A Rosenbrock–Nystrom state space implicit approach for the dynamic analysis of mechanical systems: II—method and numerical examples

D Negrut[1]*, **A Sandu**[2], **E J Haug**[3], **F A Potra**[4] and **C Sandu**[5]
[1]MSCsoftware, Ann Arbor, MI, USA
[2]Department of Computer Science, Michigan Technological University, Houghton, MI, USA
[3]Department of Mechanical Engineering, The University of Iowa, Iowa City, IA, USA
[4]Department of Mathematics and Statistics, University of Maryland, Baltimore County, Baltimore, MD, USA
[5]Department of Mechanical Engineering—Engineering Mechanics, Michigan Technological University, Houghton, MI, USA

Abstract: When performing dynamic analysis of a constrained mechanical system, a set of index three differential-algebraic equations (DAE) describes the time evolution of the system. A state–space based method for the numerical solution of the resulting DAE has also been developed. The numerical method uses a linearly implicit time stepping formula of the Rosenbrock type, which is suitable for medium accuracy integration of stiff systems. This paper discusses choices of method coefficients and presents numerical results. For stiff mechanical systems, the proposed algorithm is shown to reduce significantly simulation times when compared to state of the art existent algorithms. The better efficiency is due to the use of an L-stable integrator, and a rigorous and general approach to providing analytical derivatives required by it.

NOTATION

$(a)_{ij}, (c)_{ij}, (b)_i,$ $(\hat{b})_i, (m)_i, (\hat{m})_i$	Rosenbrock–Nystrom method coefficients
$Atol_i$	user prescribed absolute tolerance for variable i
c_1, c_2	damping coefficients
e	generic variable considered for error analysis
err	integration local error
E	values obtained for the generic variable during the reference simulation
E^*	value of E at the time step where the integration error is maximum
$fac, facmin,$ $facmax$	step-size selection safety factors
h	integration step size
J	Rosenbrock Jacobian w.r.t. y
J_1	Rosenbrock–Nystrom Jacobian w.r.t. y
J_2	Rosenbrock–Nystrom Jacobian w.r.t. y'
k_i	stage vector of Rosenbrock method
k_1, k_2	spring stiffnesses
ℓ_i	stage vector of Rosenbrock–Nystrom method
L_1, L_2	length of pendulums
m_1, m_2	mass of pendulums
$\mathbf{M}$	system mass matrix
n	number of integration steps
$ndof$	number of degrees of freedom in the mechanical system
N	number of integration steps taken during reference simulation
p	integration order for the Rosenbrock–Nystrom method
$\hat{p}$	integration order for the embedded method
$\mathbf{q}$	vector of generalized coordinates
$\mathbf{Q}^A$	generalized external forces
$\mathbf{r}(i)$	integration grid points
$RelErr$	integration relative error
$Rtol_i$	user prescribed relative tolerance for variable i
sc_i	integration composite tolerance for variable i
$x_1, y_1, \theta_1,$ $\dot{x}_1, \dot{y}_1, \dot{\theta}_1$	initial conditions for first pendulum

K00603 © IMechE 2003

Proc. Instn Mech. Engrs Vol. 217 Part K: J. Multi-body Dynamics

$x_2, y_2, \theta_2,$ $\dot{x}_2, \dot{y}_2, \dot{\theta}_2$	initial conditions for second pendulum
y	generic variable used in the definition of the initial value problem
y_n	solution of Rosenbrock method at t_n
$\hat{y}_n$	numerical solution of the embedded method at t_n
$(\alpha)_{ij}, (\gamma)_{ij},$ $(\delta)_{ij}, (\theta)_{ij}$	Rosenbrock–Nystrom method coefficients
Δ_i	error at time step i
$\Delta^{(k)}, \bar{\Delta}^{(k)}$	maximum and average trajectory errors
Φ	array of position kinematic constraints
α_1^0, α_2^0	zero tension angles for the rotational springs
$(\alpha)_i, (\gamma)_i, (\beta)_{ij},$ $(\beta')_i, (\mu)_i, (\hat{\mu})_i$	coefficients used in the derivation of the Rosenbrock–Nystrom formula
γ	diagonal element of the Rosenbrock formula
λ	vector of Lagrange multipliers
τ	kinematic acceleration equation right-hand side

1   INTRODUCTION

For the dynamic analysis of a mechanical system, this paper presents a method that uses a state–space implicit Rosenbrock-type integrator. The generalized coordinates $\mathbf{q}$ considered are Cartesian coordinates for position, and Euler parameters for orientation of body centroidal reference frames. Without loss of generality and in order to simplify the presentation, an assumption is made that the constraints are holonomic and scleronomic. The kinematic constraints are then formulated as algebraic expressions involving generalized coordinates

$$\Phi(\mathbf{q}) = [\, \Phi_1(\mathbf{q}), \dots, \Phi_m(\mathbf{q}) \,]^{\mathbf{T}} = \mathbf{0} \qquad (1a)$$

where m is the total number of independent constraint equations that must be satisfied by the generalized coordinates throughout the simulation. Differentiating equation (1a) with respect to time leads to the velocity kinematic constraint equation

$$\Phi_{\mathbf{q}}(\mathbf{q})\dot{\mathbf{q}} = 0 \qquad (1b)$$

where the over dot denotes differentiation with respect to time and the subscript denotes partial differentiation. The kinematic acceleration equation is obtained by taking the time derivative of the velocity constraint equations to obtain

$$\Phi_{\mathbf{q}}(\mathbf{q})\ddot{\mathbf{q}} = \tau(\mathbf{q}, \dot{\mathbf{q}}) \qquad (1c)$$

