frac{1}{2}\dot{\mathbf{\Phi}}^T \boldsymbol{\alpha}\, \dot{\mathbf{\Phi}}. \tag{17}$$

A set of Rayleigh's dissipative forces is also added to the system

$$G_k = -2\alpha_k\, \omega_k\, \mu_k \frac{d\phi_k}{dt} \equiv -2\boldsymbol{\alpha}\, \mathbf{\Omega}\, \boldsymbol{\mu}\, \dot{\mathbf{\Phi}}, \tag{18}$$

where α_k are large positive real values (penalty numbers), and ω_k and μ_k represent the natural frequency and the damping ratio of the dynamic penalty system (mass, dashpot and spring) corresponding to the constraint $\Phi_k = 0$. Matrices $\boldsymbol{\alpha}$, $\mathbf{\Omega}$ and $\boldsymbol{\mu}$ are $m \times m$ diagonal matrices that contain the values of the penalty numbers, the natural frequencies and the damping ratios of the penalty systems assigned to each constraint condition. If the same values are used for each constraint these matrices become identity matrices multiplied by the respective penalty numbers. Note that in equations (16) through (18) we have used both index as well as matrix notation, hoping that this will lead to a better understanding of the physical significance of the different terms. In the following discussion we will only use the matrix form in order to be consistent with the notation used so far in the paper.

The differentiation of the new Lagrangian leads to

$$\frac{\partial L^*}{\partial \mathrm{q}} = L_{\mathrm{q}} + \dot{\mathbf{\Phi}}_{\mathrm{q}}^T \boldsymbol{\alpha}\, \dot{\mathbf{\Phi}} - \mathbf{\Phi}_{\mathrm{q}}^T \boldsymbol{\alpha}\, \mathbf{\Omega}^2 \mathbf{\Phi} \tag{19}$$

$$\frac{\partial L^*}{\partial \dot{\mathrm{q}}} = \mathrm{M}\, \dot{\mathrm{q}} + \mathbf{\Phi}_{\mathrm{q}}^T \boldsymbol{\alpha}\, \dot{\mathbf{\Phi}} \tag{20}$$

$$\frac{\mathrm{d}}{\mathrm{dt}}\left(\frac{\partial L^*}{\partial \dot{\mathrm{q}}}\right) = \mathrm{M}\, \ddot{\mathrm{q}} + \dot{\mathrm{M}}\, \dot{\mathrm{q}} + \dot{\mathbf{\Phi}}_{\mathrm{q}}^T \boldsymbol{\alpha}\, \dot{\mathbf{\Phi}} + \mathbf{\Phi}_{\mathrm{q}}^T \boldsymbol{\alpha}\, \ddot{\mathbf{\Phi}}, \tag{21}$$

where the relation $\ddot{\mathbf{\Phi}}_{\mathbf{q}}^{T} = \dot{\mathbf{\Phi}}_{\mathbf{q}}^{T}$, which can be easily verified, has been used. L is the Lagrangian corresponding to the system without constraints.

The work done by the fictitious Rayleigh forces is

$$\delta W_R = -2(\delta\mathbf{\Phi})^T \boldsymbol{\alpha}\ \boldsymbol{\Omega}\ \boldsymbol{\mu}\ \dot{\mathbf{\Phi}} = -2\delta\mathrm{q}^T \mathbf{\Phi}_{\mathbf{q}}^T \alpha\ \boldsymbol{\Omega}\ \mu\ \dot{\mathbf{\Phi}}. \tag{22}$$

Therefore the final expression obtained by the application of the Lagrange's equations (3) is

$$\mathbf{M}\ddot{\mathrm{q}} + \mathbf{\Phi}_{\mathbf{q}}^{\mathbf{T}}\alpha\left(\ddot{\mathbf{\Phi}} + 2\mathbf{\Omega}\mu\dot{\mathbf{\Phi}} + \mathbf{\Omega}^{2}\mathbf{\Phi}\right) + \mathbf{\Phi}_{\mathbf{q}}^{\mathbf{T}}\lambda^{*} = \mathbf{Q} + L_{\mathbf{q}} - \dot{\mathbf{M}}\dot{\mathrm{q}}, \tag{23}$$

where λ^* are the new Lagrange multipliers of the modified system. Note that the second term in the LHS of equation (23) represents the projection in the direction of the coordinates $\mathbf{q}$ of all the internal forces that are generated by the penalty system when the constraints $\mathbf{\Phi}$, $\dot{\mathbf{\Phi}}$ and $\ddot{\mathbf{\Phi}}$ are violated. Introducing $\ddot{\mathbf{\Phi}} = \mathbf{\Phi}_{\mathbf{q}}\ddot{\mathrm{q}} + \dot{\mathbf{\Phi}}_{\mathbf{q}}\dot{\mathrm{q}} + \dot{\mathbf{\Phi}}_t$ the final result is obtained

$$\begin{aligned}&\left(\mathbf{M} + \mathbf{\Phi}_{\mathbf{q}}^{\mathbf{T}}\alpha\mathbf{\Phi}_{\mathbf{q}}\right)\ddot{\mathrm{q}} + \mathbf{\Phi}_{\mathbf{q}}^{\mathbf{T}}\lambda^{*}\\ &\quad = \mathbf{Q} + L_{\mathbf{q}} - \dot{\mathbf{M}}\dot{\mathrm{q}} - \mathbf{\Phi}_{\mathbf{q}}^{\mathbf{T}}\alpha\left(\dot{\mathbf{\Phi}}_{\mathbf{q}}\dot{\mathrm{q}} + \dot{\mathbf{\Phi}}_t + 2\mathbf{\Phi}\mu\dot{\mathbf{\Phi}} + \mathbf{\Omega}^{2}\mathbf{\Phi}\right).\end{aligned} \tag{24}$$

This equation may be viewed as the "generic penalty method" [28] to which the Lagrange's multipliers are added. As clearly shown in [28] this augmented Lagrangian formulation allows the analyst to choose from a wide range of penalty values that assure convergence and avoid numerical ill-conditioning. As we will see later, the solution provided by this method is not sensitive to the value taken by the penalty factor, and therefore, equation (24) represents and elegant and attractive way of avoiding the problems customarily attributed to the penalty formulations.

It is important to note that there is a very important difference between equation (24) and the classical dynamic algorithms represented by (4) and (15). As we indicated before, the leading matrices of the latter equations become singular in singular positions. However, although the mass matrix M is in general positive semi-definite, it is always strictly positive definite in the nullspace of the Jacobian matrix. Therefore, a look at equation (24) reveals that its leading matrix $(\mathbf{M} + {\mathbf{\Phi}_{\mathbf{q}}}^{T}\alpha\mathbf{\Phi}_{\mathbf{q}})$ is always positive definite, which means that it can always be factorized, even in singular positions and/or with linearly dependent constraints. In practice, the augmented Lagrangian formulation is superior to the generic penalty method since the former allows for smaller values of the penalty parameter, hence, for a better conditioning of the leading matrix.

