Journal of Computational and Applied Mathematics 236 (2011) 375–383

Contents lists available at SciVerse ScienceDirect

Journal of Computational and Applied Mathematics

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

A note on the efficient implementation of Hamiltonian BVMs[☆]

Luigi Brugnano[a,*], Felice Iavernaro[b], Donato Trigiante[c]

[a] *Dipartimento di Matematica, Università di Firenze, Viale Morgagni 67/A, 50134 Firenze, Italy*
[b] *Dipartimento di Matematica, Università di Bari, Via Orabona 4, 70125 Bari, Italy*
[c] *Dipartimento di Energetica, Università di Firenze, Via Lombroso 6/17, 50134 Firenze, Italy*

ARTICLE INFO

Keywords:
Runge–Kutta methods
Hamiltonian problems
Hamiltonian Boundary Value Methods
Energy preserving methods
Blended methods
Energy drift

ABSTRACT

We discuss the efficient implementation of Hamiltonian BVMs (HBVMs), a recently introduced class of energy preserving methods for canonical Hamiltonian systems (see Brugnano et al. [8] and references therein), also sketching their *blended* formulation. We also discuss the case of separable problems, for which the structure of the problem can be exploited to gain efficiency.

1. Introduction

The conservation of energy allows to avoid the numerical drift sometimes observed when using standard numerical methods for solving canonical Hamiltonian problems, i.e., problems in the form

$$y' = J\nabla H(y), \quad J = \begin{pmatrix} 0 & I_m \\ -I_m & 0 \end{pmatrix}, \qquad y(t_0) = y_0 \in \mathbb{R}^{2m}, \tag{1}$$

where $H(y)$ is a smooth scalar function and, in general, I_r will hereafter denote the identity matrix of dimension r (when the lower index will be omitted, the size of the matrix can be deduced from the context). In this respect, *Hamiltonian Boundary Value Methods (HBVMs)* is a recently introduced class of methods able to conserve energy when $H(y)$ is a polynomial of arbitrarily high degree. Clearly, this implies a *practical* conservation of energy for any suitably regular Hamiltonian function, which will be assumed hereafter.

Energy preserving methods have been investigated by many authors even though the process has been initially so slow that the first unsuccessful attempts to derive energy-preserving Runge–Kutta methods culminated in the wrong general feeling that such methods could not even exist [1]. The first successful attempts to solve the problem have been derived outside the class of Runge–Kutta methods, resulting in *discrete gradient methods* (see [2] and references therein) which are second order accurate. The first instance of energy-conserving Runge–Kutta method is given by the *s-stage trapezoidal methods* [3]. Such methods have been subsequently rederived in [4], through the discretization of the *averaged vector field method* defined in [5]. Additional examples of fourth and sixth-order conservative Runge–Kutta methods (for polynomial Hamiltonians of suitable degree) were presented in [6,7]. All such energy-conserving Runge–Kutta methods have been derived by means of the new concept of *discrete line integral*. The evolution of this idea eventually led to the definition of *Hamiltonian Boundary Value Methods (HBVMs)* mentioned above [8–13], which admit a straightforward Runge–Kutta formulation.

The discrete line integral is nothing but a discrete counterpart of the line integral associated with a conservative vector field which is obtained by simply approximating the line integral by a suitable quadrature formula. In particular, in [9]

[☆] Work developed within the project "Numerical methods and software for differential equations".

[*] Corresponding author.
E-mail addresses: luigi.brugnano@unifi.it (L. Brugnano), felix@dm.uniba.it (F. Iavernaro), trigiant@unifi.it (D. Trigiante).

0377-0427/$ – see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.cam.2011.07.022

HBVMs based on Lobatto quadratures have been analyzed, whereas in [10] HBVMs based on Gauss–Legendre formulae have been considered. When the Hamiltonian function is a polynomial, such an approximation turns out to be exact, provided that a suitable number of nodes is introduced. In [10], it has been shown that all HBVMs having the same order are equivalent (i.e. they provide the very same numerical solution), under the assumptions that $H(y)$ is a polynomial and that the discrete line integral matches the line integral.

When $H(y)$ is not a polynomial, one can still get conservative formulae by letting the number of nodes tend to infinity, thus forcing the discrete line integral to approach the line integral. This is tantamount to skipping the discretization of the line integral during the computation of the formulae. Limit formulae based on the Lagrange basis have been proposed in [14] and generalized to Poisson systems in [15]. Evidently, the polynomial case or the discretization of the integrals appearing in a limit method by means of a quadrature formula, leads back to a HBVM (see [10] for a discussion of this issue and for the definition of limit formulae over the Legendre and other bases).

In the case where the Hamiltonian is separable, i.e., when

$$H(y) \equiv H(q,p) = \frac{1}{2}p^T M p - U(q), \quad q, p \in \mathbb{R}^m, \tag{2}$$

with M a symmetric and positive definite matrix, then (1) reduces to a special second order equation,

$$q'' = M\nabla U(q),$$

and the associated HBVM may be properly formulated in order to take advantage, in terms of efficiency, from the above simplification. As a matter of fact, in this paper we investigate the efficient implementation of HBVMs also in the case of separable problems. In more detail, in Section 2 we briefly derive HBVMs, also providing the most convenient formulation of the discrete problem. Then, in Section 3 we investigate its efficient solution via the *blended* implementation of the methods, which has already proved to be very effective in other settings (see, e.g., [16–23]). The case of separable problems is then discussed in Section 4. A few numerical tests, along with some concluding remarks are then given in Section 5.

