A new class of symplectic integration schemes based on generating functions

J. Makazaga, A. Murua
Konputazio Zientziak eta A.A. saila, Informatika Fakultatea,
EHU/UPV, Donostia/San Sebastian, Spain

May 7, 2009

Abstract

We present a new family of one-step symplectic integration schemes for Hamiltonian systems of the general form $\dot{y} = J^{-1}\nabla H(y)^T$. Such a class of methods contains as particular cases the methods of Miesbach and Pesch [13], and also the family of symplectic Runge-Kutta methods. As in the case of the methods introduced in [13], the new integration methods are constructed by defining a generating function, which automatically determines a symplectic map. The resulting methods are implicit, and require the evaluation of the gradient of the Hamiltonian function as well as the Hessian times a vector.

1 Introduction

We present a new class of one-step symplectic integration schemes for Hamiltonian systems of the general form

$$\dot{y} = J^{-1}\nabla H(y), \quad J = \begin{pmatrix} 0 & I \\ -I & 0 \end{pmatrix}. \tag{1}$$

(we assume that $H : \mathcal{U} \subset \mathbb{R}^{2d} \to \mathbb{R}$ is a smooth function defined in an open set $\mathcal{U}$ of $\mathbb{R}^{2d}$). It is well known [2], [11], [7] that there exist explicit symplectic methods for Hamiltonian systems (1) with Hamiltonian function of the form

$$H(p, q) = T(p) + V(q). \tag{2}$$

However, all known symplectic schemes that can be applied to any general Hamiltonian system of the form (1) are implicit. Even for systems with Hamiltonian of the form (2), implicit methods will be required if a symplectic variable time-step strategy [9],[15] (see also [7]) is needed (for instance, when integrating numerically the solution of an initial value problem for which the time scale changes significantly along the solution).

The integration schemes we present are implicit methods that are symplectic when applied to a Hamiltonian system of the form (1), and can be interpreted as a generalization of the implicit midpoint rule. Actually, the schemes we introduce can be used to integrate numerically any system of ordinary differential equations of the form

$$\dot{y} = f(y), \quad f : \mathcal{U} \subset \mathbb{R}^D \to \mathbb{R}^D. \tag{3}$$

Integrating with our new schemes an ODE system (3) that is not of the form (1) may be of particular interest if (3) has one or more quadratic invariants. In fact, the family of integrators we introduce in the present work fall within the class of symplectic B-series methods, which can be characterized as B-series methods that conserve all the quadratic invariants of the original system.

Our new family of methods contains as particular cases the symplectic Runge-Kutta methods and the methods of Miesbach and Pesch [13]. As in the case of the methods introduced in [13], the new integration methods are constructed for the Hamiltonian system (1) by defining a generating function, which automatically determines a symplectic map. The resulting methods are implicit, and require the evaluation of the gradient of the Hamiltonian function as well as the Hessian times a vector. It is worth noting that, as it is standard in automatic differentiation, the computational effort needed to evaluate the gradient $\nabla H(y)$ of a scalar function in several variables together with the function $H(y)$ itself, is a small factor (depending on the function H, typically between one and two) times the cost of evaluating $H(y)$ alone. Similarly occurs with the evaluation of $\nabla^2 H(y)v$ (v a given vector) together with $\nabla H(y)$ (see [17]) (observe that $\nabla^2 H(y)v$ actually is the gradient of the scalar function $\nabla H(y)v$ with v a constant vector).

Consider a one-parameter family of functions $S(\cdot, h) : \mathcal{U}_h \subset \mathbb{R}^{2d} \to \mathbb{R}$ that satisfies the Hamilton-Jacobi equation

$$\dot{S}(z,h) = H\left(z + \frac{1}{2}J^{-1}\nabla S(z,h)\right) \tag{4}$$

with initial condition $S(y,0) \equiv 0$. Hereafter, $\dot{S}(z,h)$ denotes the partial derivative of $S(z,h)$ with respect to h, while $\nabla S(z,h)$ denotes the gradient of $S(z,h)$ with respect to z. Then, for any solution $y(t)$ of (1), it holds that

