

A Discussion of Low-Order Numerical Integration Formulas for Rigid and Flexible Multibody Dynamics

Dan Negrut¹

Department of Mechanical Engineering,
University of Wisconsin,
Madison, WI 53706
e-mail: negrut@engr.wisc.edu

Laurent O. Jay

Department of Mathematics,
University of Iowa,
14 MacLean Hall,
Iowa-City, IA 52242
e-mail: ljay@math.uiowa.edu

Naresh Khude

Department of Mechanical Engineering,
University of Wisconsin,
Madison, WI 53706
e-mail: khude@engr.wisc.edu

The premise of this work is that the presence of high stiffness and/or frictional contact/impact phenomena limits the effective use of high order integration formulas when numerically investigating the time evolution of real-life mechanical systems. Producing a numerical solution relies most often on low-order integration formulas of which the paper investigates three alternatives: Newmark, HHT, and order 2 BDFs. Using these methods, a first set of three algorithms is obtained as the outcome of a direct index-3 discretization approach that considers the equations of motion of a multibody system along with the position kinematic constraints. The second batch of three algorithms draws on the HHT and BDF integration formulas and considers, in addition to the equations of motion, both the position and velocity kinematic constraint equations. Numerical experiments are carried out to compare the algorithms in terms of several metrics: (a) order of convergence, (b) energy preservation, (c) velocity kinematic constraint drift, and (d) efficiency. The numerical experiments draw on a set of three mechanical systems: a rigid slider-crank, a slider-crank with a flexible body, and a seven body mechanism. The algorithms investigated show good performance in relation to the asymptotic behavior of the integration error and, with one exception, result in comparable CPU simulation times with a small premium being paid for enforcing the velocity kinematic constraints. [DOI: 10.1115/1.3079784]

1 Introduction

A multitude of phenomena, processes, and applications are described in terms of mixed systems of differential equations combined with linear and nonlinear algebraic equations, most often corresponding to models coming from engineering, physics, and chemistry. Differential equations relate certain quantities to their derivatives with respect to time and/or space variables. Algebraic equations usually model conservation laws and constraints present in the system. When there are derivatives with respect to only one independent variable (usually time), the equations are called differential-algebraic equations (DAEs). DAEs are basically differential equations defined on submanifolds of \mathbb{R}^n . For the dynamics of multibody systems, the constrained equations of motion can be expressed in the form (see, for instance, Refs. [1,2])

$$\begin{aligned} \dot{\mathbf{q}} &= \mathbf{v} \\ \mathbf{M}(\mathbf{q})\dot{\mathbf{v}} &= \mathbf{Q}(t, \mathbf{q}, \mathbf{v}, \lambda, \mu, \mathbf{u}(t)) - \Phi_{\mathbf{q}}^T(\mathbf{q}, t)\lambda - \Gamma_{\mathbf{v}}^T(\mathbf{v}, \mathbf{q}, t)\mu \\ \mathbf{0} &= \Phi(\mathbf{q}, t) \\ \mathbf{0} &= \Gamma(\mathbf{v}, \mathbf{q}, t) \end{aligned} \quad (1)$$

where $\mathbf{q} \in \mathbb{R}^n$ are the generalized coordinates, $\mathbf{v} \in \mathbb{R}^n$ are the generalized velocities, $\lambda \in \mathbb{R}^m$ and $\mu \in \mathbb{R}^p$ are the Lagrange multipliers, and $\mathbf{u}: \mathbb{R} \rightarrow \mathbb{R}^c$ represent time dependent external dynamics, e.g., control variables. The matrix $\mathbf{M}(\mathbf{q})$ is the generalized mass matrix, $\mathbf{Q}(t, \mathbf{q}, \mathbf{v}, \lambda, \mu, \mathbf{u}(t))$ represents the vector of generalized

applied forces, $\Phi(\mathbf{q}, t)$ is the set of m holonomic constraints, i.e., position-level kinematic constraints, and $\Gamma(\mathbf{v}, \mathbf{q}, t)$ is the set of p nonholonomic constraints, i.e., velocity-level kinematic constraints [3,4,1]. Differentiating the kinematic constraints with respect to time leads to the additional equations

$$\begin{aligned} \mathbf{0} &= \Phi_{\mathbf{q}}(\mathbf{q}, t)\mathbf{v} + \Phi_t(\mathbf{q}, t) \\ \mathbf{0} &= \Phi_{\mathbf{q}}(\mathbf{q}, t)\dot{\mathbf{v}} + (\Phi_{\mathbf{q}}(\mathbf{q}, t)\mathbf{v})_{\mathbf{q}}\mathbf{v} + 2\Phi_{\mathbf{q}t}(\mathbf{q}, t)\mathbf{v} + \Phi_{tt}(\mathbf{q}, t) \\ \mathbf{0} &= \Gamma_{\mathbf{v}}(\mathbf{v}, \mathbf{q}, t)\dot{\mathbf{v}} + \Gamma_{\mathbf{q}}(\mathbf{v}, \mathbf{q}, t)\mathbf{v} + \Gamma_t(\mathbf{v}, \mathbf{q}, t) \end{aligned} \quad (2)$$

Equations (1) and (2) form an overdetermined system of DAEs, having strictly more equations than variables. The ability to solve such systems is relevant for several classes of applications such as multibody dynamics and molecular dynamics.

When finding the solution of Eqs. (1) and (2), most of the numerical solvers currently used share some or all of the following drawbacks: numerical drift that occurs when the solution does not stay on the manifold of constraints at the position and/or velocity levels and as such might become nonphysical, inability to deal efficiently with stiffness, loss of underlying properties of the exact flow and trajectories, no preservation of invariants such as energy, introduction of undesired numerical damping, and the reduction in convergence order when solving stiff problems that arise often in applications. Whereas techniques for the numerical solution of ordinary differential equations (ODEs) go back more than three centuries and are well established, the numerical solution of DAEs has a comparatively short history [5–7]. The first class of numerical techniques eventually applied to DAEs was published in Ref. [8] for the solution of ODEs. Since then DAEs have widely penetrated the numerical analysis, engineering, and scientific computing communities and are increasingly encountered in practical applications. Still, numerically solving DAEs poses fundamental difficulties not encountered when solving ODEs. Specialized numerical techniques have been developed,

¹Corresponding author.

Contributed by the Design Engineering Division of ASME for publication in the JOURNAL OF COMPUTATIONAL AND NONLINEAR DYNAMICS. Manuscript received December 31, 2007; final manuscript received July 30, 2008; published online March 9, 2009. Review conducted by Aki Mikkola. Paper presented at the ASME 2007 Design Engineering Technical Conferences and Computers and Information in Engineering Conference (DETC2007), Las Vegas, NV, September 4–7, 2007.

typically belonging to one of the two classes: state-space methods or direct methods. For a recent review of this topic, the reader is referred to Ref. [9].

State-space methods first reduce the DAEs to a smaller dimension ODE problem, thus benefiting from the extensive body of knowledge associated with ODE solvers. Specifically, the DAEs induce differential equations on the constraint manifold [10], which can be reduced on a subspace of the n -dimensional Euclidean space. The resulting state-space ODEs (SSODEs) are integrated using classical numerical integration formulas. The one-to-one local mapping from the manifold to the subspace of independent coordinates is then used to determine the point on the manifold corresponding to the solution of the SSODEs. This framework formalizes the theory of numerical solution of DAEs using the language of differential manifolds [11]. Practical algorithms drawing on this class of methods are presented in Refs. [12,13,10,14]. The main factor that differentiates these algorithms is the choice of manifold parametrization.

State-space methods have been subjected to criticism in two aspects. First, the choice of parametrization generally is not global. Second, poor choices of the projection space result in SSODEs that are numerically demanding, mainly at the expense of overall efficiency and robustness of the algorithm [15]. Although the theoretical framework for these methods was outlined several years ago [16,10], it was only relatively recently that implicit numerical integration methods for DAEs have been proposed in the context of SSODEs for multibody dynamics analysis [17,18]. The major intrinsic drawback associated with state-space methods remains the expensive DAE to ODE reduction process that is further exacerbated in the context of implicit integration, which is the norm in industry applications.

Alternatively, direct methods discretize the constrained equations of motion in Eq. (1), possibly after reducing the index of the DAEs by considering some or all of the kinematic constraint equations in Eq. (2). Original contributions in this direction are found in Refs. [19,20,5,21–26]. When dealing with systems that include flexible substructures and bodies, numerical methods have been sought that are capable of introducing controllable numerical dissipation to damp out spurious high frequencies, an artifact of the spatial discretization, without affecting the low frequency modes of the system and the accuracy of the method [27,28]. Several methods have been proposed for structural dynamics simulation, such as the HHT method (also called α -method) [29] and the generalized α -method [30]. These are order 2 methods proposed in conjunction with the ODE problems. For DAEs stemming from multibody dynamics analysis, several α -type algorithms have been reported in the literature [31,32]. A thorough discussion of theoretical and implementation aspects related to a HHT-based numerical integrator for the simulation of large mechanical systems with flexible bodies and penalty-based contact/impact can be found in Ref. [33], while a convergence analysis of the generalized- α method has been provided in Ref. [34]. However, until recently there has been no HHT type method that also stabilized the solution on the velocity constraint manifold, an attribute that is important in mechatronics applications and in dealing with joint friction/contact models. Two of the six algorithms considered in this study address this issue of velocity constraint stabilization and they draw on work presented in Refs. [35,36].

This paper is organized as follows: First, the six algorithms investigated in this study are introduced. For each algorithm, a short overview of existing convergence results is provided along with the expression of the Jacobian associated with the nonlinear discretization system. The emphasis is on HHT-SI2, a new variable-damping stabilized overdetermined index-2 algorithm whose convergence analysis is upcoming [37]. Next, a set of numerical experiments drawing on three mechanical systems compares the algorithms in terms of several metrics: global integration error, energy preservation, velocity constraint violation, and efficiency. A set of brief remarks concludes the paper.

