

EXPLICIT MOMENTUM CONSERVING ALGORITHMS FOR RIGID BODY DYNAMICS

G. M. HULBERT

Department of Mechanical Engineering and Applied Mechanics, 321 W. E. Lay Automotive Lab.,
The University of Michigan, Ann Arbor, MI 48109-2121, U.S.A.

(Received 4 June 1991)

Abstract—Two new explicit time integration algorithms are presented for solving the equations of motion of rigid body dynamics that identically preserve angular momentum in the absence of applied torques. This is achieved by expressing the equations of motion in conservation form. Both algorithms also eliminate the need for computing the angular acceleration. The first algorithm employs a one-pass predictor–corrector scheme while the second algorithm is based upon the staggered time integration approach of Park. Numerical results are presented comparing the new algorithms to the algorithms of Simo and Wong and Park *et al.* The predictor–corrector algorithm is shown to suffer weak instabilities while the staggered conserving algorithm exhibits improved performance compared to the staggered algorithm of Park *et al.*

1. INTRODUCTION

The development of time integration schemes for solving equations associated with finite rotations has received considerable attention during the past several years. These research efforts have been motivated by the development of structural models in which proper orthogonal matrices are used to describe the motion; see, e.g., [1–5] as well as the references contained therein. The work of Simo and Wong is particularly interesting in that they present an implicit, single-step, time integration algorithm that identically preserves angular momentum and energy in the absence of applied torques. The basic idea of their approach is to write the equations of motion in conservation form. Integrating the equations over a time step ensures conservation of angular momentum for torque-free motions provided the configuration and angular velocity updates are computed properly. The algorithm obviates the need to compute the angular acceleration; if needed, the acceleration may be obtained by post-processing. An explicit version of the algorithm is presented in their paper that preserves angular momentum but requires the angular acceleration to be computed.

Within the context of linear structural dynamics, Tamma and Namburu developed an explicit algorithm in which the acceleration need not be computed [6]. An extension of their approach to rigid/flexible body dynamics is given in [8]. However, this extended algorithm loses the desirable momentum conservation property achieved by the original version for linear structural dynamics. One purpose of this paper is to present an algorithm that conserves angular momentum using the Tamma–Namburu approach. As described in Sec. 4.3, this requires a one pass predictor–corrector strategy.

Park *et al.* developed an explicit–implicit time integration procedure for multibody dynamics that uses a staggered form of the central difference algorithm coupled with a midpoint update of the configuration [9]. One advantage of their procedure is the implicit treatment of constraint equations; compared to standard central difference methods, twice the number of equations needs to be solved per time step. Their algorithm is not cast in conservation form and thus does not conserve angular momentum. In this paper, an alternate staggered algorithm is presented that inherits the momentum conservation property.

An outline of the paper follows. Since the time integration algorithms for the rotation group may be considered as extensions of algorithms developed for structural dynamics, a review of the classical and staggered central difference algorithms and the Tamma–Namburu algorithm is given in Sec. 2 for linear structural dynamics. Section 3 briefly presents the notation and formulation of rigid body kinematics and dynamics. In Sec. 4, parametrization of the rotation group is discussed, the Simo–Wong algorithm and the proposed momentum conserving predictor–corrector and staggered time integration algorithms are presented and different choices for evaluating configuration-dependent moments are given. Results from numerical simulations are provided in Sec. 5. Conclusions are drawn in Sec. 6.

2. EXPLICIT SECOND-ORDER ACCURATE ALGORITHMS FOR LINEAR STRUCTURAL DYNAMICS

To provide a basis for presenting explicit algorithms for rigid body rotations, we review two explicit, second-order accurate algorithms for solving the equations of linear structural dynamics: classical central differences and the algorithm of Tamma and

Namburu. For the undamped case, the ordinary differential equations for structural dynamics have the form

$$\mathbf{M}\ddot{\mathbf{X}} + \mathbf{K}\mathbf{X} = \mathbf{F}, \tag{1}$$

where \mathbf{M} is the mass matrix (assumed to be a diagonal matrix), \mathbf{K} is the stiffness matrix, \mathbf{F} is the vector of applied loads (a given function of time), \mathbf{X} is the vector of displacement unknowns, and superposed dots indicate differentiation with respect to time. The initial value problem consists of finding a function $\mathbf{X} = \mathbf{X}(t)$ that satisfies (1) for all $t \in [0, T]$, $T > 0$, and the initial conditions

$$\mathbf{X}(0) = \mathbf{d}_0 \tag{2}$$

$$\dot{\mathbf{X}}(0) = \mathbf{v}_0, \tag{3}$$

where \mathbf{d}_0 and \mathbf{v}_0 are given vectors of initial displacements and velocities, respectively. In what follows, \mathbf{d}_n , \mathbf{v}_n and \mathbf{a}_n denote corresponding approximations to $\mathbf{X}(t_n)$, $\dot{\mathbf{X}}(t_n)$ and $\ddot{\mathbf{X}}(t_n)$.

2.1. Central difference algorithm

The equations of motion, (1), are replaced by their discrete approximation at time t_{n+1} as follows:

$$\mathbf{M}\mathbf{a}_{n+1} + \mathbf{K}\mathbf{d}_{n+1} = \mathbf{F}_{n+1}, \tag{4}$$

where $\mathbf{F}_{n+1} = \mathbf{F}(t_{n+1})$ is prescribed. Given the values of \mathbf{d}_n , \mathbf{v}_n and \mathbf{a}_n , \mathbf{d}_{n+1} is obtained using

$$\mathbf{d}_{n+1} = \mathbf{d}_n + \Delta t \mathbf{v}_n + \frac{1}{2} \Delta t^2 \mathbf{a}_n, \tag{5}$$

where $\Delta t = t_{n+1} - t_n$ is the time step. After solving (4) for \mathbf{a}_{n+1} , \mathbf{v}_{n+1} is computed using

$$\mathbf{v}_{n+1} = \mathbf{v}_n + \frac{1}{2} \Delta t (\mathbf{a}_n + \mathbf{a}_{n+1}). \tag{6}$$

The central difference algorithm is a one-step, three-value algorithm since only the displacement, velocity and acceleration values at time t_n are needed to compute the solution for the next time level, t_{n+1} . As discussed in [10], second-order accuracy necessitates computing \mathbf{a}_0 using

$$\mathbf{M}\mathbf{a}_0 = \mathbf{F}_0 - \mathbf{K}\mathbf{d}_0 \tag{7}$$

for problems in which $\mathbf{F}_0 \neq \mathbf{K}\mathbf{d}_0$.

Often the central difference algorithm is cast in 'staggered' form, particularly for nonlinear problems. In this alternate form, (5) and (6) are replaced by

$$\mathbf{d}_{n+1} = \mathbf{d}_n + \Delta t \mathbf{v}_{n+1/2} \tag{8}$$

$$\mathbf{v}_{n+3/2} = \mathbf{v}_{n+1/2} + \Delta t \mathbf{a}_{n+1}. \tag{9}$$

That is, the velocities are computed at the midpoint of time intervals while displacements, accelerations

(and internal forces) are evaluated at the end of the time intervals. The staggered scheme is useful for nonlinear dynamics since constitutive equations typically are given in rate form. Note that a special starting condition is needed to obtain $\mathbf{v}_{1/2}$; often the motion is assumed to be quiescent at time $t_0 = 0$.

2.2. Tamma-Namburu explicit algorithm

In the algorithm of Tamma and Namburu [6], the equations of motion are written in conservation form, i.e.

$$\mathbf{M}\dot{\mathbf{V}} + \mathbf{K}\mathbf{X} = \mathbf{F}, \tag{10}$$

where $\mathbf{V} = \dot{\mathbf{X}}$. Integrating the expressions on both sides of (10) from t_n to t_{n+1} and rearranging terms yields

$$\mathbf{M}(\mathbf{V}(t_{n+1}) - \mathbf{V}(t_n)) = \int_{t_n}^{t_{n+1}} \mathbf{F} dt - \int_{t_n}^{t_{n+1}} \mathbf{K}\mathbf{X} dt. \tag{11}$$

The integrals on the right-hand side of (11) are approximated using

$$\int_{t_n}^{t_{n+1}} \mathbf{F} dt \approx \Delta t \mathbf{F}_{n+1/2} \tag{12}$$

$$\int_{t_n}^{t_{n+1}} \mathbf{K}\mathbf{X} dt \approx \Delta t \mathbf{K}(\mathbf{d}_n + \frac{1}{2} \Delta t \mathbf{v}_n). \tag{13}$$

In (11), $\mathbf{V}(t_n)$ and $\mathbf{V}(t_{n+1})$ are replaced by their approximations, \mathbf{v}_n and \mathbf{v}_{n+1} , respectively. After solving (11), \mathbf{d}_{n+1} is obtained using

$$\mathbf{d}_{n+1} = \mathbf{d}_n + \frac{1}{2} \Delta t (\mathbf{v}_n + \mathbf{v}_{n+1}). \tag{14}$$

The Tamma-Namburu explicit algorithm is a one-step, two-value method. This algorithm is interesting in that there is no need to compute accelerations; in this sense, the algorithm is a true self-starting method. The spectral properties of the Tamma-Namburu and central difference algorithms are identical, thus the two algorithms have the same rates of convergence and the same maximum stable time step. However, the error engendered by the Tamma-Namburu method is smaller; see [6] for numerical examples.