$$y(t+h) = y(t) + J^{-1}\nabla S(\tfrac{1}{2}(y(t) + y(t+h)), h).$$

In other words, $S(z,h)$ is a generating function of the (symplectic) h-flow of (1) (denoted by $S^3(z,h)$ in [7]). In [13], a family of symplectic integrators is introduced by considering explicit approximations of the generating function $S(z,h)$ defined in terms of values of the Hamiltonian function H and its gradient ∇H. A slight generalization of the family of generating functions that they consider is given as

$$S(z,h) = h\sum_{i=1}^{s} b_i H(Y_i), \tag{5}$$

where for $i = 1,\ldots,s$,

$$Y_i = z + hJ^{-1}\sum_{j=1}^{s} \alpha_{ij}\nabla H(Y_j),$$

and the real parameters b_i, α_{ij} are appropriately chosen so that $S(z,h)$ is an approximate solution of the Hamilton-Jacobi equation (4). Thus, one step $y^* = \psi_h(y)$ of the method is implicitly defined as

$$y^* = y + J^{-1}\nabla S\left(\tfrac{1}{2}(y+y^*),h\right). \tag{6}$$

High order symplectic integrator based on generating functions were also considered in [5],[6], and [4], but in such case, derivatives of the Hamiltonian function H of order higher than two are required.

It is interesting to note that any symplectic Runge-Kutta method can also be written in a very similar form [2],[7], the only difference being that the generating function (5) must be replaced by

$$S(z,h) = h\sum_{i=1}^{s} b_i H(Y_i) + \frac{h^2}{2}\sum_{i,j=1}^{s} b_i\alpha_{ij}\nabla H(Y_i)^T J^{-1}\nabla H(Y_j), \tag{7}$$

where the parameters b_i, α_{ij} are such that $b_i\alpha_{ij} + b_j\alpha_{ji} = 0$ for all i,j (with $a_{ij} = \alpha_{ij} + b_i/2$, one gets the more familiar conditions $b_i a_{ij} + b_j a_{ji} - b_i b_j = 0$, i.e. the conditions for a Runge-Kutta method to be symplectic).

The generalization we propose consists on replacing the coefficients $b_i\alpha_{ij}$ in (7) by arbitrary real parameters β_{ij}.

The main purpose of this paper is to introduce a new family of symplectic integration methods based on generating functions intended to solve numerically Hamiltonian systems of the form (1). The definition of the methods is given in Section 2, and we show how to compute efficiently the gradient $\nabla S(z,h)$ of the generating function $S(z,h)$ required in (6). In addition, we generalize the methods to general (non-necessarily Hamiltonian) ODE systems. In section 3 we apply the theory of B-series to derive the order conditions of our new family of methods and we use them in Section 4 to construct a time-symmetric 6th order method of explicit type (i.e., with a explicitly defined generating function $S(z,h)$) with $s=4$. Finally, we summarize the conclusions in Section 5.

2 The new family of methods

2.1 Definition of the methods

Given a positive integer s and real numbers b_i ($1 \leq i \leq s$), α_{ij} and β_{ij} ($1 \leq i,j \leq s$) such that $\beta_{ij} = -\beta_{ji}$ for all i,j, one step $y^* = \psi_h(y)$ of the method is implicitly defined by (6), where the generating function $S(z,h)$ is defined as

$$\begin{aligned} S(z,h) &= h\sum_{i=1}^{s} b_i H(Y_i) + \frac{h^2}{2}\sum_{i,j=1}^{s} \beta_{ij} \nabla H(Y_i)^T J^{-1} \nabla H(Y_j) \qquad (8) \\ &= h\sum_{i=1}^{s} b_i H(Y_i) + h^2 \sum_{i=1}^{s}\sum_{j=1}^{i-1} \beta_{ij} \nabla H(Y_i)^T J^{-1} \nabla H(Y_j), \end{aligned}$$

where for $i = 1,\dots,s$,

$$Y_i = z + hJ^{-1}\sum_{j=1}^{s} \alpha_{ij} \nabla H(Y_j). \qquad (9)$$

Notice that considering a matrix $(\beta_{ij})_{i,j=1}^{s}$ that is not skew-symmetric would not give more generality to the method: Since J^{-1} is skew-symmetric, β_{ij} can be replaced by $(\beta_{ij} - \beta_{ji})/2$ in the definition of $S(z,h)$ in (8).

