Multibody System Dynamics **10:** 81–105, 2003.

Stabilization Methods for the Integration of DAE in the Presence of Redundant Constraints

MARIA AUGUSTA NETO
FCTUC, Universidade de Coimbra, Coimbra, Portugal; E-mail: augusta.neto@dem.uc.pt

JORGE AMBRÓSIO
IDMEC/IST, Av. Rovisco Pais, P-1049-001 Lisboa, Portugal; E-mail: jorge@dem.ist.utl.pt

(Received: 30 December 2002; accepted in revised form: 24 January 2003)

Abstract. The use of multibody formulations based on Cartesian or natural coordinates lead to sets of differential-algebraic equations that have to be solved. The difficulty in providing compatible initial positions and velocities for a general spatial multibody model and the finite precision of such data result in initial errors that must be corrected during the forward dynamic solution of the system equations of motion. As the position and velocity constraint equations are not explicitly involved in the solution procedure, any integration error leads to the violation of these equations in the long run. Another problem that is very often impossible to avoid is the presence of redundant constraints. Even with no initial redundancy it is possible for some systems to achieve singular configurations in which kinematic constraints become temporarily redundant. In this work several procedures to stabilize the solution of the equations of motion and to handle redundant constraints are revisited. The Baumgarte stabilization, augmented Lagrangian and coordinate partitioning methods are discussed in terms of their efficiency and computational costs. The LU factorization with full pivoting of the Jacobian matrix directs the choice of the set of independent coordinates, required by the coordinate partitioning method. Even when no particular stabilization method is used, a Newton–Raphson iterative procedure is still required in the initial time step to correct the initial positions and velocities, thus requiring the selection of the independent coordinates. However, this initial selection does not guarantee that during the motion of the system other constraints do not become redundant. Two procedures based on the single value decomposition and Gram–Schmidt orthogonalization are revisited for the purpose. The advantages and drawbacks of the different procedures, used separately or in conjunction with each other and their computational costs are finally discussed.

Key words: multibody systems, differential algebraic equations, constraint-vibration, pseudo-inverse.

1. Introduction

In the simulation of multibody systems it is necessary to devise efficient and versatile formulations of the models in order to achieve computational efficiency and accuracy in the solution of any problem. The choice of the coordinates has a direct influence in the structure of the equations of the motion that describe the multibody model and it can be another reason for a method to be more efficient and accurate than another [1]. Cartesian coordinates based formulations lead to sets of differential-algebraic equations, which require proper numerical methods to ensure the stability and accuracy of the solution [2]. The work presented here is developed in the framework of Cartesian coordinates.

The numerical solution of the differential equations of motion is an approximation of the exact solution in which the accuracy depends on the quality of the initial guess for the positions and velocities, ability to recover from the perturbations introduced by the solution process, aptitude to deal with redundant constraints, capability to handle singular positions for the multibody system and eventually the existence of intermittent and unilateral constraints. The set of differential algebraic equations of motion does not use explicitly the position and velocity equations associated to the kinematic constraints. Therefore, small errors in the state variables of the system due to the integration process or to their initial guess by the user cannot be corrected in the course of the solution of the dynamic problem. The strategies generally used to overcome this problem are the coordinate partitioning method [3], the Baumgarte stabilization method [4] or the augmented Lagrangian formulation [5].

In general spatial models it is very often impossible to avoid the use of redundant constraints that lead to Jacobian matrices with linear dependent rows. Consequently, the system of equations formed by the equations of motion and the constraint acceleration equations has a leading matrix which is singular or that becomes ill-conditioned [6]. The application of formulations that use a generalized inverse of the system non-square matrix, which results from the existence of redundant constraints, have been proposed in multibody dynamics in recent years [6–9]. Among these, the Singular Value Decomposition is suitable to solve singular, overconstrained or undetermined problems. Kim and Vanderploeg [8] use QR decomposition for the same purpose. Based on the work by Udwadia and Kalaba [10] a formulation using the K-U formulation has been proposed by Arabyan and Wu [6]. The advantage of this formulation is that the redundant constraints are handled in the solution of the system equations of motion and the problems that involve singular configurations and intermittent constraints, associated with changing the number of degrees-of-freedom, are managed. However, as the proposed methodology does not include the position and velocity constraint equations it does not provide any solution for the constraint violation problem. Therefore, techniques able to minimize or eliminate the constraint violation errors are still required.

This paper presents a discussion on the use of the different methodologies to handle the constraint violation correction or their stabilization and the existence of redundant constraints. Among these methods the use of the K-U formulations for multibody systems with holonomic constrains is emphasized. Different forms of calculating this pseudo-inverse based on Singular Value Decomposition and Gram–Schmidt orthogonalization are presented.

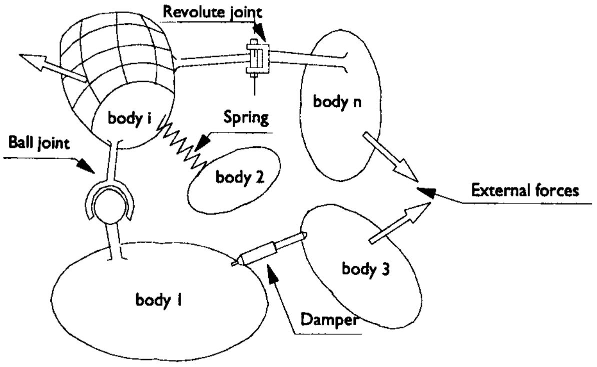

Figure 1. Schematic representation of a multibody system.

2. Dynamic Analysis of Multibody Systems

A multibody system is a collection of rigid and flexible bodies joined together by kinematic joints and force elements as depicted in Figure 1. For the ith body of the system $\mathbf{q}_i$ denotes a vector containing the translational coordinates $\mathbf{r}_i$, and a set of rotational coordinates $\mathbf{p}_i$. A vector of velocities for a rigid body i is defined as $\dot{\mathbf{q}}_i$ and it contains a 3-vector of translational velocities $\dot{\mathbf{r}}_i$ and a 3-vector with the rotational coordinates velocities $\dot{\mathbf{p}}_i$. Another body velocity vector defined as $\dot{\mathbf{q}}_i^*$ includes the angular velocities ω_i instead of the time derivatives of the rotational coordinates. When Euler parameters are used as rotational coordinates the relation between their time derivatives and the angular velocities is given by $2\dot{\mathbf{p}} = \mathbf{L}^T\omega'$, where matrix $\mathbf{L}$ is defined in [1] and ω' represents the angular velocity of the body expressed in the body fixed coordinate frame. The vector of accelerations for the body is denoted by $\ddot{\mathbf{q}}_i$, the time derivative of $\dot{\mathbf{q}}_i$. For a multibody system containing nb bodies, the vectors of coordinates, velocities, and accelerations are $\mathbf{q}$, $\dot{\mathbf{q}}$ and $\ddot{\mathbf{q}}$ that contain the elements of $\mathbf{q}_i$, $\dot{\mathbf{q}}_i$ and $\ddot{\mathbf{q}}_i$, respectively, for $i = 1, \dots, nb$. The system velocity and acceleration vectors, using angular velocities and accelerations instead of the Euler parameters time derivatives are denoted by $\dot{\mathbf{q}}^*$ and $\ddot{\mathbf{q}}^*$.

The constraint equations describing the relative motion between contiguous bodies arise from the kinematic joints. The kinematic constraints are described by mr independent equations as

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

The first and second time derivatives of the position constraint equations yield the kinematic velocity and acceleration equations, respectively

$$\dot{\boldsymbol{\Phi}}(\mathbf{q}, \dot{\mathbf{q}}^*, t) = \mathbf{0} \equiv \boldsymbol{\Phi}_{\mathbf{q}}\dot{\mathbf{q}}^* = \boldsymbol{\nu}, \tag{2}$$

$$\ddot{\boldsymbol{\Phi}}(\mathbf{q}, \dot{\mathbf{q}}^*, \ddot{\mathbf{q}}^*, t) = \mathbf{0} \equiv \boldsymbol{\Phi}_{\mathbf{q}}\ddot{\mathbf{q}}^* = \boldsymbol{\gamma}, \tag{3}$$

