Journal of Computational and Applied Mathematics 262 (2014) 217–222

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

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

Reducing round-off errors in symmetric multistep methods

Paola Console, Ernst Hairer *

Section de Mathématiques, Université de Genève, 2-4 rue du Lièvre, CH-1211 Genève 4, Switzerland

ARTICLE INFO

Article history:
Received 15 December 2012
Received in revised form 24 May 2013

MSC:
65L06
65G50
65P10

Keywords:
Symmetric linear multistep methods
Constrained Hamiltonian systems
Propagation of round-off error

ABSTRACT

Certain symmetric linear multistep methods have an excellent long-time behavior when applied to second order Hamiltonian systems with or without constraints. For high accuracy computations round-off can be the dominating source of errors. This article shows how symmetric multistep methods should be implemented such that round-off errors are minimized and propagate like a random walk.

© 2013 Elsevier B.V. All rights reserved.

1. Introduction

This article considers the numerical solution of constrained Hamiltonian systems

$$\begin{aligned} M\ddot{q} &= -\nabla U(q) - G(q)^{\mathsf{T}}\lambda \\ 0 &= g(q), \end{aligned} \tag{1}$$

where $q \in \mathbb{R}^d$, M is a positive definite constant matrix, $U(q)$ is a smooth real potential, $g(q) \in \mathbb{R}^m$ (with $m < d$) collects holonomic constraints, and $G(q) = g'(q)$ is the matrix of partial derivatives. Assuming that $G(q)M^{-1}G(q)^{\mathsf{T}}$ is invertible, the system (1) is a differential-algebraic equation of index 3. With the momentum $p = M\dot{q}$, the problem can be interpreted as a differential equation on the manifold

$$\mathcal{M} = \{(q, p);\ g(q) = 0,\ G(q)M^{-1}p = 0\}. \tag{2}$$

Its flow is a symplectic transformation on $\mathcal{M}$, and it preserves the Hamiltonian (total energy)

$$H(q, p) = \frac{1}{2}\, p^{\mathsf{T}}M^{-1}p + U(q). \tag{3}$$

For a qualitative correct long-time integration of such systems the use of a geometric integrator is essential. An excellent choice is the Rattle algorithm [1], which is a symplectic, symmetric one-step method. However, due to its low order of 2, it is not efficient for high accuracy computations. Symplectic partitioned Runge–Kutta methods (such as the Lobatto IIIA–IIIB pair) have arbitrarily high order, but they are implicit in the force evaluations. This article considers the use of explicit,

* Corresponding author. Tel.: +41 22 379 11 71; fax: +41 22 379 11 76.
E-mail addresses: Paola.Console@unige.ch (P. Console), Ernst.Hairer@unige.ch (E. Hairer).

0377-0427/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.cam.2013.07.025

symmetric multistep methods. With the notation $f(q) = -\nabla U(q)$ they are given by

$$\begin{aligned} &\sum_{j=0}^{k} \alpha_j\, q_{n+j} = h^2 \sum_{j=1}^{k-1} \beta_j\, M^{-1}\Big(f(q_{n+j}) - G(q_{n+j})^{\mathsf{T}} \lambda_{n+j}\Big) \\ &0 = g(q_{n+k}). \end{aligned} \tag{4}$$

For given $q_n, \ldots, q_{n+k-1}$ and $\lambda_{n+1}, \ldots, \lambda_{n+k-2}$, the second relation implicitly defines λ_{n+k-1}, and the first relation gives an explicit expression for q_{n+k}. An approximation of the momentum $p = M\dot{q}$ is obtained *a posteriori* by symmetric finite differences supplemented with a projection onto $\mathcal{M}$:

$$p_n = M\, \frac{1}{h} \sum_{j=-l}^{l} \delta_j q_{n+j} + h\, G(q_n)^{\mathsf{T}} \mu_n, \qquad G(q_n) M^{-1} p_n = 0. \tag{5}$$

The second relation represents a linear system for μ_n, and the first relation is an explicit formula for p_n. By definition, this method yields a numerical solution on the manifold $\mathcal{M}$. It is proved in [2] (in the absence of constraints, see [3]) that under suitable assumptions on the coefficients α_j and β_j, the method can have high order, and the numerical approximations nearly conserve the Hamiltonian over very long time intervals. Its computational cost is essentially the same as that for the Rattle algorithm, which makes it an excellent choice for high accuracy computations.