Clearly, if the matrix $(\alpha_{ij})_{i,j=1}^{s}$ is strictly lower triangular, then $S(z,h)$ is explicitly given in (8)–(9) as a function of h and z. In such case, (we will say that the method is of explicit type,) the method (6) with the generating function $S(z,h)$ given by (8)–(9) is only implicit in y^*.

We will be mainly interested in time-symmetric (or self-adjoint) methods, that is, methods satisfying that $\psi_h^{-1} = \psi_{-h}$, which is easily seen to be equivalent to $S(z,-h) \equiv -S(z,h)$ for one-step methods $y^* = \psi_h(y)$ defined as (6). It is not difficult to check that, if there exists a permutation π of the set of indices $\{1,\dots,s\}$ such that $\pi^{-1} = \pi$, and

$$b_{\pi(i)} = b_i, \quad \alpha_{\pi(i),\pi(j)} = -\alpha_{i,j} \quad \beta_{\pi(i),\pi(j)} = -\beta_{i,j}, \qquad (10)$$

for all $i,j = 1,\dots,s$, then $S(z,-h) \equiv -S(z,h)$, and thus the method is time-symmetric.

2.2 Efficient computation of the gradient of the generating function

In order to implement the integration scheme (6) with the generating function defined by (8)–(9), the gradient of $S(z,h)$ need to be computed. Let us denote

$$k_i = h\nabla H(Y_i),$$

then we have that

$$\frac{\partial S(z,h)}{\partial z} = \sum_{i=1}^{s} b_i k_i^T \frac{\partial Y_i}{\partial z} + \sum_{i,j=1}^{s} \beta_{ij} k_i^T J^{-1} \frac{\partial k_j}{\partial z},$$

where for each $i = 1, \ldots, s$,

$$\frac{\partial Y_i}{\partial z} = I + J^{-1} \sum_{j=1}^{s} \alpha_{ij} \frac{\partial k_j}{\partial z}, \qquad \frac{\partial k_i}{\partial z} = h\nabla^2 H(Y_i) \frac{\partial Y_i}{\partial z}.$$

However, computing the Jacobian matrices $\frac{\partial Y_i}{\partial z}$ and $\frac{\partial k_i}{\partial z}$ in order to evaluate the gradient of $S(z,h)$ is in general highly inefficient. It is standard in automatic differentiation theory that the gradient of a scalar function of several variables is more efficiently evaluated by applying the chain rule in reverse order. In our case, the computation of the expensive Jacobian matrices $\frac{\partial Y_i}{\partial z}$ and $\frac{\partial k_i}{\partial z}$ can be avoided as follows. Given arbitrary column vectors $\hat{Y}_i, v_i \in \mathbb{R}^{2d}$ ($1 \leq i \leq s$), we have that

$$\begin{aligned} \frac{\partial S(z,h)}{\partial z} \quad = \quad & \sum_{i=1}^{s} b_i k_i^T \frac{\partial Y_i}{\partial z} + \sum_{i,j=1}^{s} \beta_{ij} k_i^T J^{-1} \frac{\partial k_j}{\partial z} \\ & + \sum_{i=1}^{s} \hat{Y}_i^T \left(I + J^{-1} \sum_{j=1}^{s} \alpha_{ij} \frac{\partial k_j}{\partial z} - \frac{\partial Y_i}{\partial z} \right) \\ & + \sum_{i=1}^{s} v_i^T \left(h\nabla^2 H(Y_i) \frac{\partial Y_i}{\partial z} - \frac{\partial k_i}{\partial z} \right), \end{aligned}$$

and reordering terms we obtain

$$\begin{aligned} \frac{\partial S(z,h)}{\partial z} \quad = \quad & \sum_{i=1}^{s} \hat{Y}_i + \sum_{i=1}^{s} \left(b_i k_i^T - \hat{Y}_i^T + h v_i^T \nabla^2 H(Y_i) \right) \frac{\partial Y_i}{\partial z} \\ & + \sum_{i=1}^{s} \left(\sum_{j=1}^{s} \left(\beta_{ji} k_j^T J^{-1} + \alpha_{ji} \hat{Y}_j^T J^{-1} \right) - v_i^T \right) \frac{\partial k_i}{\partial z}. \end{aligned}$$