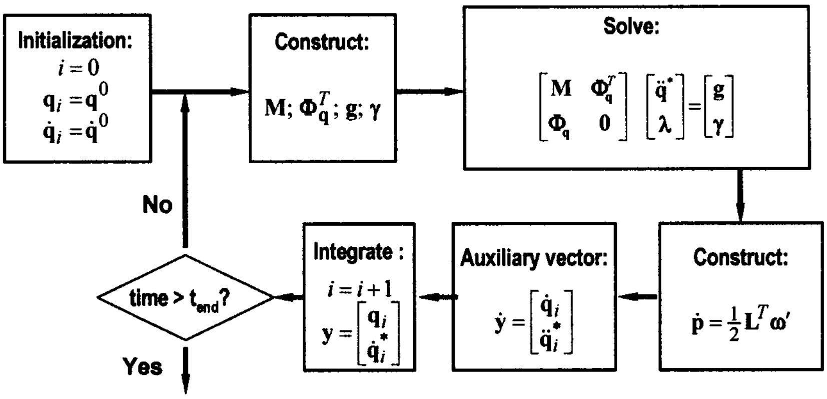
Initialization:
$i=0$
$\mathbf{q}_i=\mathbf{q}^0$
$\dot{\mathbf{q}}_i=\dot{\mathbf{q}}^0$
Construct:
$\mathbf{M};\ \mathbf{\Phi}_{\mathbf{q}}^T;\ \mathbf{g};\ \gamma$
Solve:
$\begin{bmatrix}\mathbf{M} & \mathbf{\Phi}_{\mathbf{q}}^T\\ \mathbf{\Phi}_{\mathbf{q}} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\ddot{\mathbf{q}}^*\\ \boldsymbol{\lambda}\end{bmatrix}=\begin{bmatrix}\mathbf{g}\\ \gamma\end{bmatrix}$
No
time > t_{end}?
Yes
Integrate :
$i=i+1$
$\mathbf{y}=\begin{bmatrix}\mathbf{q}_i\\ \dot{\mathbf{q}}_i\end{bmatrix}$
Auxiliary vector:
$\dot{\mathbf{y}}=\begin{bmatrix}\dot{\mathbf{q}}_i\\ \ddot{\mathbf{q}}_i^*\end{bmatrix}$
Construct:
$\dot{\mathbf{p}}=\frac{1}{2}\mathbf{L}^T\boldsymbol{\omega}'$

Figure 2. Solution procedure for the dynamic analysis of multibody systems.

where $\mathbf{\Phi}_q$ is the Jacobian matrix of the constraints. The equations of motion for the system of rigid bodies are written as [14]

$$\mathbf{M}\ddot{\mathbf{q}}^* + \mathbf{\Phi}_{\mathbf{q}}^T\boldsymbol{\lambda} = \mathbf{g}, \tag{4}$$

where $\mathbf{M}$ is the inertia matrix, $\boldsymbol{\lambda}$ is a vector of unknown Lagrange multipliers, and vector $\mathbf{g} = \mathbf{g}(\mathbf{q}, \dot{\mathbf{q}})$ contains the forces, moments and gyroscopic terms.

Equations (3) and (4) form a system of differential-algebraic equations that must be solved together to obtain the accelerations and Lagrange multipliers. This system is

$$\begin{bmatrix}\mathbf{M} & \mathbf{\Phi}_{\mathbf{q}}^T\\ \mathbf{\Phi}_{\mathbf{q}} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\ddot{\mathbf{q}}^*\\ \boldsymbol{\lambda}\end{bmatrix}=\begin{bmatrix}\mathbf{g}\\ \gamma\end{bmatrix}. \tag{5}$$

The standard numerical solution of these equations proceeds as illustrated in Figure 2. No special provision is made to correct for the position and velocity constraint violations. Moreover, it is assumed that the matrix is not singular or ill-conditioned.

2.1. STANDARD SOLUTION OF THE SYSTEM EQUATIONS OF MOTION

The system of Equations (5) can be solved by applying any method suitable for the solution of linear algebraic equations. The existence of null elements in the main diagonal of the matrix and the possibility of ill-conditioned matrices suggest that methods using partial or full pivoting are preferred. However, none of these formulations help in the presence of redundant constraints. The direct solution of Equation (5) is hereafter designated by the Lagrange Multiplier Method 1 (LM1).

2.2. DIRECT INVERSION OF THE SYSTEM MATRIX

The left-hand side matrix of the system of Equations (5) can be inverted analytically. Equation (4) is rearranged to put the acceleration vector in evidence in the left-hand side and the result is substituted in Equation (3), which is also rearranged to give

$$\boldsymbol{\lambda} = (\boldsymbol{\Phi}_{\mathbf{q}}\mathbf{M}^{-1}\boldsymbol{\Phi}_{\mathbf{q}}^{T})^{-1}\boldsymbol{\Phi}_{\mathbf{q}}\mathbf{M}^{-1}\mathbf{g} - (\boldsymbol{\Phi}_{\mathbf{q}}\mathbf{M}^{-1}\boldsymbol{\Phi}_{\mathbf{q}}^{T})^{-1}\boldsymbol{\gamma}. \tag{6}$$

In these equations it is assumed that the multibody model does not include any body with null mass or inertia so that the inverse of the mass matrix **M** exists. The substitution of Equation (6) in Equation (4) provides the expression for the system accelerations written as

$$\begin{aligned}\ddot{\mathbf{q}}^{*} &= [\mathbf{M}^{-1} - \mathbf{M}^{-1}\boldsymbol{\Phi}_{\mathbf{q}}^{T}(\boldsymbol{\Phi}_{\mathbf{q}}\mathbf{M}^{-1}\boldsymbol{\Phi}_{\mathbf{q}}^{T})^{-1}\boldsymbol{\Phi}_{\mathbf{q}}\mathbf{M}^{-1}]\mathbf{g}\\ &\quad + \mathbf{M}^{-1}\boldsymbol{\Phi}_{\mathbf{q}}^{T}(\boldsymbol{\Phi}_{\mathbf{q}}\mathbf{M}^{-1}\boldsymbol{\Phi}_{\mathbf{q}}^{T})^{-1}\boldsymbol{\gamma}.\end{aligned} \tag{7}$$

Equations (6) and (7) are now rearranged in a compact form and written as

$$\begin{aligned}\begin{bmatrix}\ddot{\mathbf{q}}^{*}\\ \boldsymbol{\lambda}\end{bmatrix} &= \begin{bmatrix}\mathbf{M}^{-1} - \mathbf{M}^{-1}\boldsymbol{\Phi}_{\mathbf{q}}^{T}(\boldsymbol{\Phi}_{\mathbf{q}}\mathbf{M}^{-1}\boldsymbol{\Phi}_{\mathbf{q}}^{T})^{-1}\boldsymbol{\Phi}_{\mathbf{q}}\mathbf{M}^{-1}] & \mathbf{M}^{-1}\boldsymbol{\Phi}_{\mathbf{q}}^{T}(\boldsymbol{\Phi}_{\mathbf{q}}\mathbf{M}^{-1}\boldsymbol{\Phi}_{\mathbf{q}}^{T})^{-1}\\ (\boldsymbol{\Phi}_{\mathbf{q}}\mathbf{M}^{-1}\boldsymbol{\Phi}_{\mathbf{q}}^{T})^{-1}\boldsymbol{\Phi}_{\mathbf{q}}\mathbf{M}^{-1} & -(\boldsymbol{\Phi}_{\mathbf{q}}\mathbf{M}^{-1}\boldsymbol{\Phi}_{\mathbf{q}}^{T})^{-1}\end{bmatrix}\\ &\quad\times\begin{bmatrix}\mathbf{g}\\ \boldsymbol{\gamma}\end{bmatrix}.\end{aligned} \tag{8}$$

The matrix in the right-hand side of Equation (8) is the inverse of the system matrix that appears in Equation (5). The solution process enunciated by Equation (8) is referred to as the Lagrange Multiplier Method 2 (LM2).

3. Solution Methods to Handle Constraints Violations

The initial conditions and the integration of the velocities and accelerations of the multibody system introduce numerical errors in the new positions and velocities obtained. These errors are due to the finite precision of the numerical methodologies and to the position and velocity constraint equations not appearing anywhere in the solution procedure outlined in Figure 2. Therefore, methods able to eliminate errors in the constraint or velocity equations or, at least, to keep such errors under control must be implemented.

3.1. BAUMGARTE STABILIZATION METHOD

Second-order equations, such as Equation (3), are unstable. Small perturbations, such as the numerical errors introduced by the integration process, cannot be corrected naturally and they only tend to be amplified. The solution is to introduce feedback terms that penalize the system response if violations on the position or velocity constraint equations occur [4]. With this purpose in mind, the right-hand side of Equation (3) is modified,

$$\mathbf{\Phi}_{\mathbf{q}}\ddot{\mathbf{q}}^{*} = \boldsymbol{\gamma} - 2\alpha\dot{\mathbf{\Phi}} - \beta^{2}\mathbf{\Phi}, \tag{9}$$

where α and β are positive constants that weight the violations of the velocity and position constraint equations respectively. These constants, for a multibody system made of rigid bodies, are values in the range of 1–10, being $\alpha, \beta = 5$ values often used [1].

The use of the Baumgarte stabilization method is done by using Equation (9) instead of Equation (3) during solution process of the system equations. It should be noted that the method does not correct the constraint violations but simply keeps them under control.

3.2. COORDINATE PARTITIONING METHOD

Based on the original work by Wehage and Haug [3], the coordinate partitioning method eliminates the errors of the velocities and positions that would otherwise accumulate during the integration process, i.e., the method reduce such errors to values below a specified tolerance. This method requires that sets of independent and dependent coordinates are first identified. Then, only the independent accelerations and velocities are integrated and the dependent quantities are calculated using partitions of the velocity and constraint equations.

