Topology Based Linear Solver for Exploiting Sparsity in Multibody Dynamics

Radu Serban[1], Dan Negrut[1], Florian A. Potra[2], and Edward J. Haug[1]

[1] Department of Mechanical Engineering, The University of Iowa, Iowa City, IA 52246
[2] Department of Mathematics and Computer Science, The University of Iowa, Iowa City, IA 52246

Summary. In this paper, a new method to efficiently compute accelerations and Lagrange multipliers in the equations of multibody dynamics is presented. These quantities are the solution of a system of linear equations whose coefficient matrix has the special structure of an optimization matrix. This matrix is likely to have a large number of zero entries, according to the connectivity among bodies of the mechanical system. Advantage is taken of both the special structure and the sparsity of the coefficient matrix. Simple manipulations bring the original problem of solving a system of $n+m$ equations in $n+m$ unknowns, to an equivalent problem in which a positive definite system of dimension $m \times m$ has to be solved for the Lagrange multipliers. Accelerations are then efficiently determined. For certain mechanical system models, the bandwidth of the positive definite $m \times m$ matrix can be significantly reduced by appropriately numbering the joints connecting bodies of the model. Numerical experiments show the effectiveness of the proposed algorithm.

1. Introduction

In terms of the number of variables used to model a multibody system, there are the following two extreme approaches:

- the Cartesian representation, or the body representation, in which the mechanical system is represented using, for each body, a set of coordinates that specify the position of a particular point on that body and a set of parameters that specify the orientation of the body with respect to a global reference frame;
- the minimal form, or joint representation, in which the mechanical system is represented in terms of a minimal set of generalized coordinates.

The Cartesian formulation is very convenient in representing a mechanical system, because kinetic and kinematic information is readily available for each body in the system. The major drawback of the Cartesian approach is that the dimension of the problem increases dramatically as the number of bodies increases, when compared to the alternative provided by the joint formulation.

The focus in this paper is on improving the efficiency in solving the system of linear equations that determines accelerations and Lagrange multipliers in the Cartesian formulation. When using explicit numerical integration in solving the equations of multibody dynamics, accelerations are needed to

progress the simulation to the next time step, while the Lagrange multipliers are used to determine constraint forces. The objective is to take advantage of both the particular structure of the optimization matrix in the Cartesian formulation and a topology-induced sparsity pattern when solving for these unknowns. As mentioned by Andrezejewski and Schwerin (1995), at this time no code seems to exploit both the structure of the optimization matrix and its sparsity in a satisfying manner.

In this paper, it is shown that it is more efficient to first solve a reduced system for the Lagrange multipliers, and then to determine the accelerations by solving diagonal or almost diagonal systems of equations. A graph representation of the mechanism is introduced, and the relationship between vertex numbering and bandwidth of the reduced system is determined. An analysis of the number of operations needed to solve this system is performed, and the problem of optimal vertex numbering is stated. The solution to this problem determines the joint numbering scheme during the modelling stage. Finally, results of numerical experiments used to benchmark the proposed algorithm are presented.

2. Equations of Motion of Multibody Dynamics

The Newton-Euler constrained equations of motion for a multibody system assume the form

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \boldsymbol{\Phi}_{\mathbf{q}}^T(\mathbf{q})\boldsymbol{\lambda} = \mathbf{Q}(\mathbf{q}, \dot{\mathbf{q}}) \tag{1}$$

$$\boldsymbol{\Phi}(\mathbf{q}) = \mathbf{0} \tag{2}$$

where $\mathbf{q}$, $\dot{\mathbf{q}}$, and $\ddot{\mathbf{q}}$ are vectors of dimension n representing the generalized position, velocity, and acceleration of the system, $\mathbf{Q}(\mathbf{q}, \dot{\mathbf{q}}) \in \Re^n$ is the vector of generalized force, $\boldsymbol{\lambda} \in \Re^m$ is the vector of Lagrange multipliers, and $\mathbf{M}(\mathbf{q})$ is the $n \times n$ generalized mass matrix. Finally, $\boldsymbol{\Phi}_{\mathbf{q}}(\mathbf{q}) = \frac{\partial \boldsymbol{\Phi}(\mathbf{q})}{\partial \mathbf{q}}$ is the $m \times n$ constraint Jacobian matrix, where $m < n$. The kinematic constraints $\boldsymbol{\Phi}(\mathbf{q})$ are assumed to be independent, so the Jacobian matrix has full row rank.

Equations 1 and 2 represent a set of index-3 differential-algebraic equations (DAE) (Hairer and Wanner, 1996). One method of solving the DAE is to reduce it to ODE employing state-space reduction via constraint parametrization (Wehage and Haug, 1982; Potra and Rheinboldt, 1991). The state-space ODE (SSODE) are then integrated, and the solution of the original DAE problem is recovered. Details of the DAE solution recovery are involved and are not the subject of this paper.