In equation (24) the Lagrange's multipliers λ^* play the role of correcting terms. In the limit the constraint conditions are satisfied, thus $\lambda = \lambda^*$ and equations (4) and (24) become equivalent except for round off errors induced by the penalty parameter and finite machine precision. By comparing those two equations one can infer that

$$\lambda \cong \lambda^{*} + \alpha\left(\ddot{\mathbf{\Phi}} + 2\mathbf{\Omega}\mu\dot{\mathbf{\Phi}} + \mathbf{\Omega}^{2}\mathbf{\Phi}\right). \tag{25}$$

We are seeking the solution of (24) without having to use the algebraic constraint equations (1). This requires that the correct values of λ^* be known so that they can be inserted in (24). Since those values are not known in advance we need to set up an iterative process that calculates the unknown multipliers λ^*. The iteration is easily established by taking advantage of equation (25)

$$\lambda^*_{i+1} = \lambda^*_i + \alpha\left(\ddot{\Phi} + 2\Omega\mu\dot{\Phi} + \Omega^2\Phi\right)_{i+1}, \qquad i = 0, 1, 2, ... \tag{26}$$

with $\lambda^*_0 = 0$ for the first iteration. Equation (26) physically represents the introduction at iteration $i+1$ of forces that tend to compensate the fact that the addition of all the constraint terms are not exactly zero. It turns out that with the augmented Lagrangian formulation, the penalty numbers do not need to be very large (thus leading to a better numerical conditioning) since the resulting error in the constraint equations will be eliminated by the Lagrange's terms during the iteration procedure. Also note that the "generic penalty" [28] method corresponds to the augmented Lagrangian formulation in which the iteration process is only carried out once.

The matrix formulation of (24), including the iterative process defined in (26), is given by the following expression:

$$\begin{aligned}\left(\mathrm{M} + \Phi_{\mathrm{q}}^{\mathrm{T}}\alpha\Phi_{\mathrm{q}}\right)\ddot{\mathrm{q}}_{i+1} &= \mathrm{M}\ddot{\mathrm{q}}_i - \Phi_{\mathrm{q}}^{\mathrm{T}}\alpha\left(\dot{\Phi}_{\mathrm{q}}\dot{\mathrm{q}} + \dot{\Phi}_t + 2\Omega\mu\dot{\Phi} + \Omega^2\Phi\right)\\ & i = 0, 1, 2,\end{aligned} \tag{27}$$

where the subscript i represents the iteration number, and $\mathrm{M}\,\ddot{\mathrm{q}}_{\mathrm{o}} = \mathrm{Q} + L_{\mathrm{q}} - \dot{\mathrm{M}}\dot{\mathrm{q}}$ for the initial iteration. Equation (27) may be used to iterate until $\|\ddot{\mathrm{q}}^{(i+1)} - \ddot{\mathrm{q}}^{(\mathrm{i})}\| < \varepsilon$, where ε is a user-specified tolerance.

The main advantage of using equation (27) is that the penalty terms are in fact used as an intermediate tool in order to compute the Lagrange's multipliers for which no new equations are integrated: only n equations are solved in the integration process. Therefore, the value of the penalty factor α does not affect the solution, but only the convergence rate. Experience shows that when the constraints are scaled to unity, penalty factors ranging from 10^5 to 10^7 give a good convergence rate, and only 2 to 4 iterations are required to converge to the machine precision, in double precision arithmetic.

Note that the added cost of using equation (27) to refine the solution and obtain the Lagrange multipliers is fairly small, since its leading matrix remains constant during the iteration process needed for a function evaluation. Therefore, at each iteration step only the computation of the independent term and a forward and a backward substitutions are required. The numerical implementation of the algorithm using standard integrators [33], available in commercial mathematical libraries, is rather simple and may be described as follows

ALGORITHM ALF1

Given q and $\dot{\mathrm{q}}$ at time step l.

1. Use (27) iteratively to solve for $\ddot{\mathrm{q}}$, with $\mathrm{M}\,\ddot{\mathrm{q}}_0 = \mathrm{Q} + L_{\mathrm{q}} - \dot{\mathrm{M}}\dot{\mathrm{q}}$ for the initial iteration. At the end of each iteration use (26) to calculate the Lagrange multipliers λ^*, if desired
2. Call the numerical integration subroutine (n.i.s) to compute q and $\dot{\mathrm{q}}$ at time step $l+1$.
3. Upon convergence of the n.i.s update the time variable and go to step 1.

We have used this algorithm very successfully in multibody dynamics simulation and has turned out to be very efficient and accurate. However, we have noticed that under repetitive singular conditions this algorithm may lead to unstable behavior (see examples below) due to the accumulation of small violations of the constraints during the integration process. This lead us to propose a more robust augmented Lagrangian method based on the canonical equations of Hamilton that is presented in the next section.

4. Augmented Lagrangian Formulation in Canonical Form

4.1. Basic Augmented Lagrangian Formulation in Canonical Form

Let us consider equation (9) as the starting point to build a modified Lagrangian that will not only contain the Lagrange multipliers σ but also the penalty terms of the previous section. Accordingly:

$$L^* = L + \frac{1}{2}\dot{\Phi}^T\alpha\dot{\Phi} - \frac{1}{2}\Phi^T\Omega^2\alpha\Phi + \dot{\Phi}^T\sigma^*. \tag{28}$$

In the limit when the constraint conditions are satisfied, the penalty terms vanish and $\sigma = \dot{\sigma}^*$. Similar to the Lagrange's formulation $\dot{\sigma}^* = \lambda^*$ and after the augmented Lagrangian iteration when the constraints are satisfied to machine precision $\dot{\sigma} = \lambda$. The differentiation of L^* with respect to $\dot{\mathrm{q}}$ leads to the following new canonical momenta in matrix form

$$\mathrm{p} = \frac{\partial L^*}{\partial \dot{\mathrm{q}}} = \mathrm{M}\dot{\mathrm{q}} + \mathbf{\Phi}_{\mathrm{q}}^{\mathrm{T}}\alpha\dot{\mathbf{\Phi}} + \mathbf{\Phi}_{\mathrm{q}}^{\mathrm{T}}\sigma^*, \tag{29}$$

where we have eliminated the subindex 'new' of (10a) for practical convenience. The modified *Hamiltonian* can be written as $H^* = \mathrm{p}^T\dot{\mathrm{q}} - L^*$ and the use of (7), including the Rayleigh forces of (18), leads to

$$\left[\mathrm{M} + \mathbf{\Phi}_{\mathrm{q}}^{\mathrm{T}}\alpha\mathbf{\Phi}_{\mathrm{q}}\right]\dot{\mathrm{q}} = \mathrm{p} - \mathbf{\Phi}_{\mathrm{q}}^{\mathrm{T}}\alpha\mathbf{\Phi}_{\mathrm{t}} + \mathbf{\Phi}_{\mathrm{q}}^{\mathrm{T}}\sigma^* \tag{30a}$$

$$\dot{\mathrm{p}} = \mathrm{Q} + L_{\mathrm{q}} + \dot{\mathbf{\Phi}}_{\mathrm{q}}^{\mathrm{T}}\alpha\dot{\mathbf{\Phi}} - \mathbf{\Phi}_{q}^{T}\alpha\left(\Omega^2\mathbf{\Phi} + 2\Omega\mu\dot{\mathbf{\Phi}}\right) + \dot{\mathbf{\Phi}}_{\mathrm{q}}^{\mathrm{T}}\sigma^*. \tag{30b}$$