3.2.1. *Automatic Selection of the Independent Coordinates*

The definition of the dependent and independent coordinates can be done automatically by using a matrix factorization technique, such as Gaussian elimination with full pivoting. For a multibody system with m constrains and n coordinates the Jacobian matrix is $m \times n$ and the order of the columns of the matrix corresponds to the order of elements of vector $\mathbf{q}$. Consider an $m \times n$ matrix $\mathbf{A}$ for which the factorization results in the form

$$\begin{array}{ccc} & \begin{array}{cc} m-k & n-(m-k) \end{array} & \\ \mathbf{A} \to & \left[\begin{array}{cc} \mathbf{B} & \mathbf{R} \\ \mathbf{S} & \mathbf{D} \end{array}\right] & \begin{array}{l} \}m-k \\ \}k \end{array} \end{array} . \tag{10}$$

Assume that there are k redundant rows in the matrix and $m-k$ dependent constraints. As a result of the full pivoting procedure used the k redundant constraints end up in the factorized matrix bottom rows. $\mathbf{B}$ is a non-singular $(m-k) \times (m-k)$ matrix associated to the dependent coordinates, and $\mathbf{R}$ is the submatrix $(m-k) \times (n-m+k)$ associated to the independent coordinates.

In what follows let it be assumed that $\mathbf{A}$ represents the Jacobian matrix $\mathbf{\Phi}_{\mathbf{q}}$. Without loss of generality, we assume that there are no redundant constraints in the multibody model, or that these have been identified and eliminated. In such case the Jacobian matrix can be partitioned into

$$\mathbf{\Phi}_{\mathbf{q}} = [\mathbf{\Phi}_{\mathbf{u}} \;\; \mathbf{\Phi}_{\mathbf{v}}]. \tag{11}$$

The Jacobian matrix in Equation (11) has the same form as the submatrix [**B** **R**] of Equation (10). Equation (11) implies the partition of the coordinate vector into vectors of dependent and independent coordinates, denoted by **u** and **v** respectively. This coordinate partition also implies the partition of the velocity and acceleration vectors given as $\dot{\mathbf{q}} = [\dot{\mathbf{u}}^T \;\; \dot{\mathbf{v}}^T]$ and $\ddot{\mathbf{q}} = [\ddot{\mathbf{u}}^T \;\; \ddot{\mathbf{v}}^T]$ respectively.

3.2.2. *Evaluation of the Dependent Coordinates and Velocities*

Let it be assumed that the vector of system accelerations is calculated by using of Equation (5) or (8) alternatively. The integration vector $\dot{\mathbf{y}}$, appearing in Figure 2, only includes the independent coordinates and it is written as

$$\dot{\mathbf{y}} = [\dot{\mathbf{v}}^T \;\; \ddot{\mathbf{v}}^T]^T, \tag{12}$$

which after integration results in vector $\mathbf{y} = [\mathbf{v}^T \;\; \dot{\mathbf{v}}^T]$.

The dependent velocities and positions are calculated using the velocity and position constraint equations respectively. To evaluate the dependent velocities let Equation (2) be partitioned

$$\mathbf{\Phi}_{\mathbf{u}}\dot{\mathbf{u}} + \mathbf{\Phi}_{\mathbf{v}}\dot{\mathbf{v}} = \boldsymbol{\nu}. \tag{13}$$

The dependent velocities $\dot{\mathbf{u}}$ are evaluated by solving the system of equations

$$\mathbf{\Phi}_{\mathbf{u}}\dot{\mathbf{u}} = -\mathbf{\Phi}_{\mathbf{v}}\dot{\mathbf{v}} + \boldsymbol{\nu}. \tag{14}$$

The dependent positions are obtained through the solution of the position constraint equations, given the independent coordinates, this is,

$$\mathbf{\Phi}(\mathbf{u}, \mathbf{v}, t) = \mathbf{0}. \tag{15}$$

The Newton–Raphson method is used for the solution of Equation (15) to obtain the dependent positions **u**. To achieve convergence some reliable estimates must be provided for these coordinates. A good estimation of $\mathbf{u}^i$ at time t^i, to start the iterative solution procedure, is found by using the information from the previous time t^{i-1} as [1]

$$\mathbf{u}^i = \mathbf{u}^{i-1} + h\dot{\mathbf{u}}^{i-1} + 0.5h^2\ddot{\mathbf{u}}^{i-1}, \tag{16}$$

where h is the integration time step from t^i to t^{i-1}.

3.3. AUGMENTED LAGRANGIAN FORMULATION

The augmented Lagrangian formulation is a methodology that penalizes the constraint violations, much in the same form as the Baumgarte stabilization method [4]. However, this is an iterative procedure that presents a great number of advantages relative to other methods because it involves the solution of a smaller set of equations, it handles redundant constraints and it can still deliver accurate results in the vicinity of singular configurations. The augmented Lagrangian formulation consists in solving the system equations of motion, represented by Equation (5), by an iterative process. Let index i denote the ith iteration. The evaluation of the system accelerations in a given time step starts as

$$\mathbf{M}\ddot{\mathbf{q}}_i^* = \mathbf{g} \quad (i = 0). \tag{17}$$

The iterative process to evaluate the system accelerations proceeds with the evaluation of

$$\bar{\mathbf{M}}\ddot{\mathbf{q}}_{i+1}^* = \bar{\mathbf{g}}, \tag{18}$$

where the generalized mass matrix $\bar{\mathbf{M}}$ and load vector $\bar{\mathbf{g}}$ are given by

$$\begin{aligned} \bar{\mathbf{M}} &= \mathbf{M} + \boldsymbol{\Phi}_{\mathbf{q}}^T \alpha \boldsymbol{\Phi}_{\mathbf{q}}, \\ \bar{\mathbf{g}} &= \mathbf{M}\ddot{\mathbf{q}}_i^* + \boldsymbol{\Phi}_{\mathbf{q}}^T \alpha (\boldsymbol{\gamma}_i - 2\omega\mu\boldsymbol{\Phi}_{\mathbf{q}}\dot{\mathbf{q}}_i^* - \omega^2\boldsymbol{\Phi}_i). \end{aligned} \tag{19}$$

In Equation (19) the mass matrix $\mathbf{M}$, the Jacobian matrix $\boldsymbol{\Phi}_{\mathbf{q}}$ and the right-hand side of the acceleration equations $\boldsymbol{\gamma}$ are the same as those used in Equation (5). The penalty terms α, β and ω ensure that the constraint violations feedback are accounted for during the solution of the system equations. The iterative process continues until

$$\|\ddot{\mathbf{q}}_{i+1}^* - \ddot{\mathbf{q}}_i^*\| \leq \varepsilon. \tag{20}$$

The augmented Lagrangian formulation involves the solution of a system of equations with a dimension equal to the number of coordinates of the multibody system. Though mass matrix $\mathbf{M}$ is generally positive semi-definite the leading matrix of Equation (18) $\mathbf{M} + \boldsymbol{\Phi}_{\mathbf{q}}^T \alpha \boldsymbol{\Phi}_{\mathbf{q}}$ is always positive definite [5]. Even when the system is close to a singular position or when in presence of redundant constraints the system of equations can still be solved.

4. Methods to Handle Redundant Constraints

In many practical situations the multibody systems models include redundant constraints. In some multibody systems it also happens that some constraints are intermittent or that they simply vanish. Several improved formulations, using the Moore–Penrose generalized inverse, that are suitable for these kinds of multibody systems, are revisited here.

4.1. THE MOORE-PENROSE GENERALIZED INVERSE

Let the accelerations of the unconstrained multibody system be $\ddot{\mathbf{q}}^*_{\text{unc}} = \mathbf{M}^{-1}\mathbf{q}$, where it is assumed again that all bodies of the system have non-null masses and inertias. Equation (7) is rewritten as

$$\ddot{\mathbf{q}}^* = \ddot{\mathbf{q}}^*_{\text{unc}} + \mathbf{M}^{-1}\mathbf{\Phi}^T_{\mathbf{q}}(\mathbf{\Phi}_{\mathbf{q}}\mathbf{M}^{-1}\mathbf{\Phi}^T_{\mathbf{q}})^{-1}(\boldsymbol{\gamma} - \mathbf{\Phi}_{\mathbf{q}}\ddot{\mathbf{q}}^*_{\text{unc}}). \tag{21}$$

Let the inverse of the mass matrix be written as $\mathbf{M}^{-1} = \mathbf{M}^{-1/2}\mathbf{M}^{-1/2}$, where $\mathbf{M}^{1/2}$ is a diagonal matrix with coefficients equal to the square root of the corresponding coefficients in the diagonal $\mathbf{M}$ matrix. Equation (21) becomes

$$\ddot{\mathbf{q}}^* = \ddot{\mathbf{q}}^*_{\text{unc}} + \mathbf{M}^{-1/2}(\mathbf{M}^{-1/2}\mathbf{\Phi}^T_{\mathbf{q}})(\mathbf{\Phi}_{\mathbf{q}}\mathbf{M}^{-1/2}\mathbf{M}^{-1/2}\mathbf{\Phi}^T_{\mathbf{q}})^{-1}(\boldsymbol{\gamma} - \mathbf{\Phi}_{\mathbf{q}}\ddot{\mathbf{q}}^*_{\text{unc}}). \tag{22}$$

An auxiliary variable $\mathbf{D} = \mathbf{\Phi}_{\mathbf{q}}\mathbf{M}^{-1/2}$ is defined, which upon substitution in Equation (22) leads to

$$\ddot{\mathbf{q}}^* = \ddot{\mathbf{q}}^*_{\text{unc}} + \mathbf{M}^{-1/2}\mathbf{D}^T(\mathbf{D}\mathbf{D}^T)^{-1}(\boldsymbol{\gamma} - \mathbf{\Phi}_{\mathbf{q}}\ddot{\mathbf{q}}^*_{\text{unc}}). \tag{23}$$

