Journal of Sound and Vibration 327 (2009) 557–563

Contents lists available at ScienceDirect

Journal of Sound and Vibration

journal homepage: www.elsevier.com/locate/jsvi

A two-loop sparse matrix numerical integration procedure for the solution of differential/algebraic equations: Application to multibody systems

Ahmed A. Shabana *, Bassam A. Hussein

Department of Mechanical and Industrial Engineering, University of Illinois at Chicago, 842 West Taylor Street, Chicago, IL 60607-7022, USA

ARTICLE INFO

Article history:
Received 14 May 2009
Accepted 20 June 2009
Handling Editor: A.V. Metrikine
Available online 14 July 2009

ABSTRACT

In this paper, a two-loop implicit sparse matrix numerical integration (TLISMNI) procedure for the solution of constrained rigid and flexible multibody system differential and algebraic equations is proposed. The proposed method ensures that the kinematic constraint equations are satisfied at the position, velocity and acceleration levels. In this method, a sparse Lagrangian augmented form of the equations of motion that ensures that the constraints are satisfied at the acceleration level is first used to solve for all the accelerations and Lagrange multipliers. The independent coordinates and velocities are then identified and integrated using HTT or Newmark formulas, expressed in this paper in terms of the independent accelerations only. The constraint equations at the position level are then used within an iterative Newton–Raphson procedure to determine the dependent coordinates. The dependent velocities are determined by solving a linear system of algebraic equations. In order to effectively exploit efficient sparse matrix techniques and have minimum storage requirements, a two-loop iterative method is proposed. Equally important, the proposed method avoids the use of numerical differentiation which is commonly associated with the use of implicit integration methods in multibody system algorithms. Numerical examples are presented in order to demonstrate the use of the new integration procedure.

1. Introduction

Sparse matrix techniques [1] are an integral part of the multibody system computational algorithms which are designed to solve a system of differential and algebraic equations (DAEs) [2]. The differential equations govern the dynamic motion of the system, while the kinematic constraint equations describe the conditions of connectivity between the bodies as well as specified motion trajectories. Mechanical joints, which are used to connect the bodies, eliminate some of the system degrees of freedom. In general, each of these joints is used to connect no more than two bodies, and therefore, the joint kinematic constraints are only function of the coordinates of the two bodies connected by this joint. By utilizing this fact, one can obtain a sparse matrix structure of the dynamic equations which can be efficiently solved, in the case of non-stiff system, using explicit numerical integration methods and sparse matrix techniques.

Sparse matrix techniques, on the other hand, are not fully utilized when implicit numerical integration methods are used. In the case of flexible multibody systems with very stiff components, the use of implicit numerical integration

* Corresponding author.
E-mail address: shabana@uic.edu (A.A. Shabana).

0022-460X/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.jsv.2009.06.020

methods becomes necessary. New large deformation flexible multibody approaches, such as the finite element absolute nodal coordinate formulation (ANCF), lead to a constant mass matrix [3]. By using Cholesky coordinates, one obtains a generalized identity inertia matrix, leading to an optimum sparse matrix structure of the equations of motion. For large systems, it is necessary to preserve this optimum sparse matrix structure in order to obtain an efficient solution. It is the purpose of this investigation to introduce a new implicit sparse matrix integration method for large scale multibody system applications. The proposed method ensures that the kinematic constraint equations are satisfied at the position, velocity and acceleration levels. The method requires minimum storage and does not require the use of numerical differentiation which characterizes many of the existing implicit integration methods when used in multibody system algorithms. In the proposed method, the Lagrangian augmented form of the equations of motion, which has a sparse matrix structure, is solved for all the accelerations and Lagrange multipliers. The independent accelerations can be identified and integrated using HHT [4], Newmark formulas [5] or other implicit integration methods to determine the independent coordinates and velocities. Dependent coordinates and velocities can be determined by solving the kinematic constraint equations at the position and velocity levels, respectively. The proposed sparse matrix implicit procedure is based on a two-loop iteration procedure. One iteration loop is associated with the solution of linear equations for the accelerations and Lagrange multipliers as well as the velocities, while the second iteration loop is based on a Newton–Raphson algorithm and is associated with the solution of the nonlinear constraint equations for the coordinates. The basic equations used in this implicit procedure which avoids the numerical differentiation are the same as the equations used in some of the existing explicit procedures [6,7]. Therefore, some of the existing explicit procedures can be easily modified to allow for the implicit solution of the equations of motion in the case of large scale stiff systems.