2 Low-Order Integration Algorithms

The first integration method considered in this study is essentially the BDF method of order 2 proposed in Ref. [8], and it serves the purpose of providing a reference when comparing the performance of the other algorithms. The second-order BDF formula is cast into a form suitable for direct numerical integration of second-order differential equations:

$$\begin{aligned}\mathbf{q}_{n+1} &= \frac{4}{3}\mathbf{q}_n - \frac{1}{3}\mathbf{q}_{n-1} + h\left(\frac{8}{9}\dot{\mathbf{q}}_n - \frac{2}{9}\dot{\mathbf{q}}_{n-1}\right) + \frac{4}{9}h^2\ddot{\mathbf{q}}_{n+1} \\ \dot{\mathbf{q}}_{n+1} &= \frac{4}{3}\dot{\mathbf{q}}_n - \frac{1}{3}\dot{\mathbf{q}}_{n-1} + \frac{2}{3}h\ddot{\mathbf{q}}_{n+1}\end{aligned}\quad (3)$$

These formulas, used in conjunction with the equations of motion and position kinematic constraint equations, lead to a second-order method herein called NSTIFF:

$$\mathbf{M}(\mathbf{q}_{n+1})\ddot{\mathbf{q}}_{n+1} + (\Phi_{\mathbf{q}}^T\lambda)_{n+1} - \mathbf{Q}_{n+1} = \mathbf{0}\quad (4)$$

$$\frac{9}{4h^2}\Phi(\mathbf{q}_{n+1}, t_{n+1}) = \mathbf{0}$$

As suggested in Ref. [33] and recently analyzed in Ref. [38], the scaling of the kinematic constraint equations by the inverse of the integration step-size h^2 is done in order to prevent an ill conditioning of the Jacobian $\mathbf{J}_{\text{NSTIFF}}$ associated with the Newton-type method employed to solve Eq. (4), which is regarded as a nonlinear system in $\dot{\mathbf{q}}_{n+1}$ and λ_{n+1} :

$$\mathbf{J}_{\text{NSTIFF}} = \begin{bmatrix} \mathbf{M} + \mathbf{P} & \Phi_{\mathbf{q}}^T \\ \Phi_{\mathbf{q}} & \mathbf{0} \end{bmatrix}$$

where $\mathbf{P} = \frac{4}{9}h^2(\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + (\Phi_{\mathbf{q}}^T\lambda) - \mathbf{Q})_{\mathbf{q}} - \frac{2}{3}h\mathbf{Q}_{\mathbf{q}}$. Note that when $h \rightarrow 0$ the condition number of $\mathbf{J}_{\text{NSTIFF}}$ remains bounded. The scaling of the position constraint equation by $9/4h^2$ leads to a bounded value. To see this, first, note that for all the numerical integration formulas considered herein, locally, $\|\mathbf{q}_{n+1} - \tilde{\mathbf{q}}_{n+1}\| = \mathcal{O}(h^2)$, where $\tilde{\mathbf{q}}_{n+1}$ is the exact solution and \mathbf{q}_{n+1} is an approximation obtained after taking an integration step. Then,

$$\Phi(\mathbf{q}, t) = \Phi(\tilde{\mathbf{q}}, t) + \Phi_{\mathbf{q}}(\tilde{\mathbf{q}}, t)(\mathbf{q} - \tilde{\mathbf{q}}) + \dots = \Phi_{\mathbf{q}}(\tilde{\mathbf{q}}, t)(\mathbf{q} - \tilde{\mathbf{q}}) + \dots$$

where the subscript $n+1$ on \mathbf{q} , $\tilde{\mathbf{q}}$, and t was dropped for convenience. It follows that $9/4h^2\Phi(\mathbf{q}_{n+1}, t_{n+1})$ is $\mathcal{O}(h^0)$, which justifies the scaling proposed in Eq. (4).

The second numerical integration method considered uses the Newmark formulas [39]. It requires the selection of two parameters $\gamma \geq 1/2$ and $\beta \geq (\gamma + 1/2)^2/4$ based on which, given the acceleration $\ddot{\mathbf{q}}_{n+1}$ at the new time step t_{n+1} , the new position and velocity are obtained as

$$\begin{aligned}\mathbf{q}_{n+1} &= \mathbf{q}_n + h\dot{\mathbf{q}}_n + \frac{h^2}{2}[(1 - 2\beta)\ddot{\mathbf{q}}_n + 2\beta\ddot{\mathbf{q}}_{n+1}] \\ \dot{\mathbf{q}}_{n+1} &= \dot{\mathbf{q}}_n + h[(1 - \gamma)\ddot{\mathbf{q}}_n + \gamma\ddot{\mathbf{q}}_{n+1}]\end{aligned}\quad (5)$$

Given an integration step-size h , the discretization scheme operates on the equations of motion and position kinematic constraint equations to lead to the nonlinear system:

$$(\mathbf{M}\ddot{\mathbf{q}})_{n+1} + (\Phi_{\mathbf{q}}^T\lambda)_{n+1} = \mathbf{Q}_{n+1}\quad (6)$$

$$\frac{1}{\beta h^2}\Phi(\mathbf{q}_{n+1}, t_{n+1}) = \mathbf{0}\quad (7)$$

The method, called hereafter Newmark, is order 1 unless $\gamma = 1/2$ and $\beta = 1/4$. This choice leads to the trapezoidal method, which is known in the literature to have stability problems when used in conjunction with index-3 DAEs [31]. Note that the Jacobian $\mathbf{J}_{\text{Newmark}}$ is identical to $\mathbf{J}_{\text{NSTIFF}}$, except that the matrix \mathbf{P} is replaced by a matrix $\hat{\mathbf{P}}$ obtained by replacing $\frac{4}{9}$ with β and $\frac{2}{3}$ with γ .

Referred to as HHT-I3, the third method considered in this study relies on the HHT method [29], widely used in the structural dynamics community and first considered in the context of multi-body dynamics analysis in Ref. [31]. HHT-I3 is defined as follows (note that the discretized equations of motion have been scaled by $1/(1+\alpha)$):

$$\begin{aligned}\mathbf{q}_{n+1} &= \mathbf{q}_n + h\dot{\mathbf{q}}_n + \frac{h^2}{2}[(1-2\beta)\mathbf{a}_n + 2\beta\mathbf{a}_{n+1}] \\ \dot{\mathbf{q}}_{n+1} &= \dot{\mathbf{q}}_n + h[(1-\gamma)\mathbf{a}_n + \gamma\mathbf{a}_{n+1}] \\ \frac{1}{1+\alpha}(\mathbf{M}(\mathbf{q})\mathbf{a})_{n+1} + (\Phi_{\mathbf{q}}^T\lambda - \mathbf{Q})_{n+1} - \frac{\alpha}{1+\alpha}(\Phi_{\mathbf{q}}^T\lambda - \mathbf{Q})_n &= \mathbf{0} \\ \frac{1}{\beta h^2}\Phi(\mathbf{q}_{n+1}, t_{n+1}) &= \mathbf{0}\end{aligned}\quad (8)$$

The notation used in Eq. (8) is meant to emphasize that there is a distinction between $\dot{\mathbf{q}}_{n+1}$ and \mathbf{a}_{n+1} (compare with Eq. (5)). Concretely, \mathbf{a}_{n+1} is an approximation of $\ddot{\mathbf{q}}(t_n + (1+\alpha)h)$. This raises some difficulties in choosing \mathbf{a}_0 , an attribute that is associated with the use of HHT in general and is not specific to HHT-I3. In Ref. [36], it is recommended to take $\mathbf{a}_0 = \dot{\mathbf{q}}_0$ and in spite of this approximation the same convergence results hold for the global behavior of the method. For more accurate results, an implicit and therefore slightly more involved way of computing \mathbf{a}_0 is suggested in Ref. [35]. Finally, note that the last two equations in Eq. (8) lead to a nonlinear system that is solved with a Newton-type method for \mathbf{a}_{n+1} and λ_{n+1} . The associated Jacobian

$$\mathbf{J}_{\text{HHTI3}} = \begin{bmatrix} \frac{1}{1+\alpha}\mathbf{M} + \hat{\mathbf{P}} & \Phi_{\mathbf{q}}^T \\ \Phi_{\mathbf{q}} & \mathbf{0} \end{bmatrix}$$

does not become ill conditioned when $h \rightarrow 0$. Taking the limit, $\hat{\mathbf{P}} \rightarrow \mathbf{0}$ and $\mathbf{J}_{\text{HHTI3}}$ is nonsingular as long as the kinematic constraints are independent and the symmetric mass matrix is nonsingular.

