Structural schemes for Hamiltonian systems

Stéphane Clain, Emmanuel Franck, Victor Michel-Dansac

STRUCTURAL SCHEMES FOR HAMILTONIAN SYSTEMS

ARTICLE

Stéphane Clain
Centre of Mathematics, Coimbra University, Largo D. Dini, 3000-143 Coimbra, Portugal
clain@mat.uc.pt

Emmanuel Franck
Université de Strasbourg, CNRS, Inria, IRMA, F-67000, Strasbourg, France
emmanuel.franck@inria.fr

Victor Michel-Dansac
Université de Strasbourg, CNRS, Inria, IRMA, F-67000, Strasbourg, France
victor.michel-dansac@inria.fr

January 24, 2025

ABSTRACT

We present an adaptation of the so-called structural method Clain et al. [2023a] for Hamiltonian systems, and redesign the method for this specific context, which involves two coupled differential systems. Structural schemes decompose the problem into two sets of equations: the physical equations, which describe the local dynamics of the system, and the structural equations, which only involve the discretization on a very compact stencil. They have desirable properties, such as unconditional stability or high-order accuracy. We first give a general description of the scheme for the scalar case (which corresponds to e.g. spring-mass interactions or pendulum motion), before extending the technique to the vector case (treating e.g. the n-body system). The scheme is also written in the case of a non-separable system (e.g. a charged particle in an electromagnetic field). We give numerical evidence of the method's efficiency, its capacity to preserve invariant quantities such as the total energy, and draw comparisons with the traditional symplectic methods.

Keywords Structural method · Compact schemes · Hamiltonian systems

1 Introduction

Hamiltonian systems, a class of ordinary differential equations (ODEs) that arise from Hamiltonian mechanics, play a fundamental role in the mathematical modeling of physical systems with conserved quantities, such as total energy for a closed system. The dynamics of such systems are described by the principle of stationary action, leading to the Euler-Lagrange equations and the associated Lagrangian function Leimkuhler and Reich [2005]. Derived from the Hamiltonian function, Hamiltonian systems govern the time evolution of state variables such as positions and momenta. Hamiltonian systems are ubiquitous in various fields, including celestial mechanics, where they describe planetary orbits, or plasma physics, capturing the behavior of charged particles. Their inherent structure, characterized by symplectic geometry and energy preservation, has to be taken into account in their numerical approximation.

Due to their rich structure, the numerical approximation of Hamiltonian systems presents several challenges. One of the main issues is the preservation of geometric features, such as symplecticity and conserved quantities like energy or angular momentum, which are integral to the physical behavior of the system. Standard numerical methods, even of very high order, such as explicit or implicit Runge-Kutta schemes, often fail to maintain these properties during simulations, generating error accumulations in the long run that can become critical and totally spoil the quality of the solution Beust [2003]. Additionally, the typically oscillatory nature of the solutions to such systems, particularly in applications

like plasma physics, require specific methods enable of handling multiple time scales without compromising accuracy or stability. On the other hand, the high-dimensional phase spaces involved in many problems further exacerbate computational costs, requiring efficient algorithms that balance precision with practicality.

These challenges have motivated the development of structure-preserving methods, such as symplectic integrators and geometric numerical schemes Feng and Qin [1987], that ensure robust long-term stability by providing a good approximation of the total energy or other invariants. The design of schemes that preserve geometric structures for ODEs provide accurate and stable numerical methods. Some classes of partial differential equations, after well-adapted spatial discretizations, are written as high-dimensional Hamiltonian ODEs. As examples, we mention the works of Kraus et al. [2017], Li et al. [2024] in plasma physics or Shepherd [1990] in climatology where these approaches, using geometric structure-preserving time discretizations, have shown their superiority.

Hamiltonian systems typically involve a system of two coupled ODEs, one describing the time evolution of a position in phase space, and the second one governing the time evolution of the momentum (or velocity). The simplest symplectic algorithm is a semi-implicit Euler integrator, where the first equation is integrated explicitly, and the second one implicitly. Therefore, it requires solving a (potentially nonlinear) implicit equation at each time step. However, this integrator is only first-order accurate. In fact, one of the most well-known symplectic algorithms for Hamiltonian systems is the Strömer-Verlet method, see Verlet [1967], Hairer et al. [2003], which can be seen as a second-order accurate version of the semi-implicit Euler scheme. From this second-order algorithm, it is relatively easy to construct high-order methods thanks to the composition strategy, introduced in Suzuki [1990], Yoshida [1990], McLachlan [1995] and summarized in Hairer et al. [2006]. Nevertheless, one drawback of this approach is that the number of function evaluations increases exponentially with the order of accuracy. Other examples of high-order schemes are provided in McLachlan and Atela [1992], Sanz-Serna and Calvo [1993], and other composition schemes include the ones from Kahan and Li [1997], which still stuffer from the same drawback in terms of number of function evaluations.

The structural method was recently introduced, in Clain et al. [2023a], to solve ODE systems. It provides a systematic way of constructing implicit schemes with a highly compact stencil, which have an arbitrarily high order of accuracy together with unconditional stability. The technique relies on separating the physical part of the problem, i.e., the system of ODEs, from the discretization, which only involves the unknown quantities associated with the grid. In practice, the data is gathered into blocks, corresponding to several time steps to be solved simultaneously. This methodology was also adapted to one-dimensional boundary-valued problems in Clain et al. [2023b]. In this case, the structural method achieves a sixth-order accurate approximations using a tiny stencil of three points.

In this paper, we develop and adapt the structural method to Hamiltonian systems, taking into account the specificities of such systems, namely the conservation of several invariants, and the two-variable nature of the problem. The paper is organized as follows. First, we present the structural method for ordinary differential equations in Section 2, introducing two formulations, the first using only the ODE and the second adding information related to the time derivative of the ODE. Then, the structural method is applied to Hamiltonian scalar problems in Section 3 and to general Hamiltonian systems in Section 4, thus constructing high-order accurate and stable schemes for Hamiltonian systems. Finally, we present numerical results in Section 5 to demonstrate the efficiency and accuracy of the proposed schemes. Several Hamiltonian systems are considered, from simple separable examples to more complex non-separable ones, to illustrate the performance of the structural method in various contexts.

2 The structural method for ODEs

We provide a short review of the structural method based on the so-called Physical and Structural Equations (denoted PE and SE respectively) detailed in Clain et al. [2023a]. To this end, let us consider the Ordinary Differential Equation (ODE) for $t \in [0, T]$

$$\dot{x} = f(x), \qquad x(0) = x_0. \tag{1}$$

Traditional schemes blend the discretization with the physical equation, that is, the function that describe the dynamic of the physical system. For example, the popular Crank-Nicholson scheme reads

$$\frac{x_{n+1} - x_n}{\Delta t} = \frac{f(x_n) + f(x_{n+1})}{2}$$

where the left-hand side is the time derivative discretization, while the right-hand side represents the physics (i.e., the function f that characterizes the physical problem). Obviously, one can split the scheme into two equations, namely

$$D_{n+1} = f(Z_{n+1}) \qquad \text{and} \qquad \frac{Z_{n+1} - Z_n}{\Delta t} - \frac{D_n + D_{n+1}}{2} = 0,$$

where Z_n and D_n are approximations of the Zeroth-order derivative $x(t_n)$ and the first-order Derivative $\dot{x}(t_n)$, applying the same notations for Z_{n+1} and D_{n+1} at the time t_{n+1}. The left relation is called the Physical Equation since it only

involves the physics of the problem and not the discretization; the right one is called the Structural Equation since it only depends on the grid structure, and involves no physics.

An extension relies on taking the time derivative of the ODE, which reads $\ddot{x} = f'(x)\dot{x}$. Denoting by S_n an approximation of the Second derivative $\ddot{x}(t_n)$, we now obtain two physical equations, respectively denoted by PE[1] and PE[2]:

$$D_{n+1} = f(Z_{n+1}) \qquad \text{and} \qquad S_{n+1} = f'(Z_{n+1})D_{n+1}.$$

Since we have three unknowns $(Z_{n+1}, D_{n+1}, S_{n+1})$, we need one additional equation that represents the discretization. We choose the following relation, exact for polynomial functions up to degree 4, and called the structural equation SE[1]:

$$12\frac{Z_{n+1} - Z_n}{\Delta t^2} - 6\frac{D_{n+1} + D_n}{\Delta t} + (S_{n+1} - S_n) = 0.$$

Given the values $(Z_{n+1}, D_{n+1}, S_{n+1})$ at a time t_n, we seek solutions at t_{n+1} of PE[1], PE[2], and SE[1]. Note that the problem is fully implicit and requires solving a small nonlinear system as soon as f is nonlinear.

Following the example, the idea of structural method consists in splitting the problem with, on the one hand, the Physical Equations and, and the other hand, the Structural Equations. The two sets of equations involve the unknowns over a block of size R steps corresponding to the time step t_{n+1} until t_{n+R}, given the initial configuration at the time t_n. We obtain a nonlinear system combining the function approximations together with the derivatives for all the intermediate steps.

We now detail the structural method for the scalar ODE (1) in the ZD and ZDS formulations, respectively involving the zeroth- and first-order derivatives in Section 2.1, and the zeroth-, first- and second-order derivatives in Section 2.2. The complete algorithm is then given, for each formulation, in Section 2.3.

2.1 ZD equations

We first tackle the simple version called the ZD scheme, where one only uses implicit combinations of the approximate function and first derivative as unknowns. The generic ZD structural equation then reads

$$\sum_{r=0}^{R} a_{r,0} Z_{n+r} + a_{r,1} D_{n+r} = 0, \tag{2}$$

where $(a_{r,s})_{r\in\{0,\dots,R\},s\in\{0,1\}}$ are the coefficients of the SE, independent of n by assuming a uniform time discretization with time step Δt. It is important to remark that $r = 0$ corresponds to the time t_n where all the variables are given. In total, $2(R + 1)$ coefficients characterize the structural equation. They can be reshaped as a vector

$$\boldsymbol{a} = [a_{0,0}, \quad a_{1,0}, \quad \dots, \quad a_{R,0}, \quad a_{0,1}, \quad a_{1,1}, \quad \dots, \quad a_{R,1}].$$

On the other hand, the implicit problem involves $2R$ unknowns with R physical equations $\mathrm{PE}[1], \dots, \mathrm{PE}[R]$ corresponding to the relations

$$D_{n+r} = f(Z_{n+r}), \qquad \text{for all } r \in \{1, \dots, R\}.$$

Hence, the structural method requires R linearly independent structural equations, i.e. R vectors, $(\boldsymbol{a}^m)_{m\in\{1,\dots,R\}}$ whose components are denoted by $(a^m_{r,s})_{r,s,m}$, to close the system of size $2R$.

We wish to provide this set of R structural equations by ensuring that the resulting scheme is high-order accurate. To this end, for any vector $\boldsymbol{a}$ and function ϕ, we define the functional

$$E(\boldsymbol{a}, \phi) = \sum_{r=0}^{R} a_{r,0}\phi(t_r) + a_{r,1}\phi'(t_r). \tag{3}$$

Taking the particular case of polynomial functions $\pi_\ell(t) = t^{\ell-1}$ for $\ell \in \{1, \dots, 2(R + 1)\}$, we consider the linear system $E(\boldsymbol{a}, \pi_\ell) = 0$, which rewrites as the $2(R + 1)$ equations

$$\forall \ell \in \{1, \dots, 2(R + 1)\}, \quad E(a; \pi_\ell) = \sum_{r=0}^{R} a_{r,0}\pi_\ell(r\Delta t) + a_{r,1}\pi'_\ell(r\Delta t) = 0.$$

We obtain a $2(R + 1) \times 2(R + 1)$ non-singular linear system given by $M\boldsymbol{a} = 0$, where the matrix M gathers all the coefficients $\pi_\ell(r\Delta t)$ and $\pi'_\ell(r\Delta t)$, for all $\ell \in \{1, \dots, 2(R + 1)\}$ and $r \in \{0, \dots, R\}$.

To determine the R structural equations, we simply withdraw the R last lines of the matrix M, leading to a reduced matrix whose kernel has dimension R. To build the structural equations, we select an orthogonal basis of this kernel and place it in the vector $(\boldsymbol{a}^m)_{m\in\{1,\dots,R\}}$. Thanks to this procedure, the obtained relations

$$\sum_{r=0}^{R} a_{r,0}^m Z_{n+r} + a_{r,1}^m D_{n+r} = 0, \quad m = 1, \cdots, R, \tag{4}$$

remain exact for polynomials up to degree $R+1$.

Remark: For small values of R (say $R=2$), we derive an explicit analytic expression for the structural equation. However, this becomes intractable for larger $R>4$, and so we compute the kernel and determine an orthogonal basis to automatically provide the R structural equations (see Clain et al. [2023a] for details).

At the end of the day, we get a set of R structural equations $\texttt{SE}[1], \dots, \texttt{SE}[R]$, linearly independent, exact for polynomials up to degree $R+1$.

2.2 ZDS equations

The ZD scheme is quite effective and provides accurate solutions, but one can upgrade the method by introducing the second derivatives. Indeed, differentiating the original physical equation provides a second physical equation. Therefore, the scheme is more compact and accurate for the same block size. More precisely, the ZDS scheme involves the zeroth-, first- and second-order derivatives as unknowns, with the structural equation reading

$$\sum_{r=0}^{R} a_{r,0} Z_{n+r} + a_{r,1} D_{n+r} + a_{r,2} S_{n+r} = 0, \tag{5}$$

with $\boldsymbol{a} = (a_{r,s})_{r,s}$ its coefficients, independent of the time index n, the block number r and the derivation order s.

The ZDS method involves the $3R$ unknowns $(Z_{n+r}, D_{n+r}, S_{n+r})$ with $r \in \{1, \dots, R\}$. The physical equations provide $2R$ relations between these unknowns:

$$\forall r \in \{1, \dots, R\}, \qquad D_{n+r} = f(Z_{n+r}) \quad \text{and} \quad S_{n+r} = f'(Z_{n+r}) D_{n+r}.$$

To close the system, we need R linearly independent structural equations of type (5) with respective coefficients $(\boldsymbol{a}^m)_{m\in\{1,\dots,R\}}$, whose entries are $a_{r,d}^m$ with $r \in \{0, \dots, R\}$ and $d \in \{0, 1, 2\}$.

To this end, we introduce the functional

$$E(\boldsymbol{a}, \phi) = \sum_{r=0}^{R} a_{r,0}\phi(r\Delta t) + a_{r,1}\phi'(r\Delta t) + a_{r,2}\phi''(r\Delta t)$$

and recall the set of polynomial functions $\pi_\ell = t^{\ell-1}$. Then the $3(R+1)(R+1)$ linear system $E(\boldsymbol{a}, \pi_\ell) = 0$ is non-singular and reads $M\boldsymbol{a} = 0$ where M is an invertible matrix.

Eliminating the R last rows of the matrix M, we get a kernel of dimension R with an orthogonal basis $(\boldsymbol{a}^m)_{m\in\{1,\dots,R\}}$ that provides the R structural equation $\texttt{PE}[1], \dots, \texttt{PE}[R]$. Note that the relations

$$\sum_{r=0}^{R} a_{r,0}^m Z_{n+r} + a_{r,1}^m D_{n+r} + a_{r,2}^m S_{n+r} = 0 \qquad \text{for } m \in \{1, \dots, R\} \tag{6}$$

are exact for all polynomials of degree lower than $2(R+1)$. The solution of the $2R$ physical equations together with the R structural equations provide an approximation of the $3R$ variables.

2.3 Full algorithm for a scalar ODE

Equipped with the physical and structural equations in the ZD and ZDS formulations, we now detail the full algorithm to compute a full R-block of solutions with each of these formulations.

2.3.1 ZD formulation

Assume that we know the solution (Z_n, D_n) at the time t_n. We seek the solution of the nonlinear problem with $2R$ unknowns

$$\forall m \in \{1, \dots, R\}, \qquad D_{n+m} = f(Z_{n+m}) \quad \text{and} \quad \sum_{r=0}^{R} a_{r,0}^m Z_{n+r} + a_{r,1}^m D_{n+r} = 0. \tag{7}$$

Let $\mathbb{b}\boldsymbol{Z}_n = (Z_{n+r})_{r\in\{1,\dots,R\}} \in \mathbb{R}^R$ and $\mathbb{b}\boldsymbol{D}_n = (D_{n+r})_{r\in\{1,\dots,R\}} \in \mathbb{R}^R$ be the R-blocks for the approximations of x and $\dot{x}$ at time t_{n+r}, for each $r \in \{1,\dots,R\}$. We rewrite the structural equations under a matrix form

$$A_z\,\mathbb{b}\boldsymbol{Z}_n + A_d\,\mathbb{b}\boldsymbol{D}_n + A_s\,a_z + D_n\,a_d + S_n = 0,$$

with

$$\begin{aligned} A_z &= (a^m_{r,0})_{m,r}, \qquad a_z = \left(a^1_{0,0}, a^2_{0,0}, \cdots, a^R_{0,0}\right)^{\mathrm{t}},\\ A_d &= (a^m_{r,1})_{m,r}, \qquad a_d = \left(a^1_{0,1}, a^2_{0,1}, \cdots, a^R_{0,1}\right)^{\mathrm{t}}. \end{aligned}$$

Assuming the matrix $A_z \in \mathbb{R}^{R\times R}$ is non-singular, we rewrite the system as

$$\mathbb{b}\boldsymbol{Z}_n + B_d\,\mathbb{b}\boldsymbol{D}_n + Z_n\,b_z + D_n\,b_d = 0$$

with $B_d = (A_z)^{-1}A_d$, $b_z = (A_z)^{-1}a_z$, $b_d = (A_z)^{-1}a_d$.

To solve problem (7), we proceed with a fixed-point method by producing a sequence $(\mathbb{b}\boldsymbol{Z}_n^{(k)}, \mathbb{b}\boldsymbol{D}_n^{(k)})$ that converges to the solution. We sketch the algorithm hereafter.

- **Initialization.** To build $\mathbb{b}\boldsymbol{Z}_n^{(0)}$ and $\mathbb{b}\boldsymbol{D}_n^{(0)}$, we set for $r \in \{1,\dots,R\}$

  $$Z^{(0)}_{n+r} = Z^{(0)}_{n+r-1} + \Delta t D^{(0)}_{n+r-1}, \quad D^{(0)}_{n+r} = f(Z^{(0)}_{n+r}),$$

  with $Z^{(0)}_n = Z_n$, $D^{(0)}_n = D_n$.
- **Iteration.** We first compute a new approximation for the solution $\mathbb{b}\boldsymbol{Z}_n$ using the structural equations:

  $$\mathbb{b}\boldsymbol{Z}_n^{(k+1)} = -\Big(Z_n\,b_z + D_n\,b_d + B_d\,\mathbb{b}\boldsymbol{D}_n^{(k)}\Big)$$

  Then, we update the first derivative with the physical equations:

  $$\forall r \in \{1,\dots,R\}, \qquad D^{(k+1)}_{n+r} = f(Z^{(k+1)}_{n+r}).$$
- **Stopping criterion.** We end the fixed-point when two successive solutions are close enough, according to the tolerance parameter `tol`, that is $\|\mathbb{b}\boldsymbol{Z}_n^{(k+1)} - \mathbb{b}\boldsymbol{Z}_n^{(k)}\| \le \texttt{tol}$.

Remark: Notice that $R = 2$ provides a 4th-order unconditionally stable scheme while $R = 4$ reaches sixth-order accuracy. Methods with $R = 6$ and $R = 8$ will also be experimented in the numerical section. □

Remark: The nonlinear part of the problem does not require some local linearization, hence the computation of the derivative is straightforward. □

2.3.2 ZDS formulation

Assume that we knew the solution (Z_n, D_n, S_n) at the time t_n. We seek the solution of the following nonlinear problem with $3R$ unknowns: for all $m \in \{1,\dots,R\}$,

$$D_{n+m} = f(Z_{n+m}), \qquad S_{n+m} = f'(Z_{n+m})D_{n+m} \quad \text{and} \quad \sum_{r=0}^{R} a^m_{r,0}Z_{n+r} + a^m_{r,1}D_{n+r} + a^m_{r,2}S_{n+r} = 0. \tag{8}$$

Denoting by $\mathbb{b}\boldsymbol{S}_n$ the R-block of the second derivative approximations, we rewrite the structural equations under a matrix form

$$A_z\,\mathbb{b}\boldsymbol{Z}_n + A_d\,\mathbb{b}\boldsymbol{D}_n + A_s\,\mathbb{b}\boldsymbol{S}_n + Z_n\,a_z + D_n\,a_d + S_n\,a_s = 0,$$

with

$$\begin{aligned} A_z &= (a^m_{r,0})_{m,r}, \qquad a_z = \left(a^1_{0,0}, a^2_{0,0}, \cdots, a^R_{0,0}\right)^{\mathrm{t}},\\ A_d &= (a^m_{r,1})_{m,r}, \qquad a_d = \left(a^1_{0,1}, a^2_{0,1}, \cdots, a^R_{0,1}\right)^{\mathrm{t}},\\ A_s &= (a^m_{r,2})_{m,r}, \qquad a_s = \left(a^1_{0,2}, a^2_{0,2}, \cdots, a^R_{0,2}\right)^{\mathrm{t}}. \end{aligned}$$

Remark: For the sake of simplicity, we still use the notation A_z, A_d, a_z and a_d introduced in the ZD formulation for the sake of simplicity. Of course, the matrix entries are different from the ones corresponding to the ZD method.

Assuming the matrix $A_z \in \mathbb{R}^{R\times R}$ is non-singular, we rewrite the system as

$$\flat\boldsymbol{Z}_n + B_d\,\flat\boldsymbol{D}_n + B_s\,\flat\boldsymbol{S}_n + Z_n\,b_z + D_n\,b_d + S_n\,b_s = 0, \tag{9}$$

with $B_d = (A_z)^{-1}A_d$, $B_s = (A_z)^{-1}A_s$, $b_z = (A_z)^{-1}a_z$, $b_d = (A_z)^{-1}a_d$,$b_s = (A_z)^{-1}a_s$.

