



Preserving Poisson Structure and Orthogonality in Numerical Integration of Differential Equations

L. O. JAY

Department of Mathematics
14 MacLean Hall
The University of Iowa
Iowa City, IA 52242-1419, U.S.A.
na.ljay@na-net.ornl.gov
ljay@math.uiowa.edu

(Received July 2002; revised and accepted February 2003)

Abstract—We consider the numerical integration of two types of systems of differential equations. We first consider Hamiltonian systems of differential equations with a Poisson structure. We show that symplectic Runge-Kutta methods preserve this structure when the Poisson tensor is constant. Using nonlinear changes of coordinates this structure can also be preserved for nonconstant Poisson tensors, as exemplified on the Euler equations for the free rigid body. We also consider orthogonal flows and the closely related class of isospectral flows. To numerically preserve the orthogonality property we take the approach of formulating an equivalent system of differential-algebraic equations (DAEs) and of integrating the system with a special combination of a particular class of Runge-Kutta methods. This approach requires only matrix-matrix products and can preserve geometric properties of the flow such as reversibility. © 2004 Elsevier Ltd. All rights reserved.

Keywords—Differential-algebraic equations, Hamiltonian systems, Orthogonality, Poisson tensor, Runge-Kutta methods.

1. INTRODUCTION

This paper is concerned with the numerical integration of two types of systems of differential equations having certain intrinsic structures. We first consider Poisson systems. The preservation of Poisson structure by numerical integration has been the subject of several articles [1–5], especially for Poisson systems on Lie groups, such as the rigid body equations [6–10]. We then consider orthogonal flows and the closely related class of isospectral flows. The numerical preservation of orthogonality and isospectrality properties has also been the topic of several articles,

This material is based upon work supported by the National Science Foundation under Grant No. 9983708. This research was also supported in part by NASA Award No. 1210679, JPL subcontract No. 1213833. Part of this work was done during a visit at the Department of Mathematics of the University of Geneva, Switzerland.

The author would like to thank Ernst Hairer and Gerhard Wanner from the Department of Mathematics of the University of Geneva, Switzerland, for their invitation. Special thanks go to Ernst Hairer and Jean-Claude Hausmann for some fruitful discussions.

e.g., [11–28] where most of the proposed methods do not preserve geometric properties of the flow, such as reversibility.

This paper is organized as follows. In Section 2, we characterize Hamiltonian systems with a Poisson structure. In Section 3, we show that symplectic Runge-Kutta methods preserve the Poisson structure when the Poisson tensor is constant. We consider nonlinear changes of coordinates in Section 4 when the Poisson tensor is not constant. An application to the Euler equations for the free rigid body is given in details. Then in Section 5, we consider orthogonal and isospectral flows. To preserve the orthogonality property, we propose in Section 6 an approach based on integrating an equivalent system of differential-algebraic equations (DAEs) by a special class of Runge-Kutta type methods. This approach requires only matrix-matrix products and, is thus, amenable to parallelization. Moreover, geometric properties of the flow can also be preserved.

2. POISSON SYSTEMS

We consider the following *Hamiltonian system* of ordinary differential equations (ODEs) in \mathbb{R}^m with a given initial value y_0 at t_0 :

$$y' = J(y)H_y^\top(y), \quad y(t_0) := y_0. \quad (1)$$

Here, $J(y)$ is a skew-symmetric matrix

$$J(y) = -J^\top(y)$$

satisfying the *Jacobi identity*

$$\sum_{l=1}^m \left(\frac{\partial J_{ij}(y)}{\partial y_l} J_{lk}(y) + \frac{\partial J_{jk}(y)}{\partial y_l} J_{li}(y) + \frac{\partial J_{ki}(y)}{\partial y_l} J_{lj}(y) \right) = 0, \quad \text{for } i, j, k = 1, \dots, m.$$

The matrix $J(y)$ is called a *Poisson tensor* and (1) is called a *Poisson system*. We can define the following operation between smooth functions $F, G : \mathbb{R}^m \rightarrow \mathbb{R}$:

$$\{F, G\}(y) := F_y(y)J(y)G_y^\top(y)$$

defining another function $\{F, G\} : \mathbb{R}^m \rightarrow \mathbb{R}$. Under the above assumptions for $J(y)$, this bilinear operation satisfies the properties of *Poisson bracket*:

- (1) *Skew-symmetry*: $\{F, G\} = -\{G, F\}$;
- (2) *Jacobi identity*: $\{\{F, G\}, H\} + \{\{G, H\}, F\} + \{\{H, F\}, G\} = 0$;
- (3) *Leibniz's rule*: $\{F \cdot G, H\} = F \cdot \{G, H\} + G \cdot \{F, H\}$.

A transformation $\Phi : \mathbb{R}^m \rightarrow \mathbb{R}^m$ is said to be *Poisson* if it preserves the Poisson bracket, i.e., if

$$\{F, G\} \circ \Phi = \{F \circ \Phi, G \circ \Phi\}.$$

This is equivalent to the following condition:

$$\Phi_y(y)J(y)\Phi_y^\top(y) = J(\Phi(y)).$$

It can be shown that the flow $\Phi^h(y(t)) := y(t+h)$ of (1) is a Poisson transformation [29], i.e., we have

$$\left(\frac{\partial y(t)}{\partial y_0} \right) J(y_0) \left(\frac{\partial y(t)}{\partial y_0} \right)^\top = J(y(t)). \quad (2)$$

When $J(y)$ is invertible this is equivalent to the condition

$$\left(\frac{\partial y(t)}{\partial y_0}\right)^\top J^{-1}(y(t)) \left(\frac{\partial y(t)}{\partial y_0}\right) = J^{-1}(y_0).$$

For the *standard symplectic matrix*

$$J_s := \begin{pmatrix} O & I \\ -I & O \end{pmatrix} \tag{3}$$

we have $J_s^{-1} = -J_s$. Hence, we obtain the equivalent *symplecticness* condition

$$\left(\frac{\partial y(t)}{\partial y_0}\right)^\top J_s \left(\frac{\partial y(t)}{\partial y_0}\right) = J_s.$$

Note that this relation is generally not satisfied when $J(y)$ is not the standard symplectic matrix (3), since we have for a constant matrix J

$$\frac{d}{dt} \left(\left(\frac{\partial y(t)}{\partial y_0}\right)^\top J \left(\frac{\partial y(t)}{\partial y_0}\right) \right) = \left(\frac{\partial y(t)}{\partial y_0}\right)^\top (J^2 H_{yy}(y(t)) - H_{yy}(y(t))J^2) \left(\frac{\partial y(t)}{\partial y_0}\right),$$

and the matrix $J^2 H_{yy}(y) - H_{yy}(y)J^2$ does not vanish in general. From the skew-symmetry of the Poisson tensor it can be seen that the Hamiltonian remains constant, i.e., $H(y(t)) = \text{Const}$. A function $C(y)$ such that $C_y(y)J(y) = 0$ is called a *Casimir* function and remains also constant under the flow. Note that the Hamiltonian $H(y)$ is generally not a Casimir function.

3. RUNGE-KUTTA METHODS FOR CONSTANT POISSON TENSOR

In this section, we consider the application of Runge-Kutta methods to (1) when J is a constant skew-symmetric matrix.

DEFINITION 1. *One step of an s -stage implicit Runge-Kutta (IRK) method applied to the system $y' = f(y)$, with initial values y_0 at t_0 and step-size h reads*

$$Y_i = y_0 + h \sum_{j=1}^s a_{ij} f(Y_j), \quad \text{for } i = 1, \dots, s, \tag{4a}$$

$$y_1 = y_0 + h \sum_{i=1}^s b_i f(Y_i). \tag{4b}$$

We are interested in characterizing methods which preserve the Poisson structure (2) of (1), i.e., satisfying

$$\left(\frac{\partial y_1}{\partial y_0}\right) J \left(\frac{\partial y_1}{\partial y_0}\right)^\top = J. \tag{5}$$

When J is the standard symplectic matrix (3) well-known fundamental conditions on the Runge-Kutta coefficients have been derived to preserve the Poisson and symplectic structure of the flow [30–32], they are

$$b_i a_{ij} + b_j a_{ji} - b_i b_j = 0, \quad \text{for } i, j = 1, \dots, s. \tag{6}$$

A question that we address here is to know if these conditions are sufficient to preserve the Poisson structure of the flow in the slightly more general situation when J is a constant skew-symmetric matrix. In the following theorem, we show that this is indeed the case, clarifying the response [33] to the comments [34] on the article [5].