The last three numerical integration methods considered herein take into account the velocity kinematic constraint equations. The salient attribute of these methods is a resulting set of consistent generalized velocities, an aspect relevant in frictional contact and controls applications. The method referred to as NSTIFF-SI2 is an implementation of the stabilized index-2 formulation reported in Ref. [20] that uses second-order BDF formulas [8]:

$$\begin{aligned}\mathbf{q}_{n+1} &= \frac{4}{3}\mathbf{q}_n - \frac{1}{3}\mathbf{q}_{n-1} + \frac{2}{3}h\dot{\mathbf{q}}_{n+1} \\ \mathbf{v}_{n+1} &= \frac{4}{3}\mathbf{v}_n - \frac{1}{3}\mathbf{v}_{n-1} + \frac{2}{3}h\dot{\mathbf{v}}_{n+1}\end{aligned}\quad (9)$$

NSTIFF-SI2 explicitly accounts for the velocity kinematic constraint equations and relies on an extra set of Lagrange multipliers μ to enforce these constraints. The unknowns are $\dot{\mathbf{v}}$, $\dot{\mathbf{q}}$, λ , and μ and the new configuration at t_{n+1} is the solution of the following system of nonlinear equations:

$$\begin{aligned}\mathbf{M}(\mathbf{q}_{n+1})\dot{\mathbf{v}}_{n+1} + \Phi_{\mathbf{q}}^T(\mathbf{q}_{n+1})\lambda_{n+1} - \mathbf{Q}(t_{n+1}, \mathbf{q}_{n+1}, \mathbf{v}_{n+1}) &= \mathbf{0} \\ \mathbf{v}_{n+1} - \dot{\mathbf{q}}_{n+1} + \Phi_{\mathbf{q}}^T(\mathbf{q}_{n+1})\mu_{n+1} &= \mathbf{0} \\ \frac{3}{2h}\Phi(\mathbf{q}_{n+1}, t_{n+1}) &= \mathbf{0} \\ \frac{3}{2h}\Phi_{\mathbf{q}}(\mathbf{q}_{n+1}, t_{n+1})\mathbf{v}_{n+1} + \frac{3}{2h}\Phi_{\mathbf{v}}(\mathbf{q}_{n+1}, t_{n+1}) &= \mathbf{0}\end{aligned}\quad (10)$$

When using a Newton-type method, the associated Jacobian assumes the form

$$\mathbf{J}_{\text{NSTIFF-SI2}} = \begin{bmatrix} \mathbf{M} & \frac{2h}{3}(\mathbf{M}\dot{\mathbf{v}} + \Phi_{\mathbf{q}}^T\lambda - \mathbf{Q})_{\mathbf{q}} & \Phi_{\mathbf{q}}^T & \mathbf{0} \\ \frac{2h}{3}\mathbf{I} & -\mathbf{I} - \frac{2h}{3}(\Phi_{\mathbf{q}}^T\mu)_{\mathbf{q}} & \mathbf{0} & \Phi_{\mathbf{q}}^T \\ \Phi_{\mathbf{q}} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ (\Phi_{\mathbf{q}}\mathbf{v})_{\mathbf{q}} + \Phi_{\mathbf{v}} & \Phi_{\mathbf{q}} & \mathbf{0} & \mathbf{0} \end{bmatrix}$$

Under mild conditions (symmetric nonsingular mass matrix and independent set of kinematic constraints), it can be easily shown that $\mathbf{J}_{\text{NSTIFF-SI2}}$ remains nonsingular when $h \rightarrow 0$. Also note that in the absence of discretization errors, μ would be identically zero.

The fifth method considered in this study introduces a correction into the Newmark formulas based on the constraint accelerations and was shown to have global convergence order 2 [35,36]. Given a configuration $(\mathbf{q}_n, \dot{\mathbf{q}}_n, \mathbf{a}_n)$ and defining $\mathbf{f}(t, \mathbf{q}, \dot{\mathbf{q}}) := \mathbf{M}^{-1}(\mathbf{q})\mathbf{Q}(t, \mathbf{q}, \dot{\mathbf{q}})$ and $\mathbf{r}(\mathbf{q}, \lambda) := -\mathbf{M}^{-1}(\mathbf{q})\Phi_{\mathbf{q}}^T\lambda$, the unknowns \mathbf{q}_{n+1} , $\dot{\mathbf{q}}_{n+1}$, \mathbf{a}_{n+1} , ψ_I , and ψ_{II} are found as the solution of the following nonlinear system:

$$\begin{aligned}\mathbf{q}_{n+1} &= \mathbf{q}_n + h\dot{\mathbf{q}}_n + \frac{h^2}{2}((1-2\beta)\mathbf{a}_n + 2\beta\mathbf{a}_{n+1}) + \frac{h^2}{2}((1-b)\mathbf{R}_I + b\mathbf{R}_{II}) \\ \dot{\mathbf{q}}_{n+1} &= \dot{\mathbf{q}}_n + h((1-\gamma)\ddot{\mathbf{q}}_n + \gamma\ddot{\mathbf{q}}_{n+1}) + \frac{h}{2}(\mathbf{R}_I + \mathbf{R}_{II}) \\ \mathbf{0} &= \Phi(\mathbf{q}_{n+1}, t_{n+1}) \\ \mathbf{0} &= \Phi_{\mathbf{q}}(\mathbf{q}_{n+1}, t_{n+1})\dot{\mathbf{q}}_{n+1} + \Phi_{\mathbf{v}}(\mathbf{q}_{n+1}, t_{n+1}) \\ \mathbf{a}_{n+1} &= (1+\alpha)\mathbf{f}(t_{n+1}, \mathbf{q}_{n+1}, \dot{\mathbf{q}}_{n+1}) - \alpha\mathbf{f}(t_n, \mathbf{q}_n, \dot{\mathbf{q}}_n)\end{aligned}\quad (11)$$

where $b \neq 1/2$ is a free coefficient, $\mathbf{R}_I := \mathbf{r}(t_n, \mathbf{q}_n, \psi_I)$, and $\mathbf{R}_{II} := \mathbf{r}(t_{n+1}, \mathbf{q}_{n+1}, \psi_{II})$. This method is referred as HHT-ADD and is discussed at length in Refs. [35,36] where the local and global error analysis results are provided along with an investigation of stability properties. In addition to displaying attractive numerical damping controlled through the parameter $\alpha \in [-0.3, 0]$, the method is shown to be order 2. The major drawback of this method is the multiplication by the inverse of the mass matrix. Specifically, this becomes a major concern in the inexact-Newton step when dealing with flexible body problems where, due to the coupling in the deformation modes, the mass matrix can have large dense blocks. The Jacobian $\mathbf{J}_{\text{HHT-ADD}}$ is not provided herein; the interested reader is referred to Ref. [36].

The last integration method investigated, HHT-SI2, represents a new algorithm that is analyzed theoretically in Ref. [37]. It represents a variation in the HHT-ADD algorithm that avoids multiplication by the inverse of the mass matrix. As such, it is amenable to handling mechanical systems with flexible bodies in which the formulation relies on the floating frame of reference approach [2]. For HHT-SI2, the Newmark integration formulas are modified slightly by introducing a correction $(h^2/2)\bar{\mathbf{a}}$:

$$\mathbf{q}_{n+1} = \mathbf{q}_n + h\dot{\mathbf{q}}_n + \frac{h^2}{2}[(1-2\beta)\mathbf{a}_n + 2\beta\mathbf{a}_{n+1}] + \frac{h^2}{2}\bar{\mathbf{a}} \quad (12a)$$

$$\dot{\mathbf{q}}_{n+1} = \dot{\mathbf{q}}_n + h[(1-\gamma)\mathbf{a}_n + \gamma\mathbf{a}_{n+1}] \quad (12b)$$

In advancing the integration from a given configuration at time t_n to t_{n+1} , the unknowns \mathbf{a}_{n+1} , $\bar{\mathbf{a}}$, λ_{n+1} , and μ are found as the solutions of the nonlinear system of equations:

$$\frac{1}{1+\alpha}\bar{\mathbf{M}}_{n+1}\mathbf{a}_{n+1} + (\Phi_{\mathbf{q}}^T\lambda - \mathbf{Q})_{n+1} - \frac{\alpha}{1+\alpha}(\Phi_{\mathbf{q}}^T\lambda - \mathbf{Q})_n = \mathbf{0}$$

$$\bar{\mathbf{M}}_{n+1}\bar{\mathbf{a}} - \Phi_{\mathbf{q}}^T(t_{n+1}, \mathbf{q}_{n+1})\mu = \mathbf{0}$$

$$\frac{1}{h^2} \Phi(\mathbf{q}_{n+1}, t_{n+1}) = \mathbf{0}$$

$$\frac{1}{h} \Phi_{\mathbf{q}}(\mathbf{q}_{n+1}, t_{n+1}) \dot{\mathbf{q}}_{n+1} + \frac{1}{h} \Phi_t(\mathbf{q}_{n+1}, t_{n+1}) = \mathbf{0} \quad (13)$$

where $\bar{\mathbf{M}}_{n+1} := \mathbf{M}(t_n + h(1 + \alpha), \mathbf{q}_n + h(1 + \alpha)\dot{\mathbf{q}}_n)$. Here $\bar{\mathbf{a}}$ and μ are the auxiliary variables local to the current time step. Introducing the notation $\mathcal{R} = h^2(\Phi_{\mathbf{q}}^T \mu)_{\mathbf{q}}$, $\mathcal{Q} = \beta h^2(\Phi_{\mathbf{q}}^T \lambda - \mathbf{Q})_{\mathbf{q}} - \gamma h \mathbf{Q}_{\mathbf{q}}$, $\mathcal{F} = h^2(\Phi_{\mathbf{q}}^T \lambda - \mathbf{Q})_{\mathbf{q}}$, and $\mathcal{V} = h(\Phi_{\mathbf{q}}(\mathbf{q}, t)\dot{\mathbf{q}} + \Phi_t(\mathbf{q}, t))_{\mathbf{q}}$, the Jacobian associated with the discretized problem assumes the expression

$$\mathbf{J}_{\text{HHT-SI2}} = \begin{bmatrix} \frac{1}{1 + \alpha} \bar{\mathbf{M}}_{n+1} + \mathcal{Q} & \frac{1}{2} \mathcal{F} & \Phi_{\mathbf{q}}^T & \mathbf{0} \\ -\beta \mathcal{R} & \bar{\mathbf{M}}_{n+1} - \frac{1}{2} \mathcal{R} & \mathbf{0} & -\Phi_{\mathbf{q}}^T \\ \beta \Phi_{\mathbf{q}} & \frac{1}{2} \Phi_{\mathbf{q}} & \mathbf{0} & \mathbf{0} \\ \gamma \Phi_{\mathbf{q}} + \beta \mathcal{V} & \frac{1}{2} \mathcal{V} & \mathbf{0} & \mathbf{0} \end{bmatrix}$$