2. Hamiltonian BVMs (HBVMs)

The derivation of HBVMs will be done according to the approach followed in [12,13], which further simplifies the already simple idea initially used in [8–11] (see also [6,7]). Let us then consider the restriction of problem (1) to the interval $[t_0, t_0+h]$, with the right-hand side expanded along an orthonormal basis $\{\hat{P}_j\}_{j\ge 0}$:

$$y'(t_0+\tau h) = J\sum_{j\ge 0}\hat{P}_j(\tau)\int_0^1 \hat{P}_j(c)\nabla H(y(t_0+ch))\,\mathrm{d}c, \quad \tau\in[0,1]. \tag{3}$$

In particular, we here consider an orthonormal polynomial basis, provided by the shifted and scaled Legendre polynomials on the interval [0, 1], even though the arguments can be easily extended to more general bases. The basic idea, is now that of looking for an approximate solution belonging to the set of polynomials of degree not larger than s. This is achieved by truncating the series at the right-hand side in (3), thus obtaining the approximate problem

$$\sigma'(t_0+\tau h) = J\sum_{j=0}^{s-1}\hat{P}_j(\tau)\int_0^1 \hat{P}_j(c)\nabla H(\sigma(t_0+ch))\,\mathrm{d}c, \quad \tau\in[0,1],\ \sigma(t_0)=y_0. \tag{4}$$

The approximation to $y(t_0+h)$ is then given by

$$y_1 \equiv \sigma(t_0+h). \tag{5}$$

It can be easily seen that the energy is conserved at the new approximation since, considering that J is skew-symmetric,

$$\begin{aligned} H(y_1)-H(y_0) &= h\int_0^1 \nabla H(\sigma(t_0+\tau h))^T\sigma'(t_0+\tau h)\,\mathrm{d}\tau \\ &= h\sum_{j=0}^{s-1}\left[\int_0^1 \hat{P}_j(\tau)\nabla H(\sigma(t_0+\tau h))\,\mathrm{d}\tau\right]^T J\left[\int_0^1 \hat{P}_j(c)\nabla H(\sigma(t_0+ch))\,\mathrm{d}c\right] = 0. \end{aligned}$$

Integrating both sides of the first equation in (4) yields

$$\sigma(t_0+\tau h) = y_0 + h\sum_{j=0}^{s-1}\int_0^\tau \hat{P}_j(x)\,\mathrm{d}x\int_0^1 \hat{P}_j(c)J\nabla H(\sigma(t_0+ch))\,\mathrm{d}c, \tag{6}$$

which may be exploited to determine the shape of the unknown polynomial σ, provided that a technique to handle the rightmost integrals is taken into account: the obvious choice is the use of quadrature formulae. If we assume that $H(y)$ is a polynomial of degree ν, then the integrals appearing in (4) can be exactly computed by a Gaussian formula with k abscissas $\{c_i\}$, in the event that

$$k \geq \frac{s\nu}{2}, \tag{7}$$

thus obtaining a discrete problem in the form

$$\sigma(t_0 + c_i h) \equiv \sigma_i = y_0 + h \sum_{j=0}^{s-1} \int_0^{c_i} \hat{P}_j(x)\,\mathrm{d}x \sum_{\ell=1}^{k} b_\ell \hat{P}_j(c_\ell) J \nabla H(\sigma_\ell), \quad i = 1, \ldots, k, \tag{8}$$

where the b_i are the quadrature weights of the formula defined over the abscissae c_i. For general, suitably regular (e.g., analytical) Hamiltonian functions, we can still use formula (8) in place of (6), provided that the integrals in (6) are approximated to machine precision[1]: in the following, we will always assume such an accuracy level when a non-polynomial function is considered, and consequently we will make no distinction between the integrals and the corresponding approximations as well as between the two polynomials σ obtained by solving either (8) or (6) (see [13] for more details).

Method (8)–(5) is called HBVM (k, s): it was shown [10] that its order is 2 s, for all $k \geq s$. In particular, for $k = s$ it reduces to the well known s-stage Gauss method.

By introducing the matrices $\Omega = \mathrm{diag}(b_1, \ldots, b_k)$ and

$$\mathcal{I}_{s-1} = \left(\int_0^{c_i} \hat{P}_{j-1}(x)\,\mathrm{d}x \right)_{\substack{i=1,\ldots,k \\ j=1,\ldots,s}} \in \mathbb{R}^{k\times s}, \qquad \mathcal{P}_{r-1} = \left(\hat{P}_{j-1}(c_i) \right)_{\substack{i=1,\ldots,k \\ j=1,\ldots,r}} \in \mathbb{R}^{k\times r}, \tag{9}$$

the HBVM (k, s) can be recast as a Runge–Kutta method with Butcher tableau

$$\begin{array}{c|c} c_1 & \\ \vdots & A \equiv \mathcal{I}_{s-1} \mathcal{P}_{s-1}^T \Omega \\ c_k & \\ \hline & b_1 \quad \cdots \quad b_k \end{array} \tag{10}$$

The next result follows from well-known properties of Legendre polynomials (hereafter e_i denotes the ith unit vector in $\mathbb{R}^s$).