A fixed-point procedure is applied to solve the system (8), by producing a sequence $(\flat\boldsymbol{Z}_n^{(k)}, \flat\boldsymbol{D}_n^{(k)}, \flat\boldsymbol{S}_n^{(k)})$ that converges to the solution. We give the algorithm for the ZDS case.

- **Initialization.** To build $\flat\boldsymbol{Z}_n^{(0)}$, $\flat\boldsymbol{D}_n^{(0)}$, $\flat\boldsymbol{S}_n^{(0)}$, we set for $r \in \{1,\dots,R\}$

$$Z_{n+r}^{(0)} = Z_{n+r-1}^{(0)} + \Delta t D_{n+r-1}^{(0)} + \frac{\Delta t^2}{2} S_{n+r-1}^{(0)}, \quad D_{n+r}^{(0)} = f(Z_{n+r}^{(0)}), \quad S_{n+r}^{(0)} = f'(Z_{n+r}^{(0)})D_{n+r}^{(0)},$$

  with $Z_n^{(0)} = Z_n$, $D_n^{(0)} = D_n$, $S_n^{(0)} = S_n$.
- **Iteration.** We first compute a new approximation for the solution using the structural equations:

$$\flat\boldsymbol{Z}_n^{(k+1)} = -\Big(Z_n\,b_z + D_n\,b_d + S_n\,b_s + B_d\,\flat\boldsymbol{D}_n^{(k)} + B_s\,\flat\boldsymbol{S}_n^{(k)}\Big).$$

  Then, we update the first and second derivatives with the physical equations, by computing

$$\forall r \in \{1,\dots,R\}, \qquad D_{n+r}^{(k+1)} = f(Z_{n+r}^{(k+1)}) \quad \text{and} \quad S_{n+r}^{(k+1)} = f'(Z_{n+r}^{(k+1)})D_{n+r}^{(k+1)}.$$

- **Stopping criterion.** We end the fixed-point when two successive solutions are close enough according to the tolerance parameter tol, that is $\|\flat\boldsymbol{Z}_n^{(k+1)} - \flat\boldsymbol{Z}_n^{(k)}\| \le \texttt{tol}$.

Remark: It has been shown that $R = 1$ provides a 4th-order unconditionally stable scheme while $R = 2$ reaches sixth-order accuracy. Methods with $R = 3$ and $R = 4$ are also investigated in the numerical section. □

Remark: Given a grid, the structural equations' coefficients are computed independently of the problem, and their evaluations may be obtained in a pre-processing process or be stored with the grid points. □

3 Structural method for Hamiltonian problems: scalar equations

We reach the main novelty of the present work by adapting the structural method to Hamiltonian systems. The main difference is the introduction of a second variable p, that requires to handle both the approximations of $t \to x(t) \in \mathbb{R}$ and $t \to p(t) \in \mathbb{R}$, coupled through Hamilton's equations. This section is dedicated to scalar Hamiltonian equations, while systems will be considered in the next section. We first introduce the physical equations for the scalar Hamiltonian problem in Section 3.1, and then detail the structural equations and the algorithm to solve the coupled physical and structural equations in Section 3.2.

3.1 Physical equations

Consider a smooth function $\mathcal{H} : \mathbb{R}^2 \to \mathbb{R}$ that takes as input position x and momentum p. We seek solutions $x = x(t)$ and $p = p(t)$ such that $\mathcal{H}\,(x(t), p(t))$ is constant. Differentiating in time gives $\partial_x\mathcal{H}\,(x,p)\dot{x} + \partial_p\mathcal{H}\,(x,p)\dot{p} = 0$, and we define the trajectories as the solution of the ODE

$$\dot{x} = \partial_p\mathcal{H}\,(x,p) \quad \text{and} \quad \dot{p} = -\partial_x\mathcal{H}\,(x,p),$$

with the initial condition $x(0) = x_0$, $p(0) = p_0$. This ODE makes up the first physical equations PE[1]. We reformulate the problem within the Z, D framework by denoting by Zx and Dx the approximations of x and $\dot{x}$, and adopt similar notations for Zp and Dp. The physical equations then read

$$D\text{x} = \partial_p\mathcal{H}\,(Z\text{x}, Z\text{p}), \tag{10}$$

$$D\text{p} = -\partial_x\mathcal{H}\,(Z\text{x}, Z\text{p}). \tag{11}$$

For each time step, we have 2 physical equations, with 4 unknowns in total, and so we need 2 structural equations to close the system.

In the ZDS case, to provide the second physical system PE[2], we differentiate in time the first physical equations, and we obtain

$$\ddot{x} = \partial_{xp}\mathcal{H}\,(x,p)\dot{x} + \partial_{pp}\mathcal{H}\,(x,p)\dot{p} \qquad \text{and} \qquad \ddot{p} = -\partial_{xx}\mathcal{H}\,(x,p)\dot{x} - \partial_{xp}\mathcal{H}\,(x,p)\dot{p}.$$

Reformulating the problem within the Z, D, S framework yields the second physical equations PE[2]:

$$S\text{x} = \partial_{xp}\mathcal{H}\,(Z\text{x}, Z\text{p})D\text{x} + \partial_{pp}\mathcal{H}\,(Z\text{x}, Z\text{p})D\text{p}, \tag{12}$$

$$S\text{p} = -\partial_{xx}\mathcal{H}\,(Z\text{x}, Z\text{p})D\text{x} - \partial_{xp}\mathcal{H}\,(Z\text{x}, Z\text{p})D\text{p}. \tag{13}$$

Note that we have 4 physical equations, with 6 unknowns in total, and so 2 structural equations are required to close the system for each time step.

3.2 Structural equations and algorithms

Equipped with the physical and structural equations in the ZD and ZDS formulations, coupling x and p, we now give the algorithm to solve them in both cases.

3.2.1 ZD schemes

Approximations of the function x and the derivative $\dot{x}$ are connected via the structural equations (4), and similarly for the function p and its derivative. Hence, the structural equations read

$$\forall m \in \{1,\dots,R\}, \qquad \begin{cases} \displaystyle\sum_{r=0}^{R} a_{r,0}^m Z\mathrm{x}_{n+r} + a_{r,1}^m D\mathrm{x}_{n+r} = 0, \\ \displaystyle\sum_{r=0}^{R} a_{r,0}^m Z\mathrm{p}_{n+r} + a_{r,1}^m D\mathrm{p}_{n+r} = 0. \end{cases}$$

It is important to note that the x and p use the **same** structural equations (the same coefficients $a_{r,s}^m$) and **only differ** through the physical ones. Let denote by

$$\flat\boldsymbol{Z}\mathrm{x}_n = \left(Z\mathrm{x}_{n+1}, Z\mathrm{x}_{n+2}, \dots, Z\mathrm{x}_{n+R}\right)^{\mathrm{t}} \qquad \text{and} \qquad \flat\boldsymbol{Z}\mathrm{p}_n = \left(Z\mathrm{p}_{n+1}, Z\mathrm{p}_{n+2}, \dots, Z\mathrm{p}_{n+R}\right)^{\mathrm{t}},$$

the respective R-block vectors for the functions x and p. Similarly, the R-block vectors for the first derivatives are denoted by $\flat\boldsymbol{D}\mathrm{x}_n$ and $\flat\boldsymbol{D}\mathrm{p}_n$. The structural equations for the ZD scheme of size R then read

$$0 = \flat\boldsymbol{Z}\mathrm{x}_n + B_d\, \flat\boldsymbol{D}\mathrm{x}_n + Z\mathrm{x}_n\, b_z + D\mathrm{x}_n\, b_d, \tag{14}$$
$$0 = \flat\boldsymbol{Z}\mathrm{p}_n + B_d\, \flat\boldsymbol{D}\mathrm{p}_n + Z\mathrm{p}_n\, b_z + D\mathrm{p}_n\, b_d. \tag{15}$$

Remark: These two linear systems are independent and should be treated in parallel.

To solve the ODEs deriving from the Hamiltonian, we proceed in a very similar way as in Section 2.3.1. We produce two sequences $\left(\flat\boldsymbol{Z}\mathrm{x}_n^{(k)}, \flat\boldsymbol{D}\mathrm{x}_n^{(k)}\right)$ and $\left(\flat\boldsymbol{Z}\mathrm{p}_n^{(k)}, \flat\boldsymbol{D}\mathrm{p}_n^{(k)}\right)$ that converge to the solution. The fixed-point algorithm is then given by the following iterative procedure.

- **Initialization.** To build $\flat\boldsymbol{Z}\mathrm{x}_n^{(0)}$, $\flat\boldsymbol{D}\mathrm{x}_n^{(0)}$, $\flat\boldsymbol{Z}\mathrm{p}_n^{(0)}$, $\flat\boldsymbol{D}\mathrm{p}_n^{(0)}$, we set for $r \in \{1,\dots,R\}$

$$\begin{aligned} Z\mathrm{x}_{n+r}^{(0)} &= Z\mathrm{x}_{n+r-1}^{(0)} + \Delta t D\mathrm{x}_{n+r-1}^{(0)}, \\ Z\mathrm{p}_{n+r}^{(0)} &= Z\mathrm{p}_{n+r-1}^{(0)} + \Delta t D\mathrm{p}_{n+r-1}^{(0)}, \\ D\mathrm{x}_{n+r}^{(0)} &= \partial_p \mathcal{H}\left(Z\mathrm{x}_{n+r}^{(0)}, Z\mathrm{p}_{n+r}^{(0)}\right), \\ D\mathrm{p}_{n+r}^{(0)} &= -\partial_x \mathcal{H}\left(Z\mathrm{x}_{n+r}^{(0)}, Z\mathrm{p}_{n+r}^{(0)}\right), \end{aligned}$$

  with $Z\mathrm{x}_n^{(0)} = Z\mathrm{x}_n$, $D\mathrm{x}_n^{(0)} = D\mathrm{x}_n$ and $Z\mathrm{p}_n^{(0)} = Z\mathrm{p}_n$, $D\mathrm{p}_n^{(0)} = D\mathrm{p}_n$.
- **Iteration.** We first compute a new approximation for the solution using the structural equations

$$\begin{aligned} \flat\boldsymbol{Z}\mathrm{x}_n^{(k+1)} &= -\left(Z\mathrm{x}_n\, b_z + D\mathrm{x}_n\, b_d + B_d\, \flat\boldsymbol{D}\mathrm{x}_n^{(k)}\right), \\ \flat\boldsymbol{Z}\mathrm{p}_n^{(k+1)} &= -\left(Z\mathrm{p}_n\, b_z + D\mathrm{p}_n\, b_d + B_d\, \flat\boldsymbol{D}\mathrm{p}_n^{(k)}\right), \end{aligned}$$

  and then update the first derivatives with the physical equations deriving from the Hamiltonian

$$\begin{aligned} D\mathrm{x}_{n+r}^{(k+1)} &= \partial_p \mathcal{H}\left(Z\mathrm{x}_{n+r}^{(k+1)}, Z\mathrm{p}_{n+r}^{(k+1)}\right), \\ D\mathrm{p}_{n+r}^{(k+1)} &= -\partial_x \mathcal{H}\left(Z\mathrm{x}_{n+r}^{(k+1)}, Z\mathrm{p}_{n+r}^{(k+1)}\right). \end{aligned}$$

- **Stopping criterion.** We end the fixed-point when two successive solutions are close enough according to the tolerance parameter tol, that is $\|\flat\boldsymbol{Z}\mathrm{x}^{(k+1)} - \flat\boldsymbol{Z}\mathrm{x}^{(k)}\| \leq \mathtt{tol}$.

Remark: As in the case of the scalar ODE problem, no local linearization is required, leading to elementary formulations. □

3.2.2 ZDS schemes

Approximations of the function x and its two derivatives $\dot{x}$ and $\ddot{x}$ are connected via the structural equations (6), as are p and its derivatives. Therefore, the structural equations read

$$\forall m \in \{1,\dots,R\}, \qquad \begin{cases} \displaystyle\sum_{r=0}^{R} a_{r,0}^m Z\mathrm{x}_{n+r} + a_{r,1}^m D\mathrm{x}_{n+r} + a_{r,2}^m S\mathrm{x}_{n+r} = 0, \\ \displaystyle\sum_{r=0}^{R} a_{r,0}^m Z\mathrm{p}_{n+r} + a_{r,1}^m D\mathrm{p}_{n+r} + a_{r,2}^m S\mathrm{p}_{n+r} = 0. \end{cases}$$

It is important to note that x and p once again use exactly the same structural equations and only differ through the physical ones. We complete the notation with

$$\mathfrak{b}\boldsymbol{S}\mathrm{x}_n = \left(S\mathrm{x}_{n+1}, S\mathrm{x}_{n+2}, \cdots, S\mathrm{x}_{n+R}\right)^{\mathrm{t}}, \qquad \mathfrak{b}\boldsymbol{S}\mathrm{p}_n = \left(S\mathrm{p}_{n+1}, S\mathrm{p}_{n+2}, \cdots, S\mathrm{p}_{n+R}\right)^{\mathrm{t}},$$

the R-block vectors for the second derivatives. The two structural equations of size R then read

$$0 = \mathfrak{b}\boldsymbol{Z}\mathrm{x}_n + B_d\, \mathfrak{b}\boldsymbol{D}\mathrm{x}_n + B_s\, \mathfrak{b}\boldsymbol{S}\mathrm{x}_n + Z\mathrm{x}_n\, b_z + D\mathrm{x}_n\, b_d + S\mathrm{x}_n\, b_s, \tag{16}$$

$$0 = \mathfrak{b}\boldsymbol{Z}\mathrm{p}_n + B_d\, \mathfrak{b}\boldsymbol{D}\mathrm{p}_n + B_s\, \mathfrak{b}\boldsymbol{S}\mathrm{p}_n + Z\mathrm{p}_n\, b_z + D\mathrm{p}_n\, b_d + S\mathrm{p}_n\, b_s. \tag{17}$$

To solve the ODE deriving from the Hamiltonian, we proceed similarly as in the previous section. We produce two sequences $\left(\mathfrak{b}\boldsymbol{Z}\mathrm{x}_n^{(k)}, \mathfrak{b}\boldsymbol{D}\mathrm{x}_n^{(k)}, \mathfrak{b}\boldsymbol{S}\mathrm{x}_n^{(k)}\right)$ and $\left(\mathfrak{b}\boldsymbol{Z}\mathrm{p}_n^{(k)}, \mathfrak{b}\boldsymbol{D}\mathrm{p}_n^{(k)}, \mathfrak{b}\boldsymbol{S}\mathrm{p}_n^{(k)}\right)$ that converge to the solution. The fixed-point algorithm is detailed below.

- **Initialization.** To build $\mathfrak{b}\boldsymbol{Z}\mathrm{x}_n^{(0)}$, $\mathfrak{b}\boldsymbol{D}\mathrm{x}_n^{(0)}$, $\mathfrak{b}\boldsymbol{S}\mathrm{x}_n^{(0)}$, $\mathfrak{b}\boldsymbol{Z}\mathrm{p}_n^{(0)}$, $\mathfrak{b}\boldsymbol{D}\mathrm{p}_n^{(0)}$, $\mathfrak{b}\boldsymbol{S}\mathrm{p}_n^{(0)}$, we set for $r \in \{1,\dots,R\}$

$$\begin{aligned}
Z\mathrm{x}_{n+r}^{(0)} &= Z\mathrm{x}_{n+r-1}^{(0)} + \Delta t D\mathrm{x}_{n+r-1}^{(0)} + \frac{\Delta t^2}{2} S\mathrm{x}_{n+r-1}^{(0)}, \\
Z\mathrm{p}_{n+r}^{(0)} &= Z\mathrm{p}_{n+r-1}^{(0)} + \Delta t D\mathrm{p}_{n+r-1}^{(0)} + \frac{\Delta t^2}{2} S\mathrm{p}_{n+r-1}^{(0)}, \\
D\mathrm{x}_{n+r}^{(0)} &= \partial_p \mathcal{H}\left(Z\mathrm{x}_{n+r}^{(0)}, Z\mathrm{p}_{n+r}^{(0)}\right), \\
D\mathrm{p}_{n+r}^{(0)} &= -\partial_x \mathcal{H}\left(Z\mathrm{x}_{n+r}^{(0)}, Z\mathrm{p}_{n+r}^{(0)}\right), \\
S\mathrm{x}_{n+r}^{(0)} &= \partial_{xp} \mathcal{H}\left(Z\mathrm{x}_{n+r}^{(0)}, Z\mathrm{p}_{n+r}^{(0)}\right) D\mathrm{x}_{n+r}^{(0)} + \partial_{pp} \mathcal{H}\left(Z\mathrm{x}_{n+r}^{(0)}, Z\mathrm{p}_{n+r}^{(0)}\right) D\mathrm{p}_{n+r}^{(0)}, \\
S\mathrm{p}_{n+r}^{(0)} &= -\partial_{xx} \mathcal{H}\left(Z\mathrm{x}_{n+r}^{(0)}, Z\mathrm{p}_{n+r}^{(0)}\right) D\mathrm{x}_{n+r}^{(0)} - \partial_{xp} \mathcal{H}\left(Z\mathrm{x}_{n+r}^{(0)}, Z\mathrm{p}_{n+r}^{(0)}\right) D\mathrm{p}_{n+r}^{(0)}.
\end{aligned}$$

  with $Z\mathrm{x}_n^{(0)} = Z\mathrm{x}_n$, $D\mathrm{x}_n^{(0)} = D\mathrm{x}_n$, $S\mathrm{x}_n^{(0)} = S\mathrm{x}_n$ and $Z\mathrm{p}_n^{(0)} = Z\mathrm{p}_n$, $D\mathrm{p}_n^{(0)} = D\mathrm{p}_n$, $S\mathrm{p}_n^{(0)} = S\mathrm{p}_n$.
- **Iteration.** We first compute a new approximation for the solution using the structural equations

$$\begin{aligned}
\mathfrak{b}\boldsymbol{Z}\mathrm{x}_n^{(k+1)} &= -\left(Z\mathrm{x}_n\, b_z + D\mathrm{x}_n\, b_d + S\mathrm{x}_n\, b_s + B_d\, \mathfrak{b}\boldsymbol{D}\mathrm{x}_n^{(k)} + B_s\, \mathfrak{b}\boldsymbol{S}\mathrm{x}_n^{(k)}\right) \\
\mathfrak{b}\boldsymbol{Z}\mathrm{p}_n^{(k+1)} &= -\left(Z\mathrm{p}_n\, b_z + D\mathrm{p}_n\, b_d + S\mathrm{p}_n\, b_s + B_d\, \mathfrak{b}\boldsymbol{D}\mathrm{p}_n^{(k)} + B_s\, \mathfrak{b}\boldsymbol{S}\mathrm{p}_n^{(k)}\right)
\end{aligned}$$

  and then update the first and second derivative with the physical equations deriving from the Hamiltonian

$$\begin{aligned}
D\mathrm{x}_{n+r}^{(k+1)} &= \partial_p \mathcal{H}\left(Z\mathrm{x}_{n+r}^{(k+1)}, Z\mathrm{p}_{n+r}^{(k+1)}\right), \\
D\mathrm{p}_{n+r}^{(k+1)} &= -\partial_x \mathcal{H}\left(Z\mathrm{x}_{n+r}^{(k+1)}, Z\mathrm{p}_{n+r}^{(k+1)}\right), \\
S\mathrm{x}_{n+r}^{(k+1)} &= \partial_{xp} \mathcal{H}\left(Z\mathrm{x}_{n+r}^{(k+1)}, Z\mathrm{p}_{n+r}^{(k+1)}\right) D\mathrm{x}_{n+r}^{(k+1)} + \partial_{pp} \mathcal{H}\left(Z\mathrm{x}_{n+r}^{(k+1)}, Z\mathrm{p}_{n+r}^{(k+1)}\right) D\mathrm{p}_{n+r}^{(k+1)}, \\
S\mathrm{p}_{n+r}^{(k+1)} &= -\partial_{xx} \mathcal{H}\left(Z\mathrm{x}_{n+r}^{(k+1)}, Z\mathrm{p}_{n+r}^{(k+1)}\right) D\mathrm{x}_{n+r}^{(k+1)} - \partial_{xp} \mathcal{H}\left(Z\mathrm{x}_{n+r}^{(k+1)}, Z\mathrm{p}_{n+r}^{(k+1)}\right) D\mathrm{p}_{n+r}^{(k+1)}.
\end{aligned}$$

- **Stopping criterion.** We end the fixed-point algorithm when two successive solutions are close enough, according to the tolerance parameter `tol`, that is $\|\mathfrak{b}\boldsymbol{Z}\mathrm{x}^{(k+1)} - \mathfrak{b}\boldsymbol{Z}\mathrm{x}^{(k)}\| \leq \texttt{tol}$.

Remark: The coupling between the primal x and dual variables p takes place in the physical equations, hence the computation of $\flat \boldsymbol{Z}\mathrm{x}_n^{(k+1)}$ and $\flat \boldsymbol{Z}\mathrm{p}_n^{(k+1)}$ are independent and can be carried out in parallel. □

Remark: We have just considered a very basic fixed-point method for the sake of simplicity, but any accelerating techniques could be exploited. In practice, the predictor is good enough and very few iterations (3 to 12 in practice, depending on the tolerance and order of the method) are required to provide the accurate approximation. Of course, for stiff problems, a more sophisticated fixed-point procedure may be implemented. □

4 Structural method for Hamiltonian problems: systems of equations

We now generalize the method to the situation where the Hamiltonian system involves K vector-valued bodies. Such problems arise when modelling planetary systems, charged particles in electromagnetic fields, multiple coupled mass-damped-spring systems, among many other cases. The main issue of this extension lies in handling the technical difficulties arising when discretizing such complex systems. Overcoming these difficulties requires new notation and specific operators, which we introduce in Section 4.1. The structural algorithm is then recast using this notation, in Section 4.2.