In the process of state space reduction, the index-3 DAE of Eqs. 1 and 2 is reduced by differentiating the kinematic position constraints twice with respect to time. The constraint acceleration equations are then appended to the equations of motion in Eq. 1, to obtain the index-1 DAE

$$\begin{bmatrix} \mathbf{M} & \boldsymbol{\Phi}_{\mathbf{q}}^T \\ \boldsymbol{\Phi}_{\mathbf{q}} & \mathbf{0} \end{bmatrix} \begin{pmatrix} \ddot{\mathbf{q}} \\ \boldsymbol{\lambda} \end{pmatrix} = \begin{pmatrix} \mathbf{Q} \\ \boldsymbol{\gamma} \end{pmatrix} \tag{3}$$

where the right side of the acceleration constraint equations is given as

$$\boldsymbol{\gamma} = -(\boldsymbol{\Phi}_{\mathbf{q}}\dot{\mathbf{q}})_{\mathbf{q}}\dot{\mathbf{q}} - 2\boldsymbol{\Phi}_{\mathbf{q}t}\dot{\mathbf{q}} - \boldsymbol{\Phi}_{tt} \tag{4}$$

To progress the simulation to the next time step, integration of the SSODE requires computation of the generalized accelerations $\ddot{\mathbf{q}}$ from Eq. 3. The remainder of the paper is focused on methods that efficiently compute $\ddot{\mathbf{q}}$.

3. Solving the Augmented System

The coefficient matrix of the linear system in Eq. 3 is of dimension $(n+m) \times (n+m)$. In what follows, it is denoted by $\mathbf{A}$ and is called the augmented matrix. At each integration step this system is solved for $\ddot{\mathbf{q}}$ and, depending on the type of analysis being performed, for the Lagrange multipliers.

The dimension of the augmented matrix assumes large values for relatively simple mechanical systems. For the seven body mechanism studied in Section 4., modeled as a planar mechanical system, the number of generalized coordinates is 21; i.e., three coordinates (two positions and one orientation angle) for each of the seven bodies. The number m of constraint equations in Eq. 2 is 20. The mechanism thus has one degree of freedom, and the augmented matrix is of dimension 41×41. The situation is more drastic if the mechanism is modeled as a spatial mechanical system. In this case, each body in the model is described by 7 generalized coordinates (three positions and four orientation Euler parameters). It can be seen that the dimension of the augmented matrix builds up fast, especially when spatial mechanical systems with few degrees of freedom are considered.

3.1 Reduction of the Augmented System

The system in Eq. 3 is formally solved for $\ddot{\mathbf{q}}$ by using the set of first n equations to express the accelerations $\ddot{\mathbf{q}}$ in terms of the unknown Lagrange multipliers $\boldsymbol{\lambda}$.

$$\mathbf{M}\ddot{\mathbf{q}} = (\mathbf{Q} - \boldsymbol{\Phi}_{\mathbf{q}}^T\boldsymbol{\lambda}) \tag{5}$$

Assuming that $\mathbf{M}$ is nonsingular, which is shown below to be the case, the accelerations can be determined and substituted into the last m equations of Eq. 3, yielding the linear system

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

which can then be solved for $\boldsymbol{\lambda}$. For notational convenience, the dependency of the generalized mass matrix and of the constraint Jacobian on the generalized

coordinates, and of the generalized force on both generalized coordinates, and their first time derivatives is suppressed.

The matrix $\mathbf{B} = \boldsymbol{\Phi}_{\mathbf{q}} \mathbf{M}^{-1} \boldsymbol{\Phi}_{\mathbf{q}}^T$, referred to as the reduced matrix, is a symmetric, block banded matrix. Diagonal blocks in $\mathbf{B}$ have the form

$$\mathbf{B}_{jj} = \left(\boldsymbol{\Phi}^j_{\mathbf{q}_{i1}}\right) \mathbf{M}_{i1}^{-1} \left(\boldsymbol{\Phi}^j_{\mathbf{q}_{i1}}\right)^T + \left(\boldsymbol{\Phi}^j_{\mathbf{q}_{i2}}\right) \mathbf{M}_{i2}^{-1} \left(\boldsymbol{\Phi}^j_{\mathbf{q}_{i2}}\right)^T \tag{7}$$

if joint j connects bodies $i1$ and $i2$, and off-diagonal blocks have the form

$$\mathbf{B}_{jk} = \left(\boldsymbol{\Phi}^j_{\mathbf{q}_i}\right) \mathbf{M}_i^{-1} \left(\boldsymbol{\Phi}^k_{\mathbf{q}_i}\right)^T \qquad \text{for} \qquad j \neq k \tag{8}$$

if joints j and k are on the same body i. In Eqs. 7 and 8, $\boldsymbol{\Phi}^j_{\mathbf{q}_i}$ is the partial derivative of the constraint function corresponding to joint j with respect to coordinates $\mathbf{q}_i$ of body i, and $\mathbf{M}^{-1}$ is the inverse of the generalized mass matrix of body i.