Lemma 1. *Let $\mathcal{I}_{s-1}$ and $\mathcal{P}_s$ be defined according to (9). Then,*

$$\mathcal{I}_{s-1} = \mathcal{P}_s \hat{X}_s \equiv \mathcal{P}_s \begin{pmatrix} X_s \\ \xi_s e_s^T \end{pmatrix} \tag{11}$$

where

$$X_s = \begin{pmatrix} \frac{1}{2} & -\xi_1 & & \\ \xi_1 & 0 & \ddots & \\ & \ddots & \ddots & -\xi_{s-1} \\ & & \xi_{s-1} & 0 \end{pmatrix}, \qquad \xi_i = \frac{1}{2\sqrt{4j^2 - 1}}, \quad i \geq 1. \tag{12}$$

Consequently, the matrix in the Butcher tableau (10) can be written as

$$A = \mathcal{P}_s \hat{X}_s \mathcal{P}_{s-1}^T \Omega. \tag{13}$$

Notice that, since $\mathcal{P}_s \hat{X}_s$ has s linearly independent columns, the $k \times k$ coefficient matrix A has rank s: it is then possible to recast the discrete problem in a more convenient form, whose (block) size is s, rather than k. This fact has been discussed in detail in [9,24] as sketched in the sequel. At first, let us partition the abscissae c_i into two sets: one with s abscissae, the other with the remaining $k - s$ ones. For the sake of simplicity, we choose them as the first s ones and the remaining $k - s$ ones, respectively. According to [9,7], the corresponding stages are called *fundamental stages* and *silent stages*, respectively. The key idea is now that the silent stages can be obtained as a linear combination of the fundamental ones. Let us then partition the matrices $\mathcal{I}_{s-1}$, $\mathcal{P}_{s-1}$, Ω as follows:

$$\mathcal{I}_{s-1} = \begin{pmatrix} \mathcal{I}_{s-1,1} \\ \mathcal{I}_{s-1,2} \end{pmatrix}, \qquad \mathcal{P}_{s-1} = \begin{pmatrix} \mathcal{P}_{s-1,1} \\ \mathcal{P}_{s-1,2} \end{pmatrix}, \qquad \Omega = \begin{pmatrix} \Omega_1 & \\ & \Omega_2 \end{pmatrix}, \quad \mathcal{I}_{s-1,1}, \mathcal{P}_{s-1,1}, \Omega_1 \in \mathbb{R}^{s\times s}.$$

[1] As we will see, increasing the order of the quadrature formula, namely the integer k, will not result in an increase of the dimension of the discrete problem generated by the implementation of the method.

Similarly, let us denote by $\mathbf{y}_1$ the (block) vector, of dimension s, containing the fundamental stages, and by $\mathbf{y}_2$ the (block) vector, of dimension $k-s$, with the silent stages. One then obtains the equations:

$$\mathbf{y}_1 = u_1 \otimes y_0 + h\mathcal{I}_{s-1,1}\left[\mathcal{P}^T_{s-1,1}\Omega_1 \otimes J\nabla H(\mathbf{y}_1) + \mathcal{P}^T_{s-1,2}\Omega_2 \otimes J\nabla H(\mathbf{y}_2)\right], \tag{14}$$

$$\mathbf{y}_2 = u_2 \otimes y_0 + h\mathcal{I}_{s-1,2}\left[\mathcal{P}^T_{s-1,1}\Omega_1 \otimes J\nabla H(\mathbf{y}_1) + \mathcal{P}^T_{s-1,2}\Omega_2 \otimes J\nabla H(\mathbf{y}_2)\right], \tag{15}$$

where $u_1 = (1, \dots, 1)^T \in \mathbb{R}^s$, $u_2 = (1, \dots, 1)^T \in \mathbb{R}^{k-s}$, and an obvious meaning of the (block) vector argument of $\nabla H(\cdot)$. By considering that from Lemma 1 one readily deduces that matrix $\mathcal{I}_{s-1,1}$ is nonsingular, from (14)–(15) one obtains that

$$\mathbf{y}_2 = \hat{u} \otimes y_0 + A_1 \otimes I_{2m}\mathbf{y}_1, \quad \hat{u} = u_2 - A_1 u_1 \in \mathbb{R}^{k-s}, \; A_1 = \mathcal{I}_{s-1,2}\mathcal{I}^{-1}_{s-1,1} \in \mathbb{R}^{k-s\times s}. \tag{16}$$

Then, by setting the matrices

$$B_1 = \mathcal{I}_{s-1,1}\mathcal{P}^T_{s-1,1}\Omega_1 \in \mathbb{R}^{s\times s}, \qquad B_2 = \mathcal{I}_{s-1,1}\mathcal{P}^T_{s-1,2}\Omega_2 \in \mathbb{R}^{s\times k-s},$$

substituting (16) into (14) results in a discrete problem involving only the s fundamental stages,

$$\Psi(\mathbf{y}_1) \equiv \mathbf{y}_1 - u_1 \otimes y_0 - h\left[B_1 \otimes J\nabla H(\mathbf{y}_1) + B_2 \otimes J\nabla H\left(\hat{u} \otimes y_0 + A_1 \otimes I_{2m}\mathbf{y}_1\right)\right] = \mathbf{0}. \tag{17}$$

The application of the simplified Newton method for solving (17) then gives

$$[I - hC \otimes G_0]\,\boldsymbol{\delta}^\ell = -\Psi(\mathbf{y}_1^\ell), \qquad \mathbf{y}_1^{\ell+1} = \mathbf{y}_1^\ell + \boldsymbol{\delta}^\ell, \quad \ell = 0, 1, \dots, \tag{18}$$

where

$$C = B_1 + B_2 A_1 \in \mathbb{R}^{s\times s} \quad \text{and} \quad G_0 = \left(J\nabla^2 H(y_0)\right). \tag{19}$$

It can be proved [24] that the spectrum of matrix C is independent of the choice of the fundamental stages and coincides with that of matrix X_s in (12). On the other hand, its condition number *crucially* depends on such a choice which, therefore, *strongly* affects the convergence of the iteration (18).

