AIDAA
XVI CONGRESSO NAZIONALE AIDAA
24–28 settembre 2001
PALERMO

MULTISTEP INTEGRATION OF ORDINARY, STIFF AND DIFFERENTIAL-ALGEBRAIC PROBLEMS FOR MULTIBODY DYNAMICS APPLICATIONS

P. MASARATI†, M. LANZ, P. MANTEGAZZA

Dipartimento di Ingegneria Aerospaziale, Politecnico di Milano, Milano

ABSTRACT

The paper presents a family of integration formulæ with different accuracy order, dedicated to the solution of initial value (IV) problems in form of Ordinary Differential Equations (ODE). Its extensions to implicit ODEs and to Differential-Algebraic Equations (DAE) is discussed, to focus on the analysis of multibody, multidisciplinary problems. The main features of the formulæ are discussed, and numerical solutions are proposed that illustrate their appropriateness to such problems.

1. INTRODUCTION

The solution of IV problems is a well-established topic in mathematics [1]; nonetheless the practical solution of large numerical problems has captured the attention of a number of researchers. While general purpose algorithms have been considered for years the preferable way to integrate systems of differential equations, recently, the idea of requiring the integrator to preserve some invariant properties of the problem to be integrated arose, resulting in the development of the so called “Energy Preserving” and “Energy Dissipating” algorithms [2]. The latter approach results in more stable integration formulas for mechanical systems, but it requires an *a priori* knowledge of the invariant properties of the problem, which may not even exist when multidisciplinary systems are considered. As a consequence, there is still a need for robust general purpose algorithms with trade-off accuracy and stability properties that can handle broadly different problems simultaneously

†Speaker, corresponding author, via La Masa 34, 20156, Milano, Italy
Tel.: +39 02 2399 8309 E-mail: *masarati@aero.polimi.it*
Fax: +39 02 2399 8334 http://www.aero.polimi.it

[3]. This work originated from the need to perform unconditionally stable integration of IV problems in the field of multibody dynamics, with particular regard to rotorcraft modeling and aerospace applications [4]. The basic algorithm is introduced and its accuracy and stability properties are discussed in the linear case. An extension to implicit ODEs and to DAEs is presented, with the aim of finding a good trade-off between performance in terms of accuracy, and low implementation and computational costs. The proposed method closely resembles the more general Implicit *Runge-Kutta* (IRK) methods for the integration of differential equations [1]. A 4^{th} order accurate, single-step A-stable algorithm represents the most performant among the proposed formulæ, its main drawback being the high computational cost of evaluating two Jacobian matrices and of performing one Jacobian multiplication for each step. The introduction of algorithmic dissipation, although reducing accuracy to the 3^{rd} order, gives a lot of flexibility in integrating stiff problems and is fundamental when DAEs are considered. The 2^{nd} order two-step formulæ, on the other hand, allow the introduction of algorithmic dissipation up to L-stability while preserving the 2^{nd} order accuracy of the non-dissipating case. The main advantage with respect to higher-order schemes is that the solution of the problem is accomplished only at the end of the time-step, requiring the computation of a single Jacobian with no Jacobian multiplications. One drawback is that the formula is not self-starting, since two time steps are required for the prediction. The problem has been overcome by using a conventional Crank-Nicholson start that preserves the order of accuracy. The two-step formula has been successfully applied to the integration of DAEs in the framework of a multibody dynamics analysis formulation. Numerical examples are proposed, to specifically illustrate the properties of the integration formulæ and to highlight their limitations as well as to show the typical problems they can successfully solve.