3.2 Cartesian Formulation for Planar Models

For a planar mechanism, since the mass matrix $\mathbf{M}$ is constant and positive definite, and since the Jacobian matrix $\boldsymbol{\Phi}_{\mathbf{q}}$ is assumed to have full row rank, the reduced matrix $\mathbf{B}$ is positive definite. Efficient methods for solving the reduced system of Eq. 6 are discussed in Section 4.. Once the Lagrange multipliers are computed, accelerations are recovered from Eq. 5. When dealing with a planar model with nb bodies, the dimension of the mass matrix $\mathbf{M}$ is $(3nb \times 3nb)$, with the following structure:

$$\mathbf{M} = \begin{bmatrix} \mathbf{M}_1 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{M}_2 & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{M}_{n_b} \end{bmatrix} \tag{9}$$

The matrices $\mathbf{M}_i$ in Eq. 9 are diagonal and constant. It is therefore convenient to compute the inverse of the generalized mass matrix $\mathbf{M}$ of Eq. 9 during the preprocessing stage, and to recover the generalized acceleration at each time step via a level 2 BLAS (Basic Linear Algebra Subroutine) operation.

3.3 Cartesian Formulation for Spatial Models

The case of spatial multibody models requires special treatment. A fundamental difficulty is due to the fact that the orientation of a rigid body in space cannot be defined without singularities by only three parameters. This problem leads to nonintegrability of angular velocity. In order to avoid singularities, an increased set of orientation parameters is used, the simplest being the set of four Euler parameters $\mathbf{p}$ (Haug, 1989). Since the Euler parameters $\mathbf{p}_i = [e_{0i}, \mathbf{e}_i]^T \in \Re^4$ corresponding to body i are not independent, but must satisfy a constraint of the form

$$\boldsymbol{\Phi}_i^P = \mathbf{p}_i^T \mathbf{p}_i - 1 = 0 \tag{10}$$

additional algebraic constraint equations must be taken into consideration. Lagrange multipliers $\boldsymbol{\lambda}^P$, corresponding to the Euler parameter constraints, are thus introduced. The accelerations and Lagrange multipliers are then the solution of the linear system

$$\begin{bmatrix} \mathbf{M} & \mathbf{0} & \boldsymbol{\Phi}_\mathbf{r}^T & \mathbf{0} \\ \mathbf{0} & 4\mathbf{G}^T\mathbf{J}'\mathbf{G} & \boldsymbol{\Phi}_\mathbf{p}^T & \boldsymbol{\Phi}_\mathbf{p}^{PT} \\ \boldsymbol{\Phi}_\mathbf{r} & \boldsymbol{\Phi}_\mathbf{p} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Phi}_\mathbf{p}^P & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{pmatrix} \ddot{\mathbf{r}} \\ \ddot{\mathbf{p}} \\ \boldsymbol{\lambda} \\ \boldsymbol{\lambda}^P \end{pmatrix} = \begin{pmatrix} \mathbf{F}^A \\ 2\mathbf{G}^T\mathbf{n}'^A + 8\dot{\mathbf{G}}^T\mathbf{J}'\dot{\mathbf{G}}\mathbf{p} \\ \boldsymbol{\gamma} \\ \boldsymbol{\gamma}^P \end{pmatrix} \tag{11}$$

where $\mathbf{G}_i = [-\mathbf{e}_i, -\tilde{\mathbf{e}}_i + e_{0i}\mathbf{I}]$ is a 3×4 matrix depending on the Euler parameters $\mathbf{p}_i$, $\gamma^P = -2\dot{\mathbf{p}}_i^T\dot{\mathbf{p}}$ is the right side of the Euler parameter acceleration constraint equation, and $\mathbf{M}$ and $\mathbf{J}'$ are constant, positive definite mass and inertia matrices (Haug, 1989).

The reduction technique presented in the previous sections cannot be directly applied to the system of Eq. 11, since it assumes invertibility of the matrix

$$\bar{\mathbf{M}} = \begin{bmatrix} \mathbf{M} & \mathbf{0} \\ \mathbf{0} & 4\mathbf{G}^T\mathbf{J}'\mathbf{G} \end{bmatrix} \tag{12}$$

However, the above matrix is singular, as the following relation shows:

$$\bar{\mathbf{M}} \begin{pmatrix} \mathbf{0} \\ \mathbf{p} \end{pmatrix} = \begin{bmatrix} \mathbf{M} & \mathbf{0} \\ \mathbf{0} & 4\mathbf{G}^T\mathbf{J}'\mathbf{G} \end{bmatrix} \begin{pmatrix} \mathbf{0} \\ \mathbf{p} \end{pmatrix} = \begin{pmatrix} \mathbf{0} \\ \mathbf{0} \end{pmatrix} \tag{13}$$

and a reduced matrix $\mathbf{B}$ of the form of Eq. 6 cannot be constructed.

The solution for the formulation of Eq. 11 is to rearrange the rows of the augmented matrix and collect terms into an extended mass matrix of the form

$$\hat{\mathbf{M}} = \begin{bmatrix} \mathbf{M} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & 4\mathbf{G}^T\mathbf{J}'\mathbf{G} & \boldsymbol{\Phi}_\mathbf{p}^{PT} \\ \mathbf{0} & \boldsymbol{\Phi}_\mathbf{p}^P & \mathbf{0} \end{bmatrix} \tag{14}$$