The time evolution of the system is governed by the Lagrange multiplier form of the constrained equations of motion [1]

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \Phi_{\mathbf{q}}(\mathbf{q})^{\mathbf{T}}\lambda = \mathbf{Q}^A(\mathbf{q}, \dot{\mathbf{q}}, t) \qquad (1d)$$

Equations (1a)–(1d) comprise a system of differential-algebraic equations (DAE). The companion paper by Sandu et al. [2] introduced a method based on the partitioning of the coordinates into dependent and independent; the integration of the resulting state–space ordinary differential equation was carried out using a Rosenbrock–Nystrom linearly implicit method. Rosenbrock methods are generally efficient for medium accuracy simulations. They do not require an iteration procedure and have optimal linear stability properties for stiff systems.

This paper introduces an actual Rosenbrock–Nystrom algorithm based on a fourth-order L-stable Rosenbrock method, and discusses a second-order method that can accommodate inexact Jacobians. Numerical experiments for two problems indicate that the Rosenbrock–Nystrom algorithm is reliable and efficient for medium accuracy integration of mechanical systems that lead to stiff state–space ordinary differential equations (SSODE).

2   THE PROPOSED ALGORITHM

For the initial value problem $y' = f(t, y)$, an s-stage Rosenbrock method is defined as [3]

$$y_{n+1} = y_n + \sum_{i=1}^{s} b_i k_i \qquad (2a)$$

$$k_i = hf\left(t_n + \alpha_i h,\, y_n + \sum_{j=1}^{i-1} \alpha_{ij} k_j\right) + \gamma_i h^2 \frac{\partial f}{\partial t}(t_n, y_n) + hJ \sum_{j=1}^{i} \gamma_{ij} k_j \qquad (2b)$$

where the coefficients α, γ and b are chosen to obtain the desired accuracy and stability properties.

For the purpose of error control in the generic Rosenbrock method a second approximation of the solution at the current time step is used to produce an estimate of the local error. This second approximation $\hat{y}_{n+1}$ is usually of lower order and it uses the same stage values k_i with a different set of coefficients $\hat{b}_i$

$$\hat{y}_{n+1} = y_n + \sum_{i=1}^{s} \hat{b}_i k_i \qquad (3)$$

The approximation $|y_{n+1} - \hat{y}_{n+1}|$ of the local error depends on the size of the integration step size, and the latter is increased or decreased to keep the local error below a user prescribed absolute and/or relative tolerance. In the multidimensional case as for the solution of the SSODE,

$\mathbf{y} \in \mathbf{R}^{ndof}$, and at time step $n + 1$ the goal is to keep the error in component i smaller than a composite error tolerance sc_i

$$|y_{n+1}^i - \hat{y}_{n+1}^i| < sc_i, \qquad sc_i = Atol_i + \max(|y_n^i|, |y_{n+1}^i|) \\ \times Rtol_i \qquad (4)$$

where $Atol_i$ and $Rtol_i$ are the user prescribed absolute and relative integration tolerances for component i, $1 \leqslant i \leqslant ndof$. The value

$$err = \left[\frac{1}{ndof} \sum_{i=1}^{ndof} \frac{(y_{n+1}^i - \hat{y}_{n+1}^i)^2}{sc_i^2} \right]^{1/2} \qquad (5)$$

is considered as a measure of local error. If the order of the proper and embedded formulas used is p and $\hat{p}$, respectively, asymptotically $err \approx Ch^{q+1}$, where C is a constant depending on the choice of formulas and $q = \min(p, \hat{p})$. Optimally, $err = 1$ and therefore $1 \approx Ch_{\text{opt}}^{q+1}$. The optimal step size is computed then as

$$h_{\text{opt}} = h \left(\frac{1}{err} \right)^{1/(q+1)} \qquad (6)$$

A safety factor fac multiplies h_{opt} to decrease the chance of a costly rejected step size, which happens whenever $err > 1$. Further, the step size is not allowed to increase or decrease too fast. This is achieved by two control parameters $facmin$ and $facmax$

$$h_{\text{new}} = h \cdot \min \left\{ facmax, \ \max \left[facmin, fac \cdot \left(\frac{1}{err} \right)^{1/(q+1)} \right] \right\} \qquad (7)$$

For most engineering applications, efficient simulation requires expeditious low to medium accuracy methods with very good stability properties. Integration formulas with few function and Jacobian evaluations are favored, since these operations for mechanical system simulation are typically costly. Based on these considerations, the integrator of choice is a four-stage L-stable order four Rosenbrock–Nystrom method, provided with order three embedded formula for step-size control. The L-stability is a desirable attribute that allows for the integration of very stiff problems, which translates in efficient simulation of models with bushing elements and flexible components. Following an idea from reference [2], the number of function evaluations for the four stage method is kept to three; that is, one function evaluation is saved. This makes the proposed Rosenbrock–Nystrom method competitive with the trapezoidal method, whenever the latter requires three or more iterations for convergence. However, the trapezoidal method is of order two and only weakly A-stable.

The notation introduced in reference [4] is going to be used in the derivation of the method coefficients. Thus, $\beta_{ij} = \alpha_{ij} + \gamma_{ij}$, $\beta_i' = \sum_{j=1}^{i-1} \beta_{ij}$, $\alpha_i = \sum_{j=1}^{i-1} \alpha_{ij}$, and $\gamma_i = \sum_{j=1}^i \gamma_{ij}$.