4.1 Notation

The Hamiltonian system involves K distinct bodies characterized by their positions $\boldsymbol{x}^k \in \mathbb{R}^I$ and momenta $\boldsymbol{p}^k \in \mathbb{R}^I$. We gather the $\boldsymbol{x}$ and $\boldsymbol{p}$ in the "body space" $\mathfrak{B} = \mathbb{R}^{I\times K}$, and we introduce the matrix notation

$$X = \begin{bmatrix} x_1^1 & x_1^2 & \cdots & x_1^K \\ x_2^1 & x_2^2 & \cdots & x_2^K \\ \vdots & & & \vdots \\ x_I^1 & x_I^2 & & x_I^K \end{bmatrix} \quad \text{and} \quad P = \begin{bmatrix} p_1^1 & p_1^2 & \cdots & p_1^K \\ p_2^1 & p_2^2 & \cdots & p_2^K \\ \vdots & & & \vdots \\ p_I^1 & p_I^2 & & p_I^K \end{bmatrix}.$$

Let $X, P \in \mathfrak{B}$ and $\mathcal{H}(X,P) \in \mathbb{R}$ be the Hamiltonian. We adopt the following notation.

- The gradients $\nabla_X \mathcal{H}(X,P) \in \mathfrak{B}$ and $\nabla_P \mathcal{H}(X,P) \in \mathfrak{B}$ are given by
$$\nabla_X \mathcal{H}(X,P) = \left[\partial_{x_i^k} \mathcal{H}(X,P)\right]_{k\in\{1,\dots,K\}, i\in\{1,\dots,I\}}.$$
Note that the gradient does not output a vector but is reshaped to provide an element in $\mathfrak{B}$.
- The second derivatives $\nabla_{XX}\mathcal{H}(X,P)$, $\nabla_{PP}\mathcal{H}(X,P)$, $\nabla_{XP}\mathcal{H}(X,P) = \nabla_X(\nabla_P\mathcal{H}(X,P))$, and $\nabla_{PX}\mathcal{H}(X,P) = \nabla_P(\nabla_X\mathcal{H}(X,P))$ belong to $\mathfrak{B}^2$. Note that the order ∇_{XP} and ∇_{PX} are, in general, different. We introduce the following computation rules, using Einstein summation:
$$\begin{aligned} \nabla_{PX}\mathcal{H}(X,P)\cdot\dot{p} &= \left[\partial_{p_j^m}\partial_{x_i^k}\mathcal{H}(X,P)\,\dot{p}_j^m\right]_{k\in\{1,\dots,K\}, i\in\{1,\dots,I\}}, \\ \nabla_{XP}\mathcal{H}(X,P)\cdot\dot{x} &= \left[\partial_{x_j^m}\partial_{p_i^k}\mathcal{H}(X,P)\,\dot{x}_j^m\right]_{k\in\{1,\dots,K\}, i\in\{1,\dots,I\}}. \end{aligned}$$

We also define the tensors that gather all the data we shall handle in the structural method.

- We denote by $\boldsymbol{Z}\mathbf{x}_n \approx \boldsymbol{x}(t_n) \in \mathfrak{B}$ an approximation of the positions of the K bodies at the time t_n. Similar notation is used for the first and second derivatives, $\boldsymbol{D}\mathbf{x}_n \approx \dot{\boldsymbol{x}}(t_n) \in \mathfrak{B}$ and $\boldsymbol{S}\mathbf{x}_n \approx \ddot{\boldsymbol{x}}(t_n) \in \mathfrak{B}$. We adopt the same notations $\boldsymbol{Z}\mathbf{p}_n$, $\boldsymbol{D}\mathbf{p}_n$ and $\boldsymbol{S}\mathbf{p}_n$ for the momentum $\boldsymbol{p}$. Moreover, for anybody k, the row vector $\boldsymbol{Z}\mathbf{x}_n[k]$ represents its space coordinates in $\mathbb{R}^I$.
- Given a block of size R, we introduce the $\mathfrak{B}$-valued vector $\flat\boldsymbol{Z}\mathbf{x}_n \in \mathfrak{B}^R$ that gathers the coordinates of all the K bodies from time t_{n+1} until time t_{n+R} with the convention $\flat\boldsymbol{Z}\mathbf{x}_n[r] = \boldsymbol{Z}\mathbf{x}_{n+r} \in \mathfrak{B}$ and $\flat\boldsymbol{Z}\mathbf{x}_n[r][k] = \boldsymbol{Z}\mathbf{x}_{n+r}[k] \in \mathbb{R}^I$.
- We introduce similar notation for $\flat\boldsymbol{Z}\mathbf{p}_n$ and the derivatives, namely $\flat\boldsymbol{D}\mathbf{x}_n$, $\flat\boldsymbol{D}\mathbf{p}_n$, $\flat\boldsymbol{S}\mathbf{x}_n$, $\flat\boldsymbol{S}\mathbf{p}_n$, elements of $\mathfrak{B}^R$. The main difference compared to the previous section is that $\boldsymbol{Z}\mathbf{x}_{n+r}$ now belongs to the body space $\mathfrak{B}$ instead of being real values.

A last point concerns the linear algebra with elements of $\mathfrak{B}$. To this end, we introduce the following notation

- Let $\boldsymbol{Z}\mathbf{x} \in \mathfrak{B}$ and $\alpha \in \mathbb{R}$. Then $\alpha\boldsymbol{Z}\mathbf{x} \in \mathfrak{B}$ is the usual product of a matrix with a real number.

- Let $\boldsymbol{Z}\mathbf{x} \in \mathfrak{B}$ and $a = (a_r)_r \in \mathbb{R}^R$, then $a \otimes \boldsymbol{Z}\mathbf{x}$ is the vector in $\mathfrak{B}^R$ given by

$$a \otimes \boldsymbol{Z}\mathbf{x} = \begin{bmatrix} a_1 \boldsymbol{Z}\mathbf{x} \\ a_2 \boldsymbol{Z}\mathbf{x} \\ \vdots \\ a_R \boldsymbol{Z}\mathbf{x} \end{bmatrix}.$$

- Let $\flat\boldsymbol{Z}\mathbf{x} \in \mathfrak{B}^R$ and $A = (a_{r,m})_{r,m}$ a square matrix of dimension R, then $A \times \flat\boldsymbol{Z}\mathbf{x}$ is the vector in $\mathfrak{B}^R$ given by

$$A \times \flat\boldsymbol{Z}\mathbf{x} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_r \end{bmatrix}, \qquad \text{with } y_r = \sum_{m=1}^{R} a_{r,m} \flat\boldsymbol{Z}\mathbf{x}[m] \in \mathfrak{B} \quad \text{for all } r \in \{1, \dots, R\}.$$

The ZD structural equations (14)–(15) adapted to vectors of $\mathfrak{B}^R$ read

$$0 = \flat\boldsymbol{Z}\mathbf{x}_n + B_d \times \flat\boldsymbol{D}\mathbf{x}_n + b_z \otimes \boldsymbol{Z}\mathbf{x}_n + b_d \otimes \boldsymbol{D}\mathbf{x}_n, \tag{18}$$
$$0 = \flat\boldsymbol{Z}\mathbf{p}_n + B_d \times \flat\boldsymbol{D}\mathbf{p}_n + b_z \otimes \boldsymbol{Z}\mathbf{p}_n + b_d \otimes \boldsymbol{D}\mathbf{p}_n, \tag{19}$$

while the ZDS structural equations (16)–(17) read

$$0 = \flat\boldsymbol{Z}\mathbf{x}_n + B_d \times \flat\boldsymbol{D}\mathbf{x}_n + B_s \times \flat\boldsymbol{S}\mathbf{x}_n + b_z \otimes \boldsymbol{Z}\mathbf{x}_n + b_d \otimes \boldsymbol{D}\mathbf{x}_n + b_s \otimes \boldsymbol{S}\mathbf{x}_n, \tag{20}$$
$$0 = \flat\boldsymbol{Z}\mathbf{p}_n + B_d \times \flat\boldsymbol{D}\mathbf{p}_n + B_s \times \flat\boldsymbol{S}\mathbf{p}_n + b_z \otimes \boldsymbol{Z}\mathbf{p}_n + b_d \otimes \boldsymbol{D}\mathbf{p}_n + b_s \otimes \boldsymbol{S}\mathbf{p}_n. \tag{21}$$

Remark: We highlight again that we use the same notation for matrices B_d and vectors b_z, b_d for the sake of simplicity, but they are different for the ZD or the ZDS methods. Additionally, the computation of the products is largely parallelizable, which may be taken advantage of in an HPC context. □

4.2 Algorithm

We rephrase the fixed-point method with the notation introduced in Section 4.1. Once again, we need an initialization and an iteration routine. We just present the ZDS version. Indeed, the simpler ZD version only consists in extracting from the ZDS version the two first physical equations (29)–(30) and structural equations (14)–(15).

Given the initial state $(\boldsymbol{Z}\mathbf{x}_n, \boldsymbol{D}\mathbf{x}_n, \boldsymbol{S}\mathbf{x}_n)$ and $(\boldsymbol{Z}\mathbf{p}_n, \boldsymbol{D}\mathbf{p}_n, \boldsymbol{S}\mathbf{p}_n)$, we define two sequences $(\flat\boldsymbol{Z}\mathbf{x}_n^{(k)}, \flat\boldsymbol{D}\mathbf{x}_n^{(k)}, \flat\boldsymbol{S}\mathbf{x}_n^{(k)})$ and $(\flat\boldsymbol{Z}\mathbf{p}_n^{(k)}, \flat\boldsymbol{D}\mathbf{p}_n^{(k)}, \flat\boldsymbol{S}\mathbf{p}_n^{(k)})$ of elements of $\mathfrak{B}^R$ that converge to the R-block solution. We recall the convention $\flat\boldsymbol{Z}\mathbf{x}_n^{(k)}[r] = \boldsymbol{Z}\mathbf{x}_{n+r} \in \mathfrak{B}$ for all $r \in \{1, \dots, R\}$.

- **Initialization.** To initialize the R-size blocks $\flat\boldsymbol{Z}\mathbf{x}_n^{(0)}$, $\flat\boldsymbol{D}\mathbf{x}_n^{(0)}$, $\flat\boldsymbol{S}\mathbf{x}_n^{(0)}$, $\flat\boldsymbol{Z}\mathbf{p}_n^{(0)}$, $\flat\boldsymbol{D}\mathbf{p}_n^{(0)}$, $\flat\boldsymbol{S}\mathbf{p}_n^{(0)}$ for all $r \in \{1, \dots, R\}$, we use the second order Taylor expansion in space $\mathfrak{B}$:

$$\boldsymbol{Z}\mathbf{x}_{n+r}^{(0)} = \boldsymbol{Z}\mathbf{x}_{n+r-1}^{(0)} + \Delta t\, \boldsymbol{D}\mathbf{x}_{n+r-1}^{(0)} + \frac{\Delta t^2}{2} \boldsymbol{S}\mathbf{x}_{n+r-1}^{(0)},$$
$$\boldsymbol{Z}\mathbf{p}_{n+r}^{(0)} = \boldsymbol{Z}\mathbf{p}_{n+r-1}^{(0)} + \Delta t\, \boldsymbol{D}\mathbf{p}_{n+r-1}^{(0)} + \frac{\Delta t^2}{2} \boldsymbol{S}\mathbf{p}_{n+r-1}^{(0)},$$

where we compute the first and second derivatives through the physical equations

$$\boldsymbol{D}\mathbf{x}_{n+r}^{(0)} = \nabla_P \mathcal{H}\left(\boldsymbol{Z}\mathbf{x}_{n+r}^{(0)}, \boldsymbol{Z}\mathbf{p}_{n+r}^{(0)}\right),$$
$$\boldsymbol{D}\mathbf{p}_{n+r}^{(0)} = -\nabla_X \mathcal{H}\left(\boldsymbol{Z}\mathbf{x}_{n+r}^{(0)}, \boldsymbol{Z}\mathbf{p}_{n+r}^{(0)}\right),$$
$$\boldsymbol{S}\mathbf{x}_{n+r}^{(0)} = \nabla_{XP} \mathcal{H}\left(\boldsymbol{Z}\mathbf{x}_{n+r}^{(0)}, \boldsymbol{Z}\mathbf{p}_{n+r}^{(0)}\right) \boldsymbol{D}\mathbf{x}_{n+r}^{(0)} + \nabla_{PP} \mathcal{H}\left(\boldsymbol{Z}\mathbf{x}_{n+r}^{(0)}, \boldsymbol{Z}\mathbf{p}_{n+r}^{(0)}\right) D\mathrm{p}_{n+r}^{(0)},$$
$$\boldsymbol{S}\mathbf{p}_{n+r}^{(0)} = -\nabla_{XX} \mathcal{H}\left(\boldsymbol{Z}\mathbf{x}_{n+r}^{(0)}, \boldsymbol{Z}\mathbf{p}_{n+r}^{(0)}\right) \boldsymbol{D}\mathbf{x}_{n+r}^{(0)} - \nabla_{PX} \mathcal{H}\left(\boldsymbol{Z}\mathbf{x}_{n+r}^{(0)}, \boldsymbol{Z}\mathbf{p}_{n+r}^{(0)}\right) \boldsymbol{D}\mathbf{p}_{n+r}^{(0)}$$

with $\boldsymbol{Z}\mathbf{x}_n^{(0)} = \boldsymbol{Z}\mathbf{x}_n$, $\boldsymbol{D}\mathbf{x}_n^{(0)} = \boldsymbol{D}\mathbf{x}_n$, $\boldsymbol{S}\mathbf{x}_n^{(0)} = \boldsymbol{S}\mathbf{x}_n$ and $\boldsymbol{Z}\mathbf{p}_n^{(0)} = \boldsymbol{Z}\mathbf{p}_n$, $\boldsymbol{D}\mathbf{p}_n^{(0)} = \boldsymbol{D}\mathbf{p}_n$, $\boldsymbol{S}\mathbf{p}_n^{(0)} = \boldsymbol{S}\mathbf{p}_n$.

- **Iteration.** We first compute new R-size blocks $\flat\boldsymbol{Z}\mathbf{x}_n^{(k+1)}$ and $\flat\boldsymbol{Z}\mathbf{p}_n^{(k+1)}$, approximating the positions and momenta, using the system of structural equations:

$$\begin{aligned}\flat\boldsymbol{Z}\mathbf{x}_n^{(k+1)} &= -\Big(b_z \otimes \boldsymbol{Z}\mathbf{x}_n + b_d \otimes \boldsymbol{D}\mathbf{x}_n + b_s \otimes \boldsymbol{S}\mathbf{x}_n + B_d \times \flat\boldsymbol{D}\mathbf{x}_n^{(k)} + B_s \times \flat\boldsymbol{S}\mathbf{x}_n^{(k)}\Big),\\ \flat\boldsymbol{Z}\mathbf{p}_n^{(k+1)} &= -\Big(b_z \otimes \boldsymbol{Z}\mathbf{p}_n + b_d \otimes \boldsymbol{D}\mathbf{p}_n + b_s \otimes \boldsymbol{S}\mathbf{p}_n + B_d \times \flat\boldsymbol{D}\mathbf{p}_n^{(k)} + B_s \times \flat\boldsymbol{S}\mathbf{p}_n^{(k)}\Big).\end{aligned}$$

  Then, for each $r \in \{1, \dots, R\}$, we update the first and second derivatives at each time step t_{n+r} by setting

$$\begin{aligned}\boldsymbol{D}\mathbf{x}_{n+r}^{(k+1)} &= \nabla_P \mathcal{H}\left(\boldsymbol{Z}\mathbf{x}_{n+r}^{(k+1)}, \boldsymbol{Z}\mathbf{p}_{n+r}^{(k+1)}\right),\\ \boldsymbol{D}\mathbf{p}_{n+r}^{(k+1)} &= -\nabla_X \mathcal{H}\left(\boldsymbol{Z}\mathbf{x}_{n+r}^{(k+1)}, \boldsymbol{Z}\mathbf{p}_{n+r}^{(k+1)}\right),\\ \boldsymbol{S}\mathbf{x}_{n+r}^{(k+1)} &= \nabla_{XP} \mathcal{H}\left(\boldsymbol{Z}\mathbf{x}_{n+r}^{(k+1)}, \boldsymbol{Z}\mathbf{p}_{n+r}^{(k+1)}\right)\boldsymbol{D}\mathbf{x}_{n+r}^{(k+1)} + \nabla_{PP} \mathcal{H}\left(\boldsymbol{Z}\mathbf{x}_{n+r}^{(k+1)}, \boldsymbol{Z}\mathbf{p}_{n+r}^{(k+1)}\right)\boldsymbol{D}\mathbf{p}_{n+r}^{(k+1)},\\ \boldsymbol{S}\mathbf{p}_{n+r}^{(k+1)} &= -\nabla_{XX} \mathcal{H}\left(\boldsymbol{Z}\mathbf{x}_{n+r}^{(k+1)}, \boldsymbol{Z}\mathbf{p}_{n+r}^{(k+1)}\right)\boldsymbol{D}\mathbf{x}_{n+r}^{(k+1)} - \nabla_{PX} \mathcal{H}\left(\boldsymbol{Z}\mathbf{x}_{n+r}^{(k+1)}, \boldsymbol{Z}\mathbf{p}_{n+r}^{(k+1)}\right)\boldsymbol{D}\mathbf{p}_{n+r}^{(k+1)}.\end{aligned}$$

- **Stopping criterion.** We end the fixed-point scheme once two successive solutions are close enough, according to the tolerance parameter `tol`, that is $\|\boldsymbol{Z}\mathbf{x}_{n+R}^{(k+1)} - \boldsymbol{Z}\mathbf{x}_{n+R}^{(k)}\| \leq \texttt{tol}$.

Remark: Once again, we highlight that the procedure is highly parallelizable since the update of the derivative for each time steps are independent.

5 Benchmarks

We propose and analyze a series of benchmarks to assess the properties of the scheme. Namely, we check the accuracy when an analytical solution is available, as well as the numerical preservation of several invariants, like the Hamiltonian, for long-time simulations. In some cases, we also check the preservation of other invariant quantities of interest. The method has been implemented in `C++` in IEEE 754 `quadruple` and `octuple` precision using the library `qd` developed by Hida et al. [2000]. The new structural method will be compared to standard symplectic schemes already implemented in `julia 1.11`, see Bezanson et al. [2017]. These schemes are provided by the `DifferentialEquations.jl` library, see Rackauckas and Nie [2017], and we use the second-order accurate `McAte2` scheme from McLachlan and Atela [1992], the fourth-order accurate `CalvoSanz4` scheme from Sanz-Serna and Calvo [1993], and the sixth- and eighth-order accurate `KahanLi6` and `KahanLi8` schemes from Kahan and Li [1997].

The benchmarks and their settings are summarized in Table 1. Moreover, Section 5.8 contains a study of the complexity of the `ZD` and `ZDS` methods, compared to the classical ones, in separable and non-separable settings, as well as a study of the computational cost of the fixed-point method.

Table 1: Summary of the benchmarks.

Benchmark	Exact Solution	Additional Invariants	Separable	Section
One-dimensional mass-spring problem	Yes	No	Yes	5.1
Two springs, two masses system	Yes	No	Yes	5.2
One-dimensional pendulum problem	Yes	No	Yes	5.3
Two-dimensional Kepler problem	No	Yes	Yes	5.4
Two-dimensional three-body problem	No	Yes	Yes	5.5
Outer Solar system	No	Yes	Yes	5.6
Particle in a 3D electromagnetic field	No	No	No	5.7

In each benchmark, the time interval $[0, T]$ is uniformly divided into N steps of constant size Δt, and set $t_n = n\Delta T$, for all $n \in \{0, \dots, N\}$ to represent the subdivision of this interval. Denoting by $\bar{q}(t_n)$ the exact quantity at time t_n (position, momentum, or invariant) and q_n its approximation, we evaluate the error by computing $\texttt{eq}_n(\Delta t) = |q_n - \bar{x}(t_n)|$, and we define the maximum error, together with the order, by

$$\texttt{eq}(\Delta t) = \max_n \texttt{eq}_n(\Delta t), \qquad \texttt{ordq}(\Delta t_1, \Delta t_2) = \frac{\log\left(\texttt{eq}(\Delta t_1)/\texttt{eq}(\Delta t_2)\right)}{\log(\Delta t_1/\Delta t_2)}.$$

For instance, `ex` will represent the position error and `eH` the deviation of the Hamiltonian. All simulations have been carried out with `quadruple` precision since the `double` precision is not enough to correctly compute the tiny errors obtained when using very high order accurate methods (methods of eighth or tenth order, for instance).

5.1 Linear case: one-dimensional mass-spring problem

As a very first sanity check experiment, we consider the linear one-dimensional mass-spring problem with no damping term. The Hamiltonian is

$$\mathcal{H}(x,p) = \frac{p^2}{2m} + \frac{1}{2}\kappa x^2,$$

with m the mass and κ the spring constant.

5.1.1 The physical equations

We derive the first set of physical equations from Hamilton's principle:

$$m\dot{x} = p, \qquad \dot{p} = -\kappa x. \tag{22}$$

Differentiating system (22) in time, we get the second set of physical equations

$$m\ddot{x} = \dot{p}, \qquad \ddot{p} = -\kappa \dot{x}. \tag{23}$$

Form relation (22), we deduce the first group of physical equations $\texttt{PE}[1]$, given by

$$m\, D\mathrm{x} = Z\mathrm{p}, \quad D\mathrm{p} = -\kappa\, Z\mathrm{x},$$

while (23) provides the second group of physical equations $\texttt{PE}[2]$, which reads

$$m\, S\mathrm{x} = D\mathrm{p}, \quad S\mathrm{p} = -\kappa\, D\mathrm{x}.$$

The ZD solver computes the R-sized block of unknowns $\flat\boldsymbol{Z}\mathrm{x}_n$, $\flat\boldsymbol{D}\mathrm{x}_n$ and $\flat\boldsymbol{Z}\mathrm{p}_n$, $\flat\boldsymbol{D}\mathrm{p}_n$, by coupling the physical equations $\texttt{PE}_{n+r}[1]$ and the first R structural equations $\texttt{SE}[r]$ given by (4), with $r \in \{1,\dots,R\}$.