With this arrangement, the coefficient matrix in Eq. 11 can be written as $\begin{bmatrix} \hat{\mathbf{M}} & \hat{\boldsymbol{\Phi}}_\mathbf{q}^T \\ \hat{\boldsymbol{\Phi}}_\mathbf{q} & \mathbf{0} \end{bmatrix}$ where $\hat{\boldsymbol{\Phi}}_\mathbf{q} = [\boldsymbol{\Phi}_\mathbf{r}, \boldsymbol{\Phi}_\mathbf{p}, \mathbf{0}]$ is obtained from $\boldsymbol{\Phi}_\mathbf{q} = [\boldsymbol{\Phi}_\mathbf{r}, \boldsymbol{\Phi}_\mathbf{p}]$ by inserting a number of zero columns associated with $\boldsymbol{\lambda}^P$. The matrix $\hat{\mathbf{M}}$ can be written as a block diagonal matrix with block 8×8 matrices

$$\hat{\mathbf{M}}_i = \begin{bmatrix} \mathbf{M}_i & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & 4\mathbf{G}_i^T\mathbf{J}_i'\mathbf{G}_i & 2\mathbf{p}_i \\ \mathbf{0} & 2\mathbf{p}_i^T & \mathbf{0} \end{bmatrix} \tag{15}$$

corresponding to body i on the diagonal. The Jacobian of the Euler parameter constraint $\left(\boldsymbol{\Phi}^P_{\mathbf{p}}\right)_i = 2\mathbf{p}_i^T$ is included in Eq. 15. The matrix $\hat{\mathbf{M}}_i$ is now invertible, with the inverse

$$\hat{\mathbf{M}}_i^{-1} = \begin{bmatrix} \mathbf{M}_i^{-1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \frac{1}{4}\mathbf{G}_i^T\mathbf{J}_i'^{-1}\mathbf{G}_i & \frac{1}{2}\mathbf{p}_i \\ \mathbf{0} & \frac{1}{2}\mathbf{p}_i^T & \mathbf{0} \end{bmatrix} \tag{16}$$

but it is not positive definite. However, the following result holds.

Proposition 3.1. *The extended mass matrix $\hat{\mathbf{M}}$ is positive definite on the null space of $\hat{\boldsymbol{\Phi}}_{\mathbf{q}}$; i.e. the reduced matrix $\hat{\mathbf{B}} = \hat{\boldsymbol{\Phi}}_{\mathbf{q}}\hat{\mathbf{M}}^{-1}\hat{\boldsymbol{\Phi}}_{\mathbf{q}}^T$ is positive definite.*

Proof. The matrix $\hat{\mathbf{B}}$ is shown to be positive definite by proving that it is identical to the reduced matrix $\mathbf{B}$ corresponding to the angular velocity formulation. Diagonal blocks in $\hat{\mathbf{B}}$ have the form

$$\hat{\mathbf{B}}_{jj} = \left(\hat{\boldsymbol{\Phi}}^j_{\mathbf{q}_{i1}}\right)\hat{\mathbf{M}}_{i1}^{-1}\left(\hat{\boldsymbol{\Phi}}^j_{\mathbf{q}_{i1}}\right)^T + \left(\hat{\boldsymbol{\Phi}}^j_{\mathbf{q}_{i2}}\right)\hat{\mathbf{M}}_{i2}^{-1}\left(\hat{\boldsymbol{\Phi}}^j_{\mathbf{q}_{i2}}\right)^T \tag{17}$$

if joint j connects bodies $i1$ and $i2$, and off-diagonal blocks have the form

$$\hat{\mathbf{B}}_{jk} = \left(\hat{\boldsymbol{\Phi}}^j_{\mathbf{q}_i}\right)\hat{\mathbf{M}}_i^{-1}\left(\hat{\boldsymbol{\Phi}}^k_{\mathbf{q}_i}\right)^T \qquad \text{for} \qquad j \neq k \tag{18}$$

if joints j and k are on the same body i. In Eqs. 17 and 18, the extended Jacobian of joint j connecting bodies $i1$ and $i2$ is

$$\hat{\boldsymbol{\Phi}}^j_{\mathbf{q}} = \left[\hat{\boldsymbol{\Phi}}^j_{\mathbf{q}_{i1}}, \hat{\boldsymbol{\Phi}}^j_{\mathbf{q}_{i2}}\right] = \left[\boldsymbol{\Phi}^j_{\mathbf{r}_{i1}}, \boldsymbol{\Phi}^j_{\mathbf{p}_{i1}}, \mathbf{0}, \boldsymbol{\Phi}^j_{\mathbf{r}_{i2}}, \boldsymbol{\Phi}^j_{\mathbf{p}_{i2}}, \mathbf{0}\right] \tag{19}$$

Consider a typical term $\hat{\mathbf{B}}_{jk}$ as in Eq. 18. Expanding the matrices involved, the following expression is obtained:

$$\begin{aligned} \hat{\mathbf{B}}_{jk} &= \left(\hat{\boldsymbol{\Phi}}^j_{\mathbf{q}_i}\right)\hat{\mathbf{M}}_i^{-1}\left(\hat{\boldsymbol{\Phi}}^k_{\mathbf{q}_i}\right)^T \\ &= \left[\boldsymbol{\Phi}^j_{\mathbf{r}_i}, \boldsymbol{\Phi}^j_{\mathbf{p}_i}, \mathbf{0}\right]\begin{bmatrix} \mathbf{M}_i^{-1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \frac{1}{4}\mathbf{G}_i^T\mathbf{J}_i'^{-1}\mathbf{G}_i & \frac{1}{2}\mathbf{p}_i \\ \mathbf{0} & \frac{1}{2}\mathbf{p}_i^T & \mathbf{0} \end{bmatrix}\begin{bmatrix} \boldsymbol{\Phi}^j_{\mathbf{r}_i}{}^T \\ \boldsymbol{\Phi}^j_{\mathbf{p}_i}{}^T \\ \mathbf{0} \end{bmatrix} \\ &= \left(\boldsymbol{\Phi}^j_{\mathbf{r}_i}\right)\mathbf{M}_i^{-1}\left(\boldsymbol{\Phi}^k_{\mathbf{r}_i}\right)^T + \frac{1}{4}\left(\boldsymbol{\Phi}^j_{\mathbf{p}_i}\right)\mathbf{G}_i^T\mathbf{J}_i'^{-1}\mathbf{G}_i\left(\boldsymbol{\Phi}^k_{\mathbf{p}_i}\right)^T \\ &= \left[\boldsymbol{\Phi}^j_{\mathbf{r}_i}, \boldsymbol{\Phi}^j_{\boldsymbol{\pi}'_i}\right]\begin{bmatrix} \mathbf{M}_i^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{J}_i'^{-1} \end{bmatrix}\left[\boldsymbol{\Phi}^j_{\mathbf{r}_i}, \boldsymbol{\Phi}^j_{\boldsymbol{\pi}'_i}\right]^T \end{aligned} \tag{20}$$

Using relations of the form (Haug, 1989)

$$\boldsymbol{\Phi}^j_{\mathbf{p}_i} = 2\boldsymbol{\Phi}^j_{\boldsymbol{\pi}'_i}\mathbf{G}_i + \mathbf{C}_i\mathbf{p}_i^T \tag{21}$$

where $\mathbf{C}_i$ is a constant matrix for body i, it follows that

$$\boldsymbol{\Phi}^j_{\mathbf{p}_i}\mathbf{G}_i^T = 2\boldsymbol{\Phi}^j_{\boldsymbol{\pi}'_i}\left(\mathbf{G}_i\mathbf{G}_i^T\right) + \mathbf{C}_i\left(\mathbf{p}_i^T\mathbf{G}_i^T\right) = 2\boldsymbol{\Phi}^j_{\boldsymbol{\pi}'_i}\mathbf{I} + \mathbf{C}_i\mathbf{0} = 2\boldsymbol{\Phi}^j_{\boldsymbol{\pi}'_i} \tag{22}$$

This result was used in the last equality of Eq. 20. Collecting blocks of the form of Eqs. 17 and 18, the reduced matrix $\hat{\mathbf{B}}$ has the form

$$\hat{\mathbf{B}} = [\boldsymbol{\Phi}_{\mathbf{r}}, \boldsymbol{\Phi}_{\boldsymbol{\pi}'}] \begin{bmatrix} \mathbf{M}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{J}'^{-1} \end{bmatrix} [\boldsymbol{\Phi}_{\mathbf{r}}, \boldsymbol{\Phi}_{\boldsymbol{\pi}'}]^T \tag{23}$$

Assuming that $[\boldsymbol{\Phi}_{\mathbf{r}}, \boldsymbol{\Phi}_{\boldsymbol{\pi}'}]$ has full row rank and since both $\mathbf{M}$ and $\mathbf{J}'$ are positive definite, it follows that $\hat{\mathbf{B}}$ is positive definite.□

The proposed solution sequence can thus be used for the case of spatial models in Cartesian coordinates with Euler parameters for orientation. Lagrange multipliers corresponding to joint constraints are first obtained by solving a system of linear equations whose coefficient matrix is the reduced matrix $\hat{\mathbf{B}}$, and the accelerations and Lagrange multipliers corresponding to Euler parameter constraint equations are recovered solving small systems with coefficient matrices of the form of Eq. 15. Note that, although the matrices of Eq. 15 are not constant, an explicit form for their inverse is given by Eq. 16. Recovery of $\ddot{\mathbf{q}}$ and $\boldsymbol{\lambda}^P$ is therefore reduced to nb matrix multiplications.

4. Solving the Reduced System

In order to efficiently solve the reduced system for Lagrange multipliers, advantage should be taken of the topology of the mechanical system. In what follows, a simple example is used to illustrate this point.

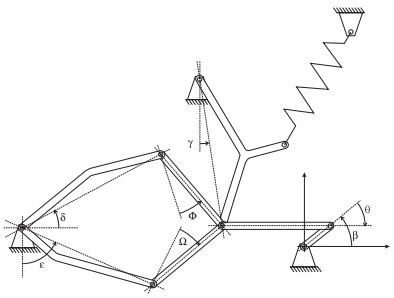

Fig. 1. Seven body mechanism

The mechanism of Fig. 1 is Andrew's squeezing mechanism, or the seven body mechanism (Schiehlen, 1990, Hairer and Wanner, 1996). This system is a closed loop mechanism, whose graph is presented in Fig. 2. In this graph, the joints are represented as vertices, while the bodies are the connecting edges. This representation is different from the usual one, in which bodies are vertices and joints are connecting edges of the graph. In the proposed

representation, one joint connects exactly two bodies, and the same body can appear more than once as an edge. This is the case when a body has more than one joint. Thus, when a body is attached through k joints with neighboring bodies it appears as k edges in the graph. The ground is not represented as a body. For the seven body mechanism, two different joint