If $\mathbf{J}_{\text{HHT-SI2}}^0 = \lim_{h \rightarrow 0} \mathbf{J}_{\text{HHT-SI2}}$, then

$$\mathbf{J}_{\text{HHT-SI2}}^0 = \begin{bmatrix} \frac{1}{1 + \alpha} \bar{\mathbf{M}}_{n+1} & \mathbf{0} & \Phi_{\mathbf{q}}^T & \mathbf{0} \\ \mathbf{0} & \bar{\mathbf{M}}_{n+1} & \mathbf{0} & -\Phi_{\mathbf{q}}^T \\ \beta \Phi_{\mathbf{q}} & \frac{1}{2} \Phi_{\mathbf{q}} & \mathbf{0} & \mathbf{0} \\ \gamma \Phi_{\mathbf{q}} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}$$

and under mild assumptions (symmetric nonsingular mass matrix and independent set of kinematic constraints) the matrix $\mathbf{J}_{\text{HHT-SI2}}^0$ turns out to be nonsingular. This guarantees acceptable behavior at small values of the step size, a situation typically encountered in penalty-based frictional contact problems. The main result regarding the convergence of the new method HHT-SI2 is stated as follows. Suppose that the initial configuration at time t_0 is such that

$$\mathbf{0} = \mathbf{M}(t_0, \mathbf{q}_0) \mathbf{a}_0 + \Phi_{\mathbf{q}}^T \lambda_0 - \mathbf{Q}(t_0, \mathbf{q}_0, \dot{\mathbf{q}}_0, \lambda_0)$$

$$\mathbf{0} = \Phi(t_0, \mathbf{q}_0)$$

$$\mathbf{0} = \Phi_t(t_0, \mathbf{q}_0) + \Phi_{\mathbf{q}}(t_0, \mathbf{q}_0) \dot{\mathbf{q}}_0$$

$$\mathbf{0} = \Phi_n(t_0, \mathbf{q}_0) + 2\Phi_{tq}(t_0, \mathbf{q}_0) \dot{\mathbf{q}}_0 + [\Phi_{\mathbf{q}}(t_0, \mathbf{q}_0) \dot{\mathbf{q}}_0]_{\mathbf{q}} \dot{\mathbf{q}}_0 + \Phi_{\mathbf{q}}(t_0, \mathbf{q}_0) \mathbf{a}_0$$

and $\mathbf{a}_\alpha - \mathbf{a}(t_0 + \alpha h) = O(h)$. Then the numerical approximation $(\mathbf{q}_n, \dot{\mathbf{q}}_n, \mathbf{a}_{n+\alpha}, \lambda_n)$ produced by the HHT-SI2 method in Eqs. (12) and (13) satisfies

$$\mathbf{q}_n - \mathbf{q}(t_n) = O(h^2)$$

$$\dot{\mathbf{q}}_n - \dot{\mathbf{q}}(t_n) = O(h^2)$$

$$\mathbf{a}_{n+\alpha} - \mathbf{a}(t_n + \alpha h) = O(h^2)$$

$$\lambda_n - \lambda(t_n) = O(h^2)$$

for $0 < h \leq h_{\max}$ and $t_n - t_0 = nh \leq \text{const}$, where h_{\max} is suitably chosen. Here $\mathbf{q}(t_n)$, $\dot{\mathbf{q}}(t_n)$, $\mathbf{a}(t_n + \alpha h)$, and $\lambda(t_n)$ denote the exact value of the respective unknown quantities at the times indicated in parentheses. A formal proof of this is provided in Ref. [37].

2.1 Implementation Details. The computational flow associated with any of the six integration methods discussed can be abstracted in the following way. A set of unknowns \mathbf{w}_{n+1} is computed as the solution of a nonlinear system $\mathbf{Y}(\mathbf{w}_{n+1}) = \mathbf{0}$. In turn,

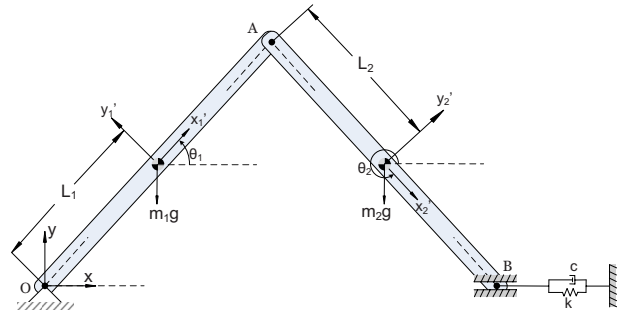


Fig. 1 Slider crank

the position and velocity at the new configuration t_{n+1} are evaluated based on a set of integration formulas: $\dot{\mathbf{q}}_{n+1} = \mathcal{I}_1(\mathbf{w}_{n+1})$ and $\mathbf{q}_{n+1} = \mathcal{I}_2(\mathbf{w}_{n+1})$. Illustrating this abstraction for HHT-SI2, the expression of \mathbf{Y} is obtained from Eq. (13), \mathcal{I}_1 is provided by Eq. (12b), and \mathcal{I}_2 is provided by Eq. (12a). Regardless of the method used, advancing the solution from t_n to t_{n+1} follows a simple recipe:

```

t_{n+1} = t_n + h %L1
w_{n+1}^{(0)} = w_n %L2
Do %L3
  q_{n+1}^{(k)} = I_1(w_{n+1}^{(k)}) %L4
  q_{n+1}^{(k)} = I_2(w_{n+1}^{(k)}) %L5
  Evaluate Jacobian J %L6
  Solve linear system J Delta w^{(k)} = Y(w_{n+1}^{(k)}) %L7
  If ||Delta w^{(k)}|| <= epsilon then break %L8
  Apply correction: w_{n+1}^{(k+1)} = w_{n+1}^{(k)} - Delta w^{(k)} %L9
End do %L10
w_{n+1} = w_{n+1}^{(k)} %L11

```

Certain variations in this algorithm can improve its efficiency. For instance, rather than evaluating it at each time step, the Jacobian can be evaluated less frequently. While a costly proposition in itself, each Jacobian evaluation is necessarily followed by a factorization step, which is also costly. Note that although the convergence test relies exclusively on the correction norm at line L8 of the pseudocode, the test could also include the norm of residual, i.e., the right side of the linear system in L7.

3 Numerical Experiments

The numerical algorithms NSTIFF, Newmark, HHT-I3, NSTIFF-SI2, HHT-ADD, and HHT-SI2 were implemented in MATLAB and used in conjunction with three models. Several experiments were run to evaluate the algorithms' performance and compare them in relation to the order of global convergence, energy preservation, constraint satisfaction, and efficiency. The models considered for testing and comparison of algorithm performance were a slider crank, a slider crank with a flexible connecting rod, and a seven body mechanism (see, for instance, Refs. [40–42]). The model parameters and the initial conditions used are summarized below.

Slider crank. The schematic of a slider-crank model including a spring-damper element is shown in Fig. 1. The parameters associated with the model are $m_1 = 3$ kg, $L_1 = 0.3$ m, $m_2 = 0.9$ kg, $L_2 = 0.6$ m, $k = 100$ N/m, and $c = 5$ N s/m. Both links are symmetric and homogeneous, and the center of mass is at the midpoint. The initial conditions used for simulation of motion were $\theta_1(0) = 3\pi/2$ and $\dot{\theta}_1(0) = 0$ rad/s.

Flexible slider crank. This model is similar to the rigid slider crank shown in Fig. 1, except that the spring and damper are not included and the connecting rod AB is flexible. The parameter values used in this model are $m_1 = 3$ kg, $L_1 = 0.3$ m, $m_2 = 0.9$ kg,

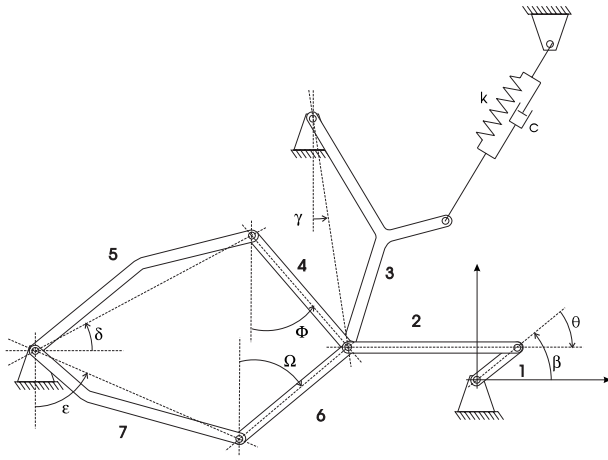


Fig. 2 Seven body mechanism

$L_2=0.6$ m, cross-section area $S=5.74 \times 10^{-6}$ m², moment of inertia $I=2.765 \times 10^{-8}$ m⁴, and Young's modulus $E=200$ GPa. Both links are symmetric and homogeneous, and the center of mass is at the midpoint. The initial conditions are $\theta_1(0)=3\pi/2$ and $\dot{\theta}_1(0)=1$ rad/s. The equations of motion are formulated using the floating frame of reference formulation, see Ref. [2] (p. 231).

Seven body mechanism. The model is presented in Fig. 2. For this set of numerical experiments, the value of the damping c was set zero. An account of the geometry of the mechanism, along with inertia properties and initial conditions, is provided in Ref. [40]. The mechanism moves due to a torque applied to crank 1. All bodies in the model are rigid.

3.1 Global Convergence Analysis. The goal of the first set of numerical experiments is to assess how the global integration error

decreases with the integration step size, i.e., to carry out a convergence analysis. From an analytical perspective, theoretical results that predict error versus step-size behavior exist for five out of the six algorithms considered herein. Thus, NSTIFF should display second-order behavior [43,44], HHT-I3 has been recently proved to be a second-order method [34], NSTIFF-SI2 should display second-order convergence [20], HHT-ADD has been proved to be a second-order method [35,36], and HHT-SI2 should display second-order global convergence [37]. The only algorithm that does not have a formal convergence proof is Newmark, but considering its track record in dealing with ODEs it is conjectured that in conjunction with index-3 DAEs of multibody dynamics it would display first order global convergence.