The Moore–Penrose generalized inverse of $\mathbf{D}$, denoted by $\mathbf{D}^+$ has the properties [6]

$$\begin{aligned}&\mathbf{D}\mathbf{D}^+\mathbf{D} = \mathbf{D},\\ &\mathbf{D}^+\mathbf{D}\mathbf{D}^+ = \mathbf{D}^+,\end{aligned} \tag{24}$$

$\mathbf{D}^+\mathbf{D}$ and $\mathbf{D}\mathbf{D}^+$ both being symmetric matrices. Consequently

$$\mathbf{D}^T(\mathbf{D}\mathbf{D}^T)^{-1} = \mathbf{D}^T(\mathbf{D}^+)^T\mathbf{D}^+ = (\mathbf{D}^+\mathbf{D})^T\mathbf{D}^+ = \mathbf{D}^+\mathbf{D}\mathbf{D}^+ = \mathbf{D}^+. \tag{25}$$

The result expressed by Equation (25) is substituted in Equation (23) leading to

$$\ddot{\mathbf{q}}^* = \ddot{\mathbf{q}}^*_{\text{unc}} + \mathbf{M}^{-1/2}\mathbf{D}^+(\boldsymbol{\gamma} - \mathbf{\Phi}_{\mathbf{q}}\ddot{\mathbf{q}}^*_{\text{unc}}). \tag{26}$$

The solution of Equation (26), hereafter designated by MPI, always exists because the Moore–Penrose pseudo-inverse exists even when the inverse of the leading matrix of Equation (5) does not exist. This means that in the presence of redundant constraints or in the presence of constraints that vanish instantaneously, such as the unilateral constraints, the pseudo-inverse matrix $\mathbf{D}^+$ still exists.

4.2. COMPUTATION OF THE GENERALIZED INVERSE

4.2.1. *Singular Value Decomposition*

The calculation of the system accelerations using Equation (26) assumes that the pseudo-inverse matrix $\mathbf{D}^+$ can be calculated. Let $\mathbf{D}$ be an $m \times n$ non-square matrix. Its Singular Value Decomposition (SDV) leads to

$$\mathbf{D} = \mathbf{U}\mathbf{S}\mathbf{V}^T, \tag{27}$$

where $\mathbf{U}$ and $\mathbf{V}$ are $m \times n$ and $n \times n$ square orthogonal matrices respectively. The non-square matrix $\mathbf{S}$ has non-zero elements only on its diagonal and, therefore, the calculation of its pseudo-inverse $\mathbf{S}^+$ is trivial [11]. The pseudo-inverse of $\mathbf{D}$ is

$$\mathbf{D}^+ = \mathbf{V}\mathbf{S}^+\mathbf{U}^T, \tag{28}$$

where the relations $\mathbf{U}^T = \mathbf{U}^{-1}$ and $\mathbf{V}^T = \mathbf{V}^{-1}$, valid for orthogonal matrices, have been used.

4.2.2. *Gram–Schmidt Orthogonalization*

The Gram–Schmidt orthogonalization process (GS) can also be used to compute the pseudo-inverse matrix $\mathbf{D}^+$. With this method, matrix $\mathbf{D}$ is decomposed in [11]

$$\mathbf{D} = \mathbf{Q}\mathbf{R}, \tag{29}$$

where $\mathbf{Q}$ is an $m \times n$ matrix whose columns are orthogonal to each other, i.e., $\mathbf{Q}^T\mathbf{Q} = \mathbf{I}$, and $\mathbf{R}$ is an $n \times n$ upper triangular matrix. The computation of the pseudo-inverse is then obtained as

$$\mathbf{D}^+ = \mathbf{R}^{-1}\mathbf{Q}^T. \tag{30}$$

The use of the Gram–Schmidt orthogonalization requires that all columns of matrix $\mathbf{D}$ are independent.

5. Application Examples

The virtues and shortcomings of the different methods proposed are based on applications to several simple mechanical systems for both kinematic and dynamic analysis. All models are simulated until an error condition happens, so that the simulation has to be stopped, or until the simulation time reaches 20 s.

5.1. KINEMATIC ANALYSIS OF MULTIBODY SYSTEMS

Typically the solution of a kinematic analysis consists in first solving Equation (1) using the Newton–Raphson method and after solving the linear systems of Equations (2) and (3). Alternatively, it is possible to carry the kinematic analysis by using the dynamic equilibrium equations provided that all degrees of freedom of the multibody system are driven by kinematic constraints. In this case the resulting motion of the system components is independent of the rigid body inertial properties and of the external applied forces. Therefore, in the applications that follow the system mass matrix is assumed to be the identity matrix and the forces in the system are assumed to be non-existing. All methodologies presented throughout this work are used to solve each of the different applications. The methods that use the generalized inverse, in particular, require the solution of Equation (26). Based on the assumptions described for the kinematic analysis, Equation (26) reduces to

$$\ddot{\mathbf{q}} = \mathbf{D}^+\boldsymbol{\gamma}. \tag{31}$$

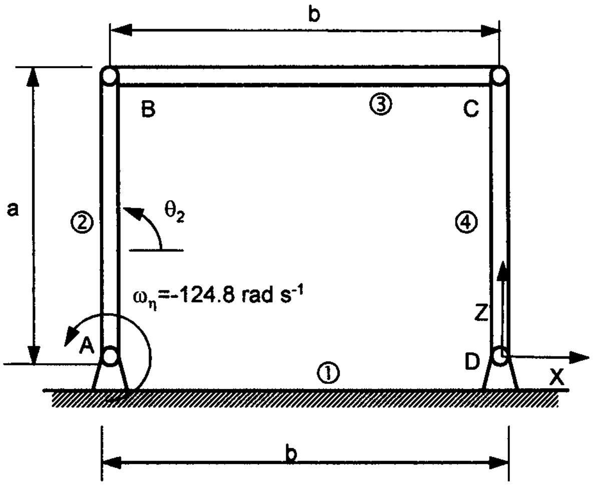

Figure 3. Parallel four-bar linkage.

The vector of accelerations is obtained by solving Equations (5), (7), (17–19) or (31), that correspond to the procedures designated by LM1, LM2, ALF and MPI respectively. The acceleration vector is integrated together with the velocity vector to obtain new velocities and positions, using the sequence depicted by Figure 2. This procedure, by itself, does not ensure that the position and velocity constraint equations are fulfilled. Therefore, the use of a methodology to stabilize or eliminate the constraint violations is still required.

5.1.1. *Four-Bar Linkage*

Consider a four-bar linkage, shown in Figure 3, with $a = b = 0.5$ m. Let the bar 2 have a constant angular velocity, $\omega_\eta = 124.8$ rad/s. Though the motion of the four-bar linkage is planar, the system model includes two revolute joints, in points A and D, and two universal joints, in points B and C, that are defined in a three-dimensional space. The crank is constrained to rotate with a prescribed angular velocity.

The model is described by four rigid bodies, which account for 24 coordinates, and by two revolute and two universal joints, one ground body and one driving constrains, which account for 25 constraints. The system has one degree-of-freedom, which is guided by the driving constraint. Therefore, one of the system constraints is redundant. The existence of this redundant constraint can be eliminated by substituting of one of the universal joints by a spherical joint. However, that is not done here in order to use the example to demonstrate the results of the proposed methodologies. In different analysis several initial positions for the initial orientation of body 2 are considered alternatively, i.e., $\theta_2 = n\pi$; $\theta_2 = \pi/4$ and $\theta_2 = \pi/2$. The system is simulated using the methods described in the previous sections. For the MPI method, the pseudo-inverse is calculated by using the Sin-

Table I. Comparison of the outcome of the use of different methods to simulate the four-bar linkage.

Methods	Initial position		
	$\theta_2 = n\pi$	$\theta_2 = \pi/4$	$\theta_2 = \pi/2$
LM2	Fail ($t = 0$ s)[a]	Fail ($t = 0$ s)[a]	Fail ($t = 0$ s)[a]
LM2 + Baumg. stab.	Fail ($t = 0$ s)[a]	Fail ($t = 0$ s)[a]	Fail ($t = 0$ s)[a]
LM2 + Coord. part.	Fail ($t = 0$ s)[a]	Fail ($t = 0$ s)[a]	Fail ($t = 0$ s)[a]
ALF	Fail ($t = 0$ s)[b]	Fail ($t = 0.006$ s)[b]	Fail ($t = 0.012$ s)[b]
ALF + Coord. part.	Fail ($t = 0$ s)[b]	Fail ($t = 0.006$ s)[b]	Fail ($t = 0.012$ s)[b]
MPI (SDV)	Fail ($t = 0.054$ s)[c]	Fail ($t = 0.031$ s)c	Fail ($t = 0.037$ s)[c]
MPI (SDV)+ Baumg. stab.	Fail ($t = 0.052$ s)[c]	Fail ($t = 0.031$ s)[c]	Fail ($t = 0.037$ s)[c]
MPI (SDV)+ Coord. part.	Convergence	Convergence	Convergence
MPI (GS)	Fail ($t = 0.050$ s)[c]	Fail ($t = 0.031$ s)[c]	Fail ($t = 0.037$ s)[c]
MPI (GS)+ Baumg. stab.	Fail ($t = 0.050$ s)[c]	Fail ($t = 0.031$ s)[c]	Fail ($t = 0.063$ s)[c]
MPI (GS) + Coord. part.	Convergence	Convergence	Convergence

[a] Matrix is singular due to the redundant constraints.
[b] The method does not converge because a singular position is reached.
[c] The constraint violations exceed the allowable tolerance.

gular Value Decomposition (SDV) or the Gram–Schmidt orthogonalization (GS). The Baumgarte stabilization or the coordinate partitioning methods are also used in order to maintain the constraint violations under control. The comparison of the results using combinations of the different methodologies is presented in Table I.