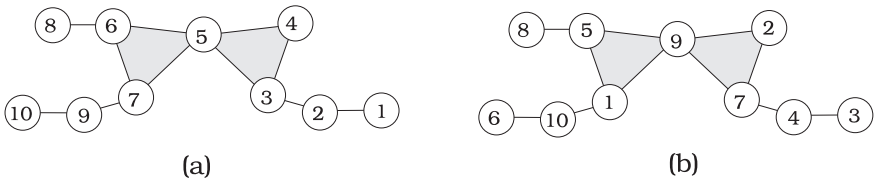

Fig. 2. Graph representation of the seven body mechanism

numbering sequences shown in Figs. 2 (a) and (b) yield the corresponding reduced matrices shown in Figs. 3 (a) and (b), respectively. Only the non-zero entries are represented.

Figure 3 shows that the bandwidth of the reduced matrix, and therefore the efficiency of a sparse solver, depend on the joint numbering scheme. The choice of joint numbering is made in the modelling stage. It is then used throughout the simulation. For improved performance, it is important to determine a strategy that automatically numbers the joints of the model, such that the amount of work in factorizing the reduced matrix is minimized.

Since the reduced matrix is positive definite, the factorization is based on block Cholesky decomposition, advantage being taken of the sparse-block structure of the reduced matrix. The block structure of this matrix is as given by Eqs. 7 and 8. The block width β_i of row i is defined in terms of blocks as

$$\beta_i = \max\{(i-j) \,:\, \mathbf{B}_{ij} \neq \mathbf{0}\,,\; j < i\} \tag{24}$$

The bandwidth β of the reduced matrix, in terms of blocks, is given by the maximum row width as

$$\beta = \max\{\beta_i \,:\, i = 1,\, 2,\, \cdots,\, m\} \tag{25}$$

The envelope (or profile) of the matrix is given by

$$env(\mathbf{B}) = \sum_{i=1}^{n} \beta_i \tag{26}$$

The computational work in Cholesky factorization of $\mathbf{B}$ that makes use of an envelope storage scheme can be bounded from above by

$$work(\mathbf{B}) = \frac{1}{2} \sum_{i=2}^{n} \beta_i(\beta_i + 3) \tag{27}$$

This estimate is an upper bound on the actual work in a block oriented Cholesky factorization algorithm. An operation in Eq. 27 is considered to be either a block-block multiplication, or a block inversion. Equation 27 does not take into account the square root of the diagonal elements (for the scalar case), or the Cholesky decomposition of a diagonal block (in the block form). If these operations are considered, the above formula becomes $work(\mathbf{B}) = \frac{1}{2}\sum_{i=2}^{n}(\beta_i^2 + 3\beta_i + 2)$.

The values of the bandwidth, envelope, and the actual work performed in the factorization of the reduced matrix depend on the choice of an ordering of the rows and columns in matrix $\mathbf{B}$. In general, the minima for these three quantities will not be obtained with the same ordering. It is known that minimizing the bandwidth of a matrix is an NP-complete problem, and minimizing either of the other two quantities considered above is a computationally demanding task as well. It is common practice to use a bandwidth and/or envelope reduction algorithm to reorder the matrix prior to applying Cholesky factorization. Although this approach does not exactly answer the question of minimizing the amount of work in Cholesky factorization, the results are satisfactory. Algorithms such as Gibbs-King, or Gibbs-Poole-Stockmeyer (Gibbs et. al., 1976; Lewis, 1982) perform this operation. These algorithms employ a local-search in the adjacency graph of the matrix. This graph is identical to the graph representation of the mechanical system being analyzed in Fig. 1. Thus, permuting rows and columns of the reduced matrix through symmetrical permutations of the form $\mathbf{P}^T\mathbf{B}\mathbf{P}$; i.e., renumbering the vertices in the adjacency graph of the matrix, is equivalent to renumbering the joints of the mechanical system. Therefore, the reordered index set given by any of the above algorithms is translated into a new numbering of the joints of the mechanical system.

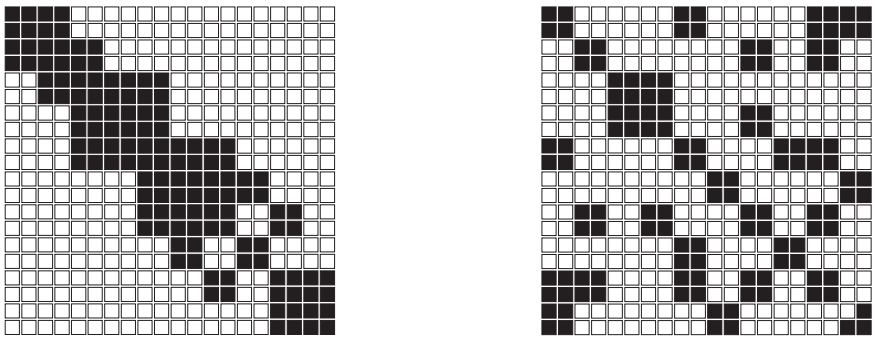

Fig. 3. Pattern of the B matrices for seven body mechanism

Blocks of the reduced matrix are manipulated as if they were simple entries in a matrix. These blocks are inherited from the structure of the problem, and their dimension is dictated by the number of constraint equations associated with different joint types. If each constraint equation were considered individually, the bandwidth and/or envelope reduction algorithm would result in better structured matrices, in terms of operations needed for Cholesky factorization. However, in this case, the immediate relationship between the topology of the mechanism and the structure of the reduced matrix is lost.