To investigate the convergence order of each numerical method for the rigid slider crank, a reference solution was first determined by deriving a set of second-order ODEs that govern the time evolution of the system. This ODE problem is subsequently solved using a fourth order Runge-Kutta method (see, for instance, Ref. [7]) with a step size of $h=10^{-6}$ s. The convergence behavior is shown in Fig. 3, which displays the crank angular velocity absolute error at time $T=2$ s obtained with each method over a set of integration step sizes. Ideally, these slopes should be 2, except for Newmark, which should display first order convergence and therefore a slope of 1. Indeed, the numerical results confirm that all methods behave as predicted by theory. Although not presented here, similar numerical results were reported for a very stiff double pendulum in Ref. [45], and they also indicate numerical convergence results aligned with theoretical predictions.

Since the equations of motion were too involved, for the flexible slider-crank model and the seven body mechanism they were not reduced first to a set of ODEs. Rather, the reference solution was obtained with HHT-ADD with a step size of $h=10^{-6}$ s. The flexible slider crank was simulated for 2 s and the numerical solution was compared with the reference solution at the final time. The results suggest that NSTIFF, HHT-I3, HHT-SI2, HHT-ADD,

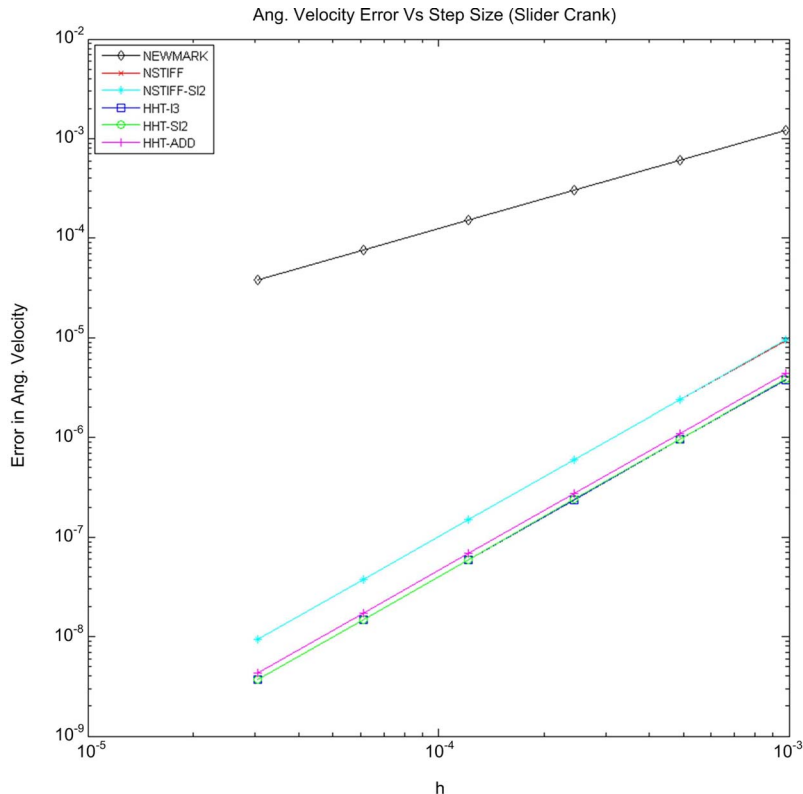


Fig. 3 Convergence, slider crank

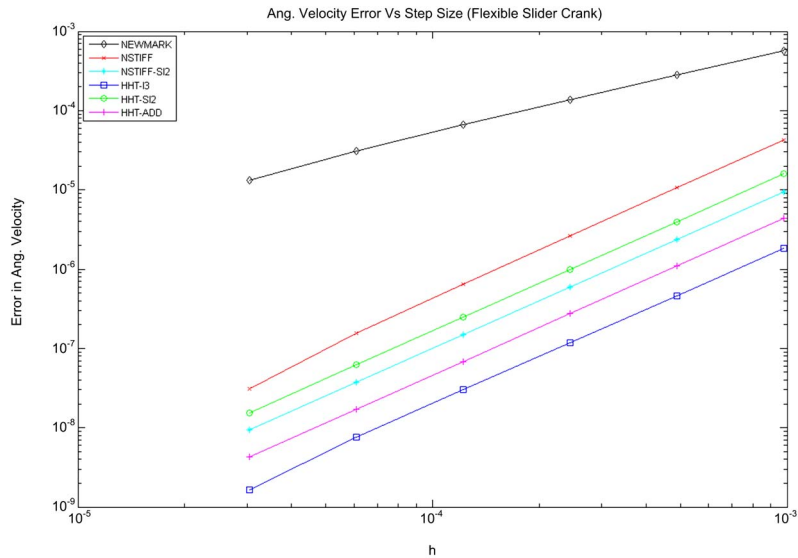


Fig. 4 Convergence, flexible slider crank

and NSTIFF-SI2 exhibit order 2 convergence, in line with the theoretical results established in conjunction with these algorithms. Furthermore, Newmark shows global convergence of order 1 for all models. The convergence orders hold both for the generalized coordinates and their time derivative, that is, both for positions and velocities. Figure 4 displays the convergence and order for the flexible slider crank; the results reported concern the angular velocity of the crank. Finally, for body 5 of the seven body mechanism, see Fig. 2, the convergence plots for its orientation and angular velocity are displayed in Figs. 5 and 6.

3.2 Energy Preservation. The HHT method came as an improvement over Newmark formulas because it preserved the A-stability and its attractive numerical damping properties while

achieving second-order accuracy. In this method, high-frequency oscillations that are not of interest, as well as parasitic high-frequency oscillations that are a by-product of the finite element discretization, are damped out through the parameter α . The choice of α is based on the desired level of damping: The more negative the value of α , the more damping is induced in the numerical solution. Note that the choice $\alpha=0$ leads to the trapezoidal method with no numerical damping. The effect of this damping can be seen from energy preservation plots shown in Figs. 7 and 8. These energy plots are for the slider-crank model from which the translational damper was removed. The system is conservative and, for the particular reference system employed, the total energy should be constant and equal to zero.

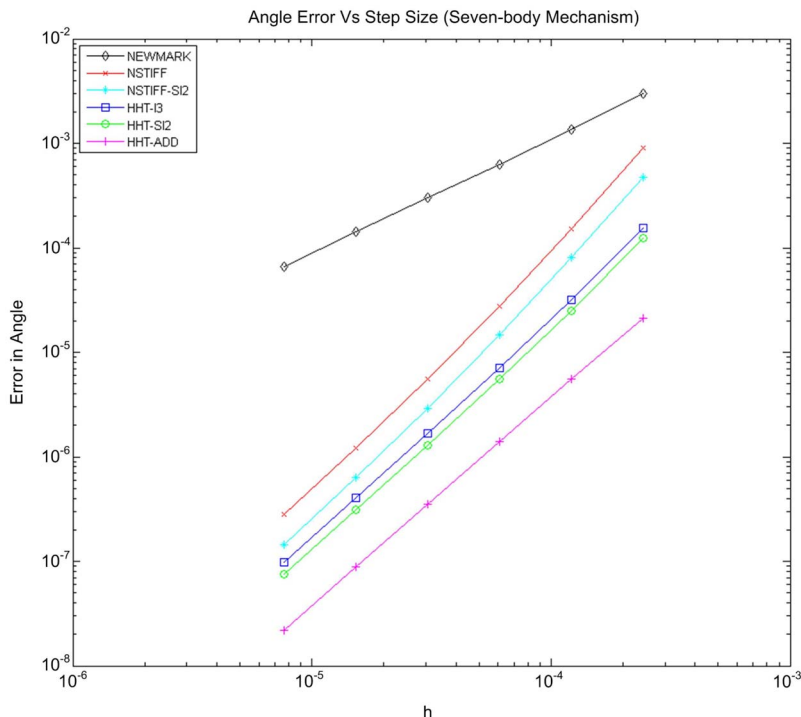


Fig. 5 Convergence, orientation

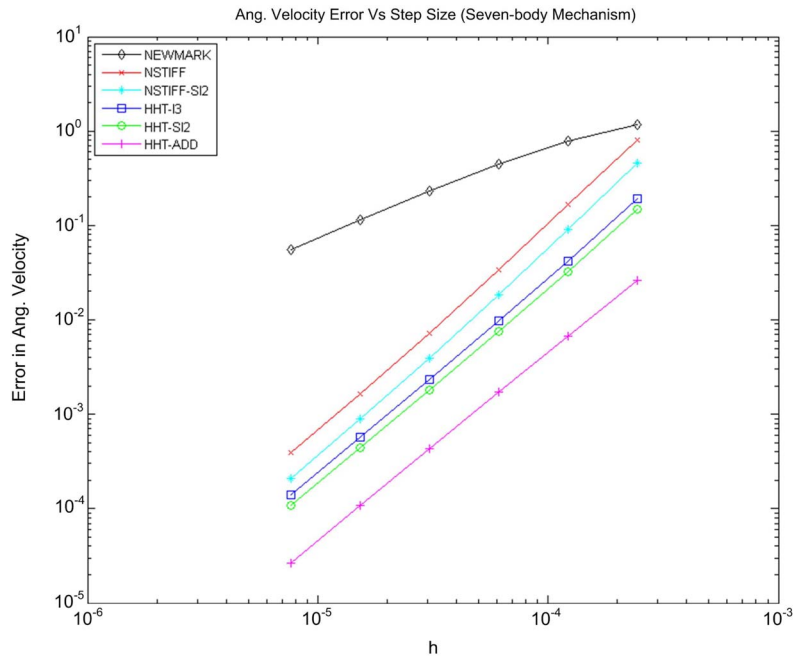


Fig. 6 Convergence, angular velocity

For $\alpha = -0.3$, the numerical damping-induced dissipation is one order of magnitude more pronounced than the $\alpha = -0.05$ case, qualitatively in line with expectations. Even more relevant is an investigation of how the numerical energy dissipation changes with the step size. The results in Fig. 8 indicate a highly oscillatory pattern. To capture the degree to which a numerical scheme dissipates energy, an average energy dissipation over an interval $[0, T]$ is computed as