2. INTEGRATION SCHEME

2.1. Definitions

Consider an ODE in explicit form, and an IV problem:

$$\dot{x} = f(x,t), \qquad x(t_0) = x_0. \tag{1}$$

The unknown x is approximated by a polynomial interpolation ranging from r time steps backward, $k-r$, to the current time step, k, being x_k the unknown solution:

$$x = \sum_{i=0,r} \left(m_i(\xi)\, x_{k-i} + h n_i(\xi)\, \dot{x}_{k-i} \right), \tag{2}$$

where h is the time step, ξ is the non-dimensional time; the origin for the ξ is at the current time step, i.e. $\xi = (t - t_k)/h$; m_i, n_i are the polynomial shape functions related to x_i, $\dot{x}_i$, to consider first-order problems. The integration is carried on by weighing the value of the derivative at $s+1$ appropriate points:

$$x_k = x_{k-r} + \int_{t_{k-r}}^{t_k} f(x,t)\, dt \cong x_{k-r} + h \sum_{j=0,s} w_j f(x(t_j), t_j), \tag{3}$$

where the problem $f(x,t)$, Equation (1), has been used *in lieu* of $\dot{x}$. The perturbation of the numerical solution, Equation (3), to be used in a *Newton-Raphson* iterative solution,

$$x_k + \Delta x_k = x_{k-r} + h \sum_{j=0,s} w_j \left(f\left(x\left(t_j\right), t_j\right) + \frac{\partial f\left(x\left(t_j\right), t_j\right)}{\partial x} \frac{\partial x\left(t_j\right)}{\partial x_k} \Delta x_k \right),$$

where

$$\frac{\partial x\left(t_j\right)}{\partial x_k} = m_0\left(\xi_j\right) + h n_0\left(\xi_j\right) \frac{\partial f\left(x_k, t_k\right)}{\partial x}$$

descends from the interpolation formula of Equation (2); after substitution it becomes:

$$\left(I - h \sum_{j=0,s} w_j \frac{\partial f\left(x\left(t_j\right), t_j\right)}{\partial x} \left(m_0\left(\xi_j\right) I + h n_0\left(\xi_j\right) \frac{\partial f\left(x_k, t_k\right)}{\partial x} \right) \right) \Delta x_k$$

$$= -\left(x_k - x_{k-1}\right) + h \sum_{j=0,s} w_j f\left(x\left(t_j\right), t_j\right).$$

The key idea is that given the order of the interpolation for the x, the $\dot{x}$ is one order less, thus determining the minimum required order for the numerical integration. However, the higher the order of the numerical integration, the higher the order of accuracy.

2.2. Stability

The linear stability properties of the algorithm can be investigated by considering the problem $\dot{x} = \lambda x$, where λ is an arbitrary complex parameter. For a homogeneous difference equation the solution at time t_k can be seen as $x(t_k) = \rho x(t_{k-1})$, and by substituting the assumed value of $\dot{x}$ in Equation (3), one obtains

$$\rho^r = 1 + h \sum_{j=0,s} w_j \lambda \sum_{i=0,r} \left(m_i(\xi_j)\, \rho^{r-i} + h n_i(\xi_j)\, \lambda \rho^{r-i} \right),$$

which, by exchanging the order of summations over i and j, yields

$$\rho^r = 1 + \lambda h \sum_{i=0,r} \rho^{r-i} \left(\sum_{j=0,s} w_j m_i(\xi_j) + \lambda h \sum_{j=0,s} w_j n_i(\xi_j) \right), \tag{4}$$

namely a polynomial in ρ of order r, which may be linear or quadratic in λh, depending on the value of the shape functions n_j. The formula is A-stable if all the r roots satisfy the inequality $|\rho| \leq 1$, with $\mathrm{Re}(\lambda) < 0$, for any λh. The formula is L-stable if in addition the numerator of all the roots is one order less than the denominator in λh. If they are of the same order, then the ratio of the highest order terms of the numerator and the denominator represents the asymptotic value of the spectral radius; its norm is comprised between 1 and 0, and the formula has a behavior between A- and L-stable, because it adds some algorithmic dissipation. In the single-step case ($r = 1$) the root is

$$\rho = \frac{1 + \lambda h \left(\sum_j w_j m_1(\xi_j) + \lambda h \sum_j w_j n_1(\xi_j) \right)}{1 - \lambda h \left(\sum_j w_j m_0(\xi_j) + \lambda h \sum_j w_j n_0(\xi_j) \right)}.$$

Thus the formula is L-stable provided $\sum_j w_j n_0(\xi_j) > 0$ and $\sum_j w_j n_1(\xi_j) = 0$, or, in case $\sum_j w_j n_0(\xi_j) = 0$, for $\sum_j w_j m_0(\xi_j) > 0$ and both $\sum_j w_j n_1(\xi_j) = 0$, $\sum_j w_j m_1(\xi_j) = 0$. Similar conditions can be analytically formulated for two-step methods.