Remarks

1. The staggered form of the central difference algorithm may be viewed as being based upon writing the equations of motion in conservation form. This may be seen as follows: let $t_{n+1/2} = (t_{n+1} + t_n)/2$. Integrate the expressions on both sides of (10) from $t_{n+1/2}$ to $t_{n+3/2}$ to obtain

$$\mathbf{M}(\mathbf{V}(t_{n+3/2}) - \mathbf{V}(t_{n+1/2})) = \int_{t_{n+1/2}}^{t_{n+3/2}} \mathbf{F} dt - \int_{t_{n+1/2}}^{t_{n+3/2}} \mathbf{K}\mathbf{X} dt. \tag{15}$$

Approximate the integrals on the right-hand side of (15) using the midpoint rule

$$\int_{t_n+1/2}^{t_{n+3/2}} \mathbf{F} dt \approx \Delta t \mathbf{F}_{n+1} \tag{16}$$

$$\int_{t_n+1/2}^{t_{n+3/2}} \mathbf{KX} dt \approx \Delta t \mathbf{Kd}_{n+1}. \tag{17}$$

Replace $\mathbf{V}(t_{n+1/2})$ and $\mathbf{V}(t_{n+3/2})$ by $\mathbf{v}_{n+1/2}$ and $\mathbf{v}_{n+3/2}$, respectively, in (15); using (15)–(17), we obtain

$$\mathbf{M}(\mathbf{v}_{n+3/2} - \mathbf{v}_{n+1/2}) = \Delta t \mathbf{F}_{n+1} - \Delta t \mathbf{Kd}_{n+1}. \tag{18}$$

This equation is equivalent to that obtained by solving (9) for \mathbf{a}_{n+1} and substituting the result into (4).

2. In the Tamma–Namburu scheme, the midpoint rule is used to approximate the integral of the forcing term; the approximation of the integral of the internal force term may be considered to arise from either application of the midpoint or trapezoidal rules; both are equivalent in the linear case. Using the midpoint rule

$$\int_{t_n}^{t_{n+1}} \mathbf{KX} dt \approx \Delta t \mathbf{Kd}_{n+1/2}, \tag{19}$$

where

$$\mathbf{d}_{n+1/2} = \mathbf{d}_n + \frac{1}{2} \Delta t \mathbf{v}_n. \tag{20}$$

Using the trapezoidal rule

$$\int_{t_n}^{t_{n+1}} \mathbf{KX} dt \approx \frac{1}{2} \Delta t \mathbf{K}(\mathbf{d}_n + \bar{\mathbf{d}}_{n+1}), \tag{21}$$

where

$$\bar{\mathbf{d}}_{n+1} = \mathbf{d}_n + \Delta t \mathbf{v}_n. \tag{22}$$

Equally valid is the use of the trapezoidal rule to approximate the integral of the forcing term.

3. The conservation form of the equations of motion used by Tamma and Namburu, i.e. (11)–(13), was first presented in [11] with $\theta = 1/2$, $\alpha = 0$. The success of the Tamma–Namburu method is due to adding the displacement corrector equation, (14).

3. RIGID BODY DYNAMICS AND THE ROTATION GROUP

The purpose of this section is to define terminology and briefly summarize the notions of rigid body dynamics; a more complete exposition may be found in Simo and Wong [5] and the references contained therein.

3.1. Rigid body kinematics and dynamics

Consider a rigid body occupying an open, bounded region $\Omega \subset \mathcal{R}^3$; Ω is the reference placement of the

body. Particles of the rigid body are labeled by \mathbf{p} ; the initial position vector of a particle, measured from the origin of an inertial reference frame, $\{\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3\}$, is given by \mathbf{X} . Given the density, ρ_0 , of the rigid body in the reference placement, the inertial reference frame is chosen such that

$$\int_{\Omega} \rho_0 \mathbf{X} d\Omega = \mathbf{0}. \tag{23}$$

For a rigid body, the motion, $\mathbf{x}: \Omega \rightarrow \mathcal{R}^3$, must be of the form

$$\mathbf{x} = \mathbf{x}_c(t) + \mathbf{A}(t)\mathbf{X}, \tag{24}$$

where $\mathbf{A}(t)$ is an orthogonal transformation [$\mathbf{A}(t) \in \text{SO}(3)$] and \mathbf{x}_c defines the position of the center of mass at time t [assuming $\mathbf{x}_c(0) = \mathbf{0}$].

Let $\{\mathbf{b}_1(t), \mathbf{b}_2(t), \mathbf{b}_3(t)\}$ be a set of nonparallel, noncoplanar unit vectors fixed in Ω and attached at the center of mass. This set of vectors is called the body frame. The body frame is time dependent since they move with the body. For notational convenience, let

$$\mathbf{E}_A = \mathbf{b}_A(0), \quad A = 1, 2, 3. \tag{25}$$

From (23)–(25), the orientation of the body frame is obtained using

$$\mathbf{b}_A(t) = \mathbf{A}(t)\mathbf{E}_A, \quad A = 1, 2, 3. \tag{26}$$

The velocity field, with respect to the inertial frame, is given by

$$\mathbf{v}(t) = \dot{\mathbf{x}}_c(t) + \dot{\mathbf{A}}(t)\mathbf{X}. \tag{27}$$

Since $\mathbf{A}(t) \in \text{SO}(3)$, $\mathbf{A}\mathbf{A}^T = \mathbf{1}$, so

$$\dot{\mathbf{A}}(t) = \mathbf{w}(t)\mathbf{A}(t) = \mathbf{A}(t)\mathbf{W}(t), \tag{28}$$

where

$$\mathbf{w}(t) = \dot{\mathbf{A}}(t)\mathbf{A}^T(t) \quad \text{and} \quad \mathbf{W}(t) = \mathbf{A}^T(t)\dot{\mathbf{A}}(t). \tag{29}$$

The matrices $\mathbf{w}(t)$ and $\mathbf{W}(t)$ are in $\text{so}(3)$, the vector space of skew-symmetric matrices. The axial vectors associated with $\mathbf{w}(t)$ and $\mathbf{W}(t)$ are denoted by $\boldsymbol{\omega}(t)$ and $\boldsymbol{\Omega}(t)$ and are called the spatial and convected angular velocities, respectively.