For reasons of computational efficiency the coefficients γ_{ii} are identical for all stages; that is, $\gamma_{ii} = \gamma$ for all $i = 1, \ldots, s$. Note that formally $\alpha_{ii} = 0$, $1 \leqslant i \leqslant s$.

With this, the defining coefficients α_{ij}, γ_{ij}, and b_i of an order four Rosenbrock method of equations (2a and b) are subject to the following order conditions [3]

$$b_1 + b_2 + b_3 + b_4 = 1 \qquad (8a)$$

$$b_2 \beta_2' + b_3 \beta_3' + b_4 \beta_4' = \frac{1}{2} - \gamma \qquad (8b)$$

$$b_2 \alpha_2^2 + b_3 \alpha_3^2 + b_4 \alpha_4^2 = \frac{1}{3} \qquad (8c)$$

$$b_3 \beta_{32} \beta_2' + b_4 (\beta_{42} \beta_2' + \beta_{43} \beta_3') = \frac{1}{6} - \gamma + \gamma^2 \qquad (8d)$$

$$b_2 \alpha_2^3 + b_3 \alpha_3^3 + b_4 \alpha_4^3 = \frac{1}{4} \qquad (8e)$$

$$b_3 \alpha_3 \alpha_{32} \beta_2' + b_4 \alpha_4 (\alpha_{42} \beta_2' + \alpha_{43} \beta_3') = \frac{1}{8} - \frac{\gamma}{3} \qquad (8f)$$

$$b_3 \beta_{32} \alpha_2^2 + b_4 (\beta_{42} \alpha_2^2 + \beta_{43} \alpha_3^2) = \frac{1}{12} - \frac{\gamma}{3} \qquad (8g)$$

$$b_4 \beta_{43} \beta_{32} \beta_2' = \frac{1}{24} - \frac{\gamma}{2} + 1.5\gamma^2 - \gamma^3 \qquad (8h)$$

For the purpose of automatic step-size control, the stage values k_i are reused to provide an embedded formula of order three of the form $\hat{y}_1 = y_0 + \sum_{i=1}^s \hat{b}_i k_i$. The order conditions for the order three algorithm are as indicated as equations (8a–d), and they lead to the system

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & \beta_2' & \beta_3' & \beta_4' \\ 0 & \alpha_2^2 & \alpha_3^2 & \alpha_4^2 \\ 0 & 0 & \beta_{32}\beta_2' & \beta_{42}\beta_2' + \beta_{43}\beta_3' \end{bmatrix} \begin{bmatrix} \hat{b}_1 \\ \hat{b}_2 \\ \hat{b}_3 \\ \hat{b}_4 \end{bmatrix} = \begin{bmatrix} 1 \\ \dfrac{1}{2} - \gamma \\ \dfrac{1}{3} \\ \dfrac{1}{6} - \gamma + \gamma^2 \end{bmatrix} \qquad (9)$$

If the coefficient matrix in equation (9) is nonsingular, uniqueness of the solution of this linear system implies $b_1 = \hat{b}_i$. To prevent this, one additional condition is considered to obtain a distinct order three embedded formula. It requires the coefficient matrix in equation (9) to be singular, which results in

$$\beta_{32}\beta_2'(\beta_2'\alpha_4^2 - \beta_4'\alpha_2^2) = (\beta_2'\alpha_3^2 - \beta_3'\alpha_2^2)(\beta_{42}\beta_2' + \beta_{43}\beta_3') \qquad (10)$$

The number of coefficients that must be determined is 17; the diagonal coefficient γ, six coefficients γ_{ij}, six coefficients α_{ij}, and four weights b_i. The number of conditions that these coefficients have to satisfy is nine. There are eight degrees of freedom in the choice of coefficients and some of these

Table 1 Coefficients for the Rosenbrock method

$\gamma = 0.57281606$	
$\alpha_{21} = 1.14563212$	$\gamma_{21} = 2.3419931271120139491770520$
$\alpha_{31} = 0.52092078913062902933516$	$\gamma_{31} = -0.0273337465434898361996505$
$\alpha_{32} = 0.13429418684250480019232$	$\gamma_{32} = 0.2138116508366996896867472$
$\alpha_{41} = 0.52092078913062902933516$	$\gamma_{41} = -0.2590838377855102221112641$
$\alpha_{42} = 0.13429418684250480019232$	$\gamma_{42} = -0.1905958077323117516166358$
$\alpha_{43} = 0.0$	$\gamma_{43} = -0.2280310359731338294777744$
$b_1 = 0.3245347078917345134174196$	$\hat{b}_1 = 0.52092078913062902928516$
$b_2 = 0.0490865544787523308684633$	$\hat{b}_2 = 0.1445497146653645995884681$
$b_3 = 0.0$	$\hat{b}_3 = 0.1245596864147020499774897$
$b_4 = 0.6263787473207421778411171$	$\hat{b}_4 = 0.2099698097893043213111906$

are used to construct a method with one less function evaluation. Thus, if

$$\alpha_{41} = \alpha_{31}, \qquad \alpha_{42} = \alpha_{32}, \qquad \alpha_{43} = 0 \qquad (11)$$

stage 4 of the algorithm saves one function evaluation. Finally, the free parameters can be determined such that several order five conditions of the otherwise order four formula are satisfied. When the conditions of equation (11) hold, one of the nine order five conditions associated with a Rosenbrock-type formula leads to

$$\alpha_3 = \frac{1/5 - \alpha_2/4}{1/4 - \alpha_2/3} \qquad (12)$$

A second order five condition is satisfied by imposing the condition

$$b_4\beta_{43}\alpha_3^2(\alpha_3 - \alpha_2) = \frac{1}{20} - \frac{\gamma}{4} - \alpha_2\left(\frac{1}{12} - \frac{\gamma}{3}\right) \qquad (13)$$