3. HIGHER-ORDER FORMULÆ

Higher-order formulas can be obtained by considering the case of a single step ($r = 1$) with appropriate interpolation and intermediate integration points.

3.1. Third-order accuracy with parabolic interpolation

Consider now a single-step parabolic interpolation, with shape functions:

$$\begin{aligned} m_0 &= 1 - \xi^2, & m_1 &= \xi^2, \\ n_0 &= \xi\left(1 + \xi\right), & n_1 &= 0. \end{aligned}$$

The interpolation discards the slope of the solution at the beginning of the time step. An *ad hoc* integration scheme must be envisaged to enforce the 3rd order accuracy, given by a two-point collocation. The appropriate choice is

point, ξ_j	ζ	0
weight, w_j	$-1/\left(2\zeta\right)$	$\left(2\zeta+1\right)/\left(2\zeta\right)$

with $\zeta = -2/3$. The symmetric combination, $\xi = \{-1, -1/3\}$, yields the same accuracy and stability properties, but lacks the very important property of collocating exactly at the end of the interval, where the unknown is computed. The integration becomes:

$$x_k = x_{k-1} + h\left(-\frac{1}{2\zeta}\left.f\right|_{\xi=\zeta} + \frac{2\zeta+1}{2\zeta}\left.f\right|_{\xi=0}\right),$$

and the linearization yields:

$$\left(I - h\left(-\frac{1}{2\zeta}\left.\frac{\partial f}{\partial x}\right|_{\xi=\zeta}\left(\left(1-\zeta^2\right)I + \zeta\left(1+\zeta\right)h\left.\frac{\partial f}{\partial x}\right|_{\xi=0}\right) + \frac{2\zeta+1}{2\zeta}\left.\frac{\partial f}{\partial x}\right|_{\xi=0}\right)\right)\Delta x_k$$

$$= -\left(x_k - x_{k-1}\right) + h\left(-\frac{1}{2\zeta}\left.f\right|_{\xi=\zeta} + \frac{2\zeta+1}{2\zeta}\left.f\right|_{\xi=0}\right).$$

The linear stability analysis yields

$$x_k = \frac{\left(1 - \frac{\zeta}{2}\lambda h\right)}{\left(1 - \frac{\zeta+2}{2}\lambda h + \frac{\zeta+1}{2}\left(\lambda h\right)^2\right)} x_{k-1}.$$

The integrator is L-stable except for $\zeta = -1$, where it yields once more the *Crank-Nickolson* rule. The accuracy is cubic for the algorithmic dissipation, and 4th order for the phase error when the system is undamped, otherwise it is cubic for both errors. It decreases as the time step increases, and with the value of the real part of λ; the accuracy is strongly problem and time step dependent. This is true when the "optimal" point $\zeta = -2/3$, which allows the exact integration of a 2nd order polynomial, is used. The same spectral radius can be obtained by using the symmetric combination of collocation points, namely the starting point of the time step, $\xi = -1$, and the point at $\xi = -1 - \zeta$. When a value of ζ other than $-2/3$ is used, the accuracy is quadratic, and the formula loses its appealing higher-order properties.

3.2. Higher-order accuracy with tunable algorithmic dissipation

COnsider a cubic interpolation, consisting in *Hermitian* shape functions:

$$\begin{aligned} m_0 &= 1 - \xi^2 (3 + 2\xi), & m_1 &= \xi^2 (3 + 2\xi), \\ n_0 &= \xi (1 + \xi)^2, & n_1 &= \xi^2 (1 + \xi). \end{aligned}$$