$$\varepsilon(T) = \frac{1}{T} \int_0^T |E_{\text{tot}}(t)| dt \quad (14)$$

If no numerical dissipation was present in the system then $\varepsilon(T) = 0, \forall T > 0$. On a log-log scale, Fig. 9 shows this quantity for the rigid slider-crank model with no physical damping, while Fig. 10 displays the same quantity for the flexible slider crank. This average energy error for Newmark converges to zero like $O(h)$, while for all the other methods, it converges to zero like $O(h^2)$. In other words, the convergence is order $O(h^q)$, where q is the order of the

method. Although this does not serve as a formal proof, this attribute deserves further investigation, since $\varepsilon(T)$ is an average quantity that captures the energy drift over the entire simulation. Such a result could be relevant, for instance, in the context of molecular dynamics (MD) simulation, where the entire classes of integrators are disqualified if they do not preserve energy. However, with values in the femtosecond range, the step size for MD simulations might be so small that particularly HHT, through its variable-damping attribute, might, in fact, be a viable numerical integration formula. This aspect is further investigated in Ref. [46].

3.3 Kinematic Constraint Drift. The rationale behind stabilizing the numerical solution of the index-3 DAE of multibody dynamics using the velocity kinematic constraint equations is to prevent drift in satisfying this set of algebraic constraints. Three of the six methods analyzed in this study, namely, HHT-ADD, HHT-SI2, and NSTIFF-SI2, enforce these equations. As such, no velocity constraint drift is expected in the numerical solution. This is

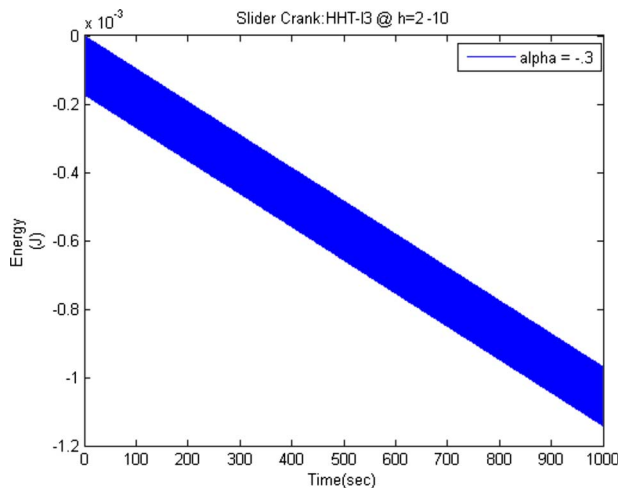


Fig. 7 Energy dissipation at $\alpha = -0.3$

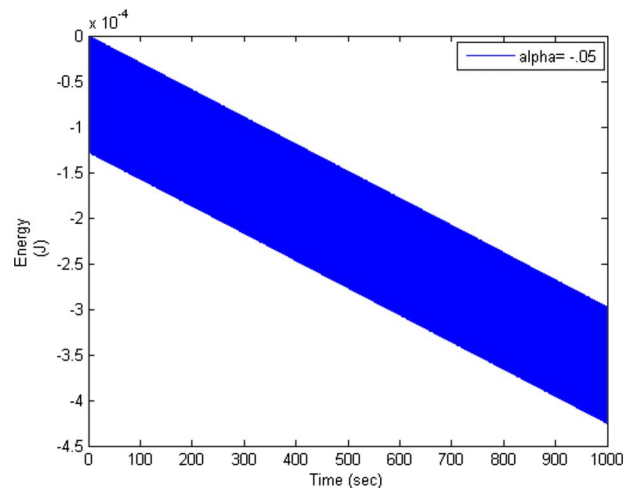


Fig. 8 Energy dissipation at $\alpha = -0.05$

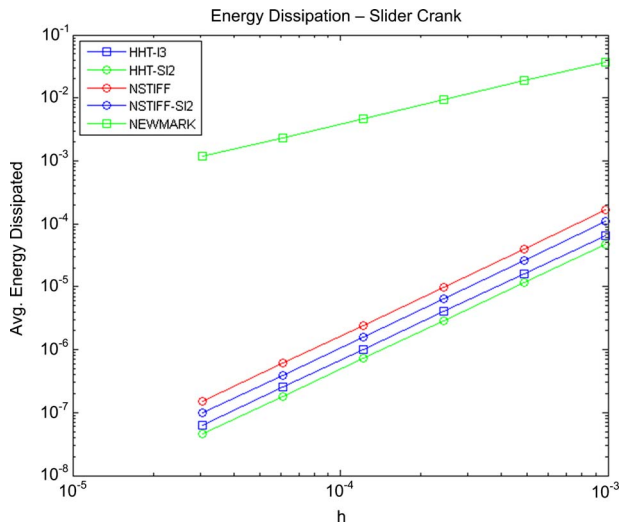


Fig. 9 Dissipation, slider crank

confirmed by the plots in Figs. 11 and 12, which display the velocity constraint violation in the X direction against the velocity constraint violation in the Y direction for the rigid slider-crank mechanism for the pin joint between the crank and ground. Data were plotted at each time step and, as anticipated, confirm that the velocity kinematic constraint equations are satisfied within machine precision. A qualitatively identical plot for NSTIFF-SI2 is provided in Ref. [47].

For Newmark, NSTIFF, and HHT-I3, Figs. 13–15 report the same information for the rigid slider crank with no damping obtained during a 10 s simulation with a step size $h=2^{-10}$ s. One remarkable property is that Newmark, HHT-I3, and NSTIFF display the same error behavior. Moreover, as the step size decreases, the box that bounds the plot shrinks but the shape of the curves remains the same for all three integration methods. The cause of this behavior remains to be investigated but these results suggest that this limit cycle behavior is a characteristic of the direct index-3 methodology; i.e., neglecting velocity kinematic con-

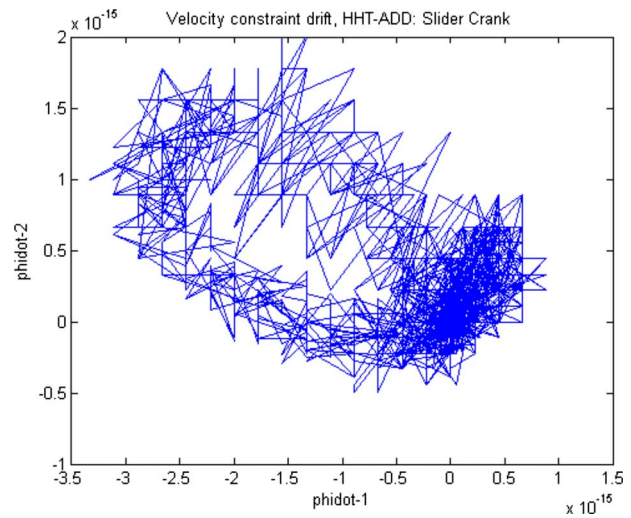


Fig. 11 Velocity drift, HHT-ADD

straint equations, rather than that of the algorithm used for the numerical solution. For now, it should be pointed out that numerical experiments indicate that the error in satisfying these constraints converges like $O(h^q)$, where q is the order of the method. A more formal investigation of these observations remains to be done. Qualitatively identical plots are provided for the flexible slider crank in Ref. [47].

3.4 Runtime Comparison. The six methods investigated in this work were used to run simulations of the time evolution of the three previously discussed models. Additionally, for comparison purposes and drawing on the results reported in Ref. [45], a double pendulum mechanism is also considered. The goal is to compare the amount of work per time step required to produce an approximation of the solution. In this undertaking, the integration step size was identical for all algorithms, although it was different for different models. Also, specific to each model was the simulation end time. In order to allow for a unified perspective on the efficiency issue, the CPU times required to complete the analyses