It is then advisable to derive a more convenient formulation of the discrete problem, still having (block) dimension s, no more requiring to distinguish between fundamental and silent stages. For this purpose, let us define the (block) vectors (see (4) and (8))

$$\mathbf{y} = \begin{pmatrix} \sigma_1 \\ \vdots \\ \sigma_k \end{pmatrix}, \qquad \boldsymbol{\gamma} = \begin{pmatrix} \gamma_0 \\ \vdots \\ \gamma_{s-1} \end{pmatrix}, \qquad \gamma_j = \sum_{\ell=1}^{k} b_\ell \hat{P}_j(c_\ell) J\nabla H(\sigma(t_0 + c_\ell h)), \quad j = 0, \dots, s-1. \tag{20}$$

In view of (4), we see that the vectors γ_j may be interpreted as the coefficients in the expansion of the degree $s-1$ polynomial $\sigma'(t_0 + \tau h)$ along the orthonormal basis $\{\hat{P}_j\}_{j=0,\dots,s-1}$. From (8) one then obtains

$$\mathbf{y} = e \otimes y_0 + h\mathcal{I}_{s-1} \otimes I_{2m}\boldsymbol{\gamma}, \tag{21}$$

with $e = (1, \dots, 1)^T \in \mathbb{R}^k$, and then, by virtue of (20), one can solve the equation in the unknown $\boldsymbol{\gamma}$,

$$F(\boldsymbol{\gamma}) \equiv \boldsymbol{\gamma} - \left(\mathcal{P}^T_{s-1}\Omega \otimes J\right)\nabla H\left(e \otimes y_0 + h\mathcal{I}_{s-1} \otimes I_{2m}\boldsymbol{\gamma}\right) = 0. \tag{22}$$

The application of the simplified Newton iteration for solving (22) yields (see (19))

$$\left[I - h\mathcal{P}^T_{s-1}\Omega\mathcal{I}_{s-1} \otimes G_0\right]\Delta^\ell = -F(\boldsymbol{\gamma}^\ell), \qquad \boldsymbol{\gamma}^{\ell+1} = \boldsymbol{\gamma}^\ell + \Delta^\ell. \tag{23}$$

By virtue of (11), and considering that

$$\mathcal{P}^T_{s-1}\Omega\mathcal{P}_s = (I_s \; 0) \in \mathbb{R}^{s\times s+1}, \tag{24}$$

(23) reduces to

$$[I - hX_s \otimes G_0]\,\Delta^\ell = -F(\boldsymbol{\gamma}^\ell), \qquad \boldsymbol{\gamma}^{\ell+1} = \boldsymbol{\gamma}^\ell + \Delta^\ell, \quad \ell = 0, 1, \dots. \tag{25}$$

As one may see, the iteration (25) now involves the matrix X_s defined in (12), only depending on the parameter s, in place of C (see (18)–(19)).

3. Blended implementation

From the arguments in the previous section, one then concludes that the discrete problem, to be solved at each integration step when approximating the Hamiltonian problem (1), is given by (22), thus requiring the solution of (25). We are going to solve such an equation by means of a *blended* implementation of the method, according to [16–18,23]. Indeed, such an implementation of block implicit methods has proved to be very effective, leading to the development of the codes `BiM` [18] and `BiMD` [22] for stiff ODE-IVPs and linearly implicit DAEs (the codes are available at the url [25], see also [26]). Let us, for the sake of simplicity, discard the iteration index. Consequently, we have to solve the linear system

$$(I - hX_s \otimes G_0)\,\Delta = -F(\boldsymbol{\gamma}) \equiv \boldsymbol{\eta}. \tag{26}$$

Considering that matrix X_s (see (12)) is nonsingular, such an equation can be equivalently written as

$$\rho\left(X_s^{-1} \otimes I_{2m} - hI_s \otimes G_0\right)\Delta = \rho X_s^{-1} \otimes I_{2m}\,\boldsymbol{\eta} \equiv \boldsymbol{\eta}_1, \tag{27}$$

where ρ is a positive constant. By introducing the (matrix) weight function

$$\theta = I_s \otimes \Sigma_0^{-1}, \quad \Sigma_0 = (I_{2m} - \rho h G_0)^{-1}, \tag{28}$$

we then obtain the following problem, which still has the same solution as (26):

$$T(\Delta) \equiv \theta\left[(I - hX_s \otimes G_0)\,\Delta - \boldsymbol{\eta}\right] + (I - \theta)\left[\rho\left(X_s^{-1} \otimes I_{2m} - hI_s \otimes G_0\right)\Delta - \boldsymbol{\eta}_1\right] = 0. \tag{29}$$