The ZDS solver computes the R-sized block of unknowns $\flat\boldsymbol{Z}\mathrm{x}_n$, $\flat\boldsymbol{D}\mathrm{x}_n$, $\flat\boldsymbol{S}\mathrm{x}_n$ and $\flat\boldsymbol{Z}\mathrm{p}_n$, $\flat\boldsymbol{D}\mathrm{p}_n$, $\flat\boldsymbol{S}\mathrm{p}_n$, by coupling the physical equations $\texttt{PE}_{n+r}[1]$, $\texttt{PE}_{n+r}[2]$ and the first R structural equations $\texttt{SE}[r]$ given by (6), with $r \in \{1,\dots,R\}$.

5.1.2 Numerical tests

Numerical simulations have been carried out for a final time $T = 100$ with $\kappa = 1$ and $m = 1$, which corresponds to a 2π-periodic motion. Initial conditions are $x(0) = 1$ and $p(0) = 0$. We report[1] in Table 2a the position error and order of accuracy, obtained by using the ZD scheme with different values of R. We proceed similarly with the ZDS scheme[2] in Table 2b, while Table 2c gives the errors for the classical symplectic schemes.

We remark that the accuracy and order of convergence are the optimal ones, given by the theoretical analysis. The structural ZDS method provides the same error magnitude as the classical symplectic schemes, while the ZDS method, for an equivalent order of accuracy, provides an error up to three orders of magnitude smaller. The ZDS scheme is clearly more compact with respect to the ZD case and also provides accurate approximations for the second derivative.

To assess the symplectic property of the schemes, we run the simulations for a very long time $T = 100\,000s$ and use two subdivisions: a coarse grid with $N = 6T$ points and a finer grid with $N = 24T$ points. We report in Table 3 the position error and the Hamiltonian error for the ZD, ZDS and classical schemes respectively. The time evolution of the Hamiltonian error is constant after a few steps (not printed in the document) for all the schemes. The ZD and ZDS methods provide a better preservation of the Hamiltonian, quite similar whatever the number of physical equations and independent of the method order.

The long-time simulation also confirms the excellent accuracy of the structural schemes, with a special mention for the ZDS case where the $R = 4$ case provides an error of the order of 10^{-15} with a discretization corresponding to 24 points per revolution. We also highlight that we recover all the convergence orders by computing the ratio of ex for $N = 6T$ and $N = 24T$. On the contrary, no convergence order is available for the Hamiltonian error since the deviations are almost the same for both time discretization.

5.2 Two springs, two masses system

We consider a system composed of two bodies linked by two springs, following Figure 1. The system is described by the Hamiltonian functional, whose inputs are two vectors $x = (x_1, x_2)$ and $p = (p_1, p_2)$ in $\mathbb{R}^2$, and whose expression is

$$\mathcal{H}(x,p) = \frac{p_1^2}{2m_1} + \frac{p_2^2}{2m_2} + \frac{1}{2}k_1 x_1^2 + \frac{1}{2}k_2(x_2 - x_1)^2. \tag{24}$$

Table 2: Mass-spring test case from Section 5.1: errors on the position at $T = 100$s.

(a) Errors obtained with the ZD method.

	R=2		R=4		R=6		R=8	
N	ex	ordx	ex	ordx	ex	ordx	ex	ordx
120	7.22e-01	—	2.26e-01	—	4.53e-02	—	2.92e-03*	—
240	5.43e-02	3.7	5.04e-03	5.5	5.07e-04	6.5	5.17e-05	9.0
480	3.57e-03	3.9	8.67e-05	5.9	2.45e-06	7.7	7.48e-08	9.4
960	2.25e-04	4.0	1.39e-06	6.0	1.01e-08	7.9	7.97e-11	9.9

(b) Errors obtained with the ZDS method.

	R=1		R=2		R=3		R=4	
N	ex	ordx	ex	ordx	ex	ordx	ex	ordx
120	5.43e-02	—	2.59e-03	—	1.20e-04	—	5.67e-07†	—
240	3.57e-03	3.9	4.58e-05	5.8	6.73e-07	7.5	1.10e-08	9.2
480	2.25e-04	4.0	7.38e-07	6.0	2.85e-09	7.9	1.28e-11	9.7
960	1.41e-05	4.0	1.16e-08	6.0	1.14e-11	8.0	1.30e-14	9.9

(c) Errors obtained with the classical method.

	MA2		CS4		KL6		KL8	
N	ex	ordx	ex	ordx	ex	ordx	ex	ordx
120	1.34e+01	—	3.48e-02	—	1.35e-01	—	2.97e-04	—
240	3.52e+00	1.9	2.16e-03	4.0	2.08e-03	6.0	1.11e-06	8.1
480	8.73e-01	2.0	1.37e-04	4.0	3.30e-05	6.0	4.37e-09	8.0
960	2.18e-01	2.0	8.58e-06	4.0	5.15e-07	6.0	1.70e-11	8.0

Table 3: Mass-spring test case from Section 5.1: error on the position and the Hamiltonian at $T = 100\,000$s. For ex and eH, both rows correspond to $N = T \times 6$ and $N = T \times 24$ respectively.

(a) Errors obtained with the ZD scheme.

	R=2	R=4	R=6	R=8
ex($\times 6$)	1.49e+00	2.54e-02	4.83e-04	9.39e-06
ex($\times 24$)	6.55e-03	6.58e-06	7.74e-09	9.89e-12
eH($\times 6$)	1.11e-25	3.89e-27	5.16e-28	5.30e-27
eH($\times 24$)	4.18e-25	7.47e-26	1.06e-25	2.25e-26

(b) Errors obtained with the ZDS scheme.

	R=1	R=2	R=3	R=4
ex($\times 6$)	1.06e-01	2.25e-04	5.60e-07	1.62e-09
ex($\times 24$)	4.18e-04	5.53e-08	8.68e-12	1.60e-15
eH($\times 6$)	8.30e-26	1.48e-25	3.10e-27	1.79e-25
eH($\times 24$)	3.09e-25	5.32e-25	1.01e-25	4.61e-26

(c) Errors obtained with the classical schemes.

	MA2	CS4	KL6	KL8
ex($\times 6$)	2.00e+01	5.69e-03	8.74e-04	7.40e-08
ex($\times 24$)	3.49e+00	2.22e-05	2.13e-07	1.13e-12
eH($\times 6$)	9.97e-05	5.22e-07	1.02e-08	7.48e-13
eH($\times 24$)	1.55e-06	2.03e-09	2.47e-12	1.15e-17

[1] case * is evaluated with $N = 156$.
[2] case † is evaluated with $N = 156$.

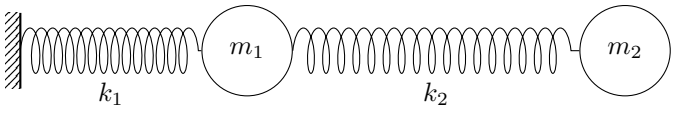

Figure 1: Schematic representation of the two masses-spring system.

The analytical solution (x, p) is given by

$$
\begin{aligned}
x_1(t) &= A\cos(\omega_1 t + \alpha_1) + B\cos(\omega_2 t + \alpha_2),\\
x_2(t) &= \frac{A\left(k_1 + k_2 - m_1\omega_1^2\right)}{k_2}\cos(\omega_1 t + \alpha_1) + \frac{Bk_2}{k_2 - m_2\omega_2^2}\cos(\omega_2 t + \alpha_2),\\
p_1(t) &= -m_1\left(A\omega_1\sin(t\omega_1 + \alpha_1) + B\omega_2\sin(t\omega_2 + \alpha_2)\right),\\
p_2(t) &= -m_2\left(\frac{A\omega_1\left(k_1 + k_2 - m_1\omega_1^2\right)}{k_2}\sin(\omega_1 t + \alpha_1) + \frac{B\omega_2 k_2}{k_2 - m_2\omega_2^2}\sin(\omega_2 t + \alpha_2)\right),
\end{aligned}
$$

where ω_1 and ω_2 are positive solutions to

$$m_1 m_2\omega^4 - (m_1k_2 + m_2k_1 + m_2k_2)\omega^2 + k_1k_2 = 0.$$

5.2.1 Physical equations

Hamilton's equation $\dot{x} = \nabla_p\mathcal{H}(x,p)$ and $\dot{p} = -\nabla_x\mathcal{H}(x,p)$, applied to the Hamiltonian (24), yields the following dynamical system:

$$\dot{x} = \begin{pmatrix} p_1/m_1 \\ p_2/m_2 \end{pmatrix}, \qquad \dot{p} = -\begin{pmatrix} k_1x_1 + k_2(x_1 - x_2) \\ k_2(x_2 - x_1) \end{pmatrix}.$$

Differentiating with respect to time leads to the system

$$\ddot{x} = \begin{pmatrix} \dot{p}_1/m_1 \\ \dot{p}_2/m_2 \end{pmatrix}, \qquad \ddot{p} = -\begin{pmatrix} k_1\dot{x}_1 + k_2(\dot{x}_1 - \dot{x}_2) \\ k_2(\dot{x}_2 - \dot{x}_1) \end{pmatrix}.$$

We deduce the first group of structural equation PE[1], given by

$$D\mathrm{x} = \begin{bmatrix} 1/m_1 & 0 \\ 0 & 1/m_2 \end{bmatrix} Z\mathrm{p}, \qquad D\mathrm{p} = -\begin{bmatrix} k_1 + k_2 & -k_2 \\ -k_2 & k_2 \end{bmatrix} Z\mathrm{x},$$

while the second group of physical equation PE[2] reads

$$S\mathrm{x} = \begin{bmatrix} 1/m_1 & 0 \\ 0 & 1/m_2 \end{bmatrix} D\mathrm{p}, \qquad S\mathrm{p} = -\begin{bmatrix} k_1 + k_2 & -k_2 \\ -k_2 & k_2 \end{bmatrix} D\mathrm{x}.$$

5.2.2 Numerical tests

The analytical solution is computed for the physical parameters $k_1 = 1$, $k_2 = 5$, $m_1 = 2$, $m_2 = 1$ that provides the two frequencies

$$\omega_1 = \sqrt{\frac{8 - 3\sqrt{6}}{2}} \approx 0.57 \quad \text{and} \quad \omega_2 = \sqrt{\frac{8 + 3\sqrt{6}}{2}} \approx 2.77.$$

Moreover, the initial condition is made of the analytical solution taken at time $t = 0$, with $A = 1$, $B = 2$, $\alpha_1 = \pi/2$ and $\alpha_2 = -\pi/4$. Numerical simulations have been carried out with the ZD, ZDS and classical schemes. The error on the position and the conservation of the Hamiltonian are the two main indicators we use to assess the quality of the simulation.

Accuracy and order of convergence. The final time is set to $T = 100$s, and we take $N = 480$, 960, 1920 and 3840. We recall that the smallest period is $T_2 = 2.27$, corresponding to ω_2. Thus, the coarse grid $N = 480$ corresponds to about 11 points to cover a full period.

Table 4 report the maximal error and order of convergence for the ZD, ZDS and classical schemes, respectively. The optimal order is clearly achieved in each situation. The errors' magnitude is almost the same for the ZDS scheme and the classical schemes for a given order, whereas the ZD scheme presents an error larger by about three orders of magnitude.

Table 4: Two springs, two masses test case from Section 5.2: error on the position at $T = 100$s.

(a) Errors obtained with the ZD method.

	R=2		R=4		R=6		R=8	
N	ex	ordx	ex	ordx	ex	ordx	ex	ordx
480	2.29e+00	—	3.70e-01	—	5.91e-02	—	8.94e-03	—
960	1.53e-01	3.9	6.97e-03	5.7	3.55e-04	7.4	2.03e-05	8.8
1920	9.83e-03	4.0	1.14e-04	5.9	1.57e-06	7.8	2.37e-08	9.7
3840	6.19e-04	4.0	1.83e-06	6.0	6.42e-09	7.9	2.41e-11	9.9

(b) Errors obtained with the ZDS method.

	R=1		R=2		R=3		R=4	
N	ex	ordx	ex	ordx	ex	ordx	ex	ordx
480	1.53e-01	—	3.66e-03	—	9.38e-05	—	2.65e-06	—
960	9.83e-03	4.0	6.06e-05	5.9	4.40e-07	7.7	3.71e-09	9.5
1920	6.19e-04	4.0	9.75e-07	6.0	1.83e-09	7.9	3.92e-12	9.9
3840	3.88e-05	4.0	1.53e-08	6.0	7.20e-12	8.0	3.96e-15	9.9

(c) Errors obtained with the classical methods.

	MA2		CS4		KL6		KL8	
N	ex	ordx	ex	ordx	ex	ordx	ex	ordx
480	6.11e+00	—	8.27e-03	—	1.54e-02	—	1.59e-05	—
960	1.72e+00	1.8	5.20e-04	4.0	2.42e-04	6.0	6.16e-08	8.0
1920	4.35e-01	2.0	3.27e-05	4.0	3.79e-06	6.0	2.40e-10	8.0
3840	1.09e-01	2.0	2.04e-06	4.0	5.93e-08	6.0	1.03e-12	7.9

Preservation of the Hamiltonian. To address the Hamiltonian preservation issue, we take $T = 10\,000$ and assess the position error together with the Hamiltonian deviation, checking the evolution of the maximum error over time. A first note concerns the accuracy of the position: the errors increase with time, but the ZD and ZDS schemes provide lower deviations than the classical schemes. Indeed, for the same order, we observe a gain of two orders of magnitude when using the ZDS scheme rather than the classical schemes, while the difference between the ZD scheme and the standard schemes was reduced to a factor 10. In conclusion, the ZD and ZDS schemes offer a better accuracy when dealing with long-time simulations (i.e., numerous time steps).

Concerning the Hamilton conservation, the ZD and ZDS schemes provide almost the same very low error of around 10^{-25}, but the standard schemes do not reach such a target. Moreover, the deviation from the initial Hamiltonian depends on the grid size for the classical schemes, while errors with the ZD and ZDS schemes are almost independent of Δt.

5.3 One-dimensional pendulum problem

We turn to a nonlinear scalar problem, and consider the one-dimensional pendulum system governed by the Hamiltonian

$$\mathcal{H}(x,p) = \frac{p^2}{2m\ell^2} + mg\ell(1-\cos(x)) \tag{25}$$

where m is the mass, ℓ the length of the pendulum and g the gravity.

5.3.1 Physical equations

The differential system deriving from Hamilton's equations reads

$$\dot{x} = \frac{p}{m}, \qquad \dot{p} = -mg\ell\sin(x) \tag{26}$$

Table 5: Two springs, two masses test case from Section 5.2: error on the position and the Hamiltonian at $T = 10\,000$s. For ex and eH, both rows correspond to $N = T \times 24$ and $N = T \times 96$ respectively.

(a) Errors obtained with the ZD scheme.

	R=2	R=4	R=6	R=8
ex($\times 24$)	4.07e-01	3.08e-03	2.74e-05	2.60e-07
ex($\times 96$)	1.60e-03	7.59e-07	4.26e-10	2.62e-13
eH($\times 24$)	3.70e-26	9.13e-27	8.00e-27	2.57e-28
eH($\times 96$)	2.54e-25	6.04e-26	2.19e-26	1.99e-26

(b) Errors obtained with the ZDS scheme.

	R=1	R=2	R=3	R=4
ex($\times 24$)	2.55e-02	2.58e-05	3.10e-08	4.35e-11
ex($\times 96$)	9.97e-05	6.33e-09	4.76e-13	4.22e-17
eH($\times 24$)	1.01e-25	4.26e-26	7.99e-26	3.28e-26
eH($\times 96$)	6.61e-25	2.17e-25	4.49e-25	1.84e-26

(c) Errors obtained with the classical schemes.

	MA2	CS4	KL6	KL8
ex($\times 24$)	7.51e+00	1.36e-03	9.98e-05	4.06e-09
ex($\times 96$)	1.73e+00	5.29e-06	2.44e-08	1.06e-11
eH($\times 24$)	6.57e-05	6.06e-08	5.56e-10	1.98e-14
eH($\times 96$)	1.02e-06	2.33e-10	1.35e-13	3.03e-19

The solution is a periodic solution Beléndez et al. [2007] with a period given by the elliptic integral

$$T_p = 4\sqrt{\frac{\ell}{mg}} \int_0^{\frac{\pi}{2}} \frac{du}{\sqrt{1-\omega^2 \sin^2(u)}},$$

with $\omega = \sin(\pi/8)$. Then, differentiating the first physical equation (26) with respect to time provides relations involving the second-order derivative, namely

$$\ddot{x} = \frac{\dot{p}}{m}, \qquad \ddot{p} = -mg\ell \cos(x)\dot{x}.$$

We then deduce the first set of physical equations PE[1], given by

$$D\mathrm{x} = Z\mathrm{p}/m, \qquad D\mathrm{p} = -mg\ell \sin(Z\mathrm{x}),$$

whereas the second set of physical equations PE[2] reads

$$S\mathrm{x} = D\mathrm{p}/m, \qquad S\mathrm{p} = -mg\ell \cos(Z\mathrm{x}) D\mathrm{x}.$$

5.3.2 Numerical tests

The numerical applications have been carried out with $m = 1$, $g = 1$, $\ell = 1$, and the initial conditions $x(0) = \pi/4$ and $p(0) = 0$. A numerical approximation of the period is $T_p = 6.53$. In Beléndez et al. [2007], an analytical solution is derived, whose expression, not reported here, involves a Jacobi elliptic function. The exact position and momentum at final time $T = 100$ are given by

$$x(t = 100) = -0.2633498226088722 \qquad \text{and} \qquad p(t = 100) = -0.7189111241830892$$

in double precision (since the Jacobi functions are not implemented with arbitrary precision). Hence, schemes with very high accuracy will reach machine error with a comparatively low number of points.

Position errors, together with the deviation of the Hamiltonian are reported[3] in Tables 6 and 7 for $T = 100$s. The deviation of $\mathcal{H}$ strongly depends on the order of the method, as well as on the time step. The ZD scheme produces larger errors, about two orders of magnitudes larger, than the ZDS scheme for the same rate of convergence. The KL6 and ZDS schemes with $R = 2$ present the same amount of error, whereas the KL8 scheme provides the best accuracy compared to its competitor ZDS with $R = 3$. We recover the same accuracy by using the ZDS scheme with $R = 4$. This is one of the main advantages of the ZDS scheme: the order is easily increased, simply by increasing the block size. In the present case, we have a substantial gain of three orders of magnitude between $R = 3$ and $R = 4$.

[3]The symbol *** indicates that machine precision has been reached.

Table 6: Pendulum system from Section 5.3: error on the position at $T = 100$s.

(a) Errors obtained with the ZD scheme.

	R=2		R=4		R=6		R=8	
N	ex	ordx	ex	ordx	ex	ordx	ex	ordx
120	8.27e-01	—	3.32e-01	—	3.25e-02	—	—	—
240	3.36e-02	4.6	1.11e-02	4.9	2.24e-04	7.2	1.32e-02	—
480	2.45e-03	3.8	5.56e-05	7.6	4.85e-06	5.5	6.89e-06	10.9
960	1.58e-04	4.0	8.81e-07	6.0	1.05e-08	8.8	4.92e-09	10.5
1920	9.80e-06	4.0	1.43e-08	5.9	4.27e-11	7.9	7.98e-13	12.6

(b) Errors obtained with the ZDS scheme.

	R=1		R=2		R=3		R=4	
N	ex	ordx	ex	ordx	ex	ordx	ex	ordx
120	3.93e-02	—	1.26e-02	—	2.00e-03	—	—	—
240	2.63e-03	3.9	2.16e-05	9.2	6.24e-06	8.3	1.58e-05	—
480	1.66e-04	4.0	4.35e-07	5.6	1.94e-09	11.7	3.32e-09	12.2
960	1.04e-05	4.0	6.93e-09	6.0	6.25e-12	8.3	1.21e-13	14.7
1920	6.52e-07	4.0	1.09e-10	6.0	4.02e-14	7.3	***	***

(c) Errors obtained with the classical schemes.

	MA2		CS4		KL6		KL8	
N	ex	ordx	ex	ordx	ex	ordx	ex	ordx
120	8.60e-01	—	3.26e-04	—	3.51e-03	—	6.71e-06	—
240	2.20e-01	2.0	3.89e-05	3.1	6.16e-05	5.8	1.98e-08	8.4
480	5.47e-02	2.0	4.84e-06	3.0	9.75e-07	6.0	7.62e-11	8.0
960	1.36e-02	2.0	6.05e-07	3.0	1.53e-08	6.0	3.02e-13	8.0
1920	3.41e-03	2.0	7.59e-08	3.0	2.40e-10	6.0	***	***

We carried out the simulation to $T = 100\,000$ seconds to check the invariance of the Hamiltonian for a very long-time simulation. We use both a coarse grid with $N = 3T$ (an intermediate between $N = 2.4T$ and $N = 4.8T$ we used for the case $T = 100$), which corresponds to about 20 point for a full revolution $T_p \approx 6.53$, and a finer grid with $N = 12T$ points, which makes it possible to compare the deviation of $\mathcal{H}$ with respect to the time step. We report in Table 8 the errors for the ZD, ZDS and classical schemes, respectively. The errors are in line with the case $T = 100$ and the orders are optimal ones.

We confirm in Figure 2 the strict invariance in time of the errors. We also plot, in the bottom right panel, several examples of non-symplectic schemes (classical Runge-Kutta methods, see e.g. Hairer et al. [2006]) to highlight their major drawback, namely a linear increase in the error over time.

