Study of geometric integrators for differential equations

Gilles Vilmart

UNIVERSITÉ DE GENÈVE
Section de mathématiques

FACULTÉ DES SCIENCES
Prof. Ernst HAIRER

UNIVERSITÉ DE RENNES 1
Institut National de Recherche en Informatique et
Automatique de Rennes Bretagne Atlantique – Projet IPSO
École Normale Supérieure de Cachan, antenne de Bretagne

UFR MATHÉMATIQUES
Dr. Philippe CHARTIER

Étude d'intégrateurs géométriques pour des équations différentielles

Thèse

présentée à la Faculté des Sciences de l'Université de Genève
en co-tutelle internationale avec l'Université de Rennes 1
pour obtenir les grades de
Docteur ès Sciences de l'Université de Genève, mention Mathématiques
Docteur de l'Université de Rennes 1, mention Mathématiques et applications

par

Gilles VILMART
(France)

Thèse N° 4038
(N° pour l'Université de Rennes 1 : 3758)

RENNES & GENÈVE
Atelier d'impression ReproMail de l'Université de Genève
2008

UNIVERSITÉ
DE GENÈVE

FACULTÉ DES SCIENCES

Doctorat ès sciences
mention mathématiques

Thèse en cotutelle avec **l'Université de Rennes 1 – France**

Thèse de *Monsieur Gilles VILMART*

intitulée :

"Etude d'intégrateurs géométriques pour des équations différentielles "

La Faculté des sciences, sur le préavis de Messieurs E. HAIRER, professeur ordinaire et codirecteur de thèse (Section de mathématiques), Ph. CHARTIER, docteur et codirecteur de thèse (Université de Rennes 1, France - Institut National de Recherche en Informatique et Automatique et Ecole Nationale Supérieure de Cachan, Bretagne, France), Ch. LUBICH, professeur (Universität Tübingen – Mathematisches Institut – Tübingen, Deutschland), H. Z. MUNTHE-KAAS, professeur (University of Bergen – Department of Matematics – Bergen, Norway), autorise l'impression de la présente thèse, sans exprimer d'opinion sur les propositions qui y sont énoncées.

Genève, le 2 décembre 2008

Thèse - 4038 -

Le Doyen, Jean-Marc TRISCONE

N.B.- La thèse doit porter la déclaration précédente et remplir les conditions énumérées dans les "Informations relatives aux thèses de doctorat à l'Université de Genève".

Nombre d'exemplaires à livrer par colis séparé à la Faculté : - 7 -

à Thierry

à Shaula

Remerciements

Ma plus sincère gratitude va d'abord à Philippe Chartier et Ernst Hairer pour leur amitié, leur disponibilité, leurs encouragements, et pour la confiance qu'ils m'ont témoignée en encadrant ma thèse, en me laissant toujours une grande liberté. Ce fut un immense honneur de travailler sous leur direction.

Je suis très reconnaissant à Christian Lubich et Hans Z. Munthe-Kaas qui ont accepté de rapporter sur ma thèse et m'ont fait l'honneur d'être membres du jury de soutenance. Je souhaite en particulier les remercier pour l'intérêt qu'ils ont bien voulu porter à ce travail.

Ces trois années de thèse en cotutelle m'ont donné l'occasion de nombreuses rencontres mathématiques, avec des discussions agréables et stimulantes (même si la prononciation de mon prénom fut parfois non-standard). Je remercie avec plaisir Assyr Abdulle, Sergio Blanes, Fernando Casas, François Castella, Elena Celledoni, David Cohen, Stéphane Descombes, Erwan Faou, Francesco Fasso, Martin Gander, Roman Kozlov, Ander Murua, Brynjulf Orwen, Gerhard Wanner et Antonella Zanna pour leurs conseils et leur gentillesse à chaque occasion, et en particulier Monique Chyba et Thomas Haberkorn pour leur hospitalité à l'Université de Hawaii. Merci également à Damien Calaque, Kurush Ebrahimi-Fard et Dominique Manchon, les algébristes, pour leur gentillesse et leur enthousiasme.

La cotutelle de la thèse m'a permis d'intégrer non seulement la Section de mathématiques de l'Université de Genève, mais aussi l'équipe IPSO à l'IRISA/INRIA à Rennes, et par la suite le département de mathématiques de l'Ecole Normale Supérieure de Cachan, antenne de Bretagne, après le déménagement d'IPSO. J'exprime ma profonde gratitude à tous leurs membres pour leur accueil chaleureux. J'ai ainsi bénéficié – triplement – d'un environnement de travail exceptionnel.

Un grand merci à Bernard Dudez et Anne-Sophie Crippa, bibliothécaires de la Section de mathématiques, pour leur gentillesse, leur disponibilité et leur dynamisme. Merci également aux secrétaires de la Section pour leur gentillesse. Je remercie chaleureusement les assistant-e-s de l'IRISA, Angélina, Myriam, Fabienne, Sara, Loïc et Laurence, la super-assistante, et aussi Elisabeth et Violaine du Service Relations extérieures, pour leur gentillesse et leur efficacité. Merci également aux membres de l'Atelier informatique pour leur compétence et leur dévouement.

Un grand merci aux membres du département de mathématiques de l'ENS Cachan Bretagne, qui m'ont d'abord supporté comme étudiant pendant mes études à Rennes. Je remercie tout particulièrement Virginie Bonnaillie-Noël, François Castella, Michel Crouzeix, Eric Darrigrand, Arnaud Debussches, Michel Pierre, Rozenn Texier-Picard et Grégory Vial qui m'ont donné le goût de l'analyse numérique. Un grand merci à Adrien, Agnès, Fanny, Jimmy, Ludovic du plateau de mathématiques, pour leur bonne humeur, les pauses cafés, les crêperies, et le reste également. Je remercie chaleureusement tous les membres de

l'équipe IPSO pour leur accueil. Remerciement spécial à Guillaume qui a tant partagé avec moi.

Un grand merci dans le désordre d'apparition aux assistants de la Section à Genève : Felix, Benjamin, Heike, Yves, Masha, Daniele, Alfredo, Benoît, Jérôme, Jean-Luc, Emanuela, Carlo, Samuel, Jérémy, Guilherme, Luc, Nicolas, Pavol, Rudolphe, Clément, Victor, Vincent, José et bien sûr Grégoire, toujours témoin, pour la bonne humeur et la sympathique ambiance qui règne à la Section. Merci aussi à Pierre-Alain. Je remercie aussi tous ceux que je n'ai pas cité.

Je remercie chaleureusement le Swiss Doctoral Program in Mathematics et ses directeurs dynamiques Bruno Colbois et Norbert Hungerbüler, pour les nombreuses et sympathiques rencontres de l'École Doctorale, et pour avoir financé certains de mes déplacements. Merci également à l'École Doctorale Matisse et son directeur Olivier Bonnaud, pour les nombreuses formations organisées. Merci au Fond National Suisse et à l'IRISA/INRIA pour leur contribution financière majeure, en particulier en finançant mes déplacements, notamment mes allers-retours entre Rennes et Genève.

Un grand merci à mes amis parisiens, rennais, genevois et maintenant d'un peu partout, pour tous les moments passés ensemble, studieux ou détendus ou les deux.

Enfin, je remercie tout particulièrement mes parents et mon frère Thierry pour leurs encouragements et pour m'avoir toujours soutenu avec fierté, en me laissant toujours libre de mes choix. Je termine par un grand merci à mon épouse Shaula en particulier pour tout (également pour m'avoir aidé à taper la fin de la traduction de l'introduction, pour passer des vacances d'été détendues).

Contents

Introduction and main results

The aim of the work described in this thesis is the construction and the study of structure-preserving numerical integrators for differential equations, which share some geometric properties of the exact flow, for instance symmetry, symplecticity of Hamiltonian systems, preservation of first integrals, Poisson structure, etc. It may be divided into three closely related parts.

In the first part (Chapters 1, 2, 3), we introduce a new approach to high-order structure-preserving numerical integrators, inspired by the theory of modified equations (backward error analysis). We focus on the class of B-series methods for which a new composition law called substitution law is introduced. This approach is illustrated with the derivation of an efficient and high-order geometric integrator for the motion of a rigid body. We also obtain an accurate integrator for the computation of conjugate points in rigid body geodesics.

In the second part (Chapter 4), we study to which extent the excellent performance of symplectic integrators for long-time integrations in astronomy and molecular dynamics carries over to problems in optimal control. We also discuss whether the theory of backward error analysis can be extended to symplectic integrators for optimal control.

The third part (Chapters 5 and 6) is devoted to splitting methods. In the spirit of modified equations, we consider splitting methods for perturbed Hamiltonian systems that involve modified potentials. Finally, we investigate the use of splitting methods involving complex coefficients for parabolic partial differential equations with special attention to reaction-diffusion problems.

Chapter 1 Inspired by the theory of modified equations (backward error analysis), a new approach to high-order, structure-preserving numerical integrators for ordinary differential equations is developed. It is called modifying (or preprocessed) vector field integrator because the vector field is modified before the method is applied. This approach is illustrated with the implicit midpoint rule applied to the full dynamics of the free rigid body. Special attention is paid to methods represented as B-series, for which explicit formulae for the modified differential equation are given. A new composition law on B-series, called substitution law, is presented.

Chapter 2 We explain the common algebraic structure of two composition laws on B-series: the Butcher composition, which corresponds to the composition of flows of integrators, and the substitution law, introduced in the previous chapter, which corresponds to the composition of B-series vector fields. Hopf algebra structures on rooted trees are a well-studied object, especially in the context of combinatorics, and are essentially characterized by the coproduct map. It is well-known that the first composition law corresponds to the convolution product on the Hopf tree algebra of Connes & Kreimer in renormalization in quantum field theory, while it was shown recently that the second composition law can be turned into a new coproduct, which allows to build another Hopf tree algebra. We explain

their algebraic relationships from the point of view of geometric numerical integration.

Chapter 3 As an application of the idea of modifying integrators, we construct a computationally efficient and highly accurate integrator for the motion of a free rigid body. The Discrete Moser-Veselov algorithm is an integrable discretisation of the equations of motion. It is symplectic and time-reversible, and it conserves all first integrals of the system. The only drawback is its low order. We present a modification of this algorithm to arbitrarily high order which has negligeable overhead but considerably improves the accuracy. We also study the propagation with time of round-off error and explain how it can be reduced. Finally we propose a modification which allows to compute the tangent map, for the accurate computation of conjugate points of rigid body geodesics.

Chapter 4 For general optimal control problems, Pontryagin's maximum principle gives necessary optimality conditions which are in the form of a Hamiltonian differential equation. For its numerical integration, symplectic methods are a natural choice. We investigate to which extent the excellent performance of symplectic integrators for long-time integrations in astronomy and molecular dynamics carries over to problems in optimal control. Numerical experiments supported by a backward error analysis show that, for problems in low dimension close to a critical value of the Hamiltonian, symplectic integrators have a clear advantage. This is illustrated using the Martinet case in sub-Riemannian geometry. For problems like the orbital transfer of a spacecraft or the control of a submerged rigid body such an advantage cannot be observed. The Hamiltonian system is a boundary value problem and the time interval is in general not large enough so that symplectic integrators could benefit from their structure preservation of the flow. We also discuss whether it is possible to extend the theory of backward error analysis to symplectic integrators for optimal control.

Chapter 5 We study splitting methods for (perturbed) Hamiltonian systems using modified potentials that involve several Lie brackets. We show that this approach initially developed for order-two differential equations (e.g. N-body problems in Jacobi coordinates) can be successfully applied also to asymmetric rigid body problems with an external potential. This is illustrated with the asymmetric heavy top, a satellite model, and a molecular dynamics simulation with dipolar soft spheres. We also build a new processor for the Takahashi-Imada method (a modification of the Störmer-Verlet method), to achieve order $\mathcal{O}(h^{10}\varepsilon + h^4\varepsilon^2)$ for perturbed Hamiltonian systems, where h is the stepsize and ε is the size of the perturbation. It turns out to be very efficient in many situations.

Chapter 6 The last chapter is devoted to splitting methods involving complex coefficients for linear and non-linear parabolic equations. It is known that all splitting methods with real coefficients of order greater than 2 must have negative coefficients. Thus, these methods with real coefficients cannot be used when one operator, like the Laplacian Δ, is not time-reversible and cannot be solved with negative times. To circumvent this order-barrier, we derive new high-order splitting methods using complex coefficients, based on composition techniques originally developed for the geometric numerical integration of ordinary differential equations. We give a theoretical justification of the order of the introduced methods in the linear case for exponential maps. Our numerical simulations show that the order of accuracy is the one expected especially in case of a non-linear source, and for the Peaceman-Rachford discretization as basic ingredient.

0.1 Geometric numerical integration

In this section, we present important aspects of geometric numerical integration for ordinary differential equations, see the monographs [SSC94, LR04, HLW06]. Geometric integration is a wide field, and we give here only a few ideas which are relevant for understanding the work in this thesis. We illustrate these ideas with the example of the Kepler problem, the three-body-problem in celestial mechanics, and the asymmetric pendulum.

Consider a system of differential equations[1],

$$\dot{y} = f(y), \qquad y(0) = y_0 \tag{0.1}$$

with sufficiently differentiable vector field $f(y)$ and an initial condition y_0. The simplest of all numerical integrators for the system (0.1) was designed by Euler in 1768 [Eul68],

$$y_{n+1} = y_n + hf(y_n).$$

It uses a stepsize h to compute recursively approximations $y_1, y_2, y_3, \dots$ to the values $y(h), y(2h), y(3h), \dots$ of the solution. It is called the explicit Euler method because the computation of y_{n+1} is performed explicitly with one evaluation of f at the already known value y_n. In contrast, the implicit Euler method

$$y_{n+1} = y_n + hf(y_{n+1})$$

requires the numerical resolution of a nonlinear system of equations at each step.

Exact flow We define the (exact) flow φ_t of differential equation (0.1) over time t to be the mapping which, to any point y_0 in the phase space associates the value $y(t)$ of the solution of the ordinary differential equation with initial value $y(0) = y_0$. This map, denoted φ_t is thus given by

$$\varphi_t(y_0) = y(t) \quad \text{if} \quad y(0) = y_0.$$

A numerical one-step method Φ_h is a mapping that approximates the time-h flow φ_h of the differential equation (0.1).

Definition 0.1.1 *A numerical method $y_{n+1} = \Phi_h(y_n)$ has order p for problem (0.1) if the local error satisfies*

$$\Phi_h(y) - \varphi_h(y) = \mathcal{O}(h^{p+1}) \quad \text{for } h \to 0.$$

It can be verified by Taylor series expansion that the implicit and explicit Euler methods have order 1, by comparing the exact and numerical flows.

To achieve higher accuracy, Runge [Run95] and Heun [Heu00] constructed methods including several Euler steps and Kutta [Kut01] then formulated general Runge-Kutta methods one century ago. For instance, the method

$$\begin{aligned} Y_1 &= y_n & Y_2 &= y_n + \frac{h}{2} f(Y_1) \\ Y_3 &= y_n + \frac{h}{2} f(Y_2) & Y_4 &= y_n + hf(Y_3) \\ y_{n+1} &= y_n + \frac{h}{6}\big(f(Y_1) + 2f(Y_2) + 2f(Y_3) + f(Y_4)\big) \end{aligned} \tag{0.2}$$

[1] Notice that a nonautonomous system $\dot{y} = f(t, y)$ can be cast into this form by considering the additional equation $\dot{t} = 1$.

is ofter referred to as 'The' Runge-Kutta method of order 4 (even if there are infinitely many choices). The derivation of order conditions for Runge-Kutta methods becomes very elegant using the framework for rooted trees and B-series, a theory initiated by Butcher in the years 1963-72 [But63, But64a, But64b, But69, But72].

B-series methods B-series were introduced by Hairer & Wanner [HW74]. The Taylor series of the exact solution of (0.1) with initial value $y(0) = y$ can be written as

$$y(h) = y + hf(y) + \frac{h^2}{2!} f'(y)f(y) + \frac{h^3}{3!}\Big(f''(f(y), f(y)) + f'(y)f'(y)f(y)\Big) + \dots$$

This is because $\dot y = f(y)$, $\ddot y = f'(y)\dot y = f'(y)f(y)$, etc. B-series methods are numerical integrators $y_{n+1} = \Phi_h(y_n)$ whose Taylor series have the same structure with real coefficients $a(\tau)$:

$$\Phi_h(y) = y + ha(\bullet)f(y) + h^2 a(\rotatebox{90}{$\curlyvee$})f'(y)f(y) + h^3\Big(\frac{a(\vee)}{2} f''(f(y), f(y)) + a(\rotatebox{90}{$\curlyvee$}\bullet)f'(y)f'(y)f(y)\Big) + \dots$$

where coefficients $a(\tau)$ are defined for all rooted trees and characterize the integrator. B-series not only comprise all Runge-Kutta methods, but also Taylor series methods, the underlying one-step method of linear multistep methods, etc (see [HLW06, Chap. XIV]).

For special classes of differential equations, it is essential to use numerical integrators that share geometric properties of the exact flow to reproduce the qualitative behavior of the solution.

Example: Newton's historical proof of Kepler's second law The Kepler problem which describes the motion of two bodies attracting each others, e.g. a planet rotating around the Sun, is given by the differential equation

$$\dot q = p, \qquad \dot p = f(q) = -\frac{q}{\|q\|^3}, \tag{0.3}$$

where $q = (q_1, q_2)$ and $p = (p_1, p_2)$ represent the positions and momenta of the planet relative to the Sun. We shall see below that this system possesses several geometric properties, in particular, it is a Hamiltonian system. Kepler's second law states that the angular momentum

$$\det(q, p) = q_1 p_2 - q_2 p_1$$

is a first integral, i.e. a conserved quantity along any solution of the system of differential equations (0.3). Of course, this can be checked by direct differentiation. In 1687, Newton gave in 'Theorema 1' of his *Principia* [New87] an elegant geometric proof of Kepler's second law. Surprisingly, his proof relies on a geometric integrator: the symplectic Euler method, which is closely related to the Störmer–Verlet scheme, a widely used integrator in molecular dynamics because of its excellent geometric behaviour. A presentation of Kepler and Newton's great discoveries, actually made by very geometric reasoning, can be found in [HLW06, Sect. I.1.4] and in the forthcoming book of Ostermann & Wanner [OW08].

Newton's proof. The proof of Newton relies on the following discretization of the differential equation (0.3)

$$q_{n+1} = q_n + hp_n, \qquad p_{n+1} = p_n + hf(q_{n+1}),$$

which is known today as the symplectic Euler method and can be interpreted as follows. Consider Newton's Figure 1, where S represents the Sun and let $A = q_{n-1}$, $B = q_n$, $C = q_{n+1}$, $D = q_{n+2}$, etc. During the first time step, the body moves from A to B without

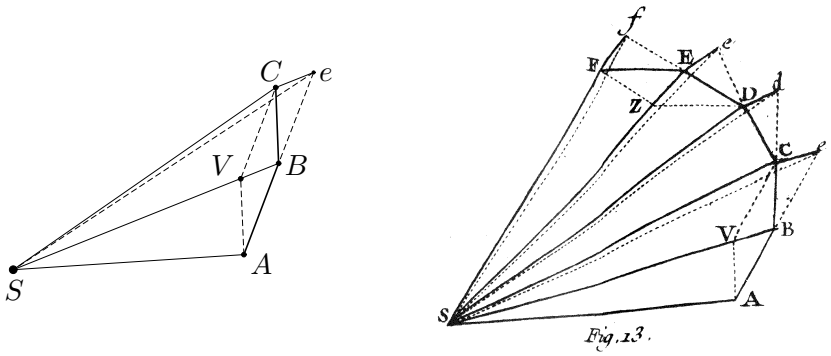


Figure 1: Facsimile from Newton's *Principa* (right picture)

force, i.e. with constant velocity p_{n-1}. At point B, we suppose to have a force-impulse $f(q_n)$, where the speed is slightly changed in direction of the Sun. During the second time step, the body moves to C with the constant speed p_n, and so on, repeatedly. A direct computation shows that the above discretization implies the following natural discretization

$$q_{n+1} - 2q_n + q_{n+1} = h^2 f(q_n). \tag{0.4}$$

In fact, considering (0.4) together with the more accurate speed approximation $p_n = (q_{n+1} - q_{n-1})/(2h)$, we obtain what is known today as the Störmer–Verlet or leap-frop method (see further on an equivalent one-step formulation). Now, Newton's geometric argument is the following. The diagonal (BV) of the parallelogram $ABCV$ points towards the Sun S because

$$\overrightarrow{BV} = \overrightarrow{BC} - \overrightarrow{AB} = (q_{n+1} - q_n) - (q_n - q_{n-1}) = h^2 f(q_n) = Const \cdot q_n.$$

Notice that in the absence of the force, the planet would have continued to move with constant speed in straight line from B to e, so $CVBe$ is a parallelogram. Then, triangles SAB and SBe have the same base length ($\overrightarrow{AB} = \overrightarrow{Be}$) and the same altitude, and thus the same area. Similarly, triangles SBC and SBe with the common base SB and the same altitude, have the same area. Hence, triangles SAB and SBC have the same areas:

$$\det(q_{n-1}, q_n - q_{n-1}) = \det(q_n, q_{n+1} - q_n).$$

In the same way, all triangles SAB, SBC, SCD, etc, have the same area. Substituting p_n as a function of $q_n, q_{n+1}, \ldots$, we obtain that the angular momentum $\det(q_n, p_n) = \det(q_{n-1}, p_{n-1})$ is exactly conserved along the discretization (0.4) (for both symplectic Euler and Störmer–Verlet). We conclude that the motion of a body, urged by any centripedal force satisfies Kepler's second law. □

0.1.1 Hamiltonian systems and symplectic integrators

One of the most important class of problems in geometric numerical integration is Hamiltonian systems, see the survey [Hai05] on long-time energy conservation. These are problems of the form

$$\dot{p} = -H_q(p, q), \qquad \dot{q} = H_p(p, q)$$

where $H(p, q)$ is a scalar function which represents the total energy, the vectors q and p of dimension d represent the position and the momenta, and d is the number of degrees of

freedom. Here, H_p and H_q denote the vectors of partial derivatives. Hamiltonian systems can be written out in the form (0.1) using matrices,

$$\dot{y} = J^{-1}\nabla H(y) \qquad \text{with} \quad J = \begin{pmatrix} 0 & Id \\ -Id & 0 \end{pmatrix}, \tag{0.5}$$

where the vector $y = (p, q)^T$ has dimension $2d$ in the phase space and Id denotes the identity matrix of size d. For instance, the Kepler problem (0.3) is a Hamiltonian system with $d = 2$ degrees of freedom and with $H(p, q) = p^T p/2 + 1/\|q\|$.

Hamiltonian systems possess the following two fundamental properties.

Energy conservation The energy $H(y) = H(p, q)$ is constant along solutions of the differential equation. We say that it is a first integral of the system. This can be checked easily by differentiation: $\frac{d}{dt}H(y(t)) = 0$.

Symplecticity The Jacobian derivative of the flow φ_t with respect to y of a Hamiltonian system (0.5) satisfies the matrix identity (Poincaré [Poi92])

$$\varphi_t'(y)^T J \varphi_t'(y) = J.$$

In fact, this property characterizes Hamiltonian systems, see [HLW06, Theorem VI.2.8]. It implies the preservation of volume ($|\det \Phi_h'(y)| = 1$) in all dimensions, and it is equivalent to the preservation of volume in dimension $d = 1$, see [HLW06, Sect. VI.2].

This motivates the following definition.

Definition 0.1.2 *A numerical integrator $y_{n+1} = \Phi_h(y_n)$ is symplectic for a Hamiltonian system (0.5) if the Jacobian matrix of the numerical flow satisfies*

$$\Phi_h'(y)^T J \Phi_h'(y) = J$$

for all stepsize h (small enough).

Unfortunately, a numerical integrator cannot be simultaneously symplectic and energy-preserving, otherwise it is a time-transformation of the exact flow. This result is due to Ge & Marsden [GM88] and an algebraic proof was given by Chartier, Faou & Murua [CFM06]. However, a symplectic integrator conserves $d(2d-1)$ invariants by definition, and we shall see further that under precise hypotheses, symplectic integrators for Hamiltonian systems well-conserve the energy over exponentially long times.

We start with examples of symplectic methods.

Implicit midpoint rule One of the simplest symplectic integrator is the implicit midpoint rule,

$$y_{n+1} = y_n + h\, f\Big(\frac{y_n + y_{n+1}}{2}\Big).$$

It is a two-stage Runge-Kutta method, and thus a B-series integrator.

The next two integrators are not B-series methods but P-series methods, a natural extension to partitioned systems, involving bi-colored rooted trees.

Symplectic Euler Combining the explicit and implicit Euler methods yields two adjoint methods (called with the same name),

$$\left\{ \begin{array}{rcl} p_{n+1} &=& p_n - hH_q(p_{n+1}, q_n) \\ q_{n+1} &=& q_n + hH_p(p_{n+1}, q_n) \end{array} \right. \qquad \text{and} \qquad \left\{ \begin{array}{rcl} p_{n+1} &=& p_n - hH_q(p_n, q_{n+1}) \\ q_{n+1} &=& q_n + hH_p(p_n, q_{n+1}) \end{array} \right. .$$

Störmer–Verlet scheme Composing a half-step of each symplectic Euler methods yields

$$\begin{aligned} p_{n+1/2} &= p_n - \frac{h}{2} H_q(p_{n+1/2}, q_n) \\ q_{n+1} &= q_n + \frac{h}{2}\Big(H_p(p_{n+1/2}, q_n) + H_p(p_{n+1/2}, q_{n+1})\Big) \\ p_{n+1} &= p_{n+1/2} - \frac{h}{2} H_q(p_{n+1/2}, q_{n+1}) \end{aligned}$$

These methods already appeared in Newton's geometric proof of Kepler's second law presented at the beginning of this introduction. For separable Hamiltonian $H(q,p) = p^T p/2 + U(q)$ it can be shown that this scheme is the one-step formulation of the equivalent discretization (0.4) where $f(q) = -\nabla U(q)$, together with the velocity approximation $p_n = (q_{n+1} - q_{n-1})/(2h)$. Notice that both the symplectic Euler method and the Störmer–Verlet method are explicit for separable Hamiltonian.

Symmetric integrators It can be shown that both the implicit midpoint rule and the Störmer–Verlet scheme are symmetric methods, i.e.,

$$\Phi_h \circ \Phi_{-h}(y) = y \qquad \text{or equivalently} \qquad \Phi_{-h}^{-1}(y) = \Phi_h(y).$$

This can be easily checked by observing that the interchanges $y_n \leftrightarrow y_{n+1}$, $h \leftrightarrow -h$ do not modify the methods. These two integrators thus have order 2, because a symmetric method always has a even order of accuracy [HLW06, Theorem II.3.2].

Numerical experiment: three-body problem We consider the three-body problem (Sun-Jupiter-Saturn) which is a Hamiltonian system with

$$H(p,q) = \frac{1}{2}\sum_{i=0}^{2} \frac{1}{m_i} p_i^T p_i - G \sum_{i=1}^{2} \sum_{j=0}^{i-1} \frac{m_i m_j}{\|q_i - q_j\|}.$$

We take the initial values $q_i(0), p_i(0)$ in $\mathbb{R}^3$, the constant G and the masses m_i from [HLW06, Table I.2.2]. To this system we apply the explicit Euler method with stepsize $h = 2$, the symplectic Euler method and the Störmer–Verlet method with much larger stepsize $h = 50$, both over a period of 450 000 days. We also give the results for the explicit Runge-Kutta method (0.2) with order 4, and thus a larger stepsize $h = 250$. In Figure 2, we observe that both the symplectic Euler method and Störmer–Verlet show the correct behaviour. For the explicit Euler method, we observe that the planets spiral outwards with increasing energy, whereas for the explicit Runge-Kutta method Jupiter falls into the Sun and is thrown away. Notice the symplectic Euler method and Störmer–Verlet would still show the correct behaviour even if we had used the larger stepsize $h = 250$.

In our next experiment (Figure 3), we study the conservation of energy. We observe that the energy error grows linearly with time for the non-symplectic methods (the explicit Euler and the Runge-Kutta method of order 4). The justification of this linear growth with time is straightforward, using the fact that the exact flow φ_h conserves the Hamiltonian, we have

$$H(y_{n+1}) - H(y_n) = H(y_{n+1}) - H(\varphi_h(y_n)) = \mathcal{O}(h^{p+1}),$$

where we use $y_{n+1} = \varphi_h(y_n) + \mathcal{O}(h^{p+1})$. After summation of this estimate from $n = 0$ to $N - 1$, we obtain the linear bound

$$H(y_N) - H(y_0) = \mathcal{O}(th^p),$$

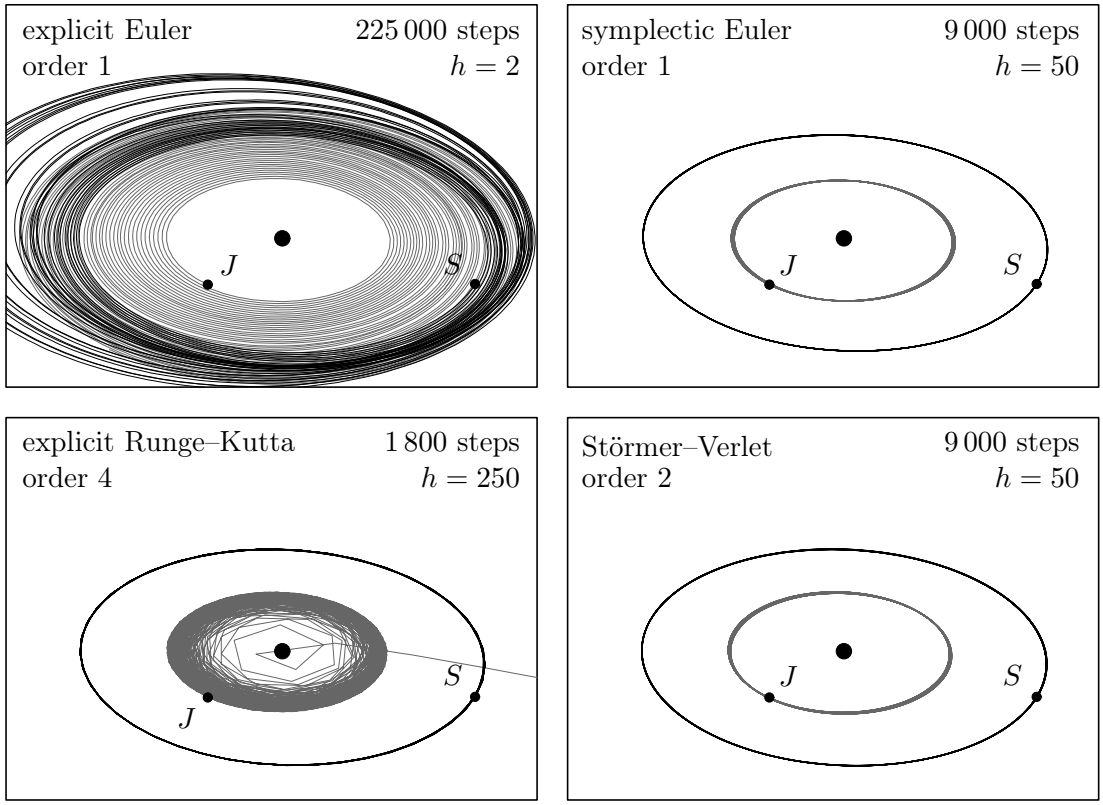


Figure 2: Symplectic and non-symplectic integrators for the Sun-Jupiter-Saturn system (large stepsizes).

where $t = Nh$ and p is the order of the method.

In contrast, the energy error remains bounded and small (without linear drift) for the symplectic integrators (symplectic Euler and Störmer–Verlet),

$$H(y_N) - H(y_0) = \mathcal{O}(h^p).$$

The theoretical explanation of this behaviour is due to Benettin & Giorgilli [BG94] and Tang [Tan94]. It is obtained using the theory of backward error analysis.

0.1.2 Backward error analysis

Consider a system of ordinary differential equations (0.1) $\dot{y} = f(y)$ and a numerical integrator

$$y_{n+1} = \Phi_h(y_n).$$

The idea of backward error analysis is to search for a modified differential equation

$$\dot{z} = \widetilde{f}_h(z) = f(z) + hf_2(z) + h^2 f_3(z) + \ldots, \qquad z(0) = y_0, \tag{0.6}$$

which is a formal series in powers of the stepsize h, such that the numerical solution $\{y_n\}$ is formally equal to the exact solution of (0.6),

$$y_n = z(nh) \qquad \text{for} \quad n = 0, 1, 2, \ldots, \tag{0.7}$$

that is (see the top picture of Figure 5)

$$\Phi_{f,h}(y) = \varphi_{\widetilde{f}_h,h}(y), \tag{0.8}$$

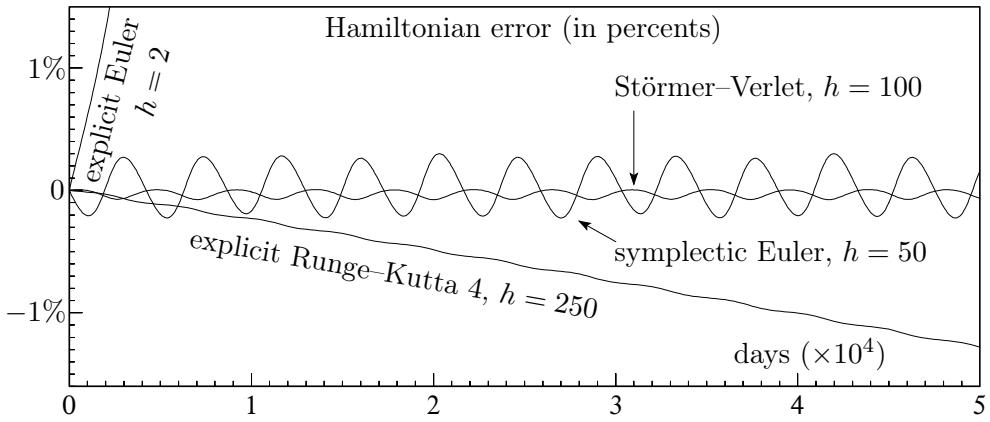


Figure 3: Energy conservation for the three-body problem Sun-Jupiter-Saturn.

where $\varphi_{\widetilde{f}_h,h}$ denotes the exact flow of (0.6).

The idea of backward error analysis was originally introduced by Wilkinson (1960) in the context of numerical linear algebra. For the integration of ordinary differential equations it was not used until one became interested in the long-time behaviour of numerical solutions. Without considering it as a theory, Ruth [Rut83] uses the idea of backward error analysis to motivate symplectic integrators for Hamiltonian systems. In fact, applying a symplectic numerical method to a Hamiltonian system $\dot{y} = J^{-1}\nabla H(y)$ gives rise to a modified differential equation that is Hamiltonian,

$$\dot{z} = \widetilde{f}_h(z) = J^{-1}\nabla H_h(z), \qquad H_h(z) = H(z) + hH_2(z) + h^2H_3(z) + \dots . \tag{0.9}$$

Backward error analysis permits to transfer known properties of perturbed Hamiltonian systems (e.g., conservation of energy, KAM theory for integrable systems) to properties of symplectic numerical integrators. One became soon aware that this kind of reasoning is not restricted to Hamiltonian systems, and new insight can be obtained with the same techniques also for reversible differential equations, for Poisson systems, for divergence-free problems, etc.

A rigorous analysis has been developed in the nineties. We have the following central Theorem which rigorously justifies the use of symplectic integrators and is due to Benettin & Giorgilli [BG94] and Tang [Tan94], see [HLW06, Sect. IX.8].

Theorem 0.1.3 *Consider a Hamiltonian system (0.5) with analytic $H : U \to \mathbb{R}$ and a B-series (or P-series) integrator $y_{n+1} = \Phi_h(y_n)$ of order p applied with constant[2] stepsize h. Assume*

- *the integrator is symplectic for all Hamiltonian systems $\dot{y} = J^{-1}\nabla H(y)$;*
- *and the numerical solution stays in a compact set.*

Then, we have for $t_n = nh$ and $h \to 0$,

$$\begin{aligned} \widetilde{H}(y_n) &= \widetilde{H}(y_0) + \mathcal{O}(e^{-\gamma/(\omega h)}), \\ H(y_n) &= H(y_0) + \mathcal{O}(h^p), \end{aligned}$$

on exponentially long time intervals $nh \le e^{\gamma/(\omega h)}$, where $\gamma > 0$ depends only on the method, and $\omega > 0$ is related to the Lipschitz constant (highest frequency) of the differential equation.

[2] The excellent behaviour of symplectic integrators is lost in general for variable stepsizes, see [HLW06, Sect. VIII.2]. Here, we always consider a constant stepsize h.

This means that for small enough stepsize h, the energy is well conserved up to a bounded term $\mathcal{O}(h^p)$ on exponentially long time intervals. The main idea of the proof is that the numerical solution $\{y_n\}$ is (formally) the exact solution of the perturbed Hamiltonian system (0.9), via backward error analysis. Then, the numerical solution (formally) exactly conserves the modified Hamiltonian $H_h(z)$. Since this modified Hamiltonian is a small perturbation of size $\mathcal{O}(h^p)$ of the original Hamiltonian $H(y)$, the original Hamiltonian is well-conserved.

Notice that modified differential equations (0.6) are formal series which do not converge in general (except for linear problems), and this makes the rigorous analysis rather technical. One has to truncate the series so that the resulting error is as small as possible. It can be shown that truncating the series (0.6) after the term of size $\mathcal{O}(h^{N(h)})$ where $N(h) = \mathcal{O}(1/h)$ yields the exponentially small truncation error appearing in Theorem 0.1.3.

Nevertheless, energy conservation results obtained using backward error analysis as described previously DO NOT apply to highly oscillatory differential equations or to infinitely dimensional problems (partial differential equations) because the conclusion of Theorem 0.1.3 becomes void for $\omega \to \infty$.

Remark 0.1.4 Not only symplectic integrators have a good long time-behaviour. For instance, the trapezoidal rule

$$y_{n+1} = y_n + \frac{h}{2}\Big(f(y_n) + f(y_{n+1})\Big) =: \Phi_h^{trap}(y_n)$$

is not symplectic, but it is conjugate to the implicit midpoint rule which is symplectic. Indeed, there exists a map χ_h, which is a $\mathcal{O}(h^2)$ perturbation of the identity, such that

$$\Phi_h^{trap} = (\chi_h)^{-1} \circ \Phi_h^{midpoint} \circ \chi_h$$

Thus, after n steps of the method, $(\Phi_h^{trap})^n = (\chi_h)^{-1} \circ (\Phi_h^{midpoint})^n \circ \chi_h$ and the trapezoidal rule has the same good long-time behaviour as the symplectic implicit midpoint rule. This is called conjugate symplecticity ([Sto88], see [HLW06, Sect. VI.8]).

Remark 0.1.5 Let us mention that similar conservation results can be obtained for B-series (or P-series) symmetric methods applied to integrable reversible systems (like the Kepler problem) and perturbed integrable reversible systems (like the tree-body problem Sun-Jupiter-Saturn), see [HLW06, Chap. XI]. This is of importance as the symmetry property is in general easier to achieve than the symplecticity of a numerical integrator.

Asymmetric pendulum To illustrate the difficulties that can be encountered by a symplectic method, we end this section with the asymmetric pendulum problem proposed in [FHP04] which is a one-degree-of-freedom Hamiltonian system with

$$H(p, q) = p^2/2 - \cos q + 0.2 \sin(2q).$$

We consider the initial condition $q(0) = 0$, $p(0) = 2.5$. The initial velocity is sufficiently large so that the pendulum turns around, and the velocity $p(t)$ remains positive. Here, the symmetry $p \leftrightarrow -p$ has no influence on the numerical solution, and the perturbation $+0.2\sin(2q)$ destroys the symmetry $q \leftrightarrow -q$. Thus, Remark 0.1.5 for symmetric methods does not apply. For the Störmer–Verlet method (see Figure 4 with stepsize $h = 0.05$), the energy is well conserved, and this is a direct consequence of Theorem 0.1.3. One may

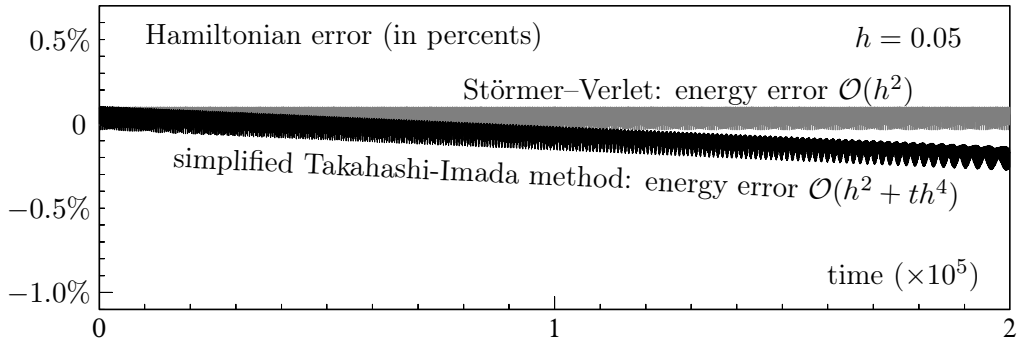


Figure 4: Hamiltonian error along the numerical solution of the asymmetric pendulum. This counter-example illustrates that symplecticity alone is not sufficient for a good long-time energy conservation. It is taken from [HMS08].

argue that the solution does not remain in a compact set because $q(t)$ grows infinitely as the pendulum turns around. However, $q(t)$ represents the angle of the pendulum which is defined modulo 2π, thus the natural phase space for (p, q) is the cylinder $\mathbb{R} \times [0, 2\pi]$, and the solution is actually periodic.

For the simplified Takahashi-Imada method, we observe a linear drift in Figure 4 and the energy is not well-conserved. This method has the same order 2 as the Störmer–Verlet method. It is a modification where $f(q)$ is replaced by $f(q+h^2/12\, f(q))$. The motivation for this modification is the integrator has improved effective order 4, i.e. there exists a change of coordinates χ_h such that $\chi_h^{-1} \circ \Phi_h \circ \chi_h$ is a method of order 4. This concept of effective order was first introduced by Butcher [But69] in the context of Runge-Kutta methods. The simplified Takahashi-Imada method is non-symplectic for all Hamiltonian systems and therefore does not satisfy the hypothesis of Theorem 0.1.3. Nevertheless, it is still a B-series symmetric method and it is volume-preserving. The numerical flow is thus symplectic for the pendulum problem because it is one-degree-of-freedom Hamiltonian system. This counter-example is taken from [HMS08], and the explanation of the non-conservation of energy is that the modified Hamiltonian is not globally defined on the cylinder: the integral on a period along the exact solution of the coefficient function $H_4(q, p)$ in the modified Hamiltonian is not zero. This simple example illustrates that symplecticity alone is not sufficient for a good long-time energy conservation.

0.2 New results

We describe here, chapter by chapter, the main ideas of the new results presented in this thesis.

Chapter 1: Modifying numerical integrators

Backward error analysis is a theoretical tool that gives much insight into the long-term integration with geometric numerical methods. We shall show that by simply exchanging the roles of the "numerical method" and the "exact solution" (cf. the two pictures in Figure 5), it can be turned into a means for constructing high order integrators that conserve geometric properties. They will be useful for integrations over long times.