One easily realizes that it is obtained as the *blending*, with weights θ and $(I - \theta)$, of the two equivalent problems (26) and (27), respectively. Problem (29) defines the *blended method* associated with the original one, which we call *blended HBVM*, in the present case. The free parameter ρ is chosen in order to optimize the convergence properties of the corresponding *blended iteration*,

$$\Delta_{n+1} = \Delta_n - \theta T(\Delta_n), \quad n \ge 0, \tag{30}$$

with an obvious meaning of the lower index. Such iteration only requires (see (28)) the factorization of the matrix Σ_0 having the same size as that of the continuous problem. According to the linear analysis of convergence in [20], the free parameter ρ is chosen as

$$\rho = \rho_s \equiv \min\{|\lambda| : \lambda \in \sigma(X_s)\}, \tag{31}$$

which provides optimal convergence properties (in particular, an *L*-convergent iteration [20]). This simple choice is due to the fact that the eigenvalues of matrix X_s coincide with those of the Butcher matrix of the *s*-stage Gauss–Legendre method [17]. A few values of (31) are listed in the table below, for the sake of completeness.

s	2	3	4	5
ρ_s	0.2887	0.1967	0.1475	0.1173

Remark 1. A nonlinear version of (30) can be readily derived, by taking $\Delta_n = 0$ and updating the vectors $\boldsymbol{\eta}$ and $\boldsymbol{\eta}_1$ in (29) at each iteration.

4. The case of separable problems

Let us now apply the method to the separable problem (2). By setting the (block) vectors

$$\mathbf{q} = \left(q_1^T, \ldots, q_k^T\right)^T, \qquad \mathbf{p} = \left(p_1^T, \ldots, p_k^T\right)^T,$$

one then obtains (see (13)),

$$\mathbf{q} = e \otimes q_0 + hA \otimes M\mathbf{p}, \quad \mathbf{p} = e \otimes p_0 + hA \otimes I_m \nabla U(\mathbf{q}),$$

i.e., since $Ae = c \equiv (c_1, \ldots, c_k)^T$,

$$\mathbf{q} = e \otimes q_0 + hc \otimes Mp_0 + h^2 A^2 \otimes M\nabla U(\mathbf{q}). \tag{32}$$

Moreover, taking into account (10)–(13) and (24), one obtains

$$A^2 = \mathcal{I}_{s-1} X_s \mathcal{P}_{s-1}^T \Omega. \tag{33}$$

The new approximations to $q(t_0 + h)$ and $p(t_0 + h)$ are then given by

$$q_0 + hMp_0 + h^2 \mathrm{e}^T \Omega A \otimes M\nabla U(\mathbf{q}), \quad p_0 + h\mathrm{e}^T \Omega \otimes I_m\, \nabla U(\mathbf{q}),$$

respectively. By using similar arguments as those given in Section 2 (see (21)), we set

$$\mathbf{q} = e \otimes q_0 + hc \otimes Mp_0 + h^2 \mathcal{I}_{s-1} X_s \otimes I_m \boldsymbol{\gamma},$$

Table 1
Number of fixed-point or blended iterations (n.c. = no convergence) required for solving problem (37), along with the maximum relative error (errH) in the numerical Hamiltonian, for the HBVM(8, 2) and GAUSS2 methods, by using step size $h_i = 2^{-i}10^{-1},\ i = 0, \dots, 6$.

i	HBVM(8, 2)					GAUSS2				
	First order		Second order		errH	First order		Second order		errH
	Fixed-pt	Blend	Fixed-pt	Blend		Fixed-pt	Blend	Fixed-pt	Blend	
0	n.c.	1 388	n.c.	1 344	1.7e−18	n.c.	1 390	n.c.	1 320	7.8e−06
1	n.c.	3 330	n.c.	3 909	1.7e−18	n.c.	3 339	n.c.	3 903	6.4e−06
2	n.c.	7 200	n.c.	10 397	2.6e−18	n.c.	7 203	n.c.	8 000	7.1e−05
3	38 353	13 148	22 104	16 038	2.8e−18	38 638	13 165	16 000	16 000	4.1e−06
4	38 458	21 312	23 268	20 846	2.6e−18	38 474	21 311	23 235	20 895	8.6e−08
5	51 267	34 932	31 959	32 000	1.7e−18	51 269	34 942	31 955	32 014	4.2e−09
6	75 800	57 600	51 202	51 200	1.7e−18	75 804	57 600	51 200	51 208	1.9e−16

where the (block) entries of $\boldsymbol{\gamma}$ are defined by (compare with (20))

$$\gamma_j = \sum_{\ell=1}^{k} b_\ell \hat{P}_j(c_\ell) M \nabla U(q_\ell), \quad j = 0, \dots, s-1,$$

i.e., $\boldsymbol{\gamma} = \mathcal{P}_{s-1}^T \varOmega \otimes M \nabla U(\mathbf{q})$. Thus, we obtain the following equation in the unknown $\boldsymbol{\gamma}$ (which is analogous to (22)):

$$F(\boldsymbol{\gamma}) \equiv \boldsymbol{\gamma} - \left(\mathcal{P}_{s-1}^T \varOmega \otimes M\right) \nabla U \left(e \otimes q_0 + hc \otimes Mp_0 + h^2 \mathcal{I}_{s-1} X_s \otimes I_m \boldsymbol{\gamma}\right) = 0. \tag{34}$$