Equations (30a–b) constitute a set of $2n$ first order ordinary differential equations. However, $\dot{\mathrm{p}}$ is given in explicit form, and therefore only n algebraic equations need be solved at each function evaluation for the numerical implementation of the algorithm.

Our numerical simulations have shown that equations (30) tend to be numerically stiff due to all the penalty terms concentrated in the RHS of (30b). This numerical stiffness limits the possible choices of numerical integrators. Standard ODE integrators [33] that are based on conditionally stable predictor-corrector multi-step formulae, lead to an increased number of function evaluations. We propose in the next section a modification of (30) that circumvents this problem.

4.2. Modified Augmented Lagrangian Formulation in Canonical Form

The canonical equation (30a) may be also written as

$$\mathrm{p} = \mathrm{M}\dot{\mathrm{q}} + \mathbf{\Phi}_{\mathrm{q}}^{\mathrm{T}}\alpha\dot{\mathbf{\Phi}} + \mathbf{\Phi}_{\mathrm{q}}^{\mathrm{T}}\sigma^*, \tag{31}$$

which indicates that the canonical momenta is stabilized through the addition of penalty terms that are proportional to the violation of the velocity constraint equations. It is important to realize that if equation (31) is differentiated and substituted into the acceleration based augmented Lagrangian equation (24) the result is precisely the additional canonical equation (30b), which lead us to see that *the canonical equations originate from the acceleration based equations by the mere canonical transformation* indicated in (31).

However, we can achieve a better stabilization of the canonical momenta if we add to the RHS of (31) two additional penalty terms: one term proportional to the constraint violation and the other to its integral. Accordingly we define a new momenta $\mathbf{p}$ as

$$\mathbf{p} = \mathbf{M}\dot{\mathbf{q}} + \mathbf{\Phi}_{\mathbf{q}}^{\mathbf{T}}\alpha\left(\dot{\mathbf{\Phi}} + 2\mu\Omega\mathbf{\Phi} + \Omega^2\int_{t_o}^{t}\mathbf{\Phi}d\tau\right) - \mathbf{\Phi}_{\mathbf{q}}^{\mathbf{T}}\sigma^*. \tag{32}$$

By expanding the term $\dot{\mathbf{\Phi}}$ equation (32) becomes:

$$\left(\mathbf{M} + \mathbf{\Phi}_{\mathbf{q}}^{\mathbf{T}}\alpha\mathbf{\Phi}_{\mathbf{q}}\right)\dot{\mathbf{q}} = \mathbf{p} - \mathbf{\Phi}_{\mathbf{q}}^{\mathbf{T}}\alpha\left(\mathbf{\Phi}_t + 2\mu\Omega\mathbf{\Phi} + \Omega^2\int_{t_o}^{t}\mathbf{\Phi}d\tau\right) - \mathbf{\Phi}_{\mathbf{q}}^{\mathbf{T}}\sigma^*. \tag{33a}$$

The differentiation of (33a) and substitution into (24) leads to the second set of modified canonical equations

$$\dot{\mathbf{p}} = \mathbf{Q} + L_{\mathbf{q}} + \dot{\mathbf{\Phi}}_{\mathbf{q}}^{\mathbf{T}}\alpha\left(\dot{\mathbf{\Phi}}2\Omega\mu\mathbf{\Phi} + \Omega^2\int_{t_o}^{t}\mathbf{\Phi}d\tau\right) + \dot{\mathbf{\Phi}}_{\mathbf{q}}^{\mathbf{T}}\sigma^*, \tag{33b}$$

which along with (33a) constitute a set of $2n$ first order ordinary differential equations in the unknowns $\mathbf{p}$, $\mathbf{q}$ and σ^*. Again only n algebraic equations need be solved at each function evaluation for the numerical implementation of the algorithm. A very important point is that, contrary to equations (30a-b), equations (33a-b) do not become stiff, and all our numerical experiments show that they even provide more numerical accuracy and better constraint stabilization than the acceleration based formulation of equation (24).

In fact we can compare this set of equations with the n second order ordinary differential equations resulting from the acceleration based formulation of (24). While both formulations require the triangularization of the same leading matrix for each function evaluation, there is a serious advantages in the use of (33a-b) as compared to (24): the kinematic constraint conditions are differentiated only once with the canonical procedure (twice in the acceleration based formulation) and this will lead to lesser violations of the constraints. We will see in the numerical simulations of Section 6, how this factor becomes detrimental for the acceleration based formulation under repetitive singular positions, whereas the canonical approach leads to a much better performance.

Note again, that the multipliers σ^* do not need to be solved for explicitly. Following the same procedure as that used with the acceleration based augmented Lagrangian formulation, the σ^* may be obtained in an iterative manner as:

$$\sigma_{i+1}^* = \sigma_i^* + \left(\dot{\mathbf{\Phi}} + 2\mu\Omega\mathbf{\Phi} + \Omega^2\int_{t_o}^{t}\mathbf{\Phi}d\tau\right)_{i+1}, \qquad i = 0, 1, 2, \ldots \tag{34}$$

with $\sigma_0^* = 0$ for the first iteration. Equation (33a) including the iterative process of (34) becomes

$$\left(\mathbf{M} + \mathbf{\Phi}_{\mathbf{q}}^{\mathbf{T}}\alpha\mathbf{\Phi}_{\mathbf{q}}\right)\dot{\mathbf{q}}_{i+1} = \mathbf{M}\dot{\mathbf{q}}_i - \mathbf{\Phi}_{\mathbf{q}}^{\mathbf{T}}\alpha\left(\mathbf{\Phi}_t + 2\mu\Omega\mathbf{\Phi} + \Omega^2\int_{t_o}^{t}\mathbf{\Phi}d\tau\right),$$

$$i = 0, 1, 2, \ldots \tag{35}$$

with $\mathbf{M}\,\dot{\mathbf{q}}_o = \mathbf{p}$ for the first iteration. Equation (35) shows that the velocity calculation at each function evaluation is refined so that the weighted summation of the constraint equations (34) are satisfied to machine precision. After the velocity calculation equation (33b) may be used to evaluate the derivative of the canonical momenta.