Using (23)–(24) and (27) and (28), the angular momentum vector is given by

$$\begin{aligned} \mathbf{J}(t) &= \int_{\Omega} \rho_0 \mathbf{x} \times \mathbf{v} d\Omega \\ &= \mathbf{x}_c(t) \times \mathbf{p}(t) + \boldsymbol{\pi}(t), \end{aligned} \tag{30}$$

where $\mathbf{p}(t)$ is the total linear momentum and $\boldsymbol{\pi}(t)$ is the total spatial angular momentum relative to the

center of mass. Expressions for $\mathbf{p}(t)$ and $\boldsymbol{\pi}(t)$ are as follows:

$$\mathbf{p}(t) = \int_{\Omega} \rho_0 \, d\Omega \dot{\mathbf{x}}_c(t) = M \dot{\mathbf{x}}_c(t) \quad (31)$$

$$\boldsymbol{\pi}(t) = \mathbf{I}_t \boldsymbol{\omega}(t), \quad (32)$$

where

$$M = \int_{\Omega} \rho_0 \, d\Omega \quad (33)$$

$$\mathbf{I}_t = A(t) \mathbf{J} A^T(t) \quad (34)$$

in which

$$\mathbf{J} = \int_{\Omega} \rho_0 ((\mathbf{X} \cdot \mathbf{X}) \mathbf{1} - \mathbf{X} \otimes \mathbf{X}) \, d\Omega \quad (35)$$

is referred to as the convected inertia dyadic while \mathbf{I}_t is the time dependent spatial inertia dyadic. The convected angular momentum relative to the center of mass is defined by

$$\boldsymbol{\Pi}(t) = A^T(t) \boldsymbol{\pi}(t) = \mathbf{J} \boldsymbol{\Omega}(t). \quad (36)$$

The balance equations of linear and angular momentum are given by

$$\frac{d\mathbf{p}}{dt} = \mathbf{n} \quad (37)$$

and

$$\frac{d\boldsymbol{\pi}}{dt} = \mathbf{m}, \quad (38)$$

where $\mathbf{n}(t)$ and $\mathbf{m}(t)$ are the applied force at the center of mass and the torque, respectively. We shall restrict our attention to solving the angular momentum balance equations. To complete the statement of the initial-value problem, initial conditions are prescribed having the form

$$A(0) = \mathbf{1}, \quad \boldsymbol{\omega}(0) = \boldsymbol{\omega}_0. \quad (39)$$

The vectors \mathbf{E}_A must also be prescribed.

4. EXPLICIT ANGULAR MOMENTUM CONSERVING ALGORITHMS

In this section, the principal difficulty with integrating the rotation group is discussed; parametrization

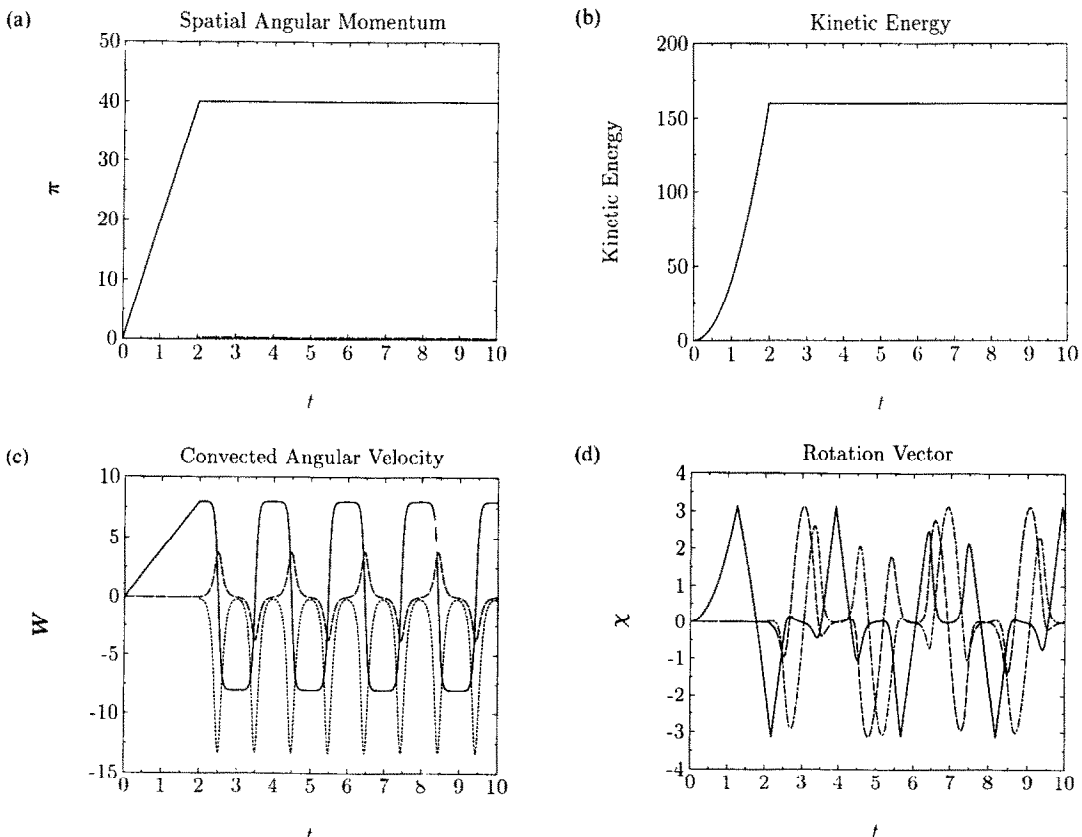


Fig. 1. Unstable motion about intermediate moment of inertia axis. Simo-Wong algorithm with $\Delta t = 0.001$. (Vector components: — $\equiv \mathbf{e}_1$, - - $\equiv \mathbf{e}_2$, - - - $\equiv \mathbf{e}_3$.)

of the rotation group is reviewed. The explicit momentum conserving algorithm of Simo and Wong is reviewed. A new momentum conserving algorithm, based upon a predictor–corrector version of the Tamma–Namburu method, is presented that obviates the need to compute the angular acceleration. Also, a new staggered central difference algorithm is given that conserves angular momentum. Different treatments of configuration-dependent moments are presented; their performance is evaluated in the numerical results section.

4.1. Parameterization of the rotation group

Standard time integration algorithms used to integrate equations associated with translational motion cannot be applied directly to solve the angular momentum balance equations since SO(3) is not a linear space. Given an incremental rotation vector, the corresponding finite rotation is desired. The finite rotation matrix is obtained in closed-form using the exponential map and the formula of Euler and Rodrigues

$$A = \exp(\hat{\chi}) = \mathbf{1} + \frac{\sin \|\chi\|}{\|\chi\|} \hat{\chi} + \frac{1 - \cos(\|\chi\|)}{\|\chi\|^2} \hat{\chi}^2, \quad (40)$$

where χ is the rotation vector and $\hat{\chi}$ is the corresponding skew-symmetric matrix. (Throughout the paper,

the skew-symmetric matrix corresponding to a given rotation vector is denoted by a superposed ‘hat’.)

Efficient parametrization of the rotation group is performed using the four unit quaternion parameters (q_0, \mathbf{q}) (Euler–Rodrigues parameters [7]) and the formula

$$A = (2q_0^2 - 1)\mathbf{1} + 2q_0\hat{\mathbf{q}} + 2\mathbf{q} \otimes \mathbf{q}. \quad (41)$$

Given the configuration at time t_n , the configuration after applying an incremental rotation, χ , is needed, i.e.

$$A(t_{n+1}) = A(t_n)\exp(\hat{\chi}). \quad (42)$$

Computing the configuration update simply requires quaternion multiplication.

4.2. Explicit momentum conserving algorithm of Simo and Wong

An essential feature of momentum conserving algorithms is to integrate (38) over the time interval $[t_n, t_{n+1}]$