THEOREM 2. *For a constant skew-symmetric matrix J , the Poisson structure (2) of the flow of (1) is preserved by Runge-Kutta methods satisfying (6).*

PROOF. Since the matrix J is skew-symmetric, there exists an orthogonal matrix U such that

$$UJU^\top = \begin{pmatrix} \tilde{J} & O \\ O & O \end{pmatrix}$$

with \tilde{J} skew-symmetric of the form

$$\tilde{J} = \begin{pmatrix} O & D \\ -D & O \end{pmatrix},$$

where $D := \text{diag}(d_1, \dots, d_n)$ is a diagonal matrix with $d_i > 0$. This is a standard result in linear algebra, see, e.g., [35]. We consider the linear change of variables $z = (v, w) := KUy$ with

$$K := \begin{pmatrix} C & O \\ O & I \end{pmatrix}$$

and $C := \text{diag}(d_1^{-1/2}, \dots, d_n^{-1/2}, d_1^{-1/2}, \dots, d_n^{-1/2})$. In the new coordinates, we have

$$z' = \begin{pmatrix} J_s & O \\ O & O \end{pmatrix} h_z^\top(z), \quad (7)$$

where J_s is the standard symplectic matrix (3) and $h(z) := H((KU)^{-1}z)$, see also [36, Proposition 3.1.2]. Runge-Kutta methods are invariant under linear changes of coordinates, hence, since the variables w are left constant the Poisson condition (5) becomes equivalent to

$$\begin{pmatrix} \frac{\partial v_1}{\partial v_0} \end{pmatrix} J_s \begin{pmatrix} \frac{\partial v_1}{\partial v_0} \end{pmatrix}^\top = J_s.$$

This also corresponds to the preservation of the symplectic structure for (7) which can be seen as a Hamiltonian system for the variables v only. Runge-Kutta methods satisfying (6) are known to be symplectic [30–32]. This argument ends the proof. \blacksquare

Theorem 2 generalizes the result given in [5] for quadratic Hamiltonian functions to general Hamiltonian functions, and also the result given in [6] for the implicit midpoint rule to symplectic Runge-Kutta methods. If the Poisson tensor J is of the form

$$\begin{pmatrix} O & E & O \\ -E^\top & O & O \\ O & O & O \end{pmatrix}$$

for a matrix E , we can also consider the application of partitioned Runge-Kutta methods [37]. With a proof similar to the one given above, it can be shown that symplectic partitioned Runge-Kutta methods also preserve the Poisson structure in this situation.

4. POISSON METHODS FOR NONCONSTANT POISSON TENSOR

When the Poisson tensor $J(y)$ is not constant, the Poisson structure is generally not preserved even by Runge-Kutta methods satisfying assumption (6), see, e.g., [6] for the midpoint rule. One way to circumvent this problem is to consider one-to-one nonlinear transformations $v = k(y)$ such that the corresponding system in the new coordinates has a constant Poisson tensor. By Darboux's theorem [38], such transformations are known to exist locally. The main difficulty is obviously to obtain them. We give, in Section 4.1, an example with various explicit global

transformations leading to constant Poisson tensors. Denoting the inverse transformation by $y = g(v)$ and defining $h(v) := H(g(v))$, from $v' = k_y(y)y'$ and (1), we get

$$v' = j(v)h_v^\top(v),$$

where

$$j(v) = k_y(g(v))J(g(v))k_y^\top(g(v))$$

is the Poisson tensor for v . Multiplying (2) from the left by $k_y(y(t))$ and from the right by $k_y^\top(y(t))$ we obtain

$$\left(\frac{\partial v(t)}{\partial v_0}\right) \left(\frac{\partial v_0}{\partial y_0}\right) J(y_0) \left(\frac{\partial v_0}{\partial y_0}\right)^\top \left(\frac{\partial v(t)}{\partial v_0}\right)^\top = j(v(t)).$$

Since $\frac{\partial v_0}{\partial y_0} = k_y(y_0)$, we get

$$\left(\frac{\partial v(t)}{\partial v_0}\right) j(v_0) \left(\frac{\partial v(t)}{\partial v_0}\right)^\top = j(v(t)).$$

Hence, the original Poisson structure is preserved if and only if it is preserved in the new coordinates. We are interested in transformations $v = k(y)$, such that the corresponding Poisson tensor $j(v)$ for v is constant. When such a transformation $k(y)$ can be obtained explicitly, we can apply for example a Poisson-symplectic Runge-Kutta method to the corresponding Hamiltonian system in order to preserve the Poisson structure of the flow. Actually, this application can be done purely formally, since we can re-express the method in terms of the original variables y , and this is equivalent to apply the method directly to the following implicit system of ODEs:

$$k(y)' = k_y(y)J(y)H_y^\top(y).$$

The application of implicit Runge-Kutta methods to this system is defined as follows.

DEFINITION 3. *One step of an s -stage implicit Runge-Kutta (IRK) method applied to the implicit system of ODEs $k(y)' = d(y)$ with initial values y_0 at t_0 and step-size h reads*

$$k(Y_i) = k(y_0) + h \sum_{j=1}^s a_{ij}d(Y_j), \quad \text{for } i = 1, \dots, s,$$

$$k(y_1) = k(y_0) + h \sum_{j=1}^s b_j d(Y_j).$$

An equivalent definition is to consider the inverse transformation k^{-1} of k , leading to

$$Y_i = k^{-1} \left(k(y_0) + h \sum_{j=1}^s a_{ij}d(Y_j) \right), \quad \text{for } i = 1, \dots, s,$$

$$y_1 = k^{-1} \left(k(y_0) + h \sum_{j=1}^s b_j d(Y_j) \right),$$

and which can be seen as a nonlinear generalization of Runge-Kutta methods applied to $y' = f(y)$ when having $d(y) := k_y(y)f(y)$, $k = \text{id}$ corresponding to the usual definition (4).

4.1. The Euler Equations for the Free Rigid Body

We consider the Hamiltonian system (1) in \mathbb{R}^3 with Poisson tensor

$$J(y) = \begin{pmatrix} 0 & -y_3 & y_2 \\ y_3 & 0 & -y_1 \\ -y_2 & y_1 & 0 \end{pmatrix}. \quad (8)$$

For $H(y) = \sum_{i=1}^3 y_i^2/I_i$ we obtain the *Euler equations* for the free rigid body

$$y_1' = \frac{I_2 - I_3}{I_2 I_3} y_2 y_3, \quad y_2' = \frac{I_3 - I_1}{I_3 I_1} y_3 y_1, \quad y_3' = \frac{I_1 - I_2}{I_1 I_2} y_1 y_2, \quad (9)$$

where the constants I_i are the moments of inertia about the coordinate axes and the variables y_i are the corresponding body angular momenta. We are looking for a transformation $v = k(y)$, such that

$$k_y(y) J(y) k_y^\top(y) = \begin{pmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}.$$

We must, therefore, determine $k(y)$, such that the following system of partial differential equations is satisfied:

$$\begin{aligned} 1 &= y_3(k_{1y_2} k_{2y_1} - k_{1y_1} k_{2y_2}) + y_2(k_{1y_1} k_{2y_3} - k_{1y_3} k_{2y_1}) + y_1(k_{1y_3} k_{2y_2} - k_{1y_2} k_{2y_3}), \\ 0 &= y_3(k_{1y_2} k_{3y_1} - k_{1y_1} k_{3y_2}) + y_2(k_{1y_1} k_{3y_3} - k_{1y_3} k_{3y_1}) + y_1(k_{1y_3} k_{3y_2} - k_{1y_2} k_{3y_3}), \\ 0 &= y_3(k_{2y_2} k_{3y_1} - k_{2y_1} k_{3y_2}) + y_2(k_{2y_1} k_{3y_3} - k_{2y_3} k_{3y_1}) + y_1(k_{2y_3} k_{3y_2} - k_{2y_2} k_{3y_3}). \end{aligned}$$

Since $(k_3(y))' = 0$, $k_3(y)$ is an invariant and we can take the Casimir function

$$k_3(y) = L(y) := \frac{1}{2} (y_1^2 + y_2^2 + y_3^2).$$

Hence, the above last two equations are satisfied and we are left with the first one. Taking $k_1(y) := y_1$ we must find a function $k_2(y)$ satisfying

$$1 = y_2 k_{2y_3} - y_3 k_{2y_2}.$$

It can be seen that $k_2(y) := \arctan(y_3/y_2)$ satisfies this equation. The original variables (y_1, y_2, y_3) are related to the symplectically conjugate variables (v_1, v_2) by

$$y_1 = v_1, \quad y_2 = \sqrt{2L - v_1^2} \cos(v_2), \quad y_3 = \sqrt{2L - v_1^2} \sin(v_2).$$

For the Euler equations (9) we obtain the standard Hamiltonian system

$$v_1' = \frac{1}{2} \left(\frac{1}{I_3} - \frac{1}{I_2} \right) (2L - v_1^2) \sin(2v_2), \quad v_2' = v_1 \left(\frac{\cos^2(v_1)}{I_2} + \frac{\sin^2(v_1)}{I_3} - \frac{1}{I_1} \right)$$

corresponding to the Hamiltonian function

$$h(v_1, v_2) = \frac{1}{2} \left(\frac{v_1^2}{I_1} + \frac{(2L - v_1^2) \cos^2(v_2)}{I_2} + \frac{(2L - v_1^2) \sin^2(v_2)}{I_3} \right).$$

Two other similar transformations and implicit systems of ODEs can be obtained by permutation of y_1, y_2 , and y_3 . The preservation of the Poisson structure (8) is ensured for each of these systems by applying a Runge-Kutta method satisfying (6).