Next, two conditions are chosen as

$$b_3 = 0, \qquad \alpha_2 = 2\gamma \qquad (14)$$

to make the task of finding the defining coefficients α_{ij}, γ_{ij}, and b_i more tractable. Finally, the last condition regards the choice of the diagonal element γ. The value of this parameter determines the stability properties of the Rosenbrock method. In this context, the diagonal entry of the Rosenbrock formula is suggested in reference [3] as $\gamma = 0.57281606$, which is adopted for the proposed algorithm. With this, there is a set of 17 equations, some of them nonlinear, in 17 unknowns. The solution of this system is provided in Table 1, along with the coefficients $\hat{b}_i$ of the order three embedded formula.

Once the coefficients of the underlying Rosenbrock formula are available, the coefficients of the Rosenbrock–Nystrom formula defined in the companion paper [2] are easily computed. The full set of coefficients for the order four, L-stable formula is provided in Table 2.

It should be recalled that any Rosenbrock-type formula requires an exact Jacobian for the numerical solution to

Table 2 Coefficients for the Rosenbrock–Nystrom method

$\theta_{21} = 1.14563212$	$a_{21} = 0.20000000000000000000000$
$\theta_{31} = 0.7895091628156386296226980$	$a_{31} = 1.8679481494982371323414476$
$\theta_{32} = 0.13429418684250480019232$	$a_{32} = 0.2344455685172388500023222$
$\theta_{41} = 0.7895091628156386296226980$	$a_{41} = 1.8679481494982371323414476$
$\theta_{42} = 0.13429418684250480019232$	$a_{42} = 0.2344455685172388500023222$
$\theta_{43} = 0.0$	$a_{43} = 0.0$
$c_{21} = -7.1376499433499798303369260$	$\delta_{21} = -1.1963610071120139491770520$
$c_{31} = 2.5809236665096577144880050$	$\delta_{31} = 1.4702802544097807143663870$
$c_{32} = 0.6516298873020320233387417$	$\delta_{32} = 0.3481058376792044900016704$
$c_{41} = -2.1371152665066191168063700$	$\delta_{41} = 0.0037650943555561657598974$
$c_{42} = -0.3214695313399510707690241$	$\delta_{42} = -0.1097624867581032555675398$
$c_{43} = -0.6949660492824452251557329$	$\delta_{43} = -0.2280310359731338294777744$
$m_1 = 2.2555662286045652437288840$	$\hat{m}_1 = 2.0683991605275837342583670$
$m_2 = 0.2870550631941576076762630$	$\hat{m}_2 = 0.2386813520675327979755493$
$m_3 = 0.4353119633799832131402707$	$\hat{m}_3 = 0.3633733454353917082661747$
$m_4 = 1.0935076564032478032114820$	$\hat{m}_4 = 0.3665571279361551443009163$
$\mu_1 = 1.5927508194095853421074900$	$\hat{\mu}_1 = 1.4349039718482094726227100$
$\mu_2 = 0.1959382663102506096593329$	$\hat{\mu}_2 = 0.2229786725886983690045153$
$\mu_3 = 0.0$	$\hat{\mu}_3 = 0.1245596864147020499774897$
$\mu_4 = 0.6263787473207421778411171$	$\hat{\mu}_4 = 0.2099698097893043213111906$
$\gamma_2 = -1.7691770671120139491770520$	$a_2 = 1.145632120$
$\gamma_3 = 0.7592939642932098536570967$	$a_3 = 0.6552149759731338294777748$
$\gamma_4 = -0.1048946214909558032206743$	$a_4 = 0.6552149759731338294777748$

Table 3 Coefficients for the W-method

| $\gamma = 1.70710678118650$	
$\alpha_1 = 0.00000000000000$	$\gamma_1 = 1.70710678118650$
$\alpha_2 = 1.00000000000000$	$\gamma_2 = -1.70710678118650$
$a_{21} = 0.58578643762690$	$c_{21} = -1.17157287525380$
$\delta_{11} = 1.70710678118650$	$\delta_{22} = 1.70710678118650$
$\delta_{21} = -2.41421356237310$	
$\theta_{21} = 1.00000000000000$	
$m_1 = 0.87867965644040$	$\hat{m}_1 = 1.17157287525380$
$m_2 = 0.29289321881340$	$\hat{m}_2 = 0.58578643762690$
$\mu_1 = 0.79289321881340$	$\hat{\mu}_1 = 0.58578643762690$
$\mu_2 = 0.50000000000000$	$\hat{\mu}_2 = 1.00000000000000$

maintain its stability and accuracy properties. Sometimes this might be a very challenging requirement. Consider, for example, the situation when complex tire models are present in a model, or for a general purpose solver the case when user-defined external routines are employed for the computation of active forces such as aerodynamic forces. Verwer *et al.* in 1999 [**5**] proposed a second-order W-method [**4**], which is a Rosenbrock type method in the sense that it does not necessitate the solution of a non-linear system, but which does not require an exact Jacobian. The defining coefficients for this method are provided in Table 3, and within the proposed implicit SSODE integration framework they can be used for the numerical solution of the index three DAE of multibody dynamics.