The algorithm may be presented as

ALGORITHM ALF2

Given $\mathbf{p}$ and $\mathbf{q}$ at time step l.

1. Use (35) iteratively to solve for $\dot{\mathbf{q}}$, with $\mathbf{M}\,\dot{\mathbf{q}}_o = \mathbf{p}$ for the first iteration. At the end of each iteration use (34) to calculate the Lagrange multipliers σ^*
2. Use (33b) to compute $\dot{\mathbf{p}}$ explicitly (no solution of equations involved).
3. Call the numerical integration subroutine to compute $\mathbf{p}$ and $\mathbf{q}$ at time step $l + 1$.
4. Upon convergence of the n.i.s.
   - If desired, use a differentiation scheme to obtain $\lambda = \dot{\sigma}$
   - Update the time variable and go to step 1.

This algorithm is as efficient numerically as ALF1 but much more stable under repetitive singular positions.

5. Canonical Augmented Lagrangian Formulation for Non-Holonomic Systems

The modified augmented Lagrangian formulation described above may also be extended to non-holonomic systems with constraints of the form

$$\boldsymbol{\Phi}\ (\dot{\mathbf{q}}, \mathbf{q}, t) = \mathbf{0}. \tag{36}$$

The acceleration based augmented Lagrangian formulation for this type of constraints is:

$$\mathbf{M}\ddot{\mathbf{q}} = Q + L_{\mathbf{q}} - \dot{\mathbf{M}}\dot{\mathbf{q}} - \boldsymbol{\Phi}_{\dot{\mathbf{q}}}^{\mathbf{T}}\alpha\left(\dot{\boldsymbol{\Phi}} + \beta\boldsymbol{\Phi}\right) - \boldsymbol{\Phi}_{\dot{\mathbf{q}}}{}^{\mathbf{T}}\lambda^*. \tag{37}$$

In order to obtain the canonical counterparts we follow a procedure similar to that used for the holonomic case, and establish the following canonical transformation:

$$\mathbf{p} = \mathbf{M}\dot{\mathbf{q}} + \boldsymbol{\Phi}_{\dot{\mathbf{q}}}^{\mathbf{T}}\alpha\left(\boldsymbol{\Phi} + \beta\int_{t_o}^{t}\boldsymbol{\Phi}d\tau\right) + \boldsymbol{\Phi}_{\dot{\mathbf{q}}}{}^{\mathbf{T}}\sigma^*, \tag{38a}$$

which indicates that a better stabilization of the canonical momenta may be achieved by considering one penalty term proportional to the constraint violation and other to its integral. The differentiation of (38a) and posterior substitution into (37) leads to the second set of canonical equations

$$\dot{\mathbf{p}} = \mathbf{Q} + L_{\mathbf{q}} + \dot{\boldsymbol{\Phi}}_{\dot{\mathbf{q}}}^{\mathbf{T}}\alpha\left(\boldsymbol{\Phi} + \beta\int_{t_o}^{t}\boldsymbol{\Phi}d\tau\right) + \dot{\boldsymbol{\Phi}}_{\dot{\mathbf{q}}}^{\mathbf{T}}\sigma^*, \tag{38b}$$

which along with (38a) constitute a set of $2n$ first order ordinary differential equations in the unknowns $\mathbf{p}$, $\mathbf{q}$ and σ^*. Again only n equations need be solved at each function evaluation.

Typically, non-holonomic constraint conditions for multibody systems take the following form

$$\mathbf{\Phi} = \mathbf{A}(\mathbf{q}, t)\, \dot{\mathbf{q}} + \mathbf{B}(\mathbf{q}, t) \tag{39}$$

and consequently the application of (38a–b) leads to

$$\left(\mathbf{M} + \mathbf{A}^{\mathrm{T}} \alpha \mathbf{A}\right) \dot{\mathbf{q}} = \mathbf{p} - \mathbf{A}^{\mathrm{T}} \alpha \left(\mathbf{B} + \beta \int_{t_o}^{t} \mathbf{\Phi} d\tau\right) - \mathbf{A}^{\mathrm{T}} \sigma^* \tag{40a}$$

$$\dot{\mathbf{p}} = \mathbf{Q} + L_{\mathbf{q}} + \dot{\mathbf{A}}^{\mathrm{T}} \alpha \left(\mathbf{\Phi} + \beta \int_{t_o}^{t} \mathbf{\Phi} d\tau\right) + \dot{\mathbf{A}}^{\mathrm{T}} \sigma^* \tag{40b}$$

and

$$\sigma_{i+1}^* = \sigma_i^* + \left(\mathbf{\Phi} + \beta \int_{t_o}^{t} \mathbf{\Phi} d\tau\right)_{i+1}, \quad i = 0, 1, 2, \ldots \tag{41}$$

with $\sigma_0^* = 0$ for the first iteration.

6. Numerical Examples

6.1. A Simple Example

To better understand the application of the augmented Lagrangian formulation in singular and non-singular positions, let us consider the slider-crank mechanism shown in Figure 1. Both links are of length l = 1m, with a uniformly distributed mass of m = 1 Kg. We take as position coordinates q, the x and y coordinates of the crank end, and the x coordinate of the slider, thus $\mathbf{q}^{\mathrm{T}} = \{x_1,\ y_1,\ x_2\}$. We consider the gravity force, with a value $g = 9.81$ m/s^2 acting in the negative Y axis direction.

The 3 × 3 mass matrix corresponding to these variables is

$$\mathbf{M} = \frac{1}{6} \begin{bmatrix} 4 & 0 & 1 \\ 0 & 4 & 0 \\ 1 & 0 & 2 \end{bmatrix}$$

This mechanism has one degree of freedom only, and therefore there are two geometrical constraints that correspond to the constant distance conditions (the reader is referred to [37] for a general use of the *fully-Cartesian* coordinates)

$$\mathbf{\Phi} = \left\{ \begin{array}{c} \frac{1}{2}\left(x_1^2 + y_1^2 - 1\right) \\ \frac{1}{2}\left[(x_2 - x_1)^2 + y_1^2 - 1\right] \end{array} \right\}$$

When the crank forms an angle of $\pi/2$ radians with the horizontal, the coupler is coincident with the crank and the crank axis is also coincident with the slider. In this position the mechanism has two instantaneous degrees of freedom, since it can undergo either the motion of a slider-crank or the motion of two superimposed rotating bars. Let us now apply the

Table 1. Convergence rate with $\alpha = 10^4$

Iteration #	Error
1	6.5792 10^{-4}
2	4.3705 10^{-8}
3	2.9206 1-^{-12}
4	1.6107 10^{-14}

algorithm ALF1 for the instantaneous solution of the accelerations, for both a nonsingular position and a singular position.