We now present another global transformation leading to a constant Poisson structure. Assuming $0 < I_1 < I_2 < I_3$, there is an infinite number of ways to write the Euler equations (9) as a Hamiltonian system (1) with a Poisson structure. Considering the linear combinations $K(y) := aH(y) + bL(y)$ and $N(y) := cH(y) + dL(y)$ with real constants a, b, c , and d satisfying $ad - bc = 1$, we can express the Euler equations as

$$y' = -K_y^\top(y) \times N_y^\top(y),$$

where \times denotes the cross product of two vectors in \mathbb{R}^3 . $N(y)$ corresponds to a Hamiltonian function and $-K_y^\top(y) \times \cdot$ to a Poisson tensor. For the choice

$$a = 1, \quad b = -\frac{1}{I_3}, \quad c = -C, \quad d = \frac{C}{I_1}, \quad \text{with } C := \frac{1}{(1/I_1) - (1/I_3)},$$

we obtain the transformation given by Holm and Marsden in [39] (see also [29, Chapter 15]), i.e.,

$$K(y) = \frac{1}{2} \left(\frac{1}{I_1} - \frac{1}{I_3} \right) y_1^2 + \frac{1}{2} \left(\frac{1}{I_2} - \frac{1}{I_3} \right) y_2^2, \tag{10a}$$

$$N(y) = \frac{C}{2} \left(\frac{1}{I_1} - \frac{1}{I_2} \right) y_2^2 + \frac{1}{2} y_3^2, \tag{10b}$$

with $-K_y^\top(y) \times \cdot$ corresponding to the Poisson tensor

$$\begin{pmatrix} 0 & 0 & -\left(\frac{1}{I_2} - \frac{1}{I_3}\right) y_2 \\ 0 & 0 & \left(\frac{1}{I_1} - \frac{1}{I_3}\right) y_1 \\ \left(\frac{1}{I_2} - \frac{1}{I_3}\right) y_2 & -\left(\frac{1}{I_1} - \frac{1}{I_3}\right) y_1 & 0 \end{pmatrix} \tag{11}$$

which is a linear combination of (8) corresponding to $L_y^\top(y) \times \cdot$ and of

$$\begin{pmatrix} 0 & -\frac{y_3}{I_3} & \frac{y_2}{I_2} \\ \frac{y_3}{I_3} & 0 & -\frac{y_1}{I_1} \\ -\frac{y_2}{I_2} & \frac{y_1}{I_1} & 0 \end{pmatrix} \tag{12}$$

corresponding to $H_y^\top(y) \times \cdot$. Defining the new variables

$$v_1(y) := \arctan \left(\frac{K_1 y_2}{K_2 y_1} \right), \quad v_2(y) := y_3,$$

where

$$K_1 := \frac{1}{\sqrt{(1/I_1) - (1/I_3)}}, \quad K_2 := \frac{1}{\sqrt{(1/I_2) - (1/I_3)}},$$

the variable v_1 and v_2 are symplectically conjugate and satisfy

$$v_1' = \frac{1}{K_1 K_2} v_2, \quad v_2' = -\frac{K_2 R^2}{2K_1 K_3^2} \sin(2v_1), \tag{13}$$

where

$$K_3 := \frac{1}{\sqrt{C((1/I_1) - (1/I_2))}}, \quad R := \sqrt{2K(y_0)}.$$

These equations correspond up to a scale factor to the pendulum equations with constant Poisson structure

$$\frac{1}{K_1 K_2} \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$$

and with Hamiltonian $h(v_1, v_2) = (1/2)v_2^2 + (K_2^2 R^2 / 2K_3^2) \sin^2(v_1) = N$. Note that the above Hamiltonian system is separable and can, thus, be integrated by explicit symplectic methods [37,40]. The original variables (y_1, y_2, y_3) are related to the variables (v_1, v_2) by

$$y_1 = K_1 R \cos(v_1), \quad y_2 = K_2 R \sin(v_1), \quad y_3 = v_2. \quad (14)$$

Note that by using these relations the Casimir function $K(y)$ (10a) is automatically preserved by any integration method applied to (13). If the constant symplectic structure of (13) is preserved in terms of the variables (v_1, v_2) then the Poisson structure (11) is preserved in terms of the variables (y_1, y_2, y_3) , but generally not any other Poisson structure such as the ones corresponding to (8) or (12). We conjecture that the numerical flow of a method able to preserve two linearly independent combinations of the Poisson structures (8) and (12) of the Euler equations (9) must necessarily be the exact flow up to a reparametrization of time. A way to preserve all the Poisson structures of the Euler equations (9) is actually to express directly its analytical solution in terms of Jacobi's elliptic functions cn , sn , and dn [29, Chapter 15], assuming that these functions can be computed exactly.

As a numerical experiment we have applied the two-stage Lobatto IIIA-B partitioned Runge-Kutta method [41–43], more commonly known as the Verlet method [44], with a constant step-size $h = 0.1$ to system (13). The values $(I_1, I_2, I_3)^\top = (2, 1, 2/3)^\top$ for the moments of inertia and the initial values $y(0) = (\cos(1.1), 0, \sin(1.1))^\top$ for the body angular momenta have been taken. We plot in Figure 1 the error in the quantities $H(y), L(y), K(y)$, and $N(y)$ over the interval $[0, 100]$ using the inverse transformations (14). As is expected for a Poisson-symplectic integrator we observe that the errors for these quantities remain bounded. The quantity $K(y)$ is actually preserved up to close to machine precision. The above integrator is not only simple to implement, but is also extremely fast since it is explicit. Higher-order explicit methods can be obtained by composition [45].

5. ORTHOGONAL AND ISOSPECTRAL FLOWS

In this section, we consider orthogonal flows and the closely related class of isospectral flows. First, we consider systems of matrix ODEs in $\mathbb{R}^{m \times m}$

$$Y' = F(t, Y) \quad (15)$$

having the property of *preserving orthogonality*, i.e., we assume that

$$Y_0^\top Y_0 = I \quad \implies \quad Y^\top(t) Y(t) = I. \quad (16)$$

The flow is said to be *orthogonal* and the quantity $Y^\top Y$ is called a *weak invariant* of (15) [46]. Generally the quantity $Y^\top Y$ is not a *strong invariant*, also called *first integral*, which is a quantity $I(Y)$ with the stronger property that $I_Y(Y)F(t, Y) = 0$ for any Y . Examples of systems of matrix ODEs having the property (16) are given by

$$Y' = J(t, Y)Y \quad (17)$$

and

$$Y' = YJ(t, Y), \quad (18)$$

where $J : \mathbb{R} \times \mathbb{R}^{m \times m} \rightarrow \mathbb{R}^{m \times m}$ is for fixed t a function mapping orthogonal matrices to skew-symmetric matrices. We stress the fact that the right-hand side of systems of matrix ODEs

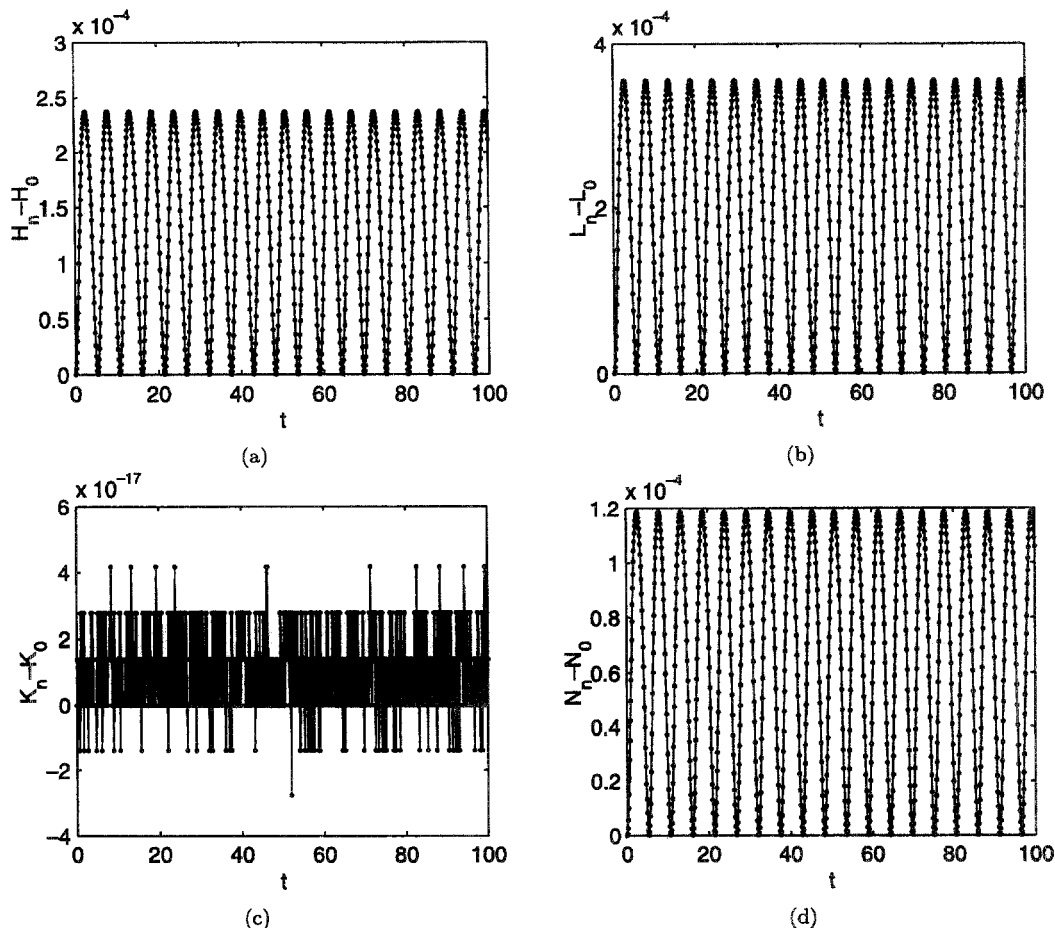


Figure 1. Errors of conserved quantities for the Euler equations using the two-stage Lobatto IIIA-B PRK method applied to (13) with step-size $h = 0.1$.