Similarly to what was seen in Section 3, the application of the simplified Newton iteration for solving (34) then gives, by virtue of (11) and (24), and setting $G_0 = M\nabla^2 U(q_0)$,

$$\left[I - h^2 X_s^2 \otimes G_0\right] \Delta^\ell = -F(\boldsymbol{\gamma}^\ell), \qquad \boldsymbol{\gamma}^{\ell+1} = \boldsymbol{\gamma}^\ell + \Delta^\ell, \quad \ell = 0, 1, \dots, \tag{35}$$

which, as in the previous case, has (block) size s, rather than k. The problem is then exactly that seen in (25), via the formal substitutions

$$h \longrightarrow h^2, \qquad X_s \longrightarrow X_s^2. \tag{36}$$

This means that we can repeat similar steps for the *blended* solution of (35), by following the same arguments seen in Section 3. In more detail, (26)–(30) can be repeated, by considering the formal substitutions (36) and, moreover,

$$\rho \longrightarrow \rho^2, \qquad I_{2m} \longrightarrow I_m.$$

Also in this case [19,20], the optimal choice of the parameter ρ turns out to be given by (31). Moreover, a nonlinear version of (35) is readily obtained, according to what is said in Remark 1.

5. Numerical tests

We here consider a few model problems to test the proposed algorithms and methods, in order to confirm the usefulness of the proposed approach. Newton-type iterations, and then, the blended implementation of the methods, proved to be very effective, when speaking about dissipative problems, in case of *stiff* problems (see [27] for a recent survey about *stiffness*). In the case of special second order problems, their counterpart is given by *highly oscillatory problems*, namely Hamiltonian problems whose solution $y(t)$ combines components with dominant short frequencies and components with large frequencies (with respect to the interval of integration).

The first test problem is defined by the non-polynomial Hamiltonian

$$H(q, p) = \frac{1}{2}p^2 + \sin^2(10^2 q), \tag{37}$$

which is solved over the interval [0, 10] starting from $(q_0, p_0) = (0,\ 10^{-1})$. We use the following fourth-order methods:

- the symplectic 2-stage Gauss (GAUSS2 hereafter) method (i.e., HBVM(2, 2), which is not conservative, for the problem at hand);
- the (practically) conservative HBVM(8, 2) method (for the used step-sizes).

For all methods, we use their blended and fixed-point implementations. Moreover, the methods are used both on the first and second order formulation of the problem. Table 1 summarizes the obtained results, in terms of iterations needed for convergence, as well as of error in the numerical Hamiltonian. It is evident that the conservative method is more effective than the symplectic one, with respect to the conservation of the Hamiltonian; moreover, the blended implementations of the methods appears to be more robust than the fixed-point one.

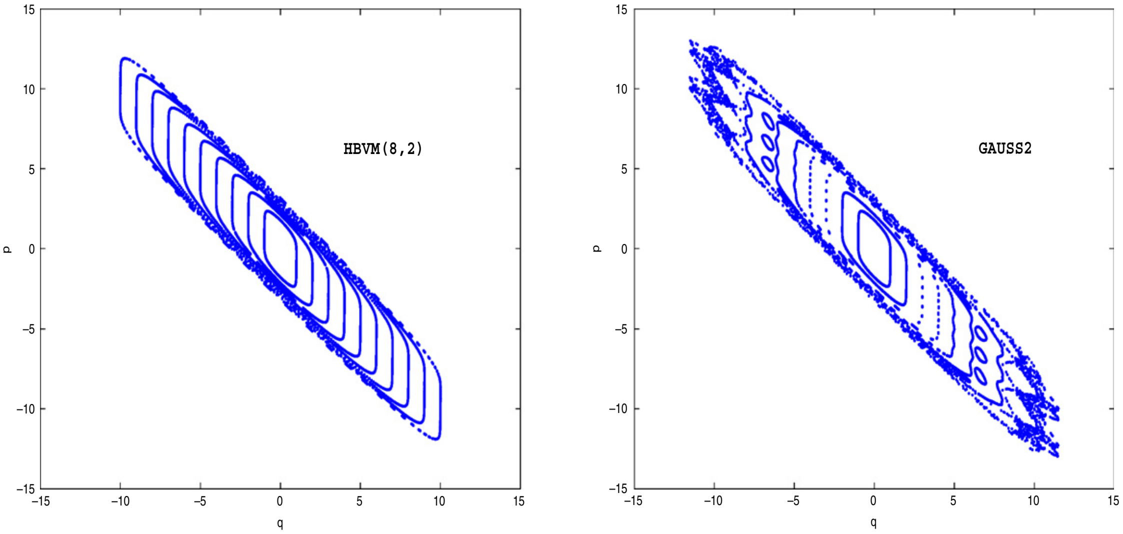

Fig. 1. Level curves for problem (38)–(40) numerically computed by the energy conserving HBVM(8, 2) method (left plot) and by the symplectic GAUSS2 method (right plot), by performing 10^3 steps with step size $h = 10^{-3}$. The conservative method evidently outperform the symplectic one, for the given step size h.