5.4 Two-dimensional Kepler problem

The two-dimensional Kepler system consists in a particle moving around a fixed point. It is characterized by the Hamiltonian

$$\mathcal{H}(\boldsymbol{x}, \boldsymbol{p}) = \frac{\|\boldsymbol{p}\|^2}{2} - \frac{1}{\|\boldsymbol{x}\|}, \tag{27}$$

corresponding to the total mechanical energy. The particle position $\boldsymbol{x}(t) \in \mathbb{R}^2$ and the momentum $\boldsymbol{p}(t) \in \mathbb{R}^2$ describe a quadratic curve (elliptic, parabolic or hyperbolic) depending on the initial total energy.

Table 7: Pendulum system from Section 5.3: error on the Hamiltonian at T=100s.

(a) Errors obtained with the ZD scheme.

	R=2		R=4		R=6		R=8	
N	eH	ordH	eH	ordH	eH	ordH	eH	ordH
120	1.66e-01	—	2.00e-01	—	5.30e-02	—	—	—
240	2.10e-02	3.0	6.23e-03	5.0	6.12e-04	6.4	8.57e-03	—
480	1.49e-03	3.8	6.71e-06	6.5	3.78e-06	7.3	2.60e-06	11.
960	9.56e-05	4.0	9.98e-08	6.0	6.52e-09	9.2	3.30e-09	9.6
1920	6.00e-06	4.0	1.51e-09	6.0	2.34e-11	8.1	3.91e-13	9.7

(b) Errors obtained with the ZDS scheme.

	R=1		R=2		R=3		R=4	
N	eH	ordH	eH	ordH	eH	ordH	eH	ordH
120	1.87e-03	—	9.90e-03	—	9.41e-04	—	—	—
240	1.15e-04	4.0	1.57e-05	9.3	3.96e-06	7.9	1.20e-05	—
480	7.21e-06	4.0	2.17e-07	6.1	9.55e-09	8.7	2.24e-09	12.
960	4.51e-07	4.0	3.33e-09	6.0	3.25e-11	8.2	5.14e-13	12.
1920	2.82e-08	4.0	5.18e-11	6.0	1.24e-13	8.0	4.60e-16	10.

(c) Errors obtained with the classical schemes.

	MA2		CS4		KL6		KL8	
N	ex	ordx	ex	ordx	ex	ordx	ex	ordx
120	1.40e-02	—	4.61e-04	—	4.76e-05	—	1.36e-06	—
240	1.71e-03	3.0	4.24e-05	3.4	5.59e-07	6.4	4.26e-09	8.3
480	2.45e-04	2.8	4.63e-06	3.2	8.15e-09	6.1	1.61e-11	8.0
960	4.20e-05	2.5	5.41e-07	3.1	1.25e-10	6.0	6.26e-14	8.0
1920	8.74e-06	2.3	6.54e-08	3.0	1.95e-12	6.0	2.44e-16	8.0

Table 8: Pendulum system from Section 5.3: error on the position and the Hamiltonian at $T = 100\,000$s. Both rows correspond to $N = T \times 3$ and $N = T \times 12$ respectively.

(a) Errors obtained with the ZD scheme.

	R=2	R=4	R=6	R=8
eH($\times 3$)	9.19e-03	2.51e-04	1.62e-04	9.60e-05
eH($\times 12$)	3.92e-05	2.58e-08	1.04e-09	7.97e-11

(b) Errors obtained with the ZDS scheme.

	R=1	R=2	R=3	R=4
eH($\times 3$)	4.72e-05	3.78e-06	8.72e-07	1.89e-07
eH($\times 12$)	1.85e-07	8.70e-10	5.39e-12	5.28e-14

(c) Errors obtained with the classical schemes.

	MA2	CS4	KL6	KL8
eH($\times 3$)	9.01e-04	2.05e-05	1.42e-07	7.04e-10
eH($\times 12$)	2.48e-05	2.73e-07	3.27e-11	1.05e-14

5.4.1 Physical equations and invariants

We first derive the dynamical system associated to the Hamiltonian (27)

$$\dot{\boldsymbol{x}} = \boldsymbol{p}, \qquad \dot{\boldsymbol{p}} = -\frac{\boldsymbol{x}}{\|\boldsymbol{x}\|^3},$$

and, differentiating with respect to time, we obtain the relations with the second derivatives:

$$\ddot{\boldsymbol{x}} = \dot{\boldsymbol{p}}, \qquad \ddot{\boldsymbol{p}} = -\frac{\boldsymbol{p}}{\|\boldsymbol{x}\|^3} + 3\boldsymbol{x}\frac{\boldsymbol{p}\cdot\boldsymbol{x}}{\|\boldsymbol{x}\|^5}.$$

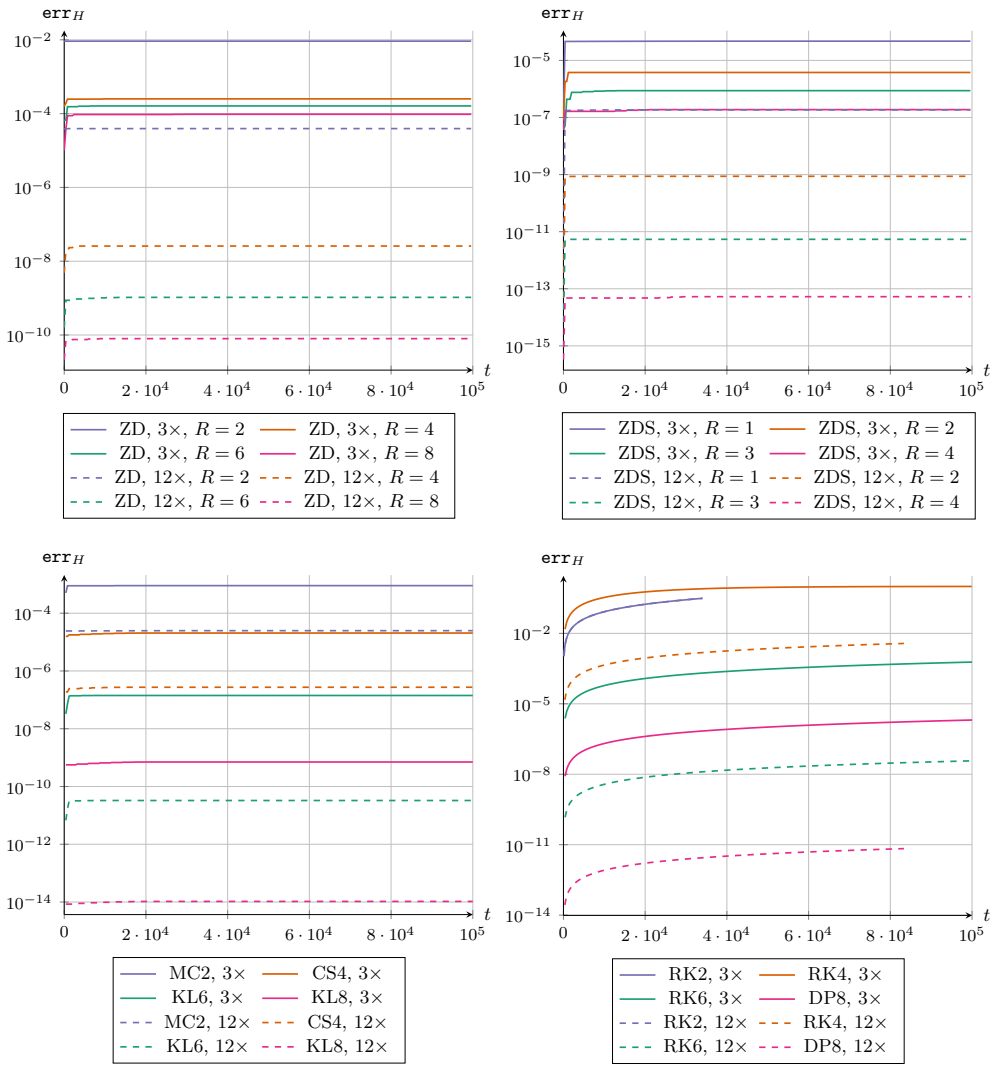

Figure 2: Pendulum problem from Section 5.3: error on the Hamiltonian over time. Top panels: ZD (left) and ZDS (right) methods; bottom panels: classical symplectic (left) and non-symplectic (right) schemes.

We deduce the first group of physical equations PE[1], given by

$$D\mathbf{x} = Z\mathbf{p}, \qquad D\mathbf{p} = -\frac{Z\mathbf{x}}{\|Z\mathbf{x}\|^3},$$

while the second group of physical equation PE[2] reads

$$S\mathbf{x} = D\mathbf{p}, \qquad S\mathbf{p} = -\frac{Z\mathbf{p}}{\|Z\mathbf{x}\|^3} + 3Z\mathbf{x}\frac{Z\mathbf{x}\cdot Z\mathbf{p}}{\|Z\mathbf{x}\|^5}.$$

In addition to the Hamiltonian, there are two other invariant quantities for the Kepler system, namely:

1. the angular momentum
$$\mathcal{L}(\boldsymbol{x},\boldsymbol{p}) = \boldsymbol{x}\times\boldsymbol{p} = p_2x_1 - p_1x_2;$$
2. the Laplace-Runge-Lenz (LRL) vector
$$\mathcal{A}(\boldsymbol{x},\boldsymbol{p}) = \boldsymbol{p}\times\mathcal{L}(\boldsymbol{x},\boldsymbol{p}) - \hat{\boldsymbol{x}}, \qquad \text{with } \hat{\boldsymbol{x}} = \frac{\boldsymbol{x}}{\|\boldsymbol{x}\|},$$
from which we extract a one-dimensional invariant by summing the two components of $\mathcal{A}(\boldsymbol{x},\boldsymbol{p})$, yielding
$$\mathcal{R}(\boldsymbol{x},\boldsymbol{p}) = (p_2x_1 - p_1x_2)(p_2 - p_1) - \frac{x_1+x_2}{\|\boldsymbol{x}\|}.$$

5.4.2 Simulations

Simulations are carried out until a final time $T = 100$ seconds with the initial conditions $\boldsymbol{x}(0) = (0.4, 0)$ and $\boldsymbol{p}(0) = (0, 2)$. Coarse and fine meshes, with $N = 24T$ and $N = 96T$ respectively, are used to check the preservation of the invariants. We denote by `eA` and `eL` stands the errors on the angular momentum and LRL vector.

Table 9: Kepler system from Section 5.4: errors on the Hamiltonian, Angular momentum and Laplace-Runge-Lenz vector at $T = 100$s. For `eH`, `eL` and `eA`, both rows correspond to $N = T \times 24$ and $N = T \times 96$ respectively.

(a) Errors obtained with the `ZD` scheme.

	R=2	R=4	R=6	R=8
eH($\times 24$)	4.15e-05	2.01e-05	1.52e-05	2.52e-05
eH($\times 96$)	1.81e-07	3.83e-09	1.06e-10	4.58e-12
eL($\times 24$)	3.25e-05	6.40e-06	1.41e-05	1.14e-05
eL($\times 96$)	1.28e-07	1.36e-09	3.89e-11	2.45e-12
eA($\times 24$)	4.54e-03	3.83e-04	8.63e-05	4.08e-05
eA($\times 96$)	1.82e-05	1.06e-07	1.29e-09	2.80e-11

(b) Errors obtained with the `ZDS` scheme.

	R=1	R=2	R=3	R=4
eH($\times 24$)	4.83e-05	1.97e-06	3.08e-07	5.96e-08
eH($\times 96$)	1.88e-07	4.47e-10	2.47e-12	2.73e-14
eL($\times 24$)	1.27e-05	4.62e-07	6.02e-08	1.04e-08
eL($\times 96$)	4.93e-08	1.06e-10	5.62e-13	6.02e-15
eA($\times 24$)	3.26e-04	5.07e-06	3.33e-07	5.65e-08
eA($\times 96$)	1.28e-06	1.25e-09	3.52e-12	2.67e-14

(c) Errors obtained with the classical schemes.

	MC2	CS4	KL6	KL8
eH($\times 24$)	3.56e-03	7.37e-05	7.27e-09	3.67e-12
eH($\times 96$)	2.22e-04	1.07e-06	1.73e-12	5.69e-17
eL($\times 24$)	7.95e-76	3.11e-76	7.00e-76	1.91e-75
eL($\times 96$)	7.17e-76	5.96e-76	1.00e-75	1.11e-75
eA($\times 24$)	5.23e-02	1.87e-04	5.13e-07	1.49e-10
eA($\times 96$)	3.16e-03	1.80e-06	1.26e-10	2.29e-15

We report in Table 9 the deviations of the three invariants using the `ZD`, `ZDS` and classical methods, respectively. First, we note that the $\mathcal{H}$ errors are now mesh-dependent, and decrease when the time step decreases. Second, the `ZDS` schemes have smaller error than their equivalent-order `ZD` versions (about 2 orders of magnitude). Comparisons with the classical symplectic schemes show that both the structural methods provide lower errors than the `CS4` scheme. On the contrary, the `KL6` (6th-order) and `KL8` (8th-order) schemes clearly provide the lowest $\mathcal{H}$ errors in comparison with the `ZD` $R = 6$ scheme (6th-order) and the `ZDS` $R = 3$ scheme (8th-order). We report a similar behavior for the LRL $\mathcal{R}$ invariant with a particular mention for the `KL8` method with a noticeable gain of three orders of magnitude regarding the equivalent `ZDS` $R = 3$ structural scheme. Another comment concerns the conservation of the angular momentum $\mathcal{L}$. The structural schemes provide errors with the same magnitude as the Hamiltonian invariant, but the classical schemes definitively deliver a perfect invariance where the errors are only the consequence of the machine errors due to the `quadruple` precision.

We print out in Figure 3 the Hamiltonian invariant (first row), the angular momentum invariant (second row) and the LRL invariant (third row) along time, up to $T = 100$. The columns correspond to the `ZD` scheme (first column), the `ZDS` scheme (second column) and the classical schemes (third column). We display both the errors for the coarse and fine grid. Two main comments arise: after a short transition stage, the Hamiltonian and Angular momentum errors remain constant along the time in all cases; the LRL invariant linearly increases with time, whatever the choice of scheme. Note that the `ZDS` scheme provides the smallest slope, and hence it is a good choice for long simulation time.

5.4.3 Projection on the third invariant manifold

Since the last invariant is not fully preserved, we revisit the scheme by adding a projection onto the third invariant manifold characterized by the equation

$$\mathcal{R}(\boldsymbol{x}, \boldsymbol{p}) = \mathcal{R}(\boldsymbol{x}_0, \boldsymbol{p}_0) = \mathcal{R}_0.$$

In short, we provide new vectors $\widetilde{Z\mathbf{x}}_n$ and $\widetilde{Z\mathbf{p}}_n$ close to $Z\mathbf{x}_n$ and $Z\mathbf{p}_n$ such that $\mathcal{R}(\widetilde{Z\mathbf{x}}_n, \widetilde{Z\mathbf{p}}_n) = \mathcal{R}_0$.

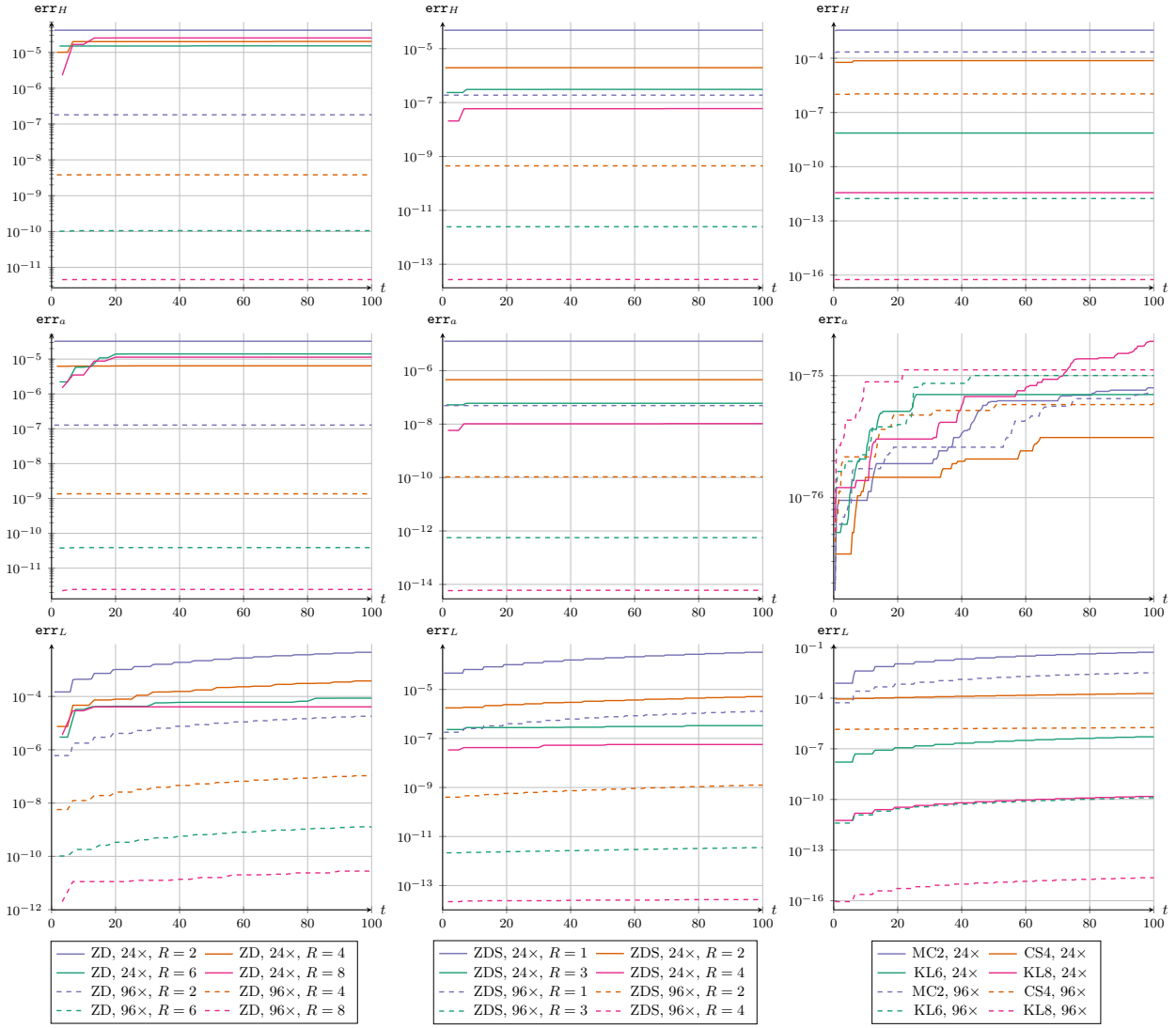

Figure 3: Kepler problem from Section 5.4: errors on the invariants (Hamiltonian, angular momentum and Laplace-Runge-Lenz vector) over time. From left to right: ZD, ZDS and classical methods; top panels: errors on the Hamiltonian; middle panels: errors on the angular momentum; bottom panels: errors on the Laplace-Runge-Lenz vector. In this figure, the results of the uncorrected ZDand ZDSschemes from Section 5.4.2 are presented.

To this end, given $\boldsymbol{x}$ and $\boldsymbol{p}$ assumed to be close to the manifold, we consider the optimization problem: find $\widetilde{\boldsymbol{x}}$ and $\widetilde{\boldsymbol{p}}$ that minimize

$$\mathcal{J} = \frac{1}{2}\left\| [\widetilde{\boldsymbol{x}}, \widetilde{\boldsymbol{p}}]^{\mathrm{t}} - [\boldsymbol{x}, \boldsymbol{p}]^{\mathrm{t}} \right\|^2, \qquad \text{under the constraint } \mathcal{R}(\widetilde{\boldsymbol{x}}, \widetilde{\boldsymbol{p}}) = \mathcal{R}_0.$$

We obtain an update $\widetilde{\boldsymbol{x}}, \widetilde{\boldsymbol{p}}$ given by

$$\begin{bmatrix} \widetilde{x}_1 \\ \widetilde{x}_2 \\ \widetilde{p}_1 \\ \widetilde{p}_2 \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ p_1 \\ p_2 \end{bmatrix} - \lambda \nabla_{x,p} \mathcal{R}(\widetilde{\boldsymbol{x}}, \widetilde{\boldsymbol{p}}), \text{ with } \mathcal{R}(\widetilde{\boldsymbol{x}}, \widetilde{\boldsymbol{p}}) = \mathcal{R}_0,$$

where

$$\nabla_{x,p}\mathcal{R}(\boldsymbol{x},\boldsymbol{p}) = \begin{bmatrix} +p_2(p_2-p_1) - \dfrac{(x_2)^2}{||x||^3} \\ -p_1(p_2-p_1) - \dfrac{(x_1)^2}{||x||^3} \\ 2x_2p_1 - x_1p_2 - x_2p_2 \\ 2x_1p_2 - x_1p_1 - x_2p_1 \end{bmatrix}.$$

We then use the following approximation

$$0 = \mathcal{R}\Big(\boldsymbol{x} - \lambda\nabla_x\mathcal{R}(\widetilde{\boldsymbol{x}},\widetilde{\boldsymbol{p}})\,,\,\boldsymbol{p} - \lambda\nabla_p\mathcal{R}(\widetilde{\boldsymbol{x}},\widetilde{\boldsymbol{p}})\Big) - \mathcal{R}_0 \approx \mathcal{R}(\boldsymbol{x},\boldsymbol{p}) - \mathcal{R}_0 - \lambda\Big\langle\nabla_{x,p}\mathcal{R}(\widetilde{\boldsymbol{x}},\widetilde{\boldsymbol{p}}),\nabla_{x,p}\mathcal{R}(\boldsymbol{x},\boldsymbol{p})\Big\rangle.$$

Substituting $\nabla_{x,p}\mathcal{R}(\widetilde{\boldsymbol{x}},\widetilde{\boldsymbol{p}})$ with $\nabla_{x,p}\mathcal{R}(\boldsymbol{x},\boldsymbol{p})$ in the inner product, and we get the approximation

$$\lambda = \frac{\mathcal{R}(\boldsymbol{x},\boldsymbol{p}) - \mathcal{R}_0}{\|\nabla_{x,p}\mathcal{R}(\boldsymbol{x},\boldsymbol{p})\|^2}.$$

Thus, the approximation of the projection onto the third manifold is given by

$$\begin{bmatrix}\widetilde{\boldsymbol{x}}\\ \widetilde{\boldsymbol{p}}\end{bmatrix} = \begin{bmatrix}\boldsymbol{x}\\ \boldsymbol{p}\end{bmatrix} - \frac{\mathcal{R}(\boldsymbol{x},\boldsymbol{p}) - \mathcal{R}_0}{\|\nabla_{x,p}\mathcal{R}(\boldsymbol{x},\boldsymbol{p})\|^2}\begin{bmatrix}\nabla_x\mathcal{R}(\boldsymbol{x},\boldsymbol{p})\\ \nabla_p\mathcal{R}(\boldsymbol{x},\boldsymbol{p})\end{bmatrix}.$$

We report in Table 10 the deviation of the invariants at the final time $T = 100$ using the projection. We obtain very similar errors to the non-projection case given in Tables 9a and 9b. The correction does not bring significant change in the error magnitude for the ZDS method. We just observe a slight reduction of the LRL error and larger errors for the two other invariants for the ZD method.