Let us be more precise. As before, we consider an initial value problem (0.1) and a numerical integrator. But now we search for a modified differential equation $\dot{z} = \widetilde{f}_h(z)$, again of the form (0.6), such that the numerical solution $\{z_n\}$ of the method applied with

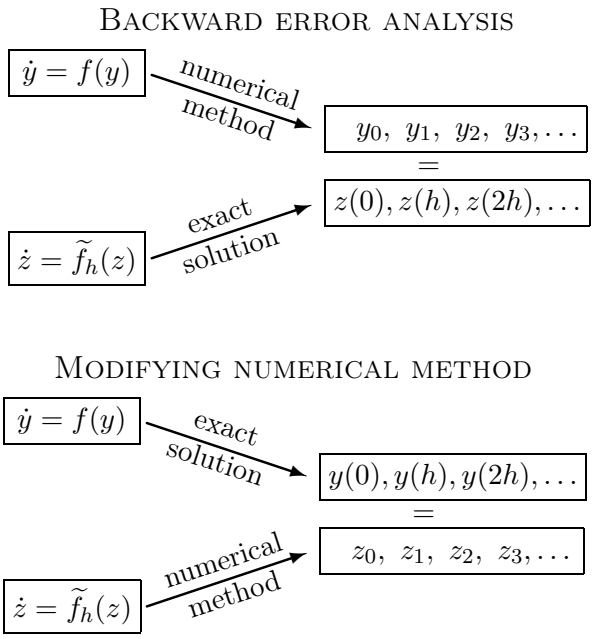


Figure 5: Backward error analysis opposed to modifying numerical integrators

stepsize h to (0.6) yields formally the exact solution

$$\Phi_{\widetilde{f}_h,h}(y) = \varphi_{f,h}(y) \tag{0.10}$$

of the original differential equation (0.1), i.e.,

$$z_n = y(nh) \qquad \text{for} \quad n = 0, 1, 2, \ldots, \tag{0.11}$$

see the bottom picture of Figure 5. Notice that this modified equation is different from the one considered before. However, due to the close connection with backward error analysis, all theoretical and practical results have their analogue in this new context. The modified differential equation is again an asymptotic series that usually diverges, and its truncation inherits geometric properties of the exact flow if a suitable integrator is applied. The coefficient functions $f_j(z)$ can be computed recursively by using a formula manipulation program like MAPLE. This can be done by developing both sides of $z(t+h) = \Phi_{\widetilde{f}_h,h}(z(t))$ into a series in powers of h, and by comparing their coefficients. Once a few functions $f_j(z)$ are known, the following algorithm arises naturally.

Algorithm 0.2.1 (modifying integrator) *Consider the truncation*

$$\dot{z} = f_h^{[r]}(z) = f(z) + hf_2(z) + \cdots + h^{r-1}f_r(z) \tag{0.12}$$

of the modified differential equation corresponding to $\Phi_{f,h}(y)$. Then,

$$z_{n+1} = \Psi_{f,h}(z_n) := \Phi_{f_h^{[r]},h}(z_n)$$

defines a numerical method of order r that approximates the solution of (0.1). We call it modifying integrator, because the vector field $f(y)$ of (0.1) is modified into $f_h^{[r]}$ before the basic integrator is applied.

This is an alternative approach for constructing high order numerical integrators for ordinary differential equations (classical approaches are multistep, Runge–Kutta, Taylor series, extrapolation, composition, and splitting methods). It is particularly interesting in the context of geometric integration because, as known from backward error analysis, the modified differential equation inherits the same structural properties as (0.1) if a suitable integrator is applied.

A few known methods can be cast into the framework of modifying integrators although they have not been constructed in this way. The most important are the generating function methods as introduced by Feng [Fen86]. These are high order symplectic integrators obtained by applying a simple symplectic method to a modified Hamiltonian system. The corresponding Hamiltonian is the solution of a Hamilton–Jacobi partial differential equation. The general approach of Algorithm 0.2.1 is introduced and discussed in [CHV07b].

Chap. 1–2: Analysis for B-series methods: a substitution law

The discrete flow of many numerical integrators (including Runge–Kutta methods) can be expanded into a B-series as introduced and studied in [HW74], see [HLW06, Chap. III].

Let $T = \{\bullet, \begin{smallmatrix}\bullet\\|\\\bullet\end{smallmatrix}, \begin{smallmatrix}\bullet\\|\\\bullet\\|\\\bullet\end{smallmatrix}, \begin{smallmatrix}\bullet\\|\\\bullet\\|\\\bullet\end{smallmatrix}, \ldots\}$ be the set of rooted trees, and let $\emptyset$ be the empty tree. For $\tau_1, \ldots, \tau_m \in T$, we denote by $\tau = [\tau_1, \ldots, \tau_m]$ the tree obtained by grafting the roots of τ_1, $\ldots$, τ_m to a new vertex which becomes the root of τ. Elementary differentials $F_f(\tau)$ are defined by induction as

$$F_f(\bullet)(y) = f(y), \qquad F_f(\tau)(y) = f^{(m)}(y)\big(F_f(\tau_1)(y), \ldots, F_f(\tau_m)(y)\big). \tag{0.13}$$

For real coefficients $a(\emptyset)$ and $a(\tau), \tau \in T$, a B-series is a series of the form

$$\begin{aligned} B(f,a) &= a(\emptyset)\, Id + \sum_{\tau \in T} \frac{h^{|\tau|}}{\sigma(\tau)} a(\tau) F_f(\tau) \\ &= a(\emptyset)\, Id + h a(\bullet)\, f + h^2 a(\begin{smallmatrix}\bullet\\|\\\bullet\end{smallmatrix})\, f'f + h^3 + h^3 a(\begin{smallmatrix}\bullet\\|\\\bullet\\|\\\bullet\end{smallmatrix})\, f''(f,f) + \ldots, \end{aligned}$$

where Id stands for the identity $Id\,(y) = y$ and the scalars $\sigma(\tau)$ are known normalization coefficients. The Taylor series of the exact solution of (1.1) can be written as a B-series $y(h) = B(f,e)(y_0)$ with coefficients $e(\tau)$. The flow $y_{n+1} = \Phi_{f,h}(y_n)$ of a Runge–Kutta method is of the form $\Phi_{f,h} = B(f,a)$ with $a(\tau)$ depending only on the coefficients of the method (see [HLW06, Chap. III] for more details).

With the aim of unifying the theory of modifying integrators with backward error analysis, we let (0.6) be the modified equation defined by

$$\Phi_{\widetilde{f}_h,h}(y) = \Psi_{f,h}(y) \tag{0.14}$$

where Φ and Ψ are two numerical integrators that can be expressed as B-series $\Phi_{f,h} = B(f,a)$ and $\Psi_{f,h} = B(f,c)$. For $\Psi_{f,h}(y) = \varphi_{f,h}(y)$ we recover formula (0.10) for modifying numerical integrators, and for $\Phi_{\widetilde{f}_h,h}(y) = \varphi_{\widetilde{f}_h,h}(y)$ we get (0.8) for backward error analysis.

In terms of B-series, formula (0.14) becomes $B(\widetilde{f}_h, a) = B(f,c)$. When computing recursively some of the coefficient functions of (1.2), one is quickly convinced that they are linear combinations of elementary differentials and that $\widetilde{f}_h(y) = h^{-1}B(f,b)(y)$ with coefficients $b(\tau)$ that have to be determined (notice that we necessarily have $b(\emptyset) = 0$). This motivates the following theorem, introduced in [CHV05].

Theorem 0.2.2 *For $b(\emptyset) = 0$, the vector field $h^{-1}B(f,b)$ inserted into $B(\cdot,a)$ gives a B-series*

$$B\big(h^{-1}B(f,b),a\big) = B(f,b\star a).$$

We have $(b\star a)(\emptyset) = a(\emptyset)$, some further coefficients are given in Table 1.2 below, and a general formula for $(b\star a)(\tau)$ is given in (1.27) of Sect. 1.4.

Sketch of proof. To illustrate Theorem 0.2.2, we now compute by hand the coefficients obtained by the substitution law, for trees up to order 3. We consider a B-series

$$\begin{aligned} B(g,a)(y) &= a(\emptyset)y + ha(\bullet)g(y) + h^2a(\bullet\!\!-\!\!\bullet)g'(y)g(y) + \frac{h^3}{2}a(\bullet^{\bullet\ \bullet})g''(y)(g(y),g(y)) \\ &\quad + h^3a(\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet)g'(y)g'(y)g(y) + \dots \end{aligned} \tag{0.15}$$

where the vector field g is replaced by a B-series $g = h^{-1}B(f,b)$. Computing each term individually and omitting the argument (y) leads to

$$\begin{aligned} hg &= hb(\bullet)f + h^2b(\bullet\!\!-\!\!\bullet)f'f + \frac{h^3}{2}b(\bullet^{\bullet\ \bullet})f''(f,f) + h^3b(\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet)f'f'f + \dots \\ h^2g'g &= h^2(b(\bullet)f + hb(\bullet\!\!-\!\!\bullet)f'f + \dots)'(b(\bullet)f + hb(\bullet\!\!-\!\!\bullet)f'f + \dots) \\ &= h^2b(\bullet)^2f'f + 2h^3b(\bullet)b(\bullet\!\!-\!\!\bullet)f'f'f + h^3b(\bullet\!\!-\!\!\bullet)b(\bullet)f''(f,f) + \dots \\ \frac{h^3}{2}g''(g,g) &= \frac{h^3}{2}(b(\bullet)f + \dots)''(b(\bullet)f + \dots, b(\bullet)f + \dots) \\ &= \frac{h^3}{2}b(\bullet)^3f''(f,f) + \dots \\ h^3g'g'g &= h^3(b(\bullet)f + \dots)'(b(\bullet)f + \dots)'(b(\bullet)f + \dots) \\ &= h^3b(\bullet)^3f'f'f + \dots \end{aligned}$$

We then substitute expressions of hg, $h^2g'g$, $h^3g'g'g$, $\frac{h^3}{2}g''(g,g)$ into (0.15), and collect terms in hf, $h^2f'f$, $h^3f'f'f$, $\frac{h^3}{2}f''(f,f)$. This gives

$$\begin{aligned} B(g,a)(y) &= a(\emptyset) + ha(\bullet)b(\bullet)f + h^2\Big(a(\bullet)b(\bullet\!\!-\!\!\bullet) + a(\bullet\!\!-\!\!\bullet)b(\bullet)^2\Big)f'f \\ &\quad + \frac{h^3}{2}\Big(a(\bullet)b(\bullet^{\bullet\ \bullet}) + 2a(\bullet\!\!-\!\!\bullet)b(\bullet)b(\bullet\!\!-\!\!\bullet) + a(\bullet^{\bullet\ \bullet})b(\bullet)^3\Big)f''(f,f) \\ &\quad + h^3\Big(a(\bullet)b(\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet) + 2a(\bullet\!\!-\!\!\bullet)b(\bullet)b(\bullet\!\!-\!\!\bullet) + a(\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet)b(\bullet)^3\Big)f'f'f + \dots \\ &= B(f,b\star a)(y) \end{aligned}$$

We obtain the substitution law for the first few trees:

$$\begin{aligned} (b\star a)(\emptyset) &= a(\emptyset) \\ (b\star a)(\bullet) &= a(\bullet)b(\bullet) \\ (b\star a)(\bullet\!\!-\!\!\bullet) &= a(\bullet)b(\bullet\!\!-\!\!\bullet) + a(\bullet\!\!-\!\!\bullet)b(\bullet)^2 \\ (b\star a)(\bullet^{\bullet\ \bullet}) &= a(\bullet)b(\bullet^{\bullet\ \bullet}) + 2a(\bullet\!\!-\!\!\bullet)b(\bullet)b(\bullet\!\!-\!\!\bullet) + a(\bullet^{\bullet\ \bullet})b(\bullet)^3 \\ (b\star a)(\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet) &= a(\bullet)b(\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet) + 2a(\bullet\!\!-\!\!\bullet)b(\bullet)b(\bullet\!\!-\!\!\bullet) + a(\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet)b(\bullet)^3 \end{aligned} \tag{0.16}$$

The question of finding the modified equation defined by (0.14), i.e., of finding the coefficients $b(\tau)$ for given $a(\tau)$ and $c(\tau)$ in the relation

$$B\big(h^{-1}B(f,b),a\big) = B(f,c),$$

comes to solving for $b(\tau)$ the algebraic system

$$(b \star a)(\tau) = c(\tau) \qquad \text{for} \quad \tau \in T. \tag{0.17}$$

We notice that

$$(b \star a)(\tau) = a(\bullet)b(\tau) + \cdots + a(\tau)b(\bullet)^{|\tau|},$$

where the three dots involve only trees of order strictly less than $|\tau|$. Consequently, for consistent integrators $\Phi_{f,h} = B(f,a)$ and $\Psi_{f,h} = B(f,c)$, for which $a(\emptyset) = a(\bullet) = 1$ and $c(\emptyset) = c(\bullet) = 1$, the coefficients $b(\tau)$ can be computed recursively from (0.17). In this way, the computation of the vector fields $f_j(y)$ in the modified differential equation (0.6) or (0.12) is reduced to that of real coefficients.

Modifying integrators. In this case $\Psi_{f,h}$ in (0.14) is the exact h-flow which is a B-series with coefficients $e(\tau)$. Consequently, the coefficients $b(\tau)$ of the modified differential equation for $\Phi_{f,h} = B(f,a)$ are obtained from

$$(b \star a)(\tau) = e(\tau) \qquad \text{for} \quad \tau \in T.$$

Backward error analysis. The modified differential equation of a method $\Psi_{f,h} = B(f,c)$ is obtained by putting $\Phi_{f,h}$ equal to the exact flow. Its coefficients $b(\tau)$ are therefore obtained from

$$(b \star e)(\tau) = c(\tau) \qquad \text{for} \quad \tau \in T.$$

A group on B-series The B-series $h^{-1}B(f,b)$ corresponding to mappings $b : T \cup \{\emptyset\} \to \mathbb{R}$ with $b(\emptyset) = 0$ represent vector fields made of elementary differentials of f. The product $b \star a$ defines a group structure on the set $\big\{c : T \cup \{\emptyset\} \to \mathbb{R}\,;\, c(\emptyset) = 0, c(\bullet) = 1\big\}$ which represents such vector fields. Its unit element is given by $c(\bullet) = 1$ and $c(\tau) = 0$ for $|\tau| > 1$, and it corresponds to the original vector field $f(y)$.

In Chapter 2, we give further algebraic properties of the substitution law on B-series. We explain the common algebraic structure of two composition laws on B-series: the Butcher composition, which corresponds to the composition of integrators, and the substitution law, introduced in the previous chapter, which corresponds to the composition of B-series vector fields.

Hopf algebra structures on rooted trees are well-studied object, especially in the context of combinatorics, and are essentially characterized by the coproduct map. It is well-known that the first composition law corresponds to the convolution product on the Hopf tree algebra of Connes & Kreimer in renormalization in quantum field theory. It was shown recently by Calaque, Ebrahimi-Fard & Manchon [CEFM08], in the context on combinatory algebra, that the substitution law on B-series can be turned into a new coproduct Δ_{CEM}, which allows to build another Hopf tree algebra, e.g. (compare with (0.16))

$$\Delta_{CEM}(\;) = \;\otimes \bullet + 2\bullet\;\otimes\;+ \bullet^3 \otimes\;$$

We prove that this new composition law is compatible with the standard composition of B-series,

$$B(f,a)\Big(B(f,b)(y)\Big) = B(f, b \cdot a)(y).$$

For instance, we have the distributivity relation

$$b \star (a \cdot c) = (b \star a) \cdot (b \star c).$$

We also show that the subgroup of symplectic B-series (for the standard composition of B-series) is in one-to-one correspondence via backward error analysis with the subgroup of Hamiltonian B-series equipped with the substitution law.

Finally, we explain the extension of the presented theory to partitioned integration methods (P-series). This is particularly important for the consideration of symplectic integrators.

Chapter 3: A high-order integrator for the motion of a rigid body

As illustration of how efficient modifying integrators can be, we consider the equations of motion for a free rigid body, which are determined by a Hamiltonian system constrained to the Lie group $SO(3)$,

$$\dot{y} = \widehat{y}\, I^{-1} y, \qquad \dot{Q} = Q\, \widehat{I^{-1}y}, \qquad \text{where} \qquad \widehat{a} = \begin{pmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{pmatrix} \tag{0.18}$$

for a vector $a = (a_1, a_2, a_3)^T$. Here, $I = \mathrm{diag}(I_1, I_2, I_3)$ is the matrix formed by the moments of inertia, y is the vector of the angular momenta, and Q is the orthogonal matrix that describes the rotation relative to a fixed coordinate system. As numerical integrator we choose the Discrete Moser–Veselov algorithm (DMV) [MV91],

$$\widehat{y}_{n+1} = \Omega_n\, \widehat{y}_n\, \Omega_n^T, \qquad Q_{n+1} = Q_n\, \Omega_n^T, \tag{0.19}$$

where the orthogonal matrix Ω_n is computed from

$$\Omega_n^T D - D\, \Omega_n = h\, \widehat{y}_n. \tag{0.20}$$

Here, the diagonal matrix $D = \mathrm{diag}(d_1, d_2, d_3)$ is determined by $d_1 + d_2 = I_3$, $d_2 + d_3 = I_1$, and $d_3 + d_1 = I_2$. This algorithm is an excellent geometric integrator and shares many geometric properties with the exact flow. It is symplectic, it exactly preserves the Hamiltonian, the Casimir and the angular momentum Qy (in the fixed frame), and it keeps the orthogonality of Q. Its only disadvantage is the low order two.

The technique of modifying integrators cannot be directly applied to increase the order of this method, because the algorithm (0.19) is not defined for general problems (0.1). It is, however, defined for arbitrary I_j, and therefore we look for modified moments of inertia $\widetilde{I}_j$ such that the DMV algorithm applied with $\widetilde{I}_j$ yields the exact solution of (0.18). It is shown in [HV06] that this is possible with

$$\frac{1}{\widetilde{I}_j} = \frac{1}{I_j}\Big(1 + h^2 s_3(y_n) + h^4 s_5(y_n) + \cdots\Big) + h^2 d_3(y_n) + h^4 d_5(y_n) + \cdots . \tag{0.21}$$

The expressions $s_k(y)$ and $d_k(y)$ can be computed by a formula manipulation package similar as the modified differential equation is obtained. The first of them are

$$\begin{aligned} s_3(y_n) &= -\frac{1}{3}\left(\frac{1}{I_1} + \frac{1}{I_2} + \frac{1}{I_3}\right) H(y_n) + \frac{I_1 + I_2 + I_3}{6\, I_1\, I_2\, I_3}\, C(y_n), \\ d_3(y_n) &= \frac{I_1 + I_2 + I_3}{6\, I_1\, I_2\, I_3}\, H(y_n) - \frac{1}{3\, I_1\, I_2\, I_3}\, C(y_n), \end{aligned}$$

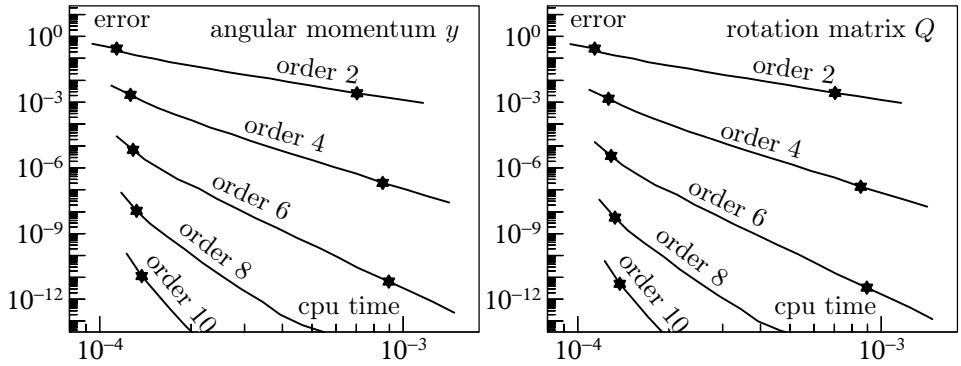


Figure 6: Work-precision diagram for the DMV algorithm (order 2) and for the modifying DMV integrators of orders 4, 6, 8, and 10.

where

$$C(y) = \frac{1}{2}\Big(y_1^2 + y_2^2 + y_3^2\Big) \qquad \text{and} \qquad H(y) = \frac{1}{2}\Big(\frac{y_1^2}{I_1} + \frac{y_2^2}{I_2} + \frac{y_3^2}{I_3}\Big)$$

are the Casimir and the Hamiltonian of the system. The physical interpretation of this result is the following: after perturbing suitably the form of the body, an application of the DMV algorithm yields the exact motion of the body. Truncating the series in (0.21) after the h^{2r-2} terms, yields a modifying DMV algorithm of order $2r$.

Numerical experiment We consider an asymmetric rigid body with moments of inertia $I_1 = 0.6$, $I_2 = 0.8$, and $I_3 = 1.0$ on the interval $[0, 10]$. Initial values are $y(0) = (1.8, 0.4, -0.9)^T$ and $Q(0)$ is the identity matrix. The implementation of the modifying DMV algorithm is done using quaternions as explained in [HV06]. Although $H(y)$ and $C(y)$ are constant along the numerical solution, we recompute the values of $\widetilde{I}_j$ in every step to simulate the presence of an external potential.

We apply the DMV algorithm and its extensions to order 4, 6, 8, and 10 with many different stepsizes, and we plot in Figure 6 the global error at the endpoint as a function of the cpu times. The execution times are the average of 1000 experiments. The symbols indicate the values obtained with the stepsizes $h = 0.1$ and $h = 0.01$, respectively.

The pictures nicely illustrate the expected orders of the algorithms (order p corresponds to a straight line with slope $-p$). Much more interesting is the fact that high accuracy is obtained more or less for free. Consider the results obtained with stepsize $h = 0.1$. The error for the DMV algorithm (order 2) is more than 20%. With very little extra work, the modification of order 10 gives an accuracy of more than 11 digits with the same stepsize.

We also study the propagation with time of round-off error and explain how it can be reduced, so that round-off behaves like a random walk. We compare with the integrators based on Jacobi elliptic functions.

Conjugate points of rigid body geodesics In [BF07], the conjugate locus (i.e. the set of conjugate points) of rigid body geodesics is studied in the case where two moments of inertia are equal (e.g. $I_2 = I_3$), and the general case is currently investigated in [BF07].

With motivation of computing conjugate points of asymmetric rigid body geodesics, we give an accurate algorithm for the computation of the derivatives of the rigid flow with respect to the initial conditions. This is called the tangent map:

$$\frac{\partial y(t)}{\partial y_0}, \qquad \frac{\partial Q(t)}{\partial y_0}.$$

We show that the derivatives of $Q(t)$ can be conveniently approximated in the form

$$\frac{\partial Q_n}{\partial y_{0,j}} = Q_n \widehat{a}_{n,j}, \quad j = 1, 2, 3$$

where the $\widehat{a}_{n,j}$'s are skew-symmetric matrices. Then, conjugate points are simply obtained when the 3×3 matrix whose columns are the vectors $a_{n,j}$ becomes singular.

The idea of the computation is to differentiate with respect to initial conditions the high-order discretization of the preprocessed Discrete Moser–Veselov algorithm. We show that it can be efficiently implemented.

Chapter 4: The role of symplectic integrators in optimal control

We consider an optimal control problem of the form

$$(P) \begin{cases} \text{Min } \Phi(x(1)), \\ \dot{x}(t) = f(x(t), u(t)), \quad t \in (0,1), \\ x(0) = x^0, \end{cases}$$

where $f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ and $\Phi : \mathbb{R}^n \to \mathbb{R}$ are two smooth functions (C^∞), and we assume for simplicity that the state $x(t) : (0,1) \to \mathbb{R}^n$ and the control function $u(t) : (0,1) \to \mathbb{R}^m$ are continuous. The necessary optimality condition given by the Pontryagin Maximum principle, a major tool in optimal control (see e.g. [Eva83, MS82]) are the following. There exists a co-state function $p : (0,1) \to \mathbb{R}^n$ such that the solution of (P) is solution of a boundary value problem,

$$(OC) \begin{cases} \dot{x}(t) = H_p(x(t), p(t), u(t)) \\ \dot{p}(t) = -H_x(x(t), p(t), u(t)) \\ H(x(t), p(t), u(t)) = \min_{\alpha \in A} H(x(t), p(t), \alpha) \\ x(0) = x^0, \quad p(1) = \Phi'(x(1)). \end{cases}$$

for $t \in (0,1)$, where the Hamiltonian function $H : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ is defined by

$$H(x, p, u) = p^T f(x, u).$$

Furthermore, the Hamiltonian is conserved, i.e. $H(x(t), p(t), u(t))$ is constant along the solution of (OC).

The work of [Hag00, BLV06] shows that applying a Runge-Kutta discretization directly to the optimal control problem (P) is equivalent to applying a partitioned symplectic Runge–Kutta discretizations to the Hamiltonian system in the Pontryagin formulation of the optimal control problem.

For instance, consider the *explicit Euler method* with $h = 1/N$, and $x(t_k) \approx x_k$, $t_k = kh$:

$$\begin{cases} \text{Min } \Phi(x_N), \\ x_{k+1} = x_k + h f(x_k, \overline{u}_k), \quad k = 0, \dots N-1 \\ x_0 = x^0. \end{cases}$$

By introducing Lagrange multipliers, this discretization is equivalent to apply a symplectic partitioned Runge-Kutta method, here the *symplectic Euler method*:

$$\begin{cases} x_{k+1} = x_k + h f(x_k, \overline{u}_k), \\ p_{k+1} = p_k - h p_{k+1}^T f_x(x_k, \overline{u}_k), \\ 0 = p_{k+1}^T f_u(x_k, \overline{u}_k), \text{ i.e } \overline{u}_k = \varphi(x_k, p_{k+1}), \\ x_0 = x^0, \quad p_N = \Phi'(x_N). \end{cases}$$

with $k = 0, \dots N-1$. It is shown in [Hag00, BLV06] that this is true for all Runge-Kutta discretizations.

The aim of Chapter 4 is to investigate to which extent the excellent performance of symplectic integrators for long time integrations in astronomy and molecular dynamics carries over to problems in optimal control. We first study the Martinet case in sub-Riemannian geometry. After elimination of the control, using the Pontryagin maximum principle, we arrive at the Hamiltonian

$$H(q,p) = \frac{1}{2}\left(\left(p_x + p_z\frac{y^2}{2}\right)^2 + \frac{p_y^2}{(1+\beta x)^2}\right).$$

where $q = (x, y, z)^T$ is the state, and $p = (p_x, p_y, p_z)^T$ is the adjoint state. The interesting dynamics takes place in the two-dimensional space of coordinates (y, p_y). Using the theory of backward error analysis, we show that symplectic integrators have a clear advantage for the integrable Martinet case where $\beta = 0$ (see Figure 7) and also a non integrable perturbation ($\beta = -10^{-4}$) (see Figure 8), even if long-time integration is not an issue here.

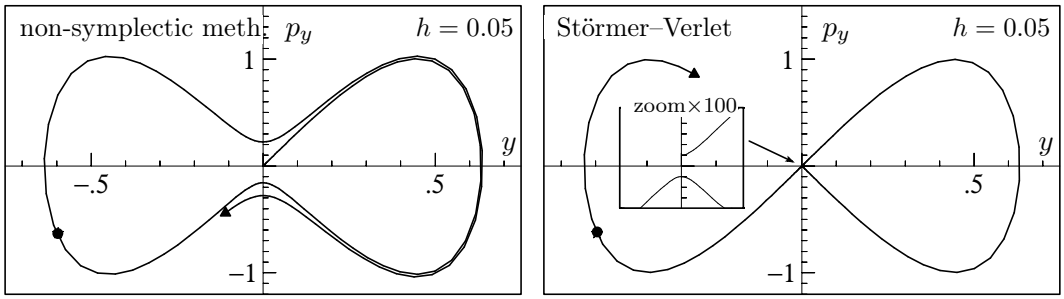


Figure 7: Phase portraits in the (y, p_y)-plane for the flat case $\beta = 0$.

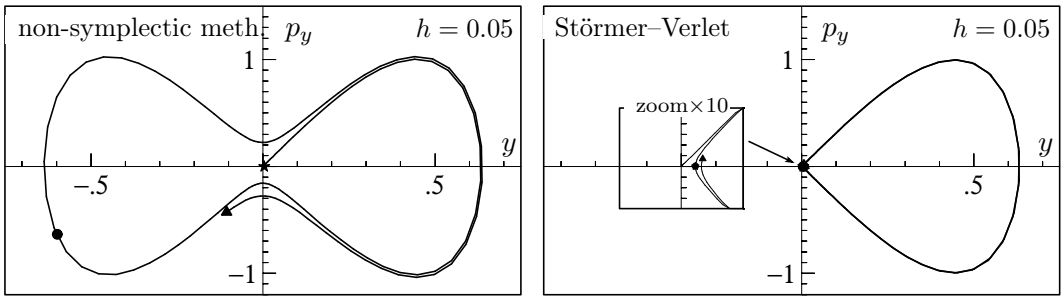


Figure 8: Phase portraits in the (y, p_y)-plane for the non integrable case $\beta = -10^{-4}$.

Nevertheless, for problems like the orbital transfer of a spacecraft or the control of a submerged rigid body such an advantage cannot be observed. The Hamiltonian system is a boundary value problem and the time interval is in general not large enough so that symplectic integrators could benefit from their structure preservation of the flow.

Backward error analysis for optimal control problems? We also discuss whether it is possible to extend the theory of backward error analysis to symplectic integrators for optimal control. We show that this is possible for linear quadratic optimal control problems, with state $x(t) \in \mathbb{R}^n$, and control $u(t) \in \mathbb{R}^m$,

$$\begin{cases} \text{Min } \frac{1}{2}\int_0^1 (x^T Z x + u^T S u)dt, \\ \dot{x} \;=\; Ax + Bu \\ x(0) \text{ given} \end{cases}$$

where $A, Z \in \mathbb{R}^{n\times n}$, with Z symmetric, and $S \in \mathbb{R}^{m\times m}$ is symmetric positive definite, and $B \in \mathbb{R}^{n\times m}$ has rank m. Precisely, we prove that the numerical solution obtained by applying a symplectic method (e.g. a symplectic partitioned Runge-Kutta method) to the Hamiltonian system in the Pontryagin formulation formally yields the exact solution of a *modified control problem*. In general, this perturbed control problem is no longer an optimal control problem, but can be interpreted as a stationary control problem, or also a min – max control problem, where we possibly add extra control functions. For instance, consider the implicit midpoint rule for the optimal control problem

$$\left\{\begin{array}{ll} \text{Min } \frac{1}{2}\int_0^1 (x_1^2+u_1^2)dt, & \\ \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -x_1+u_1 \\ x_1(0), x_2(0) \text{ given} & \end{array}\right.$$

The numerical solution can be interpreted as the exact solution of the following perturbed control problem, where we add an extra control u_2.

$$\left\{\begin{array}{ll} \min\limits_{u_1}\max\limits_{u_2} \frac{1}{2}\int_0^1 \big(x_1^2+u_1^2-h^2u_2^2+\frac{h^2}{12}(-2x_1^2-x_2^2)+\ldots\big)dt, & \\ \dot{x}_1 &= x_2+\frac{h^2}{\sqrt{12}}u_2-\frac{h^2}{12}x_2+\ldots \\ \dot{x}_2 &= -x_1+u_1-\frac{h^2}{12}u_1+\ldots \\ x_1(0), x_2(0) \text{ given} & \end{array}\right.$$

This result has an interpretation in game theory, where the first player controls u_1 to minimize the cost function, while the second player controls u_2 and tries to maximize the cost.

Chapter 5: Splitting methods based on modified potentials

In Chapter 5, we consider splitting methods for perturbed Hamiltonian systems of the form $H = H^A + H^B$, see the survey [MQ02]. The vector field $f(x) = J^{-1}\nabla H(x)$ is split as $f(x) = A(x) + B(x)$, and we assume that the flows of vector fields $A = J^{-1}\nabla H^A$ and $B = J^{-1}\nabla H^B$ can be approximated efficiently either exactly or with high-accuracy. A standard approach for this type of problem is to consider splitting methods of the form

$$e^{a_m hA}e^{b_m hB}e^{a_{m-1}hA}e^{b_{m-1}hB}\cdots e^{a_1 hA}e^{b_1 hB}$$

where e^{hA} and e^{hB} denote the flows associated to A and B. We also consider splitting methods with modified potentials, e.g.

$$\widetilde{B} = B + h^2 C$$

where $C = [B,[B,A]]$ involves Lie-brackets. In the context of geometric integration, this kind of integrator is of great interest because it preserves qualitative properties of the exact solution. Indeed, when A and B are two Hamiltonian vector fields, all the flows $e^{a_i hA}$ and $e^{b_i hB}$ are symplectic, and the resulting splitting method is symplectic as a composition of symplectic flows, which guarantees the correct conservation of energy over exponentially long times.

A significant improvement, to reduce the number of compositions, and thus the computational cost, is to consider processed methods. In order to reduce the number of evaluations

per step in the integration, the idea of processing, first introduced by Butcher [But69] in the context of Runge-Kutta methods, is to consider a composition of the form

$$e^P e^{hK} e^{-P}$$

where e^{hK} is called the Kernel and should be cheap, and the order of $e^P e^{hK} e^{-P}$, called effective order, is higher than that of e^{hK}. Using a constant stepsize h, after N steps, we obtain $e^P (e^{hK})^N e^{-P}$. At first, we apply the processor (or corrector) e^{-P}, then e^{hK} once per step, and the postprocessor e^P is evaluated only when output is needed. A general analysis of symplectic splitting methods with processing in given in [BCR99].

In practice, the main tool for the derivation of order conditions for splitting methods is the Baker-Campell-Hausdorff (BCH) formula (see e.g. [HLW06, Sect. III.4.2]) which implies that the local error for these methods is formally a linear combination of Lie bracket terms in the Lie-algebra generated by the vector fields A and B.

Now, assume that the vector field B is a small perturbation of vector field A, i.e.

$$B = \mathcal{O}(\varepsilon)$$

where ε is a small parameter. In this case, Lie brackets involving few B's are dominant and should be canceled in priority to reduce the error of the method. For instance, $[A,[A,B]] = \mathcal{O}(\varepsilon)$ is dominant compared to $[B,[B,A]] = \mathcal{O}(\varepsilon^2)$. The idea of processing has been applied to the symplectic integration of near-integrable Hamiltonian systems in [WHT96, McL96].

These methods are called 'Runge-Kutta Nyström methods' in [BCR01] because they were introduced in the context of second order differential equations $\ddot{x} = f(x)$. However, the class of methods applies not only to second order differential equations. (e.g. the N-body problems in Jacobi coordinates as studied in [WH91]).

The main contribution of this chapter is the construction of a new processor for the Takahashi–Imada method (a modification [Row91, TI86] of the Stang splitting),

$$e^{\frac{h}{2}B - \frac{h^3}{48}C} e^{hA} e^{\frac{h}{2}B - \frac{h^3}{48}C},$$

to achieve order $\mathcal{O}(h^{10}\varepsilon + h^4\varepsilon^2)$. We also show that this class of methods can be successfully applied to asymmetric rigid body problems with an external potential:

- the asymmetric heavy top (linear external potential) ;
- a satellite simulation (quadratic external potential) ;
- a molecular dynamics simulation: this is a N-body problem where N water molecules are modelised as asymmetric rigid bodies and interact as dipolar magnetic soft sphere.

Our numerical experiments show that this method is very efficient for small ε when the cost of evaluating the vector field $C = [B,[B,A]]$ together with B is small compared to the cost of evaluating of A and B alone.

Chapter 6: Splitting methods involving complex coefficients for parabolic equations

The last chapter is devoted to splitting methods involving complex coefficients for parabolic equations.

Although the numerical simulation of the Heat equation in several space dimensions is now well understood, there remain a lot of challenges in the presence of an external source,

e.g. for reaction-diffusion problems, or more generally for the complex Ginzburg-Landau equation. From a mathematical point of view, they belong to the class of semi-linear parabolic partial differential equations and can be represented in the general form

$$\frac{\partial u}{\partial t} = D\Delta u + F(u),$$

where each component of the vector $u(x,t) \in \mathbb{R}^d$ represents the population of one species, D is the matrix of diffusion coefficients (often diagonal) and F accounts for all local interactions between species. The solutions of reaction-diffusion equations display a wide range of behaviours, like traveling waves and wave-like phenomena or dissipative solitons.

For the sake of simplicity, let us illustrate the method on the linear case

$$\frac{\partial u}{\partial t} = \Delta u + Vu, \tag{0.22}$$

where V is a linear operator, say $Vu = v(x)u$ with $v(x)$ a smooth function. Splitting methods basically rely on the identity

$$e^{h(\Delta+V)} = e^{h\Delta}\, e^{hV} + \mathcal{O}\left(h^2\right),$$

or on higher order approximations obtained by combining $e^{h\Delta}$ and e^{hV} in the appropriate fashion. Dividing time t into n time steps of size h (where $t = nh$), the above approximation indeed leads to the equality

$$u(t) = e^{t(\Delta+V)}\, u(0) = e^{nh(\Delta+V)}\, u(0) = \left(e^{h\Delta}\, e^{hV}\right)^n u(0) + \mathcal{O}(h).$$

The extension to the non-linear case is straightforward, replacing e^{hV} by the flow of a nonlinear differential equation.

For a positive stepsize h, the most simple numerical integrator is the Lie-Trotter splitting

$$e^{hV} e^{h\Delta} \tag{0.23}$$

which is an approximation of order 1 of the solution of (0.22), while the symmetric version

$$e^{h/2\,V} e^{h\Delta} e^{h/2\,V} \tag{0.24}$$

is referred to as the Strang splitting and is an approximation of order 2. For higher orders, one can consider general splitting methods of the form

$$e^{b_1 hV} e^{a_1 h\Delta} e^{b_2 hV} e^{a_2 h\Delta} \dots e^{b_s hV} e^{a_s h\Delta}. \tag{0.25}$$

However, achieving higher order is not as straightforward as it looks. A disappointing result indeed shows that all splitting methods (or composition methods) with real coefficients must have negative coefficients a_i and b_i in order to achieve order 3 or more. The existence of at least one negative coefficient was shown in [She89, SW92], and the existence of a negative coefficient for both operators was proved in [GK96]. An elegant geometric proof can be found in [BC05]. As a consequence, such splitting methods *cannot* be used when one operator, like Δ, is not time-reversible.

In order to circumvent this order-barrier, there are two possibilities. One can use linear, convex combinations (see [GRT02, GRT04] for methods of order 3 and 4) or non-convex combinations (see [Sch02, Des01] where an extrapolation procedure is exploited) of elementary splitting methods like (0.25). Another possibility is to consider splitting methods with

complex coefficients a_i and b_i with positive real parts. In 1962/1963, Rosenbrock [Ros63] considered complex coefficients in a similar context.

It is interesting to note that raising the order can also be achieved by considering composition methods of the form

$$\Psi_h := \Phi_{\gamma_s h} \circ \ldots \circ \Phi_{\gamma_1 h}, \tag{0.26}$$

where Φ_h is a low order approximation. Symmetry can even be obtained by imposing $\gamma_j = \gamma_{s+1-j}$ $(1 \leq j \leq s)$, and by choosing Φ_h symmetric. For instance, when Φ_h is the Strang splitting (0.24), this approach leads to

$$\Psi_h = e^{h\gamma_s/2\,V} e^{h\gamma_s \Delta} e^{h(\gamma_s+\gamma_{s-1})/2\,V} e^{h\gamma_{s-1}\Delta} \ldots e^{h\gamma_1 \Delta} e^{h\gamma_1/2\,V}.$$

The advantage of the approach with composition methods is that we can replace the Strang splitting with exponential maps (0.24) by a symmetric discretization, for instance,

$$\Phi_h = \Phi^I_{h/2} \circ \Phi^M_h \circ \Phi^E_{h/2}$$

where Φ^E_h denotes the flow of the explicit Euler method $y_{n+1} = y_n + hf(y_n)$ and Φ^I_h denotes the flow of the implicit Euler method $y_{n+1} = y_n + hf(y_{n+1})$ for the approximation of the reaction, and Φ^M_h is the Crank-Nicholson discretization (which is equivalent to the implicit midpoint rule for linear systems)

$$\Phi^M_h = \left(Id - \frac{h}{2}\Delta\right)^{-1} \left(Id + \frac{h}{2}\Delta\right).$$

This is called the Peaceman-Rachford formula [PJ55] originally developed for the Heat equation, and extended to reaction-diffusion problems in [DR03].

What is new in this chapter is that we consider splitting methods of the form (0.26), and we derive new high-order methods using composition techniques originally developed for the geometric numerical integration of ordinary differential equations [HLW06]. The main advantages of this approach are the following:

– the splitting method inherits the stability property of exponential operators;

– we can replace the costly exponentials of the operators by cheap low order approximations without altering the overall order of accuracy;

– using complex coefficients allows to reduce the number of compositions needed to achieve any given order;

Our numerical simulations show that the order of accuracy is the one expected especially in case of a non-linear source, and for the Peaceman-Rachford discretization.

Chapter 1

Numerical integrators based on modified differential equations

Note: This chapter is identical to the article [CHV07b] in collaboration with P. Chartier and E. Hairer.

For an accurate numerical integration of a system of differential equations

$$\dot{y} = f(y), \qquad y(0) = y_0 \tag{1.1}$$

it is important to use methods of high order (say, at least order 4). Classical approaches for getting high order are multistep, Runge–Kutta, Taylor series, extrapolation, composition, and splitting methods. In this article we present a new approach for constructing high order methods by using modified differential equations.

The idea is the following: for a given one-step method $y_{n+1} = \Phi_{f,h}(y_n)$ (typically very simple to implement, and of order 1 or 2), find a modified differential equation, written as a formal series in powers of the stepsize h,

$$\dot{y} = \widetilde{f}(y) = f(y) + hf_2(y) + h^2 f_3(y) + \cdots, \qquad y(0) = y_0, \tag{1.2}$$

such that the numerical solution of the method Φ_h applied to the modified differential equation (1.2) yields the exact solution of (1.1) in the sense of formal power series, i.e.,

$$\Phi_{\widetilde{f},h}(y) = \varphi_{f,h}(y). \tag{1.3}$$

Here, $\varphi_{f,t}(y)$ denotes the exact time-t flow of the problem $\dot{y} = f(y)$.

Once a few coefficient functions $f_j(y)$ are known, this permits us to construct high order integration methods for (1.1). We suggest the name *modifying integrators* for this approach, because the vector field (1.1) is modified into (1.2) before the basic method is applied.

Modifying integrator. *For $r > 1$, consider the truncation*

$$\dot{y} = f^{[r]}(y) = f(y) + hf_2(y) + \cdots + h^{r-1} f_r(y) \tag{1.4}$$

of the modified equation (1.2) for which (1.3) holds. Then,

$$y_{n+1} = \Phi_{f^{[r]},h}(y_n) \tag{1.5}$$

defines a numerical method of order r for (1.1).

An intrinsic feature of this approach is that geometric properties of the flow of (1.1) which are conserved by the basic method, are in general retained by the high order modifying integrator (see Sect. 1.1 below).

There are a few methods that can be cast into the framework of modifying integrators. This is the case for the generating function methods of Feng Kang [Fen86], Feng, Wu, Qin & Wang [FWQW89], and Channel & Scovel [CS90]. There, Hamiltonian systems $f(y) = J^{-1}\nabla H(y)$ in canonical form are considered together with simple symplectic integrators (e.g., symplectic Euler method, or the implicit midpoint rule). It turns out that the modified differential equation is Hamiltonian and can be obtained as formal solution of the Hamilton–Jacobi partial differential equation (see [HLW06, Sect. VI.5.4]). A recent modification by McLachlan & Zanna [MZ05] of the discrete Moser–Veselov algorithm for solving the Euler equations for the free rigid body can also be interpreted as a modifying method (although it is not constructed in this way).

Modifying integrators will be efficient when the evaluation of the truncated vector field in (1.4) is not much more expensive than that of $f(y)$. This is definitely the case for the equations of motion for the full dynamics of a rigid body (see Sect. 1.2). We shall see later in Sect. 1.3 that the coefficient functions $f_j(y)$ depend on derivatives of $f(y)$. McLachlan [McL07] discusses situations (N-body problems, lattice systems) where the computation of derivatives is cheap when it is performed together with the evaluation of $f(y)$. In these situations the modifying integrators have a large potential.

This paper is organized as follows: the construction of the modified differential equation (1.2) is discussed in Sect. 1.1, where also some important geometric properties are presented. As an example of modifying integrators, a new efficient high-order method (based on the implicit midpoint rule) is developed in Sect. 1.2 for the motion of a free rigid body. Many numerical one-step methods (e.g., all Runge–Kutta and Taylor series methods) can be represented as a B-series. For this case, a substitution law for B-series is introduced, which yields general formulae for the modified equation (Sect. 1.3), with technical details postponed to Sect. 1.4.

1.1 The modified differential equation

We explain the construction of the modified equation (1.2), and we discuss how the modified equation inherits the geometric properties of the numerical integrator.

1.1.1 Construction of the modified equation

In the following, we assume that the vector field of (1.1) is infinitely differentiable, and that the numerical integrator $\Phi_{f,h}$ is smooth in h and in f, and of order at least one.

If the basic integrator $\Phi_{f,h}(y)$ is well-defined for all smooth vector fields $f(y)$, then one can simply develop both sides of (1.3) into a Taylor series around $h = 0$. A comparison of equal powers of h then yields recursively the functions $f_j(y)$ of the modified differential equation (1.2). This can conveniently be done with a formula manipulation program like MAPLE.