$$\pi_{n+1} - \pi_n = \int_{t_n}^{t_{n+1}} m \, dt. \quad (43)$$

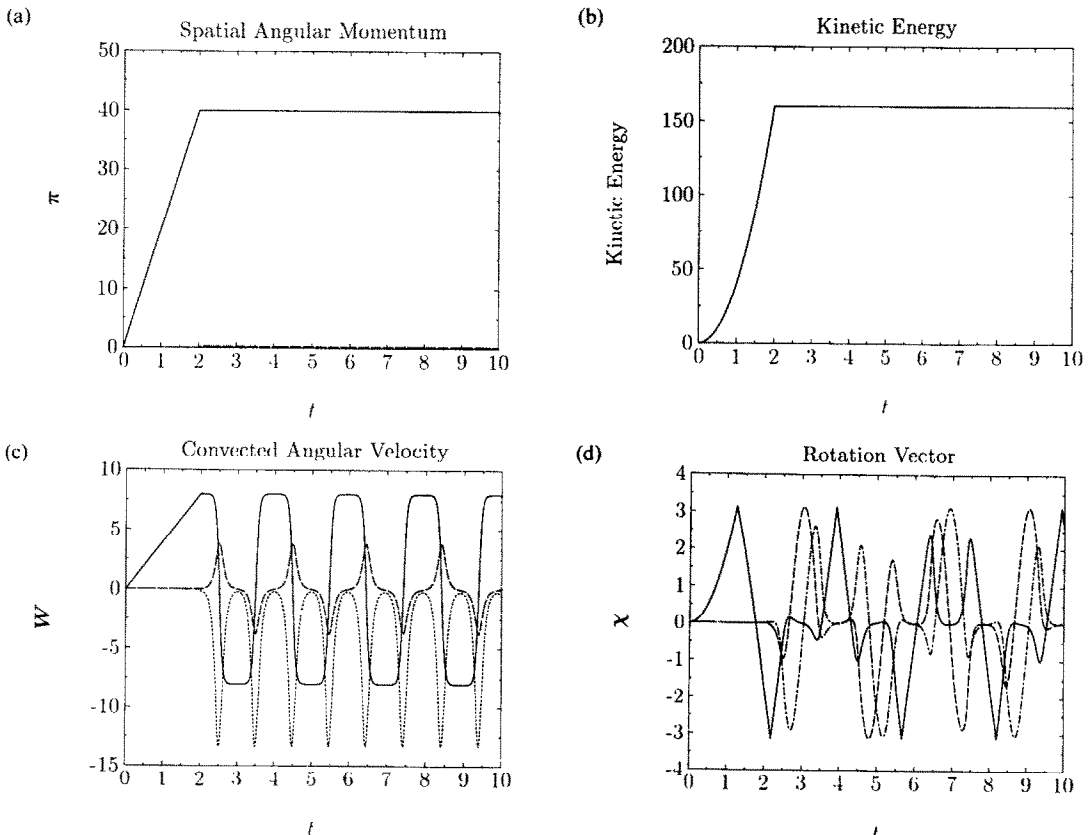


Fig. 2. Unstable motion about intermediate moment of inertia axis. Predictor–corrector algorithm with $\Delta t = 0.001$. (Vector components: — $\equiv e_1$, - - - $\equiv e_2$, - - - - $\equiv e_3$.)

It is important to emphasize the configuration-dependence of π ; the left-hand side of (43) may be written as

$$\pi_{n+1} - \pi_n = A_{n+1} \mathbf{J} \Omega_{n+1} - A_n \mathbf{J} \Omega_n. \quad (44)$$

Let A_n denote the convected angular acceleration vector at time t_n . Then, the explicit algorithm of Simo and Wong proceeds as follows:

1. Given: A_n , Ω_n and A_n . Compute the relative rotation vector, Θ_{n+1} , using

$$\Theta_{n+1} = \Delta t \Omega_n + \frac{1}{2} \Delta t^2 A_n. \quad (45)$$

2. Compute the configuration at time t_{n+1} using the exponential map

$$A_{n+1} = A_n \exp(\Theta_{n+1}). \quad (46)$$

3. Compute the convected angular velocity at time t_{n+1} using (43) and (44)

$$\Omega_{n+1} = \mathbf{J}^{-1} A_{n+1}^T (A_n \mathbf{J} \Omega_n + \Delta t \tilde{\mathbf{m}}_{n+1}), \quad (47)$$

where $\tilde{\mathbf{m}}_{n+1}$ is an appropriate evaluation of the impulse [the right-hand side of (43)]. Impulse evaluation is discussed in Sec. 4.5.

4. Update the convected angular acceleration using

$$A_{n+1} = -A_n + \frac{2}{\Delta t} (-\Omega_n + \Omega_{n+1}). \quad (48)$$

Note that (48) is equivalent to (6) with appropriate substitution of notation. The initial convected angular acceleration may be computed using

$$A_0 = \mathbf{J}^{-1} (A_0^T \mathbf{m}_0 - \Omega_0 \times \mathbf{J} \Omega_0), \quad (49)$$

where \mathbf{m}_0 is the applied initial torque.

4.3. A predictor-corrector explicit momentum conserving algorithm

The algorithm presented in this section achieves for the rotation group what the Tamma-Namburu algorithm attains for linear structural dynamics, namely, convected angular acceleration need not be computed. Boutaghou *et al.* [8] applied an alternative form of the Tamma-Namburu algorithm to planar multibody systems; however, their approach does not conserve momentum. The main difficulty in applying the Tamma-Namburu scheme to the rotation group is the configuration-dependence of the angular

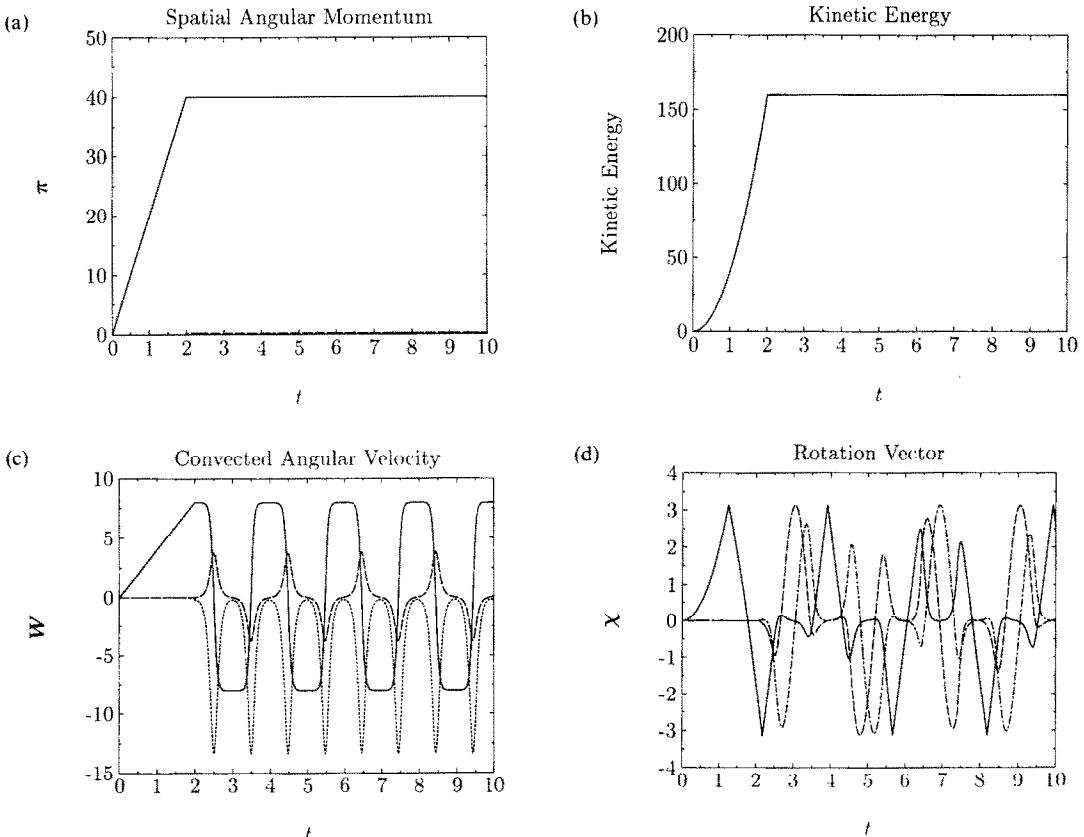


Fig. 3. Unstable motion about intermediate moment of inertia axis. Staggered conservation algorithm with $\Delta t = 0.001$. (Vector components: — $\equiv \mathbf{e}_1$, - - - $\equiv \mathbf{e}_2$, - · - · - $\equiv \mathbf{e}_3$.)

momentum. Angular momentum is conserved, in the case of torque-free motions, using (43) and (44). Thus, the configuration and convected angular velocity must be consistent in the sense that given the configuration A_{n+1} , (43) and (44) are used to compute Ω_{n+1} . As noted in Remark 3 of Sec. 2.2, the displacement update equation, (14), may be viewed as a displacement corrector equation. A similar approach might be employed for the rotation group; i.e., given Ω_n and Ω_{n+1} , the incremental rotation vector and hence the configuration at time t_{n+1} could be obtained using

$$\Theta_{n+1} = \frac{1}{2} \Delta t (\Omega_n + \Omega_{n+1}) \quad (50)$$

$$A_{n+1} = A_n \exp(\Theta_{n+1}). \quad (51)$$

However, if Ω_{n+1} is not updated based upon the configuration computed using (51), then conservation of angular momentum is lost. This problem is eliminated using the following predictor–corrector explicit momentum conserving algorithm:

1. Given: A_n and Ω_n . Compute the predictor relative rotation vector, Θ_{n+1}

$$\Theta_{n+1} = \Delta t \Omega_n. \quad (52)$$

2. Compute the predictor configuration at time t_{n+1} using the exponential map

$$\tilde{A}_{n+1} = \tilde{A}_n \exp(\Theta_{n+1}). \quad (53)$$

3. Compute the predictor convected angular velocity at time t_{n+1} using (43) and (44)

$$\tilde{\Omega}_{n+1} = J^{-1} \tilde{A}_{n+1}^T (A_n J \Omega_n + \Delta t \tilde{m}_{n+1}). \quad (54)$$