5. Numerical Results

First, a comparison of the performance of different solvers for the reduced system of Eq. 6 is presented. A dense solver (`dgetrf-dgetrs`) from Lapack, based on LU decomposition; a sparse solver (`ma47`) from Harwell, also using an LU decomposition; a dense Cholesky solver for full matrices (`dpptrf-dpptrs`) from Lapack; and a band Cholesky solver (`dpbtrf-dpbtrs`) from Lapack, were used to solve sets of linear equations with reduced matrices **B** as coefficient matrices.

The solvers `dgetrf-dgetrs` and `dpptrf-dpptrs` from Lapack are both dense solvers, the latter dealing with systems whose coefficient matrices are symmetric and positive definite. They are included here to gain a better understanding of the advantage gained when using sparsity information along with topology information in multibody dynamics simulation. The Harwell solvers are state-of-the-art solvers for sparse linear systems. They are based on multifrontal sparse Gaussian elimination, combined with a modified Markowitz strategy for local pivoting. The `ma47` routines used for this study are the latest Harwell routines available. The older version `ma28` is currently in use in a series of commercial packages (DADS).

All numerical experiments were performed on a HP 9000 model J210 with two PA 7200 processors. Results for the seven body mechanism are presented in Table 1. These results highlight several ways of improving the efficiency

Table 1. Timing results for the seven body mechanism. Solution of the reduced system

		Dense LU [a]	Dense Cholesky [a]	Sparse LU [b]	Band Cholesky [a]
Joint numbering	'bad'				0.410 ms
	'good'	0.794 ms	0.467 ms	0.880 ms	0.227 ms

[a] Lapack
[b] Harwell

of solving the given linear equations; The advantage of using sparse solvers

over dense solvers, the advantage of using Cholesky decomposition over LU decomposition when the matrix is known to be symmetric and positive definite, and the advantage of a better ordering of the blocks within the matrix. Note that, in the case of the seven body mechanism, the Harwell solver performed worse than both Lapack solvers. This is a reflection of the internal overhead in the `ma47` routines, which is an important factor for relatively small problems.

Next, focus is on comparing two different strategies of solving for both accelerations and Lagrange multipliers. The first approach is to use the Harwell solver on the augmented linear system. This method is the most common one in use. The second approach is to solve the reduced linear solver of Eq. 6 for the Lagrange multipliers and then to recover accelerations. The systems of linear equations was solved 1000 times and the average solution time for solving the system once is given in Table 2. Note that, for the second method, the best joint numbering scheme was used. The proposed algorithm was 5.59 times as efficient as the usual approach.

Table 2. Comparison for the seven body mechanism

Method 1 [a]	Method 2 [b]
1.2933 ms	0.2313 ms

[a] Harwell solver applied to Eq. 3.
[b] Band Cholesky solver applied to Eq. 6 and recovery of accelerations from Eq. 5.

As an example of a realistic spatial multibody mechanism, a 14-body High Mobility Multipurpose Wheeled Vehicle (HMMWV) model is used. The associated graph representation, for an optimal joint numbering scheme, is presented in Fig. 4. The model consists of the following bodies:

B1	chassis	B8	steering rack
B2	front left lower control arm	B9	rear left lower control arm
B3	front left upper control arm	B10	rear left upper control arm
B4	front right wheel assembly	B11	rear right wheel assembly
B5	front right lower control arm	B12	rear right lower control arm
B6	front right upper control arm	B13	rear right upper control arm
B7	front right wheel assembly	B14	rear right wheel assembly

connected by 8 revolute joints, 8 spherical joints, 4 spherical-spherical (or distance) composite joints, and one translational joint. In the Cartesian formulation with Euler parameters for orientation, the vector of generalized coordinates has dimension $n = 98$. The 21 joints in the model and Euler parameter normalization constraints yield $m = 87$ constraints. Therefore, the

model has $n_{dof} = 11$ degrees of freedom; six for the position and orientation of the chassis, one jounce degree of freedom for each suspension, and one degree of freedom for the steering.