It may happen that the basic integrator is only defined for a subclass of differential equations (e.g., the Discrete Moser–Veselov algorithm for the motion of a free rigid body, cf. [HV06]). In this case, the following recursive construction is in general possible. Suppose that the functions $f_j(y)$ are known for $j = 1, \dots, r$ (we use $f_1(y) = f(y)$). If the basic method is well-defined for the vector field $f^{[r]}(y)$ of (1.4) (this is certainly the case for $r = 1$) and if it satisfies $\Phi_{f+\varepsilon g,h}(y) = \Phi_{f,h}(y) + h\varepsilon g(y) + \mathcal{O}(h^2\varepsilon)$, the function $f_{r+1}(y)$ is

obtained from the relation

$$\Phi_{f^{[r]},h}(y) = \varphi_{f,h}(y) - h^{r+1} f_{r+1}(y) + \mathcal{O}(h^{r+2}). \tag{1.6}$$

Remark 1.1.1 The above construction is similar to that for modified differential equations considered in the theory of backward error analysis. There, one interprets the numerical solution $\Phi_{f,h}(y)$ as the exact solution of a modified differential equation of the form (1.2), i.e.,

$$\varphi_{\widetilde{f},h}(y) = \Phi_{f,h}(y). \tag{1.7}$$

The only difference between (1.3) and (1.7) is that the roles of the integrator Φ and of the exact flow φ are interchanged. Backward error analysis is fundamental for the study of geometric integrators and it is treated in much detail in the monographs of Sanz-Serna & Calvo [SSC94], Hairer, Lubich & Wanner [HLW06], and Leimkuhler & Reich [LR04].

1.1.2 Geometric properties

The importance of backward error analysis in the context of geometric numerical integration lies in the fact that properties of numerical integrators are transfered to corresponding properties of modified equations (see [HLW06, Chap. IX]). Due to the close relationship between backward error analysis and our approach of modifying integrators, it is not a surprise that most results of backward error analysis can be extended to our situation. Let us collect the most important properties of the modified equation (1.2):

- if the numerical integrator $\Phi_{f,h}(y)$ has order p, i.e., the local error satisfies $\Phi_{f,h}(y) - \varphi_{f,h}(y) = \mathcal{O}(h^{p+1})$, then we have $f_j(y) = 0$ for $j = 2, \dots, p$;
- if the integrator $\Phi_{f,h}(y)$ is symmetric, i.e., $\Phi_{f,-h}(y) = \Phi^{-1}_{f,h}(y)$, then the modified differential equation has an expansion in even powers of h, i.e., $f_{2j}(y) = 0$ for all j, and the modifying integrator is symmetric;
- if the basic method $\Phi_{f,h}(y)$ exactly conserves a first integral $I(y)$ of (1.1), then the modified differential equation has $I(y)$ as first integral, and the modifying integrator exactly conserves $I(y)$;
- if the basic method is symplectic for Hamiltonian systems of the form $\dot{y} = J^{-1}\nabla H(y)$, then the modified differential equation is also Hamiltonian, i.e., $\widetilde{f}(y) = J^{-1}\nabla \widetilde{H}(y)$; the modifying integrator is also symplectic;
- if the basic method is a Poisson integrator for Poisson systems of the form $\dot{y} = B(y)\nabla H(y)$, then the modified differential equation is also a Poisson system with the same structure matrix $B(y)$, and the modifying integrator is a Poisson integrator;
- if the basic method is reversible for reversible differential equations, then the modified differential equation and the modifying integrator are reversible;
- if the basic method is volume preserving for divergence-free differential equations, then the modified differential equation is also divergence-free, and the modifying integrator is volume preserving.

Rigorous proofs of these statements are obtained by adapting those of Theorems IX.1.2, IX.2.2, IX.2.3, IX.3.1, IX.3.5, and Corollary IX.5.4 in [HLW06]. One only has to interchange the roles of the numerical and the exact flows.

1.2 Modifying midpoint rule for the rigid body

As an example of a modifying integrator, we introduce a new efficient high-order method for the dynamics of a free rigid body. As basic numerical integrator $y_{n+1} = \Phi_{f,h}(y_n)$, we choose the implicit midpoint rule,

$$y_{n+1} = y_n + hf\Big(\frac{y_n + y_{n+1}}{2}\Big). \tag{1.8}$$

It is a simple symmetric method that exactly preserves quadratic first integrals. For simplicity, we present the modifying implicit midpoint rule of order 6, but the procedure can be extended straight-forwardly to higher orders.

1.2.1 Solving the Euler equations of the rigid body

The Euler equations of motion for the free rigid body are

$$\begin{aligned} \dot{y}_1 &= \alpha\, y_2 y_3, \qquad & \alpha &= I_3^{-1} - I_2^{-1}, \\ \dot{y}_2 &= \beta\, y_3 y_1, \qquad & \beta &= I_1^{-1} - I_3^{-1}, \\ \dot{y}_3 &= \gamma\, y_1 y_2, \qquad & \gamma &= I_2^{-1} - I_1^{-1}, \end{aligned} \tag{1.9}$$

where $y_1(t), y_2(t), y_3(t)$ are the angular momenta of the rigid body, and the constants I_1, I_2, I_3 are the three moments of inertia. This system has two quadratic first integrals (Casimir and Hamiltonian)

$$C(y) = \frac{1}{2}\Big(y_1^2 + y_2^2 + y_3^2\Big) \qquad \text{and} \qquad H(y) = \frac{1}{2}\Big(\frac{y_1^2}{I_1} + \frac{y_2^2}{I_2} + \frac{y_3^2}{I_3}\Big). \tag{1.10}$$

Since the midpoint rule exactly conserves $C(y)$ and $H(y)$, the modified differential equation (1.2) has these two functions as first integrals (see Sect. 1.1.2). Therefore, it is a time transformation of (1.9). Since the method is also symmetric, it is in even powers of h, and the truncated modified equation (order 6) reduces to

$$\dot{y} = f^{[5]}(y) = \big(1 + h^2 s_3(y) + h^4 s_5(y)\big) f(y), \tag{1.11}$$

where $f(y)$ is the right-hand side of (1.9). The scalar functions $s_3(y), s_5(y)$ can be computed using MAPLE and are given by

$$\begin{aligned} s_3(y) &= -\frac{1}{12}\Big(\beta\gamma y_1^2 + \alpha\gamma y_2^2 + \alpha\beta y_3^2\Big), \\ s_5(y) &= \frac{6}{5} s_3^2(y) + \frac{1}{60}\,\alpha\,\beta\,\gamma\,\left(\beta\, {y_1}^2 {y_3}^2 + \gamma\, {y_2}^2 {y_1}^2 + \alpha\, {y_3}^2 {y_2}^2\right). \end{aligned} \tag{1.12}$$

Notice that the scalar functions $s_3(y)$ and $s_5(y)$ are not constant along a particular solution (except in the case of a symmetric body). A modified equation of the same structure has been studied in [Zan05] in the context of backward error analysis.

1.2.2 The full dynamics: the configuration update

To obtain the full dynamics of the free rigid body, one has to solve the augmented differential equation

$$\begin{pmatrix} \dot{y} \\ \dot{Q} \end{pmatrix} = \begin{pmatrix} f(y) \\ QW(y) \end{pmatrix} \qquad \text{with} \qquad W(y) = \begin{pmatrix} 0 & -\frac{y_3}{I_3} & \frac{y_2}{I_2} \\ \frac{y_3}{I_3} & 0 & -\frac{y_1}{I_1} \\ -\frac{y_2}{I_2} & \frac{y_1}{I_1} & 0 \end{pmatrix}, \tag{1.13}$$

where $Q(t)$ is an orthogonal matrix that gives the position of the body in the fixed coordinate system at time t. The modified vector field for the implicit midpoint rule is given by

$$\begin{pmatrix} \dot{y} \\ \dot{Q} \end{pmatrix} = \begin{pmatrix} f^{[5]}(y) \\ QW^{[5]}(y) \end{pmatrix}, \tag{1.14}$$

where $f^{[5]}(y)$ is the vector field of (1.11) and the skew-symmetric matrix $W^{[5]}(y)$ is given by

$$W^{[5]}(y) = W(y^{[5]}). \tag{1.15}$$

Here, $y^{[5]}$ is the vector with components

$$y_j^{[5]} \;=\; y_j\Big(1 + h^2\big(s_3(y) + I_j\, d_3(y)\big) + h^4\big(s_5(y) + I_j\, d_5(y)\big)\Big), \quad j = 1, 2, 3,$$

where $s_3(y)$ and $s_5(y)$ are the functions of (1.12), and (using MAPLE)

$$\begin{aligned} d_3(y) &= \frac{1}{3\Delta}\Big(-C(y) + \delta_0 H(y)\Big), \\ d_5(y) &= \frac{1}{30\Delta}\Big(\delta_1 C(y)^2 + \delta_2 C(y)H(y) + \delta_3 H(y)^2 + y_1^2\big(\delta_4 C(y) + \delta_5 H(y)\big)\Big). \end{aligned}$$

The constants Δ, $\delta_0, \ldots, \delta_5$ only depend on the three moments of inertia I_1, I_2, I_3, and are given by

$$\begin{aligned} &\Delta = I_1 I_2 I_3, && \delta_2 = \frac{1}{\Delta}\big(2I_2^2 + 2I_3^2 - 3I_1^2\big) + \frac{8}{I_1} - \frac{7}{I_2} - \frac{7}{I_3}, \\ &\delta_0 = \frac{1}{2}\big(I_1 + I_2 + I_3\big), && \delta_3 = 3 + 2\frac{I_1 + I_3}{I_2} + 2\frac{I_1 + I_2}{I_3} - 3\frac{I_2 + I_3}{I_1}, \\ &\delta_1 = \frac{1}{\Delta}\big(10I_1 - 6\delta_0\big), && \delta_4 = 5\Big(\frac{1}{I_1} - \frac{1}{I_3}\Big)\Big(\frac{1}{I_2} - \frac{1}{I_1}\Big), \qquad \delta_5 = -\delta_0\delta_4. \end{aligned}$$

Since the vectors y and $y^{[5]}$ are not collinear, the modified equation (1.14) is not a time transformation of the original system (except in the case of a symmetric body).

Applying the implicit midpoint rule to the system (1.14) thus yields a numerical integrator of order 6 for the full dynamics of the free rigid body.

1.2.3 Efficient implementation

Since $C(y)$ and $H(y)$ are two invariants, for the modifying integrator of order 4 (and similarly for higher orders) it is possible to avoid some costly multiplications in (1.12) for the computation of $s_3(y)$ by writing it in the form

$$s_3(y) = c_1 C(y) + c_2 H(y) + c_3\, y_1^2, \tag{1.16}$$

where the constants c_j only depend on I_1, I_2, I_3, and can be calculated once for all. Then, when using a fixed point iteration to compute the internal stage

$$Y = \frac{y_n + y_{n+1}}{2}, \tag{1.17}$$

it is not necessary to evaluate $C(Y)$ and $H(Y)$ with the formulae (1.10). Indeed, one can use the estimates $C(y_n)$ and $H(y_n)$ instead of $C(Y)$ and $H(Y)$,

$$s_3(Y) \approx c_1 C(y_n) + c_2 H(y_n) + c_3\, Y_1^2,$$

where Y_1 is the first component of Y. The method is still symmetric because $C(y_n) = C(y_{n+1})$, and the order remains 4 since $C(Y) = C(y_n) + \mathcal{O}(h^2)$ (and similarly for $H(Y)$).

We now turn our attention to the computation of the configuration update. For an efficient implementation, it is a standard approach to use quaternions to represent orthogonal matrices (see [HLW06] in the context of rigid body integrators implementations). This reduces the midpoint rule

$$Q_{n+1} = Q_n + h\Big(\frac{Q_n + Q_{n+1}}{2}\Big)\, W^{[5]}(Y),$$

where $W^{[5]}$ in given in (1.15) and Y is defined in (1.17), to a simple multiplication of quaternions through the equivalent formulation

$$Q_{n+1} = Q_n\, \Omega.$$

Here, Ω is the orthogonal matrix defined by the Cayley transform

$$\Omega = \Big(I + \frac{h}{2} W^{[5]}(Y)\Big)\Big(I - \frac{h}{2} W^{[5]}(Y)\Big)^{-1}$$

which can be represented by the quaternion $\frac{\omega}{\|\omega\|}$ of norm 1 given by

$$\omega = 1 + \frac{h}{2}\Big(i\frac{Y_1^{[5]}}{I_1} + j\frac{Y_2^{[5]}}{I_2} + k\frac{Y_3^{[5]}}{I_3}\Big).$$

Numerical experiment. We consider the system (1.13) for the free rigid body on the interval $[0, 100]$, and we use $I_1 = 0.9144$, $I_2 = 1.0980$, $I_3 = 1.6600$, and initial values $y(0) = (0.4165, 0.9072, 0.0577)^T$ as in [MZ05]. As numerical integrators we apply the standard implicit midpoint rule and also the modifying versions of orders 4 and 6. The errors as a function of the computational work (number of steps) are drawn as solid lines in Figure 1.1.

We are also curious to see how much work the modifying versions require with respect to the standard application of the midpoint rule. For this, we have carefully implemented the implicit midpoint rule IMR2 and the modifying versions IMR4 and IMR6 of orders 4 and 6 (using quaternions for the rotation matrices). Table 1.1 shows the cpu time (normalized with respect to that of IMR2) of the different implementations, and also the error in the angular momentum for three different choices of the stepsize. Although the numbers should not be overestimated, one clearly sees that IMR4 needs not more than twice and IMR6 not more than 2.5 times the work of IMR2. This is cheaper than what can be expected for either s-stage Runge–Kutta methods of the same order or for composition methods. FORTRAN codes for the modifying implicit midpoint rule introduced in this article can be obtained from the authors on request.

Table 1.1: Normalized computational work and accuracy

	IMR2		IMR4		IMR6	
nstep	work	error	work	error	work	error
100	1.0	$4.0 \cdot 10^{-2}$	1.5	$7.4 \cdot 10^{-4}$	1.8	$2.1 \cdot 10^{-5}$
400	1.0	$2.5 \cdot 10^{-3}$	1.9	$3.0 \cdot 10^{-6}$	2.5	$5.4 \cdot 10^{-9}$
1600	1.0	$1.5 \cdot 10^{-4}$	1.8	$1.2 \cdot 10^{-8}$	2.2	$1.3 \cdot 10^{-12}$

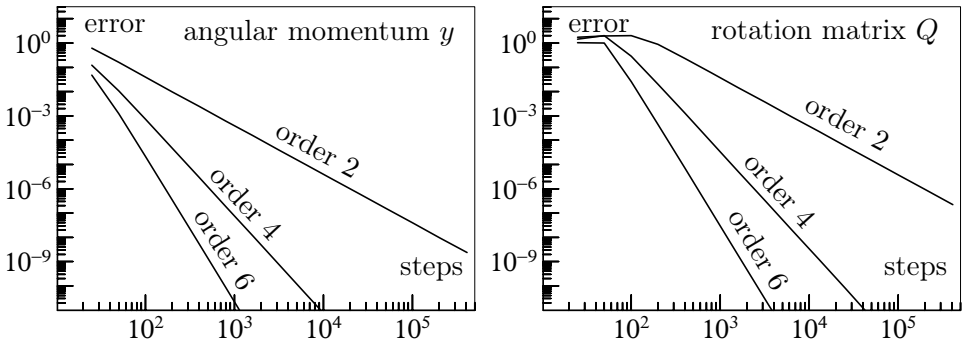


Figure 1.1: Work-precision diagram for the implicit midpoint rule (order 2) and its modifying versions of orders 4 and 6.

1.3 Analysis for B-series methods

The discrete flow of many numerical integrators (including Runge–Kutta methods) can be expanded into a B-series as introduced and studied in [HW74]. We follow the notation of [HLW06, Chap. III], where a more comprehensive presentation of this theory is given.

1.3.1 Substitution law for B-series vector fields

Let $T = \{\bullet, \, \begin{smallmatrix}\bullet\\|\\\bullet\end{smallmatrix}, \, \begin{smallmatrix}\bullet\\|\\\bullet\end{smallmatrix}\!\!\begin{smallmatrix}\diagup\\\diagdown\end{smallmatrix}\!\!\bullet, \ldots\}$ be the set of rooted trees, and let $\emptyset$ be the empty tree. For $\tau_1, \ldots, \tau_m \in T$, we denote by $\tau = [\tau_1, \ldots, \tau_m]$ the tree obtained by grafting the roots of $\tau_1, \ldots, \tau_m$ to a new vertex which becomes the root of τ. The order $|\tau|$ of a tree τ is its number of vertices and its symmetry coefficient is defined recursively by

$$\sigma(\bullet) = 1, \qquad \sigma(\tau) = \sigma(\tau_1) \cdots \sigma(\tau_m) \mu_1! \mu_2! \cdots \, , \tag{1.18}$$

where the integers $\mu_1, \mu_2, \ldots$ count equal trees among τ_1, ..., τ_m. Eventually, elementary differentials $F_f(\tau)$ are given by

$$F_f(\bullet)(y) = f(y), \qquad F_f(\tau)(y) = f^{(m)}(y)\big(F_f(\tau_1)(y), \ldots, F_f(\tau_m)(y)\big). \tag{1.19}$$

For real coefficients $a(\emptyset)$ and $a(\tau), \tau \in T$, a B-series is a series of the form

$$B(f, a) = a(\emptyset)\, Id + \sum_{\tau \in T} \frac{h^{|\tau|}}{\sigma(\tau)} a(\tau) F_f(\tau), \tag{1.20}$$

where Id stands for the identity $Id\,(y) = y$. The Taylor series of the exact solution of (1.1) can be written as a B-series $y(h) = B(f, e)(y_0)$ with coefficients $e(\tau) = \gamma(\tau)^{-1}$, where

$$\gamma(\bullet) = 1, \qquad \gamma(\tau) = |\tau| \, \gamma(\tau_1) \cdots \gamma(\tau_m) \, . \tag{1.21}$$

The flow $y_{n+1} = \Phi_{f,h}(y_n)$ of a Runge–Kutta method is of the form $\Phi_{f,h} = B(f, a)$ with $a(\tau)$ depending only on the coefficients of the method (see [HLW06, Chap. III] for more details).

With the aim of unifying the theory of this article with backward error analysis, we let (1.2) be the modified equation defined by

$$\Phi_{\widetilde{f},h}(y) = \Psi_{f,h}(y) \tag{1.22}$$

where Φ and Ψ are two numerical integrators that can be expressed as B-series $\Phi_{f,h} = B(f, a)$ and $\Psi_{f,h} = B(f, c)$. For $\Psi_{f,h}(y) = \varphi_{f,h}(y)$ we recover formula (1.3), and for $\Phi_{\widetilde{f},h}(y) = \varphi_{\widetilde{f},h}(y)$ we get (1.7).

In terms of B-series, formula (1.22) becomes $B(\widetilde{f},a) = B(f,c)$. When computing recursively some of the coefficient functions of (1.2), one is quickly convinced that they are linear combinations of elementary differentials and that $\widetilde{f}(y) = h^{-1}B(f,b)(y)$ with coefficients $b(\tau)$ that have to be determined (notice that we necessarily have $b(\emptyset) = 0$). This motivates the following theorem, introduced in [CHV05].

Theorem 1.3.1 *For $b(\emptyset) = 0$, the vector field $h^{-1}B(f,b)$ inserted into $B(\cdot,a)$ gives a B-series*

$$B\big(h^{-1}B(f,b),a\big) = B(f,b\star a).$$

We have $(b\star a)(\emptyset) = a(\emptyset)$, some further coefficients are given in Table 1.2, and a general formula for $(b\star a)(\tau)$ is given in (1.27) of Sect. 1.4 below.

Table 1.2: Coefficients of the substitution law for B-series vector fields.

$$
\begin{aligned}
b\star a(\emptyset) &= a(\emptyset),\\
b\star a(\bullet) &= a(\bullet)b(\bullet),\\
b\star a(\tau_2) &= a(\bullet)b(\tau_2)+a(\tau_2)b(\bullet)^2,\\
b\star a(\tau_3) &= a(\bullet)b(\tau_3)+2a(\tau_2)b(\bullet)b(\tau_2)+a(\tau_3)b(\bullet)^3,\\
b\star a(\tau_{21}) &= a(\bullet)b(\tau_{21})+2a(\tau_2)b(\bullet)b(\tau_2)+a(\tau_{21})b(\bullet)^3,\\
b\star a(\tau_4) &= a(\bullet)b(\tau_4)+a(\tau_2)b(\bullet)b(\tau_{21})+a(\tau_2)b(\tau_2)^2+a(\tau_2)b(\bullet)b(\tau_3)\\
&\quad +2a(\tau_3)b(\bullet)^2b(\tau_2)+a(\tau_{21})b(\bullet)^2b(\tau_2)+a(\tau_4)b(\bullet)^4,\\
b\star a(\tau_{31}) &= a(\bullet)b(\tau_{31})+a(\tau_2)b(\bullet)b(\tau_3)+2a(\tau_2)b(\bullet)b(\tau_{21})+a(\tau_3)b(\bullet)^2b(\tau_2)\\
&\quad +2a(\tau_{21})b(\bullet)^2b(\tau_2)+a(\tau_{31})b(\bullet)^4,\\
b\star a(\tau_{22}) &= a(\bullet)b(\tau_{22})+2a(\tau_2)b(\bullet)b(\tau_{21})+a(\tau_2)b(\tau_2)^2+3a(\tau_{21})b(\bullet)^2b(\tau_2)+a(\tau_{22})b(\bullet)^4,\\
b\star a(\tau_{32}) &= a(\bullet)b(\tau_{32})+3a(\tau_2)b(\bullet)b(\tau_3)+3a(\tau_3)b(\bullet)^2b(\tau_2)+a(\tau_{32})b(\bullet)^4.
\end{aligned}
$$

We postpone the proof of this theorem to Sect. 1.4, and briefly discuss some of the most important properties and applications. Further properties may be found in [CHV05].

The question of finding the modified equation defined by (1.22), i.e., of finding the coefficients $b(\tau)$ for given $a(\tau)$ and $c(\tau)$ in the relation

$$B\big(h^{-1}B(f,b),a\big) = B(f,c),$$

results in solving for $b(\tau)$ the algebraic system

$$(b\star a)(\tau) = c(\tau) \qquad \text{for} \quad \tau\in T. \tag{1.23}$$

We notice that

$$(b\star a)(\tau) = a(\bullet)b(\tau)+\cdots+a(\tau)b(\bullet)^{|\tau|},$$

where the three dots involve only trees of order strictly less than $|\tau|$. Consequently, for consistent integrators $\Phi_{f,h} = B(f,a)$ and $\Psi_{f,h} = B(f,c)$, for which $a(\emptyset) = a(\bullet) = 1$ and $c(\emptyset) = c(\bullet) = 1$, the coefficients $b(\tau)$ can be computed recursively from (1.23). In this way, the computation of the vector fields $f_j(y)$ in the modified differential equation (1.2) is reduced to that of real coefficients.

Modifying integrators. In this case $\Psi_{f,h}$ is the exact h-flow of (1.1) which is a B-series with coefficients $e(\tau) = \gamma(\tau)^{-1}$. Consequently, the coefficients $b(\tau)$ of the modified differential equation for $\Phi_{f,h} = B(f,a)$ are obtained from

$$(b \star a)(\tau) = e(\tau) \qquad \text{for} \quad \tau \in T. \tag{1.24}$$

Backward error analysis. The modified differential equation of a method $\Psi_{f,h} = B(f,c)$ is obtained by putting $\Phi_{f,h}$ equal to the exact flow. Its coefficients $b(\tau)$ are therefore obtained from

$$(b \star e)(\tau) = c(\tau) \qquad \text{for} \quad \tau \in T. \tag{1.25}$$

Remark 1.3.2 The B-series $h^{-1}B(f,b)$ of mappings $b : T \cup \{\emptyset\} \to \mathbb{R}$ with $b(\emptyset) = 0$ represent vector fields. The product $b \star a$ defines a group structure on the set $\{c : T \cup \{\emptyset\} \to \mathbb{R}\,;\, c(\emptyset) = 0, c(\bullet) = 1\}$ which represents such vector fields. Its unit element is given by $c(\bullet) = 1$ and $c(\tau) = 0$ for $|\tau| > 1$, and it corresponds to the original vector field $f(y)$.

We mention that the presented theory can be extended straightforwardly to partitioned integration methods (P-series). This is particularly important for the consideration of symplectic integrators.

1.3.2 Modifying implicit midpoint rule

As an example, consider the implicit midpoint rule (1.8) which admits a B-series expansion $B(f,a)$ with $a(\tau) = \big(\frac{1}{2}\big)^{|\tau|-1}$. Determining the functions $f_3(y)$ and $f_5(y)$ in the modified differential equation (1.2) amounts to computing (up to order 5) the coefficients $b(\tau)$ of the B-series $B(f,b)$ from the relation (1.24). The formulae of Table 1.2 yield

$$b(\bullet) = 1, \qquad b([\bullet]) = 0, \qquad b([\bullet,\bullet]) = \frac{1}{12}, \qquad b([[\bullet]]) = -\frac{1}{12}.$$

The coefficients for trees of order 4 vanish due to the symmetry of the method, and those for order 5 can be calculated from (1.27). We thus arrive at the following modified vector field:

$$\begin{aligned} f_h^{[5]} \;&=\; f \;+\; \frac{h^2}{12}\Big(-f'f'f + \frac{1}{2} f''(f,f)\Big) \\ &+\; \frac{h^4}{120}\Big(f'f'f'f'f - f''(f,f'f'f) + \frac{1}{2} f''(f'f,f'f)\Big) \\ &+\; \frac{h^4}{240}\Big(-\frac{1}{2} f'f'f''(f,f) + f'f''(f,f'f) + \frac{1}{2} f''(f,f''(f,f))\Big) \\ &+\; \frac{h^4}{240}\Big(-\frac{1}{2} f^{(3)}(f,f,f'f) - \frac{1}{2} f'f^{(3)}(f,f,f) + \frac{1}{8} f^{(4)}(f,f,f,f)\Big) \end{aligned} \tag{1.26}$$

This formula reduces to (1.11) for the Euler equations and to (1.14) for the full dynamics of the rigid body.

1.3.3 Elementary differential Runge–Kutta methods

The idea of modifying integrators applied to Runge–Kutta methods provides an easy way to construct high-order methods for the numerical solution of (1.1). Methods obtained in this manner are a particular case of the so-called *elementary differential Runge-Kutta methods* (EDRK), introduced by Murua [Mur94].

Consider a s-stage Runge–Kutta method $y_{n+1} = \Phi_{f,h}(y_n)$ of order p. It admits a B-series expansion $\Phi_{f,h} = B(f,a)$. Applying this method to the modified vector field $f^{[r]}(y)$,

truncated at some order r greater than p, leads to a r-derivative EDRK method of order (at least) r given by

$$\begin{aligned} Y_i &= y_n + h\sum_{j=1}^{s} a_{ij} f^{[r]}(Y_j), \quad i = 1, \dots, s, \\ y_{n+1} &= y_n + h\sum_{j=1}^{s} b_j f^{[r]}(Y_j). \end{aligned}$$

By Theorem 1.3.1 the modified vector field is a B-series

$$\begin{aligned} f^{[r]}(y) &= f(y) + h^p f_{p+1}(y) + \cdots + h^{r-1} f_r(y), \\ f_j(y) &= \sum_{|\tau|=j} \frac{b(\tau)}{\sigma(\tau)} F_f(\tau)(y), \quad j = 1, \dots, r. \end{aligned}$$

Its coefficients $b(\tau)$ are obtained from the relation (1.24).

Example 1.3.3 Consider the s-stage Runge–Kutta method of order $p = 2s$ (this is the Gauss method). Since it is symplectic and symmetric, the modified vector field $f^{[r]}(y)$ is Hamiltonian for all Hamitonian systems $\dot{y} = J^{-1}\nabla H(y)$, and we have $f_{2j}(y) = 0$ for all j (see Sect. 1.1.2). Then, if we take an odd integer r, we obtain an implicit symplectic and symmetric r-derivative EDRK method of order (at least) $r + 1$. The special case $s = 1$ yields the symplectic generating function methods based on the implicit midpoint rule [FWQW89].

For instance, for $s = 2$ and $r = 5$, we obtain a 5-derivative EDRK method of order 6, and coefficients $b(\tau)$ for trees of order $|\tau| = 5$ are given by

$$b(\tau_1) = \frac{1}{180}, \quad b(\tau_2) = \frac{1}{360}, \quad b(\tau_3) = \frac{1}{720},$$

together with the algebraic conditions on the coefficients $b(\tau)$ for $f^{[r]}(y)$ to be a Hamiltonian vector field (see [HLW06, Sect. IX.9.2]).

Remark 1.3.4 It would be interesting to know whether there exist symplectic (and symmetric) EDRK methods that are not modifying classical Runge–Kutta methods and have an order higher than $\max(2s, r + 1)$.

1.4 An explicit formula for the substitution law

In this section, we give a computation formula for the substitution law of B-series introduced in Sect. 1.3.1. We begin with some definitions.

1.4.1 Partitions and skeletons

A *partition* p^τ of a tree τ is obtained by cutting some of its edges [CHV05]. The resulting list of trees is denoted $P(p^\tau)$. Eventually, the set of all partitions p^τ of τ is denoted $\mathcal{P}^\tau$. Now, given a partition p^τ, the corresponding *skeleton* $\chi(p^\tau)$, as introduced in [CL98], is the tree obtained by contracting each tree of $P(p^\tau)$ to a single vertex $\bullet$ and by re-establishing the cut edges (see Table 1.3). We observe that a tree $\tau \in T$ has exactly $2^{|\tau|-1}$ partitions $p^\tau \in \mathcal{P}^\tau$, and that different partitions may lead to the same list $P(p^\tau)$.

Table 1.3: The 8 partitions of a tree of order 4 with associated functions

p^τ								
$\chi(p^\tau)$								
$P(p^\tau)$								

1.4.2 The substitution law formula

We are now in position to state the main result of this section. The coefficients $(b \star a)(\tau)$ of the substitution law can be expressed in terms of the coefficients $a(\theta)$ and $b(\theta)$ with $|\theta| \leq |\tau|$ in the following polynomial expression:

$$(b \star a)(\tau) = \sum_{p^\tau \in \mathcal{P}(\tau)} a\big(\chi(p^\tau)\big) \prod_{\delta \in P(p^\tau)} b(\delta) \tag{1.27}$$

for $\tau \in T$. For the example of Table 1.3, this formula yields

$$\begin{aligned} (b \star a)(\,\raisebox{-0.3ex}{}\,) &= a(\bullet)b(\,\raisebox{-0.3ex}{}\,) + a(\,\raisebox{-0.3ex}{  }\,)b(\bullet)b(\,\raisebox{-0.3ex}{  }\,) + 2a(\,\raisebox{-0.3ex}{}\,)b(\bullet)b(\,\raisebox{-0.3ex}{  }\,) \\ &+ a(\,\raisebox{-0.3ex}{  }\,)b(\bullet)^2b(\,\raisebox{-0.3ex}{}\,) + 2a(\,\raisebox{-0.3ex}{}\,)b(\bullet)^2b(\,\raisebox{-0.3ex}{}\,) + a(\,\raisebox{-0.3ex}{}\,)b(\bullet)^4. \end{aligned}$$

1.4.3 Proof of the substitution law formula

Multiplying (1.28) with $a(\theta)$ and summing up yields $B(g, a) = B(f, b \star a)$. Formula (1.27) is thus obtained by multiplying (1.29) with $a(\theta)$ and summing up. It therefore remains to prove the following lemma.

Lemma 1.4.1 *Let $g(y) = h^{-1}B(f, b)(y)$ be a (h-dependent) vector field defined by a B-series with $b(\emptyset) = 0$. Then, for $\theta \in T$, we have*

$$\frac{h^{|\theta|}}{\sigma(\theta)} F_g(\theta) = B(f, b_\theta), \tag{1.28}$$

where the coefficients $b_\theta(\tau)$ are given by $b_\theta(\emptyset) = 0$, and for $\tau \in T$,

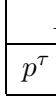
$$b_\theta(\tau) = \sum_{p^\tau \in \mathcal{P}^\tau,\ \chi(p^\tau) = \theta} \ \prod_{\delta \in P(p^\tau)} b(\delta). \tag{1.29}$$

Before proving this lemma, we need the following Lemma 1.4.2 which requires a few more definitions illustrated in Table 1.4. Given a partition p^τ of a tree τ, the tree of $P(p^\tau)$ which contains the root of τ is denoted $r(p^\tau)$. For brevity of formulae, we further use $P^*(p^\tau) = P(p^\tau) \backslash \{r(p^\tau)\}$. A partition p^τ is said to be *admissible* if the path from the root to any vertex has at most one cut. The set of admissible partitions of τ is denoted $\mathcal{AP}^\tau$.

Lemma 1.4.2 *Let $g(y)$ be defined by $g(y) = h^{-1}B(f, b)(y)$ with $b(\emptyset) = 0$. Then, for $\delta = [\delta_1, \ldots, \delta_m] \in T$, we have*

$$\frac{h^{|\delta|}}{\sigma(\delta)}\ g^{(m)}(y)\ \Big(F_f(\delta_1)(y), \ldots, F_f(\delta_m)(y)\Big) = B(f, d_\delta b)(y), \tag{1.30}$$

Table 1.4: The 8 partitions of a tree of order 4 with other associated functions

p^τ								
$r(p^\tau)$								
$P^*(p^\tau)$	$\emptyset$	$\{$  $\}$	$\{\bullet\}$	$\{\bullet\}$	$\{\bullet\bullet\}$	$\{\bullet$$\}$	$\{\bullet$$\}$	$\{\bullet\bullet\bullet\}$
$p^\tau \in \mathcal{AP}^\tau$?	yes	yes	yes	yes	yes	no	no	no

where $d_\delta b(\tau)$ is defined by $d_\delta b(\emptyset) = 0$, and for $\tau \in T$,

$$d_\delta b(\tau) = \sum_{p^\tau \in \mathcal{AP}^\tau,\, P^*(p^\tau)=\{\delta_1,\ldots,\delta_m\}} b\big(r(p^\tau)\big). \tag{1.31}$$

Proof. The proof follows closely that of Lemma IX.9.1 in [HLW06] and it is thus omitted. Notice that admissible partitions correspond to ordered subtrees in [HLW06]. □

Proof of Lemma 1.4.1. We proceed by induction on $|\theta|$. From $F_g(\bullet) = g = h^{-1}B(f,b)$ we have $b_\bullet(\tau) = b(\tau)$ for all $\tau \in T$. Consider now a tree $\theta = [\theta_1, \ldots, \theta_m]$ with $|\theta| \geq 2$, and assume (1.28) and (1.29) are satisfied for trees of order strictly less than $|\theta|$. By definition of $F_g(\theta)$ and multi-linearity of $g^{(m)}(y)(\cdot, \ldots, \cdot)$, we have

$$\begin{aligned}\frac{h^{|\theta|}}{\sigma(\theta)} F_g(\theta)(y) &= \frac{\sigma(\theta_1)\cdots\sigma(\theta_m)}{\sigma(\theta)} \sum_{\tau_1,\ldots,\tau_m \in T} \frac{1}{\sigma(\tau_1)\cdots\sigma(\tau_m)} \left(\prod_{i=1}^m b_{\theta_i}(\tau_i)\right) \\ &\quad \cdot h^{|\tau|} g^{(m)}(y)(F_f(\tau_1)(y), \ldots, F_f(\tau_m)(y))\end{aligned}$$

with $\tau = [\tau_1, \ldots, \tau_m]$. Formula (1.30) of Lemma 1.4.2 then gives, for $v \in T$:

$$b_\theta(v) = \frac{\sigma(\theta_1)\cdots\sigma(\theta_m)}{\sigma(\theta)} \sum_{\tau_1,\ldots,\tau_m \in T} \frac{\sigma(\tau)}{\sigma(\tau_1)\cdots\sigma(\tau_m)} \left(\prod_{i=1}^m b_{\theta_i}(\tau_i)\right) d_\tau b(v).$$

Now, taking into account the fact that permutations among $\tau_1, \ldots, \tau_m$ do not change the tree $\tau = [\tau_1, \ldots, \tau_m]$ (and similarly for θ), it follows that

$$b_\theta(v) = \sum_{\tau=[\tau_1,\ldots,\tau_m]\in T} \sum_{\substack{\theta_1,\,\ldots,\,\theta_m \in T,\\ [\theta_1,\ldots,\theta_m]=\theta}} \left(\prod_{i=1}^m b_{\theta_i}(\tau_i)\right) d_\tau b(v)$$

and Formula (1.31) allows one to write

$$b_\theta(v) = \sum_{[\tau_1,\ldots,\tau_m]\in T} \sum_{\substack{p^v \in \mathcal{AP}^v,\\ P^*(p^v)=\{\tau_1,\,\ldots,\,\tau_m\}}} \sum_{\substack{\theta_1,\,\ldots,\,\theta_m \in T,\\ [\theta_1,\,\ldots,\,\theta_m]=\theta}} b\big(r(p^v)\big) \prod_{i=1}^m b_{\theta_i}(\tau_i).$$

Using the induction hypothesis we eventually obtain

$$\begin{aligned} b_\theta(v) &= \sum_{\substack{[\tau_1,\ldots,\tau_m]\in T,\\ p^v\in\mathcal{AP}^v,\\ P^*(p^v)=\{\tau_1,\,\ldots,\,\tau_m\}}} \;\sum_{\substack{p^{\tau_1}\in\mathcal{P}^{\tau_1},\;\ldots,\;p^{\tau_m}\in\mathcal{P}^{\tau_m},\\ [\chi(p_1^\tau),\,\ldots,\,\chi(p_m^\tau)]=\theta}} b\big(r(p^v)\big)\prod_{\delta\in\bigcup_{i=1}^m P(p^{\tau_i})} b(\delta) \\ &= \sum_{p^v\in\mathcal{P}^v,\,\chi(p^v)=\theta}\;\prod_{\delta\in P(p^v)} b(\delta), \end{aligned}$$

which proves the statement of Lemma 1.4.1. □

Chapter 2

An algebraic counterpart of modified fields

The concept of B-series was introduced in [HW74], following the pioneering work of Butcher [But69, But72], for studying order conditions for Runge-Kutta methods, and is now exposed in various textbooks and articles, though possibly with different normalizations. It turned out to be a powerful tool for the study of numerical integrators in geometric integration [CMSS94, Hai94, Hai99, HLW06].

A surprising connection of the B-series theory with renormalization in quantum field theory was established by Brouder [Bro00, Bro04] (see also Arne Dür [Dür86]). It turns out that the composition law (2.3) of B-series is precisely the one underlying the coproduct in the graded Hopf algebra of (rooted) trees, constructed by Connes & Moscovici [CM98] in the context of non-commutative geometry and by Connes & Kreimer [CK98, CK00] where it is used to describe renormalization in quantum field theory. This Hopf algebra is generalized in [MKW08] to a Hopf algebraic structure of unordered rooted trees in the context of Lie group integrators. It is also studied in [Mur06] in the context of splitting methods.

The goal of this chapter is to explain the fundamental role in numerical analysis of two composition laws on B-series, the Butcher composition [HW74] and the substitution law $\star$ (Theorem 1.3.1) introduced in [CHV05, CHV07b] (see Chapter 1), emphasize their common algebraic structure, and expose their relationships. The motivation for putting forward these results is the the recent article of Calaque, Ebrahimi-Fard & Manchon [CEFM08], where it is shown that the substitution law on B-series can be turned into a new coproduct, which allows to build a new Hopf tree algebra, which is also graded, commutative and non-cocommutative.

The ideas sustaining the substitution law $\star$ on B-series rely on the theory of modified differential equations. *Modified* equations have proved to be of great importance for the study of integration methods. Consider for instance the well-known Euler-McLaurin formula

$$\sum_{j=p}^{q-1} \frac{f(j)+f(j+1)}{2} = \int_p^q f(t)dt + \sum_{j\geq 1} \frac{B_{2j}}{(2j)!}\big(f^{(2j-1)}(q) - f^{(2j-1)}(p)\big)$$

where $f(t)$ is a sufficiently differentiable function on $\mathbb{R}$ and the B_j's are the Bernoulli numbers. Here, the series should be considered as formal because it does not converge in general. It can be interpreted by saying that the trapezoidal rule for approximating the

integral of $f(t)$ formally yields the exact integral of a modified function $\widetilde{f}(t)$:

$$\frac{f(j)+f(j+1)}{2} = \int_j^{j+1} \widetilde{f}(t)dt, \qquad \widetilde{f}(t) = f(t) + \sum_{j\geq 1} \frac{B_{2j}}{(2j)!} f^{(2j)}(t),$$

as explained in [HLW06, Example IX.7.1] More generally, the idea of *backward error analysis* for ordinary differential equations (not necessarily Hamiltonian) of the form

$$\left\{ \begin{array}{rcl} \dot{y} & = & f(y) \\ y(0) & = & y_0 \end{array} \right. , \tag{2.1}$$

comes to computing the first coefficients of a *modified* equation. For instance, considering again the trapezoidal rule with stepsize h,

$$\begin{aligned} y_1 &= y_0 + \frac{h}{2}\Big(f(y_0)+f(y_1)\Big) \\ &= y_0 + hf(y_0) + \frac{h^2}{2} f'(y_0)f(y_0) \\ &\quad + \frac{h^3}{4}\Big(f'(y_0)f'(y_0)f(y_0) + f''(y_0)(f(y_0),f(y_0))\Big) + \dots \end{aligned} \tag{2.2}$$

we search for a h-dependent modified field $\widetilde{f}(y)$ such that the trapezoidal rule formally yields the exact solution of a modified differential equation,

$$\dot{\widetilde{y}} = \widetilde{f}(\widetilde{y}) = f(\widetilde{y}) + \frac{h^2}{12} f'(\widetilde{y})f'(\widetilde{y})f(\widetilde{y}) + \frac{h^2}{12} f''(\widetilde{y})(f(\widetilde{y}),f(\widetilde{y})) + \dots,$$

i.e. $y_n = \widetilde{y}(nh)$ for $n = 0, 1, 2, \dots$

In [CHV07b], we introduce the idea of *modifying (or preprocessed) vector integrators*. For instance, applying the trapezoidal rule to a suitable modified differential equation

$$\dot{y} = f(y) - \frac{h^2}{12} f'(y)f'(y)f(y) - \frac{h^2}{12} f''(y)(f(y),f(y)) + \dots,$$

formally yields the exact solution of (2.1). Truncating after the second-order terms then provides a fourth-order approximation to the solution of (2.1).

Though the computations can theoretically be carried on up to any order, getting further terms soon becomes very tedious. A general procedure -which amounts to solving the Hamilton-Jacobi equation in the case where (2.1) is Hamiltonian [Fen86, FWQW89, CS90] - appears to be of great help. Getting the modified equation is not important per se: one is usually only interested in exhibiting some of its structural properties (such as symmetry, existence of a Hamiltonian, ...), a task usually accomplished without the knowledge of the coefficients themselves. However, recurrence formulas have been given by Hairer [Hai94] and by Calvo, Murua and Sanz-Serna [CMSS94] and allow to give alternative algebraic proofs of some known results. The approach followed in [Hai99] is based on a formula for the Lie-derivatives of a B-series and leads to recursive formulas.

2.1 Two composition laws on B-series

2.1.1 The Butcher group

The term 'Butcher group' was introduced by Hairer & Wanner [HW74], and relies on the following fundamental result on the composition of B-series. A direct proof can be

found in [HLW06, Theorem III.1.10]. From a numerical analysis point of view, the most fundamental realization of this group is the composition of Runge-Kutta methods [But72], which can be expended into B-series (or P-series for partitioned Runge-Kutta methods).

Theorem 2.1.1 *Let $a, b : \mathcal{T} \cup \{\emptyset\} \to \mathbb{R}$ be two mappings, with $a(\emptyset) = 1$. Then the B-series $B(a, y)$ inserted into $B(b, \cdot)$ is still a B-series*

$$B(b, B(a, y)) = B(a \cdot b, y),$$

and $a \cdot b : \mathcal{T} \cup \{\emptyset\} \to \mathbb{R}$ is defined by

$$a \cdot b(\emptyset) = b(\emptyset), \qquad a \cdot b(\tau) = b(\emptyset)a(\tau) + \sum_{p^\tau \in \mathcal{AP}(\tau)} b(r(p^\tau)) \prod_{\delta \in P^*(p^\tau)} a(\delta). \tag{2.3}$$

Here, we use the notations on partitions defined is Sect. 1.4.1. In particular, $\mathcal{AP}$ denotes the set of *admissible* partitions.