4. Compute the corrector relative rotation vector

$$\Theta_{n+1} = \frac{1}{2} \Delta t (\Omega_n + \tilde{\Omega}_{n+1}). \quad (55)$$

5. Compute the corrector configuration at time t_{n+1}

$$A_{n+1} = A_n \exp(\Theta_{n+1}). \quad (56)$$

6. Compute the corrector convected angular velocity at time t_n

$$\Omega_{n+1} = J^{-1} A_{n+1}^T (A_n J \Omega_n + \Delta t \tilde{m}_{n+1}). \quad (57)$$

Compared to the algorithm of Simo and Wong, an additional pass over the quaternion computations is

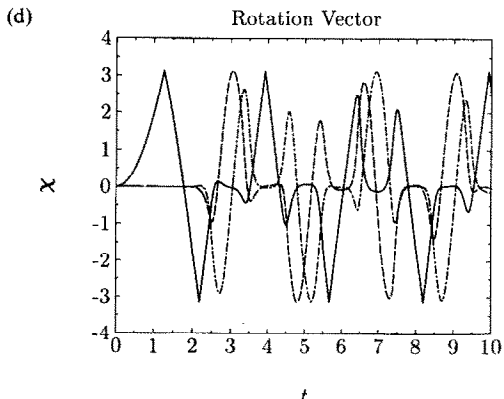
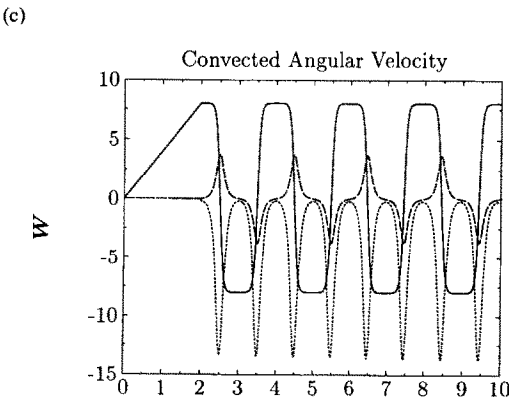
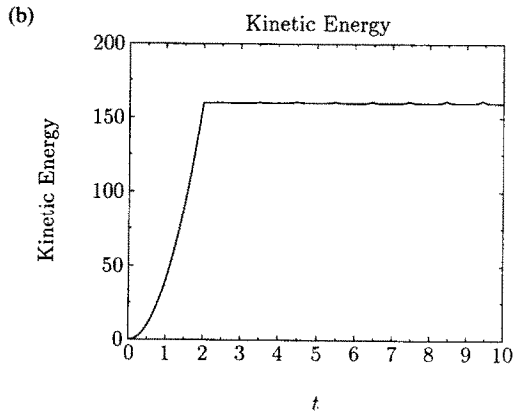
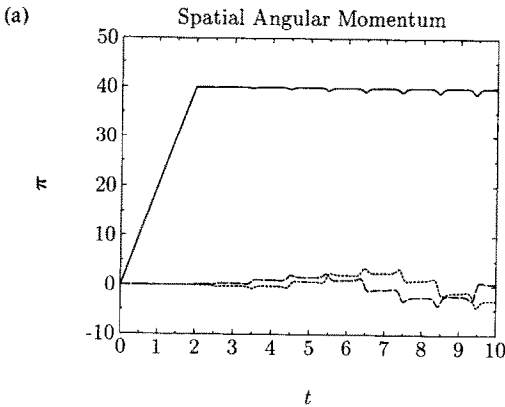


Fig. 4. Unstable motion about intermediate moment of inertia axis. Park–Chiou–Downer algorithm with $\Delta t = 0.001$. (Vector components: — $\equiv e_1$, - - - $\equiv e_2$, ···· $\equiv e_3$.)

required. If the applied moment is not configuration-dependent or if the midpoint rule is used to evaluate the impulse, then $A_n \mathbf{J} \boldsymbol{\Omega}_n + \Delta t \bar{\mathbf{m}}_{n+1}$ needs to be computed only once per time step.

4.4. *A staggered explicit momentum conserving algorithm*

Direct application of the staggered form of the central difference algorithm written in conservation form, (15), to the rotation group yields

$$A_{n+3/2} \mathbf{J} \boldsymbol{\Omega}_{n+3/2} = A_{n+1/2} \mathbf{J} \boldsymbol{\Omega}_{n+1/2} + \int_{t_n+1/2}^{t_{n+3/2}} \mathbf{m} dt. \quad (58)$$

Note that the configuration at time $t_{n+3/2}$ is needed; however, using the basic form of the staggered scheme, the configuration is defined only at the end of time intervals using

$$\boldsymbol{\Theta}_{n+1} = \Delta t \boldsymbol{\Omega}_{n+1/2}. \quad (59)$$

Park *et al.* [9] developed a staggered algorithm for rigid body dynamics that resolves this difficulty. Basically, the equations of motion are solved twice for every time step which enables configurations to be obtained at the end and midpoint of time intervals. However, their approach is not based upon the conservation form of the momentum balance and so is unable to preserve angular momentum during torque-free motions. The following staggered algorithm preserves angular momentum:

1. Given: $A_n, \boldsymbol{\Omega}_n, A_{n+1/2}$ and $\boldsymbol{\Omega}_{n+1/2}$. Compute the relative rotation vector, $\boldsymbol{\Theta}_{n+1}$, using

$$\boldsymbol{\Theta}_{n+1} = \Delta t \boldsymbol{\Omega}_{n+1/2}. \quad (60)$$

2. Compute the configuration at time t_{n+1} using the exponential map:

$$A_{n+1} = A_n \exp(\boldsymbol{\Theta}_{n+1}). \quad (61)$$

3. Compute the convected angular velocity at time t_{n+1} using (43) and (44)

$$\boldsymbol{\Omega}_{n+1} = \mathbf{J}^{-1} A_{n+1}^T (A_n \mathbf{J} \boldsymbol{\Omega}_n + \Delta t \bar{\mathbf{m}}_{n+1}). \quad (62)$$

4. Compute the relative rotation vector, $\boldsymbol{\Theta}_{n+3/2}$, using

$$\boldsymbol{\Theta}_{n+3/2} = \Delta t \boldsymbol{\Omega}_{n+1}. \quad (63)$$

5. Compute the configuration at time $t_{n+3/2}$ using the exponential map

$$A_{n+3/2} = A_{n+1/2} \exp(\boldsymbol{\Theta}_{n+3/2}). \quad (64)$$

6. Compute the convected angular velocity at time $t_{n+3/2}$ using (58) with appropriate evaluation of the impulse.

For $n = 0$, a separate starting procedure is needed to compute $A_{1/2}$ and $\boldsymbol{\Omega}_{1/2}$. A typical procedure is as follows:

- (i) Obtain A_0 using (49).
- (ii) Compute the relative rotation vector, $\boldsymbol{\Theta}_{1/2}$, using

$$\boldsymbol{\Theta}_{1/2} = \frac{1}{2} \Delta t \boldsymbol{\Omega}_0 + \frac{1}{8} \Delta t^2 \mathbf{A}_0. \quad (65)$$

- (iii) Compute the configuration at time $t_{1/2}$ using the exponential map

$$A_{1/2} = A_0 \exp(\boldsymbol{\Theta}_{1/2}). \quad (66)$$

- (iv) Compute the convected angular velocity at time $t_{1/2}$ using

$$\boldsymbol{\Omega}_{1/2} = \mathbf{J}^{-1} A_{1/2}^T \left(A_0 \mathbf{J} \boldsymbol{\Omega}_0 + \int_{t_0}^{t_{1/2}} \mathbf{m} dt \right). \quad (67)$$

With the exception of the starting condition, the angular acceleration need not be computed. The angular acceleration calculation associated with the starting procedure may be eliminated completely by using the predictor-corrector algorithm as the starting procedure with Δt replaced by $1/2 \Delta t$ and $n = 0$.