2. Constrained multibody system equations

The constrained multibody system dynamics is governed by the following differential and algebraic equations:

$$\mathbf{M}\ddot{\mathbf{q}} + \mathbf{C}_{\mathbf{q}}^{\mathrm{T}}\boldsymbol{\lambda} = \mathbf{Q}, \tag{1}$$

and

$$\mathbf{C}(\mathbf{q}, t) = \mathbf{0}. \tag{2}$$

In these equations, $\mathbf{M}$ is the system mass matrix, $\mathbf{q}$ is the vector of the system coordinates, $\boldsymbol{\lambda}$ is the vector of Lagrange multipliers, $\mathbf{Q}$ is the vector of all forces including external, elastic, Coriolis and centrifugal forces, $\mathbf{C}$ is the vector of the constraint functions, t is time, and $\mathbf{C}_{\mathbf{q}}$ is the Jacobian matrix of the kinematic constraint equations. In the case of flexible bodies, the vector of coordinates $\mathbf{q}$ includes the coordinates $\mathbf{q}_r$ that describe the reference motion and the elastic coordinates $\mathbf{q}_f$ that describe the deformation of the bodies. The elastic coordinates can be the modal coordinates used in the small deformation floating frame of reference formulation or the absolute nodal coordinates used in the case of large deformations [3].

Differentiating Eq. (2) with respect to time once and twice, one obtains the following two sets of equations that define the constraints at the velocity and acceleration levels:

$$\mathbf{C}_{\mathbf{q}}\dot{\mathbf{q}} = -\mathbf{C}_t, \tag{3}$$

$$\mathbf{C}_{\mathbf{q}}\ddot{\mathbf{q}} = \mathbf{Q}_d, \tag{4}$$

where $\mathbf{C}_t$ is the partial derivative of the constraint equations with respect to time, and $\mathbf{Q}_d$ is a quadratic velocity vector that results from the differentiation of the constraint equations twice with respect to time [3].

It is important in the sparse matrix implicit procedure proposed in this investigation to note that Eq. (2) is a nonlinear function of the coordinates, while Eqs. (3) and (4) are linear functions of the velocities and accelerations, respectively. Furthermore, Eq. (1) is linear in the accelerations and Lagrange multipliers. These observations will be utilized in the sparse matrix implicit procedure proposed in this paper.

3. Basic equations

In the proposed implicit procedure, as in the case of the explicit procedure, Eqs. (1) and (4) are combined to form the following augmented form of the equations of motion:

$$\begin{bmatrix} \mathbf{M} & \mathbf{C}_{\mathbf{q}}^{\mathrm{T}} \\ \mathbf{C}_{\mathbf{q}} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \mathbf{Q} \\ \mathbf{Q}_d \end{bmatrix}. \tag{5}$$

This equation, which has a sparse matrix structure, is linear in the accelerations and Lagrange multipliers. Therefore, the solution of this equation for the accelerations and Lagrange multipliers does not require the use of an iterative procedure. Furthermore, in the case of the absolute nodal coordinate formulation, Cholesky coordinates can be used leading to a constant identity mass matrix associated with the large deformation Cholesky coordinates. This leads to an optimum sparse matrix structure that requires minimum storage.