The set of mappings $a : \mathcal{T} \cup \{\emptyset\} \to \mathbb{R}$, satisfying $a(\emptyset) = 1$, is a group for the composition law (2.3), called the Butcher group [But72]. Its unit element is $\delta_\emptyset$, defined as

$$\delta_\emptyset(\emptyset) = 1, \quad \delta_\emptyset(\tau) = 0 \text{ for } \tau \in \mathcal{T},$$

and the inverse of a, as shown in [CK98], is given by

$$a^{\cdot -1}(\emptyset) = 1, \qquad a^{\cdot -1}(\tau) = \sum_{p^\tau \in \mathcal{P}(\tau)} (-1)^{\#(p^\tau)} \prod_{\delta \in P(p^\tau)} a(\delta). \tag{2.4}$$

We shall see further in Remark 2.4.2 that the general formula (1.27) for the substitution law $\star$ allows to retrieve this formula (2.4) for the inverse in the Butcher group.

$$\begin{aligned}
a \cdot b(\bullet) &= b(\emptyset)a(\bullet) + b(\bullet), \\
a \cdot b(\begin{smallmatrix}\bullet\\|\\\bullet\end{smallmatrix}) &= b(\emptyset)a(\begin{smallmatrix}\bullet\\|\\\bullet\end{smallmatrix}) + b(\bullet)a(\bullet) + b(\begin{smallmatrix}\bullet\\|\\\bullet\end{smallmatrix}), \\
a \cdot b(\begin{smallmatrix}&\bullet&\\\diagup&&\diagdown\\\bullet&&\bullet\end{smallmatrix}) &= b(\emptyset)a(\begin{smallmatrix}&\bullet&\\\diagup&&\diagdown\\\bullet&&\bullet\end{smallmatrix}) + b(\bullet)a(\bullet)^2 + 2b(\begin{smallmatrix}\bullet\\|\\\bullet\end{smallmatrix})a(\bullet) + b(\begin{smallmatrix}&\bullet&\\\diagup&&\diagdown\\\bullet&&\bullet\end{smallmatrix}), \\
a \cdot b(\begin{smallmatrix}\bullet\\|\\\bullet\\|\\\bullet\end{smallmatrix}) &= b(\emptyset)a(\begin{smallmatrix}\bullet\\|\\\bullet\\|\\\bullet\end{smallmatrix}) + b(\bullet)a(\begin{smallmatrix}\bullet\\|\\\bullet\end{smallmatrix}) + b(\begin{smallmatrix}\bullet\\|\\\bullet\end{smallmatrix})a(\bullet) + b(\begin{smallmatrix}\bullet\\|\\\bullet\\|\\\bullet\end{smallmatrix}), \\
a^{\cdot -1}(\bullet) &= -a(\bullet), \\
a^{\cdot -1}(\begin{smallmatrix}\bullet\\|\\\bullet\end{smallmatrix}) &= -a(\begin{smallmatrix}\bullet\\|\\\bullet\end{smallmatrix}) + a(\bullet)^2, \\
a^{\cdot -1}(\begin{smallmatrix}&\bullet&\\\diagup&&\diagdown\\\bullet&&\bullet\end{smallmatrix}) &= -a(\begin{smallmatrix}&\bullet&\\\diagup&&\diagdown\\\bullet&&\bullet\end{smallmatrix}) + 2a(\bullet)a(\begin{smallmatrix}\bullet\\|\\\bullet\end{smallmatrix}) - a(\bullet)^3, \\
a^{\cdot -1}(\begin{smallmatrix}\bullet\\|\\\bullet\\|\\\bullet\end{smallmatrix}) &= -a(\begin{smallmatrix}\bullet\\|\\\bullet\\|\\\bullet\end{smallmatrix}) + 2a(\bullet)a(\begin{smallmatrix}\bullet\\|\\\bullet\end{smallmatrix}) - a(\bullet)^3.
\end{aligned}$$

Table 2.1: Composition law (2.3) on B-series and inverse (2.4) in the Butcher group for trees of order ≤ 3.

2.1.2 Substitution law

In the previous chapter, we introduced a new composition law on B-series, see [CHV05, CHV07b], denoted $\star$ and called *law of substitution*, obtained as the result of the substitution

of a vector field $g(y) = h^{-1}B(f,b)(y)$ with $b(\emptyset) = 0$ into another B-series $B(g,a)(y)$. Formally, this writes

$$\begin{aligned} B(h^{-1}B(f,b),a)(y) &= a(\emptyset)y + a(\bullet)B(f,b)(y) + a(\bullet\!\!\nearrow)\partial_y\Big(B(f,b)(y)\Big)B(f,b)(y) \\ &\quad + \frac{a(\vee)}{2}\partial_y^2\Big(B(f,b)(y)\Big)\Big(B(f,b)(y),B(f,b)(y)\Big) + \dots \\ &:= B(f,b\star a)(y). \end{aligned}$$

and an explicit formula (1.27) for $b\star a(\tau)$ is given in Theorem 1.3.1. We have $(b\star a)(\emptyset) = a(\emptyset)$, and a general formula for $(b\star a)(\tau)$ is given by

$$(b\star a)(\tau) = \sum_{p^\tau\in\mathcal{P}(\tau)} a\big(\chi(p^\tau)\big)\prod_{\delta\in P(p^\tau)} b(\delta)$$

for $\tau\in T$.

2.2 The Hopf tree algebra of Connes & Kreimer

We describe the Hopf algebra H of trees [CM98, CK98, CK00], and expain its close connection with the Butcher group.

Let $(H,+,\cdot,\mu,\eta)$ denote the commutative $\mathbb{R}$-algebra of polynoms on trees $\mathcal{T}\cup\{\emptyset\}$. Then, H is the set of all linear combinations of forests of rooted trees. For instance it contains $2\bullet\!\!\nearrow - 5\emptyset$, $\vee\bullet^4 - \bullet\!\!\nearrow$, and we have

$$\mu\big((\bullet + 2\vee)\otimes(\bullet\!\!\nearrow + 5\,\bullet\!\!\nearrow\!\!\!\bullet)\big) = \bullet\bullet\!\!\nearrow + 5\bullet\!\!\nearrow\!\!\!\bullet + 2\bullet\!\!\nearrow\vee + 10\vee\!\!\nearrow\!\!\!\bullet.$$

The unit for multiplication is simply the monomial of the empty forest $\emptyset$:

$$\mu(\tau\otimes\emptyset) = \mu(\emptyset\otimes\tau) = \tau,$$

and we have $\vee^2\emptyset = \vee^2$, $\bullet\!\!\nearrow\emptyset = \bullet\!\!\nearrow$, etc.

2.2.1 The coproduct and antipode

To make H a coalgebra, we need a coproduct $\Delta : H\to H\otimes H$ which has to be coassociative,

$$(id\otimes\Delta)\circ\Delta = (\Delta\otimes id)\circ\Delta.$$

and if it is compatible with the algebra laws, we obtain a bialgebra. Finally, to obtain a Hopf algebra, we need an antipode $S : H\to H$, which is a map satisfying

$$\mu\circ(S\otimes Id)\circ\Delta = \mu\circ(Id\otimes S)\circ\Delta = \delta_\emptyset$$

We shall see below that the antipode allows to compute the inverse in the Butcher group, and also for the substitution law.

For a tree $\tau = [\tau_1,\dots,\tau_m]\in T$, we put $B^-(\tau) = \tau_1\cdots\tau_m$, $B^+(\tau_1\cdots\tau_m) = [\tau_1,\dots,\tau_m]$. For instance, $B^-(\vee) = \bullet\bullet\!\!\nearrow$ and $B^+(\bullet\bullet\!\!\nearrow) = \vee$. The coproduct $\Delta_{CK} : H\to H\otimes H$ is now defined recursively as

$$\Delta_{CK}(\tau) = \tau\otimes\emptyset + (Id_H\otimes B^+)\circ\Delta_{CK}\circ B^-(\tau). \tag{2.5}$$

with $\Delta_{CK}(\emptyset) = \emptyset\otimes\emptyset$, and $\Delta_{CK}(\prod_j\tau_j) = \prod_j\Delta_{CK}(\tau_j)$.

It can be shown by induction the equivalent formula, very similar to (2.3),

$$\Delta_{CK}(\tau) = \tau \otimes \emptyset + \sum_{p^\tau \in \mathcal{AP}(\tau)} \Big(\prod_{\delta \in P^*(p^\tau)} \delta \Big) \otimes r(p^\tau).$$

For the first trees, this yields (compare with formulas in Table 2.1),

$$\begin{array}{rcl} \Delta_{CK}(\emptyset) & = & \emptyset \otimes \emptyset \\ \Delta_{CK}(\bullet) & = & \bullet \otimes \emptyset + \emptyset \otimes \bullet \\ \Delta_{CK}(\,\bullet\!\!-\!\!\bullet\,) & = & \,\bullet\!\!-\!\!\bullet\, \otimes \emptyset + \bullet \otimes \bullet + \emptyset \otimes \,\bullet\!\!-\!\!\bullet\, \\ \Delta_{CK}(\,\bullet\!\!<\!\!\bullet\,) & = & \,\bullet\!\!<\!\!\bullet\, \otimes \emptyset + \bullet\,\bullet \otimes \bullet + 2\bullet \otimes \,\bullet\!\!-\!\!\bullet\, + \emptyset \otimes \,\bullet\!\!<\!\!\bullet\, \\ \Delta_{CK}(\,\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet\,) & = & \,\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet\, \otimes \emptyset + \,\bullet\!\!-\!\!\bullet\, \otimes \bullet + \bullet \otimes \,\bullet\!\!-\!\!\bullet\, + \emptyset \otimes \,\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet\,. \end{array} \tag{2.6}$$

2.2.2 Hopf algebra convolution and the Butcher group

A map $a : T \to \mathbb{R}$ can by extended as an algebra map on H by linearity and using $a(\emptyset) = 1$ and $a(\tau\tau') = a(\tau)a(\tau')$. The convolution product $b \cdot a$ of two mappings $b, a : H \to \mathbb{R}$ (with $a(\emptyset) = b(\emptyset) = 1$) is then defined as:

$$(b \cdot a) = \mu_{\mathbb{R}} \circ (b \otimes a) \circ \Delta_{CK} \tag{2.7}$$

Of course $a \cdot b \neq b \cdot a$ and the Hopf algebra in not cocommutative. The convolution product $a \cdot b$ makes the set of algebra mappings from H to $\mathbb{R}$ a group: the Butcher group.

Then, it is standard that the inverse for the convolution product of a mapping $a : H \to \mathbb{R}$ is given by the antipode:

$$a^{\cdot -1}(\tau) = a \circ S_{CK}(\tau).$$

In fact, this is true for all graded and connected Hopf algebra: the set of characters (i.e. unitary algebra mappings) of the Hopf algebra is a group for the convolution product.

For the first trees, this yields (compare with formulas in Table 2.1),

$$\begin{array}{rcl} S_{CK}(\emptyset) & = & \emptyset, \\ S_{CK}(\bullet) & = & -\bullet, \\ S_{CK}(\,\bullet\!\!-\!\!\bullet\,) & = & -\,\bullet\!\!-\!\!\bullet\, + \bullet^2, \\ S_{CK}(\,\bullet\!\!<\!\!\bullet\,) & = & -\,\bullet\!\!<\!\!\bullet\, + 2\bullet\,\,\bullet\!\!-\!\!\bullet\, - \bullet^3, \\ S_{CK}(\,\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet\,) & = & -\,\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet\, + 2\bullet\,\,\bullet\!\!-\!\!\bullet\, - \bullet^3. \end{array}$$

Explicit formula for the antipode S_{CK} is discussed in Remark 2.4.2.

2.3 A Hopf trees algebra based on the substitution law

In the recent article [CEFM08] a Hopf tree algebra has been constructed, with a new coproduct that is closely related to the substitution law (1.27). We consider the coproduct[1]

$$\Delta_{CEM}(\tau) = \sum_{p^\tau \in \mathcal{P}(\tau)} \Big(\prod_{\delta \in P(p^\tau)} \delta \Big) \otimes \chi(p^\tau)$$

[1]This formula is slightly different from the one given in [CEFM08], see Remark 2.3.1.

$$
\begin{aligned}
(b\star a)(\bullet) &= a(\bullet)b(\bullet)\\
(b\star a)(\bullet\!-\!\bullet) &= a(\bullet)b(\bullet\!-\!\bullet)+a(\bullet\!-\!\bullet)b(\bullet)^2\\
(b\star a)(\bullet\!-\!\bullet\!-\!\bullet) &= a(\bullet)b(\bullet\!-\!\bullet\!-\!\bullet)+2a(\bullet\!-\!\bullet)b(\bullet)b(\bullet\!-\!\bullet)+a(\bullet\!-\!\bullet\!-\!\bullet)b(\bullet)^3\\
(b\star a)(\begin{smallmatrix}\bullet\\ |\\ \bullet\!-\!\bullet\end{smallmatrix}) &= a(\bullet)b(\begin{smallmatrix}\bullet\\ |\\ \bullet\!-\!\bullet\end{smallmatrix})+2a(\bullet\!-\!\bullet)b(\bullet)b(\bullet\!-\!\bullet)+a(\begin{smallmatrix}\bullet\\ |\\ \bullet\!-\!\bullet\end{smallmatrix})b(\bullet)^3
\end{aligned}
$$

$$
\begin{aligned}
\Delta_{CEM}(\bullet) &= \bullet\otimes\bullet\\
\Delta_{CEM}(\bullet\!-\!\bullet) &= \bullet\!-\!\bullet\otimes\bullet+\bullet^2\otimes\bullet\!-\!\bullet\\
\Delta_{CEM}(\bullet\!-\!\bullet\!-\!\bullet) &= \bullet\!-\!\bullet\!-\!\bullet\otimes\bullet+2\bullet\!-\!\bullet\otimes\bullet\!-\!\bullet+\bullet^3\otimes\bullet\!-\!\bullet\!-\!\bullet\\
\Delta_{CEM}(\begin{smallmatrix}\bullet\\ |\\ \bullet\!-\!\bullet\end{smallmatrix}) &= \begin{smallmatrix}\bullet\\ |\\ \bullet\!-\!\bullet\end{smallmatrix}\otimes\bullet+2\bullet\!-\!\bullet\otimes\bullet\!-\!\bullet+\bullet^3\otimes\begin{smallmatrix}\bullet\\ |\\ \bullet\!-\!\bullet\end{smallmatrix}
\end{aligned}
$$

Table 2.2: Substitution law $\star$ on B-series and coproduct Δ_{CEM}.

so that the substitution law $\star$ simply corresponds to the convolution product associated to Δ_{CEM}, see Table 2.2

$$b\star a=\mu_{\mathbb{R}}\circ(b\otimes a)\circ\Delta_{CEM}.$$

The Hopf tree algebra of Connes & Kreimer is defined as the free commutative algebra generated by all trees $\tau\in\mathcal{T}\cup\{\emptyset\}$, including the empty tree $\emptyset$, which is the unit element for multiplication. In contrast, the new Hopf tree algebra of Calaque, Ebrahimi-Fard & Manchon in [CEFM08] is generated by trees $\tau\in T\backslash\{\bullet\}$ with at least one edge, and the unit element of the algebra is identified with the tree $\bullet$ with no edge.

Remark 2.3.1 In a private communication, Dominique Manchon pointed out that there are two possible algebraic structures for the coproduct Δ_{CEM}:

Choice 1 The first choice is the new Hopf tree algebra given in [CEFM08], and we have for instance $\bullet\!-\!\bullet\!-\!\bullet\,\bullet^k=\bullet\!-\!\bullet\!-\!\bullet$, and $\bullet^k=\bullet$ for all $k\geq 1$. In practice, we simply omit all components $\bullet$ in formulas of Table 2.2, e.g.

$$\Delta_{CEM}(\begin{smallmatrix}\bullet\\ |\\ \bullet\!-\!\bullet\end{smallmatrix})=\begin{smallmatrix}\bullet\\ |\\ \bullet\!-\!\bullet\end{smallmatrix}\otimes\bullet+2\bullet\!-\!\bullet\otimes\bullet\!-\!\bullet+\bullet\otimes\begin{smallmatrix}\bullet\\ |\\ \bullet\!-\!\bullet\end{smallmatrix}$$

In [CEFM08], an elegant formula for the antipode S_{CEM} is also given (see a few terms in Table 2.3), and they obtain a new Hopf tree algebra, which is graded, commutative and not cocommutative, like the Hopf tree algrebra of Connes & Kreimer.

Choice 2 If one wishes to distinguish the empty tree $\emptyset$, the tree with only one vertex $\bullet$, and forests like $\bullet^k$, $k\geq 2$, one can consider the tree algebra of Connes & Kreimer presented in Sect. 2.2. However, the coproduct Δ_{CEM} yields a bialgebra which is not a Hopf algebra, because we would need virtual inverses like for the antipode, e.g. $S(\bullet^k)=(\bullet^k)^{-1}=\bullet^{-1}$. We also loose what is called connexity: there are infinitely many elements of degree zero (no edge): the unity $\emptyset$ (the empty tree), the tree with one vertex $\bullet$, and the forests $\bullet^k$.

If we consider the quotient of the bialgebra obtained in the second choice by the ideal generated by $\bullet-\emptyset$ (which is in fact a bi-ideal, so the quotient is a bialgebra), this is equivalent to identify again the empty tree $\emptyset$ with $\bullet$, and we recover exactly the Hopf tree algebra of the first choice. The convolution product $\star$ is defined in the same manner in

$$
\begin{aligned}
b^{\star -1}(\bullet) &= b(\bullet) = 1 \\
b^{\star -1}(\,\bullet\!\!-\!\!\bullet\,) &= -b(\,\bullet\!\!-\!\!\bullet\,) \\
b^{\star -1}(\,\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet\,) &= -b(\,\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet\,) + 2b(\,\bullet\!\!-\!\!\bullet\,)^2 \\
b^{\star -1}(\,\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet\,) &= -b(\,\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet\,) + 2b(\,\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet\,^2).
\end{aligned}
$$

$$
\begin{aligned}
S_{CEM}(\bullet) &= \bullet, \\
S_{CEM}(\,\bullet\!\!-\!\!\bullet\,) &= -\,\bullet\!\!-\!\!\bullet\,, \\
S_{CEM}(\,\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet\,) &= -\,\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet\, + 2\,\bullet\!\!-\!\!\bullet\,^2, \\
S_{CEM}(\,\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet\,) &= -\,\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet\, + 2\,\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet\,^2.
\end{aligned}
$$

Table 2.3: Inverse for the subtitution law $\star$ on B-series and antipode of the Hopf tree algebra.

both choices. A linear form b on the vector space generated by non-empty trees $\tau \in \mathcal{T}$ can be extended into a non trivial algebra morphism if and only if $b(\bullet) = 1$. In that case, all terms $a(\bullet), b(\bullet)$ disappear in the formulas of the convolution product $b \star a$.

Remark 2.3.2 Notice that if $b : \mathcal{T} \to \mathbb{R}$, $b(\bullet) = 1$ is the map of coefficients for backward error analysis of some method B-series integrator, then its inverse for the substitution law $\star$ is simply $b^{\star -1}(\tau) = b \circ S_{CEM}(\tau)$ (see Table 2.3), which yields the map of B-series coefficients for modifying vector fields integrators (see Sect. 1.3.1).

In [CEFM08], the authors show that their new Hopf tree algebra interact with the well-known Hopf tree algebra of Connes & Kreimer presented in Sect. 2.2, by means of a natural bicomodule structure. This also allows them to given alternate proofs of some algebraic results given in this chapter.

2.4 Algebraic properties of the substitution law for modified fields

The product $b \star a$ defines a group structure on the set $\left\{c : T \cup \{\emptyset\} \to \mathbb{R}\,;\, c(\emptyset) = 0, c(\bullet) = 1\right\}$ which represents vector fields. In this section, we shall study its algebraic properties. In particular, we show that it is compatible with the composition (2.3) of B-series for the Butcher group.

Proposition 2.4.1 *Let $a, b, \widetilde{b}, c, \widetilde{c} : \mathcal{T} \cup \{\emptyset\} \to \mathbb{R}$ be mappings satisfying $a(\emptyset) = 1$ and $b(\emptyset) = \widetilde{b}(\emptyset) = 0$. The following properties hold for all λ and μ in $\mathbb{R}$:*

$$
\begin{aligned}
b \star \delta_\emptyset &= \delta_\emptyset, && (\delta_\emptyset \textit{ absorbing element for } \star), && (2.8)\\
b \star \delta_\bullet &= \delta_\bullet \star b = b, && (\delta_\bullet \textit{ unit element for } \star), && (2.9)\\
b \star (\lambda c + \mu \widetilde{c}) &= \lambda (b \star c) + \mu (b \star \widetilde{c}), && (\textit{right-sided linearity of } \star), && (2.10)\\
(\widetilde{b} \star b) \star c &= \widetilde{b} \star (b \star c), && (\textit{associativity of } \star), && (2.11)\\
b \star (a \cdot c) &= (b \star a) \cdot (b \star c), && (\textit{right-sided distributivity of } \star \textit{ on } \cdot), && (2.12)\\
(b \star a)^{\cdot -1} &= b \star a^{\cdot -1}, && && (2.13)\\
a^{\cdot -1} &= (a - \delta_\emptyset) \star (\delta_\emptyset + \delta_\bullet)^{\cdot -1}, && && (2.14)
\end{aligned}
$$

Remark 2.4.2 We recall that the notation $a^{\cdot -1}$ denotes the inverse (2.4) for the standard composition $\cdot$ of B-series as defined in (2.3). We can notice that the last formula (2.14) is equivalent to formula (2.4). As a matter of fact, given a tree τ, we have $(\delta_\emptyset + \delta_\bullet)^{-1}(\tau) = (-1)^{|\tau|}$, and for all partitions p^τ of τ, $|\chi(p^\tau)| = \#(p^\tau)$. Therefore, formula (2.14) together with (1.27) gives formula (2.4).

An alternative proof of these algebraic properties is given in [CEFM08, Corollary 10, Prop. 11, Prop. 12] in a pure algebraic manner.

Proof. (of Proposition 2.4.1) Formulas (2.8), (2.9), (2.10) are an immediate consequence of (1.27). Now, consider the fields $g, \widetilde{g} : \mathbb{R}^d \to \mathbb{R}^d$ defined by

$$\begin{aligned} h\widetilde{g}(y) &= B_f(\widetilde{b}, y), \\ hg(y) &= B_{\widetilde{g}}(b, y). \end{aligned}$$

On one hand, we have $hg(y) = B_f(\widetilde{b} \star b, y)$, and therefore, $B_g(c, y) = B_f((\widetilde{b} \star b) \star c, y)$. On the other hand, we have $B_g(c, y) = B_{\widetilde{g}}(b \star c, y)$, which leads to $B_g(c, y) = B_f(\widetilde{b} \star (b \star c), y)$. This proves (2.11).

Consider the field $g : \mathbb{R}^d \to \mathbb{R}^d$ defined by

$$hg(y) = B_f(b, y).$$

We have $B_g(a \cdot c, y) = B_g(a, B_g(c, y))$. However, for all mapping $s : \mathcal{T} \cup \{\emptyset\} \to \mathbb{R}$, $B_g(s, y) = B_f(b \star s, y)$. By taking s successively equal to $a \cdot c$, c and a, we get

$$B_f(b \star (a \cdot c), y) = B_f(b \star a, B_f(b \star c, y)).$$

Therefore, $B_f(b \star (a \cdot c), y) = B_f((b \star a) \cdot (b \star c), y)$, which leads to (2.12).

The mapping a is invertible for the composition law $\cdot$ because $a(\emptyset) = 1$. Applying (2.12) to $c = a^{-1}$ together with (2.8) gives (2.13).

Finally, notice $(a - \delta_\emptyset) \star (\delta_\emptyset + \delta_\bullet) \overset{(2.10)}{=} (a - \delta_\emptyset) \star \delta_\emptyset + (a - \delta_\emptyset) \star \delta_\bullet \overset{(2.8)-(2.9)}{=} \delta_\emptyset + (a - \delta_\emptyset) = a$. Formula (2.13) then leads to (2.14). □

2.5 The logarithmic map

Equation (1.25) giving the coefficients $b(\tau)$ for backward error analysis can be solved as explicitly computed in terms of $a(\tau)$ (with $a(\emptyset) = 1$). Indeed, let $\omega : \mathcal{T} \cup \{\emptyset\} \to \mathbb{R}$ denote the inverse element of $\frac{1}{\gamma} - \delta_\emptyset$ for $\star$. This gives a new formula to compute the coefficients $b(\tau)$ for backward error analysis:

$$b(\tau) = (a - \delta_\emptyset) \star \omega(\tau). \tag{2.15}$$

Since, $\omega \star (\frac{1}{\gamma} - \delta_\emptyset) = \delta_\bullet$, we get by adding $\delta_\emptyset$ on both sides of the equation

$$\omega \star \frac{1}{\gamma} = \delta_\emptyset + \delta_\bullet.$$

The coefficients $\omega(\tau)$ can thus be interpreted as the coefficients of the modified field obtained by backward error analysis, for the Euler explicit method $y_1 = y_0 + hf(y_0)$, corresponding to $a = \delta_\emptyset + \delta_\bullet$. They may be computed by induction using formula (1.27).

Note that the same ω already appears in [Mur06] with the same definition and a recurrence formula is also given therein. This corresponds to the *formal* logarithm of $B_f(a, y)$

$$\log(a) = (a - \delta_\emptyset) \star \omega \tag{2.16}$$

as defined in [Mur06]. Algebraic properties of the logarithmic map are studied in [CEFM08, Sect. 9] using quasi-shuffle products, which provides alternative proofs of Lemma 2.5.1 and Propositions 2.5.2 and 2.5.3.

2.5.1 The ω map

It is now of great interest to study the coefficients $\omega(\tau)$. In order to obtain recurrence formulas involving only ω, we first derive properties of ω that are mostly inherited from those of $\frac{1}{\gamma}$ and which shall turn out to be interesting per se. To this aim, we recall in particular the following relations (see for instance [CFM06]).

Lemma 2.5.1 *The coefficients $\frac{1}{\gamma}$ of the exact flow satisfy the following relation for all $m \geq 2$:*

$$\forall (t_1, \dots, t_m) \in \mathcal{T}^m, \quad \sum_{i=1}^{m} \frac{1}{\gamma}(t_i \circ \prod_{j \neq i} t_j) = \prod_{i=1}^{m} \frac{1}{\gamma}(t_i), \tag{2.17}$$

This relation corresponds to the preservation of polynomial invariants of degree m, and it is shown in [CFM06] that if a B-series method $y_{n+1} = B(\cdot, a)(y_n)$ (with $a(\emptyset) = a(\bullet) = 1$) satisfies this relation with $m = 2$ and $m = 3$ then $a = \frac{1}{\gamma}$ and it reduces to the exact flow.

Here, $\circ$ and $\times$ denote respectively the *Butcher* and the *merging* products (see for instance [HLW06], Definition III.3.7),

$$u \circ v = [u_1, \dots u_n, v] \qquad u \times v = [u_1, \dots u_n, v_1, \dots v_n],$$

where $u = [u_1, \dots, u_n] \in \mathcal{T}$ and $u = [v_1, \dots, v_n] \in \mathcal{T}$. For the sake of brevity, we shall sometimes use in the sequel the following notation:

$$c\Big(\sum_i t_i\Big) := \sum_i c(t_i)$$

for any mapping $c : \mathcal{T} \to \mathbb{R}$ and any t_i's in $\mathcal{T}$.

Relations (2.17) fully determine the (scaled) γ-function. For $m = 2$, the induced relation for ω is given in the next proposition:

Proposition 2.5.2 *The coefficients ω of the modified field (obtained by backward analysis) of the explicit Euler method satisfy the following relation:*

$$\forall\, (u, v) \in \mathcal{T}^2, \quad \omega(u \circ v) + \omega(v \circ u) + \omega(u \times v) = 0. \tag{2.18}$$

Proof. We show that a map $b : \mathcal{T} \cup \{\emptyset\} \to \mathbb{R}$ with $b(\emptyset) = 0$ satisfies (2.18) if and only if its inverse $b^{\star -1}$ for the substitution law satisfies (2.17) with $m = 2$. This result can be proved using the ideas of the Switching Lemma [HLW06, Lemma III.3.8]. A direct proof can be obtained using the substitution law formula, see [CHV05, Prop. 4.4] for details. An other algebraic proof of Prop. 2.5.2 using quasi-shuffle products is given in [CEFM08, Sect. 9]. □

We give the full generalization of Proposition 2.5.2. Its proof relies on the same ideas as the proof of Proposition 2.5.2, and can be found in [CHV05].

Proposition 2.5.3 *The coefficients ω of the modified equation of the explicit Euler method satisfy the following relation for all m-uplets, $m \geq 2$, of trees $(u_1, \ldots, u_m) \in \mathcal{T}^m$:*

$$\sum_{\substack{I \cup J = \{1,\ldots,m\},\\ I \cap J = \emptyset}} \omega\Big(\times_{i\in I} u_i \circ \prod_{j\in J} u_j \Big) = 0, \tag{2.19}$$

with the conventions $u \circ e = u$ and $e \circ u = e$.

Remark 2.5.4 If we take $u_1 = u_2 = \ldots = u_m = \bullet$, relation (2.19) becomes:

$$\sum_{i=1}^{m} \frac{m!}{i!(m-i)!} \omega([\bullet^{m-i}]) = 0.$$

Since $\omega(\bullet) = 1$, this shows that

$$\omega([\bullet^{i}]) = B_i$$

where the B_i's are the Bernoulli numbers. Hence, we have

$$\omega(\begin{array}{c}\bullet\\|\end{array}) = B_1 = -1/2, \quad \omega(\begin{array}{c}\bullet\\|\\\bullet\end{array}) = B_2 = 1/6, \quad \omega(\begin{array}{ccc}&\bullet&\\&|&\\\bullet&&\bullet\end{array}) = B_3 = 0, \ldots$$

Notice that this result is also an immediate consequence of [HLW06, Example IX.7.1].

More generally, the values of ω can be computed from Formula (2.19) by induction, as the following lemma shows.

Proposition 2.5.5 *The coefficients $\omega(\tau)$ can be computed recursively for increasing values of $\gamma(\tau)$.*

Proof. The proof proceeds by induction on the values of γ. For $\gamma(\tau) = 1$, we have $\omega(\bullet) = 1$. Assume the value of ω is determined for all trees with $\gamma \leq n-1$ with $n \geq 2$ and consider $\tau = [\tau_1, \ldots, \tau_m]$ such that $\gamma(\tau) = n$. By (2.19) with the $m+1$ trees $\tau_0 = \bullet, \tau_1, \ldots, \tau_m$, we have

$$(k+1)\omega(\tau) + \sum_{j=0, \tau_j \neq \bullet}^{m} \omega(\tau_j \circ \prod_{i\neq j} \tau_i) + \alpha(\tau_0, \tau_1, \ldots, \tau_m) = 0$$

where $\alpha(\tau_0, \tau_1, \ldots, \tau_m)$ involves a sum of $\omega(u)$'s for u of the form

$$u = \times_{l=0}^{i} \tau_{j_l} \circ \prod_{l=i+1}^{m} \tau_{j_l}, \quad i \geq 1.$$

This defines $\omega(\tau)$ uniquely since

$$\gamma\Big(\times_{l=0}^{i} \tau_{j_l} \circ \prod_{l=i+1}^{m} \tau_{j_l} \Big) = \frac{|\tau| - i}{|\tau|\,|\tau_{j_1}| \ldots |\tau_{j_i}|}\, \gamma(\tau) < \gamma(\tau),$$

and similarly $\gamma(\tau_j \circ \prod_{i\neq j} \tau_i) < \gamma(\tau)$.

Remark 2.5.6 Another relation can be obtained from the functional equality

$$\widetilde{f}_h(y + hf(y)) = \widetilde{f}_h(y) + hf'(y)\widetilde{f}_h(y),$$

where $\widetilde{f}_h(y) = h^{-1}B_f(\omega, y)$. This translates in terms of B-series in

$$(\delta_\emptyset + \delta_\bullet) \cdot \omega = \omega + \tilde{\omega}$$

where $\tilde{\omega}$ is defined by

$$\tilde{\omega}([t]) = \omega(t) \text{ and } \tilde{\omega}(u) = 0 \text{ if } u \text{ is not of the form } [t].$$

Hence, ω can also be computed recursively for increasing values of γ by

$$\omega(\tau) = \sum_{\substack{p^\tau \in \mathcal{AP}(\tau)\backslash\{\tau\}, \\ P^*(p^\tau) = \{\bullet, \ldots, \bullet\}}} \omega([R(p^\tau)]).$$

τ	$\emptyset$	$\bullet$							
$\omega(\tau)$	0	1	$-\frac{1}{2}$	$\frac{1}{6}$	$\frac{1}{3}$	0	$-\frac{1}{6}$	$-\frac{1}{4}$	$-\frac{1}{12}$
τ									
$\omega(\tau)$	$-\frac{1}{30}$	$\frac{1}{30}$	$\frac{1}{5}$	$\frac{1}{20}$	$\frac{1}{30}$	$\frac{3}{20}$	$\frac{1}{60}$	$-\frac{1}{60}$	$\frac{1}{10}$

Table 2.4: Coefficients $\omega(\tau)$ for trees of order ≤ 5.

2.5.2 Hamiltonian fields and symplectic methods

A remarkable result of Calvo and Sanz-Serna [CSS94] gives an algebraic characterization of symplectic B-series, while the characterization of Hamiltonian vector fields has been obtained in [Hai94].

Theorem 2.5.7 *(Hairer, [Hai94]) The set of mappings*

$$\{b : \mathcal{T} \cup \{\emptyset\} \to \mathbb{R} \ ; \ b(\emptyset) = 0, \forall u, v \in \mathcal{T}, \ b(u \circ v) + b(v \circ u) = 0\}, \tag{2.20}$$

is the set of Hamiltonian fields. More precisely, the vector field $g(y) = \frac{1}{h}B_f(b, y)$ with $b(\emptyset) = 0$ is Hamiltonian for all $f(y) = J^{-1}\nabla H(y)$ if and only if b belongs to this set.

Theorem 2.5.8 *(Calvo & Sanz-Serna, [CSS94]) The set of mappings*

$$\{a : \mathcal{T} \cup \{\emptyset\} \to \mathbb{R} \ ; \ a(\emptyset) = 1, \ \forall u, v \in \mathcal{T}, \ a(u \circ v) + a(v \circ u) = a(u)a(v)\},$$

is the set of symplectic mappings. More precisely, the numerical method $\Phi_h^f(y) = B_f(a, y)$ is symplectic for all $f(y) = J^{-1}\nabla H(y)$ if and only if a belongs to this set.

An important result in geometric integration (see Theorem IX.3.1 in [HLW06]) states that a B-series is symplectic if and an only if the B-series vector fields for backward error analysis (or for modifying integrators) is Hamiltonian. This result can be interpreted in terms of abstract group mappings in the following result.

Theorem 2.5.9 *The logarithmic map* log *in (2.16) establishes a one-to-one correspondence between the subgroup of symplectic B-series in Theorem 2.5.8 (in the Butcher group equipped with the composition $\cdot$ of B-series) and the subgroup Hamiltonian B-series vector fields in Theorem 2.5.7 (equipped with the substitution law $\star$).*

Notice that a similar statement holds for symmetric B-series methods. The corresponding subgroup for B-series vector fields is $\{b : \mathcal{T} \cup \{\emptyset\} \to \mathbb{R}; b(|\tau|) = 0 \text{ for } |\tau| \text{ even}\}$.

2.6 Extension to P-series

To conclude this chapter, we explain how the results for the substitution law on B-series can be extended straightforwardly to P-series. Let us mention that it can be also extended to the more general S-series [CFM06]. Many numerical methods, such as partitioned Runge-Kutta methods, can me written as P-series (see [HLW06]). Given two fields $f^{[1]}, f^{[2]} : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^d$, we consider the partitioned system of the form

$$\dot{p} = f^{[1]}(p, q), \quad \dot{q} = f^{[2]}(p, q).$$

Let $\mathcal{TP} = \{\bullet, \circ, \overset{\bullet}{\bullet}, \overset{\circ}{\bullet}, \overset{\bullet}{\circ}, \ldots\}$ denote the set of bi-coloured trees, and $\emptyset_p, \emptyset_q$ denote empty trees. We first recall the definition of a P-series.

Definition 2.6.1 *Consider a mapping $a : \mathcal{TP} \cup \{\emptyset_p, \emptyset_q\} \to \mathbb{R}$. A P-series is a series of the form*

$$P(f, a)(p, q) = \begin{pmatrix} a(\emptyset_p)p \\ a(\emptyset_q)q \end{pmatrix} + h\begin{pmatrix} a(\bullet)f^{[1]}(p, q) \\ a(\circ)f^{[2]}(p, q) \end{pmatrix} + h^2\begin{pmatrix} a(\overset{\bullet}{\bullet})(f_p^{[1]}f^{[1]}) + a(\overset{\circ}{\bullet})(f_q^{[1]}f^{[2]}) \\ a(\overset{\bullet}{\circ})(f_p^{[2]}f^{[1]}) + a(\overset{\circ}{\circ})(f_q^{[2]}f^{[2]}) \end{pmatrix} + \ldots,$$

where terms like $f_p^{[2]}f^{[1]}$, $f_q^{[1]}f^{[2]}, \ldots$ are evaluated at (p, q).

We may now give the following extension of Theorem 1.3.1.

Theorem 2.6.2 *Let $a, b : \mathcal{TP} \cup \{\emptyset_p, \emptyset_q\} \to \mathbb{R}$ be two mappings with $b(\emptyset_p) = b(\emptyset_q) = 0$. Given two fields $f^{[1]}, f^{[2]} : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^d$, consider the (h-dependent) fields $g^{[1]}, g^{[2]} : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^d$ defined by*

$$h\begin{pmatrix} g^{[1]}(p, q) \\ g^{[2]}(p, q) \end{pmatrix} = P(f, b)(p, q).$$

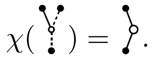
Then, there exists a mapping $b \star a : \mathcal{TP} \cup \{\emptyset_p, \emptyset_q\} \to \mathbb{R}$ satisfying

$$P(g, a)(p, q)) = P(f, b \star a)(p, q))$$

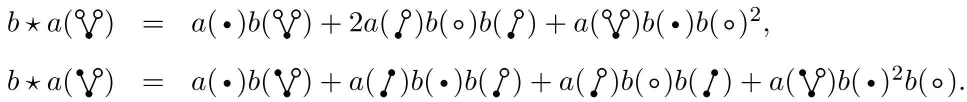
and $b \star a$ is defined by $b \star a(\emptyset_p) = a(\emptyset_p)$ and the same formula (1.27) in Theorem 1.3.1 where analogously to Sect. 1.4.1, $\mathcal{P}(\tau)$ denotes the set of partitions of τ.

Remark 2.6.3 Similarly to Sect. 1.4.1, the skeleton $\chi(p^\tau) \in \mathcal{TP}$ of a partition $p^\tau \in \mathcal{P}(\tau)$ of a bi-coloured tree $\tau \in \mathcal{TP}$ is the tree obtained by replacing in p^τ each tree of $P(p^\tau)$ by

a single vertex *with the same color as the color of its root*, and then dashed edges by solid ones. For instance, we have



We finally give a few examples for trees of order 3



Results of previous sections for B-series may now be extended to P-series. For instance, if the method $(p_1, q_1) = \Phi_{f,h}(p_0, q_0)$ can be written as a P-series $P(f, a)(p, q)$ with $a(\emptyset_p) = a(\emptyset_q) = 1$, then the real coefficients $b(\tau)$ for backward error analysis are defined by

$$b = (a - \delta_\emptyset) \star \omega,$$

where the $\omega(\tau)$ and the $\delta_\emptyset(\tau)$ with $\tau \in \mathcal{TP}$ have the same values as for mono-coloured trees.

Chapter 3

A high-order geometric integrator for the motion of a rigid body

Note: Sections 3.1,3.2,3.3,3.4 are taken without modification from the article [HV06] in collaboration with E. Hairer. Section 3.5 is taken from the article [Vil08] but with a slightly different presentation. Results of Section 3.6 where obtained while the author was a Program Participant at the Isaac Newton Institute programme on Highly Oscillatory Problems on May 2007.

The motion of a rigid body, relative to a fixed coordinate system, is described by an orthogonal matrix $Q(t)$. Its dynamics is determined by a Hamiltonian system constrained to the Lie group $SO(3)$. In the absence of an external potential, the Hamiltonian is given by $T = \frac{1}{2}\big(I_1\omega_1^2 + I_2\omega_2^2 + I_3\omega_3^2\big)$, where $(\omega_1, \omega_2, \omega_3)^T$ is the angular velocity in the body frame and the constants I_1, I_2, I_3 are the three moments of inertia of the rigid body. The equations of motion can be written in terms of the angular momentum $y = (y_1, y_2, y_3)^T$, $y_j = I_j\omega_j$, as follows:

$$\dot{y} = \widehat{y}\, I^{-1} y, \qquad \dot{Q} = Q\, \widehat{I^{-1}y}, \tag{3.1}$$

where $I = \operatorname{diag}(I_1, I_2, I_3)$ (see [HLW06, Sect. VII.5]). We use the standard *hatmap* notation for the correspondence between vectors and skew-symmetric matrices,

$$y = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix}, \qquad \widehat{y} = \begin{pmatrix} 0 & -y_3 & y_2 \\ y_3 & 0 & -y_1 \\ -y_2 & y_1 & 0 \end{pmatrix}. \tag{3.2}$$

We notice that the flow of (3.1) exactly conserves the energy and the angular momentum relative to the fixed frame. In formulae, this means that Qy and

$$C(y) = \frac{1}{2}\Big(y_1^2 + y_2^2 + y_3^2\Big) \qquad \text{and} \qquad H(y) = \frac{1}{2}\Big(\frac{y_1^2}{I_1} + \frac{y_2^2}{I_2} + \frac{y_3^2}{I_3}\Big) \tag{3.3}$$

(Casimir and Hamiltonian) are first integrals of the system.

As numerical integrator we consider the Discrete Moser–Veselov (DMV) algorithm [MV91] with update for Q_n proposed by [LS96]. It can be written as

$$\widehat{y}_{n+1} = \omega_n\, \widehat{y}_n\, \omega_n^T, \qquad Q_{n+1} = Q_n\, \omega_n^T, \tag{3.4}$$

where the orthogonal matrix ω_n is computed from

$$\omega_n^T D - D\, \omega_n = h\, \widehat{y}_n. \tag{3.5}$$

Here, $y_n \approx y(t_n)$, $Q_n \approx Q(t_n)$, and h is the stepsize. The entries of the diagonal matrix $D = \operatorname{diag}(d_1, d_2, d_3)$ are determined by

$$d_1 + d_2 = I_3, \quad d_2 + d_3 = I_1, \quad d_3 + d_1 = I_2, \tag{3.6}$$

so that $\omega^T I\,\omega = \operatorname{trace}(\widehat{\omega} D\,\widehat{\omega}^T)$. It is shown in [MZ05] that this discretisation is equivalent to the RATTLE algorithm which is designed to solve general constrained Hamiltonian systems (see also [LR04, Chap. 8] and [HLW06, Sect. VII.5.3]).

This Discrete Moser–Veselov algorithm (3.4)-(3.5) is an excellent geometric integrator. It exactly conserves (up to round-off) the Hamiltonian $H(y)$, the angular momentum Qy (in the fixed frame) and, since Q is orthogonal, also the Casimir $C(y)$. It is a symmetric (time-reversible) and symplectic discretisation of (3.1) and therefore well suited for long-time integrations.

The DMV algorithm gives a second order approximation to the solution of (3.1), and this low order is its only drawback. Based on the ideas of [CHV07b] we propose here a modification that allows us to increase the order arbitrarily high, so that a significantly improved accuracy can be obtained. The modification simply consists in replacing the moments of inertia I_j by expressions that depend in a suitable way on $H(y)$ and $C(y)$ (Sect. 3.1). Numerical experiments and a theoretical justification are presented in Sects. 3.2 and 3.3, respectively. An important suggestion for the implementation of the algorithm using quaternions (Sect. 3.4) and a MAPLE script for the computation of the modified moments of inertia is given in Appendix (Sect. A). It is natural to study the accumulation with time of round-off errors now that we have a developed an efficient high-order geometric integrator. This is done in Sect. 3.5. Finally, in Sect. 3.6 we present a modification of the Proprecessed DMV algorithm that allows to compute efficiently the tangent map, in the context of the conjugate points computations.

Let us mention that a time transformation has been proposed recently in [MZ05] which improves the order of the DMV algorithm for the angular momentum y but not for the rotation matrix Q. Our modification for the y variables is closely related to but different from this time transformation.