The results show that the existence of redundant constraints prevents the LM2 method to deliver any results, as expected. The ALF method handles the constraint redundancy and it is able to provide results even close to the singular positions. However, when the mechanism singular positions are actually reached the ALF method fails. This failure of the MPI methods is attributed to the small constraint violations that built up during the simulation, which are sufficient to prevent the method to converge near the singular positions due to the mechanism locking. The configurations of the four-bar linkage, for the cases that correspond to the initial crank position of $\theta_2 = \pi/2$, are presented in Figure 4 from the simulation start until its failure time. All sequences in Figure 4 end with the position of the mechanism in which the methodology used failed.

The MPI methods lead to the correct solution of the system accelerations but the use of coordinate partitioning is required in order to ensure that the system position and velocity constraint equations are always fulfilled. This suggests that the constraint violations are in fact the responsible for the failure of the methodologies in the vicinity of the singular positions. Figure 5 shows the maximum constraint violations for MPI method, using the GS and SDV techniques, without the application of the coordinate partitioning method. In Figure 6 the same maximum constraint violations are presented when the MPI method is used in conjunction with the coordinate partitioning strategy. In any of the cases, when the mechanism

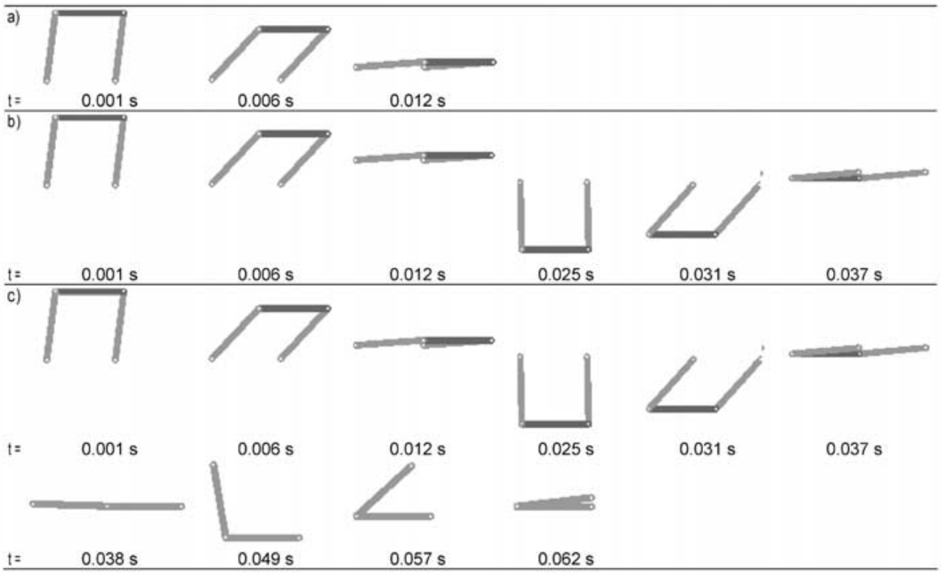
a)
t= 0.001 s 0.006 s 0.012 s
b)
t= 0.001 s 0.006 s 0.012 s 0.025 s 0.031 s 0.037 s
c)
t= 0.001 s 0.006 s 0.012 s 0.025 s 0.031 s 0.037 s
t= 0.038 s 0.049 s 0.057 s 0.062 s

Figure 4. Instantaneous position of the four-bar linkage using different methods: (a) ALF; (b) MPI (GS) and MPI (SDV); (c) MPI (GS) + B.

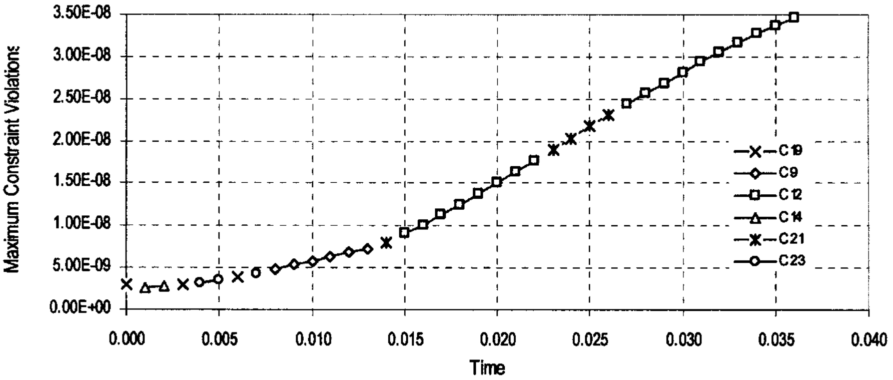

Figure 5. Maximum constraint violations observed when using the GS or the SDV methods to evaluate the leading matrix pseudo-inverse. The coordinate partitioning is not used.

passes by a singular position there is no preferred branch of the motion for the system to take.

Table I shows that unless the coordinate partitioning method is used, all simulations eventually fail due to excessive constraint violations, even when a constraint stabilization methodology is used. In all cases solved it is observed that the use of different penalization parameters in the Baumgarte stabilization method or in the Augmented Lagrangian Method can, at the most, delay the moment for which the constraint violations exceed the allowable tolerance or become large enough to lead to the mechanism lockup, but they never avoid the problem.

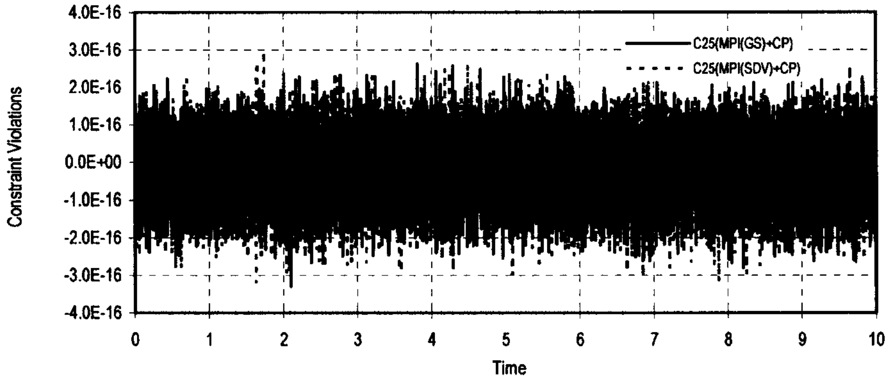

Figure 6. Maximum constraint violation observed (constraint 25) when the MPI methods are used together with coordinate partitioning.

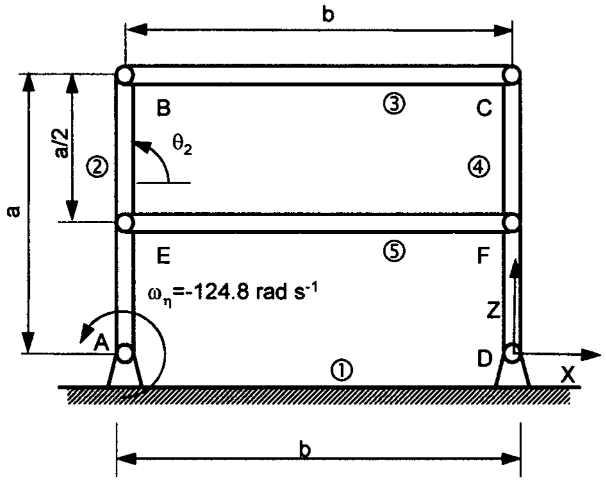

Figure 7. Parallel five-bar linkage.

5.1.2. *Parallel Five-Bar Linkage*

Consider the five-bar linkage presented in Figure 7. This mechanism has the kinematic structure proposed for the four-bar linkage presented in Figure 3 plus an extra rigid body, which is pinned to bodies 2 and 4 by two revolute joints. In this case the system continues to exhibit only one degree-of-freedom but its model uses six more coordinates, corresponding to the extra body, and ten more constraints, due to the new revolute joints. The model for the mechanism has five redundant constraints that need to be dealt with during the solution of its kinematic equations.

The results of the simulation of the five-bar linkage with the combinations of the different methodologies are presented in Table II. Sequences of the configurations observed for mechanism, simulated using the different procedures proposed, are presented in Figure 8 from their initial positions until the solution process fails.

Table II. Comparison of the outcome of the different methods used to simulate the five-bar linkage.