3 NUMERICAL EXPERIMENTS

This paper discusses the implementation of a Rosenbrock–Nystrom method based on Algorithm 1 of reference [**2**] and the four stage, order four L-stable Rosenbrock formula introduced in the previous section. Note that a W-method can be similarly implemented by replacing the corresponding coefficients of the Rosenbrock–Nystrom formula with the appropriate W-coefficients of Table 3. A set of numerical experiments is first carried out to validate the proposed Rosenbrock–Nystrom method. Then a comparison with an explicit integrator is performed to assess the efficiency of the proposed algorithm for numerical integration of a more complex mechanical system.

3.1 Validation of proposed algorithm

Validation is carried out using the double pendulum mechanism shown in Fig. 1. Stiffness is induced by means of two rotational spring-damper-actuators (RSDA). The masses of the two pendulums are $m_1 = 3$ and $m_2 = 0.3$, the dimension of the pendulums are $L_1 = 1$ and $L_2 = 1.5$, the stiffness coefficients are $k_1 = 400$ and $k_2 = 3.E5$, and the damping coefficients are $c_1 = 15$ and $c_2 = 5.E4$. The zero-tension angles for the two RSDA elements are $\alpha_1^0 = 3\pi/2$ and $\alpha_2^0 = 0$. All units are SI.
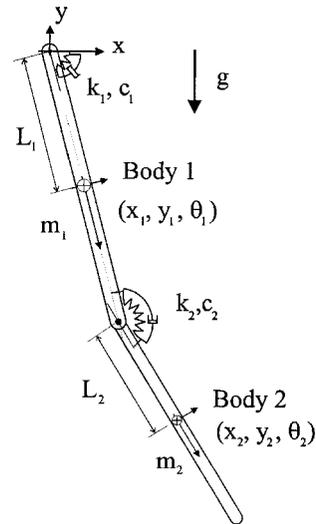
Fig. 1 Double pendulum problem

In its initial configuration, the two degree of freedom dynamic system has a dominant eigenvalue with a small imaginary part and a real part of the order $-10E5$. Since the two pendulums are connected through two parallel revolute joints the problem is planar. In terms of initial conditions, the centers of mass (CM) of bodies 1 and 2 are located at $x_1^{CM} = 1$, $y_1^{CM} = 0$, and $x_2^{CM} = 3.4488887$, $y_2^{CM} = -0.388228$. In the initial configuration, the centroidal principal reference frame of body 1 is parallel with the global reference frame, while the centroidal principal reference frame of body 2 is rotated with $\theta_2 = 23\pi/12$ around an axis perpendicular on the plane of motion. For body 1, $\dot{x}_1^{CM} = \dot{y}_1^{CM} = \dot{\theta}_1^{CM} = 0$, while for body 2, $\dot{x}_2^{CM} = 3.8822857$, $\dot{y}_2^{CM} = 14.4888887$, and $\dot{\theta}_2^{CM} = 10$. All initial conditions are in SI units, and are consistent with the kinematic constraint equations at position and velocity levels [equations (1a and b)].

The first set of numerical experiments focuses on assessing the reliability of the step-size control mechanism. The goal is to verify that user-imposed levels of absolute and relative error are met by the simulation results. A *reference simulation* is first run using a very small constant integration step size. Other simulations, run with different combinations of absolute and relative tolerances, are compared to the *reference simulation* to find the infinity norm of the error, the time at which this largest error occurred, and average error per time step. Suppose that n time steps are taken during the current simulation and that the variable whose accuracy is analysed is denoted by e. The grid points of the current simulation are denoted by $t_{\text{init}} = t_1 < t_2 < \cdots < t_n = t_{\text{end}}$. If N is the number of time steps taken during the *reference simulation*, that is, $T_{\text{init}} = T_1 < T_2 < \cdots < T_N = T_{\text{end}}$, assume that for the quantity of interest the computed reference values are E_j, for $1 \leqslant j \leqslant N$. For each $1 \leqslant i \leqslant n$, an integer $\mathbf{r}(i)$ is defined such that $T_{\mathbf{r}(i)} \leqslant t_i \leqslant T_{\mathbf{r}(i)+1}$. Using the reference values $E_{\mathbf{r}(i)-1}$, $E_{\mathbf{r}(i)}$, $E_{\mathbf{r}(i)+1}$, and $E_{\mathbf{r}(i)+2}$, cubic spline interpolation algorithm is used to generate an interpolated value E_i^* at time t_i. If $\mathbf{r}(i) - 1 \leqslant 0$,

the first four reference points are considered for interpolation, while if $\mathbf{r}(i) + 2 \geqslant N$, the last four reference points are used for interpolation. The error at time step i is defined as $\Delta_i = |E_i^* - e_i|$. For each tolerance set k, accuracy is measured by both the maximum $\Delta^{(k)}$ and the average $\bar{\Delta}^{(k)}$ trajectory errors, as well as by the percentage relative error

$$\Delta^{(k)} = \max_{1 \leqslant i \leqslant n} (\Delta_i), \qquad \bar{\Delta}^{(k)} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} \Delta_i^2},$$

$$RelErr[\%] = \frac{\Delta^{(k)}}{E^*} \times 100$$

where $E^* = E_p$, with p defined such that $\Delta^{(k)} = \Delta_p$. Simulations are run for tolerances between 10^{-2} and 10^{-5}, a range that typically covers mechanical engineering accuracy requirements. The length of the simulation is 2 seconds. The time variation of the angle θ_1 is presented on the left of Fig. 2. Note that body 1 eventually stabilizes in the configuration $\theta_1 = 3\pi/2$, which is the zero-tension angle for the RSDA.