preserving orthogonality are not necessarily directly expressed in one of these forms as a function of t and Y , unless (15) is explicitly multiplied from the right by $Y^{-1}Y$ or from the left by YY^{-1} . Theorem 1 of [24], also cited as [22, Theorem 1.1] may be somehow misleading: a system of matrix ODEs preserving orthogonality is generally not directly expressed in the form (17) as the one-dimensional example $y' = y^2 - 1$ simply shows.¹

A class of systems closely related to orthogonal flows is given by isospectral flows. The flow in $\mathbb{R}^{m \times m}$ of systems of the form

$$X' = K(t, X)X - XK(t, X), \quad X(t_0) = X_0 \tag{19}$$

with $K(t, X)$ skew-symmetric possesses the property of *isospectrality*, i.e., the eigenvalues of the solution matrix $X(t)$ are invariant under the action of the flow. This can be easily seen by defining $Y(t)$ as the solution of the associated system (17) with $J(t, Y) := K(t, YX_0Y^T)$ and initial condition $Y_0 = I$. The solutions $X(t)$ and $Y(t)$ are related by $X(t) = Y(t)X_0Y^T(t)$, hence the eigenvalues of $X(t)$ are preserved since $Y(t)$ remains orthogonal. Thus, instead of solving (19) directly, we can solve the associated system for $Y(t)$ and then form $X(t)$ by using

¹The proof of Theorem 1 of [24] is slightly incorrect. To have the factorization $F(t, Y) = J(t, Y)Y$ not only for orthogonal matrices, the matrix function $J(t, Y)$ must satisfy $J(t, Y) = F(t, Y)Y^{-1}$, not $J(t, Y) = F(t, Y)Y^T$. Of course, since an orthogonal solution $Y(t)$ satisfies $Y^{-1}(t) = Y^T(t)$, it also verifies that $F(t, Y(t)) = J(t, Y(t))Y(t)$ for some nonunique matrix function $J(t, Y)$, for example $J(t, Y) := F(t, Y)Y^{-1}$ or $J(t, Y) := F(t, Y)Y^T$. However, for nonorthogonal matrices the matrix functions $F(t, Y)$ and $F(t, Y)Y^TY$ are generally distinct. Simply multiplying (15) from the right by Y^TY changes the matrix differential equations.

the aforementioned relation between $X(t)$ and $Y(t)$. Hence, to preserve the isospectrality of $X(t)$ we can simply preserve the orthogonality of $Y(t)$. Similar to orthogonal flows, it is not necessary for an isospectral flow to be the solution of a system of matrix ODEs of the form (19).

6. NUMERICAL ORTHOGONAL INTEGRATION

We are interested in numerical integration methods preserving the orthogonality property (16) of the system of matrix ODEs (15). Several methods have been proposed in the literature. Some of them require the ODEs to be under the specific form (17) [12,16,18,19,21,22,25–27], whereas others do not make such an assumption [14–16,23]. In this paper, no assumption on the form of the matrix ODEs (15) is made. We also do not assume the orthogonality condition to be a strong invariant of the flow, for example in the computation of Lyapunov exponents [11,17–19]. This precludes the direct use of Gauss Runge-Kutta schemes [24].

We propose an approach requiring only matrix-matrix products and not the solution of any linear system such as in [12,22]. Our approach is therefore amenable to parallelization. Moreover, geometric properties of the flow, such as reversibility, can be preserved, which is important for long-term integration. Projection algorithms generally destroy geometric properties. Our approach is based on reformulating the system of matrix ODEs (15) as a system of matrix differential-algebraic equations (DAEs) [47–49]. Additional algebraic variables are introduced and the orthogonality condition is seen as an algebraic constraint on the system.

6.1. Orthogonality Constraints and Jacobian Properties

The orthogonality property (16) can be expressed as algebraic constraints to be enforced. Note that there is not a total of m^2 constraints, but in effect of $m(m + 1)/2$ since $G(Y) := Y^T Y - I$ satisfies $G(Y) = G^T(Y)$, $m(m - 1)/2$ constraints of $G(Y) = 0$ are redundant. We denote by $y := \text{vec}(Y) \in \mathbb{R}^{m^2}$ the vector of length m^2 corresponding to the matrix $Y =: \text{mat}(y) \in \mathbb{R}^{m \times m}$. We denote by $g(y) =: \text{up}(G(Y))$ the $m(m + 1)/2$ linearly independent constraints corresponding to $0 = G(Y) = Y^T Y - I =: \text{sym}(g(y))$. Although quite obvious, the notation $v = \text{up}(S)$ means that the upper triangular part of a symmetric matrix S is taken and stacked into a vector v , whereas the notation $\text{sym}(v) = S$ stands for the inverse operation.

LEMMA 4. *The Jacobian matrix $g_y(y)$ of $g(y) := \text{up}(G(Y))$ with $G(Y) := Y^T Y - I$ satisfies*

$$\begin{aligned} \text{sym}(g_y(y)f) &= Y^T F + (Y^T F)^T, \\ \text{mat}(g_y^T(y)z) &= Y(Z + \text{diag}(Z)), \end{aligned}$$

where $F := \text{mat}(f)$, $Y := \text{mat}(y)$, and $Z := \text{sym}(z)$. If in addition $y \in \mathbb{R}^{m^2}$ satisfies $g(y) = 0$ then we have

$$g_y(y)g_y^T(y) = 2 \cdot \text{diag} \left(2, \underbrace{1, 1, \dots, 1}_{m-1}, 2, \underbrace{1, \dots, 1}_{m-2}, \dots, 2, 1, 1, 2, 1, 2 \right) =: d. \tag{20}$$

PROOF. These results are easily verified for $m = 1, 2$. For $m = 3$ we have

$$g(y) = \begin{pmatrix} y_{11}^2 + y_{21}^2 + y_{31}^2 - 1 \\ y_{11}y_{12} + y_{21}y_{22} + y_{31}y_{32} \\ y_{11}y_{13} + y_{21}y_{23} + y_{31}y_{33} \\ y_{12}^2 + y_{22}^2 + y_{13}^2 - 1 \\ y_{12}y_{13} + y_{22}y_{23} + y_{32}y_{33} \\ y_{13}^2 + y_{23}^2 + y_{33}^2 - 1 \end{pmatrix},$$

$$\frac{\partial g(y)}{\partial y} = \begin{pmatrix} 2y_{11} & 0 & 0 & 2y_{21} & 0 & 0 & 2y_{31} & 0 & 0 \\ y_{12} & y_{11} & 0 & y_{22} & y_{21} & 0 & y_{32} & y_{31} & 0 \\ y_{13} & 0 & y_{11} & y_{23} & 0 & y_{21} & y_{33} & 0 & y_{31} \\ 0 & 2y_{12} & 0 & 0 & 2y_{22} & 0 & 0 & 2y_{32} & 0 \\ 0 & y_{13} & y_{12} & 0 & y_{23} & y_{22} & 0 & y_{33} & y_{32} \\ 0 & 0 & 2y_{13} & 0 & 0 & 2y_{23} & 0 & 0 & 2y_{33} \end{pmatrix}.$$

The results can be verified directly by simple calculation. The extension to arbitrary m is straightforward. \blacksquare

We consider an approximation to the inverse matrix

$$\begin{pmatrix} I & g_y^\top(y) \\ 0 & -g_y(y)g_y^\top(y) \end{pmatrix}^{-1} = \begin{pmatrix} I & g_y^\top(y) (g_y(y)g_y^\top(y))^{-1} \\ 0 & -(g_y(y)g_y^\top(y))^{-1} \end{pmatrix} \quad (21)$$

with $g(y)$ as in Lemma 4. Assuming Y close to being orthogonal, i.e., the corresponding vector y being close to satisfy $g(y) = 0$, we can approximate the expression $(g_y(y)g_y^\top(y))^{-1}$ in (21) by d^{-1} of (20). The application of the corresponding approximate inverse matrix A_0^{-1} to a vector reads as follows:

$$\begin{pmatrix} \delta y \\ \delta z \end{pmatrix} := A_0^{-1} \begin{pmatrix} f \\ g \end{pmatrix} = \begin{pmatrix} f + g_y^\top(y)d^{-1}g \\ -d^{-1}g \end{pmatrix}.$$

In terms of matrices $\Delta Y := \text{mat}(\delta y)$, $\Delta \Lambda := \text{sym}(\delta z) + \text{diag}(\text{sym}(\delta z))$, etc., this leads to the following algorithm which involves no matrix inverse, but only one matrix-matrix product with matrix Y .