After solving Eq. (5), the independent accelerations can be identified and integrated using simple HHT, Newmark formulas or other known integration methods. In this case, the independent velocities and coordinates can be determined using the following two equations [4]:

$$(\dot{\mathbf{q}}_i)_{n+1} = (\dot{\mathbf{q}}_i)_n + h((1-\gamma)(\ddot{\mathbf{q}}_i)_n + \gamma(\ddot{\mathbf{q}}_i)_{n+1}), \tag{6}$$

$$(\mathbf{q}_i)_{n+1} = (\mathbf{q}_i)_n + h(\dot{\mathbf{q}}_i)_n + \frac{h^2}{2}((1-2\beta)(\ddot{\mathbf{q}}_i)_n + 2\beta(\ddot{\mathbf{q}}_i)_{n+1}), \tag{7}$$

where $\mathbf{q}_i$ refers to the vector of independent coordinates, subscripts n and n+1 refer, respectively, to solutions at time step n and n+1; h is the time step, and β and γ are assumed parameters [4]. Note that in Eqs. (6) and (7) only the independent accelerations are integrated to determine the independent coordinates and velocities.

Eqs. (2), (3), (5)–(7) can be combined to obtain a system of nonlinear equations that can be solved iteratively. These equations are given as follows:

$$\mathbf{F} = \begin{bmatrix} \mathbf{F}_1 \\ \mathbf{F}_2 \\ \mathbf{F}_3 \\ \mathbf{F}_4 \\ \mathbf{F}_5 \end{bmatrix} = \begin{bmatrix} (\mathbf{M}\ddot{\mathbf{q}} + \mathbf{C}_{\mathbf{q}}^{\mathrm{T}}\boldsymbol{\lambda} - \mathbf{Q})_{n+1} \\ \mathbf{C}((\mathbf{q})_{n+1}, t_{n+1}) \\ (\mathbf{C}_{\mathbf{q}}\dot{\mathbf{q}})_{n+1} + (\mathbf{C}_t)_{n+1} \\ (\mathbf{C}_{\mathbf{q}}\ddot{\mathbf{q}})_{n+1} - (\mathbf{Q}_d)_{n+1} \\ (\dot{\mathbf{q}}_i)_{n+1} - (\dot{\mathbf{q}}_i)_n - h((1-\gamma)(\ddot{\mathbf{q}}_i)_n + \gamma(\ddot{\mathbf{q}}_i)_{n+1}) \\ (\mathbf{q}_i)_{n+1} - (\mathbf{q}_i)_n - h(\dot{\mathbf{q}}_i)_n - \frac{h^2}{2}((1-2\beta)(\ddot{\mathbf{q}}_i)_n + 2\beta(\ddot{\mathbf{q}}_i)_{n+1}) \end{bmatrix} = \mathbf{0}. \tag{8}$$

Since the number of constraint equations is equal to the number of dependent coordinates, the number of equations in the nonlinear system of Eq. (8) is equal to the number of unknowns $\ddot{\mathbf{q}}$, $\boldsymbol{\lambda}$, $\mathbf{q}$, and $\dot{\mathbf{q}}$. Therefore, in principle, one can solve this system iteratively using a Newton–Raphson method at each time step to determine all the unknowns. The solutions of these nonlinear equations will satisfy the kinematic constraints at the position, velocity and acceleration levels.

Using Eq. (8), however, as the basis for the numerical solution of the multibody system equations has several drawbacks. First, numerical differentiation of the equations of motion with respect to the coordinates and velocities is necessary in many multibody system applications. Second, the resulting coefficient matrix used in the Newton–Raphson iterations does not have an optimum sparse matrix structure as the result of the numerical differentiation. Third Newton–Raphson method and numerical differentiation can be source of numerical problems in the case of a very large system of equations. In order to avoid these drawbacks a two-loop implicit sparse matrix numerical integration (TLISMNI) method is proposed in the following section. By using the TLISMNI method, numerical differentiation can be avoided and optimum sparse matrix structure that requires minimum storage can be achieved. Furthermore, the basic equations used in the TLISMNI procedure are the same as the equations used in some existing explicit procedures [3,6,7]. Therefore, these explicit procedures can be easily modified to provide as an alternate an implicit solution procedure that can be used in the case of large scale stiff constrained multibody systems.