Table 10: Kepler system from Section 5.4, with projection on the invariant LRL manifold: errors on the Hamiltonian, Angular momentum and Laplace-Runge-Lenz vector at $T = 100$s. For eH, eL and eA, both rows correspond to $N = T \times 24$ and $N = T \times 96$ respectively.

(a) Errors obtained with the ZD scheme.

	R=2	R=4	R=6	R=8
eH($\times 24$)	3.75e-03	3.88e-04	1.19e-04	2.16e-05
eH($\times 96$)	8.78e-05	1.89e-07	4.21e-09	1.13e-10
eL($\times 24$)	1.40e-03	5.68e-05	3.38e-05	1.48e-05
eL($\times 96$)	5.16e-05	8.48e-08	1.41e-09	4.19e-11
eA($\times 24$)	1.39e-04	3.70e-05	2.15e-05	2.66e-05
eA($\times 96$)	2.97e-06	9.69e-09	3.95e-11	3.70e-12

(b) Errors obtained with the ZDS scheme.

	R=1	R=2	R=3	R=4
eH($\times 24$)	3.17e-03	1.66e-05	2.07e-07	9.13e-08
eH($\times 96$)	1.25e-03	5.01e-08	1.57e-10	6.59e-13
eL($\times 24$)	1.28e-03	2.70e-06	2.27e-07	2.25e-08
eL($\times 96$)	7.22e-04	2.78e-08	7.71e-11	2.88e-13
eA($\times 24$)	9.87e-05	6.85e-07	9.61e-08	4.61e-08
eA($\times 96$)	2.45e-05	1.37e-09	4.71e-12	1.93e-14

Figure 4 shows the evolution of the three invariants with respect to time, up to $T = 100$. We observe two significant changes: the error of the third invariant is constant in time after a fast transition, whereas the other two invariants are now increasing. Therefore, we manage to bound the LRL invariant error along with time thanks to the projection, but the price is a degradation of the other two invariants (roughly, a linear growth of the error over time). Such behavior was also observed in e.g. Andrews and Farrell [2024].

5.5 Two-dimensional three-Body problem: the figure-eight solution

The K-body system is characterized by the position $X[k] = \boldsymbol{x}^k \in \mathbb{R}^2$ and the momentum $P[k] = \boldsymbol{p}^k = m_k\boldsymbol{v}^k \in \mathbb{R}^2$ of each body, where matrices $X, P \in \mathfrak{B}$ are given by $X_{i,k} = \boldsymbol{x}_i^k$ and $P_{i,k} = \boldsymbol{p}_i^k$, for all $k \in \{1,\dots,K\}$ and $i \in \{1,\dots,I\}$. The Hamiltonian corresponds to the time-invariant total mechanical energy

$$\mathcal{H}(X,P) = \sum_{k=1}^{K}\frac{1}{2m_k}\|\boldsymbol{p}^k\|^2 - \sum_{k\neq\ell}\frac{Gm_km_\ell}{2\|\boldsymbol{x}^k - \boldsymbol{x}^\ell\|}. \tag{28}$$

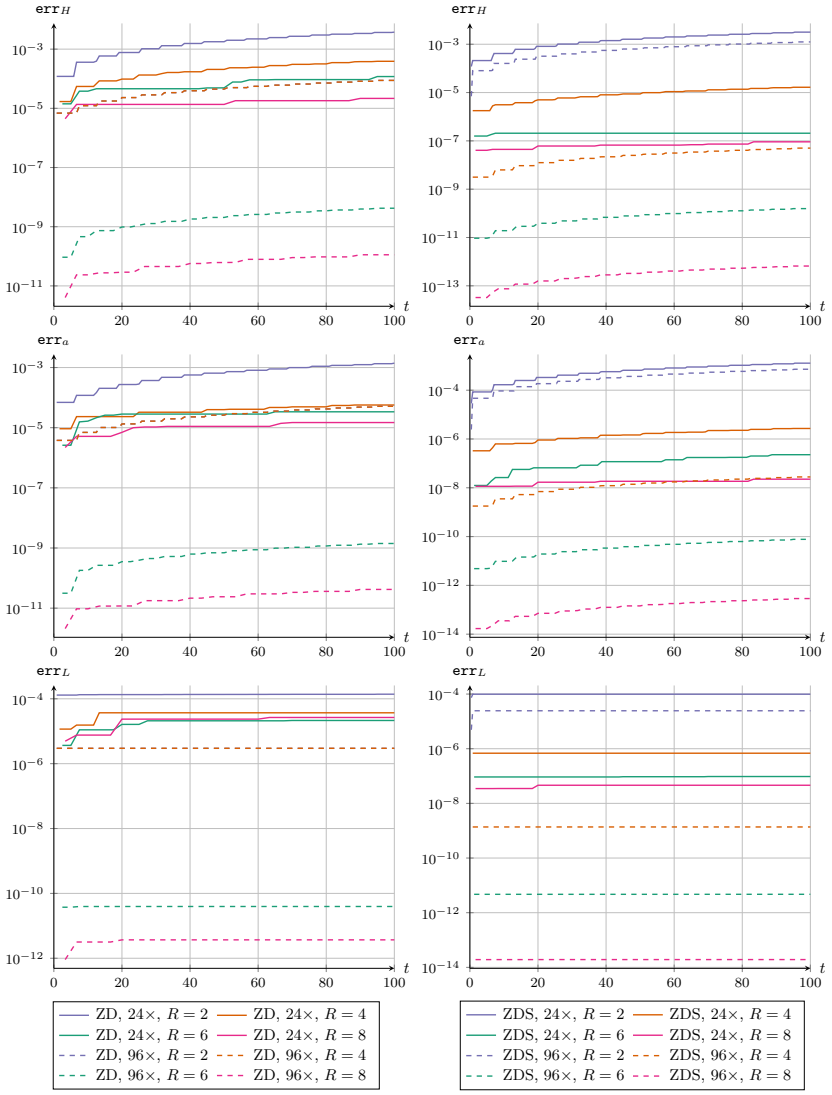

Figure 4: Kepler problem from Section 5.4: errors on the invariants (Hamiltonian, angular momentum and Laplace-Runge-Lenz vector) over time. From left to right: ZDand ZDS methods; top panels: errors on the Hamiltonian; middle panels: errors on the angular momentum; bottom panels: errors on the Laplace-Runge-Lenz vector. In this figure, the results of the projected ZD and ZDS schemes from Section 5.4.3 are presented.

5.5.1 Physical equations and invariants

Hamilton's equations $\dot{X} = \nabla_P \mathcal{H}(X,P)$ and $\dot{P} = -\nabla_X \mathcal{H}(X,P)$ give the first set of physical equations:

$$\text{for all } k \in \{1,\dots,K\}, \qquad \dot{\boldsymbol{x}}^k = \frac{\boldsymbol{p}^k}{m_k}, \qquad \dot{\boldsymbol{p}}^k = -\sum_{\ell \neq k} G m_k m_\ell \frac{\boldsymbol{x}^k - \boldsymbol{x}^\ell}{\|\boldsymbol{x}^k - \boldsymbol{x}^\ell\|^3}.$$

Hence, the first set of physical equations PE[1] given by relations (29) – (30) derives from Hamilton's principle:

$$\boldsymbol{D}\mathbf{x}[k] = \frac{\boldsymbol{Z}\mathbf{p}[k]}{m_k}, \tag{29}$$

$$\boldsymbol{D}\mathbf{p}[k] = -\sum_{\ell \neq k} G m_k m_\ell \frac{\boldsymbol{Z}\mathbf{x}[\ell] - \boldsymbol{Z}\mathbf{x}[k]}{\left\|\boldsymbol{Z}\mathbf{x}[\ell] - \boldsymbol{Z}\mathbf{x}[k]\right\|^3}. \tag{30}$$

Differentiating the dynamical system once more provides the second set of physical equations PE[2], given by relations (31) – (32):

$$\boldsymbol{S}\mathbf{p}[k] = \frac{\boldsymbol{D}\mathbf{p}[k]}{m_k}, \tag{31}$$

$$\boldsymbol{S}\mathbf{p}[k] = -\sum_{\ell \neq k} G m_k m_\ell \left(\frac{\boldsymbol{D}\mathbf{x}[\ell] - \boldsymbol{D}\mathbf{x}[k]}{\left\| \boldsymbol{Z}\mathbf{x}[\ell] - \boldsymbol{Z}\mathbf{x}[k] \right\|^3} - 3 \left\langle \boldsymbol{Z}\mathbf{x}[k] - \boldsymbol{Z}\mathbf{x}[\ell], \boldsymbol{D}\mathbf{x}[k] - \boldsymbol{D}\mathbf{x}[\ell] \right\rangle \frac{\boldsymbol{Z}\mathbf{x}[k] - \boldsymbol{Z}\mathbf{x}[\ell]}{\left\| \boldsymbol{Z}\mathbf{x}[k] - \boldsymbol{Z}\mathbf{x}[\ell] \right\|^5} \right), \tag{32}$$

with $\langle \cdot, \cdot \rangle$ the usual inner product, while $\boldsymbol{Z}\mathbf{x}[k]$, $\boldsymbol{D}\mathbf{x}[k]$ and $\boldsymbol{S}\mathbf{x}[k]$ represent the position, the velocity and the acceleration of body k. A second invariant of the K-body system is the angular momentum $\mathcal{L}(X, P)$, given by

$$\mathcal{L}(X, P) = \sum_{k=1}^{K} \mathcal{L}_k(X[k], P[k]) = \sum_{k=1}^{K} \boldsymbol{x}^k \times \boldsymbol{p}^k.$$

5.5.2 Numerical tests

We choose the figure-eight orbit [4,5,6] for the three-body problem ($K = 3$) given by Moore [1993]. To simplify the problem, we assume that all masses are equal to 1 and take the gravity constant $G = 1$. The initial positions are

$$x^1 = \begin{pmatrix} 0.97000436 \\ -0.24308753 \end{pmatrix}, \quad x^2 \begin{pmatrix} -0.97000436 \\ 0.24308753 \end{pmatrix}, \quad x^3 = \begin{pmatrix} 0 \\ 0 \end{pmatrix},$$

while the initial momenta are

$$\begin{pmatrix} 0.466203685 \\ 0.43236573 \end{pmatrix}, \quad \begin{pmatrix} 0.466203685 \\ 0.43236573 \end{pmatrix}, \quad \begin{pmatrix} -0.93240737 \\ -0.86473146 \end{pmatrix},$$

leading to a periodic motion, with period $T_p = 6.32591401228$.

We report in Table 11 the deviation of both invariants at two different final times using the ZD, ZDS and classical schemes, respectively. The conclusions are very similar to the Kepler case, namely the ZDS method shows an improvement of three orders of magnitude regarding its equivalent (in order) ZD method. The angular momentum reaches machine epsilon when using the classical schemes. Nevertheless, for this more complex case, we observe that, at equal convergence order, the ZDS scheme with $R = 2$ (sixth-order accurate) is much more accurate than its classical equivalent KL6, and the ZDS with $R = 3$ (eight-order accurate) reaches an improvement of three orders of magnitude with respect to its equivalent KL8. Finally, we highlight that the schemes orders are the optimal ones.

Furthermore, we display in Figure 5 the errors on the Hamiltonian (first row) and on the angular momentum (second row) for the three schemes (ZD on the left, ZDS in the middle, classical schemes on the right). Once again, the deviations of the invariants are bounded in time after a short transition stage. The classical symplectic schemes provide a prefect preservation of the angular momentum (up to the `quadruple` accuracy), whereas the structural schemes provide an error in line with the method order.

5.6 The outer Solar system

We revisit the K-body Hamiltonian (28) for three-dimensional trajectories ($I = 3$). The 6-body system we consider corresponds to the so-called outer solar system: the Sun and Jupiter, Saturn, Uranus, Neptune and Pluto Hairer et al. [2006]. The masses and initial conditions are given [7,8] in Table 12. The initial positions are given in Astronomical Units (1 au=149 597 870 km) and the initial velocities are given in au per earth day. The gravity constant is set to $G = 2.95912208286 \cdot 10^{-4}$. Together with the Hamiltonian, we analyze the angular momentum, another quantity that is time-invariant, given by the vector

$$\boldsymbol{a} = \mathcal{L}(X, P) = \sum_{k=1}^{K} \mathcal{L}_k(X[k], P[k]) = \sum_{k=1}^{K} \boldsymbol{x}^k \times \boldsymbol{p}^k.$$

[4] https://www.ams.org/notices/200105/fea-montgomery.pdf
[5] https://www.math.uni-bielefeld.de/~rehmann/ECM/cdrom/3ecm/pdfs/pant3/simo.pdf
[6] https://perso.imcce.fr/alain-chenciner/huit.pdf
[7] https://dspace.mit.edu/bitstream/handle/1721.1/6442/AIM-877.pdf
[8] https://rebound.readthedocs.io/en/latest/c_examples/outer_solar_system/

Table 11: 2D 3-body system from Section 5.5: errors on the Hamiltonian and angular momentum at $T = 10$s (left subtables) and $T = 1000$s (right subtables). For eH and eL, both rows correspond to $N = T \times 12$ and $N = T \times 48$ respectively.

(a) ZD scheme, $T = 10$s.

	R=2	R=4	R=6	R=8
eH($\times 12$)	5.05e-05	6.88e-05	4.20e-05	9.16e-05
eH($\times 48$)	1.60e-07	9.82e-09	4.10e-10	3.52e-11
eL($\times 12$)	7.10e-05	2.30e-05	1.75e-05	4.85e-05
eL($\times 48$)	2.79e-07	4.90e-09	1.84e-10	1.53e-11

(b) ZD scheme, $T = 1000$s.

	R=2	R=4	R=6	R=8
eH($\times 12$)	5.14e-05	6.93e-05	2.50e-04	9.26e-05
eH($\times 48$)	1.60e-07	9.82e-09	4.10e-10	3.52e-11
eL($\times 12$)	7.16e-05	2.46e-05	2.03e-05	9.26e-05
eL($\times 48$)	2.79e-07	4.90e-09	1.84e-10	1.59e-11

(c) ZDS scheme, $T = 10$s.

	R=1	R=2	R=3	R=4
eH($\times 12$)	7.67e-05	3.62e-06	1.10e-06	8.74e-07
eH($\times 48$)	2.98e-07	8.14e-10	5.41e-12	8.55e-14
eL($\times 12$)	2.86e-05	1.25e-06	1.68e-07	8.17e-08
eL($\times 48$)	1.11e-07	2.83e-10	1.98e-12	2.92e-14

(d) ZDS scheme, $T = 1000$s.

	R=1	R=2	R=3	R=4
eH($\times 12$)	7.68e-05	3.69e-06	1.78e-06	8.75e-07
eH($\times 48$)	2.98e-07	8.14e-10	5.42e-12	8.55e-14
eL($\times 12$)	2.86e-05	1.27e-06	1.68e-07	8.71e-08
eL($\times 48$)	1.11e-07	2.83e-10	1.98e-12	2.92e-14

(e) Classical schemes, $T = 10$s.

	MA2	CS4	KL6	KL8
eH($\times 12$)	1.41e-01	1.37e-02	5.06e-04	1.79e-05
eH($\times 48$)	5.71e-03	2.01e-04	1.06e-07	2.62e-10
eL($\times 12$)	2.07e-76	4.84e-76	5.35e-76	4.32e-76
eL($\times 48$)	3.63e-76	1.11e-75	7.60e-76	1.07e-75

(f) Classical schemes, $T = 1000$s.

	MA2	CS4	KL6	KL8
eH($\times 12$)	3.97e-01	6.97e-02	1.16e-03	3.61e-05
eH($\times 48$)	5.71e-03	2.01e-04	1.06e-07	2.62e-10
eL($\times 12$)	1.19e-73	1.12e-73	1.89e-73	1.98e-73
eL($\times 48$)	1.38e-73	1.84e-73	5.36e-73	7.53e-73

We provide in Table 13 the maximum errors for the Hamiltonian and Angular momentum using the ZD, ZDS and classical schemes, with coarse and fine grids, and $T = 100\,000$ years. In each situation, we obtain the optimal order for the ZD, ZDS and classical methods, noting that we reach the `quadruple` precision for the angular momentum with the symplectic scheme. We also observe the same gain of two or three orders of magnitude between the ZD and ZDS schemes. The KL6 scheme (6th-order accurate) provides very similar deviations compared to its equivalent ZDS with $R = 2$, while the KL8 scheme (8th-order accurate) is comparable to the ZDS with $R = 3$. The main advantage of the structural scheme is its ability to choose or adapt the order without modifying its implementation, only by changing the value of the block size R.

We plot in Figure 6 the variations of the two invariants (Hamiltonian in the first row, angular momentum in the second one) with respect to time, and with the ZD (left column), ZDS (middle column) and classical schemes (right column). Clearly, the Hamiltonian invariant is bounded in time, and, after a short transition, the error is constant. We observe the same behavior for the angular momentum for the ZD and ZDS methods, while the classical methods reach the `quadruple` precision.

5.7 Motion of a particle in a 3D electromagnetic field

We consider a charged particle subjected to electromagnetic forces, characterized by the electric potential $\varphi = \varphi(\boldsymbol{x}) \in \mathbb{R}$ and the magnetic potential $A = A(\boldsymbol{x}) \in \mathbb{R}^3$. This situation is governed by the following Hamiltonian:

$$\mathcal{H}(\boldsymbol{x}, \boldsymbol{p}) = \frac{1}{2m}\|\boldsymbol{p} - eA(\boldsymbol{x})\|^2 + e\varphi(\boldsymbol{x}),$$

with m and e the mass and electric charge of the particle. We denote by $A(\boldsymbol{x}) = [A_1, A_2, A_3]^{\mathrm{t}}$ the components of the magnetic potential.

This is an example of a non-separable Hamiltonian, i.e. a problem where $\nabla_{\boldsymbol{p}}\mathcal{H}$ depends on both $\boldsymbol{x}$ and $\boldsymbol{p}$. The classical schemes already used in the previous section and implemented in `julia` failed to solve such a problem. Hence, we have implemented the second-order symplectic Strömer-Verlet schemes Hairer et al. [2003]. The composition method

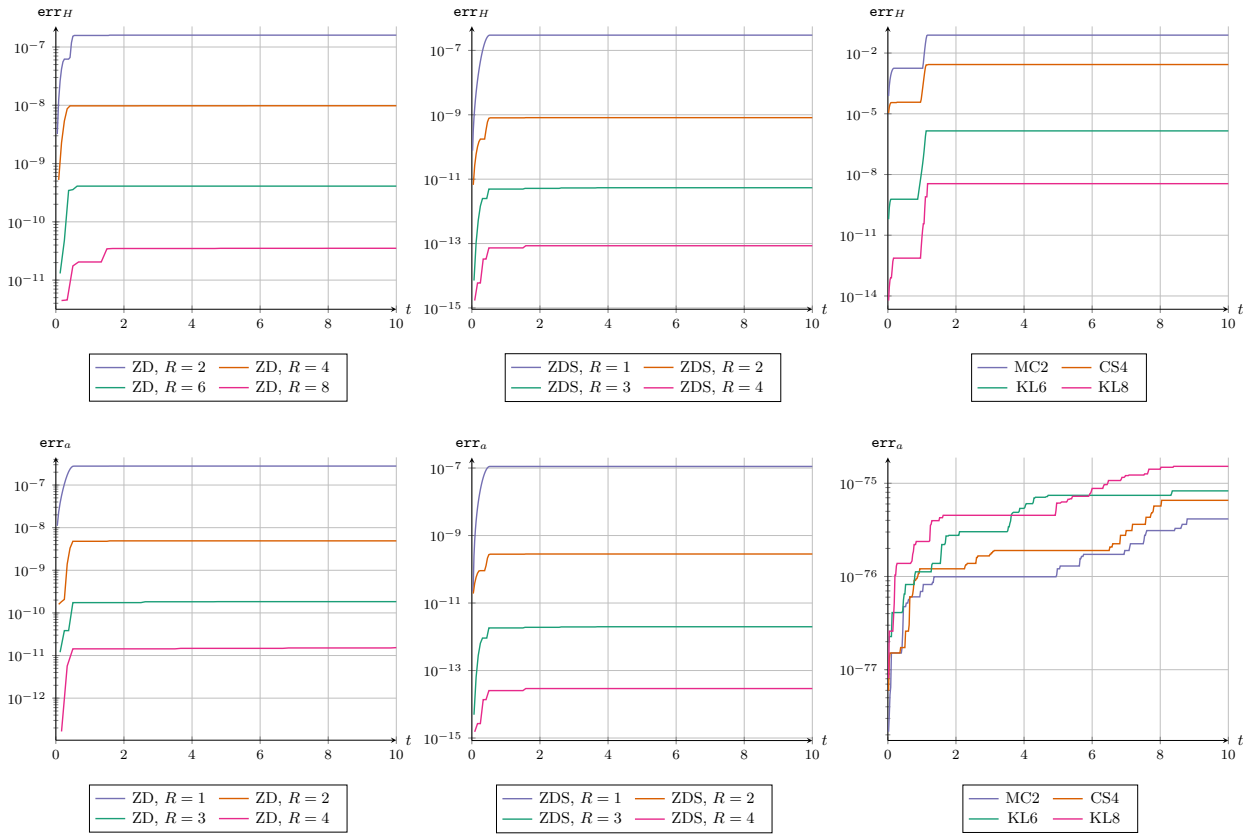

Figure 5: Two-dimensional three-body problem from Section 5.5: errors on the invariants (Hamiltonian and angular momentum) over time. From left to right: ZD, ZDS and classical methods; top panels: errors on the Hamiltonian; bottom panels: errors on the angular momentum.