Now, we define the vectors $\hat{Y}_i$ and v_i ($1 \le i \le s$) as

$$\begin{aligned}
\hat{Y}_i &= b_i k_i + h\nabla^2 H(Y_i) v_i, \\
v_i &= J^{-T} \sum_{j=1}^{s} (\beta_{ji} k_j + \alpha_{ji} \hat{Y}_j) = -J^{-1} \sum_{j=1}^{s} (\beta_{ji} k_j + \alpha_{ji} \hat{Y}_j),
\end{aligned}$$

so that $\frac{\partial S(z,h)}{\partial z} = \sum_{i=1}^{s} \hat{Y}_i$. We thus have that

$$J^{-1} \nabla S(z,h) = \Theta(z,h), \tag{11}$$

where

$$\Theta(z,h) = h \sum_{i=1}^{s} \left(b_i f(Y_i) + f'(Y_i) v_i \right), \tag{12}$$

$f(z) = J^{-1} \nabla H(z)$ and $f'(z) = J^{-1} \nabla^2 H(z)$, and

$$Y_i = z + h \sum_{j=1}^{s} \alpha_{ij} f(Y_j), \quad i = 1, \dots, s, \tag{13}$$

$$v_i = h \sum_{j=1}^{s} \left(\gamma_{ij} f(Y_j) - \alpha_{ji} f'(Y_j) v_j \right), \quad i = s, \dots, 1, \tag{14}$$

where $\gamma_{ij} = \beta_{ij} - b_j \alpha_{ji}$ for all i, j.

It is worth noting that the definition of one step $y^* = \psi_h(y)$ implicitly defined by

$$y^* = y + \Theta(\tfrac{1}{2}(y + y^*), h) \tag{15}$$

where $\Theta(z,h)$ is given by (12)-(14) also makes sense in the general case of a system of autonomous (non-necessarily Hamiltonian) ODEs (3) where $f'(Y_j)$ in (12)-(14) denotes the Jacobian matrix of f evaluated at Y_j.

Obviously, $\Theta(z,h)$ is explicitly defined when the matrix (α_{ij}) is strictly lower triangular, and the definition of one step $y^* = \psi_h(y)$ of the method (6) is only implicit in y^*. For an arbitrary matrix (α_{ij}), the method is fully implicit, and it makes more sense expressing one step of the method as

$$\psi_h(y) = y + h \sum_{i=1}^{s} \left(b_i f(Y_i) + f'(Y_i) v_i \right), \tag{16}$$

where Y_i, v_i ($i = 1, \ldots, s$) are implicitly defined by

$$\begin{aligned} Y_i &= y + h \sum_{j=1}^{s} \left(a_{ij} f(Y_j) + \tfrac{1}{2} f'(Y_j) v_j \right), & (17) \\ v_i &= h \sum_{j=1}^{s} \left(\gamma_{ij} f(Y_j) - \alpha_{ji} f'(Y_j) v_j \right), & (18) \end{aligned}$$

where

$$a_{ij} = \alpha_{ij} + \frac{b_j}{2}, \quad \gamma_{ij} = \beta_{ij} - b_j \alpha_{ji} \quad \forall i, j. \qquad (19)$$

Observe that in the particular case where

$$\beta_{ij} = b_j \alpha_{ji} = b_j a_{ji} - \frac{1}{2} b_i b_j \text{ for all } i, j \qquad (20)$$

(hence $b_i a_{ij} + b_j a_{ji} = b_i b_j$ since $\beta_{ji} = -\beta_{ij}$), we have that $\gamma_{ij} = 0$, and thus $v_i = 0$ for all i, and the scheme (16)-(18) is just a symplectic Runge-Kutta scheme.