4. TLISMNI method

As pointed out in the preceding section, an optimum sparse matrix structure that requires minimum storage can be obtained when Eq. (8) is iteratively solved. In order to obtain this optimum sparse matrix structure, a two-loop iterative procedure is proposed. All the equations solved in this two-loop procedure are sparse matrices, and therefore, sparse matrix techniques can be employed. The solution obtained, as previously mentioned, will satisfy the constraint equations at the position, velocity and acceleration levels. Furthermore, numerical differentiation of the equations of motion can be avoided.

The two-loop implicit procedure proposed in this study has an iterative outer loop that involves equations which are linear in the velocities, accelerations, and Lagrange multipliers. Another iterative inner Newton–Raphson loop that involves the kinematic constraint equations which are nonlinear functions in the coordinates is used. The proposed implicit procedure requires identifying the system independent accelerations, which are integrated. The dependent coordinates and velocities are determined by solving the kinematic constraint equations at the position and velocity levels, respectively.

The computational algorithm based on the TLISMNI method can be summarized as follows:

1. Assuming that the state of the system is known at time t_n, the constraint Jacobian matrix $\mathbf{C}_{\mathbf{q}}$ can be evaluated and used with a Gaussian procedure to determine the independent coordinates of the system $\mathbf{q}_i$. Therefore, the total vector of the system coordinates $\mathbf{q}$ can be partitioned to independent and dependent coordinates and can be written as $\mathbf{q} = [\,\mathbf{q}_i^{\mathrm{T}} \quad \mathbf{q}_d^{\mathrm{T}}\,]^{\mathrm{T}}$, where $\mathbf{q}_d$ is the vector of dependent coordinates. Note that the number of dependent coordinates is equal to the number of kinematic constraint equations.
2. Having identified the independent coordinates at time t_n, the outer loop iterations start in order to reach the solution at time t_{n+1}. To this end, Eq. (5), which has a sparse coefficient matrix, is constructed and solved for all the accelerations and Lagrange multipliers. At the iteration k, the solution of Eq. (5) defines the accelerations $\ddot{\mathbf{q}}_k$ and Lagrange multipliers

$\boldsymbol{\lambda}_k$. Note that Eq. (5) is linear in the accelerations and Lagrange multipliers, and therefore, its solution does not require the use of an iterative procedure.

3. Using the independent accelerations, the independent velocities and coordinates can be obtained at time t_{n+1} using the HHT formulas of Eqs. (6) and (7), respectively, or any other integration formulas.
4. Knowing the independent coordinates at time t_{n+1}, the nonlinear kinematic constraint equations can be solved for the dependent coordinates using an iterative Newton–Raphson procedure. This step defines the inner loop. In order to exploit sparse matrix techniques, the following system is solved during the Newton–Raphson iterations [6,7, p. 359]:

$$\begin{bmatrix} \mathbf{C_q} \\ \mathbf{I}_{in} \end{bmatrix} \Delta\mathbf{q} = \begin{bmatrix} -\mathbf{C} \\ \mathbf{0} \end{bmatrix}, \tag{9}$$

where $\Delta\mathbf{q}$ is the vector of Newton differences, and $\mathbf{I}_{in}$ is a matrix that contains zeros and ones, with ones in the columns associated with the independent coordinates in order to ensure that the independent coordinates remain fixed during the Newton–Raphson iterations.