Table 4 contains error analysis information for angle θ_1. The first column contains the value of the tolerance with which the simulation is run. Relative and absolute tolerances [$Rtol_i$ and $Atol_i$ of equation (4)] are set to 10^k, and they are applied for both position and velocity error control. The second column contains the time t^* at which the largest error $\Delta^{(k)}$ occurred. The third column contains the values of $\Delta^{(k)}$. Column four contains the relative error, and the last column shows the average trajectory error. Table 5 shows the number of integration steps selected by the numerical integrator for different values of the tolerance parameter k.

The most relevant information for step-size control validation is $\Delta^{(k)}$. If, for example, $k = -3$, that is, accuracy of the order 10^{-3} is demanded, $\Delta^{(-3)}$ should have this order of magnitude. It can be seen from the results in Table 4 that this is the case for all tolerances. Note that these results are

Table 4   Position error analysis for the double pendulum problem

k	t^*	$\Delta^{(k)}$	$RelErr$ [%]	$\bar{\Delta}^{(k)}$
-2	0.592127	5.223e$-$2	0.12126	3.234e$-$3
-3	0.599954	4.198e$-$3	0.00964	2.631e$-$4
-4	0.626135	4.196e$-$4	0.00108	2.946e$-$5
-5	1.065146	1.902e$-$5	0.00039	9.868e$-$6

obtained with a nonzero relative tolerance. According to equation (4), depending on the magnitude of the variable being analysed, the relative tolerance directly impacts the step-size control. Based on the position results shown in Fig. 2, the relative tolerance is multiplied by a value that for θ_1 oscillates between 4.0 and 6.0. Consequently, the actual upper bound of accuracy imposed on θ_1 fluctuates and reaches values up to 7×10^{-k}. Thus, the step-size controller is slightly conservative. For an explanation of this stiffness-induced order reduction, the reader is referred to references [**3**] or [**6**]. In the latter reference, the local truncation error $(\hat{y}_1 - y_1)$ in equation (4) is replaced by the scaled value $\delta = (I - h\gamma\partial f/\partial y)^{-1}(\hat{y}_1 - y_1)$. This step-size control strategy remains to be investigated.

Error analysis is also performed at the velocity level. The time variation of angular velocity $\dot{\theta}_1$ is shown in Fig. 2. The angular velocity of body 1 fluctuates between -10 and 7 rad/s. As a result, values of $\Delta^{(k)}$ of up to the order 10^{k+1} are considered very good. Error analysis results for $\dot{\theta}_1$ are presented in Table 6. The step-size controller performs well, slightly on the conservative side.

The error analysis results presented suggest that the step-size controller employed is reliable, as the pre-imposed accuracy requirements are met or exceeded by the numerical results. In order to avoid unjustified CPU penalties, the algorithm may be improved for extremely stiff mechanical systems by adopting a modified step-size controller proposed in reference [**6**].
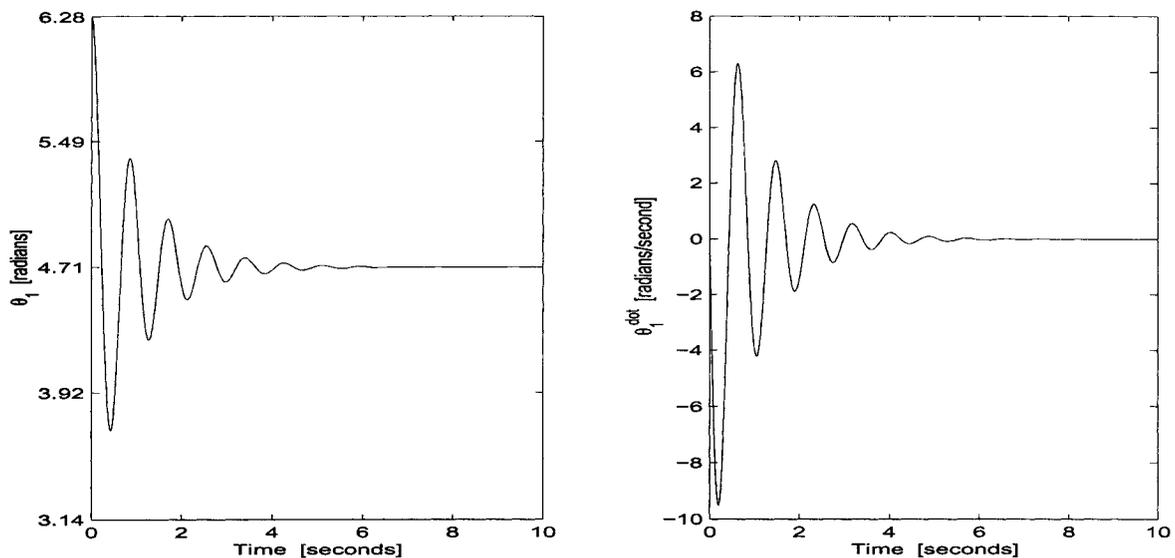