Two-point Gauss quadrature gives the expected result, but results in a computationally inefficient formula, requiring two Jacobian matrix multiplications. Consider now an integration by means of the three-points trapezoid rule that is required to exactly integrate the 2[nd] order polynomial only. Points and weights are:

point, ξ_j	-1	ζ	0
weight, w_j	$\dfrac{(2+3\zeta)}{6(1+\zeta)}$	$\dfrac{-1}{6\zeta(1+\zeta)}$	$\dfrac{(1+3\zeta)}{6\zeta}$

where ζ is the mid-point position, which becomes a parameter. The integration yields:

$$x_k = x_{k-1} + h \left(\frac{(2+3\zeta)}{6(1+\zeta)} \left. f \right|_{\xi=-1} + \frac{-1}{6\zeta(1+\zeta)} \left. f \right|_{\xi=\zeta} + \frac{(1+3\zeta)}{6\zeta} \left. f \right|_{\xi=0} \right),$$

and the solution is:

$$\begin{aligned} \left(I - h \left(\frac{-1}{6\zeta(1+\zeta)} \left. \frac{\partial f}{\partial x} \right|_{\xi=\zeta} \left(\left(1 - \zeta^2 (3 + 2\zeta)\right) I + h \left(\zeta (1+\zeta)^2\right) \left. \frac{\partial f}{\partial x} \right|_{\xi=0} \right) \right.\right. \\ \left.\left. + \frac{(1+3\zeta)}{6\zeta} \left. \frac{\partial f}{\partial x} \right|_{\xi=0} \right) \right) \Delta x_k = -(x_k - x_{k-1}) \\ + h \left(\frac{(2+3\zeta)}{6(1+\zeta)} \left. f \right|_{\xi=-1} + \frac{-1}{6\zeta(1+\zeta)} \left. f \right|_{\xi=\zeta} + \frac{(1+3\zeta)}{6\zeta} \left. f \right|_{\xi=0} \right). \end{aligned}$$

When applied to the linear differential problem $\dot{x} = \lambda x$, this formula gives:

$$x_k = \frac{\left(1 + \frac{(1-\zeta)}{3} \lambda h - \frac{\zeta}{6} (\lambda h)^2\right)}{\left(1 - \frac{(2+\zeta)}{3} \lambda h + \frac{(1+\zeta)}{6} (\lambda h)^2\right)} x_{k-1}.$$

The asymptotic radius is $\rho_\infty = -\zeta/(1+\zeta)$; for $\zeta < -1/2$ the formula is no longer unconditionally stable. The asymptotic radius vanishes for $\zeta = 0$, when the mid-point moves to the end of the time step. In this case, the last two weights become singular, but since they weigh the function $f(x,t)$ evaluated at the same point, their sum must be considered, whose limit is $2/3$; the Jacobian product does not vanish, reducing to the square of the Jacobian evaluated at the end of the time step. The spectral radius at $\zeta = 0$ is exactly the same that is obtained with the parabolic interpolation. For $\zeta = -1/2$ a 4[th] order accurate integrator with no algorithmic dissipation is obtained. So as the mid-point moves towards the end of the time step, the asymptotic radius of the method goes from 1 to 0, thus allowing the tuning of the algorithmic dissipation. The position of the mid-point can be written as function of the desired radius:

$$\zeta \;=\; -\rho_\infty / \left(1+\rho_\infty\right). \tag{5}$$

In case of $\zeta = 0$ the integrator becomes L-stable, requiring the evaluation of only one Jacobian, and of its square. It is worth recalling that the three-point integration rule used herein integrates polynomials up to the third degree exactly for $\zeta = -1/2$, otherwise only 2^{nd} degree polynomials are integrated exactly. This explains the change in order of accuracy when the mid-point departs from the middle of the time step.