Table 2
Number of fixed-point or blended iterations (n.c. = no convergence) required for each level curve of problem (38)–(40), along with the maximum relative error (errH) in the numerical Hamiltonian, by using the HBVM(8, 2) and GAUSS2 methods.

i	HBVM(8, 2)			GAUSS2		
	Fixed-pt	Blend	errH	Fixed-pt	Blend	errH
1	11 885	9 524	1.8e−15	11 889	9 603	1.0e−04
2	14 723	11 882	2.0e−15	14 611	11 921	9.3e−04
3	17 003	13 808	3.3e−15	17 354	13 532	5.3e−03
4	19 141	15 452	1.7e−15	18 937	15 528	1.7e−02
5	21 552	17 152	4.0e−15	21 135	16 881	3.5e−02
6	24 351	19 064	1.8e−15	24 212	18 907	5.9e−02
7	27 728	21 067	1.7e−15	27 068	19 725	1.9e−01
8	31 624	23 347	1.8e−15	30 028	22 964	7.6e−02
9	38 075	24 823	1.7e−15	40 350	28 491	3.6e−01
10	42 911	29 263	1.2e−14	n.c.	32 997	3.5e−01

The second test problem is defined by the polynomial Hamiltonian

$$H(q,p) = p^2 + \beta^2 q^2 + \alpha^2 (q+p)^{2n}, \tag{38}$$

which has a marginally stable equilibrium at the origin. We consider the following set of parameters

$$\beta = 10, \qquad \alpha = 1, \qquad n = 4, \tag{39}$$

so that the Hamiltonian has degree 8 and the problem in not separable. We consider the same fourth-order methods as in the previous example, noticing that, for the problem at hand, the HBVM(8, 2) method is conservative. Both methods are used, with step size $h = 10^{-3}$, for integrating the problem on the interval [0, 1], with initial points

$$(q_0,\ p_0) = (i,\ -i), \quad i = 1, \ldots, 10, \tag{40}$$

to obtain the corresponding level curves of the Hamiltonian. We notice that, since the problem has one degree of freedom, the numerical solution provided by the HBVM(8, 2) lies on the very same continuous trajectory, or level curve, of the original problem. Consequently the level curves are recovered without error, as illustrated in the left plot of Fig. 1. This is not the case with the GAUSS2 method which evidently introduces huge deformations of the level curves especially when the numerical solution is not close to the equilibrium point. Table 2 summarizes the obtained results, in terms of iterations to obtain convergence (blended or fixed-point). The maximum error in the Hamiltonian is also listed in the table. As one can see, the blended implementation of the method is more efficient than the fixed-point implementation.

Finally, we consider the well-known Fermi–Pasta–Ulam problem (see [28, Section I.5.1]), defined by the Hamiltonian

$$H(p,q) = \frac{1}{2}\sum_{i=1}^{m}\left(p_{2i-1}^2 + p_{2i}^2\right) + \frac{\omega^2}{4}\sum_{i=1}^{m}(q_{2i} - q_{2i-1})^2 + \sum_{i=0}^{m}(q_{2i+1} - q_{2i})^4, \tag{41}$$

Table 3
Number of fixed-point or blended iterations (n.c. = no convergence) required for solving problem (41)–(42), along with the maximum relative error (errH) in the numerical Hamiltonian, for the HBVM(8, 2) and GAUSS2 methods, by using step size $h_i = 2^i 10^{-1}$, $i = 0, \dots, 9$.

i	HBVM(4, 2)					GAUSS2				
	First order		Second order		errH	First order		Second order		errH
	Fixed-pt	Blend	Fixed-pt	Blend		Fixed-pt	Blend	Fixed-pt	Blend	
0	n.c.	1 786	n.c.	6 258	4.4e−16	n.c.	1 798	n.c.	4 675	5.0e−05
1	n.c.	4 176	n.c.	10 647	4.4e−16	n.c.	4 141	n.c.	9 627	4.8e−05
2	40 143	8 300	54 858	17 324	6.7e−16	40 165	8 368	39 121	15 956	1.4e−05
3	27 554	13 517	34 875	25 144	4.4e−16	27 505	13 770	30 778	25 663	6.9e−07
4	34 292	21 242	40 200	38 233	4.4e−16	34 251	21 442	32 581	32 800	1.2e−08
5	47 680	34 982	45 948	46 478	6.7e−16	48 056	35 096	40 812	40 956	4.5e−10
6	75 922	58 423	60 943	63 291	5.6e−16	75 850	58 468	54 115	56 486	8.0e−12
7	126 208	102 400	93 563	107 968	6.7e−16	126 191	102 403	87 130	91 097	2.4e−13
8	204 677	179 662	155 546	155 988	6.7e−16	204 666	179 686	153 453	154 291	5.8e−15
9	352 100	324 764	260 928	275 912	6.7e−16	352 102	324 705	258 067	273 855	7.8e−16

with $q_0 = q_{2m+1} = 0$. We use the following parameters and starting point, with [0, 10] the integration interval:

$$m = 6, \qquad \omega = 100, \qquad p_i = 0, \qquad q_i = (i-1)/10, \quad i = 1, \dots, 6. \tag{42}$$

Here, the Hamiltonian function is a polynomial of degree 4, so that the fourth-order HBVM(4, 2) method is energy conserving. Table 3 summarizes the obtained results by using the HBVM(4, 2) and GAUSS2 methods, both for the first and second order formulation of the problem, as well as by using the blended or fixed-point iterations for solving the discrete problems. The maximum relative error in the numerical Hamiltonian is also listed, thus confirming that the method HBVM(4, 2) is conservative, contrary to GAUSS2.

From the numerical tests, one can then conclude that the proposed blended implementation of HBVMs turns out to be robust and efficient for larger step sizes (as it usually happens for Newton-like iterations). Moreover, the energy-conserving property of such methods turns out to be very remarkable. Finally, the methods can be conveniently recast into a second order formulation, in the case of separable problems.

References

[1] A. Iserles, A. Zanna, Preserving algebraic invariants with Runge–Kutta methods, J. Comput. Appl. Math. 125 (2000) 69–81.