5. After determining the coordinates, the algorithm exits the inner loop and returns to the outer loop. Using the coordinates and the independent velocities, the constraint equations at the velocity level can be solved for the dependent velocities. In order to obtain a sparse matrix structure, the following system of equations is solved for the velocities:

$$\begin{bmatrix} \mathbf{C_q} \\ \mathbf{I}_{in} \end{bmatrix} \dot{\mathbf{q}} = \begin{bmatrix} -\mathbf{C}_t \\ \mathbf{0} \end{bmatrix}. \tag{10}$$

Note that this system is linear in the velocities, and therefore, its solution does not require the use of an iterative procedure.

6. Knowing the coordinates and velocities, Eq. (5) can be constructed again and solved for the accelerations $\ddot{\mathbf{q}}_{k+1}$ and Lagrange multipliers $\boldsymbol{\lambda}_{k+1}$. If the norms $\|\ddot{\mathbf{q}}_{k+1} - \ddot{\mathbf{q}}_k\|$, and $\|\boldsymbol{\lambda}_{k+1} - \boldsymbol{\lambda}_k\|$ are smaller than a specified tolerance, convergence is achieved, otherwise go to Step 2 for the next iteration of the outer loop.

It is clear that all systems of equations used in this algorithm have sparse coefficient matrices that require minimum storage. Furthermore, no numerical differentiation of the external or inertia forces with respect to the coordinates and velocities is required.

5. Implementation issues

There are several implementation issues that need to be considered in order to improve the efficiency and robustness of the proposed algorithm. These issues are discussed below:

1. In most multibody system applications, the locations of the non-zero elements of the mass and constraint Jacobian matrices do not change. Therefore, these locations can be defined only once in advance of the dynamic simulation. If one set of independent coordinates is used throughout the simulation, the locations of the non-zero elements of all the sparse coefficient matrices do not change. In this case, symbolic factorization and scaling of the coefficient matrices are not necessary at every iteration. In fact, in some applications, the LU factorization can be skipped for several iterations, and only back substitution is used to solve the sparse matrix equations.
2. In the case of the absolute nodal coordinate formulation, an optimum sparse matrix structure that requires minimum storage can be obtained by using the large displacement Cholesky coordinates. The generalized inertia matrix associated with the large displacement Cholesky coordinates is the identity matrix. The use of this identity matrix leads to a significant reduction in the number of non-zero entries of the coefficient matrix of Eq. 5.
3. Different numerical integration formulas and methods can be used with the proposed implicit procedure. For example, numerical damping can be added to the equations of motion as suggested by Hilber, Hughes and Taylor [4]. It is important, however, to make sure that the numerical damping does not lead to a deterioration of the accuracy of the solution. One method that can be used to include the effect of numerical damping is to replace Eq. (1) in the algorithm by the following equation [8,9]:

$$(\mathbf{M}\ddot{\mathbf{q}})_{n+1} + (1+\alpha)(\mathbf{C}_\mathbf{q}^{\mathrm{T}}\boldsymbol{\lambda} - \mathbf{Q})_{n+1} - \alpha(\mathbf{C}_\mathbf{q}^{\mathrm{T}}\boldsymbol{\lambda} - \mathbf{Q})_n = \mathbf{0}, \tag{11}$$

where α is a selected numerical damping parameter.

4. Error check, selection of the time step, and convergence criteria can have a significant effect on the accuracy and efficiency of the proposed method. The formulas for error check and selection of the time step and the convergence criteria proposed in [8,9] can be used with the proposed method. The use of these criteria with the implicit integration method proposed in this study is currently being investigated and the results will be reported in future publications.
5. Formulas of Eqs. (6) and (7) are used to integrate the independent accelerations. However, these equations can also be applied to predict the dependent coordinates before the start of the iterations of the inner loop. These predicted

coordinates can be used as initial guess for the Newton–Raphson iterations that involve the nonlinear kinematic constraints. Such a procedure can significantly speed up the convergence.

6. Numerical examples

In order to test the performance of the sparse matrix implicit integration method proposed in this investigation, two different simple examples are used. The results of the simulation obtained using the proposed method are compared with the results obtained using the explicit predictor–corrector Adams method [10]. The explicit predictor–corrector Adams method is used in a numerical algorithm that also ensures that the kinematic constraints are satisfied at the position, velocity and acceleration levels [3,6].