ALGORITHM 1. *Approximate inverse of (21) matrix-matrix product.*

1. $\Delta \Lambda := -(1/2) \cdot G$;
2. $C := Y \Delta \Lambda$;
3. $\Delta Y := F - C$.

Similarly, when considering an approximation to the inverse matrix

$$\begin{pmatrix} I & g_y^\top(y) \\ g_y(y) & 0 \end{pmatrix}^{-1} = \begin{pmatrix} I - g_y^\top(y) (g_y(y)g_y^\top(y))^{-1} g_y(y) & g_y^\top(y) (g_y(y)g_y^\top(y))^{-1} \\ (g_y(y)g_y^\top(y))^{-1} g_y(y) & -(g_y(y)g_y^\top(y))^{-1} \end{pmatrix}, \quad (22)$$

we obtain the following algorithm.

ALGORITHM 2. *Approximate inverse of (22) matrix-matrix product.*

1. $A := Y^\top F$;
2. $B := A + A^\top - G$;
3. $\Delta \Lambda := (1/2) \cdot B$;
4. $C := Y \Delta \Lambda$;
5. $\Delta Y := F - C$.

6.2. Methods Based on DAEs

For a system of ODEs $y' = h(t, y)$ with strong or weak invariants $g(t, y) = 0$ it is natural to consider either projecting the numerical solution at selected steps onto $g(t, y) = 0$ or to consider an equivalent system of DAEs

$$y' = h(t, y) - g_y^\top(t, y)z =: f(t, y, z), \quad 0 = g(t, y) \quad (23)$$

with algebraic variables z . Under the assumption that $g_y(t, y)$ is of full rank the system of DAEs (23) is of index 2 since $g_y(t, y)f_z(t, y, z) = -g_y(t, y)g_y^\top(t, y)$ is invertible [47–49]. From

the condition $g_t(t, y(t)) + g_y(t, y(t))h(t, y(t)) = 0$ the algebraic variables z satisfy $z(t) = 0$ as expected.

The approach to preserve orthogonality of the solution that we take here is based on the numerical integration of an equivalent system of DAEs. By introducing a symmetric matrix of algebraic variables Λ we can consider the following semiexplicit system of index 2 DAEs:

$$Y' = F(t, Y) - Y\Lambda, \quad 0 = Y^\top Y - I \tag{24}$$

corresponding to (23) with $\Lambda := Z + \text{diag}(Z)$. Differentiating the constraint $0 = Y^\top(t)Y(t) - I$ once with respect to the independent variable t we obtain

$$0 = F^\top(t, Y(t))Y(t) + F(t, Y(t))Y^\top(t) - \Lambda^\top(t)Y^\top(t)Y(t) - Y^\top(t)Y(t)\Lambda(t) = -\Lambda^\top(t) - \Lambda(t) = -2\Lambda(t).$$

Therefore, Λ satisfies $\Lambda(t) = 0$ as expected.

We can now consider the application of numerical methods to semiexplicit index 2 DAEs. To be more general we can consider a semiexplicit system of DAEs

$$y' = f(t, y, z) = \sum_{m=1}^M f_m(t, y, z), \quad 0 = g(t, y) \tag{25}$$

of index 2, i.e., under the assumptions that $g_y(t, y)f_z(t, y, z)$ is invertible along the solution [47–49] and that $f_1 = f_1(t, y)$ is independent of z [50,51].

DEFINITION 5. *One step of an s -stage specialized (or super) partitioned additive Runge-Kutta (SPARK) method applied to the system of DAEs (25) with initial values y_0, z_0 at t_0 and step-size h reads [50,51]*

$$Y_i = y_0 + h \sum_{j=1}^s \sum_{m=1}^M a_{ij,m} f_m(T_j, Y_j, Z_j), \quad \text{for } i = 1, \dots, s, \tag{26a}$$

$$0 = \sum_{j=1}^s a_{ij,1} g(T_j, Y_j), \quad \text{for } i = 2, \dots, s, \tag{26b}$$

$$y_1 = y_0 + h \sum_{i=1}^s b_i f(T_i, Y_i, Z_i), \tag{26c}$$

$$0 = g(t_1, y_1), \tag{26d}$$

where $T_i := t_0 + c_i h, t_1 := t_0 + h$. The coefficients $(a_{ij,m})_{i,j=1,\dots,s}$, for $m = 1, \dots, M$, are the coefficients of M Runge-Kutta (RK) methods based on the same quadrature formula $(b_i, c_i)_{i=1,\dots,s}$.

It is assumed that the number s of internal stages satisfies $s \geq 2$. SPARK methods satisfying the following assumptions are considered:

$$e_1^\top A_1 = 0_s^\top, \tag{27a}$$

$$e_s^\top A_1 = b^\top, \tag{27b}$$

$$A_1 A_m = \begin{pmatrix} 0_s^\top \\ N \end{pmatrix}, \quad \text{for } m = 2, \dots, M, \tag{27c}$$

$$\begin{pmatrix} N \\ b^\top \end{pmatrix} \text{ is invertible,} \quad \Omega := \begin{pmatrix} N \\ b^\top \end{pmatrix}^{-1}. \tag{27d}$$

These assumptions are satisfied for example by the s -stage Lobatto SPARK families with $M = 5$ and A_1, A_2, A_3, A_4, A_5 being the RK matrices of Lobatto IIIA-B-C-C*-D coefficients, respectively,

[43,50,51]. Assumption (27a) is a *stiff accuracy* condition. Gauss RK coefficients do not satisfy this assumption [49]. To preserve in addition any reversibility property of the flow symmetric SPARK methods should be considered. Symmetric SPARK methods satisfying (27b) must necessarily satisfy (27a) when the weights b_i are nonzero and the nodes c_i are distinct [40, Theorem II.8.8]. Assumption (27d) is a condition to ensure existence and uniqueness of the numerical solution [50,51]. In a $o(1)$ -neighborhood of y_0 and z_0 the solution of (26) does not depend on z_0 . The value z_0 only determines to which branch $z = z(t, y)$ of $g_t(t, y) + g_y(t, y)f(t, y, z) = 0$ the solution is close. An accurate value z_1 is therefore not required since the values z_n do not influence the global convergence properties of the differential variable y . There are several ways to define the numerical solution z_1 for the algebraic variable, such as $z_1 := Z_s$ or z_1 being the solution of $g_t(t_1, y_1) + g_y(t_1, y_1)f(t_1, y_1, z_1) = 0$. For the orthogonal integration (24) the latter corresponds to taking $\Lambda_1 := 0$ and is a natural choice. We will not discuss in details local error estimates which can be found in [51]. For s -stage Lobatto SPARK methods superconvergence of order $2s - 2$ is attained [51]. Notice that the above combination of RK methods does not include the PRK methods of Murua [52] where the constraints $0 = g(t, y)$ are treated differently. Our main interest is in combining the Lobatto IIIA coefficients for A_1 with symmetric RK coefficients A_m also based on Lobatto points, such as Lobatto IIIA, IIIB, and IIID [43]. Our purpose here is not in convergence analysis, see [51], but in presenting an efficient implementation of SPARK methods for orthogonal integration.