4. SECOND-ORDER FORMULA

A useful 2^{nd} order formula, for a two-step algorithm ($r = 2$), results by assuming a parabolic interpolation for x, using the three-point trapezoid rule with points and weights:

point, ξ_j	-2	-1	0
weight, w_j	$1/2+\delta$	$1-2\delta$	$1/2+\delta$

The integration of $\dot{x}$ leads to:

$$x_k \;=\; x_{k-2} + h\left(\left(\frac{1}{2}+\delta\right) f|_{\xi=0} + (1-2\delta)\, f|_{\xi=-1} + \left(\frac{1}{2}+\delta\right) f|_{\xi=-2}\right), \tag{6}$$

Since the x is assumed parabolic, its derivative is expected to be linear, so the integration of a linear function with three points allows an indetermination δ in the weights ($\delta = -1/6$ yields the usual trapezoid rule). By adding a *Crank-Nicholson* integration formula from $k-2$ to $k-1$, which ought to be identically null because of the assumptions on x, namely:

$$x_{k-1} - x_{k-2} - \frac{h}{2}\left(f|_{\xi=-1} + f|_{\xi=-2}\right) \;=\; 0,$$

and by weighing it by a factor $1-\beta$, the integration formula becomes:

$$\begin{aligned} x_k \;=\;& (1-\beta)\, x_{k-1} + \beta x_{k-2} \\ &+ h\left(\left(\frac{1}{2}+\delta\right) f|_{\xi=0} + \left(\frac{1}{2}(1+\beta) - 2\delta\right) f|_{\xi=-1} + \left(\frac{\beta}{2}+\delta\right) f|_{\xi=-2}\right). \end{aligned} \tag{7}$$

The solution is:

$$\begin{aligned} \left(I - \left(\frac{1}{2}+\delta\right) \left.\frac{\partial f}{\partial x}\right|_{\xi=0}\right) \Delta x_k \;=\;& -\left(x_k - (1-\beta)\, x_{k-1} - \beta x_{k-2}\right) \\ &+ h\left(\left(\frac{1}{2}+\delta\right) f|_{\xi=0} + \left(\frac{1}{2}(1+\beta) - 2\delta\right) f|_{\xi=-1} + \left(\frac{\beta}{2}+\delta\right) f|_{\xi=-2}\right). \end{aligned}$$

The resulting formula is 2^{nd} order accurate, but does not require any Jacobian multiplication. The linear stability analysis yields

$$\begin{aligned} \left(1 - \lambda h\left(\frac{1}{2}+\delta\right)\right) x_k \;=\;& \left((1-\beta) + \lambda h\left(\frac{1}{2}(1+\beta) - 2\delta\right)\right) x_{k-1} \\ &+ \left(\beta + \lambda h\left(\frac{\beta}{2}+\delta\right)\right) x_{k-2}. \end{aligned}$$

The solution of the characteristic polynomial yields the spectral radius of the integrator. An appropriate choice of the weighing parameters δ and β allows to control the algorithmic dissipation while preserving the intrinsic 2[nd] order accuracy of the method. The parameters can be expressed as functions of the desired asymptotic spectral radius:

$$\beta = \frac{4\rho_\infty^2 - (1-\rho_\infty)^2}{4-(1-\rho_\infty)^2}, \qquad \delta = \frac{1}{2}\frac{(1-\rho_\infty)^2}{4-(1-\rho_\infty)^2}.$$

Consider two special cases: $\rho_\infty = 1$ yields $\beta = 1$ and $\delta = 0$ and the method results in:

$$x_k = x_{k-2} + h\left(\frac{1}{2}f(x_k, t_k) + f(x_{k-1}, t_{k-1}) + \frac{1}{2}f(x_{k-2}, t_{k-2})\right),$$

which can be viewed as a sort of two-step *Crank-Nicholson* rule, or as a double-size step *Crank-Nicholson* with a better weighed average of the function f, that results in no algorithmic dissipation. On the other hand, for $\rho_\infty = 0$, the coefficients are $\beta = -1/3$ and $\delta = 1/6$ and the method results in the L-stable two-step Backward Difference Formula:

$$x_k = \frac{4}{3}x_{k-1} - \frac{1}{3}x_{k-2} + \frac{2}{3}hf(x_k, t_k).$$

A practical optimal value of algorithmic dissipation has been found for $\rho_\infty \cong 0.6$, which dissipates enough for most practical cases while maintaining good accuracy. Such value minimizes the 3[rd] order residual in the linear case.