We want to stress that the computational complexity of the scheme (16)-(18) is not reduced if one takes $\beta_{ij} = 0$ for all i, j, and thus our generalization to the methods introduced in [13] increases the number of parameters of the method for free.

2.3 Order barriers

The method (16)–(19) when applied to the equation $\dot{y} = y$ gives $y^* = R(h) y$, where

$$R(h) = \frac{2 + Q(h)}{2 - Q(h)}, \qquad (21)$$

where $Q(h)$ is in general a rational function given by

$$\begin{aligned} Q(h) &= h \left(e^T B - h e^T (I + hA^T)^{-1} \Gamma \right) (I - hA)^{-1} e, \\ &= e^T (I + hA^T)^{-1} (hB + h^2 C)(I - hA)^{-1} e. \end{aligned}$$

where we use the notation $B = \mathrm{diag}(b_1, \ldots, b_s)$, $e^T = (1, \ldots, 1)$, $A = (\alpha_{ij})_{i,j=1}^s$, $C = (\beta_{ij})_{i,j=1}^s$, and $\Gamma = (\gamma_{ij})_{i,j=1}^s = C + A^T B$ (where $\mathrm{diag}(v)$ with a vector v stands for a square diagonal matrix with v in the main diagonal).

In the particular case of methods of explicit type, $Q(h)$ is a polynomial of degree $d \leq 2s$.

A method of order p necessarily has to satisfy that $R(h) - e^h = \mathcal{O}(h^{p+1})$ as $h \to 0$, which by virtue of (21) is equivalent to

$$Q(h) = 2\tanh(h/2) + \mathcal{O}(h^{p+1}) \quad \text{as } h \to 0.$$

This gives an order barrier of $2s$ for methods of explicit type, since in that case, $Q(h)$ is a polynomial of degree $\leq 2s - 1$. In the particular case of time-symmetric methods of explicit type, the degree of $Q(h)$ can be seen to be at most $(4s + 2)/3$.

3 The order conditions

The order conditions of the method (16)–(19) applied to the system of autonomous (non-necessarily Hamiltonian) ODEs (3) can be obtained by expanding $y^* = \psi_h(y)$ as a B-series $B(c, y)$, and then comparing it with the B-series expansion of the exact solution. This is the standard procedure to obtain the order conditions for B-series methods [8],[7] which gives rise to one order condition per rooted tree. As ψ_h is by construction symplectic, the B-series $B(c, y)$ will necessarily satisfy the symplecticity conditions [3, 7], which introduce certain dependencies among the set of order conditions.

The coefficients $c(u)$ of the B-series expansion $B(c, y)$ of $\psi_h(y)$ can be obtained with standard techniques by applying the result due to Hairer and Wanner [10] (see also [7]) for the B-series expansion of $hf(B(a, y))$, and the result given by Nørsett and Wolfbrandt [14] for $hf'(B(a, y))B(b, y)$, where $B(a, y)$ and $B(b, y)$ are B-series (with $a(\emptyset) = 1$ and $b(\emptyset) = 0$).

The scheme (16)–(19) is of order p if

$$c(u) = \frac{1}{\gamma(u)} \tag{22}$$

for all rooted trees with $|u| \leq p$ where $\gamma(u)$ (sometimes referred to as the *density* of the rooted tree) is given in [8] . Since by construction the B-series $B(c)$ is a symplectic (or canonical) B-series, such conditions are not independent [3] (see also [7]).