In the first example, a simulation of rigid pendulum falling under the effect of the gravity force is performed. The pendulum used in this example is assumed to be initially horizontal and connected to the ground by a spherical joint, as shown in Fig. 1. The pendulum is assumed to have a length of 0.4 m, cross-sectional area of $0.02 \times 0.02\,m^2$, a density of $7800\,kg/m^3$. The vertical position of the center of gravity of the beam when the two integration methods are used is shown in Fig. 2. The energy balance of the kinetic and potential energies is shown in Fig. 3. The results presented in this figure, obtained using the proposed integration method, demonstrate that the solution satisfies the principle of work and energy.

The same example is solved again by assuming that the beam is a flexible body. The beam is assumed to have a modulus of elasticity $2 \times 10^7\,N/m^2$, and a Poisson ratio of 0.3. The beam is modeled by using the large displacement absolute nodal coordinate formulation [11]. The beam elastic forces are formulated using the general continuum mechanics approach [12]. Fig. 4 shows the area ratio of the deformed cross-section relative to the undeformed cross-section at the point on the beam that is connected by the spherical joint. The results presented in this figure are obtained using the proposed integration method. Fig. 5 compares the area ratios, as defined by Nonson's formula, predicted using the proposed integration method and Adams method. Fig. 6 shows how the mid-point transverse deformation is measured in this investigation, while Fig. 7 compares the transverse deformation results obtained using the two integration methods.

7. Summary

Many of the existing implicit numerical integration procedures designed for multibody system applications require the use of numerical differentiation in order to construct the coefficient matrix used in the Newton–Raphson iterations.

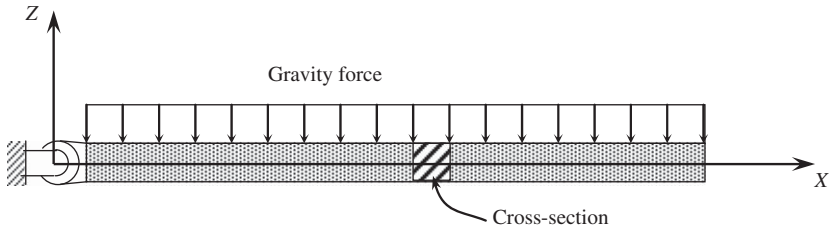

Fig. 1. Pendulum initial configuration.

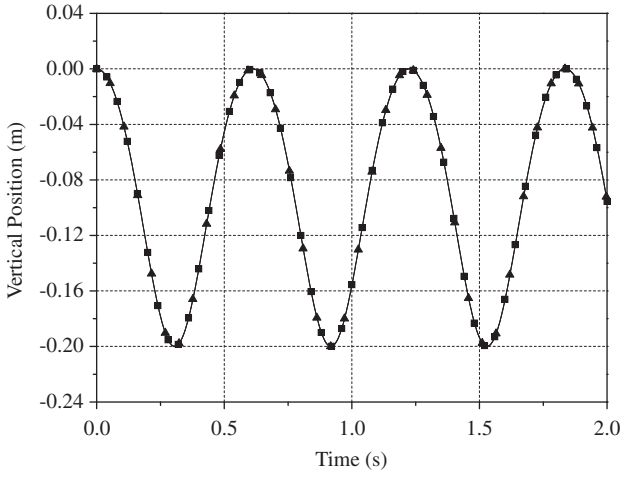

Fig. 2. Vertical position of the pendulum center of gravity (– ■ – proposed method; – ▲ – Adams method).