3.1 Preprocessed DMV algorithm

A technique for increasing the order of numerical methods has recently been proposed in [CHV07b] (modifying vector field integrators). It consists in applying the same numerical scheme to a modified differential equation. In the context of the equations of motion for the free rigid body, we consider a modified equation which consists in replacing the moments of inertia I_j by $\widetilde{I}_j = \widetilde{I}_j(y)$ of the form ($j = 1, 2, 3$)

$$\frac{1}{\widetilde{I}_j} = \frac{1}{I_j}\Big(1 + h^2 s_3(y) + h^4 s_5(y) + \dots\Big) + h^2 d_3(y) + h^4 d_5(y) + \dots\ . \tag{3.7}$$

In the DMV algorithm we only have to use $\widetilde{D} = \operatorname{diag}(\widetilde{d}_1, \widetilde{d}_2, \widetilde{d}_3)$ instead of D, where the $\widetilde{d}_j$ are computed from $\widetilde{I}_j = \widetilde{I}_j(y_n)$ via the relations (3.6).

Theorem 3.1.1 *There exist two formal series,*

$$\begin{aligned} 1 + h^2 s_3(y) + h^4 s_5(y) + \dots &= s\big(H(y), C(y)\big), \\ h^2 d_3(y) + h^4 d_5(y) + \dots &= d\big(H(y), C(y)\big), \end{aligned}$$

depending on y only via $H(y)$ and $C(y)$, such that the DMV algorithm (3.4)-(3.5) applied with $\widetilde{I}_j(y_n)$ from (3.7) yields the exact solution of (3.1) in the sense of formal power series in h. The first terms of these series are given in Table 3.1 (see also Sect. A).

The proof of this theorem is postponed to Sect. 3.3. We notice that the modified differential equation

$$\dot{y} = \widehat{y}\ \widetilde{I}(y)^{-1}y, \qquad \dot{Q} = Q\ \widehat{\widetilde{I}(y)^{-1}y}, \tag{3.8}$$

with $\widetilde{I}(y)$ from (3.7) shares most of the geometric properties with that of (3.1). It still has Qy, the Casimir $C(y)$, and the Hamiltonian $H(y)$ as first integrals. For the angular momentum this is true for general $s_j(y)$ and $d_j(y)$; for the Hamiltonian only if they depend exclusively on $H(y)$ and $C(y)$. However, the Hamiltonian structure is inherited only if $\widetilde{I}(y)^{-1}y$ is the gradient of a scalar function. This is the case when the series in (3.7) are truncated after the h^2 term, but not in general.

Theorem 3.1.1 suggests the following modification of the DMV algorithm.

Algorithm 3.1.2 (Preprocessed DMV of order $2r$)

1. *Compute the modified moments of inertia $\widetilde{I}_1, \widetilde{I}_2, \widetilde{I}_3$ from (3.7) truncated after the h^{2r-2} terms and evaluated at y_n.*

2. *Apply the DMV algorithm (3.4)-(3.5) to a rigid body with the moments of inertia $\widetilde{I}_1, \widetilde{I}_2, \widetilde{I}_3$ instead of I_1, I_2, I_3.*

Table 3.1: Scalar functions for the preprocessed DMV algorithm

$$\delta = I_1 I_2 I_3, \qquad \sigma_a = I_1^a + I_2^a + I_3^a, \qquad \tau_{b,c} = \frac{I_2^b + I_3^b}{I_1^c} + \frac{I_3^b + I_1^b}{I_2^c} + \frac{I_1^b + I_2^b}{I_3^c},$$

$$s_3(y) = -\frac{\sigma_{-1}}{3}\, H(y) + \frac{\sigma_1}{6\delta}\, C(y), \qquad d_3(y) = \frac{\sigma_1}{6\delta}\, H(y) - \frac{1}{3\delta}\, C(y),$$

$$s_5(y) = \frac{3\sigma_1 + 2\delta\sigma_{-2}}{60\delta}\, H(y)^2 + \frac{1 - \tau_{1,1}}{30\delta}\, C(y)H(y) + \frac{\sigma_2 - \delta\sigma_{-1}}{30\delta^2}\, C(y)^2,$$

$$d_5(y) = -\frac{9 + \tau_{1,1}}{60\delta}\, H(y)^2 + \frac{6\delta\sigma_{-1} - \sigma_2}{60\delta^2}\, C(y)H(y) - \frac{\sigma_1}{60\delta^2}\, C(y)^2,$$

$$\begin{aligned} s_7(y) &= \frac{15 - \delta\sigma_{-3} - 2\tau_{1,1}}{630\delta}\, H(y)^3 + \frac{6\delta\tau_{1,2} - 100\delta\sigma_{-1} + 53\sigma_2}{2520\delta^2}\, C(y)H(y)^2 \\ &+ \frac{9\sigma_1 + 10\delta\sigma_{-2} - 6\tau_{2,1}}{420\delta^2}\, C(y)^2H(y) + \frac{4\delta + 17\sigma_3 - 15\delta\tau_{1,1}}{2520\delta^3}\, C(y)^3, \end{aligned}$$

$$\begin{aligned} d_7(y) &= \frac{9\delta\sigma_{-1} + \delta\tau_{1,2} - 11\sigma_2}{1260\delta^2} H(y)^3 + \frac{47\sigma_1 + 13\tau_{2,1} - 38\delta\sigma_{-2}}{2520\delta^2} C(y)H(y)^2 \\ &+ \frac{\sigma_3 + 2\delta\tau_{1,1} - 85\delta}{1260\delta^3}\, C(y)^2H(y) + \frac{34\delta\sigma_{-1} - 19\sigma_2}{2520\delta^3}\, C(y)^3. \end{aligned}$$

For instance, the preprocessed version of order 4 reads

$$\begin{aligned}
\frac{1}{\widetilde{I}_j} &= \frac{1}{I_j}\Big(1 + h^2 s_3(y_n)\Big) + h^2 d_3(y_n), \quad j = 1,2,3,\\
s_3(y_n) &= -\frac{1}{3}\Big(\frac{1}{I_1} + \frac{1}{I_2} + \frac{1}{I_3}\Big) H(y_n) + \frac{I_1 + I_2 + I_3}{6\, I_1\, I_2\, I_3}\, C(y_n),\\
d_3(y_n) &= \frac{I_1 + I_2 + I_3}{6\, I_1\, I_2\, I_3}\, H(y_n) - \frac{1}{3\, I_1\, I_2\, I_3}\, C(y_n).
\end{aligned}$$

Proposition 3.1.3 *The numerical solution obtained with Algorithm 3.1.2 satisfies the following properties:*

- *it has order $2r$;*
- *it exactly preserves Qy, $C(y)$, and $H(y)$;*
- *it is symmetric (time-reversible);*
- *restricted to the angular momentum y, it is a Poisson integrator.*

Proof. By Theorem 3.1.1 the error after one step is a $\mathcal{O}(h^{2r+1})$ perturbation of the exact flow. This implies that the method is of order $2r$.

One step of Algorithm 3.1.2 is precisely the DMV method with I_j replaced by the constant value $\widetilde{I}_j(y_n)$. Hence, it exactly conserves Qy, $C(y)$ and $\widetilde{H}(y)$, where $\widetilde{H}(y) = \frac{1}{2}\sum_j \widetilde{I}_j(y_n)^{-1} y_j^2$. Due to the particular structure in (3.7) we have

$$\widetilde{H}(y) = \big(1 + h^2 s_3(y_n) + \ldots\big) H(y) + \big(1 + h^2 d_3(y_n) + \ldots\big) C(y),$$

and the conservation of $C(y)$ and $\widetilde{H}(y)$ implies that of $H(y)$.

The statement on the symmetry follows from the exact conservation of $H(y)$ and $C(y)$, so that $\widetilde{I}_j(y_{n+1}) = \widetilde{I}_j(y_n)$. In Sect. 3.3.3 we shall show that this algorithm is a Poisson integrator for the angular momentum. □

Remark 3.1.4 The time transformation of the DMV algorithm (3.4)-(3.5) proposed in [MZ05] is equivalent to replace the stepsize h by a modified stepsize $\widetilde{h}$ of the form

$$\widetilde{h} = h\big(1 + h^2 s_3(y_n) + h^4 s_5(y_n) + \ldots\big). \tag{3.9}$$

It is possible to complement this time transformation to obtain high order also for the rotation matrix Q. Since the matrix ω_n^T is orthogonal, it can be represented by a Cayley transform

$$\omega_n^T = \Big(Id + \frac{h}{2}\widehat{I^{-1}Y_n}\Big)\Big(Id - \frac{h}{2}\widehat{I^{-1}Y_n}\Big)^{-1}, \tag{3.10}$$

where Id stands for the identity matrix, and Y_n is a vector close to y_n. Now, one can use the new update

$$Q_{n+1} = Q_n\, \widetilde{\omega}_n^T,$$

where the matrix $\widetilde{\omega}_n^T$ is defined as in (3.10), but with modified moments of inertia $\widetilde{I}(y_n) = \mathrm{diag}\,(\widetilde{I}_1, \widetilde{I}_2, \widetilde{I}_3)$ of the form (3.7), instead of the diagonal matrix I.

We notice that this modification of the DMV algorithm is not equivalent to the preprocessed DMV Algorithm 3.1.2 (for y and also for Q). The scalar functions $s_k(y), d_k(y)$ in (3.9) and in $\widetilde{I}(y_n)$ are the same as in Table 3.1 for $k = 3$ but not for $k > 3$. Our numerical tests revealed that this modification of DMV is inferior to that of Algorithm 3.1.2.

3.2 Comparison with other rigid body integrators

In this section, we compare the preprocessed Discrete Moser–Veselov Algorithm 3.1.2 (denoted DMV$2r$), with several free rigid body integrators[1]:

- DMV, the *Discrete Moser–Veselov algorithm* (3.4)-(3.5),
- IMR$2r$, the *implicit midpoint rule* for $r = 1$, and the *modifying implicit midpoint rule* for $r > 1$, introduced in [CHV07b],
- JEM$2r$ [CS06] where the Euler equations are integrated exactly using Jacobi elliptic functions, and the rotation matrix is approximated using a truncated Magnus series,
- SR$2r$, the so-called *Symmetric+Rotation Splitting* algorithm based on the Strang splitting $H(y) = \frac{1}{2}R(y) + S(y) + \frac{1}{2}R(y)$ where

$$R(y) = \Big(\frac{1}{I_1} - \frac{1}{I_2}\Big)\frac{y_1^2}{2}, \quad S(y) = \frac{1}{2}\Big(\frac{y_1^2 + y_2^2}{I_2} + \frac{y_3^2}{I_3}\Big),$$

  combined with a *composition method* of order $2r$ (see for instance [HLW06]). For the numerical experiments, SR4 and SR6 are chosen as compositions of respectively 5 and 9 times the basic method SR2.

Geometric properties In Table 3.2, we compare the geometric properties of the above integrators. Column "symplectic" indicates whether the method is a symplectic integrator. In the context of backward error analysis (see Sect. 3.3.3) this means that the modified differential equation is of the form

$$\dot{y} = \widehat{y}\,\nabla H_h^{[2r]}(y), \qquad \dot{Q} = Q\,\widehat{\nabla H_h^{[2r]}}(y).$$

If the modified equation has this form only for the y component, the method is still a Poisson integrator. This is indicated in column "Poisson".

Table 3.2: Geometric properties

integrator	order of accuracy		exact preservation of quadratic invariants			Poisson	symplectic
	y	Q	Qy	$C(y)$	$H(y)$		
DMV$2r$	$2r$	$2r$	√	√	√	√	no
DMV	2	2	√	√	√	√	√
IMR$2r$	$2r$	$2r$	√	√	√	no[1]	no
JEM$2r$	exact	$2r$	no	√	√	√	no
SR$2r$	$2r$	$2r$	√	√	no	√	√

Numerical experiments We consider the system (3.1) for the free rigid body on the interval $[0, 10]$ with two different sets of moments of inertia: an asymmetric body with $I_1 = 0.6$, $I_2 = 0.8$, $I_3 = 1.0$ (as in [CS06]) and a flat body with $I_1 = 0.345$, $I_2 = 0.653$, $I_3 =$

[1] The FORTRAN codes are available from the authors upon request (see also Appendix B).

[1] It can be shown that IMR$2r$ ($r \geq 1$) is Poisson with respect to a different bracket generated by $\gamma_h^{[2r]}(y)\,\widehat{y}$, where $\gamma_h^{[2r]}(y)$ is a scalar function (see [FGS05]). We thank an anonymous referee for drawing our attention to this fact.

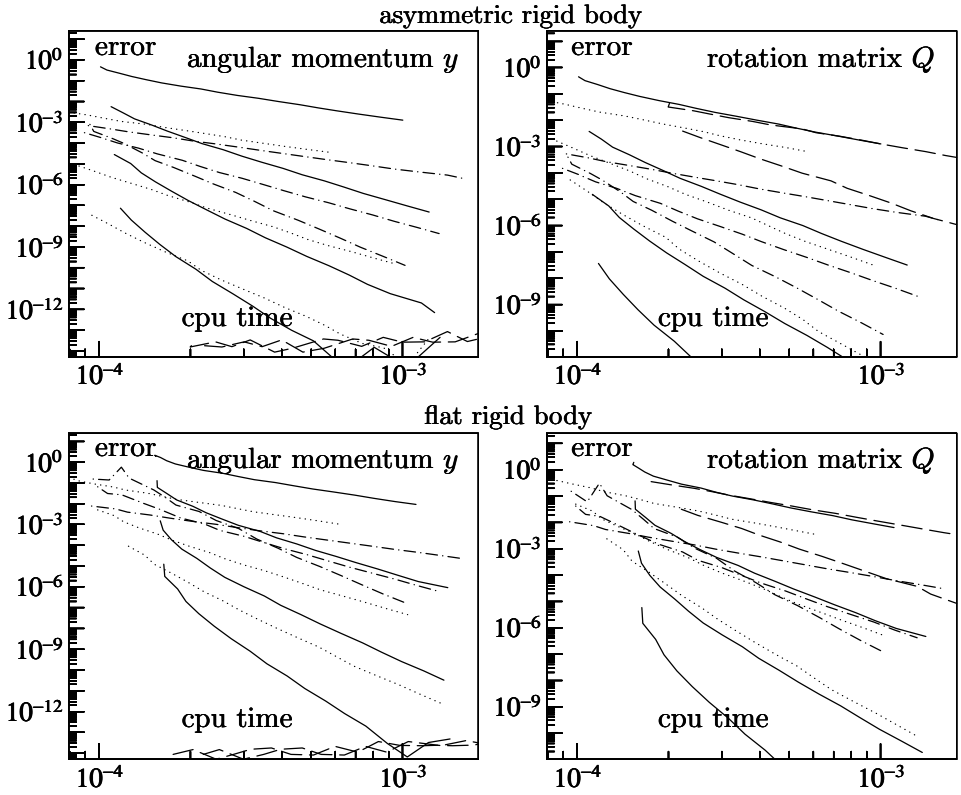


Figure 3.1: Work-precision diagrams for rigid body integrators: DMV, DMV4, DMV6, DMV8 (solid lines), IMR2, IMR4, IMR6 (dotted lines), JEM2, JEM4 (dashed lines), SR2, SR4, SR6 (dashed-dotted lines).

1.0, which corresponds to the water molecule as considered in [Fas03]. Initial values are $y(0) = (1.8,\, 0.4,\, -0.9)^T$ and $Q(0)$ is the identity matrix.

We have carefully implemented the above integrators in FORTRAN, using quaternions for the rotation matrices in all codes. Since there is no external potential, the invariants $H(y)$, $C(y)$ and the modified moments of inertia are constant along the numerical solution, so they need to be computed only once. However, to simulate the presence of an external potential, in our implementation we recalculate them in every step. All codes permit to include an external potential.

For each method and many different stepsizes, we plot in Figure 1 the global errors at the endpoint for the angular momentum (left pictures) and the quaternion representation of the rotation matrix (right pictures), as a function of the cpu times (on a SUN Blade 1500 work station). The execution times are taken as the average of 1000 experiments. For symmetric bodies similar results are obtained with the exception that the splitting method used yields the exact solution.

We observe that all methods show the correct order (lines of slope 2, 4, and 6). It is remarkable that the modifying (preprocessed) vector field integrators IMR2r and DMV2r significantly improve the accuracy with increasing order, even if we use very large stepsizes. This is due to the fact that the higher order versions have only very little overhead with respect to the basic methods. For example, one step of DMV8 costs only about 50% more cpu time than DMV.

Remark 3.2.1 In the situation of a small and costly external potential, one would like to use a large stepsize for DMV2r. If it is so large that the iterations for the nonlinear equations do not converge, one can replace the step of DMV2r by m steps with stepsize

h/m. It can be efficiently implemented, because the $\widetilde{I}_k(y_n)$ do not change within these m steps and need to be computed only once. As illustration, 10 steps of DMV8 is only about 4 times more expensive than one step.

3.3 Proof of the main theorem

The proof of Theorem 3.1.1 relies heavily on backward error analysis [HLW06, Chapter IX] and on the theory of modified differential equations presented in [CHV07b].

3.3.1 Backward error analysis for DMV

The DMV algorithm is equivalent to RATTLE [MZ05] which is a symplectic discretisation for constrained Hamiltonian systems. Backward error analysis allows one to interpret formally the numerical solution of this method as the exact solution of a modified constrained Hamiltonian system (Theorem IX.5.6 of [HLW06]). For the special case of the rigid body problem it follows from Sect. VII.5.5 of [HLW06] that the modified differential equation is of the form

$$\dot{y} = \widehat{y}\,\nabla H_h(y), \qquad \dot{Q} = Q\,\widehat{\nabla H_h(y)}. \tag{3.11}$$

where $H_h(y)$ is the modified Hamiltonian,

$$H_h(y) = H(y) + h^2 H_3(y) + h^4 H_5(y) + \dots\ ,$$

so that $y_n = y(nh)$ and $Q_n = Q(nh)$ in the sense of formal power series. It is in even powers of h because the numerical method is symmetric.

Lemma 3.3.1 *The numerical solution of the DMV algorithm is formally equal to the exact solution of (3.11) where the modified Hamiltonian $H_h(y)$ depends on y only via the conserved quantities $H(y)$ and $C(y)$,*

$$\begin{aligned} H_h(y) &= K(H(y), C(y)) \\ K(H, C) &= H + h^2 K_3(H, C) + h^4 K_5(H, C) + \dots\ . \end{aligned} \tag{3.12}$$

Proof. The Hamiltonian $H(y)$ is a first integral of (3.11), because it is exactly preserved by the DMV agorithm (see [HLW06, Sect. IX.5.1]), i.e., for all k

$$\nabla H(y)^T\, \widehat{y}\, \nabla H_k(y) = 0.$$

Since $\nabla H(y)$ and $\nabla C(y)$ are also orthogonal to $\widehat{y}\nabla H(y)$, the vector $\nabla H_k(y)$ lies in the span of the two vectors $\nabla H(y), \nabla C(y)$, as long as they are linearly independent (which is the case when y is not a stationary point of (3.1)).

We choose local coordinates $z = \chi(y)$, where $z_1 = H(y)$ and $z_2 = C(y)$, and we define $K_k(z)$ via $K_k(\chi(y)) = H_k(y)$. Since $\nabla H_k(y)$ is a linear combination of $\nabla H(y)$ and $\nabla C(y)$, the function $K_k(z)$ does not depend on the variable z_3. □

The scalar functions $K_j(H, C)$ for the modified Hamiltonian can be computed recursively as

$$\begin{aligned} K_3(H,C) &= \frac{\sigma_{-1}}{6} H^2 - \frac{\sigma_1}{6\delta} CH + \frac{1}{6\delta} C^2, \\ K_5(H,C) &= \frac{3\sigma_1 + 2\delta\sigma_{-2}}{20\delta} H^3 - \frac{7 + 3\tau_{1,1}}{20\delta} CH^2 \\ &\quad + \frac{\sigma_2 + 4\delta\sigma_{-1}}{20\delta^2} C^2 H - \frac{\sigma_1}{20\delta^2} C^3, \\ &\dots \end{aligned}$$

where the constants $\delta, \sigma_a, \tau_{b,c}$ are those of Table 3.1.

3.3.2 The modified moments of inertia

We shall show that the modified equation (3.11) is of the form (3.1) with modified moments if inertia.

Lemma 3.3.2 *The numerical solution of DMV applied to the rigid body problem (3.1) can be interpreted (formally) as the exact solution of a rigid body problem (3.8) with modified moments of inertia $\overline{I}_1, \overline{I}_2, \overline{I}_3$, given by*

$$\frac{1}{\overline{I}_j} = \frac{1}{I_j}\frac{\partial K}{\partial H}\big(H(y), C(y)\big) + \frac{\partial K}{\partial C}\big(H(y), C(y)\big), \quad j = 1, 2, 3, \tag{3.13}$$

where $K(H, C)$ is the function of Lemma 3.3.1.

Proof. The special form of the modified Hamiltonian $H_h(y)$ in (3.12) implies that

$$\nabla H_h(y) = \frac{\partial K}{\partial H}\big(H(y), C(y)\big)\,\nabla H(y) + \frac{\partial K}{\partial C}\big(H(y), C(y)\big)\,\nabla C(y) = \overline{I}^{-1} y,$$

where $\overline{I} = \mathrm{diag}\,(\overline{I}_1, \overline{I}_2, \overline{I}_3)$ with $\overline{I}_j$ from (3.13). □

Proof of Theorem 3.1.1. For fixed y, formula (3.13) defines a mapping

$$\Psi : (I_1, I_2, I_3) \longmapsto (\overline{I}_1, \overline{I}_2, \overline{I}_3)$$

which is $\mathcal{O}(h^2)$-close to the identity. Notice that in (3.13) the moments of inertia I_j also appear in $K(H, C)$ and in $H(y)$.

Letting $(\widetilde{I}_1, \widetilde{I}_2, \widetilde{I}_3) = \Psi^{-1}(I_1, I_2, I_3)$, it follows from Lemma 3.3.2 that the DMV algorithm applied with $\widetilde{I}_j(y_n)$ yields the exact solution of (3.1). This relation can be reformulated as

$$\frac{1}{\widetilde{I}_j} = \frac{1}{I_j} - h^2\left(\frac{1}{\widetilde{I}_j}\frac{\partial K_3}{\partial H}\big(\widetilde{H}(y), C(y)\big) + \frac{\partial K_3}{\partial C}\big(\widetilde{H}(y), C(y)\big)\right) - \dots\ ,$$

where $\widetilde{H}(y) = \frac{1}{2}\sum_j \widetilde{I}_j^{-1} y_j^2$. Formal fixed point iteration shows that the $\widetilde{I}_j$ are of the form (3.7). □

3.3.3 Backward error analysis for the preprocessed DMV

We study here symplecticity properties of the preprocessed DMV algorithm. This will be done with help of backward error analysis.

Theorem 3.3.3 *The numerical solution of the preprocessed DMV Algorithm 3.1.2 applied to (3.1) is (formally) the exact solution of*

$$\dot{y} = \widehat{y}\ \overline{I}(y)^{-1} y, \qquad \dot{Q} = Q\ \widehat{\overline{I}(y)^{-1} y}, \tag{3.14}$$

where $\overline{I}(y)$ is obtained from (3.13), with I_j replaced by $\widetilde{I}_j^{[2r]}(y)$ given by

$$\frac{1}{\widetilde{I}_j^{[2r]}} = \frac{1}{I_j}\Big(1 + h^2 s_3(y) + \dots + h^{2r-2} s_{2r-1}(y)\Big) + h^2 d_3(y) + \dots + h^{2r-2} d_{2r-1}(y)\ .$$

Furthermore, there exists a modified Hamiltonian

$$H_h^{[2r]}(y) = H(y) + h^{2r} H_{2r+1}^{[2r]}(y) + h^{2r+2} H_{2r+3}^{[2r]}(y) + \dots \ ,$$

such that the modified equation for the angular momentum y in (3.14) has the Poisson structure

$$\dot{y} = \widehat{y}\, \nabla H_h^{[2r]}(y). \tag{3.15}$$

Proof. The first statement is an immediate consequence of Lemma 3.3.2, where the I_j are replaced by $\widetilde{I}_j^{[2r]}$.

The fixed point argument in the proof of Theorem 3.1.1 implies that

$$\frac{1}{\overline{I}_j} = \frac{1}{I_j}\Big(1 + h^2\sigma_3\big(H(y), C(y)\big) + \dots\Big) + h^2\delta_3\big(H(y), C(y)\big) + \dots \ ,$$

for some scalar functions $\sigma_k(H, C), \delta_k(H, C)$, $k = 3, 5, \dots$ Since $\widehat{y}\, y = 0$, the modified equation (3.14) for y has the form

$$\dot{y} = \widehat{y}\, I^{-1} y \left(1 + h^2\sigma_3\big(H(y), C(y)\big) + \dots\right) = \widehat{y}\, \nabla H_h^{[2r]}(y), \tag{3.16}$$

where $H_h^{[2r]}(y) = K_h^{[2r]}\big(H(y), C(y)\big)$ and $K_h^{[2r]}(H, C)$ is chosen as an integral with respect to H of the scalar factor $1 + h^2\sigma_3(H, C) + \dots$. The derivative of $K_h^{[2r]}(H, C)$ with respect to C is not involved in (3.16), because $\widehat{y}\, \nabla C(y) = 0$. □

Theorem 3.3.3 implies that the preprocessed DMV Algorithm 3.1.2 is a Poisson integrator for all orders $2r$. However, in the modified equation (3.14) for the rotation matrix Q we cannot replace $\overline{I}(y)^{-1} y$ by $\nabla H_h^{[2r]}(y)$. This means that the preprocessed DMV Algorithm 3.1.2 is not symplectic for the complete system for $r > 1$.

3.4 Quaternion implementation of DMV

For an efficient implementation, it is a standard approach to use quaternions to represent orthogonal matrices (see [HLW06] in the context of RATTLE and splitting implementations). Let Y_n be the vector defined from ω_n^T through the Cayley transform mentioned in (3.10). The orthogonal matrix ω_n^T can then be represented by the quaternion ρ_n of norm 1 given by

$$\begin{aligned} \rho_n &= \frac{1}{\sqrt{\alpha_n}}\left(1 + \frac{h}{2}\Big(i\,\frac{Y_{n,1}}{I_1} + j\,\frac{Y_{n,2}}{I_2} + k\,\frac{Y_{n,3}}{I_3}\Big)\right), \\ \alpha_n &= 1 + \frac{h^2}{4}\Big(\frac{Y_{n,1}^2}{I_1^2} + \frac{Y_{n,2}^2}{I_2^2} + \frac{Y_{n,3}^2}{I_3^2}\Big). \end{aligned} \tag{3.17}$$

In a similar way, we represent the rotation matrix Q_n by a quaternion q_n. Some algebraic manipulations show that the DMV algorithm (3.4)-(3.5) reduces to the following computation, with a simple multiplication of quaternions for the update of the rotation matrix,

$$y_{n+1} = y_n + \alpha_n^{-1} h\, f(Y_n), \qquad q_{n+1} = q_n \cdot \rho_n, \tag{3.18}$$

where $f(y) = \widehat{y}\ I^{-1} y$, and α_n, ρ_n are defined in (3.17). Here, the internal stage Y_n can be computed from the implicit relation

$$Y_n = \alpha_n y_n + \frac{h}{2}\, f(Y_n). \tag{3.19}$$

A simple way for solving the nonlinear (quadratic) system (3.19) is by fixed-point iteration. To improve efficiency, one may calculate the vector $e_n = \frac{h}{2} I^{-1} Y_n$ instead of Y_n, so that the computation of α_n reduces to

$$\alpha_n = 1 + e_{n,1}^2 + e_{n,2}^2 + e_{n,3}^2.$$

Formulae (3.18) for y_{n+1} and q_{n+1} are explicit. Other approaches for the solution of (3.5) are discussed in [MZ05]. Suppressing the factor α_n in (3.18) and in (3.19), but not in the definition (3.17) of ρ_n, yields the implicit midpoint rule for problem (3.1) which is discussed in [CHV07b].

3.5 Reducing round off errors

Now that we have developed an efficient high-order geometric integrator for the equations of motion of a rigid body, it is natural to study the accumulation with time of round-off errors and explain how they can be reduced. We analyze the propagation of round-off errors, first for the Preprocessed Discrete Moser-Veselov algorithm, and then we compare with the algorithm based on Jacobi elliptic functions, as proposed in several recent publications [CS06, CFSZ08, vZS07a, vZS07b]. We focus on the conservation of the first integrals of the system, i.e. the energy $H(y)$ and the angular momentum relative to the fixed frame (Qy and $C(y)$), see (3.3). These quantites are exactly conserved (up to round-off errors) by the considered numerical integrators.

3.5.1 Probabilistic explanation of the error growth

The long-time behavior of round-off errors can be explained with probabilistic arguments like those developed in the classical book of Henrici [Hen62]. We assume that the error contribution over one step in the Hamiltonian $H(y)$ (and similarly for the other invariants) is a sequence of independent random variables

$$H(y_{n+1}) - H(y_n) = \varepsilon_n$$

with variance $\mathrm{Var}(\varepsilon_n)$ proportional to the square of the round-off unit *eps* of the computer.

Under the additional assumption that the mean of all ε_n is zero, the sum for N steps of the ε_n's is a random variable with mean zero and variance proportional to $N\,eps^2$. This shows that the error err_N in the Hamiltonian after N steps grows like (random walk, see Figure 3.2)

$$\mathrm{Var}(err_N)^{1/2} = \mathcal{O}(\sigma\, eps\, \sqrt{N})$$

for some constant σ (e.g. $\sigma \approx 0.11$ in right picture of Figure 3.3). It is often called Brouwer's law [Bro37] in celestial mechanics, see also [HLW06, Section VIII.5].

If the mean average of the ε_n's is different from zero, due to a deterministic error source, then the round-off errors accumulate linearly (see further in left picture of Figure 3.3).

Numerical experiment In figure 3.2, we consider the following initial condition with norm 1,

$$y_1(0) = 0.5, \quad y_2(0) = 0.2, \quad y_3(0) = \sqrt{1 - y_1(0)^2 - y_2(0)^2} \tag{3.20}$$

and integrate for 200 initial values randomly chosen close to this initial condition, on the interval of time $[0, 10^4]$ with stepsize $h = 0.01$ (one million steps). We consider a rigid body with moment of inertia $I_1 = 0.345, I_2 = 0.653, I_3 = 1.0$, which corresponds to the water

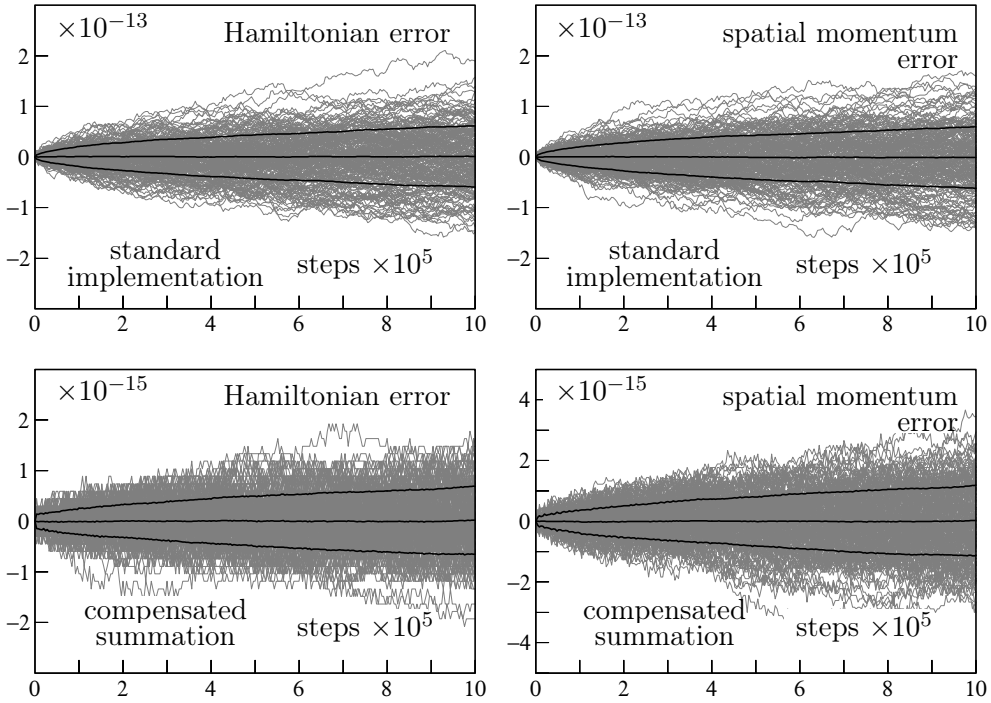


Figure 3.2: Preprocessed Discrete Moser-Veselov algorithm of order 10. Roundoff errors in Hamiltonian and spatial momentum (first component of Qy) for 200 initial values randomly chosen close to the one in (3.20). One million steps with stepsize $h = 0.01$. Top pictures: standard implementation. Bottom pictures: compensated summation. The average as a function of time and the standard deviation over all 1000 trajectories are included as bold curves.

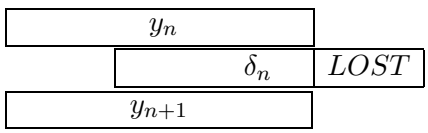
molecule, as considered in [Fas03]. The angular momentum $y(t)$ is a periodic function of time (in the absence of an external potential), and we integrate over about 822 periods. We also tried many different initial values and moments of inertia, and numerical results where similar to those presented here.

3.5.2 Compensated summation

For solving the Euler equations (3.1), the Preprocessed Discrete Moser-Veselov algorithm [HV06] requires the computation of a recursion of the form

$$y_{n+1} = y_n + \delta_n \tag{3.21}$$

where the increment δ_n has size $\mathcal{O}(h)$. In the case of a standard summation in floating point arithmetic, the last digits of the increment δ are lost at each step, because δ_n is smaller than y_n by a factor $\mathcal{O}(h)$:



However, it is possible to apply the so-called 'compensated summation' algorithm of [Kah65, Møl65] for reducing round-off errors. A famous analysis and presentation is given in [Hig93]. The idea of compensated summation is to store the lost digits $LOST$ at each

step, and add them to the next increment δ_{n+1}:

$$\begin{aligned} y_{n+1} &:= y_n + \delta_n \\ LOST &:= \delta_n + (y_n - y_{n+1}) \\ \delta_{n+1} &:= \delta_{n+1} + LOST \end{aligned}$$

Applying compensated summation allows to compute recursion 3.21 in high-precision and thus to reduce by a factor h the effect of round-off errors, as illustrated in bottom pictures in Figure 3.2. Notice that we do not lose information if we do not normalize to 1 the quaternions $q_n, n = 0, 1, 2, \ldots$ representing rotation matrices. This allows to apply compensated summation also for the attitude $Q(t)$ in (3.1) of the rigid body (see bottom right picture in Figure 3.2). To avoid the manipulation of large quaternion components, one can divide the quaternion q_n by 2 from time to time (this operation is performed exactly due to the binary representation in the computer). The round-off errors in the preservation of invariants $H(y), C(y)$ and Qy now grow like $\mathcal{O}(eps\, h\sqrt{N})$, or equivalently

$$\mathrm{Var}(err_N)^{1/2} = \mathcal{O}(eps\, h^{1/2} t^{1/2})$$

where $t = Nh$.

Here, the only constants involved are the modified moments of inertia $\widetilde{I}_1, \widetilde{I}_2, \widetilde{I}_3$, and we avoid using other constants in the numerical implementation. The DMV algorithm shows the correct probabilistic behavior (see Figure 3.2). As recommended in [HMR08], the fixed point iteration is performed until convergence: the stopping criterion is $\Delta^{(k)} = 0$ or $\Delta^{(k)} > \Delta^{(k-1)}$ which indicates that the increments $\Delta^{(k)}$ of the iteration starts to oscillate due to round-off.

3.5.3 Algorithm based on Jacobi elliptic functions: study of round-off errors

In the historical article [Jac50], Jacobi derived the analytic solution for the motion of a free rigid body and defined to this aim the so-called 'Jacobi analytic functions' as

$$\mathrm{sn}(u,k) = \sin(\varphi), \quad \mathrm{cn}(u,k) = \cos(\varphi), \quad \mathrm{dn}(u,k) = \sqrt{1 - k^2 \sin^2(\varphi)}, \tag{3.22}$$

where the Jacobi amplitude $\varphi = \mathrm{am}(u,k)$ is defined implicitly by an elliptic integral of the first kind.

In several recent publications [CS06, CFSZ08, vZS07a, vZS07b], it is proposed to integrate the equations of motion of the free rigid body analytically, using the Jacobi elliptic functions. Although this approach yields the exact solution, a standard implementation yields an unexpected linear propagation (accumulation) of round-off errors (see Figure 3.3 and [CFSZ08, Fig. 3.1]). We explain how round-off errors can be reduced, to achieve Brouwer's law (Sect. 3.5.1).

3.5.3.1 Standard implementation

We consider here the numerical algorithm based on Jacobi elliptic functions as proposed in [CS06, CFSZ08, vZS07a, vZS07b], and we focus on the numerical resolution of the Euler equations (left equation in (3.1)), see e.g. Proposition 2.1 in [CFSZ08].

Algorithm 3.5.1 Assume $I_1 \leq I_2 \leq I_3$ and $y(t_0)$ is not a saddle point. Consider the quantities

$$a_1 = \sqrt{2H(y)I_3 - 2C(y)} \qquad a_3 = \sqrt{2C(y) - 2H(y)I_1},$$

which are conserved along time. To simulate the presence of an external potential, they are recalculated before each step. For $(I_2 - I_1)a_1^2 \leq (I_3 - I_2)a_3^2$, the solution of the Euler equations at time $t = t_0 + h$ is

$$y_1(t) = b_1 a_1 \mathrm{cn}(u,k), \quad y_2(t) = b_2 a_1 \mathrm{sn}(u,k), \quad y_3(t) = b_3 a_3 \delta \mathrm{dn}(u,k),$$

where $\delta = \mathrm{sign}(y_3) = \pm 1$ and

$$b_1 = \sqrt{I_1/(I_3 - I_1)}, \quad b_2 = \sqrt{I_2/(I_3 - I_2)}, \quad b_3 = \sqrt{I_3/(I_3 - I_1)}.$$

Here, $\mathrm{cn}(u,k)$, $\mathrm{sn}(u,k)$ and $\mathrm{dn}(u,k)$ are the Jacobi elliptic functions (3.22) with modulus k and parameter u,

$$k^2 = b_0 a_1^2/a_3^2, \quad b_0 = (I_2 - I_1)/(I_3 - I_2), \quad u = h\delta\sqrt{(I_3 - I_2)/(I_1 I_2 I_3)}a_3 + \nu,$$

where ν is a constant of integration (see [CS06, Sect. 3] for details). Similar formulas hold for $(I_2 - I_1)a_1^2 \geq (I_3 - I_2)a_3^2$.

Notice that round-off errors in the computation of u and φ for the Jacobi elliptic functions (3.22) have no influence on the preservation of first integrals, because it can be interpreted as a time transformation.

Numerical experiment The algorithm based on Jacobi elliptic functions is fully explicit and no iterative solution of non-linear equations is involved (excepted the code for computing Jacobi elliptic functions). Nevertheless, the standard implementation (Algorithm 3.5.1) shows a linear growth of round-off errors (see left picture in Figure 3.3). The error for the Hamiltonian is about 1.25×10^{-17} per step, or $0.056 \times eps$ per step, where $eps = 2^{-52}$ is the machine precision. The error is a superposition of a small statistical error and a deterministic error which grows linearly with time, due to a tiny non-zero bias in the pattern of positive and negative round-off errors.

In [HMR08], it is shown that for implicit Runge-Kutta methods, the use of rounded coefficients a_{ij} and b_j induces a systematic error in long-time integrations. Here, the situation is similar, because there are many constants involved: $b_0, b_1, b_2, b_3, I_1, I_3, \ldots$ The same rounded coefficients are used along the numerical integration, and this induces a deterministic error which propagates linearly with time.

To reduce round-off errors in the Jacobi elliptic functions based algorithm, our first idea was to compute all above constants in quadruple-precision arithmetic, and then make all corresponding multiplications in quadruple-precision. Alternatively, we explain in the next section how round-off errors can be reduced using only standard double-precision arithmetic.

3.5.3.2 New implementation

We present here a modification of Algorithm 3.5.1 which makes round-off behave like a random walk (see right picture of Figure 3.3). The idea is to reduce the number of constants involved, so that, in the spirit of backward error analysis, all constants can be interpreted as exact values corresponding to modified moments of inertia. We show that this can be achieved with Algorithm 3.5.2 which uses only two independent constants c_1 and λ defined below.

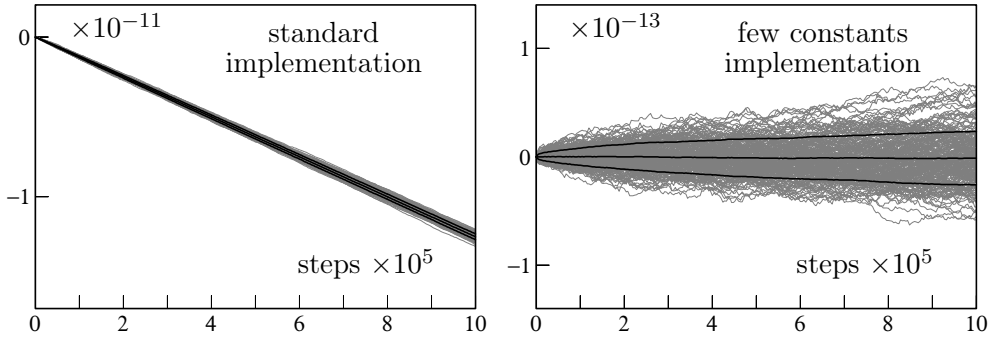


Figure 3.3: Hamiltonian errors for the integrators based on Jacobi elliptic functions. One million steps with stepsize $h = 0.01$. Left picture: standard implementation (Algorithm 3.5.1). Right picture: new implementation (Algorithm 3.5.2). The plots show the error as a function of time for 200 initial values (with norm 1) randomly chosen close to the one in (3.20). The mean as a function of time and the standard deviation over all 1000 trajectories are included as bold curves.

Algorithm 3.5.2 Consider the constants (we still assume $I_1 \leq I_2 \leq I_3$),

$$c_1 = \frac{I_1(I_3 - I_2)}{I_2(I_3 - I_1)}, \qquad c_2 = 1 - c_1, \tag{3.23}$$

and the quantities

$$d_1 = \sqrt{y_1^2 + c_1 y_2^2}, \qquad d_3 = \sqrt{c_2 y_2^2 + y_3^2},$$

(recalculated before each step). For $c_2 d_1^2 \leq c_1 d_3^2$, the solution of the Euler equations at time $t = t_0 + h$ is

$$y_1(t) = d_1 \mathrm{cn}(u,k), \quad y_2(t) = d_2 \mathrm{sn}(u,k), \quad y_3(t) = \delta\sqrt{d_3^2 - c_2 y_2(t)^2},$$

where $d_2 = \sqrt{y_1^2/c_1 + y_2^2}$.Here, $\mathrm{cn}(u,k)$, $\mathrm{sn}(u,k)$ are the Jacobi elliptic functions (3.22) with

$$k^2 = (c_2 d_1^2)/(c_1 d_3^2), \quad u = \delta h \lambda d_3 + \nu, \quad \lambda = \sqrt{(I_3 - I_2)(I_3 - I_1)/(I_1 I_2 I_3^2)},$$

$\delta = \mathrm{sign}(y_3) = \pm 1$, and ν is a constant of integration. We have similar formulas for $c_2 d_1^2 \geq c_1 d_3^2$.

It is essential in (3.23) that the identity $c_1 + c_2 = 1$ holds exactly. This can be done as follows:

compute c_1;
$c_2 = 1 - c_1$;
$c_1 = 1 - c_2$;

It makes c_1 and c_2 have a floating point arithmetic representation with the same exponent.

3.6 Accurate computation of the tangent map

We consider here the computation of the derivatives of the flow of the free rigid body equations with respect to initial conditions. Notice that $y(t)$ and $Q(0)^T Q(t)$ are independent of the initial rotation matrix $Q(0)$ (this is also true for the numerical solution). Thus, the only interesting part is the computation of the derivatives $\partial y/\partial y_0$ and $\partial Q/\partial y_0$ with respect to the initial angular momentum $y_0 = y(0)$. The idea is to formally differentiate the discretization given by the high-order DMV10 algorithm.