Nonsingular Position. Consider the mechanism in an initial position in which the crank forms an angle of $\pi/4$ with the horizontal and in which the slider has a velocity $\dot{x}_2 = -2$ m/s. The exact acceleration has been computed first with the classical Lagrange's multiplier method of equation (4). Then, the accelerations have been calculated with the algorithm ALF1, using equation (27) iteratively, with a value $\alpha = 10^4$. Table I shows the norm of the difference between the exact acceleration and the one obtained with ALF1.

Table I also shows that the convergence rate of the iterative algorithm is considerably fast. This rate agrees with that predicted analytically in [36]. A higher penalty value gives a faster convergence rate but a lower precision. For instance, a value of $\alpha = 10^7$ yields an error of the order of 10–12 in one iteration, however, further iterations are unable to improve the solution, since some precision is lost in floating point arithmetic operations between numbers with exponents of significantly different values.

Singular Position. Now, consider the crank in a vertical position, forming and angle of $\pi/2$ radians with the horizontal. As we did in the nonsingular case, we take again a slider velocity value $\dot{x}_2 = -2$ m/s. Since the mechanism is in a singular position with 2 instantaneous degrees of freedom, we also have to specify the horizontal velocity of the crank end. It can be easily shown that, theoretically, the crank end can have any velocity value $\dot{x}_1 = v$. However, the slider-crank motion must satisfy the condition $x_1 = x_2/2$ over all its motion, and therefore the velocity $\dot{x}_1 = -1$ seems the obvious choice. Note that in this example the choice for the crank-end velocity is being made explicitly, but during a dynamic simulation the numerical integrator will provide the value of the crank-end velocity. Since the integrator assumes a continuous variation of the variables, this condition will be automatically guaranteed.

In this case, the exact acceleration value cannot be computed with equation (4) because the leading matrix is singular. However, the application of equation (27) with a value of $\alpha = 10^4$ leads to

$$\begin{bmatrix} \frac{2}{3} & 0 & \frac{1}{6} \\ 0 & 2\left(10^4 + \frac{1}{3}\right) & 0 \\ \frac{1}{6} & 0 & \frac{1}{3} \end{bmatrix} \begin{Bmatrix} \ddot{x}_1 \\ \ddot{y}_1 \\ \ddot{x}_2 \end{Bmatrix} = \begin{Bmatrix} 0 \\ -\left(2(10)^4 + 9.81\right) \\ 0 \end{Bmatrix}$$

which can be inverted and leads to the solution $(0,\ -1.000473825436242729,\ 0)$. After 3 iterations, the result is $(0,\ -1,\ 0)$, accurate to 14 digits.

This simple example clearly and simply illustrates that the penalty-augmented Lagrangian formulation works in singular positions, when the classical formulations, such as the Lagrange's multipliers method or the reduction to independent coordinates, fail. Also note that the condition number of the leading matrix increases at the same rate as the penalty parameter.

6.2. Dynamic Simulation of the Slider-Crank Mechanism

Let us consider again the same slider-crank mechanism of Section 6.1, in an initial position such that the crank forms an angle of $\pi/4$ radians with the X axis and that the slider's velocity is $\dot{x}_2 = -4$ m/s.

We perform a dynamic simulation by integrating the equations of motion for a total of 10 seconds, using a conditionally stable variable step and order integrator based on predictor-corrector multistep formulae [33]. We set the error tolerance to 10^{-5} and choose as penalty parameters $\alpha = 10^7$, $\Omega = 10$ and $\mu = 1$. During the simulation, the mechanism goes through the singular position 11 times, following a periodical response.

First, the simulation was carried out with the acceleration based algorithm ALF1. Figure 3 shows the X acceleration of the crank-end over the time period of 10 seconds. Figure 4 shows the value of the Lagrange multiplier λ_1, corresponding to the constant distance constraint condition between the crank axis and the crank end. Finally, Figure 5 shows the time history of the total energy, which should be kept constant, since the system is conservative. A very interesting point can be noted in Figures 3, 4 and 5. The value of the acceleration of the crank-end and λ_1 present spikes around $t = 9.25$ s and, at the same time, the energy presents a sudden discontinuity. The cause of this phenomenon is a small violation of the constraints around the singular position, due to the combination of the errors produced by the numerical integration routine and by the round-off errors produced by augmented Lagrangian procedure. These errors are more critical in the acceleration based algorithm ALF1 because the constraint equations are differentiated twice.

The simulation was repeated, this time using the algorithm ALF2, with the same error tolerance and values for the penalty parameters. This time, the values of λ_1 and the crank-end acceleration, illustrated in Figures 6 and 7, no longer show the spikes resulting from ALF1. In addition, the total energy, shown in Figure 8, does not show the sudden discontinuity that results in Figure 5.

The accumulation of integration errors that lead to small constraint violations in the neighborhood of the singular position is the cause for the sudden peaks and jumps in the constraint forces and accelerations produced by ALF1. These can be removed by tighter error tolerances in the integrator. The better results obtained under the same conditions with ALF2 are due to its better constraint stabilization properties.

6.3. An Assembly of Two Four-Bar Linkages

Figure 9 shows the initial position of a one degree-of-freedom assembly of two four-bar linkages. This mechanism constitutes a particularly critical example, because when it reaches

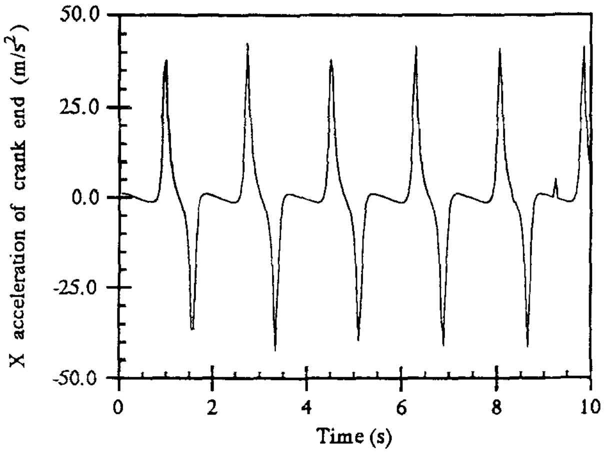

Fig. 3. Acceleration of the crank end with ALF1.

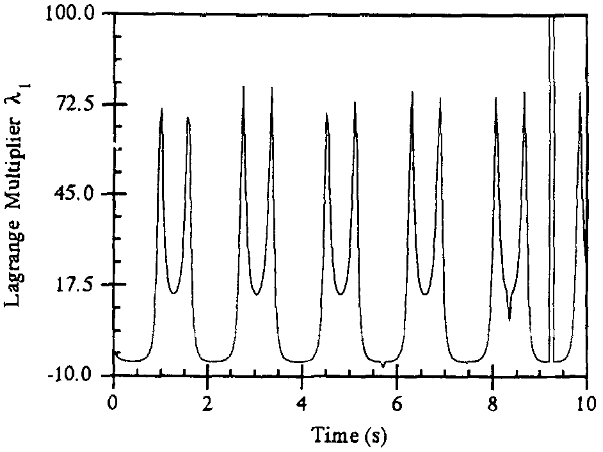

Fig. 4. Lagrange's multiplier with ALF1.