Such a dependency is related to a equivalence relation $\sim$ in $\mathcal{T}$, where two rooted trees are equivalent if they only differ in the location of the root, that is, if they have the same underlying free tree (obtained from the rooted tree by forgetting the location of the root). In particular, $u \circ v \sim v \circ u$ for arbitrary $u, v \in \mathcal{T}$. Thus, each equivalence class of rooted trees can be identified with a free tree. A free tree is said to be superfluous if it can be represented by a rooted tree of the form $u \circ u$ with $u \in \mathcal{T}$. In [3] it is shown that, given a positive integer p, if for each non-superfluous free tree with $n \leq p$ vertices, the order condition (22) of one rooted

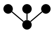
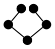
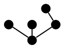
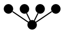
u	$\bullet$					
$\|u\|$	1	3	4	5	5	5
$\gamma(u)$	1	3	4	20	10	5

Table 1: Values of $|u|$ and $\gamma(u)$ for rooted trees representing non-superfluous free trees with up to 5 vertices

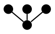
tree u in the equivalence class associated to that free tree is fulfilled, then the order conditions for all rooted trees with $n \leq p$ vertices hold (and thus the method is at least of order p).

Let be $\widehat{T}_p \subset \mathcal{T}$ a set of canonical representatives of the equivalence classes of rooted trees associated to non-superfluous free trees with $n \leq p$ vertices. In Table 1, the subset $\widehat{T}_p$ of rooted trees, corresponding each one to a different equivalence class of non-superfluous free trees with $n \leq 5$ vertices is displayed. A set of independent condition for our scheme to be of order p can then be obtained by considering the condition (22) for each $v \in \widehat{T}_p$.

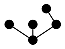
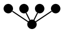
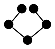
We thus have that the method is of order at least 3 if and only if

$c(\bullet) = 1, \quad c($ $) = \frac{1}{3}.$ (23)

If in addition

$c($  $) = \frac{1}{4},$ (24)

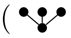
then the method is of order at least 4. If the method satisfy the assumptions (10), so that it is time-symmetric, then (24) automatically holds if the conditions (23) for order three are fulfilled. The method is of order at least 5 if and only if, in addition to (23)–(24), the following conditions hold,

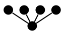
$c($  $) = \frac{1}{10}, \quad c($  $) = \frac{1}{5}, \quad c($  $) = \frac{1}{20}.$ (25)

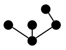
If the method satisfy the symmetry assumptions (10), then (23) together with (25) guarantees that the method is at least of order six.

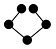
The coefficients $c(u)$ for the trees in Table 1 can be seen to be

$c(\bullet) \;=\; \sum_{i=1}^{s} b_i, \qquad c($ $) \;=\; \sum_{i=1}^{s} \big(b_i\, c_i(\bullet)^2 + 2c_i(\bullet) d_i(\bullet)\big),$

$c($  $) = \sum_{i=1}^{s} \big(b_i\, c_i(\bullet)^3 + 3c_i(\bullet)^2 d_i(\bullet)\big)\,,$

$c($  $) = \sum_{i=1}^{s} \big(b_i\, c_i(\bullet)^4 + 4c_i(\bullet)^3 d_i(\bullet)\big)\,,$

$c($  $) = \sum_{i=1}^{s} \big(b_i\, c_i(\bullet)^2 c_i(\bullet\!\diagdown\!\bullet) + c_i(\bullet)^2 d_i(\bullet\!\diagdown\!\bullet) + 2c_i(\bullet) c_i(\bullet\!\diagdown\!\bullet) d_i(\bullet)\big)\,,$

$c($  $) = \sum_{i=1}^{s} \big(b_i\, c_i(\bullet\!\diagdown\!\bullet)^2 + 2c_i(\bullet\!\diagdown\!\bullet) d_i(\bullet\!\diagdown\!\bullet)\big)\,,$

where for each $i = 1, \dots, s$, the coefficients $c_i(\bullet)$, $c_i(\bullet\!\diagdown\!\bullet)$, $d_i(\bullet)$ and $d_i(\bullet\!\diagdown\!\bullet)$, are given by

$$\begin{aligned} c_i(\bullet) &= \frac{c(\bullet)}{2} + \sum_{j=1}^{s} \alpha_{ij}, \quad c_i(\bullet\!\diagdown\!\bullet) = \frac{c(\bullet\!\diagdown\!\bullet)}{2} + \sum_{j=1}^{s} \alpha_{ij}\, c_j(\bullet), \\ d_i(\bullet) &= \sum_{j=1}^{s} \gamma_{ij}, \quad d_i(\bullet\!\diagdown\!\bullet) = \sum_{j=1}^{s} \gamma_{ij} c_j(\bullet) - \sum_{j=1}^{s} \alpha_{ji}\, d_j(\bullet), \end{aligned}$$

where γ_{ij} is given in terms of the parameters $b_i, \alpha_{ij}, \beta_{ij}$ of the method in (19).