The equations (26) can be rewritten equivalently

$$0 = Y_i - y_0 - h \sum_{j=1}^s \sum_{m=1}^M a_{ij,m} f_m(T_j, Y_j, Z_j), \quad \text{for } i = 1, \dots, s, \quad (28a)$$

$$0 = \sum_{j=1}^s a_{ij,1} g \left(T_j, y_0 + h \sum_{k=1}^s \sum_{m=1}^M a_{jk,m} f_m(T_k, Y_k, Z_k) \right), \quad \text{for } i = 2, \dots, s, \quad (28b)$$

$$0 = y_1 - y_0 - h \sum_{j=1}^s b_j f(T_j, Y_j, Z_j), \quad (28c)$$

$$0 = g \left(t_1, y_0 + h \sum_{i=1}^s b_i f(T_i, Y_i, Z_i) \right). \quad (28d)$$

We introduce the notations $Y_{s+1} := y_1$ and

$$\alpha_m := \begin{pmatrix} A_m \\ b^\top \end{pmatrix}, \quad \text{for } m = 1, \dots, M.$$

An approximate Jacobian matrix of this nonlinear system (28) with respect to $(Y_1, \dots, Y_s, Y_{s+1}, Z_1, \dots, Z_s)$ at $Y_i = y_0, Z_i = z_0$ is given by

$$\begin{pmatrix} I_{s+1} \otimes I_m & -h \sum_{m=2}^M \alpha_m \otimes f_{mz}(t_0, y_0, z_0) \\ 0 & h \begin{pmatrix} N \\ b^\top \end{pmatrix} \otimes (g_y(t_0, y_0) f_z(t_0, y_0, z_0)) \end{pmatrix},$$

where we have neglected the terms involving $f_{my}(y_0, z_0)$ as in [48] by implicitly assuming the system of DAEs (25) to be nonstiff. When system (25) is of the form (23) and such that $g_y(t_0, y_0)g_y^\top(t_0, y_0)$ can be approximated by a matrix d , as in (20) in Lemma 4, we obtain the following approximate inverse Jacobian:

$$\begin{pmatrix} I_s \otimes I_m & \sum_{m=2}^M \alpha_m \Omega \otimes g_y^\top(t_0, y_0) d^{-1} \\ 0 & -\frac{1}{h} \cdot \Omega \otimes d^{-1} \end{pmatrix}.$$

For system (24) the application of modified Newton iterations to the system of nonlinear equations (28) is given by the following algorithm.

ALGORITHM 3. *Modified Newton method for the SPARK equations (28) applied to the DAEs (24).*

0. $k := 0$;
1. $\Lambda_i^0 := 0$ for $i = 1, \dots, s$;
2. Predict Y_i , e.g., $Y_i^0 := Y_0$ for $i = 1, \dots, s+1$;
3. Do while stopping criterion not satisfied
 4. $f_{i,m}^k := (F(t_0 + c_i h, Y_i^k) - Y_i^k \Lambda_i^k)_m$ for $m = 1, \dots, M, i = 1, \dots, s$;
 5. $Y_i^k := Y_0 + h \sum_{j=1}^s \sum_{m=1}^M \alpha_{ij,m} f_{j,m}^k$ for $i = 1, \dots, s+1$;
 6. $G_i^k := Y_i^{k\top} Y_i^k - I$ for $i = 1, \dots, s+1$;
 7. $R_i^k := \sum_{j=1}^s \alpha_{i+1,j,1} G_j^k$ for $i = 1, \dots, s-1, \quad R_s^k := G_{s+1}^k$;
 8. $\Delta \Lambda_i^k := \frac{1}{2h} \sum_{j=1}^s \Omega_{ij} R_j^k$ for $i = 1, \dots, s$;
 9. $\Lambda_i^{k+1} := \Lambda_i^k + \Delta \Lambda_i^k$ for $i = 1, \dots, s$;
 10. $Y_i^{k+1} := Y_i^k - h \sum_{j=1}^s \alpha_{ij, m_{Y\Lambda}} Y_0 \Delta \Lambda_j^k$ for $i = 1, \dots, s+1$;
 11. $k := k+1$;
12. Repeat from 3.

It is natural to replace in Step 10 of this algorithm the matrix products $Y_0 \Delta \Lambda_j^k$ for $j = 1, \dots, s$ by $Y_j^k \Delta \Lambda_j^k$. This leads to a markedly faster algorithm. The above iterations involve no matrix factorization, they require only matrix-matrix products and are thus parallelizable. Provided the stepsize h is not too large, these modified Newton iterations converge. They converge linearly and at each iteration one power of h is gained, the analysis being similar to that given in [48, Chapter 7].

6.3. The Free Rigid Body

As a numerical experiment we consider the equations of motion for the free rigid body. These include the Euler equations, see (9), for the body angular momenta $\Pi := (\Pi_1, \Pi_2, \Pi_3)^\top$

$$\Pi_1' = \frac{I_2 - I_3}{I_2 I_3} \Pi_2 \Pi_3, \quad \Pi_2' = \frac{I_3 - I_1}{I_3 I_1} \Pi_3 \Pi_1, \quad \Pi_3' = \frac{I_1 - I_2}{I_1 I_2} \Pi_1 \Pi_2, \quad (29a)$$

and the equations for the orientation matrix Y of the body

$$Y' = Y \hat{\Omega}, \quad (29b)$$

where $\Omega := \mathbf{I}^{-1} \Pi$ is the body angular velocity, $\mathbf{I} = \text{diag}(I_1, I_2, I_3)$ is the inertia matrix, and

$$\hat{\Omega} := \begin{pmatrix} 0 & -\Omega_3 & \Omega_2 \\ \Omega_3 & 0 & -\Omega_1 \\ -\Omega_2 & \Omega_1 & 0 \end{pmatrix}.$$

The matrix Y is a rotation matrix. Since the equations (29b) are of the form (18), Y remains orthogonal as it should for a rotation matrix. There are several ways of writing differential equations to obtain the orientation matrix using, e.g., by using Euler angles or quaternions, which would actually reduce the number of equations. We do not advocate here to use or not the above formulation to solve the rigid body equations. One goal here is only to illustrate with a simple example the fact that orthonormality can be preserved and at the same time the good long-term properties of the numerical scheme can be maintained. It is well known that the spatial angular momenta $\pi := Y \Pi$ are preserved, i.e., they satisfy $\pi(t) = \pi(t_0)$. We will use these quantities to monitor the quality of the integrator. We integrate the system (29a) exactly as

described at the end of Section 4.1 using the two-stage Lobatto IIIA-B method with a constant step-size $h = 0.1$ and with the same initial values. We couple this system of ODEs (29a) with the system of DAEs corresponding to (24)–(29b)

$$Y' = Y\hat{\Omega} - Y\Lambda, \quad 0 = Y^\top Y - I,$$

where Λ is a symmetric matrix. We apply for illustration purpose the two-stage Lobatto IIID coefficients to the term $Y\hat{\Omega}$, the two-stage Lobatto IIIB coefficients to the term $-Y\Lambda$, and the two-stage Lobatto IIIA coefficients to the constraints $0 = Y^\top Y - I$. This demonstrates the flexibility of SPARK methods. We plot in Figure 2 the error for the spatial angular momenta $(\pi_1, \pi_2, \pi_3)^\top$. As is expected for a symmetric integrator we observe that the errors for these invariants remain bounded. We also plot the quantity $\|Y^\top Y - I\|_\infty$ to show that orthogonality is preserved up to close to machine precision.

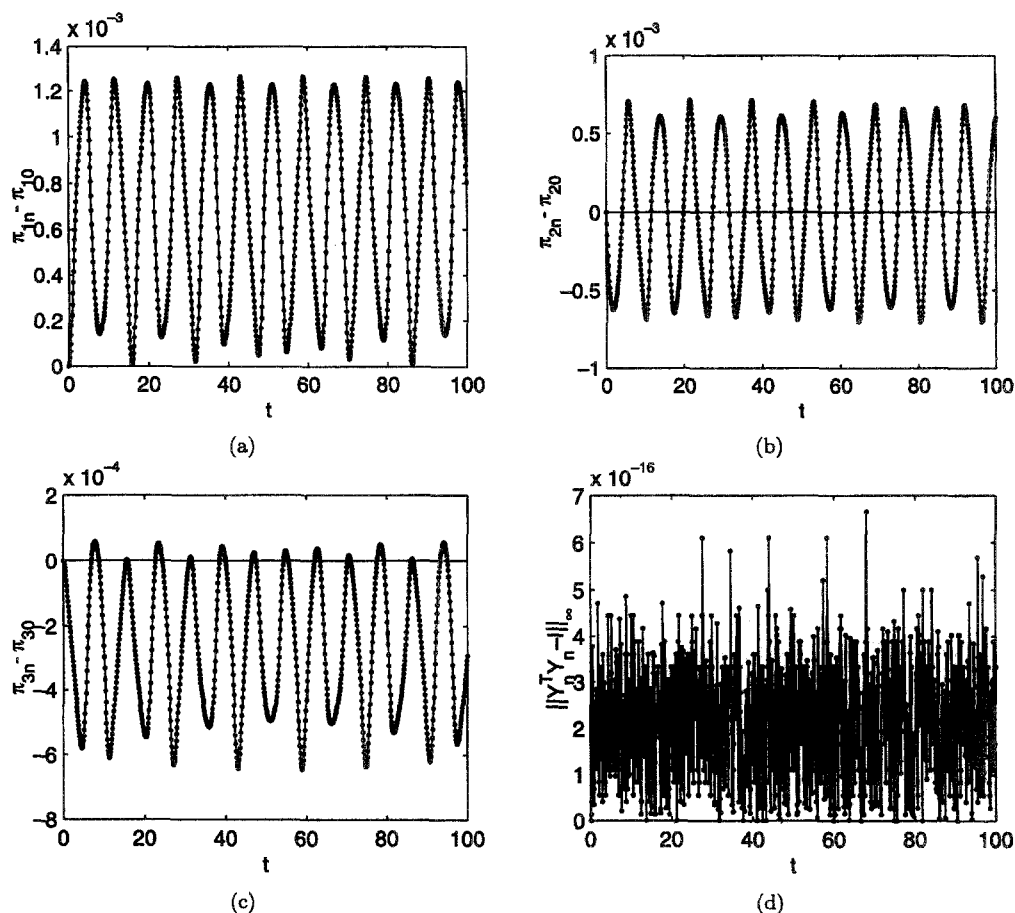


Figure 2. Errors of conserved quantities for the free rigid body equations using a two-stage Lobatto IIIA-B-D SPARK method with step-size $h = 0.1$.