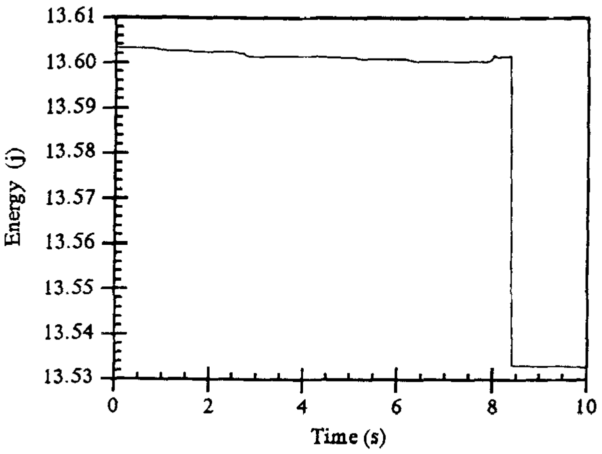

Fig. 5. Total energy with ALF1.

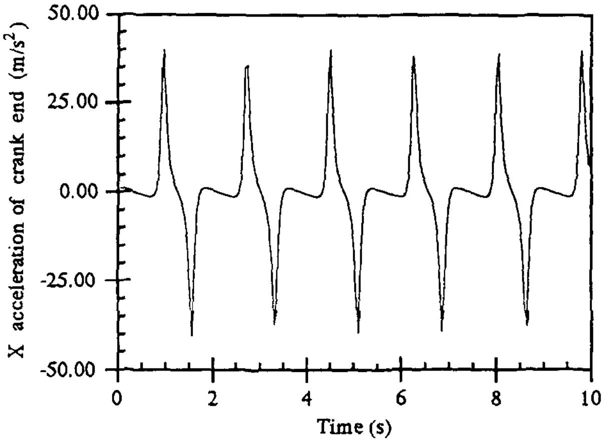

Fig. 6. Acceleration of the crank end with ALF2.

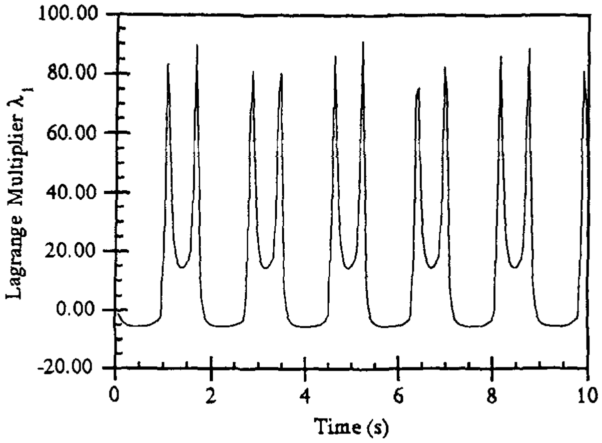

Fig. 7. Lagrange's multiplier with ALF2.

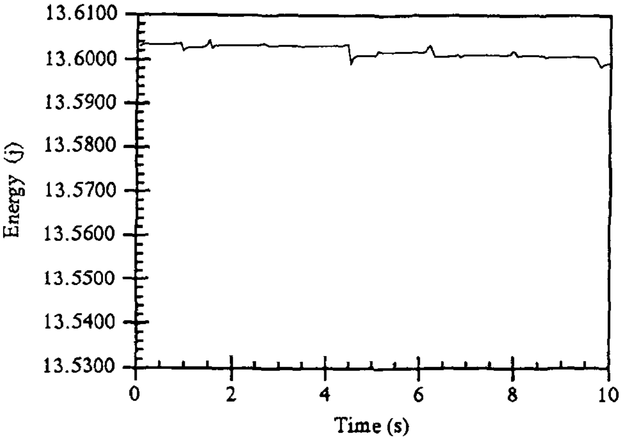

Fig. 8. Total energy with ALF2.

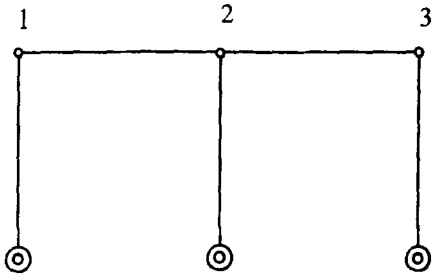

Fig. 9. Assembly of two four-bar linkages.

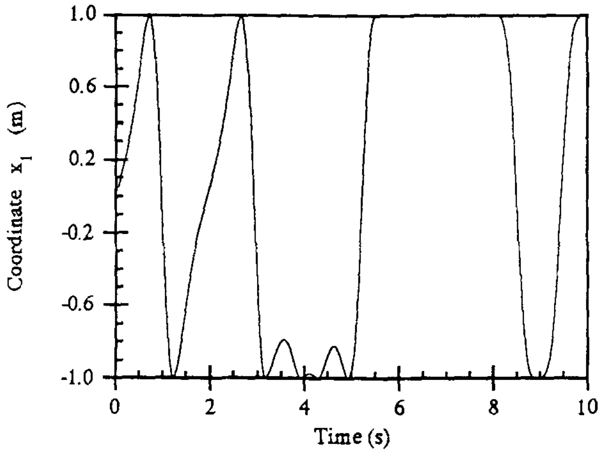

Fig. 10. Time history of x_1 with ALF1.

the horizontal position the number of degrees of freedom increases instantaneously from 1 to 3. To define the position of the system, we use the 6 position variables $(x_1, y_1, x_2, y_2, x_3, y_3)$. All the links are of length $l = 1$ m and have a uniformly distributed mass $m = 1$ Kg. The gravity force acts in the negative Y direction, with a value $g = 9.81$ m/s^2. At $t = 0$ the initial velocity is $\dot{x}_1 = 1$. We integrate the motion for 10 seconds, using the same integrator and tolerance as before, and the values $\alpha = 10^7$, $\Omega = 10$ and $\mu = 1$ for the penalty parameters.

The analysis was carried out twice, first with the algorithm ALF1 and then with the algorithm ALF2. The results obtained with ALF1 are displayed in Figures 10 and 11, which show the time variation of the coordinate x_1 and the Lagrange multiplier λ_1, corresponding to a constant distance constraint between point 1 and the fixed end of the leftmost link. Figures 12 and 13 show the variation of the same variables, obtained this time with the algorithm ALF2. As may be seen, the solution with ALF1 becomes unstable after 3.3 seconds, while ALF2 gives congruent results.