Methods	
LM2	Fail ($t = 0$ s)[a]
LM2 + Baumgarte stabilization	Fail ($t = 0$ s)[a]
LM2 + Coordinate partition	Fail ($t = 0$ s)[a]
ALF	Fail ($t = 0.012$ s)[b]
ALF + Coordinate partition	Fail ($t = 0.012$ s)[b]
MPI (SDV)	Fail ($t = 0.095$ s)[c]
MPI (SDV) + Baumgarte stabilization	Fail ($t = 0.095$ s)[c]
MPI (SDV) + Coordinate partition	Fail ($t = 3.538$ s)[d]
MPI (GS)	Fail ($t = 0.059$ s)[c]
MPI (GS) + Baumgarte stabilization	Fail ($t = 0.059$ s)[c]
MPI (GS) + Coordinate partition	Convergence

[a] Matrix is singular due to the redundant constraints.
[b] The method does not converge because a singular position is reached.
[c] The constraint violations exceed the allowable tolerance.
[d] The integration process fails to converge.

As expected, the simulation methodology that uses the LM2 formulation is unable to handle the redundant constraints and, therefore, it fails to start the solution procedure. The remaining methodologies behave similarly for the simulations of the five and four-bar linkages. The drifts of the system constraints with larger violations that are observed during the simulations are similar to those presented for the four-bar linkage in Figure 5, if the coordinate partitioning method is not used, or in Figure 6 if that procedure is applied. Though the nature of the problem that causes the different methodologies to succeed or fail in the solution of the system equations is the same for simple or complex systems the results suggest that, at this level of complexity of the multibody models, the number of redundant constraints present in the models does not pose extra numerical difficulties.

A close look at the configurations of the five-bar linkage, displayed in Figure 8, shows that the MPI methods are still able to obtain a solution for the system equations even when the constraint violations are large. In fact, the simulation process is halted because the constraint violations exceed the allowable tolerances and not because the SDV or the GS methodologies fail to deliver a solution for the system equations. When the coordinate partitioning method is used together with the MPI formulations a feasible solution for the complete motion of the system is obtained. The failure of the MPI method that uses the SDV with coordinate partitioning for the singular position of the mechanism that occurs for 3.538 s is due to the difficulty of the integration process to converge.

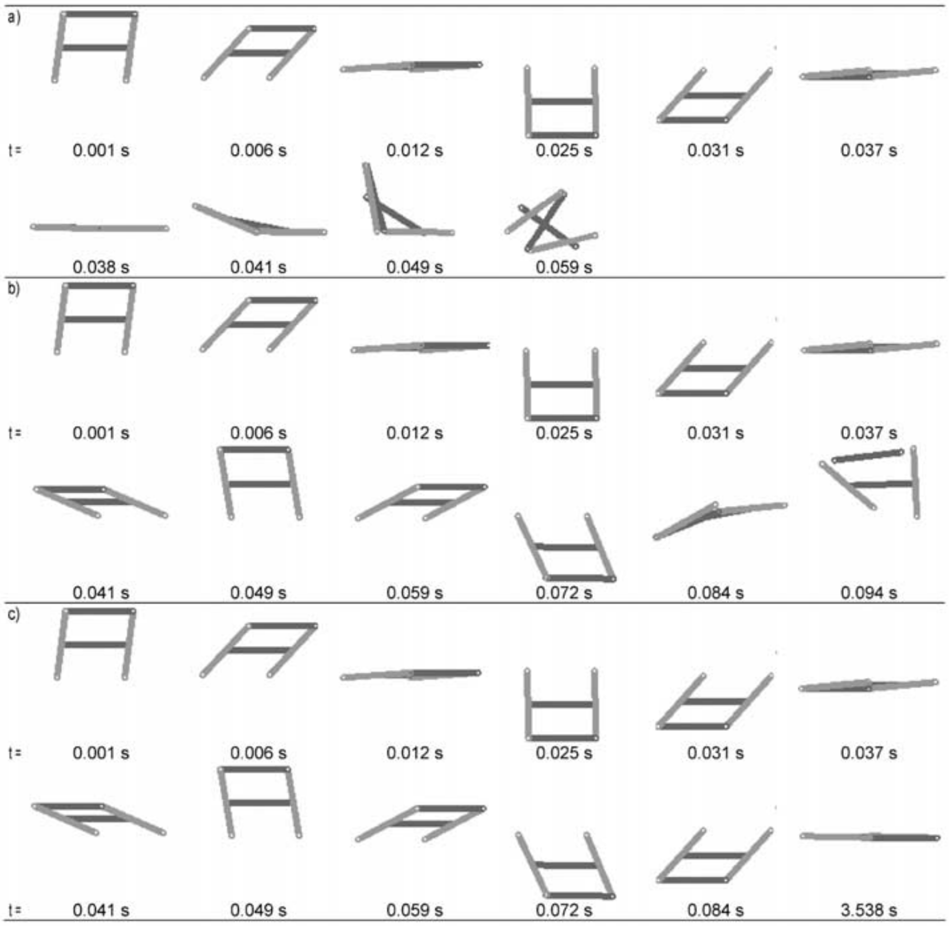

Figure 8. Instantaneous position of the five-bar linkage: (a) MPI (GS); (b) MPI (SDV); (c) MPI (SDV) + CP.

5.1.3. *Spatial Slider Crank*

The application of the spatial kinematic analysis to a simple spatial multibody system is presented for the study of the offset slider crank shown in Figure 9. The crank rotates about Y with a constant angular velocity $\omega = 124.8$ rad/s. The slider translates in a direction parallel to Y with an offset d with respect to this axel. The kinematic joints of the multibody system include one revolute joint between the ground and the crank, a universal joint between the crank and the rod, a spherical joint between the rod and the slider and a translation joint between the slider and the ground. The mechanism is modeled with 24 coordinates, which result from four rigid bodies, and 23 constraint equations, due to the kinematic constraints. Because the system has only one degree-of-freedom the model used to represent it has no

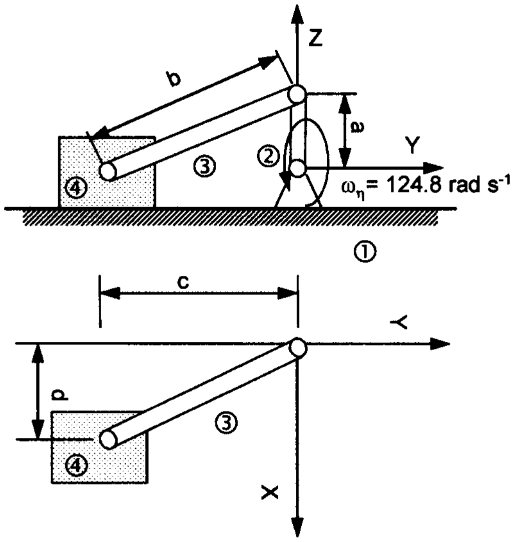

Figure 9. Slider-crank 3D.

redundant constraints. The kinematic analysis is carried on with the methodologies proposed and their outcome is resumed in Table III.

The results presented in Table III show that if the coordinate partitioning method is not used all methodologies fail after some time. The use of the Baumgarte stabilization method in conjunction with the different procedures applied in the solution of the system equations simply delays the time at which the constraint violations become too large. The ALF is able to control the drift of the kinematic constraints violations allowing for the normal termination of the simulation process. When the coordinate partitioning method is used, together with any of the proposed procedures to solve the system equations, the analysis is successfully carried on for long simulation times.

The use of the coordinate partitioning method is known to require a larger number of computer operations than the constraint stabilization procedures. The results presented in Table III show that the use of the coordinate partitioning method leads to solutions of the system motion that is much smoother that those obtained when using the Baumgarte stabilization method or no constraint stabilization at all. The integration of the system velocities and accelerations requires the use of variable time step integration algorithms that actually adjust it to the frequency content of the system response. By eliminating the constraint violations the system response becomes smoother and the step size of the integration becomes about 30 times larger than that selected when the coordinate partitioning is not used. The result is that the methods that use coordinate partitioning are computationally cheaper that those not using it. Though the ALF requires about 3,300 time steps to be taken by the integration procedure for each second of simulation, like any other meth-

Table III. Comparison of the outcome of the spatial slider crank simulation for different methods.

Methods	Results	Num. of steps per sec. of simulation	Execution time per sec. of simulation
LM2	Fail (t = 6.163 s)[a]	3,303	53.95
LM2 + Baumgarte stabilization	Fail (t = 8.079 s)[a]	3,292	52.53
LM2 + Coordinate partition	Convergence	103	36.84
ALF	Convergence	3,286	36.84
ALF + Coordinate partition	Convergence	103	48.69
MPI (SDV)	Fail (t = 6.163 s)[a]	3,303	79.64
MPI (SDV) + Baumgarte stabilization	Fail (t = 8.079 s)[a]	3,292	79.19
MPI (SDV) + Coordinate partition	Convergence	103	42.42
MPI (GS)	Fail (t = 6.163 s)[a]	3,303	92.87
MPI (GS) + Baumgarte stabilization	Fail (t = 8.079 s)[a]	3,292	93.92
MPI (GS) + Coordinate partition	Convergence	103	42.14

[a] The constraint violations exceed the allowable tolerance.

odology that does not use the coordinate partitioning method, its computational cost is smaller. The methods that use the MPI are computationally more expensive than those that use the LM2 or the ALF procedures. The costs of evaluating the pseudo-inverse by using the GS or the SVD are not different.

5.2. DYNAMIC ANALYSIS OF MULTIBODY SYSTEMS

Multibody models of spatial and planar slider cranks described are used here in the framework of dynamic analysis of multibody systems. Instead of using driving constraints to guide the mechanisms cranks a constant moment of 100 Nm is applied to the crank body. Both multibody models are simulated until the methodologies fail or 20 s of simulation are reached.