Fig. 2   Time variation of orientation θ_1 (left) and of angular velocity $\dot{\theta}_1$ (right) for body 1

**Table 5** Number of integration time steps for the double pendulum problem

k	-2	-3	-4	-5	-6	-7
Steps	29	49	85	148	264	467

Table 6 Velocity error analysis for the double pendulum problem

k	t^*	$\Delta^{(k)}$	$RelErr$ [%]	$\bar{\Delta}^{(k)}$
-2	0.795548	4.061e$-$2	1.84434	2.348e$-$2
-3	0.373114	3.792e$-$3	0.12340	2.181e$-$3
-4	0.217757	8.652e$-$4	0.00922	3.445e$-$4
-5	0.186183	2.343e$-$4	0.00246	9.357e$-$5

3.2 Performance comparison with explicit integrator

In order to compare the performance of the proposed implicit algorithm with an SSODE algorithm based on a state-of-the-art explicit integrator [6], a model of the US Army High Mobility Multipurpose Wheeled Vehicle (HMMWV) is considered for dynamic analysis. The HMMWV shown in Fig. 3 is modeled using 14 bodies. In this figure, vertices represent bodies, while edges represent joints connecting the bodies of the system. Thus, vertex number 1 is the chassis, 2 and 5 are the right and left front upper control arms, 3 and 6 are the right and left front lower control arms, 9 and 12 are the right and left rear lower control arms, and 8 and 11 are the right and left rear upper control arms. Bodies 4, 7, 10, and 13 are the wheel spindles, and body 14 is the steering rack. Spherical joints are denoted by S, revolute joints by R, distance constraints by D, and translational joints by T. This set of joints imposes 79 constraint equations. One additional constraint equation is imposed on the steering system, such that the steering angle is zero, that is, the vehicle drives straight. A total of 98 generalized coordinates are used to model the vehicle, which renders 18 degrees of freedom to the model.
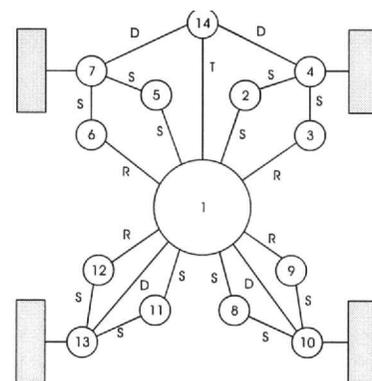
Stiffness is induced in the model by means of four translational spring-damper actuators (TSDA). These TSDAs act between the front/rear and right/left upper control arms and the chassis. The stiffness coefficient of each TSDA is 2E7 N/m, while the damping coefficient is 2E6 N·s/m. For the purpose of this numerical experiment, the tires of the vehicle are modeled as vertical TSDA elements with stiffness coefficient 296 325 N/m and damping coefficient 3502 N·s/m. Finally, the dominant eigenvalue of the corresponding SSODE has a real component of approximately -2.6E5, and a small imaginary part.

Dynamic analysis of the model is carried out for the vehicle driving straight at 10 mph over a bump. The shape of the bump is a half-cylinder of diameter 0.1 m. Figure 4 shows the time variation of the vehicle chassis height. The front wheels hit the bump at time 0.5 s, and the rear wheels hit the bump at time 1.2 s. The length of the simulation in this plot is 5 s. Toward the end of the simulation (after 4 s), due to over-damping, the chassis height stabilizes at approximately $z_1 = 0.71$ m.

The test problem is first run with an explicit integrator based on the code DEABM [6]. Algorithm 1 below outlines the explicit integration approach used for SSODE integration of the equations of motion for the HMMWV model. The first three steps are identical to the ones in Algorithm 1 in reference [2]. Step 4 computes the acceleration $\ddot{\mathbf{q}}$, by solving the linear system of equation (1d). A topology-based approach [7], that takes into account the sparsity of the coefficient matrix is used to solve for the generalized accelerations $\ddot{\mathbf{q}}$. The DDEABM integrator is then used to integrate for independent velocities $\dot{\mathbf{v}}_n$, and independent positions $\mathbf{v}_n$ [2]. The integrator is also used to integrate for the dependent coordinates $\mathbf{u}_n$, with the sole purpose of providing a good starting point during Step 6 that computes $\mathbf{u}_n$ by ensuring that the kinematic position constraint equations are satisfied; that is, solving $\Phi(\mathbf{v}_n, \mathbf{u}_n) = \mathbf{0}$. Likewise, dependent velocities $\dot{\mathbf{u}}_n$ are the solution of the linear system $\Phi_{\mathbf{u}}(\mathbf{u}_n, \mathbf{v}_n)\dot{\mathbf{u}}_n = -\Phi_{\mathbf{v}}(\mathbf{u}_n, \mathbf{v}_n)\dot{\mathbf{v}}_n$ [2], which thus guarantees that the generalized velocities satisfy the kinematic velocity constraint equations. The dependent/independent partitioning of the generalized coordinates is checked during Step 7.
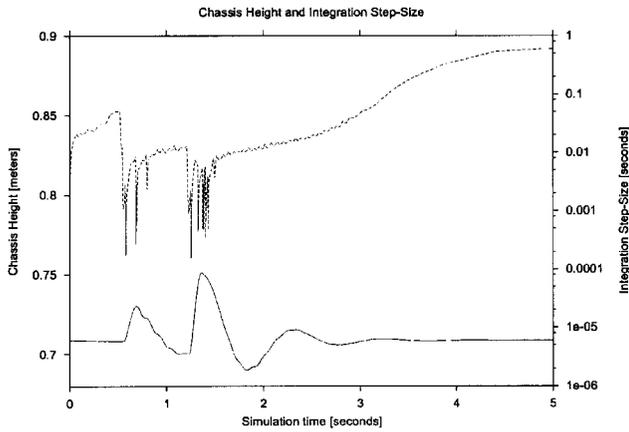