The reason for the failure of ALF1 and the success of ALF2 are found again in the better stability properties of ALF2 with respect to constraint violations (it even yields a successfull integration when just the generic penalty formulation is used with no augmented lagrangian iteration). The way ALF1 may be improved, if it is to be used in repetitive singular positions, is by setting tighter error tolerances and rising the value of the parameter Ω. However, this will introduce numerical stiffness in the problem and therefore will increase the computational

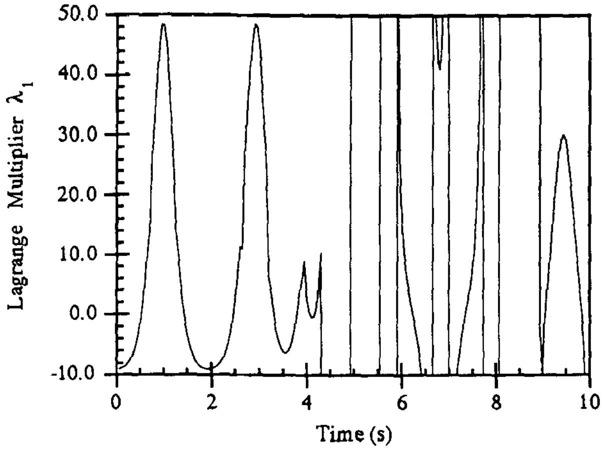

Fig. 11. Lagrange's multiplier with ALF1.

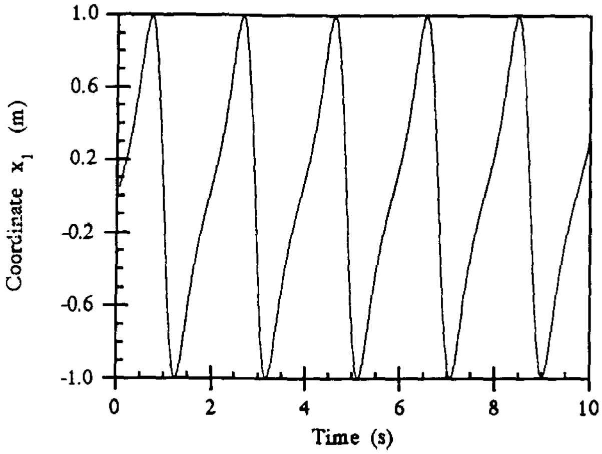

Fig. 12. Time history of x_1 with ALF2.

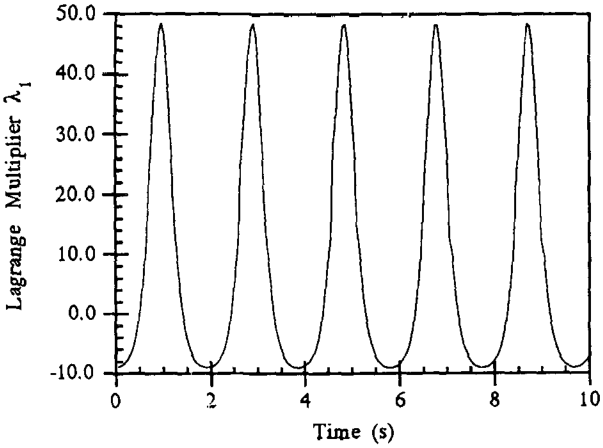

Fig. 13. Lagrange's multiplier with ALF2.

effort. In this example, the value of $\omega = 20$ solves the problem satisfactorily at the cost of a lengthier integration.

7. Conclusions

In this paper we have concisely reviewed the state of the art in multibody dynamic simulation. We have also revisited the acceleration based augmented Lagrangian formulation in the context of singular positions (ALF1) and proposed a new one based on the canonical equations of Hamilton (ALF2) for both holonomic and non-holonomic systems. Both formulations, ALF1 and ALF2, successfully solve the simulation problem in singular positions, however the canonical formulation ALF2 proves to be more accurate and robust than ALF1 under repetitive singular configurations.

The advantages of the proposed method can be summarized as follows:

- The method is very simple to implement and can use standard off the shelf conditionally stable numerical integrators such as those available in commercial mathematical libraries.
- The fact that the leading matrix of the equations of motion is always positive definite, symmetric and sparse, allows for a very efficient solution of the equations without the use of pivoting. This applies even in the presence of redundant (linearly dependent) constraints and coordinates, and most importantly in singular positions.
- Both the generic penalty and augmented Lagrangian methods do not require special provisions such as, detection, elimination of constraints or regularization, near the singular position. The integration goes through the singularity in a procession manner with no need for additional changes.
- The Lagrange multipliers (reaction forces at the constraints) are obtained without having to integrate additional equations.
- The leading matrix is strongly banded, feature that in principle makes it an order n method, where n is the number of dependent coordinates. Therefore it may become a very efficient formulation for those systems with a large number of multibodies, although this assertion needs to be corroborated by further research.
- The acceleration based formulation ALF1 shows numerical inestabilities under repetitive singular positions that are due to the accumulation of round-off and constraint errors. These can be circumvented with tighter tolerances and increased values in the frequency of the dynamical penalty system at the expense of additional computational cost.
- The canonically based method ALF2 is more robust and has not shown pathological behavior in any of our simulations (even when we used it in the generic penalty way). These authors do not know of any other algorithm that can simulate the motion of a multibody undergoing repetitive singular positions as ALF2 does.

As a final remark, one must keep in mind that the actual behavior of multibody systems around singular positions is physically uncertain, due to the uncertainty in the manufacturing tolerances. It is, therefore, unlikely that the behavior of the systems simulated with the algorithms presented in this paper may be experimentally reproduced. However, the usefulness of the algorithms and numerical results presented herein is twofold. Firstly, they provide an efficient and reliable tool for multibody dynamics, which avoids the program crashing that occurs with the classical formulations. Secondly, these algorithms become useful for the study of the different alternative motions that a multibody system may undergo in the neighborhood of a singular position, when one or several geometrical parameters are slightly varied to simulate manufacturing errors.

Acknowledgments

The partial support of this work by the Air Force Office of Scientific Research under contract #F49620-91-C-0095, and the NATO travel grant 0877-87 that has allowed team cooperation between the University of Navarra and the University of California are greatly acknowledged. The authors wish to thank Dr. Chang and Dr. Wu of the AFOSR for their interest in our work, and Prof. Andrew Kurdila for his helpful comments.