5.2.1. *Slider Crank*

A model of the slider crank mechanism, shown in Figure 10, is analyzed here. This model has crank and connecting rod bodies with equal sizes of 0.308 m, and consequently it exhibits a singular configuration when the joint between the ground and the crank coincides with the joint between the connecting rod and the slider. The model with four rigid bodies is described by 24 coordinates. Its 24 kinematic constraints result from the application of two revolute joints, one spherical joint, one translation joint and one ground body. The slider crank mechanism has a single degree-of-freedom and consequently one of the kinematic constraints is redundant.

Due to the existence of the redundant kinematic constraint the leading matrices in the system equations of motion are singular and, therefore, the LM1 and LM2

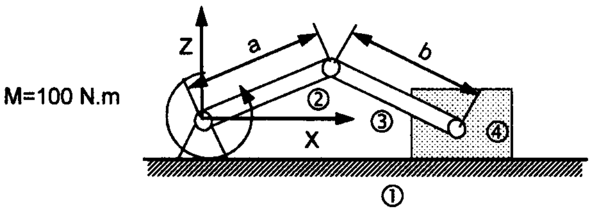

Figure 10. Planar slider-crank.

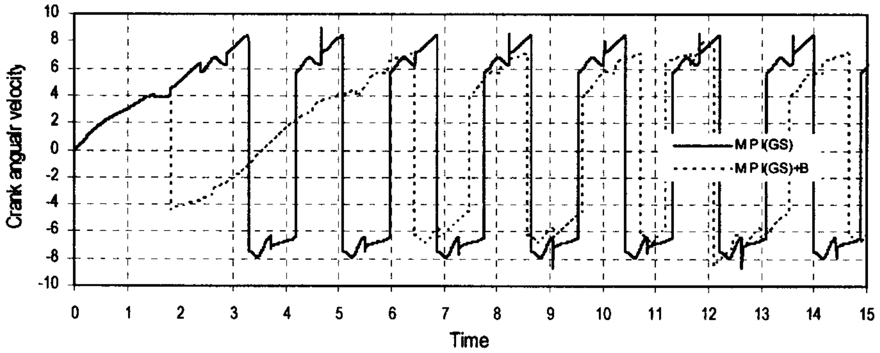

Figure 11. Crank angular velocity for the planar slider crank mechanism.

methods cannot be used. The simulation of the motion of the slider-crank is successfully performed for 20 s. with the methodologies ALF, MPI (GS) and MPI (SDV), using or not the coordinate partitioning or the Baumgarte stabilization methods in conjunction. However, when the mechanism goes over a singular point there is no preferred branch of motion to be followed. Figure 11 shows the angular velocity of the crank obtained in the simulation with two different methodologies. It is observed that after the first singular point, reached at 1.81 s, the solution of the system obtained with the MPI (GS) method alone follows a different path relative to the one obtained when using the same procedure together with the Baumgarte stabilization method.

The sequence of the configurations of the slider crank when using the MPI (GS) and MPI (GS) + B methodologies are presented in Figure 12. When the first singular position is reached the coupler and crank rotate together as if they were a single body, but the direction of their rotations is opposite for the MPI (GS) and the MPI (GS) + B. Though not pictured in any of the figures, when using the MPI (SDV) methodology with and without coordinate partitioning this type of branching is also observed. Moreover, just the modification of the penalization constants of the Baumgarte method is enough to vary the branching of the system motion after a singular position.

The behavior of the mechanism accelerations close to the singular positions is characterized by sudden high values, emphasized in Figure 13, where the logarithm of the crank acceleration is shown. Clearly, these peaks in the acceleration are

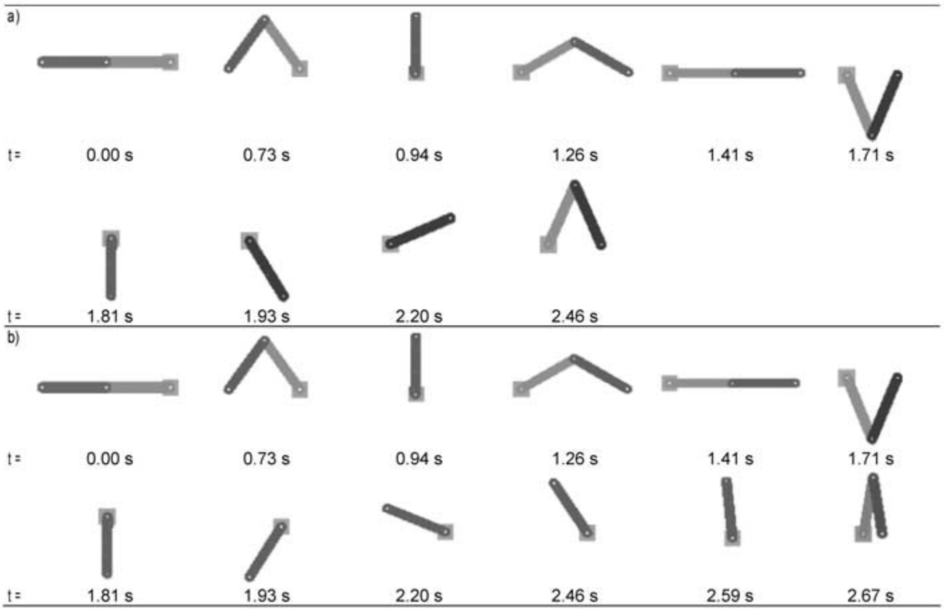

Figure 12. Instantaneous position of the slider crank: (a) MPI (GS); (b) MPI (GS) with Baumgarte stabilization.

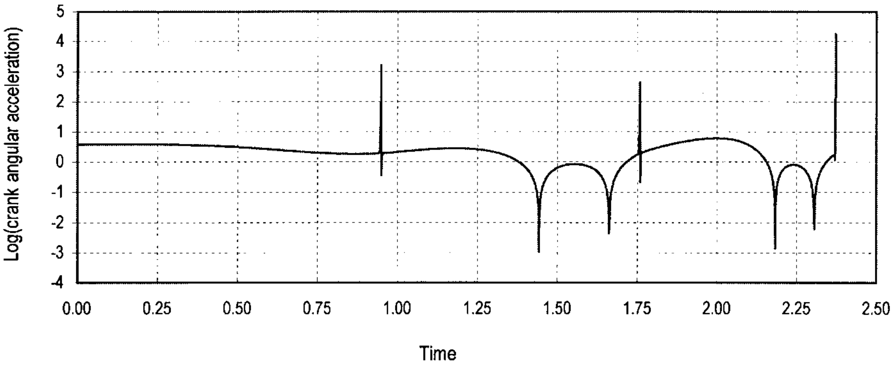

Figure 13. Crank angular acceleration when the MPI (GS) method is used together with coordinate partitioning.

associated to sudden forces that develop in the vicinity of the singular positions of the mechanism, which can eventually lead to halting the numerical integration process. This effect is even more evident when the coordinate partitioning method is used. Therefore, a possible failure of the simulation procedure must be attributed to difficulties in the integration process rather than in the lack of ability of numerical methodologies suggested throughout this work to deliver solutions for the system equations of motion.

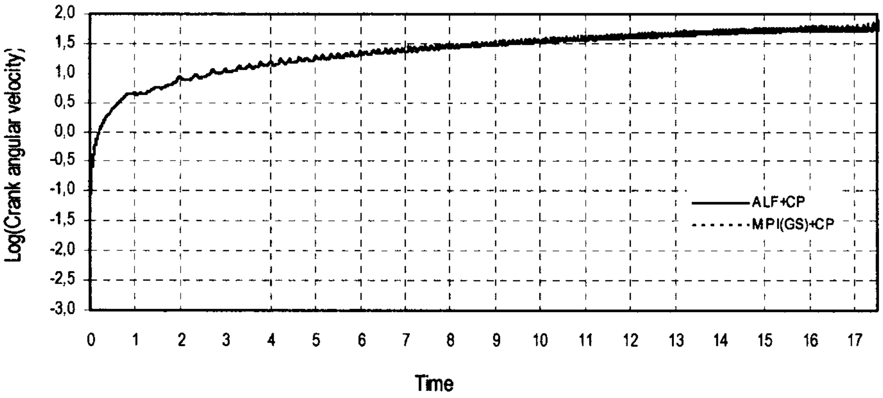

Figure 14. Crank angular velocity for methodologies that use coordinate partitioning.

When the dimensions of the crank and connecting rod are different, the system does not exhibit any singular positions. Another model, with the dimensions of the crank and connecting rod of a = 0.304 m and b = 0.4048 m respectively, is also studied. Figure 14 illustrates the time history of the crank angular velocity obtained in the simulations with the different methodologies and coordinate partitioning. Though not displayed, the same results are obtained when the methodologies described are used without correcting for the constraint violations. Therefore, all formulations lead to the same result and no serious problems of constraint violations are observed during the simulations.

5.2.2. *Spatial Slider Crank*

The dynamics of the spatial slider crank depicted in Figure 9 is simulated here using the different methodologies implemented. The system is modeled with no redundant constraints and there are no singular positions. Therefore, all methodologies are expected to deliver some results, at least during the initial phases of the analysis. In fact, all formulations provide similar results, as exemplified by the outcome of the crank angular velocity, depicted in Figure 15, for the methodologies ALF, MPI (SDV) and MPI (GS) and in Figure 16, where the same methodologies are used in conjunction with the coordinate partitioning method.