[2] R.I. McLachlan, G.R.W. Quispel, N. Robidoux, Geometric integration using discrete gradient, Philos. Trans. R. Soc. Lond. Ser. A 357 (1999) 1021–1045.

[3] F. Iavernaro, B. Pace, *s*-stage trapezoidal methods for the conservation of Hamiltonian functions of polynomial type, AIP Conf. Proc. 936 (2007) 603–606.

[4] E. Celledoni, R.I. McLachlan, D. McLaren, B. Owren, G.R.W. Quispel, W.M. Wright, Energy preserving Runge–Kutta methods, M2AN Math. Model. Numer. Anal. 43 (2009) 645–649.

[5] G.R.W. Quispel, D.I. McLaren, A new class of energy-preserving numerical integration methods, J. Phys. A: Math. Theor. 41 (2008) 045206. 7 pp.

[6] F. Iavernaro, B. Pace, Conservative block-boundary value methods for the solution of polynomial Hamiltonian systems, AIP Conf. Proc. 1048 (2008) 888–891.

[7] F. Iavernaro, D. Trigiante, High-order symmetric schemes for the energy conservation of polynomial Hamiltonian problems, JNAIAM J. Numer. Anal. Ind. Appl. Math. 4 (1–2) (2009) 87–101.

[8] L. Brugnano, F. Iavernaro, D. Trigiante, The Hamiltonian BVMs (HBVMs) Homepage. arXiv:1002.2757, URL: http://www.math.unifi.it/~brugnano/HBVM/.

[9] L. Brugnano, F. Iavernaro, D. Trigiante, Analysis of Hamiltonian boundary value methods (HBVMs): a class of energy-preserving Runge–Kutta methods for the numerical solution of polynomial Hamiltonian dynamical systems (2009) (submitted for publication). arXiv:0909.5659.

[10] L. Brugnano, F. Iavernaro, D. Trigiante, Hamiltonian boundary value methods (energy preserving discrete line integral methods), JNAIAM J. Numer. Anal. Ind. Appl. Math. 5 (1–2) (2010) 17–37. arXiv:0910.3621.

[11] L. Brugnano, F. Iavernaro, D. Trigiante, The lack of continuity and the role of infinite and infinitesimal in numerical methods for ODEs: the case of symplecticity, Appl. Math. Comput. (in press) (doi:10.1016/j.amc.2011.03.022).

[12] L. Brugnano, F. Iavernaro, D. Trigiante, Numerical solution of ODEs and the Columbus' egg: three simple ideas for three difficult problems, Math. Eng. Sci. Aerosp. 1 (4) (2010) 105–124. arXiv:1008.4789.

[13] L. Brugnano, F. Iavernaro, D. Trigiante, A unifying framework for the derivation and analysis of effective classes of one-step methods for ODEs (2010) (submitted for publication). arXiv:1009.3165.

[14] E. Hairer, Energy-preserving variant of collocation methods, JNAIAM J. Numer. Anal. Ind. Appl. Math. 5 (1–2) (2010) 73–84.

[15] D. Cohen, E. Hairer, Linear energy-preserving integrators for Poisson systems, BIT (in press).

[16] L. Brugnano, Blended block BVMs (B_3VMs): a family of economical implicit methods for ODEs, J. Comput. Appl. Math. 116 (2000) 41–62.

[17] L. Brugnano, C. Magherini, Blended implementation of block implicit methods for ODEs, Appl. Numer. Math. 42 (2002) 29–45.

[18] L. Brugnano, C. Magherini, The `BiM` code for the numerical solution of ODEs, J. Comput. Appl. Math. 164–165 (2004) 145–158.

[19] L. Brugnano, C. Magherini, Blended implicit methods for solving ODE and DAE problems, and their extension for second order problems, J. Comput. Appl. Math. 205 (2007) 777–790.

[20] L. Brugnano, C. Magherini, Recent advances in linear analysis of convergence for splittings for solving ODE problems, Appl. Numer. Math. 59 (2009) 542–557.

[21] L. Brugnano, C. Magherini, Blended general linear methods based on boundary value methods in the GBDF family, JNAIAM J. Numer. Anal. Ind. Appl. Math. 4 (1–2) (2009) 23–40.

[22] L. Brugnano, C. Magherini, F. Mugnai, Blended implicit methods for the numerical solution of DAE problems, J. Comput. Appl. Math. 189 (2006) 34–50.

[23] L. Brugnano, D. Trigiante, Block implicit methods for ODEs, in: D. Trigiante (Ed.), Recent Trends in Numerical Analysis, Nova Science Publ. Inc., New York, 2001, pp. 81–105.

[24] L. Brugnano, F. Iavernaro, D. Trigiante, Isospectral property of Hamiltonian boundary value methods (HBVMs) and their blended implementation, Preprint, 2010. arXiv:1002.1387.

[25] The codes BiM and BiMD Home Page: http://www.math.unifi.it/~brugnano/BiM/.
[26] TestSet for IVP Solvers (rel. 2.4): http://pitagora.dm.uniba.it/~testset/.
[27] L. Brugnano, F. Mazzia, D. Trigiante, Fifty years of stiffness, in: Recent Advances in Computational and Applied Mathematics, Springer, 2011, pp. 1–21. doi:10.1007/978-90-481-9981-5_1.
[28] E. Hairer, C. Lubich, G. Wanner, Geometric Numerical Integration, 2nd ed., Springer, Berlin, 2006.