Fig. 3 The HMMWV (left) and its topology representation (right)

Fig. 4 Chassis height and integration step size

Algorithm 1

1. Initialize simulation
2. Set integration tolerance
3. While (time < time-end) do
4.   Get acceleration
5.   Apply integration step
     Check accuracy. Determine new step-size
6.   Recover dependent generalized coordinates
7.   Check partition
8. End do

Timing results reported are obtained on an SGI Onyx computer with an R10000 processor. Computer times required by Algorithm 1 are listed in Table 7. Results for the Rosenbrock–Nystrom algorithm are presented in Table 8.

The results in Table 7 are typical for the situation when an explicit integrator is used for the numerical solution of a stiff IVP. For the stiff test problem considered, the performance limiting factor is stability of the explicit code. For a given simulation length, any tolerance in the range 1E−2 through 1E−5 results in almost identical CPU times. The average explicit integration step size turns out to be between 1E−5 and 1E−6, and it is not affected by accuracy requirements. The code is compelled to select very small step sizes to assure stability of the integration process, and this is the criteria for step size selection for a broad spectrum of tolerances. Only if extreme accuracy is imposed does the step size become limited based on accuracy considerations. In this context, note that the results in Table 8 indicate that stability is of no concern for the proposed algorithm, and solution accuracy solely determines the duration of the

Table 8 Implicit integrator timing results for the HMMWV problem

Tol	10^{-2}	10^{-3}	10^{-4}	10^{-5}
1 s	5.6	13.2	40.7	172
2 s	12.6	32.6	95	405
3 s	13	36.3	105	422
4 s	13.3	37	106	428

simulation. The integration step size is automatically adjusted to keep integration error within the user prescribed limits. Figure 4 shows the time variation for the integration step size when the absolute and relative errors at position and velocity levels are set to 10^{-3}. The y-axis for the step size is provided at the right of Fig. 4, on a logarithmic scale. In the lower half of the same figure, relative to the left y-axis is provided the time variation of the chassis height. Note that when the vehicle hits the bump; that is, when in Fig. 4 the z-coordinate of the chassis increases suddenly, the step size is simultaneously decreased to preserve accuracy of the numerical solution. On the other hand, for the region in which the road becomes flat, that is, toward the end of the simulation, the integrator is capable of taking larger integration steps, thus decreasing simulation time.

4 CONCLUSIONS

A generalized coordinate partitioning based state–space implicit integration method is introduced for the dynamic analysis of multibody systems. In the companion paper of reference [2] the derivation of a Rosenbrock–Nystrom family of methods for state–space implicit integration was presented. Based on this, the paper defines a particular method based on a four-stage order four L-stable Rosenbrock formula that has an order three embedded formula for error control. For a 14-body 18-degree-of-freedom vehicle, the proposed method is almost two orders of magnitude faster than an explicit integrator-based method. The most restrictive condition associated with the use of the Rosenbrock formula employed is the requirement of an exact integration Jacobian. In this context, a formalism for systematically computing the state–space integration Jacobian is presented in reference [2]. When providing an exact Jacobian is not feasible, a lower order W-method is suggested as an alternative.

Table 7 Explicit integrator timing results for the HMMWV problem

Tol	10^{-2}	10^{-3}	10^{-4}	10^{-5}
1 s	3618	3641	3667	3663
2 s	7276	7348	7287	7276
3 s	10865	11122	10949	10965
4 s	14480	14771	14630	14592

REFERENCES

1 Haug, E. J. *Computer-Aided Kinematics and Dynamics of Mechanical Systems*, 1989 (Allyn and Bacon, Boston, London, Sydney, Toronto).

2 Sandu, A., Negrut, D., Haug, E. J., Potra, F. A. and **Sandu, C.** A Rosenbrock–Nystrom state space implicit approach for the dynamic analysis of mechanical systems: I—theoretical formulation. *Proc. Instn Mech. Engrs, Part K: J. Multi-body Dynamics*, 2003, **217**, 263–271.

3 Hairer, E. and **Wanner, G.** *Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems*, 1996 (Springer-Verlag, Berlin Heidelberg, New York).

4 Hairer, E., Norsett, S. P. and **Wanner, G.** *Solving Ordinary Differential Equations I. Nonstiff Problems*, 1993 (Springer-Verlag, Berlin Heidelberg, New York).

5 Verwer, J. G., Spree, E. J., Bloom, J. G. and **Hundsdorfer, W.** A second order Rosenbrock method applied to photochemical problems. *SIAM J. Sci. Comput.*, 1999, **20**, 1456–1480.

6 Shampine, L. F. and **Watts, H. A.** The art of writing a Runge–Kutta code II. *Appl. Math. Comput.*, 1979, **5**, 93–121.

7 Serban, R., Negrut, D., Haug, E. J. and **Potra, F. A.** A topology based approach for exploiting sparsity in multibody dynamics in cartesian formulation. *Mech. Struct. Mach.*, 1997, **25**(3), 379–396.