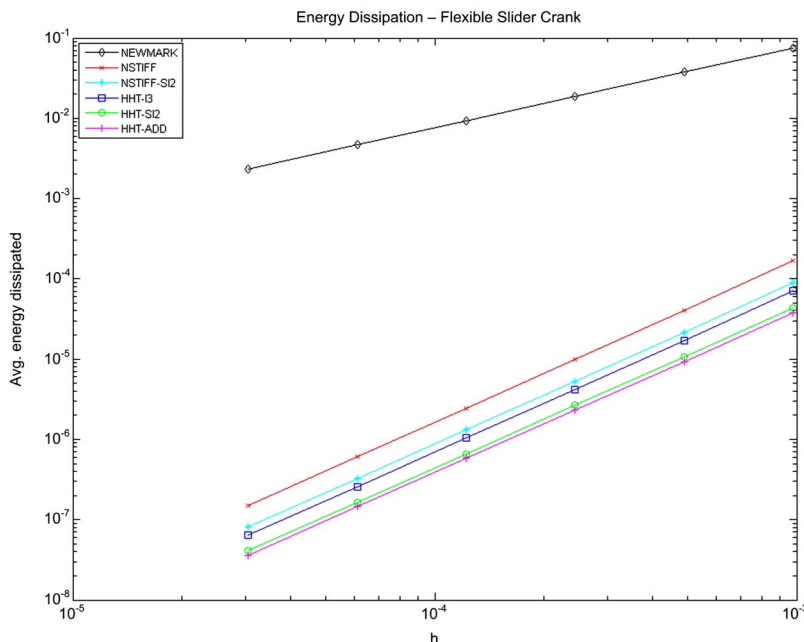


Fig. 10 Dissipation, flexible slider crank

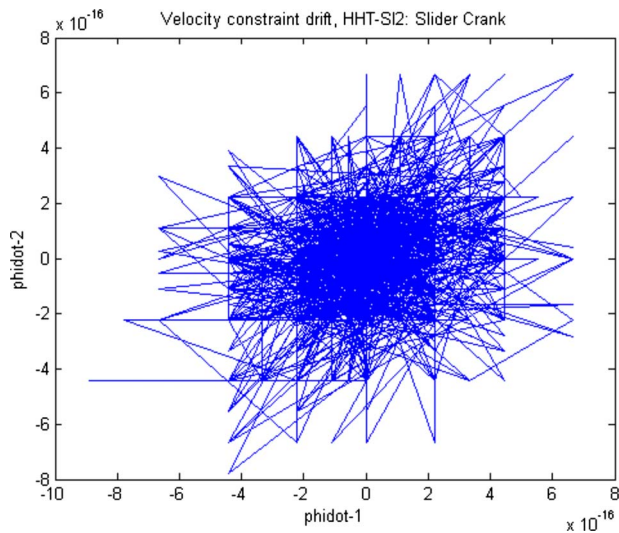


Fig. 12 Velocity drift, HHT-SI2

were reported in Fig. 16 after being normalized to the time it took the HHT-I3 method to finish the simulation. In other words, for each of the four models, the HHT-I3 provides the reference. The results in Fig. 16 suggest that having the kinematic velocity constraint equations enforced usually leads to an approximate simulation slowdown of 30%, unless the model is heavily constrained, as is the case with the seven body mechanism. For the seven body mechanism, the number of second-order differential equations was 21 and the number of constraints 20, in which case relying on the velocity kinematic constraint equations for stabilization purposes slows down the overall simulation due to a rather significant increase in the dimension of the problem: from 41 nonlinear differential-algebraic equations for HHT-I3 to 61 for HHT-SI2 and NSTIFF-SI2. Finally, as expected, the HHT_ADD is very costly for the flexible body model given that the mass matrix ceases to be constant. This trend gets exacerbated as the dimension of the problem increases, as is the case with the seven body mechanism, effectively making HHT_ADD an algorithm that is robust but of limited practical interest. Note that the mass matrix associated with the HHT-SI2 algorithm being evaluated somewhere midstep and then kept constant led to improved performance when compared with the NSTIFF-SI2 alternative. In other words, for large models, it is anticipated that the newly proposed algorithm HHT-

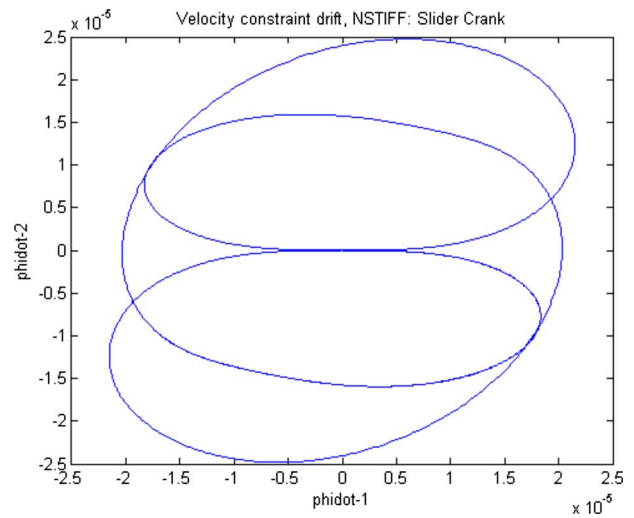


Fig. 14 Velocity drift, NSTIFF

SI2 will be attractive both on grounds of efficiency and variable-damping characteristics. It should also be pointed out that the timing results reported herein are only qualitative as there are a multitude of factors that ultimately dictate the efficiency of an algorithm: memory access, step-size selection, Newton-convergence issues, predictor, etc. The impact of these factors is

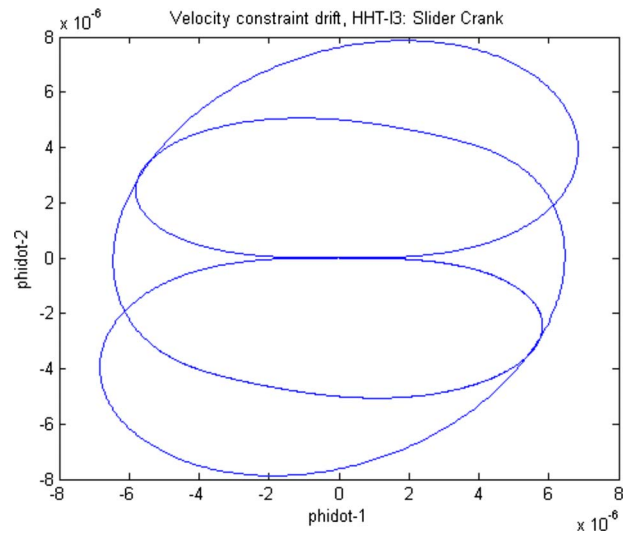


Fig. 15 Velocity drift, HHT-I3

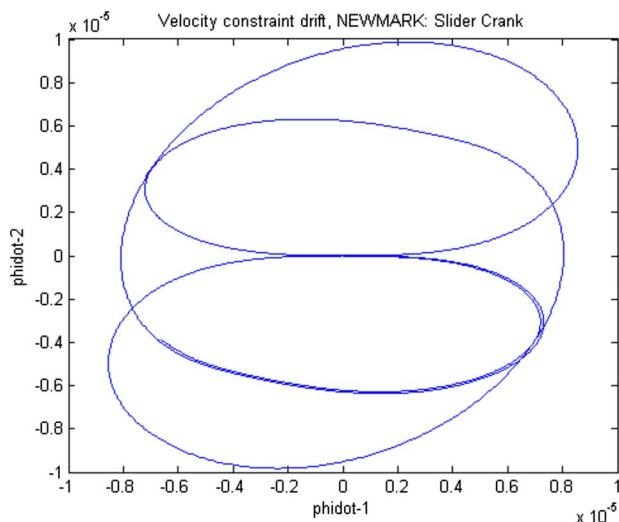


Fig. 13 Velocity drift, Newmark

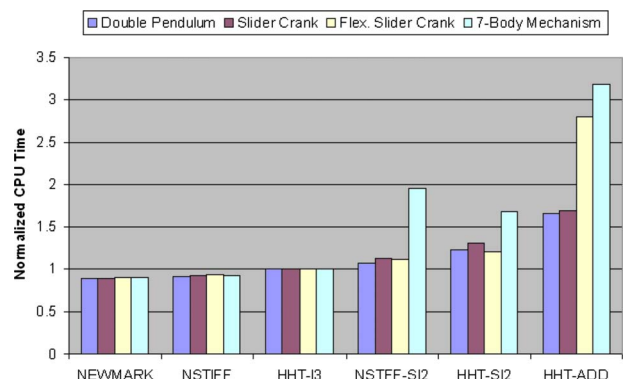


Fig. 16 Runtime comparison

highlighted in Ref. [33], where it is reported that these implementation issues actually rendered HHT-I3 two times faster than NSTIFF.

4 Conclusions

This paper investigates six low-order numerical integration formulas for determining the time evolution of constrained multibody systems. The motivation for this effort was twofold. First, the vast majority of large real-life models contain high stiffness, discontinuities, friction, and contacts that effectively make low-order integration formulas the only viable alternative for numerical simulation. The comparison of these commonly used integration formulas shed light on some advantages and disadvantages associated with each method. Second, the comparison served as the vehicle that introduced a new integration method, HHT-SI2, and placed it in the wider family of index-3 and stabilized index-2 methods for the numerical solution of the DAEs of multibody dynamics.

Compared with higher-order implicit formulas, the numerical methods investigated herein are robust and straightforward to implement. The algorithms discussed do not have ill-conditioning issues associated with small integration step sizes due to the suggested scaling, are backed up (with the exception of Newmark) by sound theoretical results, and come in two flavors: index-3 and stabilized index-2. Based on the convergence order and timing results presented, for problems where accurately satisfying the velocity kinematic constraint equations is not a priority, HHT-I3, an algorithm extensively tested and validated on large models, represents a good choice. It is a second-order method that has the ability to change the amount of numerical damping that enters the solution process and has recently been implemented in the commercial package ADAMS [48]. The NSTIFF method is the next best alternative. However, the method is plagued by a somewhat more intense numerical damping that cannot be controlled like in HHT-I3. For a slower but more robust approach, one can select either HHT-SI2 or NSTIFF-SI2 methods. They are comparable in terms of efficiency, yet HHT-SI2 has an edge due to (i) its ability to adjust the value of numerical damping introduced in the solution and (ii) the handling of the mass matrix, which is bound to lead to efficiency gains for large models. Relative to the simulation times associated with the straight I3 methods, preliminary results indicate that satisfying both the position and velocity kinematic constraint equations comes at a price of about a 30% increase in simulation time.

Acknowledgment

This material is based on work supported by the National Science Foundation under Grant No. CMMI-0700191. Additional financial support was provided by MSC.Software, and, for the first author, by the Wisconsin Space Grant Consortium. The authors would like to thank Radu Serban, Nick Schafer, and Tim Knapp for reading the manuscript and providing suggestions for improvement, and Toby Heyn for running the convergence order simulations for the BDF integrators.