The number of time steps taken in the integration process and the computing time spent for the simulation of the spatial slider crank per second, using the different procedures to solve for the system equations of motion, are presented in Table IV. The use of the coordinate partitioning method leads to the selection of 30% larger time steps in the integration procedure. This computational savings associated to these larger time steps compensates partially for the overhead in computational costs that result from the extra computations required by the coordinate partitioning method. In any case, the ALF procedure is still more efficient

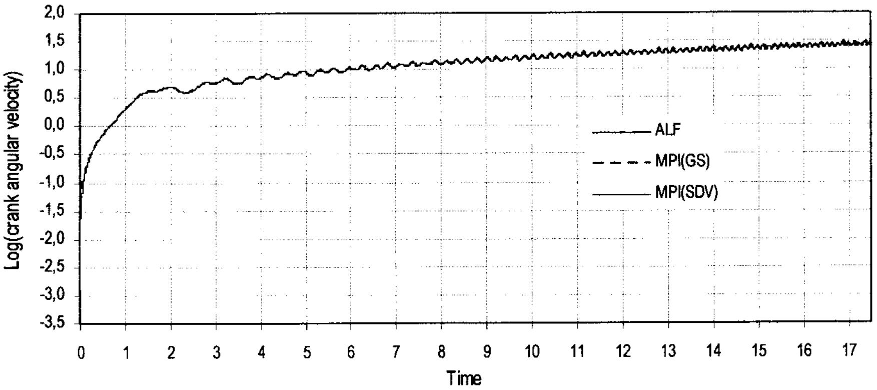

Figure 15. Crank angular velocity of the spatial slider crank obtained for the ALF, MPI (GS) and MPI (SDV) methods without using coordinate partitioning.

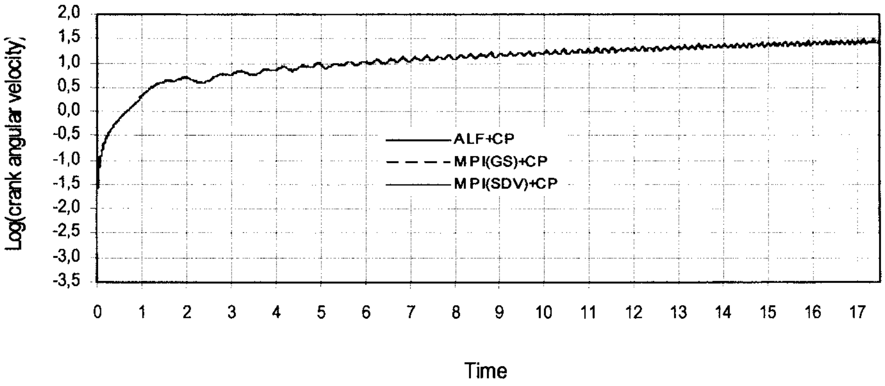

Figure 16. Crank angular velocity of the spatial slider crank obtained for the ALF, MPI (GS) and MPI (SDV) methods using coordinate partitioning.

than the other methodologies, even when used in conjunction with the coordinate partitioning method.

6. Conclusions

General formulations for the dynamic and kinematic analysis of rigid mechanical systems have been reviewed here. For all proposed applications it was observed that the control of the constraint violation is fundamental for long running times. Therefore, the use of the coordinate partitioning method is the only reliable form to ensure that the analysis does not fail due to constraint violations. It was also observed that the elimination of the redundant kinematic constraints, after they are identified by using the factorization procedure, is more efficient than the use of the pseudo-inverse methodologies. However, in the case of singular positions the use of the Moore–Penrose pseudo-inverse is fundamental. Nevertheless, the decision

Table IV. Computational costs per second of simulation for the different methodologies in the dynamic analysis.

	LM1	LM2	GS	SDV	LM2 + CP	GS + CP	SVD + CP	ALF + CP
No. Steps	2428	2428	2428	2428	1627	1768	1774	1368
Time (s)	30.31	30.34	55.52	47.07	39.5	58.27	58.82	28.47

on the branch of motion that the mechanism takes cannot be done automatically. It was also observed that the use of the coordinate partitioning method is not necessarily more costly than the constraint stabilizing procedures. In many of the cases simulated, the extra computational costs associated to the coordinate partitioning are compensated by a more stable integration process characterized by larger time steps.

The violation of the position and velocity constraint equations, that arise from the integration of the differential-algebraic equations, associated to the multibody models described by Cartesian coordinates, can be handled by using constraint stabilization methodologies, such as the Baumgarte stabilization method or the augmented Lagangian formulation, or by using the coordinate partitioning method. It was shown that the coordinate partitioning is the only method that ensures the stability of the integration process during all simulation time. The failures of the integration of the system state variables observed are attributed to the difficulty of the integration methodology to converge next to the singular positions because of the existence of several possible branches of motion. The use of stabilization methods, such as the ALF and the Baumgarte method, showed to be efficient in handling the constraints violations drifts, or at least to delay their growth, in the dynamic analysis but much less effective for the kinematic analysis. In all cases studied here the use of the coordinate partitioning method had the extra advantage of leading to a more stable response of the mechanical system resulting in the selection of larger time steps by the integration algorithms. For the dynamic analysis applications the average time step required when using the coordinate partitioning was 1.3 times larger than the time step required when other constraint stabilization procedure was selected while for the kinematic analysis it was 30 times larger. In the case of the kinematic analysis the computer time saved by having a more stable system response largely compensated the extra costs resulting from the computations associated to the coordinate partitioning method.

It was shown that the use of the coordinate partitioning method requires that a set of independent coordinates is defined beforehand. The factorization procedures that use full pivoting of the leading matrices not only provide the necessary tools for the coordinate partitioning but also identify the redundant constraints of the model. Therefore, in the process of using the coordinate partitioning method the elimination of the redundant constraints comes for free. The existence of redundant constraints or singular configurations for the multibody system can be handled by applying the augmented Lagrangian formulation or procedures that use the Moore-Penrose pseudo-inverse. This pseudo-inverse can be calculated using a singular value decomposition or by applying the Gram–Schmidt orthogonalization procedure. These methodologies proved to be able to handle efficiently the solution of the system equations of motion in the presence of redundant constraints and when singular positions are reached. The failures observed in the solution of the system response in some application cases was due to the convergence of the integration process rather than to difficulties of the ALF or MPI methods to handle the redundant constraints or the singular positions. The MPI methods always delivered solutions for the system equations of motion, even when such solutions were unfeasible and had no physical meaning. None of the methodologies used was able to steer the system solution when singular configurations were reached, making the selection of the branch of motion followed by the system virtually random.

Acknowledgement

The support of Fundação para a Ciência e Técnologia (FCT) through the PRAXIS XXI Project, with the reference PRAXIS/P/EME/14040/1998, on 'Biomecánica da Locomoção Humana Utilizando Modelos Avançados e Metodologias de Optimização' (Human Locomotion Biomechanics Using Advanced Models and Optimization Methodologies) is gratefully acknowledged.

References

1. Nikravesh, P.E., *Computer-Aided Analysis of Mechanical Systems*, Prentice Hall, Englewood Cliffs, NJ, 1988.
2. Petzold, L., Serban, R., Li, S., Raha, S. and Cao, Y., 'Sensitivity analysis and design optimization of differential-algebraic equation systems', in *Computational Aspects of Nonlinear Structural Systems with Large Rigid Body Motion*, J. Ambrósio and M. Kleiber (eds.), IOS Press, Amsterdam, 2001, 153–167.
3. Wehage, R. and Haug, E., 'Generalized coordinate partitioning of dimension reduction in analysis of constrained dynamic systems', *ASME Journal of Mechanical Design* **104**, 1982, 247–255.
4. Baumgarte, J., 'Stabilization of constraints and integrals of motion', *Conceptual Methods of Applied Mechanics and Engineering* **1**, 1972, 1–66.
5. Jalon, J. de and Bayo, E., *Kinematic and Dynamic Simulation of Multibody Systems*, Springer-Verlag, Heidelberg, 1993.
6. Arabyan, A. and Wu, F., 'An improved formulation for constrained mechanical systems', *Multibody Systems Dynamics* **2**(1), 1998, 49–69.
7. Kim, S.S., Vanderploeg, M.J., 'QR decomposition for state space representation of constrained mechanical dynamical systems', *ASME Journal of Mechanisms, Transmissions, and Automation in Design* **108**, 1986, 183–188.
8. Singh, R.P. and Likins, P.W., 'Singular value decomposition for constrained dynamical systems', *Journal of Applied Mechanics* **52**, 1985, 943–948.
9. Meijaard, J.P., 'Applications of the single value decomposition in dynamics', *Computer Methods in Applied Mechanics and Engineering* **103**, 1993, 161–173.

10. Udwadia, F.E. and Kalaba, R.E., *Analytical Dynamics: A New Approach*, Cambridge University Press, Cambridge, 1996.

11. Pina, H.L.G., *Métodos Numéricos (Numerical Methods)*, McGraw-Hill, Lisboa, Portugal, 1995.