Hairer et al. [2006] make it possible to reach the fourth, sixth and eighth orders of accuracy, to provide comparisons with the structural method.

5.7.1 Physical equations

From the Hamiltonian, we derive the equations of motion given by

$$\dot{\boldsymbol{x}} = \partial_p \mathcal{H}(\boldsymbol{x}, \boldsymbol{p}) = \frac{1}{m}(\boldsymbol{p} - eA), \tag{33}$$

$$\begin{aligned}\dot{\boldsymbol{p}} &= -\partial_x \mathcal{H}(\boldsymbol{x}, \boldsymbol{p}) = \frac{e}{m}\left[\partial_x A\right]^{\mathrm{t}}(\boldsymbol{p} - eA) - e\partial_x \varphi \\ &= e\left(\left[\partial_x A\right]^{\mathrm{t}} \frac{dx}{dt} - \partial_x \varphi\right),\end{aligned} \tag{34}$$

with the matrix $\partial_x A(\boldsymbol{x}) = \left[\partial_j A_i(\boldsymbol{x})\right]_{i,j} \in \mathbb{R}^{3\times 3}$ and the vector $\partial_x \varphi(\boldsymbol{x}) = [\partial_1\varphi(\boldsymbol{x}), \partial_2\varphi(\boldsymbol{x}), \partial_3\varphi(\boldsymbol{x})]^{\mathrm{t}} \in \mathbb{R}^3$.

From relations (33) – (34), we deduce the first group of physical equations (PE[1]) connecting the first-order derivatives

$$\begin{aligned} mD\mathbf{x} &= Z\mathbf{p} - eA(Z\mathbf{x}), \\ D\mathbf{p} &= e\left(\left[\partial_x A(Z\mathbf{x})\right]^t D\mathbf{x} - \partial_x \varphi(Z\mathbf{x})\right), \end{aligned}$$

Table 12: Data and initial conditions for the outer solar system problem. Masses are normalized regarding the solar mass, and the initial positions and velocities are given, respectively, in au and au per earth day.

Celestial body	Mass	Initial Position	Initial Velocity	period
Sun	1.00000597682e+00	0 0 0	0 0 0	0
Jupiter	9.547861040430e-04	-3.5023653 -3.8169847 -1.5507963	0.00565429 -0.00412490 -0.00190589	4333
Saturn	2.855837331510e-04	9.0755314 -3.0458353 -1.6483708	0.00168318 0.00483525 0.00192462	10759
Uranus	4.37273164546e-05	8.3101420 -16.2901086 -7.2521278	0.00354178 0.00137102 0.00055029	30687
Neptune	5.17759138449e-05	11.4707666 -25.7294829 -10.8169456	0.00288930 0.00114527 0.00039677	60190
Pluto	(10/13) e-08	-15.5387357 -25.2225594 -3.1902382	0.00276725 -0.00170702 -0.00136504	90560

Table 13: Three-dimensional n-body problem (i.e., the outer Solar system) from Section 5.6: errors on the Hamiltonian and angular momentum at $T = 100\,000$ years. For eH and eL, both rows correspond to $N = 480$ and $N = 1920$ respectively.

(a) Errors obtained with the ZD scheme.

	R=2	R=4	R=6	R=8
eH(480)	3.06e-04	1.07e-04	4.02e-05	3.79e-05
eH(1920)	1.26e-06	2.72e-08	5.11e-10	2.29e-11
eL(480)	9.12e-09	2.42e-09	9.37e-10	8.87e-10
eL(1920)	3.70e-11	6.15e-13	1.08e-14	5.46e-16

(b) Errors obtained with the ZDS scheme.

	R=1	R=2	R=3	R=4
eH(480)	7.92e-05	2.33e-06	9.99e-08	3.48e-08
eH(1920)	3.10e-07	5.69e-10	1.52e-12	1.03e-14
eL(480)	1.76e-09	5.03e-11	2.63e-12	6.56e-13
eL(1920)	6.88e-12	1.23e-14	3.12e-17	2.36e-19

(c) Errors obtained with the classical schemes.

	MA2	CS4	KL6	KL8
eH(480)	1.24e-01	6.26e-03	2.27e-05	3.29e-07
eH(1920)	7.43e-03	1.22e-04	3.55e-09	2.71e-12
eL(480)	7.91e-81	1.90e-80	2.06e-80	2.11e-80
eL(1920)	3.37e-80	2.21e-80	4.53e-80	1.17e-79

Differentiating relations (33) – (34) with respect to time, we get the second group of physical equations

$$\begin{aligned} m\ddot{\boldsymbol{x}} &= \dot{\boldsymbol{p}} - e\partial_x A\dot{\boldsymbol{x}} - e\partial_t A \\ &= e\left(\left[\partial_x A\right]^{\mathrm{t}}\dot{\boldsymbol{x}} - \partial_x\varphi - \partial_x A\dot{\boldsymbol{x}} - \partial_t A\right), \\ \ddot{\boldsymbol{p}} &= e\frac{d}{dt}\left(\left[\partial_x A\right]^{\mathrm{t}}\dot{\boldsymbol{x}}\right) - e\partial_x^2\varphi\,\dot{\boldsymbol{x}} \\ &= e\Big(\left[\partial_x A\right]^{\mathrm{t}}\ddot{\boldsymbol{x}} + w - \partial_x^2\varphi\,\dot{\boldsymbol{x}}\Big), \end{aligned}$$

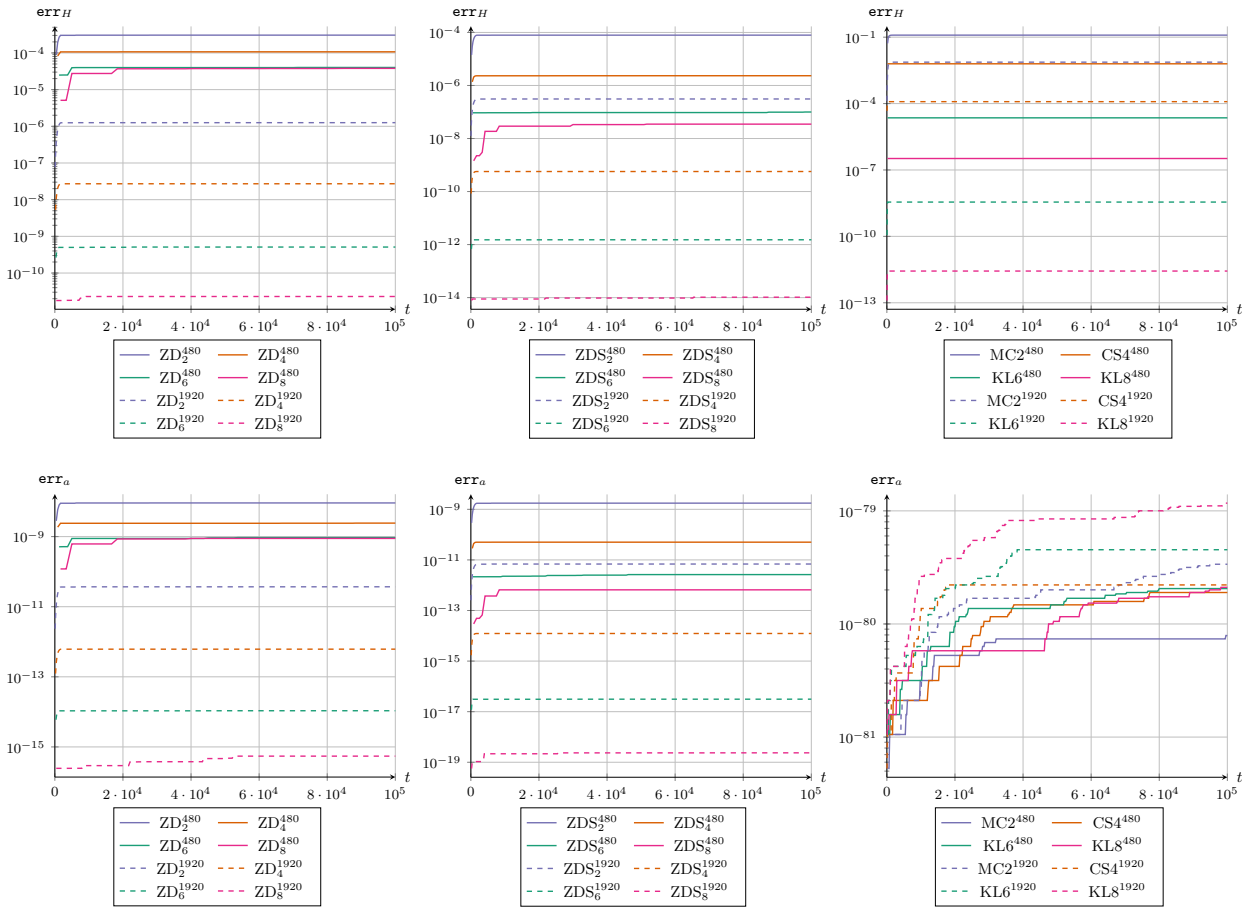

Figure 6: Three-dimensional n-body problem (i.e., the outer Solar system) from Section 5.6: errors on the invariants (Hamiltonian and angular momentum) over time. From left to right: ZD, ZDS and classical methods; top panels: errors on the Hamiltonian; bottom panels: errors on the angular momentum.

with

$$\partial_x^2\varphi = \left[\partial_i\partial_j\varphi\right]_{i,j} \qquad \text{and, for all } \ell \in \{1,2,3\}, w_\ell = \sum_{i=1}^{3}\sum_{j=1}^{3} \partial_\ell\partial_i\Big(A_j\Big)\dot{x}_i\dot{x}_j.$$

The second group of physical equations (PE[2]) then reads

$$\frac{m}{e}S\mathbf{x} = \Big(\left[\partial_x A(Z\mathbf{x})\right]^t - \left[\partial_x A(Z\mathbf{x})\right]\Big)D\mathbf{x} - \partial_t A(Z\mathbf{x}) - \partial_x\phi(Z\mathbf{x}),$$
$$S\mathbf{p} = e\Big(\left[\partial_x A(Z\mathbf{x})\right]S\mathbf{x} + w - \left[\partial_x^2\varphi(Z\mathbf{x})\right]D\mathbf{x}\Big).$$

5.7.2 A Sanity Check Benchmark (SCB)

The SCB consists in considering a pseudo-2D problem with very smooth magnetic and electric potentials. We take

$$\varphi(\boldsymbol{x}) = -\frac{1}{0.1 + \|\boldsymbol{x}\|}, \quad A(\boldsymbol{x}) = \begin{pmatrix} 0 \\ 1000x_1 \\ 0 \end{pmatrix}.$$

The initial conditions are $x(0) = (1,0,0)^{\mathrm{t}}$ and $p(0) = (0, 1+100, 0)^{\mathrm{t}}$, while the physical parameters are $m = 1$ and $q = 1$, with final time $T = 100$.

We have carried out the simulation up to the time $T = 100$ using a coarse grid ($N = 12T$) and finer one ($N = 48T$) to check the order of accuracy and the preservation of the Hamiltonian. We report in Table 14 the Hamiltonian deviation for

the ZD, ZDS and Strömer-Verlet (SV) methods. Clearly, the structural schemes provide a better accuracy in comparison to the SV version. Namely, the ZD methods provide a gain of two orders of magnitude compared to the classical SV methods, while we obtain a gain of four to five orders of magnitude with the ZDS approach. We highlight the quality of the compact second-derivative method, able to reach such very low errors.

Table 14: Particle in an electromagnetic field, Sanity Check Benchmark from Section 5.7.2: error on the Hamiltonian at $T = 100$s. Both rows correspond to $N = 1200$ and $N = 4800$, respectively.

(a) Errors obtained with the ZD scheme.

	R=2	R=4	R=6	R=8
eH($\times 12$)	8.69e-07	2.27e-08	9.53e-10	8.56e-11
eH($\times 48$)	3.41e-09	5.57e-12	1.43e-14	6.96e-17

(b) Errors obtained with the ZDS scheme.

	R=1	R=2	R=3	R=4
eH($\times 12$)	2.89e-07	9.42e-10	7.37e-12	1.18e-13
eH($\times 48$)	1.13e-09	2.29e-13	8.42e-17	1.04e-19

(c) Errors obtained with the classical schemes.

	SV2	SV4	SV6	SV8
eH($\times 12$)	6.10e-04	4.66e-06	1.98e-08	2.07e-09
eH($\times 48$)	3.80e-05	1.82e-08	4.43e-12	3.23e-14

Figure 7 displays the growth of the Hamiltonian error in time for the ZD (left panel), ZDS (middle panel) and SV (right panel) schemes. In each case, the Hamiltonian error becomes constant after a very short growth.

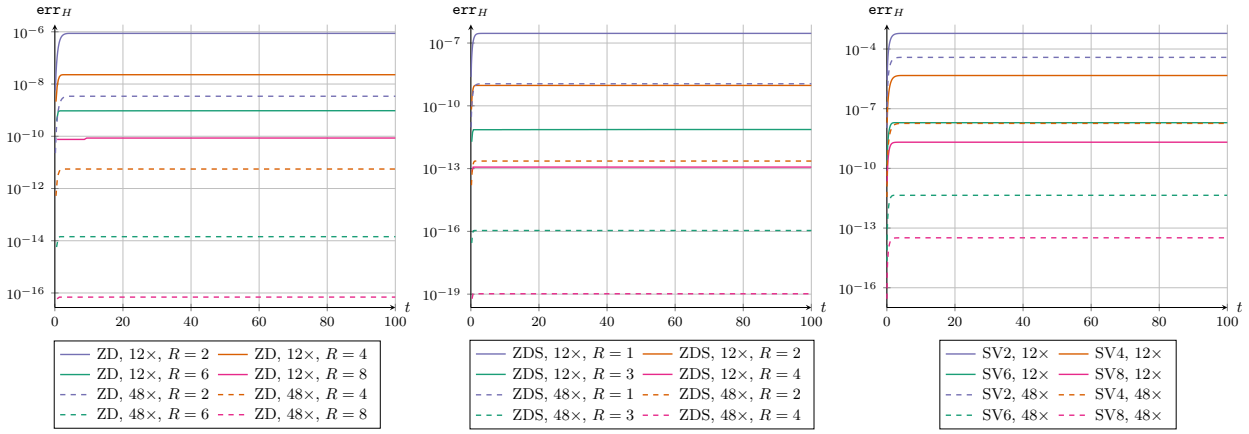

Figure 7: Particle in an electromagnetic field, Sanity Check Benchmark from Section 5.7.2: error on the Hamiltonian over time. From left to right: ZD, ZDS and SV methods.

5.7.3 Non-separable case: particle motion for challenging potentials

We propose a more challenging benchmark, where the magnetic potential presents a singularity at $x_1 = 0$, leading to additional numerical difficulties to preserve the Hamiltonian. The electrostatic potential is given by $\varphi(\boldsymbol{x}) = 2\cos(x_1)^2 + \sin(x_1)^2(\sin(x_2)\cos(x_2) + \sin(x_3)\cos(x_3))$, while the magnetic potential reads

$$A(\boldsymbol{x}) = \begin{pmatrix} r^2 \\ r^2 \frac{x_2}{x_1} \\ -2\log(1+r^2) \end{pmatrix},$$

with $r^2 = x_1^2 + x_2^2 + x_3^2$. The mass and charge are set to unity, i.e., $m = 1$ and $q = 1$, and the final simulation time is $20\,000\,s$. At last, we use the initial conditions $\boldsymbol{x}(0) = (0.5, -0.25, -0.25)^{\mathrm{t}}$ and $\boldsymbol{p}(0) = (0, 0, -1)^{\mathrm{t}}$. Not that this potential leads to a truly non-separable problem.

We run the ZD and ZDS simulation for $N = 12T$ and $N = 48T$ whereas, for stability reason, the SV method required much finer grids with $N = 128T$ and $N = 256T$. The results are reported in Tables 15a to 15c. Once again, we observe the superior advantage of the structural scheme to achieve very high accuracy, even with coarser grids. As usual, we get

a gain of about two orders of magnitude between the ZD and ZDS scheme for a given order of convergence. Moreover, we display in Figure 8 the evolution of the deviation of the theoretically invariant Hamiltonian. On the one hand, the SV method presents a linear growth of the error as a linear function of the time, independently of the method order (almost the same slope in log scale). We observe a similar behaviour for the fourth- and sixth-order structural methods. At last, we notice a dramatic cut of the error growth when dealing with very high orders of accuracy, in particular the ZDS scheme with $R = 4$ shows a horizontal line that indicates the boundedness of the Hamiltonian deviation.

Table 15: Particle in an electromagnetic field, non-separable case from Section 5.7.3: error on the Hamiltonian at $T = 20\,000$ s. For the ZD and ZDS schemes, both rows correspond to $N = 12 \times T$ and $N = 48 \times T$, respectively; for the classical scheme, they correspond to $N = 128 \times T$ and $N = 256 \times T$.

(a) Errors obtained with the ZD scheme.

	R=2	R=4	R=6	R=8
eH($\times 12$)	2.28e-01	1.31e-02	6.00e-04	3.19e-04
eH($\times 48$)	8.45e-04	3.51e-06	1.11e-08	6.18e-11

(b) Errors obtained with the ZDS scheme.

	R=1	R=2	R=3	R=4
eH($\times 12$)	1.34e-02	1.15e-04	7.59e-07	6.44e-07
eH($\times 48$)	5.29e-05	2.95e-08	1.24e-11	2.64e-14

(c) Errors obtained with the classical schemes.

	SV2	SV4	SV6	SV8
eH($\times 128$)	8.97e-02	3.82e-04	5.01e-06	4.90e-08
eH($\times 256$)	4.52e-02	2.41e-05	7.88e-08	1.76e-10

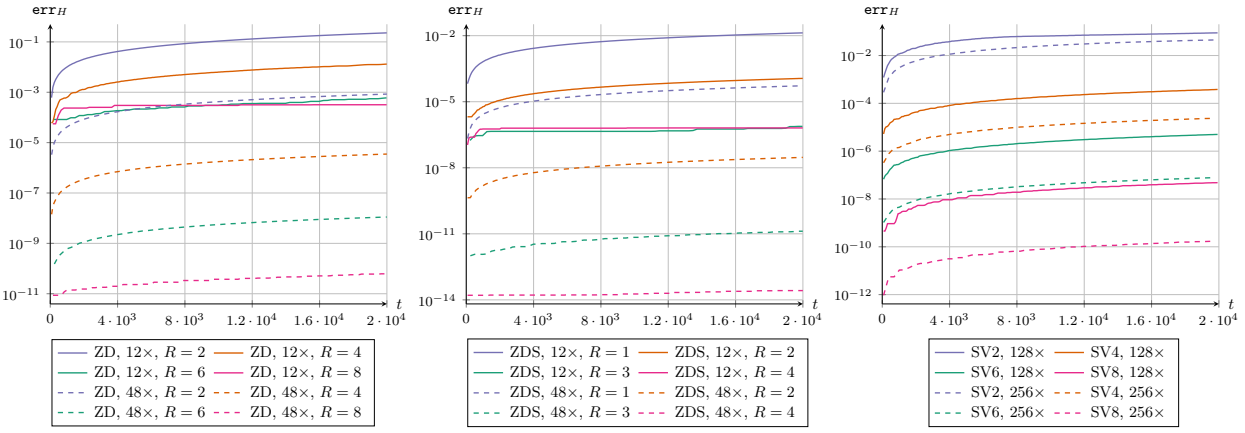

Figure 8: Particle in an electromagnetic field, non-separable case from Section 5.7.3: error on the Hamiltonian over time. From left to right: ZD, ZDS and classical methods.

5.8 Complexity and function calls

In this section, we analyse the complexity and function calls of the structural methods (ZD and ZDS) compared to classical symplectic schemes. We first provide a detailed complexity analysis for both separable and non-separable cases in Section 5.8.1. We then present, in Section 5.8.2, the average number of iterations and function calls required by the fixed-point method used in the structural schemes. This complexity and function call analysis helps to understand the computational efficiency and performance of the proposed methods.

5.8.1 Complexity analysis for a single time step

We start with comparing the methods introduced in this article to classical ones, in terms of algorithmic complexity. We assume that the Hamiltonian system governs the behaviour of K bodies in dimension I: therefore, the system has $n_{\text{eq}} := 2IK$ equations. Moreover, in the structural method (as well as the classical ones in the non-separable case), non-linear equations have to be solved. We denote by n_{it} the number of iterations of the non-linear solver (a fixed-point method in the structural method, and Newton's method for the classical schemes). This number of iterations depends on the method, and we denote by $\mathsf{n}_{\text{it}}^{\text{SV}}$, $\mathsf{n}_{\text{it}}^{\text{ZD}}$ and $\mathsf{n}_{\text{it}}^{\text{ZDS}}$ the number of iterations for the SV, ZD and ZDS methods, respectively.