6.4. Toda Lattice Equations

As a last numerical experiment we consider the *Toda lattice equations*

$$u'_k = v_k, \quad v'_k = \phi'(u_{k+1} - u_k) - \phi'(u_k - u_{k-1}), \quad \text{for } k = 1, \dots, n, \quad (30)$$

with periodic conditions $u_0 := u_n, u_{n+1} := u_1, v_0 := v_n, v_{n+1} := v_1$. This is a standard Hamiltonian system with Hamiltonian

$$H(u, v) = \sum_{k=1}^n \left(\frac{v_k^2}{2} + \phi(u_k - u_{k-1}) \right). \quad (31)$$

Letting $\phi(r) := e^{-r}$ and considering the nonlinear change of variables

$$\alpha_k := \frac{1}{2}e^{(1/2)(u_k - u_{k+1})}, \quad \beta_k := \frac{1}{2}v_k, \quad \text{for } k = 0, \dots, n,$$

we can rewrite the corresponding system of ODEs as follows [53,54]

$$\alpha'_k = \alpha_k(\beta_k - \beta_{k+1}), \quad \beta'_k = 2(\alpha_{k-1}^2 - \alpha_k^2), \quad \text{for } k = 1, \dots, n.$$

The matrix

$$X := \begin{pmatrix} \beta_1 & \alpha_1 & 0 & \cdots & 0 & \alpha_n \\ \alpha_1 & \beta_2 & \alpha_2 & \ddots & 0 & 0 \\ 0 & \alpha_2 & \beta_3 & \alpha_3 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & \ddots & \alpha_{n-2} & \beta_{n-1} & \alpha_{n-1} \\ \alpha_n & 0 & \cdots & 0 & \alpha_{n-1} & \beta_n \end{pmatrix}$$

satisfies (19) for the skew-symmetric matrix

$$K(X) := \begin{pmatrix} 0 & -\alpha_1 & 0 & \cdots & 0 & \alpha_n \\ \alpha_1 & 0 & -\alpha_2 & \ddots & 0 & 0 \\ 0 & \alpha_2 & 0 & -\alpha_3 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & \ddots & \alpha_{n-2} & 0 & -\alpha_{n-1} \\ -\alpha_n & 0 & \cdots & 0 & \alpha_{n-1} & 0 \end{pmatrix}.$$

Hence, the flow of matrix X is isospectral and as described in Section 5, we can consider the orthogonal flow given by the system of matrix ODEs (17) with $J(Y) := K(YX_0Y^\top)$ and $Y_0 := I$. The matrix X can be recovered from the relation $X := YX_0Y^\top$. To preserve the eigenvalues of matrix X we can apply any orthogonal integrator to the orthogonal flow of Y . The system of DAEs corresponding to (24) reads

$$Y' = K(YX_0Y^\top)Y - Y\Lambda, \quad 0 = Y^\top Y - I.$$

We apply for illustration purpose the three-stage Lobatto IIIA coefficients to the term $K(YX_0Y^\top)Y$ and the constraints $0 = Y^\top Y - I$, and the three-stage Lobatto IIID coefficients to the term $-Y\Lambda$. As in [55] we have taken $n = 3$ and initial values corresponding to $u_1 = 1, u_2 = 2, u_3 = 4, v_1 = 0, v_2 = -1, v_3 = -0.5$ for (30). We plot in Figure 3 the error for the eigenvalues λ_i ($i = 1, 2, 3$) of X , and for the Hamiltonian (31) which satisfies $H = 2\|X\|_F^2$ where $\|\cdot\|_F$ is the Frobenius norm. Since $X_n = Y_n X_0 Y_n^\top$ and Y_n is orthogonal we have $H_n = 2\|X_n\|_F^2 = 2\|X_0\|_F^2$. We observe that all these quantities are preserved up to close to machine precision.

7. CONCLUSION

The numerical integration of systems of ODEs with Poisson structure has first been considered. We have proven that symplectic PRK methods preserve the Poisson structure when the Poisson tensor is constant. We have shown on the Euler equations for the rigid body that a nonconstant Poisson structure can still be preserved by symplectic PRK methods when considering nonlinear changes of coordinates. The numerical integration of orthogonal flows and of the closely related isospectral flows have also been considered. To preserve the orthogonality property an approach has been proposed based on integrating an equivalent system of semiexplicit index 2 DAEs with SPARK methods. This approach requires only matrix-matrix products, and is therefore, parallelizable. Moreover, geometric properties of the flow can be preserved.

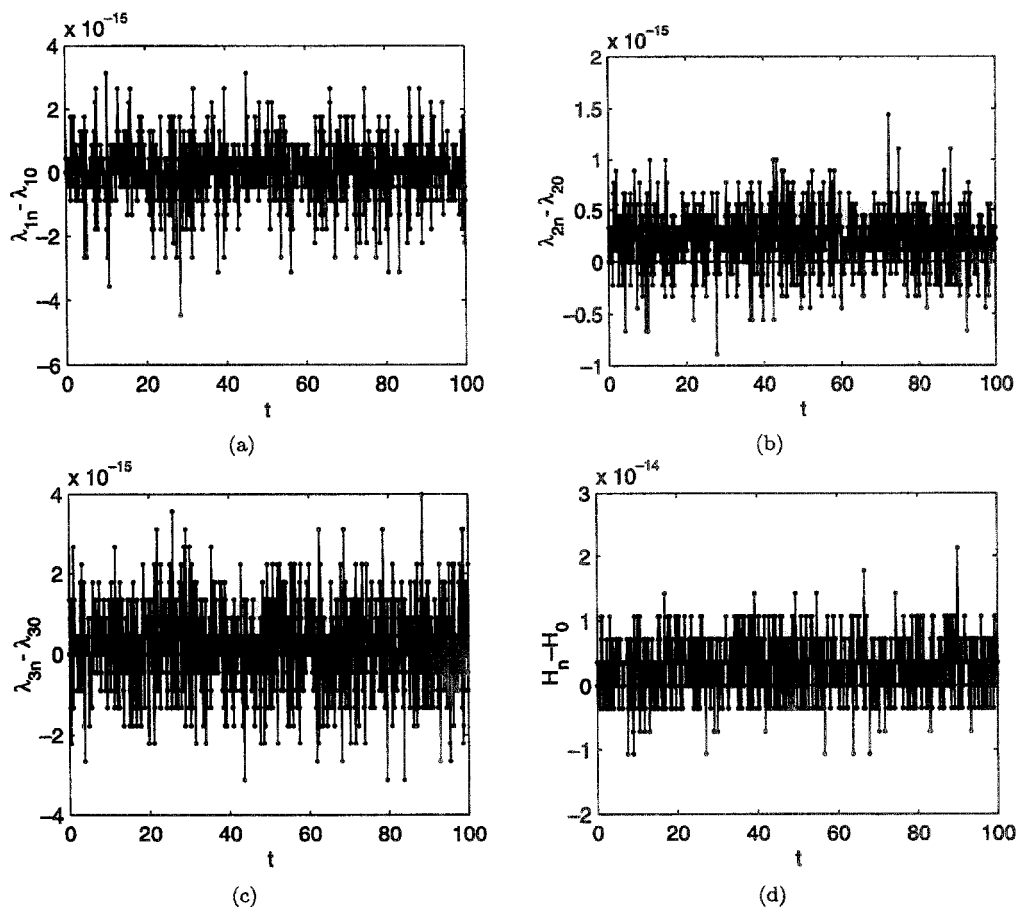


Figure 3. Errors of conserved quantities associated to the Toda lattice equations using a three-stage Lobatto IIIA-D SPARK method with step-size $h = 0.1$.