4.5 *Evaluation of the impulse*

Of principal concern in evaluating time integrals of \mathbf{m} is appropriate treatment of configuration-dependent moments, e.g., the fast, symmetric top described in Sec. 5.2. In this study, the midpoint and trapezoidal rules were employed to approximate time integrals. Using the midpoint rule

$$\int_{t_n}^{t_{n+1}} \mathbf{m} dt \approx \Delta t \mathbf{m}(t_{n+1/2}) = \Delta t \bar{\mathbf{m}}_{n+1}. \quad (68)$$

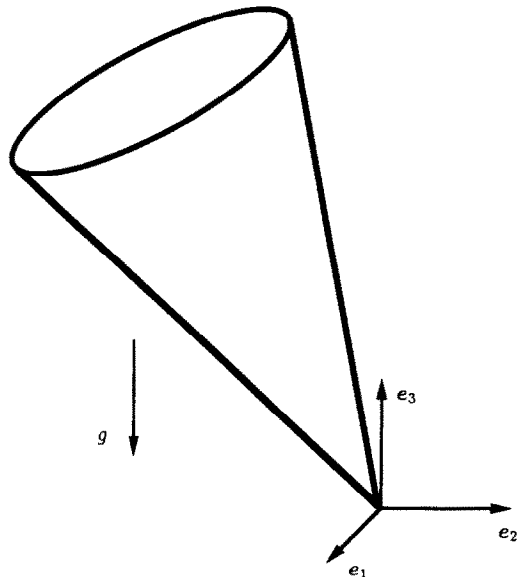


Fig. 5. Fast spinning top with fixed end.

Thus, for configuration-dependent moments, the configuration at $t_{n+1/2}$ is needed. Different formulae were used for the various explicit momentum conserving algorithms.

Simo-Wong algorithm

$$\Theta_{n+1/2} = \frac{1}{2} \Delta t \Omega_n + \frac{1}{8} \Delta t^2 \mathbf{A}_n \quad (69)$$

$$\mathbf{A}_{n+1/2} = \mathbf{A}_n \exp(\hat{\Theta}_{n+1/2}). \quad (70)$$

Predictor-corrector algorithm

$$\Theta_{n+1/2} = \frac{1}{2} \Delta t \Omega_n \quad (71)$$

$$\mathbf{A}_{n+1/2} = \mathbf{A}_n \exp(\hat{\Theta}_{n+1/2}). \quad (72)$$

Recall that the angular momentum balance equation needs to be solved twice when using the predictor-corrector algorithm [see (54), (57)]. Employing the midpoint rule requires only one evaluation of the residual, $\mathbf{A}_n, \mathbf{J}\Omega_n + \Delta t \dot{\mathbf{m}}_{n+1}$, per time step since none of the quantities is a function of the solution at time t_{n+1} .

Staggered algorithm

In addition to (68), approximations are needed for integrals appearing in (58) and (67). Using the midpoint rule

$$\int_{t_n}^{t_{n+3/2}} \mathbf{m} dt \approx \Delta t \mathbf{m}(t_{n+1}) \quad (73)$$

$$\int_{t_0}^{t_{1/2}} \mathbf{m} dt \approx \frac{1}{2} \Delta t \mathbf{m}(t_{1/4}), \quad (74)$$

where $t_{1/4} = t_1/4$. For (73), the configuration at t_{n+1} was computed in (61), thus no additional calculations are needed. For (74)

$$\Theta_{1/4} = \frac{1}{4} \Delta t \Omega_0 + \frac{1}{32} \Delta t^2 \mathbf{A}_0 \quad (75)$$

$$\mathbf{A}_{1/4} = \mathbf{A}_0 \exp(\hat{\Theta}_{1/4}). \quad (76)$$

Using the trapezoidal rule

$$\int_{t_n}^{t_{n+1}} \mathbf{m} dt \approx \frac{1}{2} \Delta t (\mathbf{m}(t_n) + \mathbf{m}(t_{n+1})) = \Delta t \bar{\mathbf{m}}_{n+1}. \quad (77)$$

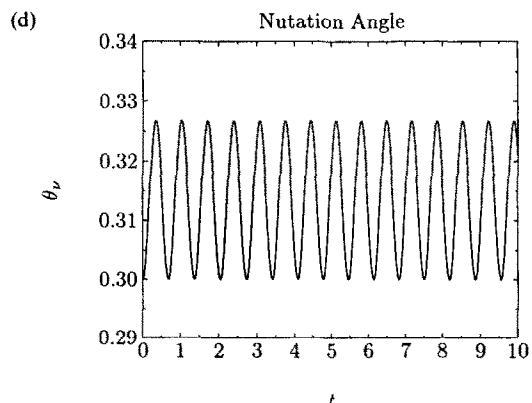
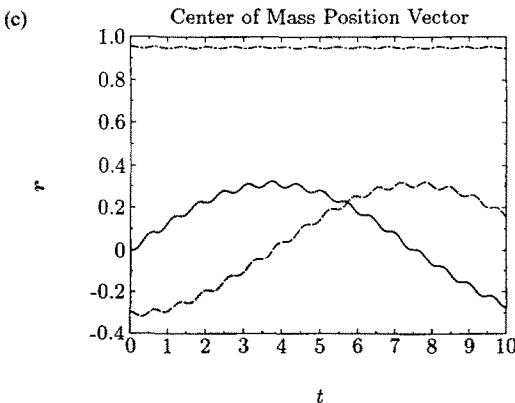
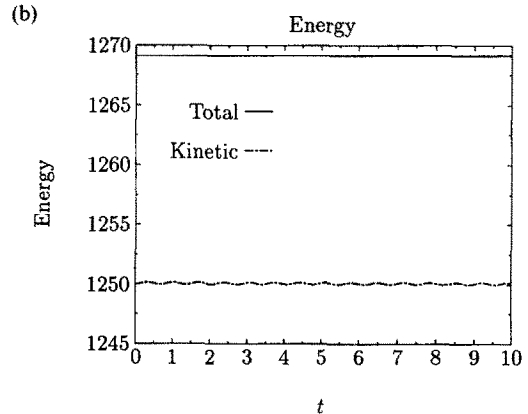
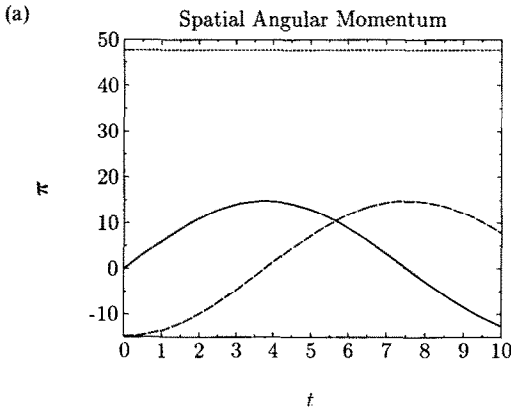


Fig. 6. Fast spinning top with fixed end. Staggered conservation algorithm with $\Delta t = 0.01$. (Vector components: — $\equiv e_1$, - - - $\equiv e_2$, - · - · - $\equiv e_3$.)

The configuration at time t_{n+1} is obtained using (46) for the Simo–Wong algorithm, (53) and (56) for the predictor–corrector algorithm and (60), (63) and (66) for the staggered algorithm. For the predictor–corrector algorithm, a corrector residual must be computed when configuration-dependent impulses are approximated using the trapezoidal rule.

5. NUMERICAL RESULTS

Results from numerical simulations are presented in this section to compare the performance of the explicit momentum conserving algorithms. Also, evaluations are given of the two choices for approximating the impulse for the case of configuration-dependent moments. These example problems are taken from [5].

5.1. Unstable rotation about the intermediate moment of inertia axis

Rotation about the axis of intermediate moment of inertia results in unstable motion if a small disturbance is introduced. The problem formulation is

1. A constant torque about the axis of intermediate moment of inertia is applied to a rigid body that is initially at rest.

2. At time $t = t_d$, the constant torque is removed and a constant disturbance torque is applied about an axis perpendicular to that of the intermediate moment of inertia. This disturbance torque is applied for a duration of Δt after which the body remained in a torque-free state.

The convected inertia dyadic, with respect to an inertial basis $(\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3)$, moment history and initial conditions are

$$[\mathbf{J}] = \begin{bmatrix} 5 & 0 & 0 \\ 0 & 10 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{m}(t) = \begin{cases} 20\mathbf{e}_1, & \text{if } 0 \leq t \leq t_d \\ (5 \Delta t)^{-1}\mathbf{e}_2, & \text{if } t_d \leq t \leq t_d + \Delta t \\ \mathbf{0}, & \text{if } t > t_d + \Delta t \end{cases}$$

$$\boldsymbol{\chi}(0) = \mathbf{0}, \quad \boldsymbol{\Omega}(0) = \mathbf{0},$$

where $t_d = 2 - \Delta t$.