4 Construction of a time-symmetric 6th order method of explicit type

Let us consider the family of time-symmetric methods of explicit type with $s = 4$. We choose the permutation π of the set of indices $\{1, 2, 3, 4\}$ determined by

$$\pi(1) = 1, \quad \pi(2) = 3, \quad \pi(3) = 2, \quad \pi(4) = 4,$$

so that the parameters of the method satisfy (10). This gives a family of time-symmetric methods of explicit type with eight parameters, namely,

$$b_1, b_3, b_4, \beta_{31}, \beta_{32}, \beta_{43}, \alpha_{31}, \alpha_{43}.$$

By imposing the conditions (23) and (25), we get a three-parameter family of time-symmetric sixth order methods of explicit type. We have chosen one particular method of the family which gives reasonably small residuals $|c(u) - 1/\gamma(u)|$ for rooted trees of order 7, 8, and 9.

One step $y^* = \psi_h(y)$ of the resulting scheme is implicitly defined by (15), where $\Theta(z,h)$ is explicitly given by

$$\begin{aligned}\Theta(z,h) &= h\left(\frac{783475}{3359232}f(Y_2)+\frac{783475}{3359232}f(Y_3)+\frac{896141}{1679616}f(Y_4)\right)\\ &\quad +h\left(f'(Y_1)v_1+f'(Y_2)v_2+f'(Y_3)v_3+f'(Y_4)v_4\right),\end{aligned}$$

where

$$\begin{aligned}
Y_1 &= z,\\
Y_2 &= z - h\,\frac{18}{55}f(Y_1),\\
Y_3 &= z + h\,\frac{18}{55}f(Y_1),\\
Y_4 &= z + h\left(\frac{9}{70}f(Y_2)-\frac{9}{70}f(Y_3)\right),\\
v_4 &= h\left(\frac{3240577}{78382080}f(Y_2)-\frac{3240577}{78382080}f(Y_3)\right),\\
v_3 &= -v_2 = h\left(-\frac{11277773}{78382080}f(Y_1)+\frac{33275}{559872}f(Y_2)+\frac{8617423}{78382080}f(Y_4)\right)\\
&\qquad\qquad + h\,\frac{9}{70}f'(Y_4)v_4,\\
v_1 &= h\,\frac{5294873}{78382080}(f(Y_3)-f(Y_2)) + h\,\frac{18}{55}(f'(Y_2)v_2 - f'(Y_3)v_3).
\end{aligned}$$

5 Concluding remarks

We have presented a new family of one-step symplectic integration schemes for Hamiltonian systems of the general form $\dot{y} = J^{-1}\nabla H(y)^T$. Our new integration methods are constructed by defining a generating function, which automatically determines a symplectic map. Such methods generalize those introduced by Miesbach and Pesch [13] in such a way that additional parameters are introduced for free (by keeping the same number of order conditions, and essentially the same computational cost). We show that the evaluation of the gradient of the generating function can be performed efficiently (without the need to compute Jacobian matrices), by evaluating one gradient of the Hamiltonian function as well as one product of Hessian times a vector per stage.

We have focused on methods that are symmetric and of explicit type (i.e., with explicitly given generating function, so that they can be seen as generalizations of the implicit midpoint rule). We give an order barrier of $(4s+2)/3$ for s-stage symmetric methods of explicit type (which, compared to the order $2s$ of s-stage Gauss methods, is in principle a negative result). This gives in particular $s \geq 4$ for methods of order 6. We actually construct a three-parameter family of particular symmetric sixth order methods with $s = 4$ stages, and choose a particular method with optimized coefficients. We however believe that more efficient methods can be constructed by considering more stages. Methods of order eight with optimized coefficients (intended to minimize the error constants) are currently under construction.