5. IMPLICIT AND DIFFERENTIAL-ALGEBRAIC IMPLEMENTATION

The integration of DAEs has gained a considerable interest in recent times because it allows to exploit High Performance Computing to solve problems that can be stated in a more simple, flexible and redundant form. While the implicit formulation can be easily cast into the explicit form used for ODEs whenever the problem is locally invertible, a DAE problem by definition is structurally singular. It can be solved by adding internal states at the collocation points. The implicit differential problem,

$$F(x, \dot{x}, t) = 0, \qquad x(t_0) = x_0 \tag{8}$$

results in a DAE when the derivative of F in Equation (8) with respect to $\dot{x}$ is singular. Consider the linearization of the discrete solution, Equation (3),

$$\Delta x_k = h\sum_j w_j \Delta\dot{x}_j \tag{9}$$

which shows that the $\Delta\dot{x}_i$ are needed to solve the problem. The basic idea consists in solving Equation (8) in every collocation point i required by the integration method:

$$\left.\frac{\partial F}{\partial x}\right|_{\xi_i} \Delta x(t_i) + \left.\frac{\partial F}{\partial \dot{x}}\right|_{\xi_i} \Delta\dot{x}(t_i) = -\left.F\right|_{\xi_i}$$

The $\Delta x(t_i)$ result from the interpolation formula $\Delta x(t_i) = m_0(\xi_i)\Delta x_k + hn_0(\xi_i)\Delta\dot{x}_k$ and Δx_k is substituted from the linearized discrete integration formula, (9), yielding:

$$\Delta x(t_i) = m_0(\xi_i)\, h\sum_j w_j\Delta\dot{x}_j + hn_0(\xi_i)\Delta\dot{x}_k$$

Now the system has been split in as many equations as the collocation points are; note that, if point t_k is not one of the collocation points, meaning that there is no need to evaluate the integration formula at the end of the time step, and the shape function n_0 of the interpolation is not null, the equation at the end of the time step must be added, since $\Delta\dot{x}_k$ is required to interpolate Δx. The system results in:

$$\left.\frac{\partial F}{\partial x}\right|_{\xi_i}\left(m_0\left(\xi_i\right)h\sum_j w_j\Delta\dot{x}\left(t_j\right)+hn_0\left(\xi_i\right)\Delta\dot{x}_k\right)+\left.\frac{\partial F}{\partial\dot{x}}\right|_{\xi_i}\Delta\dot{x}\left(t_i\right) \;=\; -\left.F\right|_{\xi_i} \qquad (10)$$

where, if n_0 is not identically null and t_k is not part of the t_i's, it is added with null weight, $w_k = 0$. Equation (10) can be easily specialized to the previously introduced algorithms. The formulation for the solution of DAEs clearly shows that the proposed method closely resembles the more general Implicit *Runge-Kutta* (IRK) method for the integration of differential equations [1]. The scheme can be summarized in the so called *Butcher* diagram

$$\begin{array}{c|c} c & \mathcal{A} \\ \hline & b^T \end{array} = \begin{array}{c|ccc} c_1 & a_{11} & \cdots & a_{1M} \\ \vdots & \vdots & \ddots & \vdots \\ c_M & a_{M1} & \cdots & a_{MM} \\ \hline & b_1 & \cdots & b_M \end{array}.$$

By considering a single-step scheme, the c_i become $1-\xi_i$, the b_j become the weights w_j; there is no immediate correspondence for the a_{ij}, because the internal values of x are interpolated from the boundary values of x and of its derivative rather than from the internal values of $\dot{x}$. Only for specific cases a direct equivalence may be found: being $j = 1$ the beginning of the time step and $j = M$ the end of the time step, the general expression is $a_{i1} = m_0(\xi_i)\,w_1 + n_1(\xi_i)$, $a_{iM} = m_0(\xi_i)\,w_M + n_0(\xi_i)$, the other cases being $a_{ij} = m_0(\xi_i)\,w_j$. The property $\sum m_i = 1$ has been exploited to collect x_{k-1}. In detail, the 4^{th} order algorithm with no dissipation is the well-known *Lobatto IIIA* method of order 4, while the 3^{rd} order L-stable algorithm is the *Radau IIA* method of order 3. Figure 1 shows the spectral radii of some of the schemes of the family, ranging from order 1 to 4, while Figure 2 shows the phase and the damping errors.