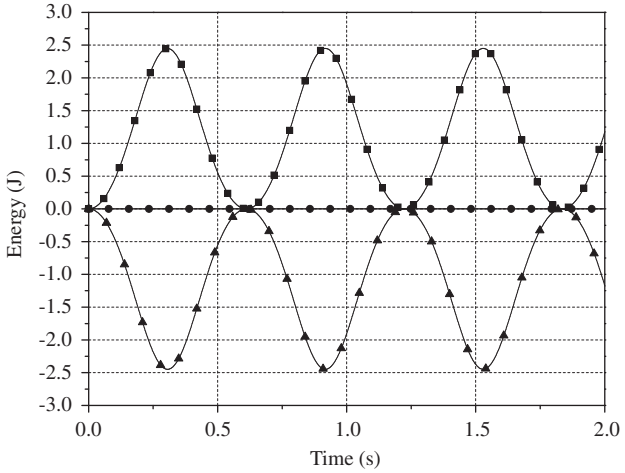

Fig. 3. Energy balance of the rigid pendulum (– ■ – kinetic energy; – ● – total energy; – ▲ – potential energy).

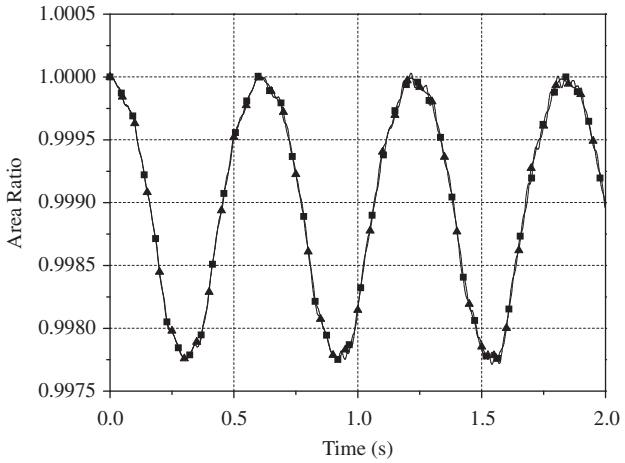

Fig. 4. Area ratio of beam cross-section at the spherical joint obtained using the proposed integration method (– ■ – 6 elements; – ▲ – 8 elements).

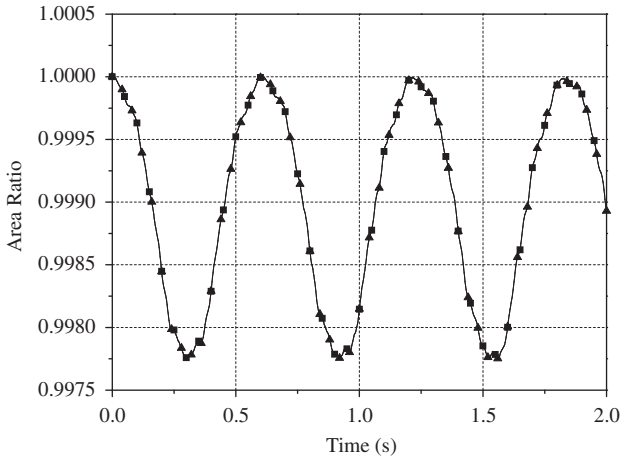

Fig. 5. Area ratio of beam cross-section at the spherical joint (– ■ – proposed method; – ▲ – Adams method).

Furthermore, some of these implicit procedures do not ensure that the constraint equations are satisfied at all levels (position, velocity, and acceleration). The TLISMNI method proposed in this investigation ensures that the constraint equations are satisfied at the position, velocity and acceleration levels. Furthermore, the method does not require the use of

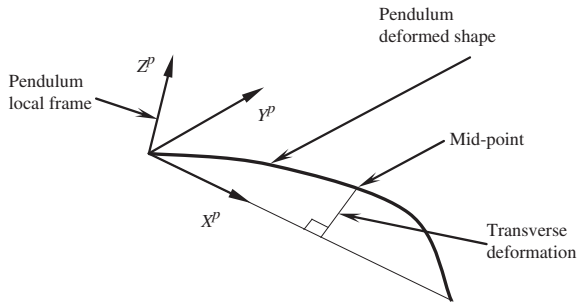

Fig. 6. Mid-point transverse deformation.