3.6.1 Motivation: conjugate points

An accurate computation of the tangent map allows to calculate the conjugate locus for the free rigid body. In [BF07], the conjugate locus is studied in the case where two moments of inertia are equal (e.g. $I_2 = I_3$), and the general case is currently investigated in [BF07]. We briefly explain first what is the issue.

Consider a point q_0 on a Riemannian manifold $\mathcal{Q}$, and an oriented geodesic γ parametrized by $q(t)$ and passing through q_0 ($q(0) = q_0$) with initial velocity $\dot{q}_0 \in T_{q_0}\mathcal{Q}$. On one hand, one can study the global optimality of the geodesic: the first point of γ at which γ ceases to minimize the distance from q (if it exists) is called the (first) *cut point*. On the other hand, on can study the local optimality of γ by introducing conjugate points, at which the minimization property is lost locally, i.e. among infinitesimally close geodesics. Precisely, if t_c is the minimal time such that the Jacobian matrix

$$\frac{\partial q(t_c)}{\partial \dot{q}_0}$$

is singular, then $q_c = q(t_c)$ is called the (first) conjugate point of q along the geodesic. The cut locus (resp. conjugate locus) of q_0 is the union of all cut points (resp. conjugate points) along all geodesics through q_0.

There is a strong connection between conjugate points and cut points: if q_c is the first conjugate point of q along γ, then either q_c is the cut point, or there is a cut point q' between q and q_c. Only the cut locus has a clear geometric significance, but it is more convenient to study the conjugate locus. Also it gives information on the cut locus.

In the case of the free rigid body equations, the flow is a geodesic flow on the manifold $SO(3)$ with a left-invariant metric induced by the scalar product

$$\langle y, y' \rangle = \frac{y_1 y_1'}{I_1} + \frac{y_2 y_2'}{I_2} + \frac{y_3 y_3'}{I_3},$$

see [BF07] for details. Consider the solution $y(t)$, $Q(t)$ of the free rigid body equations with initial angular momentum $y(0) = y_0 \in \mathbb{R}^3$ (we take $Q(0) = Id$). The (first) conjugate time is then defined as

$$t_c = \inf\{t > 0\,;\, \operatorname{rank} \frac{\partial Q(t)}{\partial y_0} < 3\},$$

and $Q(t_c)$ is called the (first) conjugate point in the direction y_0 (if it exists).

3.6.2 Representation of the tangent map

A first approach could be to differentiate all four components $q_0(t)$, $q_1(t)$, $q_2(t)$, $q_3(t)$ of the quaternion representing the rotation matrix $Q(t)$ with respect to the initial angular momentum $y_0 = (y_{0,1}, y_{0,2}, y_{0,3})^T$. This produces a 4×3 rectangular matrix $\left(\frac{\partial q_i(t)}{\partial y_{0,j}}\right)_{i=0\ldots3, j=1\ldots3}$. Then, conjugate points are obtained when this matrix has rank < 3, which can be detected when all three 3×3 submatrices obtained by deleting one line have determinant zero, or by considering the singular value decomposition (SVD). However, this is not very convenient from a numerical point of view.

Alternatively, we propose the following approach, where only a 3×3 square matrix A_n is involved to represent the derivative $(\partial Q_n)/(\partial y_0)$ of Q_n with respect to the initial angular momentum y_0.

Proposition 3.6.1 *The derivatives of the attitude matrix can be represented as*

$$\frac{\partial Q_n}{\partial y_{0,j}} = Q_n \widehat{a}_{n,j}, \quad j = 1, 2, 3$$

where the $\widehat{a}_{n,j}$'s are skew-symmetric matrices. Thus, conjugate points are obtained when the 3×3 matrix

$$A_n = (a_{n,1},\, a_{n,2},\, a_{n,3}),$$

whose columns are the $a_{n,j}$'s becomes singular.

Moreover, the matrix A_n can be computed recursively as $A_0 = 0$ and

$$A_{n+1} = \alpha_n^{-1}(Id - \widehat{e}_n)(\alpha_n(Id + \widehat{e}_n)^{-1}A_n + 2\,\frac{\partial e_n}{\partial y_0}). \tag{3.24}$$

Here, vector $e_n = (e_{n,1}, e_{n,2}, e_{n,3})^T$ and $\alpha_n = 1 + e_{n,1}^2 + e_{n,2}^2 + e_{n,3}^2$ are defined via the Cayley transform

$$V_n = \frac{Id + \widehat{e}_n}{Id - \widehat{e}_n}$$

where $V_n = Q_n^T Q_{n+1}$ is an orthogonal matrix.

Notice that the coefficients of the matrix $\alpha_n(Id+\widehat{e}_n)^{-1}$ are in fact polynomials in the $e_{n,j}$'s (so there is no need to compute any matrix inverse).

Proof. Let dQ denote a derivative with respect to a scalar parameter (in our situation, the $y_{0,j}$'s). Differentiating the orthogonality condition $Q_n^T Q_n = Id$ yields $dQ_n^T Q_n + Q_n^T dQ_n = 0$, which means that $\widehat{a}_n := Q_n^T dQ_n$ is a skew-symmetric matrix. The update for the rotation matrix Q_n can be written as

$$Q_{n+1} = Q_n V_n$$

Then, differentiating this relation,

$$dQ_{n+1} = dQ_n V_n + Q_n dV_n.$$

Using $dQ_n = Q_n \widehat{a}_n$ and $Q_{n+1}^T Q_n = V_n^T$ yields

$$\widehat{a}_{n+1} = V_n^T \widehat{a}_n V_n + V_n^T dV_n.$$

Then, computing the derivative of the Cayley transform,

$$dV_n = 2(Id - \widehat{e}_n)^{-1} d\widehat{e}_n (Id - \widehat{e}_n)^{-1},$$

and substituting,

$$\widehat{a}_{n+1} = \frac{Id - \widehat{e}_n}{Id + \widehat{e}_n}\widehat{a}_n \frac{Id + \widehat{e}_n}{Id - \widehat{e}_n} + 2(Id + \widehat{e}_n)^{-1} d\widehat{e}_n (Id - \widehat{e}_n)^{-1}.$$

This relation between skew-symmetric matrices can be written out using vectors,

$$a_{n+1} = (Id - \widehat{e}_n)(Id + \widehat{e}_n)^{-1} a_n + 2\alpha_n^{-1}(Id - \widehat{e}_n) de_n,$$

and one can factorize the term $\alpha_n^{-1}(Id - \widehat{e}_n)$ to gain one matrix multiplication. □

3.6.3 Numerical implementation

It remains to compute the 3×3 Jacobian matrices $\frac{\partial e_n}{\partial y_0}$ appearing in Proposition 3.6.1. We explain how this can be performed efficiently, by differentiating the discretization given by the DMV10 algorithm. The algorithm involves three modified moments of inertia $\widetilde{I}_j$ defined as

$$\frac{1}{\widetilde{I}_j} = \frac{s}{I_j} + d$$

where the scalars $s = s(y)$ and $d = s(y)$ are given in Theorem 3.1.1. As detailed in Sect. 3.4, the internal stage in the algorithm can be implemented as

$$e_{n,1} = \frac{\alpha_n h}{2\widetilde{I}_1} y_{n,1} + \frac{\widetilde{I}_2 - \widetilde{I}_3}{\widetilde{I}_1} e_{n,2} e_{n,3}$$

and similarly for the other components. The idea is to introduce new variables

$$z_{n,j} := s\widetilde{I}_j e_{n,j},\ j = 1, 2, 3,$$

so that the nonlinear system reduces to

$$z_{n,1} = b y_{n,1} + \left(\frac{1}{I_3} - \frac{1}{I_2} \right) z_{n,2} z_{n,3}$$

with non-modified moments of inertia I_1, I_2, I_3 in the non-linearity. This makes the derivatives of z_n with respect to initial conditions simpler to compute than those of e_n. Here, the scalar

$$b := \alpha_n s \frac{h}{2} \tag{3.25}$$

is the same for every component. Similarly, the update for y becomes

$$y_{n+1,1} = y_{n,1} + \frac{2}{b} \left(\frac{1}{I_3} - \frac{1}{I_2} \right) z_{n,2} z_{n,3}$$

(and similarly for the other components).

Next, when computing the derivatives of $b, \alpha_n, e_n, \ldots$, one has to compute derivatives of the terms $1/(s\widetilde{I}_j),\ j = 1, 2, 3$. The idea is to use the identity

$$\frac{1}{s\widetilde{I}_j} = \frac{1}{I_j} + \frac{d}{s},$$

which implies that these derivatives are independent of j, and depend only on the derivatives of d/s.

We are now in position to state the modification of the DMV10 algorithm to compute the tangent map together with the equations of motion of the free rigid body.

Algorithm 3.6.2 Numerical computation of the tangent map using DMV10.

The derivatives of the angular momentum $y_n \approx y(t_n)$ and the attitude matrix $Q_n \approx Q(t_n)$ with respect to the initial angular momentum $y_0 = y(0)$ can be computed recursively at each step of DMV10 as detailed below.

1. Compute the derivatives of the scalar functions $s = s(H, C)$ and $d = d(H, C)$ with respect to H and C, and then the following scalars:

$$s^{[j]} := \frac{1}{I_j} \frac{\partial s}{\partial H} + \frac{\partial s}{\partial C}, \qquad (d/s)^{[j]} := \frac{1}{I_j} \frac{\partial (d/s)}{\partial H} + \frac{\partial (d/s)}{\partial C}, \quad j = 1, 2, 3.$$

   In the absence of an external potential, this computation can be performed only one.

2. Compute the 3×3 Jacobian matrix $\frac{\partial z_n}{\partial y_0}$ by solving the following linear system

$$A\frac{\partial z_n}{\partial y_0} = B$$

where A and B are 3×3 matrices,

$$A = \begin{pmatrix} 1 & -(\frac{1}{I_3}-\frac{1}{I_2})z_{n,3} & -(\frac{1}{I_3}-\frac{1}{I_2})z_{n,2} \\ -(\frac{1}{I_1}-\frac{1}{I_3})z_{n,3} & 1 & -(\frac{1}{I_1}-\frac{1}{I_3})z_{n,1} \\ -(\frac{1}{I_2}-\frac{1}{I_1})z_{n,2} & -(\frac{1}{I_2}-\frac{1}{I_1})z_{n,1} & 1 \end{pmatrix} - \left(\frac{h}{\widetilde{I}_j}e_{n,j}y_{n,i}\right)_{i,j=1,2,3}$$

$$B = \left(b\frac{\partial y_{n,i}}{\partial y_{0,j}} + h\sum_{k=1}^{3}\Big(s^{[k]}\frac{\alpha_n}{2} + (d/s)^{[k]}(\frac{z_{n,1}^2}{\widetilde{I}_1}+\frac{z_{n,2}^2}{\widetilde{I}_2}+\frac{z_{n,3}^2}{\widetilde{I}_3})\Big)\frac{\partial y_{n,k}}{\partial y_{0,j}}y_{n,i}y_{n,j}\right)_{i,j=1,2,3}$$

and b is defined in (3.25). This linear system can be solved with the Gaussian elimination without pivot search because the matrix A is close to identity.

3. Compute the derivative $\frac{\partial e_n}{\partial y_0}$:

$$\frac{\partial e_{n,i}}{\partial y_{0,j}} = \Big(\frac{1}{I_i}+\frac{d}{s}\Big)\frac{\partial z_{n,i}}{\partial y_{0,j}} + z_{n,i}\sum_{k=1}^{3}(d/s)^{[k]}y_{n,k}\frac{\partial y_{n,k}}{\partial y_{0,j}}, \quad i,j=1,2,3.$$

4. Finally update $\frac{\partial y_{n+1}}{\partial y_0}$:

$$\frac{\partial y_{n+1}}{\partial y_{0,j}} = \frac{\partial y_n}{\partial y_{0,j}} + \frac{2}{b}\begin{pmatrix} (\frac{1}{I_3}-\frac{1}{I_2})\Big(\frac{\partial z_{n,2}}{\partial y_{0,j}}z_{n,3} + z_{n,2}\frac{\partial z_{n,3}}{\partial y_{0,j}} - \frac{1}{b}\frac{\partial b}{\partial y_{0,j}}z_{n,2}z_{n,3}\Big) \\ (\frac{1}{I_1}-\frac{1}{I_3})\Big(\frac{\partial z_{n,3}}{\partial y_{0,j}}z_{n,1} + z_{n,3}\frac{\partial z_{n,1}}{\partial y_{0,j}} - \frac{1}{b}\frac{\partial b}{\partial y_{0,j}}z_{n,3}z_{n,1}\Big) \\ (\frac{1}{I_2}-\frac{1}{I_1})\Big(\frac{\partial z_{n,1}}{\partial y_{0,j}}z_{n,2} + z_{n,1}\frac{\partial z_{n,2}}{\partial y_{0,j}} - \frac{1}{b}\frac{\partial b}{\partial y_{0,j}}z_{n,1}z_{n,2}\Big) \end{pmatrix}$$

where

$$\frac{1}{b}\frac{\partial b}{\partial y_{0,j}} = \frac{1}{s}\sum_{k=1}^{3}s^{[k]}y_{n,k}\frac{\partial y_{n,k}}{\partial y_{0,j}} + \frac{2}{\alpha_n}\sum_{k=1}^{3}\frac{\partial e_{n,k}}{\partial y_{0,j}}e_{n,k} \quad j=1,2,3.$$

5. The derivatives $\frac{\partial Q_{n+1}}{\partial y_{0,j}} = Q_{n+1}\widehat{a}_{n+1,j},\ j=1,2,3$, can be computed recursively using (3.24) in Proposition 3.6.1 below.

Numerical experiment. Our computations have shown that the accurate computation of the solution together with the tangent map costs about 4 times more cpu time compared to the computation of the solution alone.

Chapter 4

The role of symplectic integrators in optimal control

Note: This chapter is identical to the article [CHV08] in collaboration with M. Chyba and E. Hairer, with the exception of the additional Section 4.6 on backward error analysis for optimal control problems.

For the numerical solution of optimal control problems there are essentially two approaches: the direct approach which consists in discretizing the problem directly and applying optimization techniques, and the so-called indirect approach which is based on Pontryagin's maximum principle. This gives necessary conditions that reduce the optimal control problem to a system of Hamiltonian differential equations with boundary conditions, which can be solved by shooting techniques. The present article is concerned with the indirect approach.

There are many arguments in favour of using symplectic integrators for the numerical solution of the Hamiltonian system. Firstly, geometric numerical integration [HLW06] puts forward the use of structure-preserving algorithms for the solution of structured problems like Hamiltonian systems. This is justified by a backward error analysis which allows one to interpret the numerical solution of a symplectic method as the exact flow of a modified Hamiltonian system. This explains the excellent long-time behavior of such integrators. Furthermore, a series of papers [WM97, MW01, BCMR02, GB04] develops variational integrators for optimal control problems and emphasizes their symplecticity. The work of [Hag00, BLV06] shows that, for partitioned Runge–Kutta discretizations based on symplectic pairs, the direct and indirect approaches are equivalent.

On the other hand, the Hamiltonian systems arising in optimal control are quite different from those in astronomy and molecular dynamics, where symplectic integrators have proven to be the method of choice. Pontryagin's maximum principle yields a boundary value problem and long-time integration is in general not an issue. Furthermore, the modified Hamiltonian system (in the sense of backward error analysis) is not necessarily a differential equation that arises from a modified optimal control problem. It is therefore not obvious whether symplectic integrators will be superior to standard methods. The aim of this article is to study this question and to investigate the practical effect of using symplectic integrators in the numerical solution of optimal control problems. This will be done at several case studies.

For problems in low dimension which are close to a critical value of the Hamiltonian, symplectic integrators turn out to have a significant advantage. This happens for the so-called Martinet case in sub-Riemannian geometry, see [ABCK97, BCK99] and the ref-

erences therein. The Martinet flat case and a non integrable perturbation are introduced in Sect. 4.1 together with the corresponding differential equations. Numerical experiments with an explicit Runge–Kutta method and with the symplectic Störmer–Verlet method are presented in Sect. 4.2 and illustrated with figures. Close to abnormal geodesics, the results are quite spectacular. For a relatively large stepsize, the symplectic integrator provides a solution with the correct qualitative behavior and a satisfactory accuracy, while for the same stepsize the non-symplectic integrator gives a completely wrong numerical solution, particularly for the non integrable case. The explanation relies on the theory of backward error analysis (Sect. 4.3).

Unfortunately, this explanation cannot be extended to general optimal control problems. We present two examples: the orbital transfer of a spacecraft (Sect. 4.4) and the control of a submerged rigid body (Sect. 4.5). The Hamiltonian system for the submerged rigid body is very sentitive when considered as an initial value problem and thus requires the use of multiple shooting for solving the boundary value problem. For both problems, symplectic integrators do not show any real advantage. The reason is that the time interval is not long enough so that the symplectic integrator could benefit from structure preservation.

4.1 A Martinet type sub-Riemannian structure

Let (U, Δ, g) be a sub-Riemannian structure where U is an open neighborhood of R^3, Δ a distribution of constant rank 2 and g a Riemannian metric. When Δ is a contact distribution, there are no abnormal geodesics, and a non-symplectic integrator is as efficient as a symplectic one. However, when the distribution is taken as the kernel of the Martinet one-form, we show that a symplectic integrator is much more efficient for the computation of the normal geodesics and their conjugate points near the abnormal directions.

We briefly recall some results of [ABCK97] for a sub-Riemannian structure (U, Δ, g). Here, U is an open neighborhood of the origin in R^3 with coordinates $q = (x, y, z)$, and g is a Riemannian metric for which a graduated normal form, at order 0, is $g = (1 + \alpha y)dx^2 + (1 + \beta x + \gamma y)dy^2$. The distribution Δ is generated by the two vector fields $F_1 = \frac{\partial}{\partial x} + \frac{y^2}{2}\frac{\partial}{\partial z}$ and $F_2 = \frac{\partial}{\partial y}$ which correspond to $\Delta = \ker \omega$ where $\omega = dz - \frac{y^2}{2}dx$ is the Martinet canonical one-form. To this distribution we associate the affine control system

$$\dot{q} = u_1(t)F_1(q) + u_2(t)F_2(q),$$

where $u_1(t), u_2(t)$ are measurable bounded functions which act as controls.

We consider two cases, the Martinet flat case $g = dx^2 + dy^2$, an integrable situation, and a one parameter perturbation $g = dx^2 + (1 + \beta x)^2 dy^2$ for which the set of geodesics is non integrable.

4.1.1 Geodesics

It follows from the Pontryagin maximum principle, see [ABCK97, BCK99], that the normal geodesics corresponding to $g = dx^2 + (1 + \beta x)^2 dy^2$ are solutions of an Hamiltonian system

$$\dot{q} = \frac{\partial H}{\partial p}(q, p), \qquad \dot{p} = -\frac{\partial H}{\partial q}(q, p), \tag{4.1}$$

where $q = (x, y, z)$ is the state, $p = (p_x, p_y, p_z)$ is the adjoint state, and the Hamiltonian is

$$H(q, p) = \frac{1}{2}\left(\left(p_x + p_z\frac{y^2}{2}\right)^2 + \frac{p_y^2}{(1 + \beta x)^2}\right).$$

In other words, the normal geodesics are solutions of the following equations:

$$\begin{aligned} \dot{x} &= p_x + p_z\frac{y^2}{2} & \dot{p}_x &= \frac{\beta\, p_y^2}{(1+\beta x)^3} \\ \dot{y} &= \frac{p_y}{(1+\beta x)^2} & \dot{p}_y &= -\Big(p_x + p_z\frac{y^2}{2}\Big)p_z y \\ \dot{z} &= \Big(p_x + p_z\frac{y^2}{2}\Big)\frac{y^2}{2} & \dot{p}_z &= 0. \end{aligned} \tag{4.2}$$

Notice that the variables z and p_z do not influence the other equations (except via the initial value $p_z(0)$), so that we are actually confronted with a Hamiltonian system in dimension four. For the Martinet flat case ($\beta = 0$), the interesting dynamics takes place in the two-dimensional space of coordinates (y, p_y). The Hamiltonian is

$$H(y, p_y) = \frac{p_y^2}{2} + \frac{1}{2}\Big(p_x + p_z\frac{y^2}{2}\Big)^2,$$

where p_x and p_z have to be considered as constants. This is a one-degree of freedom mechanical system with a quartic potential. For $p_x < 0 < p_z$, the Hamiltonian $H(y, p_y)$ has two local minima at $(y = \pm\sqrt{-2p_x/p_z},\ p_y = 0)$, which correspond to stationary points of the vector field. In this case, the origin $(y = 0,\ p_y = 0)$ is a saddle point.

Whereas normal geodesics correspond to oscillating motion, it is shown in [ABCK97, BCK99] that the abnormal geodesics are the lines $z = z_0$ contained in the plane $y = 0$. For the considered metrics, the abnormal geodesics can be obtained as projections of normal geodesics, we say that they are not strictly abnormal. In [BCK99], the authors introduce a geometric framework to analyze the singularities of the sphere in the abnormal direction when $\beta \neq 0$. See also [BLT99, BT01] for a precise description of the role of the abnormal geodesics in sub-Riemannian geometry in the general non-integrable case, i.e., when the abnormal geodesics can be strict. The major result of these papers is the proof that the sub-Riemannian sphere is not sub-analytic because of the abnormal geodesics.

Interesting phenomena arise when the normal geodesics are close to the separatrices connecting the saddle point. Therefore, we shall consider in Sect. 4.2 the computation of normal geodesics with $y(0) = 0$ and $p_y(0) > 0$ but small.

4.1.2 Conjugate points

For the Hamiltonian system (4.1) we consider the exponential mapping

$$\exp_{q_0,t} : p_0 \longmapsto q(t, q_0, p_0)$$

which, for fixed $q_0 \in R^3$, is the projection $q(t, q_0, p_0)$ onto the state space of the solution of (4.1) starting at $t = 0$ from (q_0, p_0). Following the definition in [ABCK97] we say that the point q_1 is conjugate to q_0 along $q(t)$ if there exists (p_0, t_1), $t_1 > 0$, such that $q(t) = \exp_{q_0,t}(p_0)$ with $q_1 = \exp_{q_0,t_1}(p_0)$, and the mapping $\exp_{q_0,t_1}$ is not an immersion at p_0. We say that q_1 is the first conjugate point if t_1 is minimal. First conjugate points play a major role when studying optimal control problems since it is a well known result that a geodesic is not optimal beyond the first conjugate point.

For the numerical computation of the first conjugate point, we compute the solution of the Hamiltonian system (4.1) together with its variational equation,

$$\dot{y} = J^{-1}\nabla H(y), \qquad \dot{\Psi} = J^{-1}\nabla^2 H(y)\Psi. \tag{4.3}$$

Here, $y = (q, p)$ and J is the canonical matrix for Hamiltonian systems. It can be shown that for Runge-Kutta methods, the derivative of the numerical solution with respect to the initial value, $\Psi_n = \partial y_n / \partial y_0$, is the result of the same numerical integrator applied to the augmented system (4.3), see [HLW06, Lemma VI.4.1]. Here, the matrix

$$\Psi = \begin{pmatrix} \partial q/\partial q_0 & \partial q/\partial p_0 \\ \partial p/\partial q_0 & \partial p/\partial p_0 \end{pmatrix}$$

has dimension 6×6. The conjugate points are obtained when $\partial q/\partial p_0$ becomes singular, i.e., $\det(\partial q/\partial p_0) = 0$.

4.2 Comparison of symplectic and non-symplectic integrators

For the numerical integration of the Hamiltonian system (4.1), where we rewrite $\frac{\partial H}{\partial q}(q,p) = H_q(q,p)$ and $\frac{\partial H}{\partial p}(q,p) = H_p(q,p)$, we consider two integrators of the same order 2:

- a non-symplectic, explicit Runge–Kutta discretization, denoted RK2 (see [HLW06, Sect. II.1.1]),

$$\begin{aligned} q_{n+1/2} &= q_n + \frac{h}{2} H_p(q_n, p_n) & \qquad p_{n+1/2} &= p_n - \frac{h}{2} H_q(q_n, p_n) \\ q_{n+1} &= q_n + h H_p(q_{n+1/2}, p_{n+1/2}) & \qquad p_{n+1} &= p_n - h H_q(q_{n+1/2}, p_{n+1/2}) \end{aligned} \tag{4.4}$$

- the symplectic Störmer–Verlet scheme (see e.g. [HLW06, Sect. VI.3]),

$$\begin{aligned} p_{n+1/2} &= p_n - \frac{h}{2} H_q(q_n, p_{n+1/2}) \\ q_{n+1} &= q_n + \frac{h}{2}\Big(H_p(q_n, p_{n+1/2}) + H_p(q_{n+1}, p_{n+1/2}) \Big) \\ p_{n+1} &= p_{n+1/2} - \frac{h}{2} H_q(q_{n+1}, p_{n+1/2}) \end{aligned} \tag{4.5}$$

where $q_n = (x_n, y_n, z_n)$ and $p_n = (p_{x,n}, p_{y,n}, p_{z,n})$. Here, $q_n \approx q(nh)$, $p_n \approx p(nh)$ and h is the stepsize.

For the computation of the conjugate points, we apply the numerical methods to the variational equation (4.3). Notice that only the partial derivatives with respect to p_0 have to be computed. Conjugate points are then detected when $\det(\partial q_n/\partial p_0)$ changes sign. We approximate them by linear interpolation which introduces an error of size $\mathcal{O}(h^2)$. This is comparable to the accuracy of the chosen integrators which are both of second order.

Remark 4.2.1 The Störmer–Verlet scheme (4.5) is implicit in general. A few fixed point iterations yield the numerical solution with the desired accuracy. Notice however that the method becomes explicit in the Martinet flat case $\beta = 0$. One simply has to compute the components in a suitable order, for instance $p_{x,n+1}$, $p_{z,n+1}$, $p_{y,n+1/2}$, y_{n+1}, x_{n+1}, z_{n+1}, $p_{y,n+1}$.

4.2.1 Martinet flat case

We consider first the flat case $\beta = 0$ in the Hamiltonian system (4.2). As initial values we choose (cf. [ABCK97])

$$x(0) = y(0) = z(0) = 0, \quad p_x(0) = \cos\theta_0, \quad p_y(0) = \sin\theta_0, \quad p_z(0) = 10, \quad \text{where} \quad \theta_0 = \pi - 10^{-3}, \tag{4.6}$$

so that we start close to an abnormal geodesics, and we integrate the system over the interval $[0, 9]$.

Figure 4.1 displays the projection onto the (x, y)-plane of the numerical solution obtained with different stepsizes h by the two integrators. The initial value is at the origin, and the final state is indicated by a triangle. The circles represent the first conjugate point detected along the numerical solution, while the stars give the position of the first conjugate point on the exact solution of the problem. There is an enormous difference between the two numerical integrators. The symplectic (Störmer–Verlet) method (4.5) provides a qualitatively correct solution already with a large stepsize $h = 0.1$, and it gives an excellent approximation for stepsizes smaller than $h = 0.05$. On the other hand, the non-symplectic, explicit Runge–Kutta method (4.4) gives completely wrong results, and stepsizes smaller than 10^{-3} are needed to provide an acceptable solution. An explanation of the different behavior of the two integrators will be given in Sect. 4.3 below.

As noticed in Sect. 4.1, the normal geodesics in the flat case are determined by a one-degree of freedom Hamiltonian system in the variables y and p_y. We therefore show in Figure 4.2 the projection onto the (y, p_y)-space of the solutions previously computed with stepsize $h = 0.05$. The exact solution starts at $(0, \sin\theta_0)$ above the saddle point, turns around the positive stationary point, crosses the p_y-axis at $(0, -\sin\theta_0)$, turns around the negative stationary point, and then continues periodically. The numerical approximation by the non-symplectic method covers more than one and a half periods, whereas the Störmer–Verlet and the exact solution cover less than one period for the time interval $[0, 9]$.

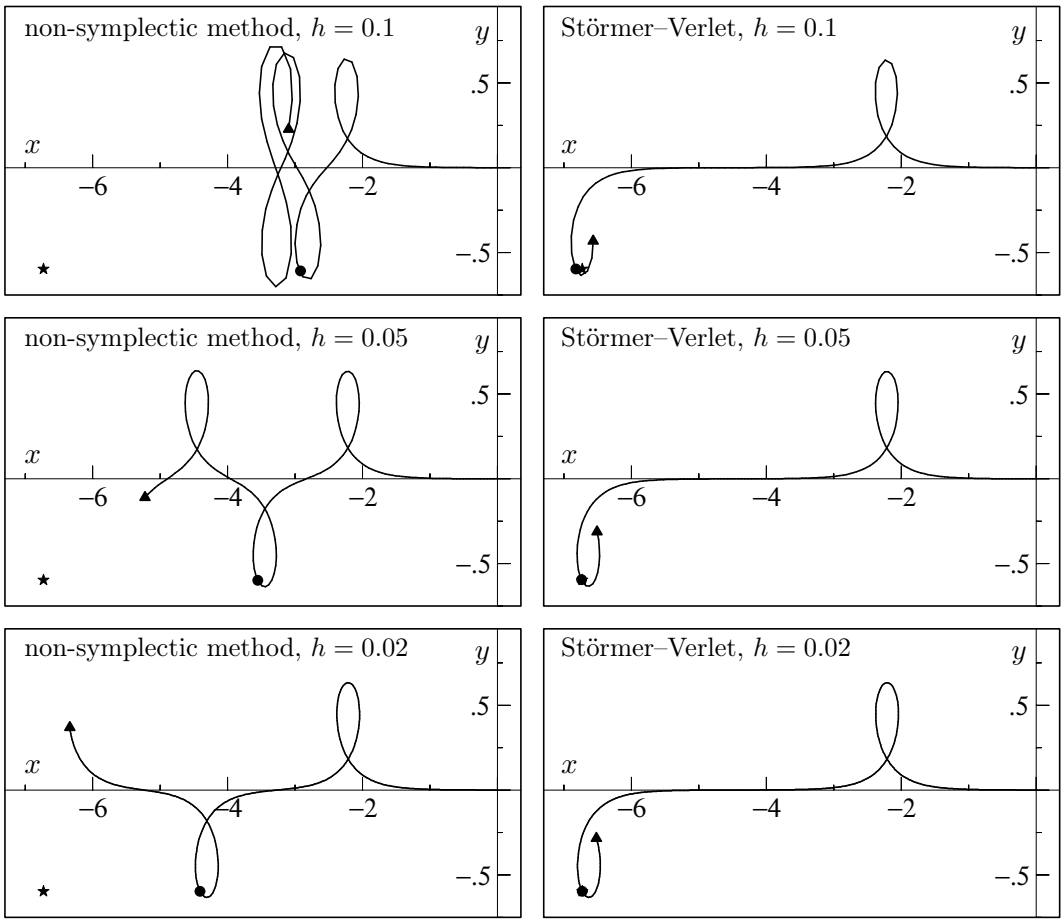


Figure 4.1: Trajectories in the (x, y)-plane for the flat case $\beta = 0$.

Table 4.1: Accuracy for the first conjugate time.

Martinet flat case		
h	RK2	Verlet
10^{-1}	4.504945	8.504716
10^{-2}	6.748262	8.416622
10^{-3}	8.360340	8.416412
10^{-4}	8.416349	8.416410
exact solution: $t_1 \approx 8.416409$		

Non integrable situation		
h	RK2	Verlet
10^{-1}	4.511294	4.883832
10^{-2}	7.380322	4.877056
10^{-3}	4.877183	4.876998
10^{-4}	4.876997	4.876997
exact solution: $t_1 \approx 4.876997$		

Since the conjugate point is not very sensitive with respect to perturbations in the initial value for p_y, the (y, p_y) coordinates of the conjugate point obtained by the non-symplectic integrator are rather accurate, but the corresponding integration time is completely wrong.

Table 4.1 lists the conjugate time obtained with the two integrators using various stepsizes. There is a significant difference between the two methods. We can see that with the Störmer–Verlet method (4.5) a stepsize of order $h = 10^{-2}$ provides a solution with 4 correct digits. A stepsize a hundred times smaller is needed to get the same precision with the non-symplectic method.

4.2.2 Non integrable perturbation

For our next numerical experiment we choose the perturbation parameter $\beta = -10^{-4}$ in the differential equation (4.2). We consider the same initial values and the same integration interval as in Sect. 4.2.1. The exact solution is no longer periodic and, due to the fact that β is chosen negative, its projection onto the (y, p_y)-space slowly spirals inwards around the positive stationary point (see right picture in Figure 4.4).

Figures 4.3 and 4.4 and Table 4.1 display the numerical results obtained by the two integrators for the differential equation (4.2) with $\beta = -10^{-4}$. The interpretation of the symbols (triangles, circles, and stars) is the same as before. The excellent behavior of the symplectic integrator is even more spectacular than in the flat case, and the pictures obtained for the Störmer–Verlet method agree extremely well with the exact solution. The non-symplectic method gives qualitatively wrong solutions for stepsizes larger than $h = 0.01$. In the (y, p_y)-space it alternatively spirals around the right and left stationary points whereas the exact solution spirals only around the positive stationary point. In contrast to the Martinet flat case, the conjugate point obtained by the non-symplectic method is here wrong also in the (y, p_y)-space.

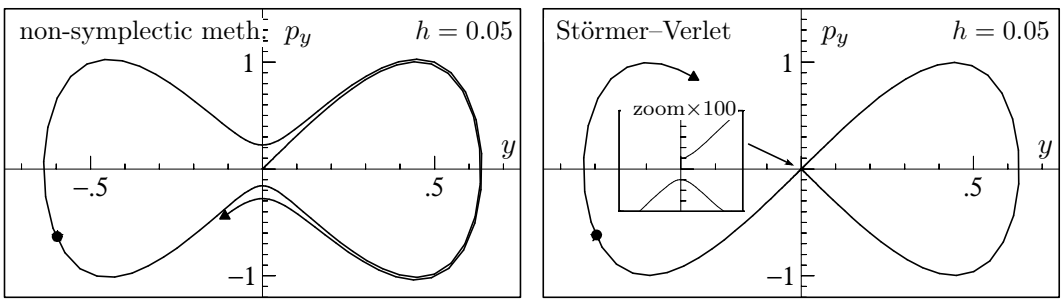


Figure 4.2: Phase portraits in the (y, p_y)-plane for the flat case $\beta = 0$.

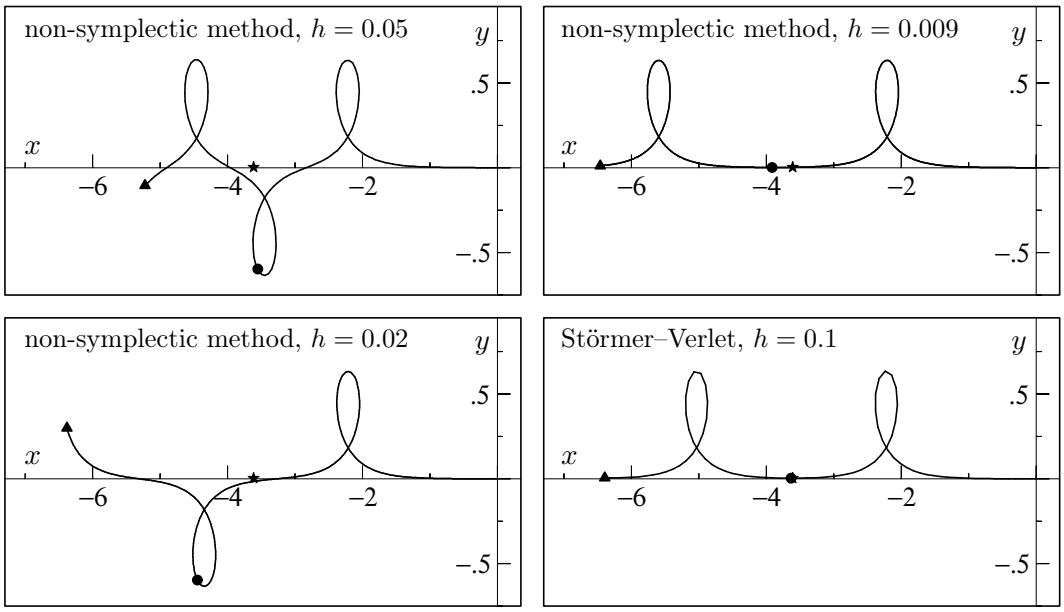


Figure 4.3: Trajectories in the (x, y)-plane for the non integrable case $\beta = -10^{-4}$.

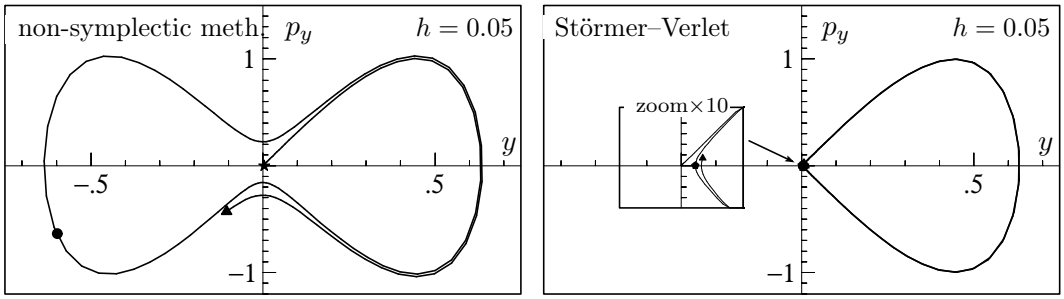


Figure 4.4: Phase portraits in the (y, p_y)-plane for the non integrable case $\beta = -10^{-4}$.

4.2.3 An asymptotic formula on the first conjugate time in the Martinet flat case

Now that we have shown the efficiency of symplectic integrators, we can make more precise the asymptotic behavior studied in [ABCK97]. For the initial values of (4.6) and $\beta = 0$, consider the ratio $R = t_1\sqrt{p_z}/(3K(k))$ where t_1 is the first conjugate time for the normal geodesic, and $K(k)$ is an elliptic integral of the first kind,

$$K(k) = \int_0^{\pi/2} \frac{1}{\sqrt{1 - k^2 \sin^2 u}} du, \qquad k = \sin(\theta_0/2).$$

By studying analytic solutions for the normal geodesics, it is proved in [ABCK97] that this ratio satisfies the inequality $2/3 \leq R \leq 1$. It follows from a rescaling of the equations (4.2) that R is independent of p_z.

In Figure 4.5, we represent the values of $1 - R$ as a function of $\varepsilon = \pi - \theta_0$, for various initial values θ_0. The numerical results indicate that the ratio R depends on θ_0, and $R \longrightarrow 1^-$ slowly for $\theta_0 \longrightarrow \pi^-$.

4.3 Backward error analysis

The theory of backward error analysis is fundamental for the study of geometric integrators and it is treated in much detail in the monographs of Sanz-Serna & Calvo [SSC94], Hairer, Lubich & Wanner [HLW06, Chap. IX], and Leimkuhler & Reich [LR04]. It allows us to explain the numerical phenomena encountered in the previous section.

4.3.1 Backward error analysis and energy conservation

We briefly present the main ideas of backward error analysis for the study of symplectic integrators, see [HLW06, Chap. IX]. Consider a system of differential equations

$$\dot{y} = f(y), \quad y(0) = y_0 \tag{4.7}$$

and a numerical integrator $y_{n+1} = \Phi_h(y_n)$ of order p. The idea is to search for a *modified differential equation* written as a formal series in powers of the stepsize h,

$$\dot{\widetilde{y}} = \widetilde{f}(\widetilde{y}) = f(\widetilde{y}) + h^p f_{p+1}(\widetilde{y}) + h^{p+1} f_{p+2}(\widetilde{y}) + \dots, \tag{4.8}$$

such that $y_n = \widetilde{y}(t_n)$ for $t_n = nh$, $n = 0, 1, 2, \dots$, in the sense of formal power series. The motivation of this approach is that it is often easier to study the modified equation (4.8) than directly the numerical solution.

What makes backward error analysis so important for the study of symplectic integrators is the fact that, when applied to a Hamiltonian system $\dot{y} = J^{-1}\nabla H(y)$, the modified equation (4.8) has the same structure $\dot{\widetilde{y}} = J^{-1}\nabla \widetilde{H}(\widetilde{y})$ with a *modified Hamiltonian*

$$\widetilde{H}(y) = H(y) + h^p H_{p+1}(y) + h^{p+1} H_{p+2}(y) + \dots \ .$$

However, the series usually diverges, so a truncation at a suitable order $N(h)$ is necessary,

$$\widetilde{H}(y) = H(y) + h^p H_{p+1}(y) + \dots + h^{N-1} H_N(y).$$

This truncation induces an error that can be made exponentially small, by choosing $N(h) \sim C/h$, see [HLW06, Theorem IX.8.1]. More precisely, we have that for $t_n = nh$ and $h \to 0$,

$$\widetilde{H}(y_n) = \widetilde{H}(y_0) + \mathcal{O}(t_n e^{-h_0/h}). \tag{4.9}$$

as long as the numerical solution $\{y_n\}$ stays in a compact set. On intervals of length $\mathcal{O}(e^{h_0/2h})$, the modified Hamiltonian $\widetilde{H}(y)$ is thus exactly conserved up to exponentially small terms.

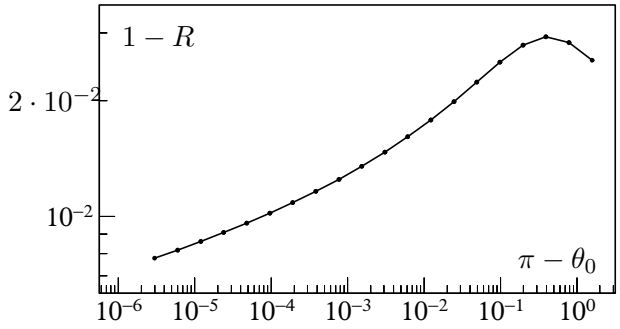


Figure 4.5: Illustration of the asymptotic behavior of R (Störmer–Verlet scheme with stepsize $h = 10^{-4}$).

4.3.2 Backward error analysis for the Martinet problem

Symplectic integrators are successfully applied in the long-time integration of Hamiltonian systems, for instance in astronomy (e.g. the Outer Solar System over 100 million years [HLW06, Sect. I.2.4]), or in molecular dynamics [LR04, Chap. 11]. The situation for the Martinet case is quite different because we are interested in the numerical integration of Hamiltonian systems on relatively short time intervals. Nevertheless, symplectic integrators still reveal very efficient. Here, the essential difficulty is that the solution approaches a few times the critical point $(y, p_y) = 0$ in the phase space. We show in this section that symplectic integrators resolve close approaches to such a critical point with high accuracy even for large stepsizes, whereas non-symplectic ones do not reproduce the correct behavior (except for very small stepsizes).

4.3.2.1 Martinet flat case

Consider the Martinet problem (4.2) in the flat case $\beta = 0$. Its interesting dynamics takes place in the (y, p_y) plane, and it is not influenced by the other variables (only by their initial values). We put $\eta = (y, p_y)$, and we denote by $f(\eta)$ the Hamiltonian vector field composed by the corresponding two equations of (4.2). For a numerical integrator of order $p = 2$, the associated modified differential equation has the form

$$\dot{\widetilde{\eta}} = f(\widetilde{\eta}) + h^2 f_3(\widetilde{\eta}) + \mathcal{O}(h^3). \tag{4.10}$$

Consider first the symplectic Störmer–Verlet method. It follows from Sect. 4.3.1 that its modified differential equation is Hamiltonian, and from (4.9) that the modified Hamiltonian $\widetilde{H}(\eta)$ is preserved up to exponentially small terms along the numerical solution. This implies that the numerical solution remains exponentially close to a periodic orbit in the (y, p_y)-space. The critical point ($y = 0$, $p_y = 0$) is a saddle point also for the modified differential equation (because the origin is stationary also for the numerical solution and thus for the modified equation). Therefore, any numerical solution starting close to the origin has to come back to it after turning around one of the stationary points. The minimal distance to the origin will always stay the same (see the zoom in Figure 4.2). This explains the good behavior of symplectic integrators.

For the non-symplectic integrator, the term $h^2 f_3(\eta)$ is not Hamiltonian. Therefore the solution of the modified differential equation (and hence also the numerical solution) is no longer periodic. In fact, it spirals outwards and after surrounding the first stationary point, the numerical solution does not approach the saddle point sufficiently close, which induces a faster dynamics as can be observed in Figures 4.1 and 4.2. This causes a huge error, because close to the saddle point the numerical solution is most sensitive to errors.