6. NUMERICAL EXAMPLES

The properties of the higher-order algorithms have been investigated by considering simple mechanical systems. The *Lobatto IIIA* and the *Radau IIA* do not require any investigation since their properties are very well known. The generalized algorithm, on turn, showed very good performances in integrating *stiff* and DAE problems because only limited algorithmic dissipation is required to stabilize their solution, so the errors in phase and damp introduced by L-stability can be reduced.
It is interesting to consider the performances of the 2^{nd} order two-step algorithm, which has been applied to the multibody analysis of sophisticated multidisciplinary systems, e.g. helicopter rotors including aerodynamics, hydraulic controls and active controls by means of "smart" materials. The results of the integration of a flexible pendulum are shown in Figure 3. It is shown how the algorithms perform when the ratio between the time step and the characteristic time of the rod, h/T, varies between 0.1 to 10. In the first case the

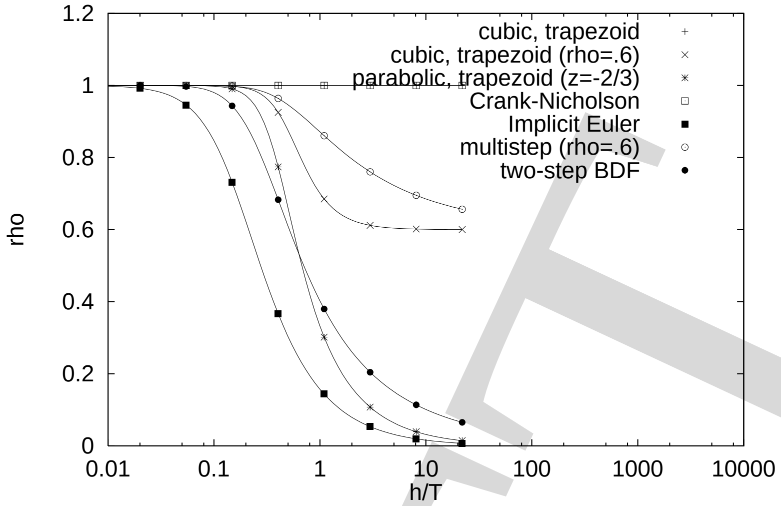

Figure 1: Spectral radii.

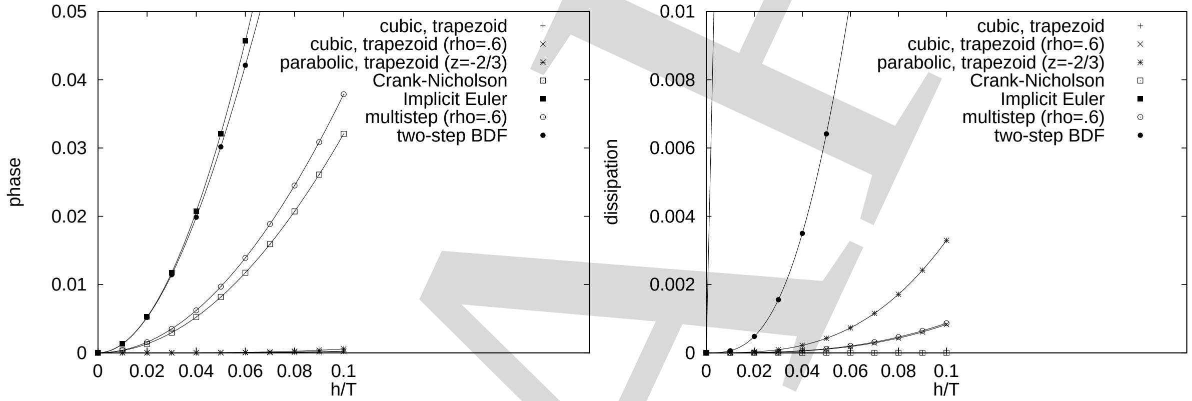

Figure 2: Phase (left) and damping (right) errors.

problem can be integrated, but the (undesired) high-frequency oscillations generated by starting with untrimmed initial conditions are damped only when algorithmic dissipation is used. In the last case, the high-frequency oscillation cannot be integrated correctly because it is completely aliased. However, spurious oscillations are introduced and only algorithmic dissipation can drive the system towards the trimmed solution.