To compute the complexity, we split the problem into two main configurations: the separable and non-separable cases. We assume that evaluating $\mathcal{H}(X, P)$ and its derivatives has complexity $O(1)$. The complexity of a single time step of each algorithm is reported in Table 16.

Table 16: Complexity of the different schemes used and introduced in this paper.

(a) separable case

Method	order	sub-steps	complexity
MA2	2	3	$\mathrm{O}(3n_{\mathrm{eq}})$
CS4	4	7	$\mathrm{O}(7n_{\mathrm{eq}})$
KL6	6	7 × 3	$\mathrm{O}(21n_{\mathrm{eq}})$
KL8	8	15 × 3	$\mathrm{O}(45n_{\mathrm{eq}})$
ZD	R+2	N/A	$\mathrm{O}\left(2Rn_{\mathrm{eq}}\left(1+n_{\mathrm{it}}^{\mathrm{ZD}}\right)\right)$
ZDS	2(R+1)	N/A	$\mathrm{O}\left(4Rn_{\mathrm{eq}}\left(1+n_{\mathrm{it}}^{\mathrm{ZDS}}\right)\right)$

(b) non-separable case

Method	order	sub-steps	complexity
SV2	2	2	$\mathrm{O}\left(2n_{\mathrm{eq}}\left(1+n_{\mathrm{it}}^{\mathrm{SV}}\right)\right)$
SV4	4	3 × 2	$\mathrm{O}\left(6n_{\mathrm{eq}}\left(1+n_{\mathrm{it}}^{\mathrm{SV}}\right)\right)$
SV6	6	9 × 2	$\mathrm{O}\left(18n_{\mathrm{eq}}\left(1+n_{\mathrm{it}}^{\mathrm{SV}}\right)\right)$
SV8	8	27 × 2	$\mathrm{O}\left(54n_{\mathrm{eq}}\left(1+n_{\mathrm{it}}^{\mathrm{SV}}\right)\right)$
ZD	R+2	N/A	$\mathrm{O}\left(2Rn_{\mathrm{eq}}\left(1+n_{\mathrm{it}}^{\mathrm{ZD}}\right)\right)$
ZDS	2(R+1)	N/A	$\mathrm{O}\left(6Rn_{\mathrm{eq}}\left(1+n_{\mathrm{it}}^{\mathrm{ZDS}}\right)\right)$

In the **separable case**, the complexity of the classical methods only depends on the order of accuracy and system size. On the contrary, for the structural method, the complexity also depends on the number of iterations n_{it} in the fixed-point method, with one additional iteration corresponding to the initialization stage. The ZDS method has twice the complexity of the ZD method, as it requires evaluating the second derivatives of the Hamtilonian in addition to its first derivatives. Therefore, in terms of complexity, the structural methods are slightly less favorable for low orders, but become comparable or even slightly better for very high orders, depending on the number of iterations. These conclusions are summarized in Table 17, where we report the ratio of the complexity of the structural methods with respect to the classical ones.

In the **non-separable case**, the complexity of the structural and classical methods depend on the order, the system size, and the number of iterations. Indeed, in the SV2 method, one has to solve two nonlinear equations and two linear ones per unknown in the system, leading to a complexity in $2n_{\mathrm{eq}}(1 + n_{\mathrm{it}})$. This complexity is then multiplied by a constant depending on the number of compositions, itself depending on the order of the method. For the structural methods, it turns out that the complexity is almost the same as in the separable case. The only difference is that the cross derivatives are now needed in the ZDS method, leading to an extra two evaluations of the Hamiltonian and its derivatives compared to the separable case. Table 17 once again reports the ratio between the complexity of the structural methods and the classical ones.

Table 17: Ratio of the complexity of the ZD and ZDS schemes with the classical ones.

Order	Separable		Non-separable	
	ZD	ZDS	ZD	ZDS
4	$\left(1+n_{\mathrm{it}}^{\mathrm{ZD}}\right) \times 4/7$	$\left(1+n_{\mathrm{it}}^{\mathrm{ZDS}}\right) \times 4/7$	$\left(1+n_{\mathrm{it}}^{\mathrm{ZD}}\right)/\left(1+n_{\mathrm{it}}^{\mathrm{SV}}\right) \times 2/3$	$\left(1+n_{\mathrm{it}}^{\mathrm{ZDS}}\right)/\left(1+n_{\mathrm{it}}^{\mathrm{SV}}\right) \times 1$
6	$\left(1+n_{\mathrm{it}}^{\mathrm{ZD}}\right) \times 8/21$	$\left(1+n_{\mathrm{it}}^{\mathrm{ZDS}}\right) \times 8/21$	$\left(1+n_{\mathrm{it}}^{\mathrm{ZD}}\right)/\left(1+n_{\mathrm{it}}^{\mathrm{SV}}\right) \times 4/9$	$\left(1+n_{\mathrm{it}}^{\mathrm{ZDS}}\right)/\left(1+n_{\mathrm{it}}^{\mathrm{SV}}\right) \times 2/3$
8	$\left(1+n_{\mathrm{it}}^{\mathrm{ZD}}\right) \times 4/15$	$\left(1+n_{\mathrm{it}}^{\mathrm{ZDS}}\right) \times 4/15$	$\left(1+n_{\mathrm{it}}^{\mathrm{ZD}}\right)/\left(1+n_{\mathrm{it}}^{\mathrm{SV}}\right) \times 2/9$	$\left(1+n_{\mathrm{it}}^{\mathrm{ZDS}}\right)/\left(1+n_{\mathrm{it}}^{\mathrm{SV}}\right) \times 1/3$

In summary, in all cases, the gain in complexity by using the structural method depends on the number of iterations of the nonlinear solver. In addition, the number of sub-steps increases linearly with the order of accuracy for the structural method, while this increase is superlinear, or even quadratic, for the classical methods. Therefore, to conclude on the potential gains in computation time secured by our schemes, one has to compare the number of iterations in the nonlinear solver, as well as the total number of time steps. This is easily done through the number of function calls, whose analysis is presented in the next section.

5.8.2 Function calls

On the one hand, the ZD and ZDS schemes solve the system using blocks of size R, while the total number of steps is denoted by N. We simply call `nb_iter_avg` the ratio between the total number of iterations, including the fixed-point solves, and the total number of steps N. Conversely, since one iteration over the block corresponds to R evaluations of the Physical Equations (PE), the average number of "PE function calls" `nb_call_avg` is given by the ratio between the total number of calls to PE and the number of steps N, that is `nb_call_avg = R × nb_iter_avg`. The classical schemes, on the other hand, compose several semi-implicit steps, with one linear and potentially one nonlinear solve for every semi-implicit step. The steps alternate between solving for $\nabla_X \mathcal{H}(X,P)$ and $\nabla_P \mathcal{H}(X,P)$. In the separable case, we have two linear solves; in the non-separable one, a linear and a nonlinear solve are carried out. Note that, in the specific case of the separable particle problem from Section 5.7.3, $\nabla_P \mathcal{H}(X,P)$ is linear in P, and the nonlinear solve for $\nabla_P \mathcal{H}(X,P)$ is not needed, which reduces the total number of nonlinear solves.

Table 18: Number of iterations of the fixed point method for the outer solar system case, ZD scheme.

(a) $N = 3000$, `tol` = 10^{-30}.

	R=2	R=4	R=6	R=8
total iter	15000	8250	5500	4500
`nb_iter_avg`	5	2.75	1.83	1.5
`nb_call_avg`	10	11	11	12

(b) $N = 12000$, `tol` = 10^{-30}.

	R=2	R=4	R=6	R=8
total iter	54000	30000	20000	15000
`nb_iter_avg`	4.5	2.5	1.66	1.25
`nb_call_avg`	9.0	10.0	10.0	10.0

(c) $N = 12000$, `tol` = 10^{-15}.

	R=2	R=4	R=6	R=8
total iter	36000	18000	12000	9000
`nb_iter_avg`	12.0	6.0	4.0	3.0
`nb_call_avg`	24.0	24.0	24.0	24.0

Table 19: Number of iterations of the fixed point method for the outer solar system case, ZDS scheme.

(a) $N = 3000$, `tol` = 10^{-30}.

	R=1	R=2	R=3	R=4
total iter	84000	42000	30420	24000
`nb_iter_avg`	28.0	14.0	10.14	8
`nb_call_avg`	28.0	28	30.4	32.0

(b) $N = 12000$, `tol` = 10^{-30}.

	R=1	R=2	R=3	R=4
total iter	36000	18000	13575	12000
`nb_iter_avg`	3.0	1.5	1.13	1.0
`nb_call_avg`	3.0	3.0	3.4	4.0

(c) $N = 12000$, `tol` = 10^{-15}.

	R=1	R=2	R=3	R=4
total iter	24000	12404	9000	6750
`nb_iter_avg`	2.0	1.03	0.75	0.5625
`nb_call_avg`	2.0	2.06	2.25	2.25

We report in Tables 18 and 19, the total number of iterations of the fixed point method (including the initialization), the average number of iterations and call to PE for the outer solar system (a separable example from Section 5.6) and in Tables Tables 20 and 21, for the particle in an electromagnetic field (a non-separable example from Section 5.7), respectively. The number of iterations decreases as we increase the value of R, since we handle larger blocks. We then observe that the `nb_iter_avg` is also decreasing in both simulations. Moreover, the ratio also decreases for larger N since we get a better predictor for the fixed point method; hence we reduce the number of iterations to reach the tolerance value. If one releases the constraint on the tolerance to 10^{-15} in the last case, we almost divide the number of iterations by two. We run faster but less accurate; therefore the user has to determine the best trade-off between computational effort and solution quality.

Additional remarks concern the outer solar system when we take $N = 12000$ instead of $N = 3000$. We observe that we have a substantial gain since the iteration ratio strongly diminishes with a larger number of steps. This comes from a

Table 20: Number of iterations of the fixed point method for the particle in electromagnetic field case, ZD scheme.

(a) $N = 12000$, `tol` = 10^{-30}.

	R=2	R=4	R=6	R=8
total iter	188656	107441	77665	62836
`nb_iter_avg`	15.7	8.9	6.5	5.2
`nb_call_avg`	31.4	35.9	39.0	42.0

(b) $N = 48000$, `tol` = 10^{-30}.

	R=2	R=4	R=6	R=8
total iter	473032	256917	182448	142823
`nb_iter_avg`	9.6	5.4	3.8	3.0
`nb_call_avg`	19.7	21.4	22.8	23.8

(c) $N = 48000$, `tol` = 10^{-15}.

	R=2	R=4	R=6	R=8
total iter	98541	57103	41795	34031
`nb_iter_avg`	2.1	1.2	0.87	0.71
`nb_call_avg`	4.1	4.8	5.2	5.7

Table 21: Number of iterations of the fixed point method for the particle in electromagnetic field case, ZDS scheme.

(a) $N = 12000$, `tol` = 10^{-30}.

	R=1	R=2	R=3	R=4
total iter	340494	202183	161918	152791
`nb_iter_avg`	28.4	16.8	13.5	12.7
`nb_call_avg`	28.4	33.7	40.5	51.0

(b) $N = 48000$, `tol` = 10^{-30}.

	R=1	R=2	R=3	R=4
total iter	856330	474821	349537	292265
`nb_iter_avg`	17.8	9.9	7.3	6.1
`nb_call_avg`	17.8	19.8	21.8	24.4

(c) $N = 48000$, `tol` = 10^{-15}.

	R=1	R=2	R=3	R=4
total iter	422185	238998	172695	143948
`nb_iter_avg`	8.8	5.0	3.4	3.0
`nb_call_avg`	8.8	10	10.8	12.0

better initialization of the fixed point method, making a smaller number of iterations with a more efficient predictor. We highlight that the proposed initialization based on a simple Taylor expansion could be strongly improved by taking advantage of the former values of the function and derivative computing in the previous R block.

Table 22: Number of time steps, function calls and Newton iterations for the classical schemes applied to the particle in electromagnetic field case. On average, 3 Newton iterations are used at every time step, leading to 6 calls to the gradient of the Hamiltonian and 3 calls to its second derivatives per time step.

(a) $N = 2\,560\,000$, `tol` = 10^{-15}

	SV2	SV4	SV6	SV8
# time steps	2 560 000	2 560 000	2 560 000	2 560 000
# function calls	23 039 068	69 119 694	207 359 500	622 079 462
# Newton iterations	7 679 534	23 039 847	69 119 750	207 359 731

(b) $N = 5\,120\,000$, `tol` = 10^{-15}

	SV2	SV4	SV6	SV8
# time steps	5 120 000	5 120 000	5 120 000	5 120 000
# function calls	46 034 006	138 218 150	414 695 588	1 244 123 488
# Newton iterations	15 337 003	46 069 075	138 227 794	414 701 744

To compare the structural method to the classical ones, we provide in Table 22 the number of time steps, function calls and Newton iterations for the classical schemes, in the case of the particle in an electromagnetic field, and with double-precision accuracy (i.e., `tol = 10^-15`). For the second-order SV2 schemes, in this case, three explicit sub-steps and one implicit sub-step are carried out for each time step. With, on average, three Newton iterations per implicit sub-step, we obtain, on average, nine function calls per time step: 3 for the explicit steps, 3 for the gradient in the Newton iterations, and 3 for the Hessian in the Newton iteration, assuming that computing the Hessian is as expensive as computing the gradient, which is a very conservative assumption. This number of function calls per time step is on par with the ZD and ZDS schemes for `tol = 10^-15`. However, since the classical schemes require much more time steps in order to ensure stability, it turns out that the total number of function calls is much higher for the classical schemes than for the structural ones.

Based on the preceding analysis, we can summarize the key conclusions as follows:

- One of the key benefits of the structural method is its unconditional stability. Indeed, one can take very large time steps, which leads to a significant reduction in the number of function calls without loss in accuracy.
- In the separable case, both approaches allow for large time steps. For high orders of accuracy and large number N of time steps, the structural schemes are faster because $\mathtt{n}_{\mathtt{it}}^{\mathtt{ZD}}$ and $\mathtt{n}_{\mathtt{it}}^{\mathtt{ZDS}}$ remain small. At lower orders and small values of N, the structural method is slower, but its precision remains superior. As a result, we maintain a better speed-to-accuracy ratio overall.
- In the non-separable case, regardless of N, our method has a smaller complexity if a similar amount of iterations in the nonlinear solver is required. Even when the number of iterations is larger, the structural method still has a significant advantage, namely its unconditional stability. In the particle case (where the number of iterations was comparable), the classical method required time steps about 20 times smaller than the structural method to ensure stability.

In conclusion, the structural method is, in general, slightly faster than classical methods in the separable case, and significantly faster in the non-separable case.

6 Conclusions

In this paper, evidence was brought forward of the ability of the structural method to provide stable and accurate solutions in the specific context of Hamiltonian systems. We have detailed the design of the method, and provided numerical experiments with different classical Hamiltonian systems, systematically comparing the results to well-known symplectic solvers. These experiments showed that our method is quite efficient and fast, and, in the ZDS case, provides a solution of excellent quality. We also tackled a non-separable system to prove the superiority of the structural method with respecto to other schemes. In short, we have provided a general framework for the numerical approximation of Hamiltonian systems, that can easily be adapted to large classes of problems.

We would like to highlight that our method is quite versatile (e.g. by adapting the block size to fix the convergence rate), very easy to implement, and with a low memory consumption. Unconditional stability is also a desirable property, since it makes it possible to reduce the computational effort involved in dealing with long-time simulations by taking large time steps. Additionally, we also mention the simplicity of the fixed point method to solve non-linear problems where all the matrices and inverse matrices associated to the structural equations are computed once in a pre-processing stage, that dramatically reduces runtime. At last, we achieve very high accuracy that requires quad-precision (even octa-precision) when we use the ZDS scheme with $R > 2$.

A final remark is the ability to produce more physical equations by differentiating the physical equations twice (or more), and by and including the third-order (or higher-order) derivatives in the structural equations. That way, extreme accuracy (more than one hundred correct digits) would be obtained, which could be useful for applications requiring very long simulation times and very accurate approximations (planet or satellite positioning, for instance).

Funding

S. Clain was partially supported by the Centre for Mathematics of the University of Coimbra - UIDB/00324/2020, funded by the Portuguese Government through FCT/MCTES. The authors extend their thanks to ANR-24-CE46-7505 SMEAGOL. V. Michel-Dansac thanks ANR-22-CE25-0017 OptiTrust. The Shark-FV conference has greatly contributed to this work.

References

S. Clain, G. J. Machado, and M. T. Malheiro. Compact schemes in time with applications to partial differential equations. *Comput. Math. Appl.*, 140:107–125, 2023a. doi:10.1016/j.camwa.2023.03.011.

B. Leimkuhler and S. Reich. *Simulating Hamiltonian Dynamics*. Cambridge University Press, 2005. ISBN 9780511614118. doi:10.1017/cbo9780511614118.

H. Beust. Symplectic integration of hierarchical stellar systems. *Astron. Astrophys.*, 400(3):1129–1144, 2003. ISSN 1432-0746. doi:10.1051/0004-6361:20030065.

K. Feng and M.-Z. Qin. *The symplectic methods for the computation of hamiltonian equations*, pages 1–37. Springer Berlin Heidelberg, 1987. ISBN 9783540481263. doi:10.1007/bfb0078537.

M. Kraus, K. Kormann, P. J. Morrison, and E. Sonnendrücker. GEMPIC: geometric electromagnetic particle-in-cell methods. *J. Plasma Phys.*, 83(4), July 2017. ISSN 1469-7807. doi:10.1017/s002237781700040x.

Y. Li, M. Campos Pinto, F. Holderied, S. Possanner, and E. Sonnendrücker. Geometric Particle-In-Cell discretizations of a plasma hybrid model with kinetic ions and mass-less fluid electrons. *J. Comput. Phys.*, 498:112671, February 2024. ISSN 0021-9991. doi:10.1016/j.jcp.2023.112671.

T. G. Shepherd. *Symmetries, Conservation Laws, and Hamiltonian Structure in Geophysical Fluid Dynamics*, pages 287–338. Elsevier, 1990. doi:10.1016/s0065-2687(08)60429-x.

L. Verlet. Computer "Experiments" on Classical Fluids. I. Thermodynamical Properties of Lennard-Jones Molecules. *Phys. Rev.*, 159(1):98–103, 1967. ISSN 0031-899X. doi:10.1103/physrev.159.98.

E. Hairer, C. Lubich, and G. Wanner. Geometric numerical integration illustrated by the Störmer-Verlet method. *Acta Numer.*, 12:399–450, 2003. doi:10.1017/s0962492902000144.

M. Suzuki. Fractal decomposition of exponential operators with applications to many-body theories and monte carlo simulations. *Phys. Lett. A*, 146(6):319–323, 1990. ISSN 0375-9601. doi:10.1016/0375-9601(90)90962-n.

H. Yoshida. Construction of higher order symplectic integrators. *Phys. Lett. A*, 150(5–7):262–268, 1990. ISSN 0375-9601. doi:10.1016/0375-9601(90)90092-3.

R. I. McLachlan. On the numerical integration of ordinary differential equations by symmetric composition methods. *SIAM J. Sci. Comput.*, 16(1):151–168, 1995. ISSN 1095-7197. doi:10.1137/0916010.

E. Hairer, C. Lubich, and G. Wanner. *Geometric Numerical Integration*. Springer Series in Computational Mathematics. Springer-Verlag, 2006. doi:10.1007/3-540-30666-8.

R. I. McLachlan and P. Atela. The accuracy of symplectic integrators. *Nonlinearity*, 5(2):541–562, 1992. ISSN 1361-6544. doi:10.1088/0951-7715/5/2/011.

J. M. Sanz-Serna and M. P. Calvo. Symplectic numerical methods for Hamiltonian problems. *Int. J. Mod. Phys. C*, 04 (02):385–392, 1993. ISSN 1793-6586. doi:10.1142/s0129183193000410.

W. Kahan and R.-C. Li. Composition constants for raising the orders of unconventional schemes for ordinary differential equations. *Math. Comput.*, 66(219):1089–1099, 1997. ISSN 1088-6842. doi:10.1090/s0025-5718-97-00873-9.

S. Clain, R. M. S. Pereira, P. A. Pereira, and D. Lopes. Structural schemes for one dimension stationary equations. *Appl. Math. Comput.*, 457:128207, 2023b. doi:10.1016/j.amc.2023.128207.

Y. Hida, X. S. Li, and D. H. Bailey. Quad-double arithmetic: Algorithms implementation, and application. *Technical Report LBNL-46996, Lawrence Berkeley National Laboratory, Berkeley, CA 94720*, 2000. URL `http://www.nersc.gov/~dhbailey/mpdist/mpdist.html`.

Jeff Bezanson, Alan Edelman, Stefan Karpinski, and Viral B Shah. `Julia`: A fresh approach to numerical computing. *SIAM review*, 59(1):65–98, 2017. URL `https://doi.org/10.1137/141000671`.

C. Rackauckas and Q. Nie. `DifferentialEquations.jl` – A Performant and Feature-Rich Ecosystem for Solving Differential Equations in `Julia`. *J. Open Res. Softw.*, 5(1):15, 2017. ISSN 2049-9647. doi:10.5334/jors.151.

A. Beléndez, C. Pascual, D.I. Méndez, T. Beléndez, and C. Neipp. Exact solution for the nonlinear pendulum. *Rev. Bras. Ens. Fis.*, 29(4):645–648, 2007.

B. D. Andrews and P. E. Farrell. High-order conservative and accurately dissipative numerical integrators via auxiliary variables, 2024.

C. Moore. Braids in classical dynamics. *Phys. Rev. Lett.*, pages 3675–3679, 1993.