4.3.2.2 Non integrable perturbation

In this case, the argument in the comparison of symplectic and non-symplectic integrators is very similar to the discussion of the Van der Pol's equation in [HLW06, Sect. XII.1]. For $\beta \neq 0$ (non integrable perturbation), the dynamics takes place in the four dimensional space with variables $\eta = (x, y, p_x, p_y)$. In this space the system (4.2) becomes

$$\dot{\eta} = f(\eta) + \beta g(\eta)$$

where $f(\eta)$ is the Hamiltonian vector field corresponding to $\beta = 0$ and $g(\eta) = \mathcal{O}(1)$ depends smoothly on β. Here, the modified equation becomes

$$\dot{\widetilde{\eta}} = f(\widetilde{\eta}) + \beta g(\widetilde{\eta}) + h^2 f_3(\widetilde{\eta}) + \mathcal{O}(h^3 + \beta h^2),$$

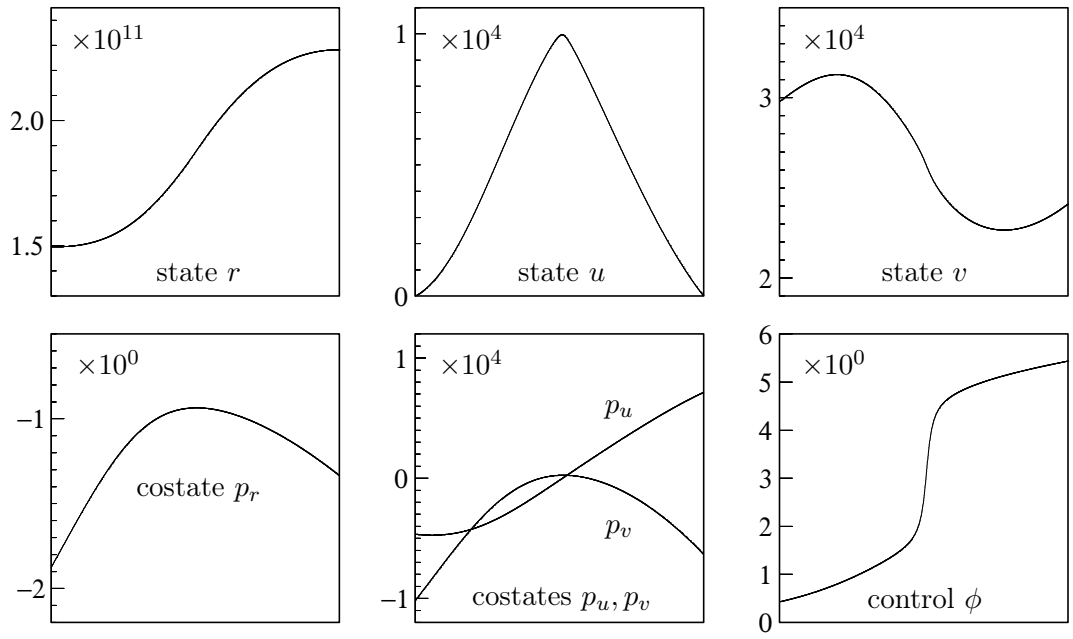


Figure 4.6: Exact solution of the orbit transfer problem on the time interval $[0, t_f]$.

where the perturbation term $h^2 f_3(\eta)$ is the same as for the Martinet flat case.

For the symplectic integrator, the perturbation $\beta g(\eta)$ has the same effect for the original problem as for $\dot{\widetilde{\eta}} = f(\widetilde{\eta}) + h^2 f_3(\widetilde{\eta}) + \dots$. This explains the correct qualitative behavior for small h and small β. There is no restriction on the stepsize h compared to the size of β.

For the non-symplectic integrator, each of the perturbation terms $\beta g(\eta)$ and $h^2 f_3(\eta)$ destroys the periodic orbits in the subsystem for the (y, p_y) variables, and the dominant one will determine the behavior of the numerical solution. Only when $h^2 \ll |\beta|$, the numerical solution will catch the correct dynamics of the problem. In Figures 4.3 and 4.4, where $\beta = -10^{-4}$, this condition is not satisfied for $h \geq 10^{-2}$. Since β is chosen small and negative, the two perturbation terms are conflicting. The term $\beta g(\eta)$ causes the solution to spiral around the positive stationary point, whereas the term $h^2 f_3(\eta)$ causes it to spiral alternatively around both stationary points. For too large stepsizes the qualitative behavior of the non-symplectic integrator (4.4) is thus completely wrong.

Remark 4.3.1 The problem (4.2) with $\beta = 0$ has a lot of symmetries. In the (y, p_y)-space the orbits are symmetric with respect to the y-axis and also with respect to the p_y-axis. If we apply a symmetric numerical integrator (not necessarily symplectic), it is possible to prove the same qualitative behavior as for the symplectic Störmer–Verlet method. This follows from the fact that the solution of the modified equation (numerical orbit) corresponding to a symmetric method has the same symmetry properties as the exact flow (see [HLW06, Sect. IX.2] for precise statements). Consequently, in the (y, p_y) plane and for $\beta = 0$, the solution will stay exponentially near to a closed orbit, as it is the case for symplectic integrators. In the non integrable case, the good behavior of symmetric methods can be explained as in Sect. 4.3.2.2 for symplectic methods.

4.4 Orbital transfer of a spacecraft

We consider the orbit transfer problem presented in [BH69, pp. 66–68] and studied in [Hag00]. The problem is to transfer a spacecraft with constant thrust force T from a given

initial circular orbit r_0 to the largest possible circular orbit for a given length of time t_f. The control function is the thrust-direction given by an angle $\phi(t)$. The state functions are (r, u, v), where $r(t)$ is the radial distance of spacecraft from attracting center, $u(t)$ is the radial component of velocity, and $v(t)$ is the tangential component of velocity.

The optimal control problem can be formulated as maximizing the radial distance $r(t_f)$ at the final time, subject to the differential equations

$$\begin{aligned} \dot{r} &= u \\ \dot{u} &= \frac{v^2}{r} - \frac{\mu}{r^2} + \frac{T \sin\phi}{m_0 - |\dot{m}|t} \\ \dot{v} &= -\frac{uv}{r} + \frac{T\cos\phi}{m_0 - |\dot{m}|t} \end{aligned} \tag{4.11}$$

with boundary conditions

$$\begin{aligned} r(0) = r_0, \qquad & u(t_f) = 0 \\ u(0) = 0 \qquad & v(t_f)^2 r(t_f) - \mu = 0 \\ v(0) = \sqrt{\frac{\mu}{r_0}} & \end{aligned} \tag{4.12}$$

The constants are $m_0 = 10000$ kg (initial mass of spacecraft), $|\dot{m}| = 12.9$ kg/day (fuel consumption rate), $r_0 = 1.496 \cdot 10^{11}$ m (distance Sun-Earth), $T = 8.336$ N (thrust force), $\mu = 1.32733 \cdot 10^{20} \mathrm{m}^3/\mathrm{s}^2$ (gravitational constant for sun), and $t_f = 193$ days ($\approx 1.67 \cdot 10^7$ seconds).

We solve this problem using Pontryagin's maximum principle. The Hamiltonian is

$$H = p_r u + p_u\Big(\frac{v^2}{r} - \frac{\mu}{r^2} + \frac{T\sin\phi}{m_0 - |\dot{m}|t}\Big) + p_v\Big(-\frac{uv}{r} + \frac{T\cos\phi}{m_0 - |\dot{m}|t}\Big) + p_t \tag{4.13}$$

and the differential equation for the adjoint state (p_r, p_u, p_v, p_t) is

$$\begin{aligned} \dot{p}_r &= p_u\Big(\frac{v^2}{r^2} - 2\frac{\mu}{r^3}\Big) - p_v\frac{uv}{r^2} \\ \dot{p}_u &= -p_r + p_v\frac{v}{r} \\ \dot{p}_v &= -2p_u\frac{v}{r} + p_v\frac{u}{r} \\ \dot{p}_t &= -p_u\frac{T|\dot{m}|}{(m_0 - |\dot{m}|t)^2}\sin\phi - p_u\frac{T|\dot{m}|}{(m_0 - |\dot{m}|t)^2}\cos\phi, \quad p_t(0) = 0. \end{aligned} \tag{4.14}$$

The extremality condition for $p(t_f)$ is given by

$$p_v(t_f)v(t_f) - 2(p_r(t_f) - 1)r(t_f) = 0. \tag{4.15}$$

Applying the Pontryagin principle, the control $\phi(t)$ minimizes the Hamiltonian (4.13) at all times. This yields

$$\sin(\phi) = \frac{-p_u}{\sqrt{p_u^2 + p_v^2}}, \qquad \cos(\phi) = \frac{-p_v}{\sqrt{p_u^2 + p_v^2}}.$$

The Hamiltonian system (4.11) & (4.14) with boundary conditions (4.12) & (4.15) is solved by the standard single shooting technique. In Figure 4.6 we plot the exact solution (computed numerically with high precision). The thrust direction ϕ starts close to the

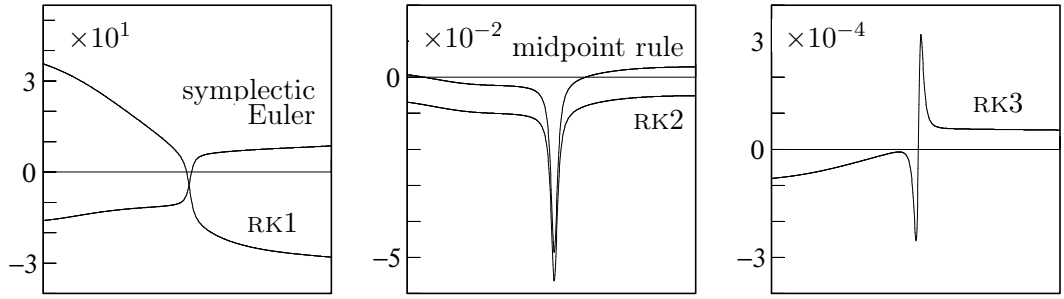


Figure 4.7: Errors in the Hamiltonian for various numerical integrators applied with constant stepsize (1000 steps). Explicit Runge Kutta methods (non-symplectic) of orders $1, 2, 3$ (RK1,RK2,RK3) compared to the symplectic Euler method (order 1) and the implicit midpoint rule (order 2, symplectic).

tangential direction, and rotates during the orbit transfer with an angle of about $\approx 2\pi$. At the middle of the time interval, the thrust direction rotates more rapidly.

In Figure 4.7 we plot the relative errors in the Hamiltonian as functions of time (compared to the Hamiltonian of the exact solution) for various symplectic and non-symplectic numerical integrators of orders 1, 2, and 3. We do not oberve any significant advantage for the symplectic integrators. We notice a qualitative different behavior of the methods of order 2. There, the error in the Hamiltonian has a peak in the middle of the integration interval, and it comes back to about the same value it had before. For the midpoint rule this is due to its symmetry, and for the second order Runge–Kutta methods this follows from the fact that for methods with even order the dominant error term behaves like that of a symmetric integrator (backward error analysis). The results of this experiment are not really surprising, because the solution does not have a quasi-oscillatory behavior, so that errors in the Hamiltonian could be compensated by a symplectic integrator.

4.5 Submerged rigid body

We consider the autonomous submarine model introduced in [CHSC08b]. For simplicity, we restrict ourself to the vertical planar situation: the rigid body moves in the $xz-$plane exclusively. The state of the rigid body is given by $q(t) = \big(b_1(t), b_3(t), \theta(t), \nu_1(t), \nu_3(t), \Omega_2(t)\big)$, where $b_1(t), b_3(t)$ denote the position vector and $\theta(t)$ represents the diving angle, and $\nu_1(t), \nu_3(t)$, and $\Omega_2(t)$ are the corresponding translational and angular velocities.

Given a fixed time interval of length $t_f > 0$, we search for the energy minimizing trajectory to get the submarine from a configuration $q(0)$ to a configuration $q(t_f)$, e.g. $t_f = 5$ and

$$q(0) = (0, 0, 0, 0, 0, 0), \qquad q(5) = (1, 1, 0, 0, 0, 0).$$

Here, the energy is defined as (a more realistic energy model is derived in [CHSC08a])

$$E(q) = \frac{1}{2} \int_0^{t_f} (\varphi_{\nu_1}^2 + \varphi_{\nu_3}^2 + \tau_{\Omega_2}^2) dt$$

where $\varphi_{\nu_1}(t), \varphi_{\nu_3}(t), \tau_{\Omega_2}(t)$ are the control functions. The dynamics are

$$\begin{aligned}
\dot{b}_1 &= \nu_1 \cos\theta + \nu_3 \sin\theta, \quad \dot{\nu}_1 = \frac{1}{m_1}\big(-m_3\nu_3\Omega_2 - D_\nu^2\nu_1^3 - D_\nu^1\nu_1 + G\sin\theta + \varphi_{\nu_1}\big) \\
\dot{b}_3 &= -\nu_1 \sin\theta + \nu_3 \cos\theta, \quad \dot{\nu}_3 = \frac{1}{m_3}\big(m_1\nu_1\Omega_2 - D_\nu^2\nu_3^3 - D_\nu^1\nu_3 - G\cos\theta + \varphi_{\nu_3}\big) \\
\dot{\theta} &= \Omega_2, \quad \dot{\Omega}_2 = \frac{1}{I_{b_2}}\big(-D_\Omega^2\Omega_2^3 - D_\Omega^1\Omega_2 + \rho g\mathcal{V}(-z_B\sin\theta + x_B\cos\theta) + \tau_{\Omega_2}\big)
\end{aligned}$$

with positive constants: $m_1 = m_3 = m + M$ (masses) $m = 126.55$ kg, $M = 70$ kg, $D_\nu^1 = -27.0273, D_\nu^2 = -897.6553, D_\Omega^1 = -13.793, D_\Omega^2 = -6.45936$ (drags), $G = -3$ N (Archimède), $I_{b_2} = 5.29$ kg.m^2, $g = 9.80$ m.s^{-2}, $\rho g\mathcal{V} = mg - G, z_B = -7{\cdot}10^{-3}$ m, $x_B = 0$ m (buoyancy). These numerical values were derived from experiments performed on a test-bed vehicule, see [CHSC08b].

Here, the Hamiltonian system is very sensitive when considered as an initial value problem (i.e. $q(0)$ and $p(0)$ given). When ones slightly perturbates initial conditions (e.g. by multiplying $p(0)$ or $q(0)$ by $1 + 10^{-10}$), the corresponding solution explodes. For this reason, single shooting methods fail to solve the boundary value problem, and we use a multiple shooting method.

For this system we found one normal extremal with a conjugate point. The corresponding states, costates and controls are represented in Figure 4.8 (using a high-order integrator).

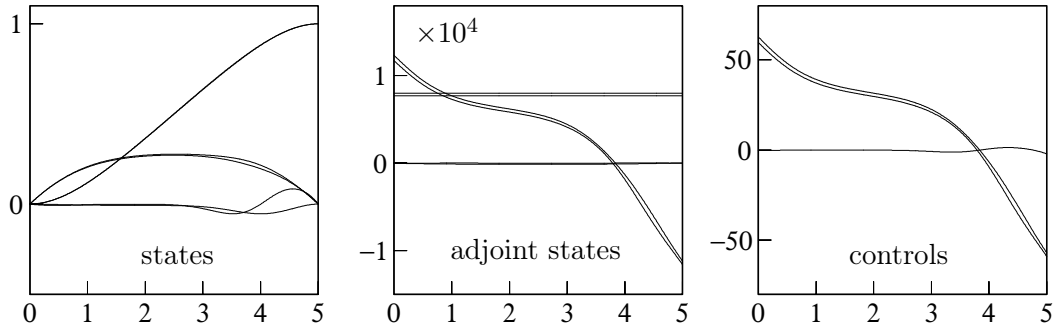


Figure 4.8: Extremal obtained for $p(0) = (7709.864500233298,\ 7988.036994413952,\ -3.0163588901640024,\ 11707.858394005056,\ 12318.149504556683,\ -1.0570538454444238)$.

In Figure 4.9, we compare for the same stepsize ($h = 0.05$) the accuracy of the implicit midpoint rule (order 2), which is a symplectic integrator, with an explicit Runge-Kutta method (non symplectic) of the same order (RK2, see (4.4)). The numerical solution and the determinant of the Jacobian $\frac{\partial q}{\partial p_0}$ are obtained on the grid points of integration. For the computation of conjugate points we need a continuous approximation of the solution, which is obtained by cubic Hermite interpolation. The resulting interpolation error is of size $\mathcal{O}(h^4)$ and thus negligible for second order methods. Notice that linear interpolation would introduce an error $\mathcal{O}(h^2)$ that is of the same size as the truncation (global) error of the numerical integrators. The mark in the middle of Figure 4.9 corresponds to the conjugate point of the exact solution. There is again no real advantage for the symplectic integrator. The implicit midpoint rule (symplectic) is only twice more accurate than the non-symplectic method RK2, and this is due to the size of the error constants of the methods and not to symplecticity.

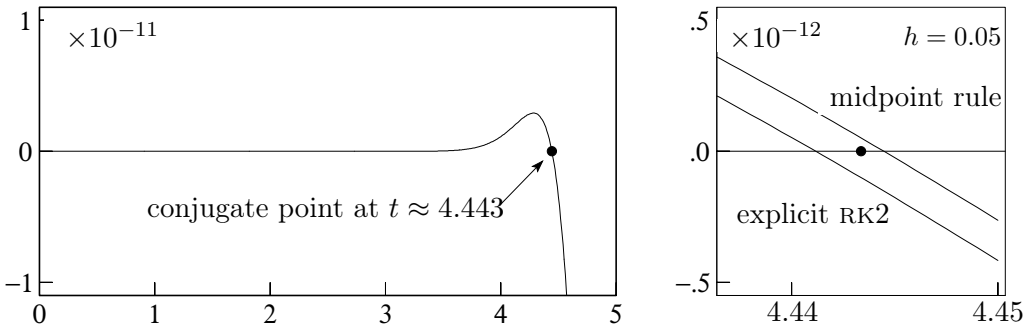


Figure 4.9: Computation of det $\frac{\partial q}{\partial p_0}$. Left picture: exact solution on the time interval $[0, 5]$. Right picture: Implicit midpoint rule (symplectic) and an explicit Runge-Kutta method (RK2, non-symplectic) for the same stepsize $h = 0.05$ with cubic Hermite interpolation.

4.6 Backward error analysis for optimal control problems?

The aim of the section is to investigate the possible extension of the well-established theory of backward error analysis for ordinary differential equations [HLW06, Sect. IX] to optimal control problems. The motivation for this study is the recent result of Hager [Hag00] and Bonnans & Laurent-Varin [BLV06]. The main result of Hager [Hag00] is that if a Runge-Kutta discretization is applied to an optimal control problem, then it is equivalent to a symplectic partitioned Runge-Kutta method applied to the Hamiltonian system arising in the Pontryagin maximum principle. In [BLV06], the symplecticity of the arising partitioned Runge-Kutta method is highlighted and used to derive order conditions for general Runge-Kutta discretizations.

4.6.1 Pontryagin principle and Runge-Kutta discretizations

Let $f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ and $\Phi : \mathbb{R}^n \to \mathbb{R}$ be two smooth functions (C^∞). Let $A \subset \mathbb{R}^m$be a domain of admissible controls (typically $A = [0, 1]^m$ or $A = [-1, 1]^m$ or $A = \mathbb{R}^m$). Note that in applications, we usually have $m \ll n$, i.e. few controls and lots of state equations.

We consider an optimal control problem of the form

$$(P) \begin{cases} \text{Min } \Phi(x(1)), \\ \dot{x}(t) = f(x(t), u(t)), \quad t \in (0, 1), \\ x(0) = x^0, \\ u(t) \in A, \quad t \in (0, 1). \end{cases}$$

In general the regularity of the control is very poor (only mesurable in the general theory). However, for simplicity we restrict our analysis to continuous control functions (in fact C^∞). The main tool for studying optimal control problems is the Pontryagin maximum principle. It states that the solution (if is exists) of on optimal control problem satisfies a boundary value problem, which is Hamiltonian, and involves a costate function $p(t)$.

Theorem 4.6.1 ***Pontryagin principle : necessary conditions of optimality*** *Consider the Hamiltonian function $H : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ defined by*

$$H(x, p, u) = p^T f(x, u). \tag{4.16}$$

Under some additional technical hypothesis (see e.g. [Eva83, MS82]), there exists a co-state function $p : (0,1) \to \mathbb{R}^n$ which never vanishes such that the solution of (P) satisfies, for $t \in (0,t)$

$$(OC)\begin{cases} \dot{x}(t) &= H_p(x(t),p(t),u(t)) \\ \dot{p}(t) &= -H_x(x(t),p(t),u(t)) \\ H(x(t),p(t),u(t)) = \min_{\alpha\in A} H(x(t),p(t),\alpha) \\ x(0) = x^0, \quad p(1) = \Phi'(x(1)). \end{cases}$$

Furthermore, the Hamiltonian is conserved, i.e. $H(x(t),p(t),u(t))$ is constant along $(0,1)$.

If $\min_{\alpha\in A} H$ est attained at an interior point of A, which is true e.g. for $A = \mathbb{R}^m$, then (OC) implies the following differential algebraic system (OC') with boundary conditions:

$$(OC')\begin{cases} \dot{x}(t) &= H_p(x(t),p(t),u(t)) = f(x(t),u(t)) \\ \dot{p}(t) &= -H_x(x(t),p(t),u(t)) = -p^T f_x(x(t),u(t)) \\ 0 &= H_u(x(t),p(t),u(t))) = p^T f_u(x(t),u(t)) \\ x(0) = x^0, \quad p(1) = \Phi'(x(1)). \end{cases}$$

From here, we assume $A = \mathbb{R}^m$ (problem without constraints).

Case of a differential algebraic system of index 1.
Assume that at each point (x,p,u) of the trajectory of the solution of (OC), the matrix

$$p^T f_{uu}(x,u) \in \mathbb{R}^{m\times m}$$

is invertible. The implicit functions theorem implies the (local) existence of a control function $\varphi : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^m$ satisfying

$$p^T f_u(x,u) = 0 \iff u = \varphi(x,p), \tag{4.17}$$

The couple $(x(t),p(t))$ is then the solution of a Hamiltonian system (see [BLV06]),

$$\dot{x}(t) = \mathcal{H}_p(x(t),p(t)), \quad \dot{p}(t) = -\mathcal{H}_x(x(t),p(t)),$$

with Hamiltonian

$$\mathcal{H}(x,p) = p^T f(x,\varphi(x,p)).$$

Symplectic partitioned Runge-Kutta discretization For the numerical solution of (P), we apply a Runge-Kutta discretization.

$$(DP)\begin{cases} \text{Min } \Phi(x_N), \\ x_{k+1} &= x_k + h\sum_{i=1}^s b_i f(x_{ki},\overline{u}_{ki}), \quad k = 0,\dots N-1 \\ x_{ki} &= x_k + h\sum_{j=1}^s a_{ij} f(x_{kj},\overline{u}_{kj}), \quad i = 0,\dots s \\ x_0 = x^0. \end{cases}$$

We obtain a discrete optimization problem with constraints which can be solved using a gradient-type method as in [Hag99]. Alternatively, a standard approach is to introduce Lagrange parameters p_k, p_{ki}, $i = 1\dots s$ associated to the constraints. The results of Hager [Hag00] and Bonnans & Laurent-Varin [BLV06] show that the discretization (DP) is equivalent to the symplectic Runge-Kutta method applied to the Hamiltonian system (OC),

$$(DOC)\begin{cases} x_{k+1} &= x_k + h\sum_{i=1}^s b_i f(x_{ki},\overline{u}_{ki}), \quad k = 0,\dots N-1 \\ x_{ki} &= x_k + h\sum_{j=1}^s a_{ij} f(x_{kj},\overline{u}_{kj}), \quad i = 0,\dots s \\ p_{k+1} &= x_k - h\sum_{i=1}^s \widehat{b}_i p_{pi}^T f_x(x_{ki},\overline{u}_{ki}), \quad k = 0,\dots N-1 \\ p_{ki} &= x_k - h\sum_{j=1}^s \widehat{a}_{ij} p_{pj}^T f_x(x_{kj},\overline{u}_{kj}), \quad i = 0,\dots s \\ 0 &= p_{ki}^T f_u(x_{ki},\overline{u}_{ki}), \text{ i.e } \overline{u}_{ki} = \varphi(x_{ki},p_{ki}), \\ x_0 = x^0, \quad p_N = \Phi'(x_N). \end{cases}$$

where

$$\widehat{b}_i = b_i, \qquad \widehat{b}_i b_j = \widehat{b}_i a_{ij} + b_j \widehat{a}_{ji} \tag{4.18}$$

is precisely the condition for partioned Runge-Kutta methods to be symplectic. This is true under the assumption $b_i \neq 0$ for all i, which permits to solve system (4.18) for the $\widehat{a}_{ij}$'s.

This result is very useful for studying order conditions of Runge-Kutta discretization in optimal control problems. It has been shown that the order of convergence of the Runge-Kutta discretization in (DP) is not the order q of the Runge-Kutta discretization when applied to (uncontrolled) ordinary differential equations but it is precisely the order of convergence p of the partitioned Runge-Kutta method in (DOC). In particular, computing the order conditions by hand, Hager [Hag00] shows that $p = q$ for $p \leq 2$ and for $p \leq 4$ if the method is explicit. Bonnans & Laurent-Varin [BLV06] give simplified order conditions for general Runge-Kutta methods with the help of bi-coloured rooted trees.

Example We consider the *explicit Euler method* with $h = \frac{1}{N}$, and $x(t_k) \approx x_k,\ t_k = kh$:

$$\left\{ \begin{array}{l} \text{Min } \Phi(x_N), \\ \quad x_{k+1} \ = \ x_k + hf(x_k, \overline{u}_k), \quad k = 0, \dots N-1 \\ x_0 = x^0. \end{array} \right.$$

This discretization is equivalent to apply a symplectic partitioned Runge–Kutta method to problem (OC'), here the *symplectic Euler method*:

$$\left\{ \begin{array}{rcl} \text{Min } \Phi(x_N), & & \\ x_{k+1} & = & x_k + hf(x_k, \overline{u}_k), \\ p_{k+1} & = & p_k - hp_{k+1}^T f_x(x_k, \overline{u}_k), \\ 0 & = & p_{k+1}^T f_u(x_k, \overline{u}_k), \text{ i.e } \overline{u}_k = \varphi(x_k, p_{k+1}), \\ x_0 = x^0, & & p_N = \Phi'(x_N). \end{array} \right.$$

with $k = 0, \dots N-1$.

4.6.2 Backward error analysis

To better understand the role of symplectic integrators for optimal control problems, a natural question is to investigate whether the numerical solution (DP) of the optimal control problem (P) can be interpreted (formally) as the exact solution of a *modified* optimal control problem $(\widetilde{P})$,

$$(\widetilde{P}) \left\{ \begin{array}{rcl} \text{Min } \Phi(x(1)), & & \\ \dot{x}(t) & = & \widetilde{f}(x(t), u(t)), \quad t \in (0,1), \\ x(0) & = & x^0, \end{array} \right.$$

where

$$\widetilde{f}(x,u) = f(x,u) + hf_2(x,u) + h^2 f_3(x,u) + \dots . \tag{4.19}$$

Here, the modified problem $(\widetilde{P})$ has also modified necessary conditions $(\widetilde{OC})$ from the Pontryagin principle,

$$(\widetilde{OC}) \left\{ \begin{array}{rcl} \dot{x}(t) & = & \widetilde{f}(x(t), u(t)), \\ \dot{p}(t) & = & -p^T \widetilde{f}_x(x(t), u(t)), \\ 0 & = & p^T \widetilde{f}_u(x(t), u(t)) \end{array} \right. \iff \quad u(t) = \widetilde{\varphi}(x(t), p(t)),$$
$$x(0) = x^0, \quad p(1) = \Phi'(x(1)),$$

where

$$\widetilde{\varphi}(x,p) = \varphi(x,p) + h\varphi_2(x,p) + h^2\varphi_3(x,p) + \dots .$$

is (formally) given by

$$0 = p^T \widetilde{f}_u(x,u) \quad \Longleftrightarrow \quad u = \widetilde{\varphi}(x,p).$$

Statement 4.6.2 *There exists a modified vector field (4.19) such that the numerical solution (x_k, p_k, u_k) of (DOC), with*

$$u_k := \widetilde{\varphi}(x_k, p_k),$$

can be interpreted (formally) as the exact solution $(x(t), p(t), u(t))$ of $(\widetilde{OC})$, i.e.

$$x_k = x(t_k), \quad p_k = p(t_k), \quad u_k = u(t_k), \quad t_k = kh.$$

4.6.3 The linear-quadratic case

A situation where Statement 4.6.2 is true. Consider the following optimal control problem, where the state and control $x, u \in \mathbb{R}^n$ have the same dimension $n = m$, $A, Z, S, B \in \mathbb{R}^{n\times n}$, with Z symmetric, and S symmetric positive definite, and B invertible (e.g. $B = Id$).

$$\left\{ \begin{array}{l} \text{Min } \frac{1}{2}\int_0^1 (x^T Z x + u^T S u)dt, \\ \dot{x} \;=\; Ax + Bu \\ x(0) \text{ given} \end{array} \right.$$

Notice that this optimal control problem can be rewritten in the form (P) by introducing an additional state c satisfying $\dot{c} = (x^T Z x + u^T S u)/2$, and by putting $\Phi(c,x) = c$. The associated adjoint variable p_c is constant, and it can be shown that it cannot be zero (we say that there exists no abnormal extremal). We normalize p_c to 1 and the Hamiltonian for this optimal control problem writes

$$H = \frac{1}{2}(x^T Z x + u^T S u) + p^T(Ax + Bu).$$

Applying the Pontryagin principle, and eliminating the control as detailed in Sect. 4.6.1, we arrive at the following reduced Hamiltonian:

$$\mathcal{H}(x,p) = \frac{1}{2}x^T Z x + p^T A x - \frac{1}{2}p^T B S^{-1} B^T p.$$

and the associated Hamiltonian system with boundary conditions is

$$\left\{ \begin{array}{ll} \dot{x} \;=\; Ax + Bu & (u = -S^{-1}B^T p) \\ \dot{p} \;=\; -Zx - A^T p & \\ x(0) \text{ given} & \\ p(1) = 0 & \end{array} \right.$$

We observe that the system is still linear,

$$\dot{y} = My, \qquad \text{where} \quad y = \begin{pmatrix} x \\ p \end{pmatrix}, \quad M = \begin{pmatrix} A & -BS^{-1}B^T \\ -Z & -A^T \end{pmatrix}.$$

Consider a symplectic integrator for this Hamiltonian system. Then the modified equation is also Hamiltonian:

$$\dot{y} = M_h y, \qquad \text{where} \quad M_h = J\Sigma_h,$$

and Σ_h is a symmetric matrix because $\Sigma_h y = \nabla^2 H(y)$. Therefore, M_h has the same form as M:

$$M_h = \begin{pmatrix} A_h & -\Lambda_h \\ -Z_h & -A_h^T \end{pmatrix}.$$

and $\Lambda_h = BS^{-1}B^T + \mathcal{O}(h)$ is still symmetric positive definite for small h.

Lemma 4.6.3 *Matrix $\Lambda_h \in \mathbb{R}^{n\times n}$ can be decomposed in the form $\Lambda_h = B_h S^{-1} B_h^T$ where $B_h \in \mathbb{R}^{n\times n}$ depends smoothly on h.*

Remark 4.6.4 Matrix B_h is not unique, for instance $B_h S^{-\frac{1}{2}} Q_h S^{\frac{1}{2}}$ instead of B_h is suitable for any orthogonal matrix Q_h.

Proof. Consider the (unique) Cholesky decomposition $\Lambda = LL^T$, where L is a lower-triangular matrix with positive diagonal. Define $B_h := L_h L^{-1} B$ where L_h is given by the Cholesky decomposition $\Lambda_h = L_h L_h^T$. Here, L_h is a smooth function of h. □

We therefore obtain the following modified optimal control problem:

$$\begin{cases} \text{Min } \frac{1}{2}\int_0^1 (x^T Z_h x + u^T S u)dt, \\ \dot{x} \;=\; A_h x + B_h u \\ x(0) \text{ given} \end{cases}$$

Counterexample for Statement 4.6.2. Consider the following optimal control problem, where $x_1(t), x_2(t), u_1(t) \in \mathbb{R}$,

$$\begin{cases} \text{Min } \frac{1}{2}\int_0^1 (x_1^2 + u_1^2)dt, \\ \dot{x}_1 \;=\; x_2 \\ \dot{x}_2 \;=\; -x_1 + u_1 \\ x_1(0), x_2(0) \text{ given} \end{cases}$$

Applying the Pontryagin principle, we arrive at a Hamiltonian system:

$$\begin{cases} \dot{x}_1 \;=\; x_2 \\ \dot{x}_2 \;=\; -x_1 + u_1 \qquad (u_1 = -p_2) \\ \dot{p}_1 \;=\; -x_1 + p_2 \\ \dot{p}_2 \;=\; -p_1 \\ x_1(0), x_2(0) \text{ given} \\ p_1(1) = p_2(1) = 0 \end{cases}$$

with Hamiltonian

$$H = \frac{1}{2}(x_1^2 + u_1^2) + p_1 x_2 + p_2(-x_1 + u_1)$$

Again, the system is linear,

$$\dot{y} = My, \qquad \text{where} \quad y = \begin{pmatrix} x_1 \\ x_2 \\ p_1 \\ p_2 \end{pmatrix}, \quad M = \begin{pmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & -1 \\ -1 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \end{pmatrix}.$$

We consider the implicit midpoint rule for solving this Hamiltonian system. It is the simplest symplectic and symmetric method. The modified differential equation for standard backward error analysis reads:

$$\dot{y} = \big(M + \frac{h^2}{12}M^3 + \mathcal{O}(h^4)\big)y, \qquad \text{where} \quad M^3 = \begin{pmatrix} 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 2 \\ 2 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{pmatrix}.$$

with modified Hamiltonian

$$\begin{aligned} \widetilde{\mathcal{H}}(x,p) &= \mathcal{H}(x,p) + h^2\mathcal{H}_3(x,p) + \dots \\ \mathcal{H}(x,p) &= \frac{1}{2}x_1^2 + p_1x_2 - p_2x_1 - \frac{1}{2}p_2^2, \\ \mathcal{H}_3(x,p) &= -x_1^2 - \frac{1}{2}x_2^2 - p_1x_2 + \frac{1}{2}p_1^2 + p_2^2. \end{aligned}$$

In particular, the modified differential equation for x_1 has the from

$$\dot{x}_1 = x_2 + h^2 f_3^1(x, \widetilde{\varphi}(x,p)) + \mathcal{O}(h^4) \stackrel{?}{=} x_2 + h^2(-x_2 + p_1) + \mathcal{O}(h^4)$$

This yields

$$f_3^1(x, \widetilde{\varphi}(x,p)) = -x_2 + p_1 + \mathcal{O}(h^2)$$

and $f_3^1(x, \widetilde{\varphi}(x,p))$ depends at order $\mathcal{O}(1)$ of p_1. However

$$\widetilde{\varphi}(x,p) = \varphi(x,p) + \mathcal{O}(h^2) = p_2 + \mathcal{O}(h^2),$$

is independent of p_1 at $\mathcal{O}(h^2)$ close, and so is $f_3^1(x, \widetilde{\varphi}(x,p))$ which is a contradiction.

However, there is a result when one considers more general stationary point control problems.

Stationary point control problems For optimal control problems, one searches for a minimum of the functional which to a control u associates the cost $\Phi(x(1))$. Now, we consider stationary points of this functional. We show that one can still interpret the numerical discretization of the optimal control problem as the exact solution of a stationary point control problem. The idea is to add extra controls to gain more freedom. In our example, we consider a new control u_2.

$$\begin{cases} \text{Stat } \frac{1}{2}\int_0^1 \big(x_1^2 + u_1^2 - h^2u_2^2 + \frac{h^2}{12}(-2x_1^2 - x_2^2) + \dots\big)dt, \\ \dot{x}_1 = x_2 + \frac{h^2}{\sqrt{12}}u_2 - \frac{h^2}{12}x_2 + \dots \\ \dot{x}_2 = -x_1 + u_1 - \frac{h^2}{12}u_1 + \dots \\ x_1(0), x_2(0) \text{ given} \end{cases}$$

Here, the cost function contains a negative term $-h^2u_2^2$. This is no longer an optimal control problem. This is why we consider a stationary point problem (notation *Stat*), related to the functional $u \to \int_0^1 (x^T Z_h x + u^T S_h u)dt$. Notice that one can replace the stationary objective *Stat* my a min-max problem $\min_{u_1}\max_{u_2}$.

For a general linear quadratic optimal control problem we have the following result.

Theorem 4.6.5 *Consider a linear quadratic optimal control problem, with state $x(t) \in \mathbb{R}^n$, and control $u(t) \in \mathbb{R}^m$,*

$$\left\{ \begin{array}{rcl} Min\ \frac{1}{2}\int_0^1 (x^T Z x + u^T S u)dt, \\ \dot{x} &=& Ax + Bu \\ x(0)\ given \end{array} \right.$$

where $A, Z \in \mathbb{R}^{n\times n}$, with Z symmetric, and $S \in \mathbb{R}^{m\times m}$ is symmetric positive definite, and $B \in \mathbb{R}^{n\times m}$ has rank m.

Consider a symplectic method applied to the Pontryagin Hamiltonian system associated to this optimal control problem (e.g. obtained by applying a direct Runge-Kutta discretization, see Sect. 4.6.1).

Then, there exists an integer r with $m \le r \le n$, and perturbed matrices $A_h, Z_h \in \mathbb{R}^{n\times n}$,$S_h \in \mathbb{R}^{r\times r}$,$B_h \in \mathbb{R}^{n\times r}$ satisfying

$$B_h = (B\ 0) + \mathcal{O}(h^p), \qquad S_h = \begin{pmatrix} S & 0 \\ 0 & 0 \end{pmatrix} + \mathcal{O}(h^p),$$

such that the numerical solution (x_k, p_k, u_k), with (as in Statement 4.6.2)

$$u_k := \widetilde{\varphi}(x_k, p_k),$$

can be interpreted as the exact solution $(x(t), p(t), u(t))$ of the modified stationary point control problem

$$\left\{ \begin{array}{rcl} Stat\ \frac{1}{2}\int_0^1 (x^T Z_h x + u^T S_h u)dt, \\ \dot{x} &=& A_h x + B_h u \\ x(0)\ given \end{array} \right.$$

i.e.

$$x_k = x(t_k), \quad p_k = p(t_k), \quad u_k = u(t_k), \quad t_k = kh.$$

Moreover, we show that the series in h for the perturbed matrices converge for h small enough, so that result is not only formal. Notice that matrices $B_h, \ldots$ are not unique, see Remark 4.6.4.

Proof. Since we apply a symplectic discretization to a Hamiltonian system, the modified equation for standard backward error analysis is of the form

$$\dot{y} = M_h y, \qquad \text{where} \quad M_h = J\Sigma_h,$$

where Σ_h is a symmetric matrix (because $\Sigma_h y = \nabla^2 H(y)$). So M_h has the form

$$M_h = \begin{pmatrix} A_h & -\Lambda_h \\ -Z_h & -A_h^T \end{pmatrix}.$$

where $A_h = A + \mathcal{O}(h^p)$, $Z_h = Z + \mathcal{O}(h^p)$, and $\Lambda_h = BS^{-1}B^T + \mathcal{O}(h)$ is not necessarily positive definite for small h:

$$\Lambda_h = \begin{pmatrix} \Lambda_{1,1} & \Lambda_{2,1}^T \\ \Lambda_{2,1} & \Lambda_{2,2} \end{pmatrix} = \begin{pmatrix} \Lambda_{1,1,0} + h^p\Lambda_{1,1,p} + \ldots & h^p\Lambda_{2,1,p}^T + h^{p+1}\Lambda_{2,1,p+1}^T + \ldots \\ h^p\Lambda_{2,1,p} + h^{p+1}\Lambda_{2,1,p+1} + \ldots & h^p\Lambda_{2,2,p} + h^{p+1}\Lambda_{2,2,p+1} + \ldots \end{pmatrix}$$

Now the idea is to notice that matrix Λ_h depends analytically on h, because for linear differential equations, modified differential equations for backward error analysis always converge as series of h for small enough. Thus, for $h > 0$ small enough, it has constant rank r, with $m \le r \le n$. Without loss of generality, we assume $B = \begin{pmatrix} Id \\ 0 \end{pmatrix} \in \mathbb{R}^{n\times m}$

and $S = Id \in R^{m\times m}$. (Indeed, one can consider the change of coordinates for the control $\widetilde{u} = S^{1/2}u$ and $\widetilde{x} = Px$ where $P \in \mathbb{R}^{n\times n}$ is a left inverse of the maximum rank matrix $BS^{-1/2} \in \mathbb{R}^{n\times m}$). We search for perturbed matrices $B_h \in \mathbb{R}^{n\times r}$ and $S_h \in \mathbb{R}^{r\times r}$ (diagonal) of the form

$$B_h = \begin{pmatrix} B_1 & 0 \\ B_2 & B_3 \end{pmatrix} = \begin{pmatrix} Id & 0 \\ 0 & 0 \end{pmatrix} + \mathcal{O}(h^p) \qquad S_h = \begin{pmatrix} Id & 0 \\ 0 & h^p D \end{pmatrix}$$

where D is a diagonal matrix with coefficients ± 1 (thus $D = D^{-1}$), so that

$$\Lambda_h = B_h S_h^{-1} B_h^T = BS^{-1}B^T + \mathcal{O}(h^p).$$

This identity is equivalent to

$$\begin{pmatrix} \Lambda_{1,1} & \Lambda_{2,1}^T \\ \Lambda_{2,1} & \Lambda_{2,2} \end{pmatrix} = \begin{pmatrix} B_1 B_1^T & B_1 B_2^T \\ B_2 B_1^T & B_2 B_2^T + h^{-p} B_3 D B_3^T \end{pmatrix}$$

Matrix $\Lambda_{1,1}$ is a perturbation of Id, so it is still positive definite. Therefore, one can take for $B_1 = Id + \mathcal{O}(h^p)$ the Cholesky decomposition of the positive definite matrix $\Lambda_{1,1} = B_1 B_1^T$. For B_2, one can take $B_2 = \Lambda_{2,1}(B_1^T)^{-1} = \mathcal{O}(h^p)$. Finally, for $B_3 = \mathcal{O}(h^p)$, we need $B_3 D B_3^T = h^p(\Lambda_{2,2} - B_2 B_2^T) = \mathcal{O}(h^{2p})$. For h small enough, this matrix has constant rank, and this decomposition exists with a constant matrix D with coefficients ± 1. This concludes the proof. □

Conclusion

From the point of view of "geometric numerical integration" it is natural to use symplectic integrators when solving Hamiltonian differential equations. This has been proved very successful in many fields, in particular, in molecular dynamics simulations and in long-time integrations of planetary motion. For Hamiltonian systems arising form the Pontryagin maximum principle in optimal control our conclusion is the following.

Due to the fact that one is concerned with boundary value problems, long-term integration is not an issue. For integrations over short intervals, e.g. half a period of the motion of a planet, there is no real advantage of using a symplectic method. The solutions of problems of Sects. 4.4 and 4.5 neither show a periodic or quasi-periodic behavior nor an ergodic behavior like in molecular dynamics simulations. Therefore, no improvement of symplectic integrators can be expected, which is confirmed by numerical experiments.

For special problems, typically in low dimension and in situations where the Hamiltonian is close to a critical value (Sect. 4.2), the structure preservation of symplectic integrators is very important and symplectic methods can be much more efficient than non-symplectic ones. Indeed, the theory of backward error analysis allows one to prove that the numerical solution of a symplectic method has the same qualitative behavior as the exact flow. For example, in the integrable Martinet case, where the exact solution is periodic, the numerical solution remains exponentially close to a periodic orbit, which explains the excellent results, even on a relatively short interval of integration (a few periods). For non symplectic integrators, this structure is destroyed in general. We have shown that backward error analysis is possible for linear quadratic problems, involving the modified control problem which are stationary point control problems, and not necessarily optimal control problems (it depends on the signs of the eigenvalues of matrix Λ_h in the proof

of Theorem 4.6.5). It is not clear whether this result can be extended also to non-linear problems

$$\begin{cases} \text{Min } \int_0^1 g(x,u)dt \\ \dot{x} = f(x,u) \\ x(0) \text{ given.} \end{cases}$$

Chapter 5

Splitting methods based on modified potentials

Consider a system of ordinary differential equations

$$\dot{x} = f(x), \quad x(0) = x_0 \in \mathbb{R}^d$$

where the vector field $f(x)$ is split as $f(x) = A(x) + B(x)$, and the flows of A and B can be approximated efficiently either exactly or with high-accuracy. Then, a standard approach for this problem is to consider splitting methods of the form

$$e^{a_m hA} e^{b_m hB} e^{a_{m-1} hA} e^{b_{m-1} hB} \cdots e^{a_1 hA} e^{b_1 hB} \tag{5.1}$$

where e^{hA} and e^{hB} denote the flows associated to A and B. In the context of geometric integration, this kind of integrator is of great interest because it preserve qualitative properties of the exact solution. If A and B are two Hamiltonian vector fields, then all the flows $e^{a_i hA}$ and $e^{b_i hB}$ are symplectic, and the splitting method is symplectic as a composition of symplectic flows, which guaranties the well conservation of energy over exponentially long times.