It is still unclear whether there exist methods within our family of symplectic integrators that, conveniently implemented, can be competitive with respect to existing symplectic schemes for general Hamiltonian systems such as Gauss methods. According to our preliminary numerical experiments, an advantage of the new methods with respect to Gauss schemes seems to be the possibility of obtaining a considerably better initial guess (by using generalized Hermite interpolation based on available data of previous steps) for the implicit equations to be solved at each step. Furthermore, we believe that a very efficient implementation can be obtained by adapting the ideas in [12] to the new class of schemes. This, together with the construction of optimized high order methods, is the subject of ongoing work.

Acknowledgement: This work has been partially supported by Ministerio de Educación y Ciencia (Spain) under project MTM2007-61572, and by Universidad del Pais Vasco / Euskal Herriko Unibertsitatea under project EHU08/43. □

References

[1] A. L. Aráujo, A. Murua, J. M. Sans-Serna, *Symplectic methods based on decompositions*, SIAM J. Numer. Anal., 34 (1997), 1926–1947.

[2] M. P. Calvo, J. M. Sanz-Serna, *Variable steps for symplectic integrators*, Numerical Analysis 1991, Griffiths, D. F. and Watson, G. A. eds., Longman, London, 1992.

[3] M. P. Calvo, J. M. Sanz-Serna, *Canonical B-series*, Numer. Math. **67** (1994), 161–175.

[4] P. J. Channell, J. C. Scovell, *Symplectic integration of Hamiltonian systems*, Nonlinearity **3** (1990), 231–259.

[5] K. Feng, *Difference schemes for Hamiltonian formalism and symplectic geometry*, J. Comp. Math. **4** (1986), 279–289.

[6] K. Feng, H. M. Wu, M. Z. Qin, D. L. Wang, *Construction of canonical difference schemes for Hamiltonian formalism via generating functions*, J. Comp. Math. **7** (1989), 71–96.

[7] E. Hairer, C. Lubich, G. Wanner, *Geometric Numerical Integration. Structure-Preserving Algorithms for Ordinary Differential Equations,* Springer Series in Computational Mathematics **31**(2002).

[8] E. Hairer, S. P. Nørsett, G. Wanner, *Solving Ordinary Differential Equations I. Nonstiff Problems, 2nd. edition*, Springer Series in Computational Mathematics **8**, Springer Berlin (1993).

[9] E. Hairer, *Variable time step integration with symplctic methods*, Appl. Numer. Math. **25** (1997), 219–227.

[10] E. Hairer, G. Wanner, *On the Butcher group and general multi-value methods*, Computing *13* (1974), 1–15.

[11] R. I. McLachlan, G. R. W. Quispel, *Splitting methods*, Acta Numerica **11** (2002) 341–434.

[12] R. I. McLachlan, *A New Implementation of Symplectic Runge-Kutta Methods*, SIAM J. Scientific Computing 29(4) (2007), 1637–1649.

[13] S. Miesbach and H. J. Pesch, *Symplectic phase flow approximation for the numerical integration of canonical systems*, Numer. Math. **61** (1992), 501–521.

[14] S. P. Nørsett, A. Wolfbrandt, *Order conditions for Rossenbrock type methods*, Numer. Math., vol 32 (1979), pp. 1–15.

[15] S. Reich, *Backward error analysis for numerical integrators*, SIAM J Numer. Anal. **36** (1999), 1549–1570.

[16] P. W. Sharp, *Comparisons of integrators on a diverse collection of restricted three-body test problems*, IMA J. of Num. Anal. **24** (2004), 557–575.

[17] R. Giering and T. Kaminski *Using TAMC to generate efficient adjoint code: Comparison of automatically generated code for evaluation of first and second order derivatives to hand written code from the Minpack-2 collection*, In C. Faure, editor, Automatic Differentiation for Adjoint Code Generation,pages 31-37. INRIA, Sophia Antipolis, France, 1998.