Results obtained from the Simo–Wong, predictor–corrector, staggered and Park–Chiou–Downer algorithms, with $\Delta t = 0.001$, are shown in Figs 1–4, respectively. By design, the first three algorithms

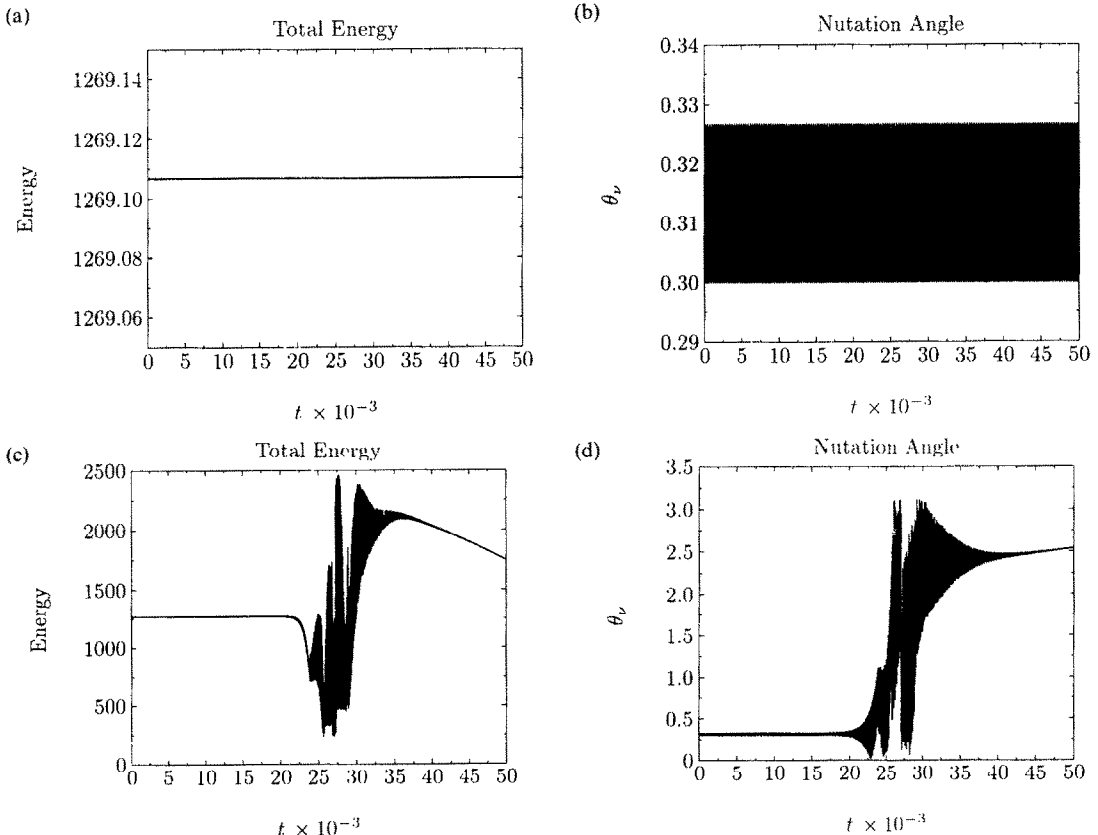


Fig. 7. Fast spinning top with fixed end. Simo–Wong algorithm with $\Delta t = 0.001$. (Top: trapezoidal rule; bottom: midpoint rule.)

identically conserve angular momentum. The results from the Simo–Wong and the staggered conserving algorithms are virtually identical. While the kinetic energy appears constant during the torque-free motion, the predictor–corrector algorithm exhibits a weak growth in energy for long times. The rate of growth is a function of time step size but the growth still appears using smaller time steps. These results imply that the predictor–corrector algorithm is unconditionally unstable. The Park–Chiou–Downer algorithm also exhibits energy growth in the form of ‘bursts’ when the e_3 component of convected angular velocity has large magnitude.

5.2. Fast, symmetric top with fixed tip

A symmetric top with fixed tip is moving in a uniform gravitational field; see Fig. 5. For this problem, the applied torque is configuration dependent. The problem data are

$$[J] = \begin{bmatrix} 5 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

$$W = 20, \quad L_c = 1$$

$$\chi(0) = \begin{Bmatrix} 0.3 \\ 0 \\ 0 \end{Bmatrix},$$

$$\Omega(0) = \begin{Bmatrix} 0 \\ 0 \\ 50 \end{Bmatrix},$$

where J is given in terms of the principal moments of inertia about the center of mass, and the components of $\chi(0)$ and $\Omega(0)$ are relative to the inertial basis $\{e_1, e_2, e_3\}$.

Figure 6 shows results obtained using the staggered conserving algorithm with $\Delta t = 0.001$. For the short time duration shown, the results obtained from the different algorithms are indistinguishable. However, the long-term behavior of the algorithms is clearly different, see Figs 7–10. (The apparent oscillations in the nutation angle are due to the plotting frequency rather than the actual oscillation.) For the Simo–Wong algorithm, instability is observed when the configuration-dependent moment is evaluated using the midpoint rule. Weak instability is present in the predictor–corrector algorithm regardless of the choice of integration rule. While the staggered conserving algorithm is stable for both

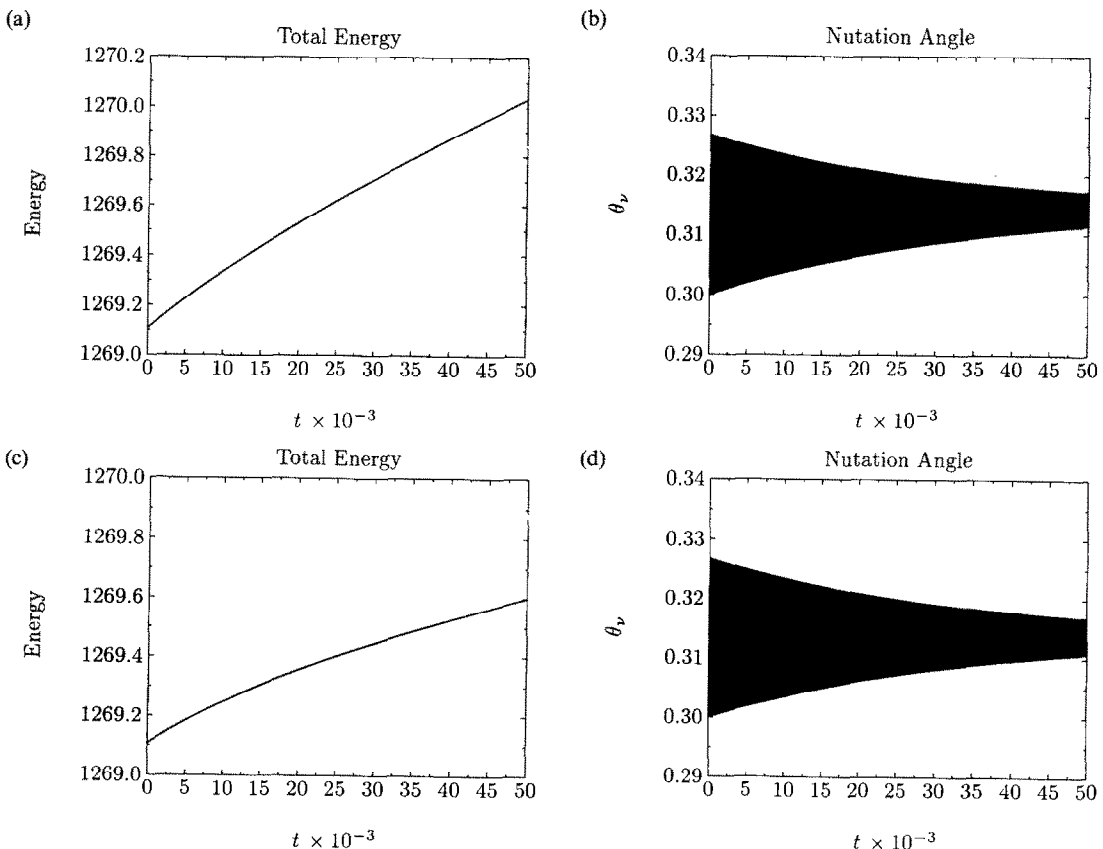


Fig. 8. Fast spinning top with fixed end. Predictor–corrector algorithm with $\Delta t = 0.001$. (Top: trapezoidal rule; bottom: midpoint rule.)