References

- [1] Haug, E. J., 1989, *Computer-Aided Kinematics and Dynamics of Mechanical Systems*, Vol. I, Prentice-Hall, Englewood Cliffs, NJ.
- [2] Shabana, A. A., 2005, *Dynamics of Multibody Systems*, 3rd ed., Cambridge University Press, Cambridge.
- [3] Abraham, R., and Marsden, J. E., 1985, *Foundations of Mechanics*, Addison-Wesley, Reading, MA.
- [4] Arnold, V., 1989, *Mathematical Methods of Classical Mechanics*, Springer, New York.
- [5] Brenan, K. E., Campbell, S. L., and Petzold, L. R., 1989, *Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations*, North-Holland, New York.
- [6] Lubich, C., and Hairer, E., 1989, "Automatic Integration of the Euler-Lagrange Equations With Constraints," *J. Comput. Appl. Math.*, **12**, pp. 77–90.
- [7] Hairer, E., and Wanner, G., 1991, *Solving Ordinary Differential Equations*,

- Vol. II (Computational Mathematics) Springer-Verlag, Berlin.
- [8] Gear, C. W., 1971, *Numerical Initial Value Problems of Ordinary Differential Equations*, Prentice-Hall, Englewood Cliffs, NJ.
- [9] Bauchau, O., and Laulusa, A., 2008, "Review of Contemporary Approaches for Constraint Enforcement in Multibody Systems," *ASME J. Comput. Nonlinear Dyn.*, **3**, pp. 011005.
- [10] Potra, F., and Rheinboldt, W. C., 1991, "On the Numerical Solution of Euler-Lagrange Equations," *Mech. Struct. Mach.*, **19**(1), pp. 1–18.
- [11] Rheinboldt, W. C., 1984, "Differential-Algebraic Systems as Differential Equations on Manifolds," *Math. Comput.*, **43**, pp. 473–482.
- [12] Wehage, R. A., and Haug, E. J., 1982, "Generalized Coordinate Partitioning for Dimension Reduction in Analysis of Constrained Dynamic Systems," *ASME J. Mech. Des.*, **104**, pp. 247–255.
- [13] Liang, C. D., and Lance, G. M., 1987, "A Differentiable Null-Space Method for Constrained Dynamic Analysis," *ASME J. Mech., Transm., Autom. Des.*, **109**, pp. 405–410.
- [14] Yen, J., 1993, "Constrained Equations of Motion in Multibody Dynamics as ODEs on Manifolds," *SIAM (Soc. Ind. Appl. Math.) J. Numer. Anal.*, **30**(2), pp. 553–558.
- [15] Alishenas, T., 1992, "Zur numerischen behandlungen, stabilisierung durch projection und modellierung mechanischer systeme mit nebenbedingungen und invarianten," Ph.D. thesis, Royal Institute of Technology, Stockholm.
- [16] Mani, N., Haug, E., and Atkinson, K., 1985, "Singular Value Decomposition for Analysis of Mechanical System Dynamics," *ASME J. Mech., Transm., Autom. Des.*, **107**, pp. 82–87.
- [17] Haug, E. J., Negrut, D., and Iancu, M., 1997, "A State-Space Based Implicit Integration Algorithm for Differential-Algebraic Equations of Multibody Dynamics," *Mech. Struct. Mach.*, **25**(3), pp. 311–334.
- [18] Negrut, D., Haug, E. J., and German, H. C., 2003, "An Implicit Runge-Kutta method for Integration of Differential-Algebraic Equations of Multibody Dynamics," *Multibody Syst. Dyn.*, **9**(2), pp. 121–142.
- [19] Orlandea, N., Chace, M. A., and Calahan, D. A., 1977, "A Sparsity-Oriented Approach to the Dynamic Analysis and Design of Mechanical Systems—Part I and Part II," *ASME J. Eng. Ind.*, **99**, pp. 773–784.
- [20] Gear, C. W., Gupta, G., and Leimkuhler, B., 1985, "Automatic Integration of the Euler-Lagrange Equations With Constraints," *J. Comput. Appl. Math.*, **12–13**, pp. 77–90.
- [21] Fuhrer, C., and Leimkuhler, B. J., 1991, "Numerical Solution of Differential-Algebraic Equations for Constrained Mechanical Motion," *Numer. Math.*, **59**(1), pp. 55–69.
- [22] Ascher, U. M., and Petzold, L. R., 1993, "Stability of Computational Methods for Constrained Dynamics Systems," *SIAM J. Sci. Comput. (USA)*, **14**(1), pp. 95–120.
- [23] Ascher, U. M., Chin, H., and Reich, S., 1994, "Stabilization of DAEs and Invariant Manifolds," *Numer. Math.*, **67**(2), pp. 131–149.
- [24] Ascher, U. M., Chin, H., Petzold, L., and Reich, S., 1995, "Stabilization of Constrained Mechanical Systems With DAEs and Invariant Manifolds," *Mech. Struct. Mach.*, **23**(2), pp. 135–157.
- [25] Lubich, C., Engstler, C., Nowak, U., and Pohle, U., 1995, "MEXX—Numerical Software for the Integration of Constrained Mechanical Multibody Systems," *Mech. Based Des. Struct. Mach.*, **23**, pp. 473–495.
- [26] Bauchau, O. A., Bottasso, C. L., and Trainelli, L., 2003, "Robust Integration Schemes for Flexible Multibody Systems," *Comput. Methods Appl. Mech. Eng.*, **192**, pp. 395–420.
- [27] Hughes, T. J. R., 1987, *Finite Element Method: Linear Static and Dynamic Finite Element Analysis*, Prentice-Hall, Englewood Cliffs, NJ.
- [28] Geradin, M., and Rixen, D., 1994, *Mechanical Vibrations: Theory and Application to Structural Dynamics*, Wiley, New York.
- [29] Hilber, H. M., Hughes, T. J. R., and Taylor, R. L., 1977, "Improved Numerical Dissipation for Time Integration Algorithms in Structural Dynamics," *Earthquake Eng. Struct. Dyn.*, **5**, pp. 283–292.
- [30] Chung, J., and Hulbert, G. M., 1993, "A Time Integration Algorithm for Structural Dynamics With Improved Numerical Dissipation: The Generalized- α Method," *Trans. ASME, J. Appl. Mech.*, **60**(2), pp. 371–375.
- [31] Cardona, A., and Geradin, M., 1989, "Time Integration of the Equation of Motion in Mechanical Analysis," *Comput. Struct.*, **33**, pp. 801–820.
- [32] Yen, J., Petzold, L., and Raha, S., 1998, "A Time Integration Algorithm for Flexible Mechanism Dynamics: The DAE α -Method," *Comput. Methods Appl. Mech. Eng.*, **158**, pp. 341–355.
- [33] Negrut, D., Rampalli, R., Ottarsson, G., and Sajdak, A., 2007, "On the Use of the HHT Method in the Context of Index 3 Differential Algebraic Equations of Multibody Dynamics," *ASME J. Comput. Nonlinear Dyn.*, **2**(1), pp. 73–85.
- [34] Arnold, M., and Bruls, O., 2007, "Convergence of the Generalized- α Scheme for Constrained Mechanical Systems," Martin Luther University, Technical Report No. 9-2007.
- [35] Lunk, C., and Simeon, B., 2006, "Solving Constrained Mechanical Systems by the Family of Newmark and α -Methods," *Z. Angew. Math. Mech.*, **86**, pp. 772–784.
- [36] Jay, L. O., and Negrut, D., 2007, "Extensions of the HHT- α Method to Differential-Algebraic Equations in Mechanics," *Electron. Trans. Numer. Anal.*, **26**, pp. 190–208.
- [37] Jay, L. O., and Negrut, D., 2008, "A Second Order Extension of the Generalized- α Method for Constrained Systems in Mechanics," unpublished.
- [38] Bottasso, C. L., Bauchau, O. A., and Cardona, A., 2007, "Time-Step-Size-Independent Conditioning and Sensitivity to Perturbations in the Numerical Solution of Index Three Differential Algebraic Equations," *SIAM J. Sci. Com-*

- put. (USA), **3**, pp. 395–420.
- [39] Newmark, N. M., 1959, “A Method of Computation for Structural Dynamics,” *J. Engrg. Mech. Div.*, **112**, pp. 67–94.
- [40] 1990, *Multibody Systems Handbook*, W. Schiehlen, ed., Springer, New York.
- [41] Hairer, E., and Wanner, G., 1996, *Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems*, Springer, New York.
- [42] Khude, N., and Negrut, D., 2007, “A MATLAB Implementation of the Seven-Body Mechanism for Implicit Integration of the Constrained Equations of Motion,” Simulation-Based Engineering Laboratory, The University of Wisconsin-Madison, Technical Report No. TR-2007-07.
- [43] Lötstedt, C., and Petzold, L., 1986, “Numerical Solution of Nonlinear Differential Equations With Algebraic Constraints I: Convergence Results for Backward Differentiation Formulas,” *Math. Comput.*, **174**, pp. 491–516.
- [44] Brennan, K., and Engquist, B. E., 1988, “Backward Differentiation Approximations of Nonlinear Differential/Algebraic Systems,” *Math. Comput.*, **51**(184), pp. 659–676.
- [45] Negrut, D., Jay, L., Khude, N., and Heyn, T., 2007, “A Discussion of Low-Order Integration Formulas for Rigid and Flexible Multibody Dynamics,” Proceedings of the Multibody Dynamics ECCOMAS Thematic Conference.
- [46] Schafer, N., Negrut, D., and Serban, R., 2008, “Experiments to Compare Implicit and Explicit Methods of Integration in Molecular Dynamics Simulation,” Simulation-Based Engineering Laboratory, The University of Wisconsin-Madison, Technical Report No. TR-2008-01.
- [47] Khude, N., Jay, L. O., and Negrut, D., 2008, “A Comparison of Low Order Numerical Integration Formulas for Rigid and Flexible Multibody Dynamics,” Simulation-Based Engineering Laboratory, The University of Wisconsin-Madison, Technical Report No. TR-2008-02.
- [48] MSC.Software, 2005, *ADAMS User’s Manual*, <http://www.mscsoftware.com>.