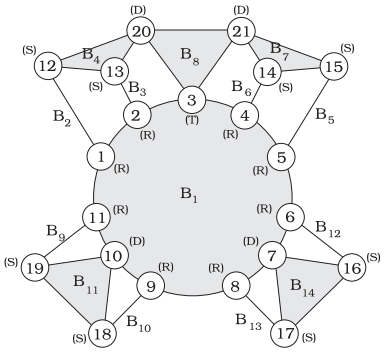

Fig. 4. Graph representation of the HMMWV 14-body model

For this example, the proposed solution technique was compared with the usual approach of decomposing the augmented matrix. The following four combinations of solution sequences and linear solvers are presented:

1 The Lapack dense linear solver `dgetrf-dgetrs`, based on LU decomposition for general matrices, was applied to the augmented linear system of Eq. 3 to evaluate both accelerations and Lagrange multipliers
2 The Harwell sparse solver `ma47`, based on LU decomposition of symmetric matrices, was used to solve the augmented linear system of Eq. 3
3 The Harwell Cholesky decomposition `ma35`, for banded symmetric, positive definite matrices, was used to factorize the reduced matrix $\hat{\mathbf{B}}$ and solve Eq. 6 for the Lagrange multipliers corresponding to joint constraints. Accelerations and Lagrange multipliers for Euler parameter constraints were then recovered using the matrix inverses of Eq. 16
4 A Lapack Cholesky decomposition routine (`dpbtrf`) for banded symmetric, positive definite matrices, replaced the Harwell routine of case 3.

Timing results are given in Table 3. The most efficient solution technique is the last. The speed-up obtained from the usual strategy for solving the augmented system, using a sparse LU decomposition (case 2), was 4.111. The small difference between cases 3 and 4 is due to the different internal storage algorithms in the subroutines `dpbtrf` from Lapack and `ma35` from Harwell. For the HMMWV example, the joint numbering sequence did not have a significant influence. Because of the particular configuration of this mechanism, even the optimal joint numbering of Fig. 4 results in a bandwidth of 63 for the reduced matrix. The joint numbering scheme becomes an important factor for mechanisms with long subchains.

CPU times for spatial models are significantly larger than those for planar models. The reason is the more involved computations required to generate

Table 3. Comparison for the HMMWV 14-body model

Experiment 1	Experiment 2	Experiment 3	Experiment 4
22.88 ms	12.62 ms	3.17 ms	3.07 ms

the coefficient matrices, as well as the vector of generalized forces. However, for all examples considered, the proposed solution technique proved to be at least twice as efficient as the classical approach of solving directly for both accelerations and Lagrange multipliers.

6. Conclusions

An efficient method for solving for accelerations and Lagrange multipliers in a multibody dynamics explicit formulation is presented. The particular structure of the augmented matrix, and the topology-induced sparsity pattern of the reduced matrix, are key factors that enable efficient linear solver sequences.

By simple manipulations, the problem of solving an $(m+n) \times (m+n)$ system of linear equations is reduced to the equivalent problem of solving two smaller systems. First, a reduced linear system is solved for the Lagrange multipliers. The $m \times m$ coefficient matrix of this system is symmetric, banded, and positive definite. By appropriately denoting the joints of the mechanical system, the band width of the reduced matrix can be minimized. A band Cholesky factorization, which takes advantage of the sparsity pattern of the reduced matrix, is employed. Next, accelerations are recovered by solving an $n \times n$ block diagonal system whose coefficient matrix can be explicitly factorized. The process of recovering accelerations, once the Lagrange multipliers are determined, is thus shown to be a computationally efficient task.

The proposed approach was shown to be effective when used in conjunction with the Euler parameter formulation. For a spatial 14-body model of the HMMWV, a speed-up of 4.111 was obtained, compared with the classical approach using the Harwell solver on the augmented linear system. Numerical experiments with different spatial models have shown similar speed-up ratios of 3.0 to 5.0.

7. Acknowledgment

This research was supported by the US Army Tank-Automotive Command (TACOM), through the Automotive Research Center (Department of Defense contract number DAAE07-94-R094).

References

Andrezejewski, T., Schwerin, R. (1995):Exploiting sparsity in the integration of multibody systems in descriptor form. Preprint 95-24, Universitat Heidelberg

DADS Reference Manual; Revision 8.0 (1995), CADSI, Coralville, Iowa

Gibbs, N.E., Poole, W.G., and Stockmeyer, P.K. (1976): An algorithm for reducing the bandwidth and profile of a sparse matrix. SIAM J. Num. Anal., vol. 13, pp. 236-249

Golub, G.H. and Van Loan, C.F. (1989): Matrix computation, John Hopkins University Press, Baltimore

Harwell subroutine library - Specifications (1995), AEA Technology, Harwell Laboratory, Oxfordshire, England

Haug, E.J. (1989): Computer-aided kinematics and dynamics of mechanical systems, VolumeI: Basic methods. Springer-Verlag, Berlin

Hairer, E. and Wanner, G. (1996): Solving ordinary differential equations II: Stiff and differential-algebraic equations. Springer-Verlag, Berlin

Lewis, J.G. (1982): Implementation of the Gibbs-Poole-Stockmeyer and Gibbs-King algorithms. ACM Trans. on Math. Soft., vol. 8, pp. 180-189

Potra, F.A. and Rheinboldt, W.C. (1991): On the numerical solution of the Euler-Lagrange equations. J. Mechanics of Structures and Machines, vol 19(1), pp. 155-192

Schiehlen, W. (1990): Multibody system handbook, Springer-Verlag, Berlin

Wehage, R.A. and Haug, E.J. (1982): Generalized coordinate partitioning for dimension reduction in analysis of constrained dynamic systems. ASME Advanced Automotive Technologies, DSC-vol. 52, pp. 71-79