CONCLUSIONS

The family of integration methods that has been presented allows the implicit, unconditionally stable integration of a wide variety of ODEs; it has been generalized to implicit ODEs and DAEs as well. A higher-order A/L-stable single-step algorithm is the most performing, at the cost of one Jacobian multiplication (which turns in doubling the size of the problem in case of DAEs). It allows algorithmic dissipation to be finely tuned, ranging from null to total dissipation. A very promising 2^{nd} order A/L-stable two-step algorithm represents a viable choice for multibody dynamics because even in the DAE

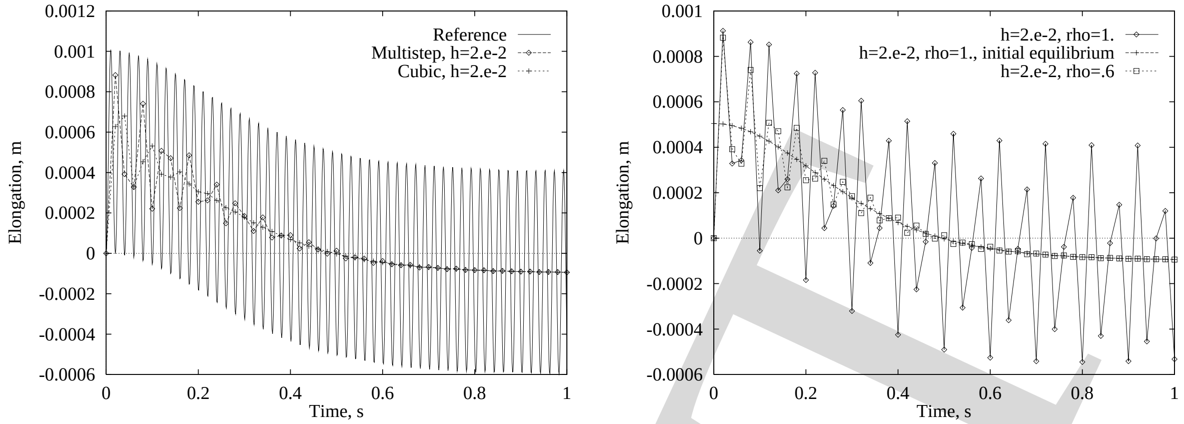

Figure 3: Effects of algorithmic dissipation on a flexible pendulum: attenuation of high-frequency oscillations at $h/T = 0.1$ (left) and $h/T = 10$ (right).

implementation no Jacobian multiplication is required, and only the collocated evaluation at the end of the time step is required. A good trade-off between accuracy and algorithmic dissipation, which is strongly related to the stability of the solution for the *Lagrange* multipliers in DAE integration, has been found by choosing an asymptotic spectral radius of about 0.6. Variable time step implementations have been tested, although not presented in this paper because the details go beyond the scope of this work.

The authors wish to acknowledge the contribution of Gabriele Gilardi and Stefano Marazzani to the development of the 2^{nd} order integration formulæ.

References

[1] E. Hairer and G. Wanner, *Solving Ordinary Differential Equations*, vol. II. Berlin Heidelberg, Germany: Springer-Verlag, 1996. 2^{nd} rev. ed.

[2] O. A. Bauchau, "Computational schemes for flexible, non-linear multi-body systems," *Multibody Systems Dynamics*, vol. 2, no. 2, pp. 169–225, 1998.

[3] K. E. Brenan, S. L. V. Campbell, and L. R. Petzold, *Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations*. New York: North-Holland, 1989.

[4] G. L. Ghiringhelli, P. Masarati, P. Mantegazza, and M. W. Nixon, "Multi-body analysis of a tiltrotor configuration," *Nonlinear Dynamics*, vol. 19, pp. 333–357, August 1999.