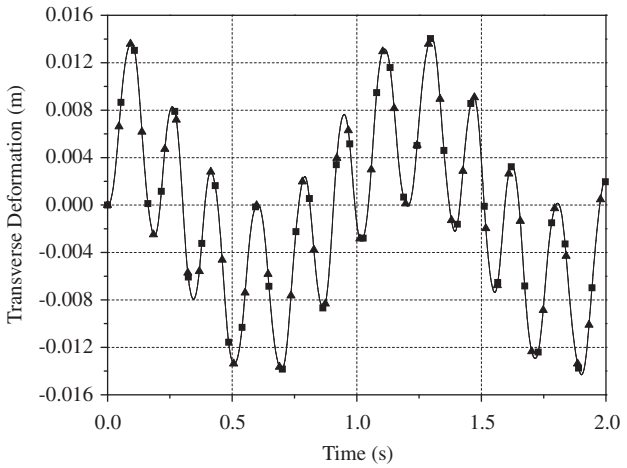

Fig. 7. Mid-point transverse deformation (– ■ – proposed method; – ▲ – Adams method).

the numerical differentiation. By using an iterative two-loop procedure, sparse matrix structure is achieved for all the systems of equations used. This sparse matrix structure requires minimum storage, particularly in problems that involve very flexible bodies modeled using the absolute nodal coordinate formulation [3]. While the inner loop requires the solution of a system of nonlinear equations, the outer loop requires the solution of a linear system of algebraic equation. Numerical results are presented in order to demonstrate the implementation and accuracy of the proposed implicit integration procedure.

Acknowledgments

This research was supported by the US Army Research Office, and by the International Program of the National Science Foundation. This financial support is gratefully acknowledged.

References

[1] I.S. Duff, A.M. Erisman, J.K. Reid, *Direct Method for Sparse Matrices*, Clarendon Press, Oxford, 1986.
[2] K.E. Brenan, S.L. Campbell, L.R. Petzold, *Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations*, North-Holland, New York, 1989.
[3] A.A. Shabana, *Dynamics of Multibody Systems*, third ed., Cambridge University Press, New York, 2005.
[4] H.M. Hilber, T.J.R. Hughes, R.L. Taylor, Improved numerical dissipation for time integration algorithms in structural dynamics, *Earthquake Engineering and Structural Dynamics* 5 (1977) 283–292.
[5] N.M. Newmark, A method of computation for structural dynamics, *Journal of Engineering Mechanics Division, ASCE* (1959) 67–94.
[6] R.A. Wehage, Generalized Coordinate Partitioning in Dynamic Analysis of Mechanical Systems, PhD. Dissertation, University of Iowa, Iowa City, 1980.
[7] A.A. Shabana, *Computational Dynamics*, second ed., Wiley, New York, 2001.
[8] B. Hussein, D. Negrut, A.A. Shabana, Implicit and explicit integration in the solution of the absolute nodal coordinate differential/algebraic equations, *Nonlinear Dynamics* 54 (2008) 283–296.
[9] D. Negrut, R. Rampalli, G. Ottarsson, A. Sajdak, On an implementation of the Hilber–Hughes–Taylor method in the context of index 3 differential-algebraic equations of multibody dynamics, *ASME Journal of Computational and Nonlinear Dynamics* 2 (2007) 73–85.
[10] L.F. Shampine, M.K. Gordon, *Computer Solution of Ordinary Differential Equations*, W.H. Freeman, San Francisco, 1975.
[11] A.A. Shabana, R.Y. Yakoub, Three dimensional absolute nodal coordinate formulation for beam elements: theory, *ASME Journal of Mechanical Design* 123 (2001) 606–613.
[12] A.A. Shabana, *Computational Continuum Mechanics*, Cambridge University Press, New York.