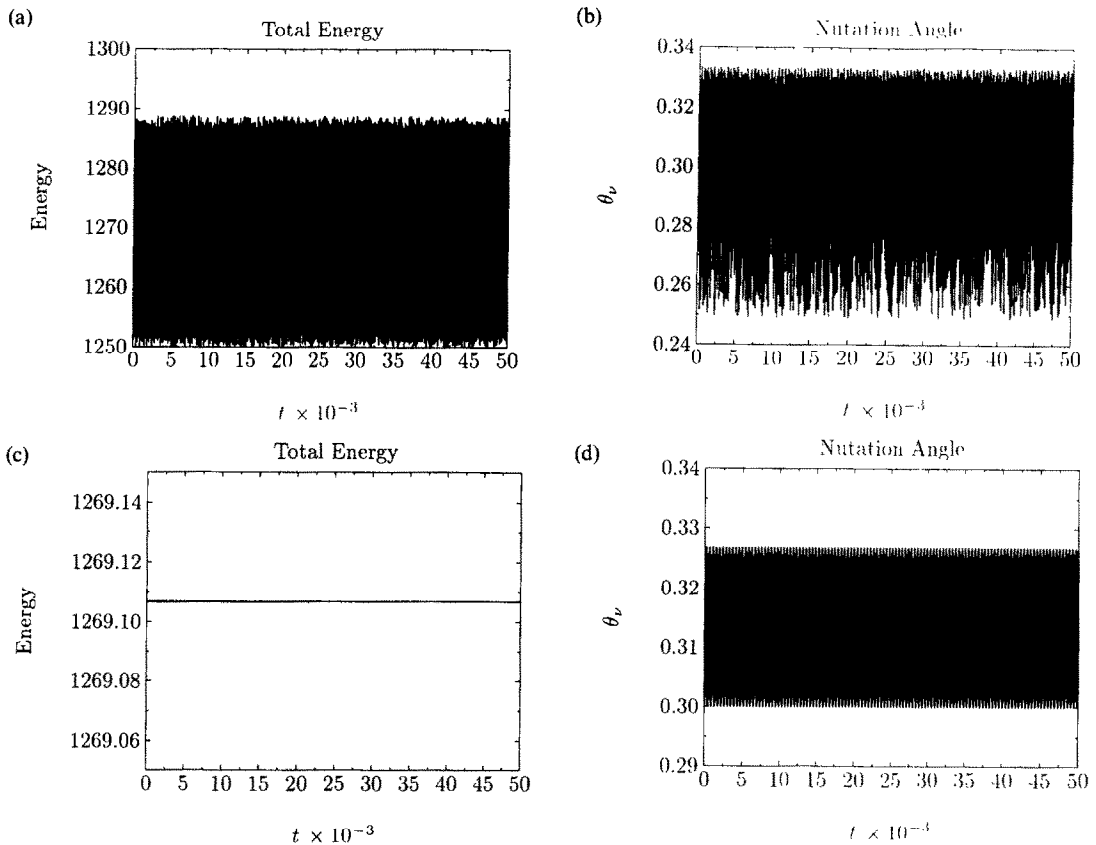


Fig. 9. Fast spinning top with fixed end. Staggered conservation algorithm with $\Delta t = 0.001$. (*Top*: trapezoidal rule; *bottom*: midpoint rule.)

midpoint and trapezoidal rule evaluations, it is apparent from Fig. 9 that the midpoint rule is a superior choice. Finally, in contrast to the first numerical example, the Park–Chiu–Downer algorithm does not exhibit instability for the fast top problem; we note that this algorithm uses the midpoint rule.

6. CONCLUSIONS

Two new explicit algorithms have been presented for integrating the equations of rigid

body dynamics; these algorithms exactly preserve angular momentum in the absence of applied torques. Numerical results show that only the staggered algorithm, as well as the Simo–Wong algorithm, avoid weak instabilities. A potential advantage of the staggered method, compared to the Simo–Wong approach, is the straightforward ability to incorporate constraint equations, in the spirit of Park and colleagues [4, 9, 12]. Such extensions shall be addressed in future publications.

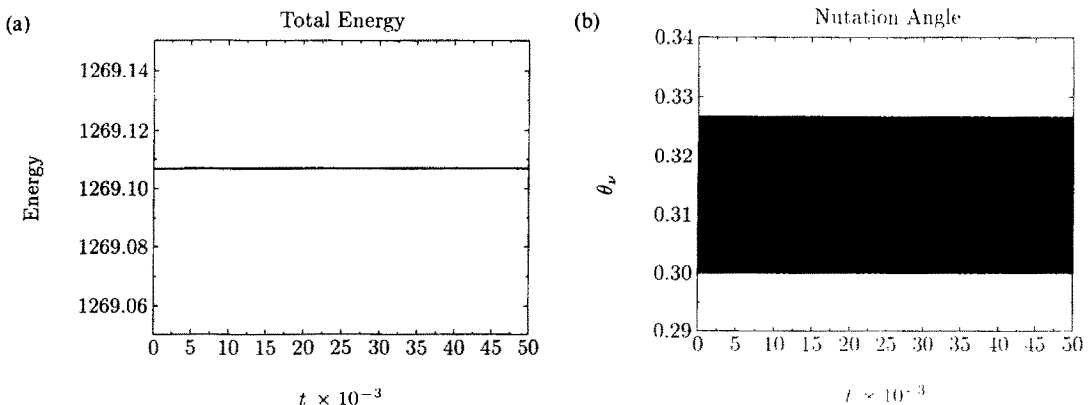


Fig. 10. Fast spinning top with fixed end. Park–Chiu–Downer algorithm with $\Delta t = 0.001$.

REFERENCES

1. M. Geradin, G. Robert and P. Buchet, Kinematic and dynamic analysis of mechanisms. A finite element approach based on Euler parameters. In *Finite Element Methods for Nonlinear Problems* (Edited by P. G. Bergan, K. J. Bathe and W. Wunderlich), pp. 41–60. Springer, New York (1985).
2. M. Geradin and A. Cardona, Kinematics and dynamics of rigid and flexible mechanisms using finite elements and quaternion algebra. *Comp. Mech.* **4**, 115–135 (1989).
3. M. Iura and S. N. Atluri, On a consistent theory and variational formulation of finitely stretched and rotated 3-D space-curved beams. *Comp. Mech.* **4**, 73–88 (1989).
4. K. C. Park, J. C. Chiou, J. D. Downer, C. Farhat, G. S. Chen and B. K. Wada, Dynamics of three-dimensional space crane: motion requirements and computational considerations. *90-WA/Aero-7*. ASME, New York (1990).
5. J. C. Simo and K. K. Wong, Unconditionally stable algorithms for the orthogonal group that exactly preserve energy and momentum. *Int. J. Numer. Meth. Engng* **31**, 19–52 (1991).
6. K. K. Tamma and R. R. Namburu, A robust self-starting explicit computational methodology for structural dynamic applications: architecture and representations. *Int. J. Numer. Meth. Engng* **29**, 1441–1454 (1990).
7. H. Cheng and K. C. Gupta, An historical note on finite rotations. *ASME J. Appl. Mech.* **56**, 139–145 (1989).
8. Z.-E. Boutaghoul, K. K. Tamma and A. G. Erdman, Continuous/discrete modeling and analysis of elastic planar multibody systems. *Comput. Struct.* **38**, 605–613 (1991).
9. K. C. Park, J. C. Chiou and J. D. Downer, Explicit-implicit staggered procedure for multibody dynamics analysis. *AIAA Jnl Guidance Control* **13**, 562–570 (1990).
10. G. M. Hulbert and T. J. R. Hughes, An error analysis of truncated starting conditions in step-by-step time integration: consequences for structural dynamics. *Earthquake Engng Struct. Dyn.* **15**, 901–910 (1987).
11. O. C. Zienkiewicz, W. L. Wood and R. L. Taylor, An alternative single-step algorithm for structural dynamics. *Earthquake Engng Struct. Dyn.* **8**, 31–40 (1980).
12. K. C. Park and J. C. Chiou, Stabilization of computational procedures for constrained dynamical systems. *AIAA Jnl Guidance Control* **11**, 365–370 (1988).