For computations close to machine accuracy, round-off errors can become more important than discretization errors. This motivates the present study of the propagation of round-off errors. This article gives hints on how the method should be implemented to reduce round-off errors and to obtain approximations for which the error in the Hamiltonian behaves like a random walk. Numerical experiments are presented in a final section.

2. Reducing round-off errors

For a straightforward implementation of method (4)–(5), the round-off error typically increases linearly with time. This can be observed for step sizes, for which the discretization error is close to machine precision. It is known [4] that symplectic implicit Runge–Kutta methods can be implemented such that the round-off error is improved quantitatively (using compensated summation) and qualitatively. This means that it behaves like a random walk and grows like the square root of time. This section shows how the same behavior can be achieved for symmetric multistep methods.

2.1. Separation into difference equations for position and momentum

For consistent multistep methods (4) the characteristic polynomial $\rho(\zeta)$ of the coefficients α_j has a double zero $\zeta = 1$. In the limit $h \to 0$, the solution of the difference equation (4) is unbounded, a fact which provokes an undesired accumulation of round-off errors. There are two possibilities to avoid this weak instability. Either one works with sums of f_j values (*summed form* of [5, Section 6.4-1]) or with differences of q_j values (*stabilized algorithm* of [6, Section III.10]). We use the second approach, because it is closer to the standard use of the Rattle algorithm.

For the difference of two consecutive q_j values we introduce momentum approximations on a staggered grid. Denoting by $\hat{\alpha}_j$ the coefficients of the polynomial $\rho(\zeta)/(\zeta - 1)$, i.e., $\alpha_k = \hat{\alpha}_{k-1}$ and $\alpha_j = \hat{\alpha}_{j-1} - \hat{\alpha}_j$ for $j = 1, \ldots, k-1$, the method (4) is mathematically equivalent to

$$\begin{aligned} &\sum_{j=0}^{k-1} \hat{\alpha}_j\, p_{n+j+1/2} = h \sum_{j=1}^{k-1} \beta_j \Big(f(q_{n+j}) - G(q_{n+j})^{\mathsf{T}} \lambda_{n+j}\Big) \\ &q_{n+k} = q_{n+k-1} + h\, M^{-1} p_{n+k-1/2} \\ &0 = g(q_{n+k}). \end{aligned} \tag{6}$$

Concerning the propagation of round-off errors there is a big difference from (4), because in the limit $h \to 0$ the two difference equations (for position and momentum) have bounded solutions. The approximation of the momenta can be expressed in terms of $p_{n+j+1/2}$ as

$$p_n = \sum_{j=-l}^{l-1} \hat{\delta}_j\, p_{n+j+1/2} + h\, G(q_n)^{\mathsf{T}} \mu_n, \qquad G(q_n) M^{-1} p_n = 0, \tag{7}$$

where the coefficients $\hat{\delta}_j$ are given by $\hat{\delta}_{l-1} = \delta_l$ and $\delta_j = \hat{\delta}_{j-1} - \hat{\delta}_j$ for $j = -l+1, \ldots, l-1$. Compared to (5) this formula for p_n is less affected by round-off errors, because the difference operator approximates a function and not a derivative. This reformulation is less important than the previous one, because p_n is not used in the step by step application of the method.

Download English Version:

https://daneshyari.com/en/article/4638971

Download Persian Version:

https://daneshyari.com/article/4638971

Daneshyari.com