REFERENCES

1. Z. Ge and J.E. Marsden, Lie-Poisson Hamilton-Jacobi theory and Lie-Poisson integrators, *Phys. Lett. A* **133**, 134–139, (1988).
2. D. Lewis and J.C. Simo, Conserving algorithms for the dynamics of Hamiltonian systems on Lie groups, *Nonlin. Sci.* **4**, 253–299, (1994).
3. S. Li and M. Qin, A note for Lie-Poisson Hamilton-Jacobi equation and Lie-Poisson integrator, *Computers Math. Applic.* **30** (7), 67–74, (1995).
4. S. Reich, Symplectic integration of conservative multibody systems, *Fields Institute Communications* **10**, 181–191, (1996).
5. W. Zhu and M. Qin, Poisson schemes for Hamiltonian systems on Poisson manifolds, *Computers Math. Applic.* **27** (12), 7–16, (1994).
6. M.A. Austin, P.S. Krishnaprasad and L.-S. Wang, Almost Poisson integration of rigid body systems, *J. Comput. Phys.* **107**, 105–117, (1993).
7. G. Benettin, A.M. Cherubini and F. Fassò, A ‘changing chart’ symplectic algorithm for rigid bodies and other Hamiltonian systems on manifolds, *SIAM J. Sci. Stat. Comput.* **23**, 1189–1203, (2001).
8. D. Lewis and J.C. Simo, Conserving algorithms for the N dimensional rigid body, *Fields Inst. Comm.* **10**, 121–139, (1996).
9. S. Li and M. Qin, Lie-Poisson integration for rigid body dynamics, *Computers Math. Applic.* **30** (9), 105–118, (1995).
10. R.I. McLachlan, Explicit Lie-Poisson integration and the Euler equations, *Phys. Rev Lett.* **71**, 3043–3046, (1993).
11. T.J. Bridges and S. Reich, Computing Lyapunov exponents on a Stiefel manifold, *Physica D* **156**, 219–238, (2001).
12. M. Calvo, M.P. Laburta, J.I. Montijano and L. Rández, On the numerical integration of orthogonal flows with Runge-Kutta methods, *J. Comput. Appl. Math.* **115**, 121–135, (2000).
13. M.P. Calvo, A. Iserles and A. Zanna, Numerical solution of isospectral flows, *Math. of Comp.* **66**, 1461–1486, (1997).

14. E. Celledoni and B. Owren, A class of intrinsic schemes for orthogonal integration, *SIAM J. Numer. Anal.* **40** (6), 2069–2084, (2002).
15. E. Celledoni and B. Owren, Lie group methods for rigid body dynamics and time integration on manifolds, *Comput. Meth. in Appl. Mech. and Engin.* **192** (3), 421–438, (2003).
16. L. Dieci, R.D. Russell and E.S. van Vleck, Unitary integrators and applications to continuous orthonormalization techniques, *SIAM J. Numer. Anal.* **31**, 261–281, (1994).
17. L. Dieci, R.D. Russell and E.S. van Vleck, On the computation of a Lyapunov exponents for continuous dynamical systems, *SIAM J. Numer. Anal.* **34**, 402–423, (1997).
18. L. Dieci and E.S. van Vleck, Computation of a few Lyapunov exponents for continuous and discrete dynamical systems, *Appl. Numer. Math.* **17**, 275–291, (1995).
19. L. Dieci and E.S. van Vleck, Computation of orthonormal factors for fundamental solution matrices, *Numer. Math.* **83**, 599–620, (1999).
20. K. Engø and S. Faltinsen, Numerical integration of Lie-Poisson systems while preserving coadjoint orbits and energy, *SIAM J. Numer. Anal.* **39** (1), 128–145, (2001).
21. K. Engø and A. Marthinsen, Application of geometric integration to some mechanical problems (132), Department of Informatics, University of Bergen, (April 1997).
22. D.J. Higham, Runge-Kutta type methods for orthogonal integration, *Appl. Numer. Math.* **22**, 217–223, (1996).
23. D.J. Higham, Time-stepping and preserving orthonormality, *BIT* **37**, 24–36, (1997).
24. A. Iserles and A. Zanna, Qualitative numerical analysis of ordinary differential equations, In *Lectures in Applied Mathematics, Volume 32*, (Edited by J. Renegar, M. Shub and S. Smale), pp. 421–442, AMS, Providence, RI, (1996).
25. B.J. Leimkuhler and E.S. van Vleck, Orthosymplectic integration of linear Hamiltonian systems, *Numer. Math.* **77**, 269–282, (1997).
26. A. Marthinsen, H. Munthe-Kaas and B. Owren, Simulation of ordinary differential equations on manifolds—Some numerical experiments and verifications, *Modeling, Identification and Control* **18**, 75–88, (1997).
27. B. Owren and A. Marthinsen, Runge-Kutta methods adapted to manifolds and based on rigid frames, *BIT* **39** (1), 116–142, (1999).
28. A. Zanna, Lie-group methods for isospectral flows (DAMTP 1997/NA02), Univ. of Cambridge, (1997).
29. J.E. Marsden and T.S. Ratiu, Introduction to mechanics and symmetry, In *Texts in Applied Math., Volume 17*, Springer, New York, (1994).
30. F.M. Lasagni, Canonical Runge-Kutta methods, *ZAMP* **39**, 952–953, (1988).
31. J.M. Sanz-Serna, Runge-Kutta schemes for Hamiltonian systems, *BIT* **28**, 877–883, (1988).
32. Y.B. Suris, The canonicity of mappings generated by Runge-Kutta type methods when integrating the systems $\dot{x} = -\frac{\partial U}{\partial x}$, *U.S.S.R. Comput. Maths. Math. Phys.* **29**, 138–144, (1989).
33. W. Zhu and M. Qin, Reply to “Comments on ‘Poisson schemes for Hamiltonian systems on Poisson manifolds’”, *Computers Math. Applic.* **29** (7), 1, (1995).
34. R.I. McLachlan, Comments on “Poisson schemes for Hamiltonian systems on Poisson manifolds”, *Computers Math. Applic.* **29** (3), 1, (1995).
35. H.-J. Kowalsky, *Lineare Algebra*, W. de Gruyter, Berlin, (1972).
36. R. Abraham and J.E. Marsden, *Foundations of Mechanics*, Second Edition, Addison-Wesley, (1978).
37. J.M. Sanz-Serna and M.P. Calvo, Numerical Hamiltonian problems, In *Appl. Math. and Math. Comput., Volume 7*, Chapman & Hall, London, (1994).
38. G. Darboux, Sur le problème de Pfaff, *Extrait Bulletin des Sciences Math. et Astron., 2e Série VI*, Gauthier-Villars, Paris, (1882).
39. D.D. Holm and J.E. Marsden, The rotor and the pendulum, In *Symplectic Geometry and Mathematical Physics, Progress in Math.*, (Edited by P. Donato, C. Duval, J. Elhadad and G.M. Tuynman), pp. 189–203, vol. 99, Birkhäuser, Boston, (1991).
40. E. Hairer, S.P. Nørsett and G. Wanner, Solving ordinary differential equations I. Nonstiff problems, In *Comput. Math., Volume 18*, Second Revised Edition, Springer, Berlin, (1993).
41. L.O. Jay, Runge-Kutta type methods for index three differential-algebraic equations with applications to Hamiltonian systems, Ph.D. Thesis, Department of Mathematics, University of Geneva, Switzerland, (1994).
42. L.O. Jay, Symplectic partitioned Runge-Kutta methods for constrained Hamiltonian systems, *SIAM J. Numer. Anal.* **33**, 368–387, (1996).
43. L.O. Jay, Structure preservation for constrained dynamics with super partitioned additive Runge-Kutta methods, *SIAM J. Sci. Comput.* **20**, 416–446, (1998).
44. L. Verlet, Computer experiments on classical fluids. I. Thermodynamical properties of Lennard-Jones molecules, *Phys. Rev.* **159**, 98–103, (1967).
45. H. Yoshida, Construction of higher order symplectic integrators, *Phys. Lett. A* **150**, 262–268, (1990).
46. B.J. Leimkuhler, Discretization and weak invariants, Univ. of Kansas, (1993).
47. K.E. Brenan, S.L. Campbell and L.R. Petzold, Numerical solution of initial-value problems in differential-algebraic equations, In *SIAM Classics in Appl. Math.*, Second Edition, SIAM, Philadelphia, PA, (1996).
48. E. Hairer, Ch. Lubich and M. Roche, The numerical solution of differential-algebraic systems by Runge-Kutta methods, In *Lect. Notes in Math., Volume 1409*, Springer, Berlin, (1989).
49. E. Hairer and G. Wanner, Solving ordinary differential equations II. Stiff and differential-algebraic problems, In *Comput. Math., Volume 14*, Second Revised Edition, Springer, Berlin, (1996).

50. L.O. Jay, Iterative solution of nonlinear equations for SPARK methods applied to DAEs, *Numer. Alg.* **31**, 171–191, (2002).
51. L.O. Jay, Solution of index 2 implicit differential-algebraic equations by Lobatto Runge-Kutta methods, *BIT* **43**, 91–104, (2003).
52. A. Murua, Partitioned Runge-Kutta methods for semi-explicit differential-algebraic systems of index 2 (EHU-KZAA-IKT-196), Univ. of the Basque Country, (1996).
53. P. Deift, T. Nanda and C. Tomei, Differential equations for the symmetric eigenvalue problem, *SIAM J. Numer. Anal.* **20**, 1–22, (1983).
54. H. Flaschka, The Toda lattice, *Phys Rev. B* **9**, 1924–1925, (1974).
55. E. Hairer, Symmetric projection methods for differential equations on manifolds, *BIT* **40**, 726–734, (2000).