References

1. Armstrong, W. W., 'Recursive solution to the equations of motion of an N-link manipulator', *Proceedings 5th World Congress on Theory of Machines and Mechanisms* **2**, 1979, 1343–1346, Montreal, Canada.
2. Featherstone, R., 'The calculation of robot dynamics using articulated body inertias', *The International Journal of Robotic Research* **2**, 1983, 13–30.
3. Featherstone, R., *Robot Dynamics Algorithms*, Kluwer Academic Publishers, Boston, MA, 1987.
4. Walker, M. W. and Orin, D. E., 'Efficient dynamic computer simulation of robotic mechanisms', *Journal of Dynamic Systems, Measurements and Control, ASME* **104**, 1982, 205–211.
5. Bae, D. S. and Haug, E. J., 'A recursive formulation for constrained mechanical system dynamics. Part I: Open loop systems', *Mechanics of Structures and Machines* **15**, 1987, 359–382.
6. Bae, D. S. and Haug, E. J., 'A recursive formulation for constrained mechanical system dynamics. Part II: Closed loop systems', *Mechanics of Structures and Machines* **15**, 1987–88, 481–506.
7. Rodriguez, G., Jain A., and Kreutz K., 'A spatial operator algebra for manipulator modelling and control', *International Journal of Robotics Research* **10**, 1991, 371–381.
8. García de Jalón, J., Unda, J., Avello, A., and Jiménez, J. M., 'Dynamic analysis of three-dimensional mechanisms in natural coordinates', *Computer Methods in Applied Mechanics and Engineering* **56**, 1986, 309–327.
9. Mani, N. K., Haug, E. J., and Atkinson, K. E., 'Application of singular value decomposition for analysis of mechanical system dynamics', *Journal of Mechanisms, Transmissions and Automation in Design* **107**, 1985, 82–87.
10. Kim, S. S. and Vanderploeg, M. J., 'QR decomposition for state space representation of constrained mechanical dynamic systems', *Journal of Mechanisms, Transmissions and Automation in Design* **108**, 1986, 183–188.
11. Liang, C. G. and Lance, G. M., 'A differentiable null space method for constrained dynamic analysis', *Journal of Mechanisms Transmissions and Automation in Design* **109**, 1987, 405–411.
12. Wehage, R. A. and Haug, E. J., 'Generalized coordinate partitioning for dimension reduction in analysis of constrained dynamic systems', *Journal of Mechanical Design* **104**, 1982, 247–255.
13. Ider, S. K. and Amirouche, F. M. L., 'Coordinate reduction in constrained spatial dynamic systems – a new approach', *Journal of Applied Mechanics* **55**, 1988, 899–905.
14. Fuhrer, C. and Schwertassek, R., 'Generation and solution of multibody systems equations', *International Journal Non-Linear Mechanics* **25**, 1990, 127–141.
15. Jerkovsky, W., 'The structure of multibody dynamic equations', *Journal of Guidance and Control* **1**, 1978, 173–182.
16. Kim, S. S. and Vanderploeg, M. J., 'A general and efficient method for dynamic analysis of mechanical systems using velocity transformations', *Journal of Mechanisms, Transmissions and Automation in Design* **108**, 1986, 176–182.
17. Nikravesh, P. E. and Gim, G.,'Systematic construction of the equations of motion for multibody systems containing closed kinematic loops', *Advances in Design Automation* **3**, 1989, 27–33.
18. Kane, T. R. and Levinson, D. A., *Dynamics: Theory and Applications*, McGraw-Hill, New York, NY, 1985.

19. García de Jal6n, J., Jiménez, J. M., Martín, F., and Cuadrado, J., 'Real-time simulation of complex 3-D multibody systems with realistic graphics', *Real Time Integration Methods for Mechanical System Simulation.* Edited by E. Haug and R. Deyo. NATO ASI Series **69**, 1990, 265–293.
20. Brenan, K. E, Campbell, S. L., and Petzold, L. R., *The Numerical Solution of Initial Value Problems in Differential-Algebraic Equations*, Elsevier, New York, NY, 1989.
21. Baumgarte, J., 'Stabilization of constraints and integrals of motion in dynamical systems', *Computer Methods in Applied Mechanics and Engineering* **1**, 1972, 1–16.
22. Baumgarte, J., 'A new method of stabilization for holonomic constraints', *Journal of Applied Mechanics* **50**, 1983, 869–870.
23. Park, T. and Haug, E. J., 'Ill-conditioned equations in kinematics and dynamics of machines', *International Journal of Numerical Methods in Engineering* **26**, 1988, 217–230.
24. Ider, S. K. and Amirouche, F. M. L., 'Numerical stability of the constraints near singular positions in the dynamics of multibody systems, *Computers & Structures* **33**, 1989, 129–137.
25. Amirouche, F. M. L. and Chin-Wei, T., 'Regularization and stability of the constraints in the dynamics of multibody systems', *Nonlinear Dynamics* **1**, 1990, 459–475.
26. Park K. C. and Chiou J. C., 'Stabilization of computational procedures for constrained dynamical systems', *Journal of Guidance, Control and Dynamics* **11**, 1988, 365–370.
27. Park K. C., Chiou J. C., and Downer, J. D., 'Explicit-implicit staggered procedure for multibody dynamic analysis', *Journal of Guidance, Control and Dynamics* **13**, 1990, 562–570.
28. Bayo, E., García de Jal6n, J., and Serna, M. A., 'A modified Lagrangian formulation for the dynamic analysis of constrained mechanical systems', *Computer Methods in Applied Mechanics and Engineering* **71**, 1988, 183–195.
29. Kurdila, A. J., Junkins, J. L., and Hsu, S., 'Lyapunov stable penalty methods for imposing nonholonomic constraints in multibody system dynamics', *Nonlinear Dynamics* **4**, 1993, 51–82.
30. Kurdila, A. J. and Narcowich, F. J.,'Sufficient conditions for penalty formulation methods in analytical dynamics', *Computational Mechanics* **12**, 1993, 81–96
31. Bayo, E., García de Jalón, J., Avello, A., and Cuadrado, J., 'An efficient computational method for real time multibody dynamic simulation in fully Cartesian coordinates', *Computer Methods in Applied Mechanics and Engineering* **92**, 1991, 377–395.
32. Goldstein H., *Classical Mechanics*, Second Edition, Addison-Wesley, London, 1980.
33. Shampine, L. F. and Gordon, M. K., *Computer Solution of Ordinary Differential Equations: the Initial Value Problem*, W. J. Freeman, San Francisco, CA, 1975.
34. Lankarani, H. M and Nikravesh, P. E., 'Application of the canonical equations of motion in problems of constrained multibody systems with intermittent motion', *Advances in Design Automation* **1**, 1988, 417–423.
35. Bae, D.-S. and Won, Y. S., 'A Hamiltonian equation of motion for real time vehicle simulation', *Advances in Design Automation* **2**, 1990, 151–157.
36. Glowinski, R. and LeTallec, P., *Augmented Lagrangian and Operator-Splitting Methods in Nonlinear Mechanics*, SIAM, Philadelphia, PA, 1989.
37. García de Jal6n, J. and Bay6, E., *Kinematic and Dynamic Simulation of Multibody Systems*, Springer-Verlag, New York, 1993