A simple way to obtain high-order splitting methods is to use composition methods. One can consider as basic method the Strang splitting $\Phi_h = e^{\frac{h}{2}B} e^{hA} e^{\frac{h}{2}B}$ of order 2, and then the associated standard composition methods of the form

$$\Phi_{\gamma_s h} \circ \ldots \circ \Phi_{\gamma_1 h} \tag{5.2}$$

of orders 4,6,8, involving respectively $s = 5,9$ and 17 compositions of the basic method, as described in [HLW06] and as already used in Sect. 3.2. We shall consider these compositions methods in the numerical experiments as references for comparison.

A significant improvement, to reduce the number of compositions in (5.1), and thus the computational cost, is to consider processed methods. In order to reduce the number of evaluations per step in the integration, the idea of processing, first introduced by Butcher [But69] in the context of Runge-Kutta methods, is to consider a composition of the form

$$e^P e^{hK} e^{-P}.$$

where e^{hK} is called the Kernel and should be cheap, and the order of $e^P e^{hK} e^{-P}$, called effective order, is higher than that of e^{hK}. Using a constant stepsize h, after N steps, we obtain $e^P (e^{hK})^N e^{-P}$. At first, we apply the processor (or corrector) e^{-P}, then e^{hK} once per step, and the postprocessor e^P is evaluated only when output is desired. The

Kernel and the processor are taken as compositions of the flows associated to A, B which makes the splitting method symplectic for Hamiltonian vector fields. A general analysis of symplectic splitting methods with processing in given in [BCR99].

In practice, the main tool for the derivation of order conditions for splitting methods is the Baker-Campell-Hausdorff (BCH) formula (see e.g. [HLW06, Sect. III.4.2]) which implies that the error for these methods is formally a linear combination of Lie bracket terms in the Lie-algebra generated by the vector fields A and B. For instance, for the Lie Trotter splitting $e^{hA}e^{hB}$, it yields

$$\exp(hA)\exp(hB) = \exp(hA + hB + h^2 S_2 + h^3 S_3 + h^4 S_4 + \ldots) \qquad (5.3)$$

$$\begin{aligned} S_2 &= \frac{1}{2}[A,B] \\ S_3 &= \frac{1}{12}[A,[A,B]] + \frac{1}{12}[B,[B,A]] \\ S_4 &= \frac{1}{24}[A,[B,[B,A]]] \end{aligned}$$

To derive the order conditions for a splitting method (5.1) with unknown coefficients $a_i, b_i, c_i, \ldots$, one can apply repeatedly the BCH formula, this yields a system of polynomial equations in the coefficients of the splitting methods which can be solved numerically.

Now, assume that the vector field B is a small perturbation of vector field A, i.e.

$$B = \mathcal{O}(\varepsilon)$$

where ε is a small parameter. Then, we obtain that Lie brackets involving few B's are higher than those with many B's and should be canceled in priority to reduce the error of the method. For instance, $[A,[A,B]] = \mathcal{O}(\varepsilon)$ is dominant compared to $[B,[B,A]] = \mathcal{O}(\varepsilon^2)$. The idea of processing has been applied to the symplectic integration of near-integrable Hamiltonian systems in [WHT96, McL96].

An other improvement is possible for following special class of problems:

$$\begin{aligned} \dot{p} &= f^A(p,q) \quad + f^B(q) \\ \dot{q} &= g^A(p,q) \end{aligned} \qquad (5.4)$$

where $x = (p,q)^T$ (p, q not necessarily of the same dimension) so that the flow associated to B can be computed explicitly:

$$p_{n+1} = p_n + h f^B(q_n), \qquad q_{n+1} = q_n$$

and we assume the flow associated to A can be computed with high-accuracy efficiently. Consider the Lie Bracket $C = [B,[B,A]]$. A straightforward computation of vector field Lie Brackets shows

$$C = \begin{pmatrix} f^C \\ 0 \end{pmatrix} \quad \text{where} \quad f^C = f^A_{pp}(f^B, f^B) - 2 f^B_q g^A_p f^B.$$

Now, we assume that the derivatives f^A_{pp} and g^A_p are independent of p. This means that $f^A(p,q)$ is (at most) quadratic in p and $g^A_p = 0$ is (at most) linear in p. Then, the flow of C can be integrated explicitly (similarly to f^B), and the flows of B and C commute ($[B,C] = 0$). It thus makes sense to compute the flow $e^{bhB+\beta h^3 C}$ associated to the vector field $bhF_B + \beta h^3 F_C$. Moreover, the brackets $[B,\ldots[B,[B,A]]]$ (with at least 3 B's) all vanishes, which reduces significantly the number of order conditions. Notice that for Hamiltonian

vector fields A and B, the bracket $C = [B,[B,A]]$ is also a Hamiltonian vector field. and the corresponding Hamiltonian is given simply by the Poisson bracket $\{B,\{B,A\}$ (see e.g. [HLW06] for details). A general study in this context is conducted in [BCR01], and also [BCR00] in the situation of near integrable Hamiltonian systems, i.e. $B = \mathcal{O}(\varepsilon)$. Here, in addition to the term $C = [B,[B,A]]$ they also consider additional terms like $[B,[B,[A,[A,B]]]]$ as well as higher order terms.

In [BC05], they give an elegant elementary proof of the fact that any splitting method (or processed splitting method) with real coefficients of order greater or equal to three must have a negative coefficient for A and also for B, as first shown in [GK96]. However, they highlight that this is not the case for splitting methods involving more general Lie brackets since

$$e^{\frac{h}{6}B}e^{\frac{h}{2}A}e^{\frac{2h}{3}B-\frac{h^3}{72}C}e^{\frac{h}{2}A}e^{\frac{h}{6}B}$$

has order 4 [Kos94], and the integrator

$$e^{\frac{h}{2}A}e^{hB-\frac{h^3}{24}C}e^{\frac{h}{2}A} \tag{5.5}$$

has effective order 4 [Row91, TI86], with positive coefficients a_i, b_i.

These methods are called 'Runge-Kutta Nyström methods' in [BCR01] because they were introduced in the context of second order differential equations $\ddot{x} = f(x)$. However, the class of problems (5.4) includes not only second order differential equations like the N-body problems in Jacobi coordinates as studied in [WH91]. The main contribution of this chapter is to show that this method can also be successfully applied to asymmetric rigid body problems with an external potential. We also build a new processor for the Takahashi–Imada method (a modification of the Störmer-Verlet method), to achieve order $\mathcal{O}(h^{10}\varepsilon + h^4\varepsilon^2)$ (Sect. 5.2). Our numerical experiments indicate that this method is very efficient for small ε when the cost of evaluating the vector field $C = [B,[B,A]]$ together with B is small compared to the cost of evaluating of A and B alone.

5.1 Examples of splitting methods

For the numerical solution of problems of the form (5.4), consider a splitting method of the form

$$e^{a_1hA}e^{b_1hB}\dots e^{a_nhA}e^{b_nhB} = e^{hK}$$

or more generally,

$$e^{a_1hA}e^{b_1hB+c_1h^3C}\dots e^{a_nhA}e^{b_nhB+c_nh^3C} = e^{hK} \tag{5.6}$$

where $C = [B,[B,A]]$. Repeatedly applying the BCH formula (5.3), we obtain that the method e^{hK} is the exact flow of a linear combination in $L(A,B)$ of nested Lie-Brackets, where $L(A,B)$ is the free Lie algebra generated by the Lie operators associated to the vector fields A and B:

$$\begin{aligned} K &= k_{1,1}A + k_{1,2}B + k_{2,1}h[A,B] + k_{3,1}h^2[A,[A,B]] + k_{3,2}h^2[B,[B,A]] + \dots \\ &= k_{1,1}A + k_{1,2}B + \sum_{i=2}^{\infty} h^{i-1}\sum_{j=1}^{d(i)} k_{i,j}E_{i,j} \end{aligned} \tag{5.7}$$

where $k_{1,1} = a_1 + \dots a_n = 1$ and $k_{1,2} = b_1 + \dots b_n = 1$ for consistent methods. Here, the set $\{E_{i,j}\}_{j=1}^{d(i)}$ is a basis of the subspace $L^n(A,B)$ generated by the independent brackets

of order n, its dimension $d(n)$ being $2, 1, 2, 3, 6$ for $n = 1 \dots 5$ (see e.g. [BCR99]). The splitting method has order p if and only if $K = A + B + \mathcal{O}(h^p)$, i.e. $k_{i,j}E_{i,j} = 0$ for all (i, j) with $i = 2 \dots p$. Notice that for symmetric splitting methods, $k_{i,j} = 0$ for even i, i.e the formal expansion of K in (5.3) is in even powers of h.

In our situation, for the special class of problems (5.4), all Lie brackets of the from $[B, [\dots [B, A]]]$ with at least three B's vanish, so the dimension of $L^n(A, B)$ (and thus the number of order conditions) is reduced to $d'(n) = 2, 1, 2, 2, 4$ for orders $n = 1 \dots 5$.

5.1.1 Splitting methods without processing

A general study of splitting methods with processing is conducted in [BCR01], and also [BCR00] in the context of near integrable Hamiltonian systems, i.e. $B = \mathcal{O}(\varepsilon)$.

Example: a method of order 6 Solving order conditions (using MAPLE), we found that the minimal number of evaluations of A and B to get a symmetric method of order 6 without processing, and involving terms of the form e^{ahA} and e^{bhB+ch^3C} is 4, and the method is unique:

$$e^{b_1hB}e^{ha_1A}e^{b_2hB+\beta_2h^3C}e^{a_2hA}e^{b_3hB+\beta_3h^3C}e^{a_2hA}e^{b_2hB+\beta_2h^3C}e^{a_1hA}e^{b_1hB} \tag{5.8}$$

and surprisingly, $\beta_1 = \beta_5 = 0$. In fact, this method was already derived in [OMF03].

Since $B = \mathcal{O}(\varepsilon)$, the dominant terms of size $\mathcal{O}(\varepsilon)$ in (5.7) are the brackets $E_{2,1} = [A, B]$, $E_{3,1} = [A, [A, B]]$, $E_{i,1} = [A, E_{i-1}] = \mathcal{O}(\varepsilon)$ with only one A. Cancelling these terms in (5.7) allows to increase the accuracy for small ε.

Example: a new method of order $\mathcal{O}(h^8\varepsilon + h^6\varepsilon^2)$ Next we searched for symmetric methods of order $\mathcal{O}(h^8\varepsilon + h^6\varepsilon^2)$. The idea is to vanish the coefficient $k_{7,1}$ of the bracket $E_{7,1} = [A, [A, [A, [A, [A, [A, B]]]]]]$ in the BCH formula. Numerically solving the order conditions, we found a dozen of solutions. The one with minimal $\sum_i |a_i|$ and positive b_i is

$$\begin{aligned} &e^{b_1hB+\beta_1h^3C}e^{ha_1A}e^{b_2hB+\beta_2h^3C}e^{ha_2A}e^{b_3hB+\beta_3h^3C}e^{ha_3A} \\ &e^{b_3hB+\beta_3h^3C}e^{ha_2A}e^{b_2hB+\beta_2h^3C}e^{ha_1A}e^{b_1hB+\beta_1h^3C} \end{aligned} \tag{5.9}$$

$$\begin{aligned} a_1 &= a_5 = 0.168735950563437422 & b_1 &= b_5 = 0.049086460976116245 \\ a_2 &= a_4 = 0.377851589220928304 & b_2 &= b_4 = 0.264177609888976700 \\ a_3 &= -0.093175079568731453 & b_3 &= 0.186735929134907054 \\ \beta_1 &= \beta_5 = 0.00166171386175851684 & \beta_2 &= \beta_4 = -0.00461492847770001641 \\ \beta_3 &= 0.0000446959494108217 \end{aligned}$$

Notice that roundoff errors are proportional to $\max(\sum_i |a_i|, \sum_i |b_i|) \geq 1$. Here, $\sum_i |a_i| \approx 1.093$ is quite small compared to standard splitting methods (5.1).

More generally, the next proposition gives a general formula to compute the coefficients $k_{i,1}$ of the brackets $E_{i,1}$ involving only one A. It is very similar to Proposition 2 in [LR01].

Proposition 5.1.1 *Consider a splitting of the form (5.6). Then the coefficients $k_{i,1}$, $i = 1, 2, 3, \dots$ of the brackets $E_{11} = A, E_{21} = [A, B]$, $E_{i,1} = [A, E_{i-1,1}]$ ($i \geq 3$), in the BCH expansion (5.7) are given by the series expansion*

$$\frac{\gamma_n t}{e^{\gamma_n t} - 1} \sum_{k=1}^{n} b_k e^{\gamma_k t} = \sum_{j \geq 0} k_{j+1,1} t^j$$

where $\gamma_k = a_1 + \dots + a_k$.

Proof. This is a consequence of the identity

$$e^{a_1 U} e^{b_1 V} \dots e^{a_n U} e^{b_n V} = e^{W} \qquad W = \gamma_n U + \frac{ad(\gamma_n U)}{e^{ad(\gamma_n U)} - 1} \sum_{k=1}^{n} b_k e^{ad(\gamma_k U)} V.$$

where $ad(X)(Y) = [X, Y]$ (see [LR01, Prop. 2] for details). □

If the method splitting method is consistent then $\gamma_n = 1$ and the condition $k_{i,1} = 0$, $i = 2 \dots p$ write

$$\frac{t}{e^t - 1} \sum_{k=1}^{n} b_k e^{\gamma_k t} = 1 + \mathcal{O}(h^{p+1})$$

or equivalently, as shown in [McL95],

$$\sum_{k=1}^{n} b_k e^{\gamma_k t} = \int_0^1 e^{tx} dx + \mathcal{O}(h^{p+1}).$$

The solution of this problem which yields the highest order is known classically as the Gauss integration formula. This permits the construction in [McL95] of a family of symmetric splittings of the form (5.1) of orders $\mathcal{O}(h^p\varepsilon + h^2\varepsilon^2)$ with positive coefficients a_i, b_i. However, for small stepsize h, the error term $\mathcal{O}(h^2\varepsilon^2)$ becomes dominant. Therefore, these integrators are improved in [LR01] to order $\mathcal{O}(h^p\varepsilon + h^4\varepsilon^2)$ in the context of the N-body Kepler problem, by performing a corrector term $e^{\beta h C}$ with $C = [B, [B, A]]$ at each step.

Order $\mathcal{O}(h^{10}\varepsilon + h^4\varepsilon^2)$ methods without processing For instance, the method denoted $SBAB_{10}$ of order $\mathcal{O}(h^{10}\varepsilon + h^4\varepsilon^2)$ proposed in [LR01] writes

$$e^{b_{11} hB + h^3 \beta C} e^{a_{10} hA} e^{b_{10} hB} \dots e^{a_1 hA} e^{b_1 hB + \beta h^3 C} \tag{5.10}$$

where the coefficients a_i, b_i are obtained from the weights of the Gauss-Lobatto formula. It is the unique method of the form (5.10) with positive coefficients a_i and b_j (see [LR01, Table I]).

5.1.2 Splitting methods with processing

We give here examples of splitting methods with processing. Our numerical experiments (see further) indicate that processed splitting methods are more efficient in general than unprocessed methods.

Order $\mathcal{O}(h^6\varepsilon)$ methods with processing In [BCR01] families of 6 and 8-order splitting method with processing are introduced. The method $6 : ABA - 3, 6; 3$ of order 6 has the form

$$e^{P} e^{a_1 hA} e^{b_1 hB} e^{a_2 hA} e^{b_2 hB + c_2 h^3 C} e^{a_2 hA} e^{a_1 hA} e^{b_1 hB} e^{-P} \tag{5.11}$$

where e^P is the processor composed of explicitly computable flows involving terms of the the type as the Kernel. They also investigate the use of higher order terms like $D = [B, B, A, A, B]$, $E = [B, A, B, B, A, A, B]$, $F = [B, B, B, A, A, A, B]$ and build the method denoted $(6 : ABA - 3, 6; 7)$ of order 6

$$e^{P} e^{a_1 hA} e^{b_1 hB} e^{a_2 hA} e^{b_2 hB + c_2 h^3 C + e_2 h^7 E + f_2 h^7 F} e^{a_2 hA} e^{a_1 hA} e^{b_1 hB} e^{-P} \tag{5.12}$$

where e^P is a processor (see [BCR01, Table 2] for the coefficients of these two methods).

Order $\mathcal{O}(h^6\varepsilon + h^4\varepsilon^2)$ method with processing In [BCR00], new processed splitting methods of orders $3, 4, 5$ for perturbed Hamiltonian systems are introduced. For instance, the first method in [BCR00, Table V] is a processor for method (5.5) of order $\mathcal{O}(h^6\varepsilon+h^4\varepsilon^2)$.

The next proposition states that by using the processing technique, it is always possible to cancel the terms $k_{i,j}E_{i,1}$ of size $\mathcal{O}(\varepsilon)$, ie. the brackets $[A, \dots A, B]$. This result, true for any splitting method, is originally due to [WHT96].

Proposition 5.1.2 *Consider a splitting method e^{hK} of the form (5.7) and assume it has order $\mathcal{O}(h^{\alpha_1}\varepsilon + h^{\alpha_2}\varepsilon^2 + \ldots + h^{\alpha_k}\varepsilon^k)$ for $B = \mathcal{O}(\varepsilon)$. Consider a processor e^P of the form*

$$P = \sum_{i=2}^{\infty} h^i \sum_{j=1}^{d(i)} p_{i,j}E_{i,j}.$$

Then, for $p_{i-1,1} = k_{i,1}, i = 2 \dots p$, the processed method $e^P e^{hK} e^{-P}$ has order $\mathcal{O}(h^p\varepsilon + h^{\alpha'_2}\varepsilon^2 + \ldots + h^{\alpha'_k}\varepsilon^k)$.

Proof. The idea is to notice that the coefficient of $E_{i,1}$ in the BCH expansion of $e^P e^{hK} e^{-P}$ is simply $k_{i,1} - p_{i-1,1}$. □

5.2 Processed Takahashi–Imada splitting method

Consider the Takahashi–Imada method of effective order 4, as introduced in [Row91, TI86],

$$e^{\frac{h}{2}B - \frac{h^3}{48}C} e^{hA} e^{\frac{h}{2}B - \frac{h^3}{48}C}.$$

Notice that in the particular case of a Hamiltonian system with Hamiltonian $H(p, q) = \frac{1}{2}p^T p + U(q)$, this method simply writes

$$\begin{aligned} p_{n+1/2} &= p_n - \frac{h}{2}\big(I - \beta h^2 \nabla^2 U(q_n)\big)\nabla U(q_n) \\ q_{n+1} &= q_n + h p_{n+1/2} \\ p_{n+1} &= p_{n+1/2} - \frac{h}{2}\big(I - \beta h^2 \nabla^2 U(q_{n+1})\big)\nabla U(q_{n+1}) \end{aligned}$$

with $\beta = 1/12$. It can be interpreted as the Störmer–Verlet method applied with the modified potential $\widetilde{U} = U - \frac{\beta}{2}h^2\|\nabla U\|^2$.

Using the techniques presented in the previous section, we present in this section the construction of a family of processors to achieve effective orders $\mathcal{O}(h^p\varepsilon + h^4\varepsilon^2)$ for arbitrary p.

For the Takahashi–Imada method, Proposition 5.1.1 with $n = 2$ yields the series

$$\frac{t}{1 - e^t}\big(\frac{1}{2}e^0 + \frac{1}{2}e^t\big) = \frac{t}{2} + \frac{t}{e^t - 1} = 1 + \sum_{j=2}^{\infty} \frac{B_j}{j!}t^j = 1 + \frac{t^2}{12} - \frac{t^4}{720} + \dots$$

where the $B'_j s$ are the Bernoulli numbers. Now, we consider a processor of the same form as in Proposition 5.1.1, with $\gamma_n = a_1 + \ldots + a_n = 0$ and $b_1 + \ldots + b_n = 0$, so the term $\frac{\gamma_n t}{e^{\gamma_n t} - 1}$ has to be replaced by 1. To achieve order p, the condition $p_{i-1,1} = k_{i,1}, i = 2, \dots p$ is equivalent to

$$1 + t\sum_{k=1}^{n} b_k e^{\gamma_k t} = \frac{t}{2} + \frac{t}{e^t - 1} + \mathcal{O}(t^p)$$

or also

$$\sum_{k=1}^{n-1} b_k(\gamma_k)^{i-1} = \frac{B_i}{i}, \qquad i = 2, \dots p-1$$

Notice that for arbitrary distinct γ_i's, and $n = p-1$, this system of equation is a square Vandermonde linear problem for coefficients b_k's which possesses a unique solution.

One solution of this system with $p = 10$ is given by the following new processor for the Takahashi–Imada method. It yields a processed method of order $\mathcal{O}(h^{10}\varepsilon + h^4\varepsilon^2)$.

$$\begin{aligned}
e^P &= e^{a_1 hA} e^{b_1 hB + c_1 h^3 C} e^{a_2 hA} e^{b_2 hB} \dots e^{a_8 hA} e^{b_8 hB} e^{a_9 hA} e^{c_2 h^3 C} \\
a_1 &= a_9 = -5/4,\ a_2 = a_8 = 1/8,\ a_3 = a_7 = 1/4,\ a_4 = a_6 = 1/2,\ a_5 = 3/4 \\
d_1 &= -d_8 = 0.041139583138698574 \qquad d_2 = -d_7 = -0.108267614767371103 \\
d_3 &= -d_6 = 0.135191882308478947 \qquad d_4 = -d_5 = -0.238887935991110594 \\
c_1 &= -1/900 \qquad c_2 = -0.014597270750786141
\end{aligned}$$

Moreover, c_1, c_2 are chosen so that the brackets $[A,B,B,A]$ and $[A,A,B,B,A]$ of size $\mathcal{O}(\epsilon^2)$ in the BCH expansion of $e^P e^{hK} e^{-P}$ vanishes. Thus, the dominant error term in the local error is given by

$$e^P e^{hK} e^{-P} - e^{hA+hB} = \frac{h^5}{1440}[B,A,A,A,B] + \mathcal{O}(h^{11}\varepsilon + h^5\varepsilon^3).$$

Notice that this dominant error term cannot be reduced by processing, because the coefficient for $[B,A,A,A,B]$ equals $1/360 - p_{21}/12 + p_{21}^2/2 - p_{41} = \frac{1}{1440}$, where parameters $p_{21} = 1/12$ and $p_{41} = -1/720$ are already used to cancel the brackets E_{31}, E_{51}.

Remark 5.2.1 Similarly to the simplified Takahashi–Imada method proposed in [WHT96] and as studied in [HMS08], a possibility to avoid the derivative evaluation $f_q^B(q)$ of the vector fields $f^B(q)$ is to replace $\big(Id - 2\beta h^2 g_p^A f_q^B(q)\big) f^B(q)$ by $f\big(q - 2\beta h^2 g_p^A f_q(q)\big)$ and thus consider the approximation

$$f^B(q) + \beta h^2 f^C(q) = f^B(q - 2\beta h^2 g_p^A f_q^B(q)) + \beta h^2 f_{pp}^A\big(f^B(q), f^B(q)\big) + \mathcal{O}(\varepsilon^3 h^4)$$

This approximation of $f^B(q) + \beta h^2 f^C(q)$ requires only two evaluations of $f^B(q)$. We give here some consequences of the analysis in [HMS08] in our context of perturbed Hamiltonian system integration. This simplification used for the Kernel of the processed Takahashi–Imada integrator still yields a symmetric and reversible integrator, with effective order $\mathcal{O}(h^p\varepsilon + h^4\varepsilon^2)$, and it is still volume preserving (because a map $(p,q) \mapsto (p + a(q), q)$ is always volume preserving), However, as shown in [HMS08, Sect. 4], the modified method it is no longer symplectic. In general, without any particular assumption on the potential, this adds a linear drift in the energy, of size $\mathcal{O}(t\varepsilon^3 h^4)$.

Notice that the considered simplification can be successfully applied to the processors e^P and e^{-P} alone, and the energy error will remain bounded with size $\mathcal{O}(h^p\varepsilon + \varepsilon^2 h^4)$ on exponentially long-time intervals, because the processed integrator is still conjugate to a symplectic integrator.

5.3 Applications to mechanical problems

5.3.1 The N-body problem in Jacobi coordinates.

We consider the N-body problem with Hamiltonian,

$$H(p,q) = \frac{1}{2}\sum_{i=0}^{N} \frac{1}{m_i} p_i^T p_i - G\sum_{i=1}^{N}\sum_{j=0}^{i-1} \frac{m_i m_0}{\|q_i - q_0\|}$$

As proposed in [WH91], the Hamiltonian can be split as $H = H^A + H^B$ where H^A represents the integrable Keplerian part, corresponding to the interactions Sun-planets, and the B part is a small perturbation, corresponding to the interactions between the planets. The use of Jacobi coordinates makes H^B depend only on the positions of the planets (see e.g. [Mey99, Sect. 3.5] for a presentation of Jacobi coordinates), and thus easy to integrate (explicit).

The integrable part A corresponds to $(N-1)$ decoupled two-body Kepler problems which can be solved exactly using the Kustaanheimo-Stiefel transformation (1965) (see Appendix C for details). The B part can be integrated explicitly (see below).

We detail the integration for the 3 body problem Sun-Jupiter-Saturn. Let m_0, m_1, m_2, q_0, q_1, q_2 and p_0, p_1, p_2 denote respectively the masses, the positions and momenta of the Sun, Jupiter and Saturn. We assume $m_0 q_0 + m_1 q_1 + m_2 q_2 = 0$ (center of mass is at the origin) and $p_0 + p_1 + p_2 = 0$ (total momentum).

Introducing the Jacobi coordinates Q_1, Q_2, and P_1, P_2,

$$\begin{aligned} Q_1 &= q_1 - q_0 \qquad Q_2 = q_2 - (\alpha_0 q_0 + \alpha_1 q_1) \\ \frac{P_1}{M_1} &= \frac{p_1}{m_1} - \frac{p_0}{m_0} \qquad \frac{P_2}{M_2} = \frac{p_2}{m_2} - \frac{p_0 + p_1}{m_0 + m_1} \end{aligned}$$

with scalars $\alpha_0 = \frac{m_0}{m_0+m_1}$, $\alpha_1 = \frac{m_1}{m_0+m_1}$ and masses $M_1 = \frac{m_0 m_1}{m_0+m_1}$, $M_2 = \frac{(m_0+m_1)m_2}{m_0+m_1+m_2}$, then the Hamiltonian becomes $H = H^A + H^B$, where

$$\begin{aligned} H^A &= \Big(\frac{P_1^2}{2M_1} - \frac{G m_1 m_0}{\|Q_1\|}\Big) + \Big(\frac{P_2^2}{2M_2} - \frac{G m_2 m_0}{\|Q_2\|}\Big) \\ H^B &= G m_2 m_0 \Big(\frac{1}{\|Q_2\|} - \frac{1}{\|Q_2 + \alpha_1 Q_1\|}\Big) - \frac{G m_1 m_2}{\|Q_2 - \alpha_0 Q_1\|} \end{aligned}$$

Since the Sun is the heaviest body, we have $\alpha_1 = \mathcal{O}(\varepsilon)$ and $B = \mathcal{O}(\varepsilon)$. For the Solar system, we have approximately $\varepsilon \approx 10^{-3}$.

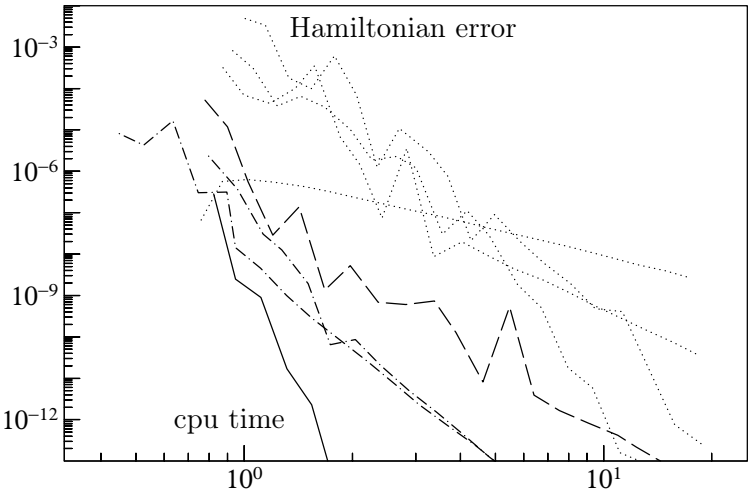


Figure 5.1: Three body problem Sun-Jupiter-Saturn. Comparison of Hamiltonian error versus cpu time for various splitting methods. Without processing: Strang splitting and composition methods (5.2) of orders $\mathcal{O}(h^p\varepsilon)$, $p = 2, 4, 6, 8$ (dotted lines), method (5.10) order $\mathcal{O}(h^{10}\varepsilon + h^4\varepsilon^2)$ (dashed lines). With processing: methods (5.5) and (5.11) of orders $\mathcal{O}(h^6\varepsilon + h^4\varepsilon^2)$ and $\mathcal{O}(h^6\varepsilon)$ (dashed-dotted lines), new method (Sect. 5.2) of order $\mathcal{O}(h^{10}\varepsilon + h^4\varepsilon^2)$ (solid line).

Numerical experiment We consider the 3-body problem Sun-Jupiter-Saturn with the initial data given in [HLW06, Table I.2.2]. We integrate on the interval of days $[0, 10^8]$ and many stepsizes (e.g. $h = 400 \cdot 2^{-j/4}, j = 1 \dots 20$ for the Strang splitting). In Figure 5.1, we plot the relative Hamiltonian error as a function of the cpu times of computation. We observe that the preprocessed methods are the most efficient. For the processed Takahashi–Imada method of order $\mathcal{O}(h^{10}\varepsilon + h^4\varepsilon^2)$ with observe a line with slop $p = 10$. This indicates that the second error term of size $\mathcal{O}(h^4\varepsilon^2)$ is negligible in this numerical experiment.

5.3.2 The motion of a rigid body with an external potential.

In the presence of an external potential $V(Q)$, the motion of a rigid rigid body, relative to a fixed coordinate system, is determined by a Hamiltonian system constrained to the Lie group $SO(3)$ and the Hamiltonian is

$$H(y, Q) = \frac{1}{2}\Big(\frac{y_1^2}{I_1} + \frac{y_1^2}{I_2} + \frac{y_1^2}{I_3}\Big) + V(Q)$$

where the constants I_1, I_2, I_3 are the moments of inertia, $y(t) = (y_1(t), y_2(t), y_3(t))$ is the angular momentum and $Q(t)$ is the orthogonal matrix that gives the orientation of the rigid body in the fixed frame (see Chapter 3). The equations of motion are

$$\dot{y} = \widehat{y}\, I^{-1} y + f(Q), \qquad \dot{Q} = Q\, \widehat{I^{-1} y},$$

where $I = \mathrm{diag}\,(I_1, I_2, I_3)$ and the hatmap notation $\widehat{y}$ is defined already in (3.2),

$$y = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix}, \qquad \widehat{y} = \begin{pmatrix} 0 & -y_3 & y_2 \\ y_3 & 0 & -y_1 \\ -y_2 & y_1 & 0 \end{pmatrix}.$$

The torque is given by

$$f(Q) = -\mathrm{rot}\,(Q^T \frac{\partial V}{\partial Q})$$

where

$$\frac{\partial V}{\partial Q} = \left(\frac{\partial V}{\partial Q_{ij}}\right)_{i,j=1\dots3} \in \mathbb{R}^{3\times3}$$

is the Jacobian matrix. Here, for all 3×3 matrix M ,the vector $\mathrm{rot}\, M$ in $\mathbb{R}^3$ is defined by

$$\widehat{\mathrm{rot}\, M} = M - M^T. \tag{5.13}$$

A standard approach to solve this problem is to split it into the free rigid body motion

$$A: \qquad \dot{y} = \widehat{y}\, I^{-1} y, \qquad \dot{Q} = Q\, \widehat{I^{-1} y},$$

which can be solved using e.g. the high-order preprocessed DMV, plus a torqued motion,

$$B: \qquad \dot{y} = f(Q), \qquad \dot{Q} = 0,$$

It can be easily verified that this problem is in the form (5.4). A direct computation shows

$$f^C(Q) = 2\widehat{f^B(Q)} I^{-1} f^B(Q) - 2\frac{\partial f^B(Q)}{\partial Q}(Q\widehat{I^{-1} f^B(Q)})$$

We consider two different external potentials.

5.3.2.1 Asymmetric heavy top

We consider an asymmetric rigid body and we assume the center of gravity to be $(0,0,1)$ in the body frame, and that the third coordinate is the stationary frame is vertical. The potential energy for gravity is given by (see e.g. [HLW06, Sect. VII.5.3.(II)])

$$V(Q) = \varepsilon Q_{33}.$$

The torques corresponding to the brackets $C = [B,[B,A]]$, D, ..., reduce to polynomials in Q_{31}, Q_{32}, Q_{33},

$$\begin{aligned} f^B(Q) &= \varepsilon(Q_{32}, -Q_{31}, 0)^T \\ f^C(Q) &= \varepsilon^2(\frac{Q_{32}Q_{33}}{I_1}, -\frac{Q_{31}Q_{33}}{I_2}, 0)^T \\ f^D(Q) &= 2\varepsilon^3 \begin{pmatrix} +\frac{2}{I_1^2}Q_{32}Q_{33}^2 - \frac{1}{I_1^2}Q_{32}^3 - \frac{1}{I_2^2}Q_{31}^2Q_{32} \\ -\frac{2}{I_2^2}Q_{31}Q_{33}^2 + \frac{1}{I_2^2}Q_{31}^3 + \frac{1}{I_1^2}Q_{31}Q_{32}^2 \\ (\frac{2}{I_2^2} - \frac{2}{I_1^2})Q_{31}Q_{32}Q_{33} \end{pmatrix} \end{aligned}$$

and similar formulas for $f^E(Q)$ and $f^F(Q)$. It requires to compute at each step the components Q_{31}, Q_{32}, Q_{33} from the quaternion $q = (q_0, q_1, q_2, q_3)$ representing the rotation matrix Q,

$$Q_{31} = -2q_0q_2 + 2q_1q_3 \qquad Q_{32} = 2q_0q_1 + 2q_2q_3 \qquad Q_{33} = q_0^2 - q_1^2 - q_2^2 + q_3^2.$$

All constants depending on the moments of inertia I_1, I_2, I_3 can be computed only once. Thus, the cost of evaluating altogether $f^B(Q), f^C(Q), f^D(Q), f^E(Q)$ is negligible compared to the resolution of the free rigid body part.

Numerical experiment In Figure 5.2 with consider a rigid body with moments of inertia $I_1 = 0.345, I_2 = 0.653, I_3 = 1.0$, which corresponds to the water molecule, and we integrate on the interval $[0, 100]$. The initial condition for the angular momentum is $y_1(0) = 1.8$, $y_2(0) = 0.4$, $y_3(0) = -0.9$ and we take $Q(0) = Id$.

We compare the relative Hamiltonian error versus the number of evaluations of the free rigid body part e^A (which clearly dominates the cost) for various sizes ε of the gravity torque. Again we observe that the processed methods are very efficient. For small ε, we observe for the processed Takahashi–Imada method of order $\mathcal{O}(h^{10}\varepsilon + h^4\varepsilon^2)$ (solide line) a line op slop $p = -10$ for large stepsize and a line of slop $p = -4$ for small stepsizes. Indeed, for $\varepsilon \to 0$, the error term $\mathcal{O}(h^4\varepsilon^2)$ becomes negligible compared to $\mathcal{O}(h^{10}\varepsilon)$, and the efficiency of the method gets better and better.

5.3.2.2 Motion of a satellite

We consider the simplified model describing the motion of a satellite is a circular orbit around the Earth considered in [Mit00, LLM06]. The numerical integration of this problem is discussed in [CFSZ08]. For this problem, the potential, due to the action of the earth on the satellite, is quadratic in Q,

$$V(Q) = \frac{\varepsilon}{2}(Q^Te_3)^T(IQ^Te_3) = \frac{\varepsilon}{2}\Big(I_1Q_{31}^2 + I_2Q_{32}^2 + I_3Q_{33}^2\Big)$$

where $e_3 = (0,0,1)^T$. The torques are

$$\begin{aligned} f^B(Q) &= \varepsilon x \times (Ix) \qquad \text{where } x = (Q^Te_3), \\ f^C(Q) &= 2f^B(Q) \times (I^{-1}f^B(Q)) + 2\varepsilon(x \times (Iy) + y \times (Ix)) \quad \text{where } y = (I^{-1}f^B) \times x, \end{aligned}$$

and similarly for $f^D(Q)$.

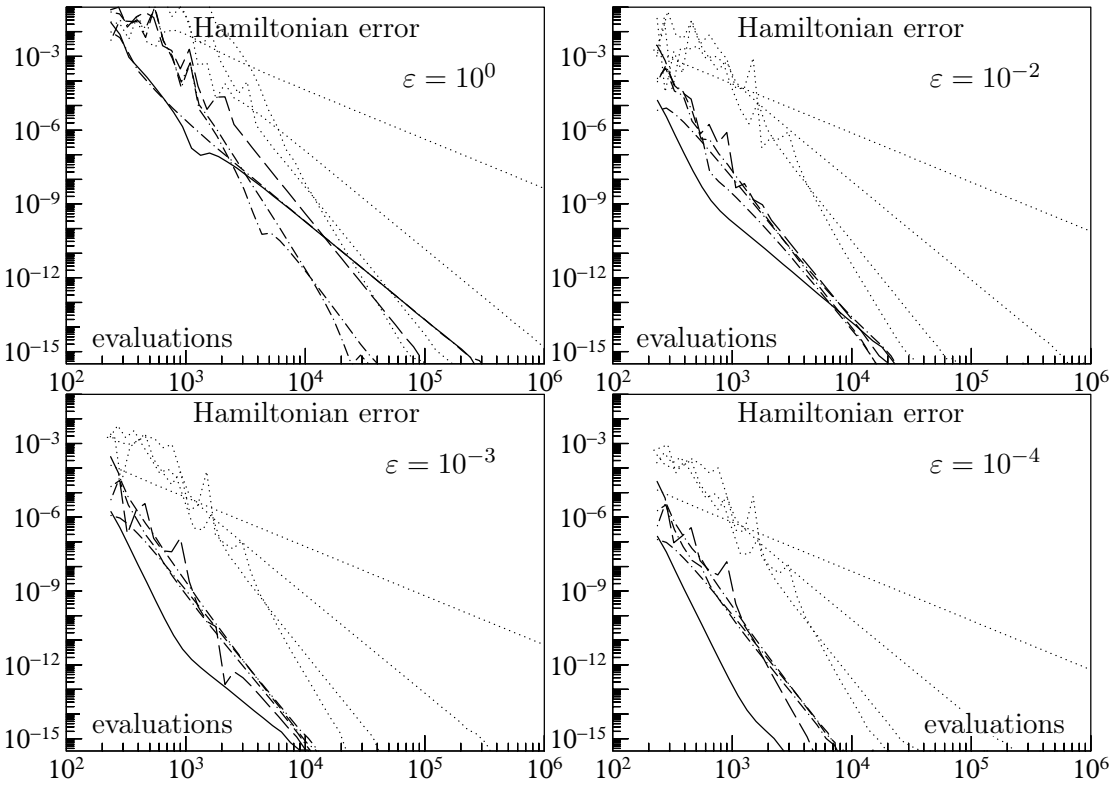


Figure 5.2: Asymmetric heavy top problem. Comparison of Hamiltonian error versus number of evaluations of e^{hA} and e^{hB} for various splitting methods. Without processing: Strang splitting and composition methods (5.2) of orders $\mathcal{O}(h^p\varepsilon)$, $p = 2, 4, 6, 8$ (dotted lines), method (5.9) order $\mathcal{O}(h^8\varepsilon + h^6\varepsilon^2)$ (dashed lines). With processing: methods (5.11) and (5.12) of order $\mathcal{O}(h^6\varepsilon)$ (dashed-dotted lines), new method (Sect. 5.2) of order $\mathcal{O}(h^{10}\varepsilon + h^4\varepsilon^2)$ (solid line).

Numerical experiment In Figure 5.3 with consider a rigid body with moments of inertia $I_1 = 0.6, I_2 = 0.8, I_3 = 1.0$, which corresponds to an asymmetric body, and we integrate on the interval $[0, 100]$. The initial condition for the angular momentum is $y_1(0) = 1.8$, $y_2(0) = 0.4$, $y_3(0) = -0.9$ and we take $Q(0) = Id$. The numerical are very similar as for the heavy top problem in the previous section. Again, the processed methods are the most efficient.

5.3.3 Molecular dynamics simulation: dipolar soft spheres

Splitting algorithm for multi-rigid body dynamics have been studied in [DLM97, BCF01]. We consider a molecular dynamics simulation where the molecule are described by dipolar soft sphere, as described in [DLM97, Appendix A]. This problem is also considered in [CFSZ08] as a numerical illustration for rigid body integrators. This model model can be used to study water and aqueous solutions, as water can be modeled by small dipoles.

We consider N molecules with mass m, positions $q_i \in \mathbb{R}^3$, orientations $Q_i \in SO(3)$, linear momenta $p_i \in \mathbb{R}^3$ and angular momenta $y_i \in \mathbb{R}^3$. The Hamiltonian for this problem is

$$H(y, p, Q, q) = T(y, p) + V(Q, q)$$

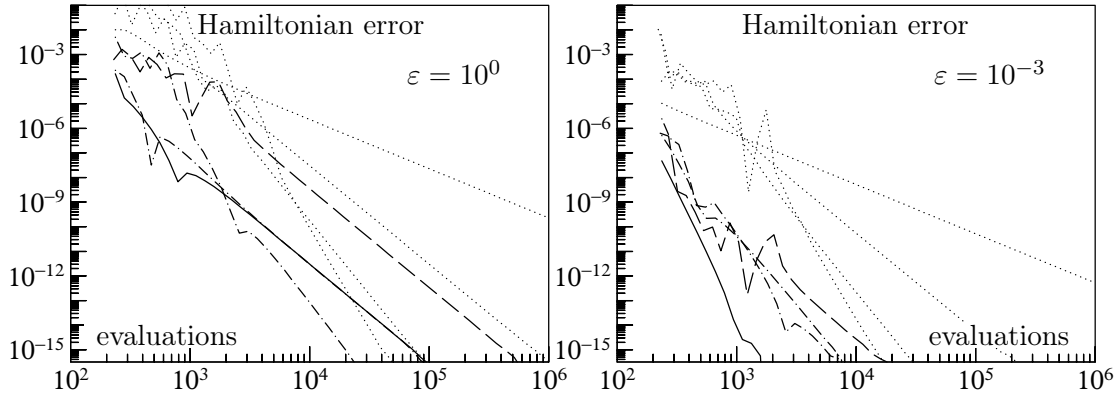


Figure 5.3: Motion of a satellite. Comparison of Hamiltonian error versus number of evaluations of e^{hA} and e^{hB} for various splitting methods. Without processing: Strang splitting and composition methods (5.2) of orders $\mathcal{O}(h^p\varepsilon)$, $p = 2, 4, 6, 8$ (dotted lines), method (5.10) order $\mathcal{O}(h^{10}\varepsilon + h^4\varepsilon^2)$ (dashed lines). With processing: methods (5.5) and (5.11) of orders $\mathcal{O}(h^6\varepsilon + h^4\varepsilon^2)$ and $\mathcal{O}(h^6\varepsilon)$ (dashed-dotted lines), new method (Sect. 5.2) of order $\mathcal{O}(h^{10}\varepsilon + h^4\varepsilon^2)$ (solid line).

where

$$T(y,p) = \frac{1}{2}\sum_{i=1}^{N}\left(\frac{p_{i,1}^2 + p_{i,2}^2 + p_{i,3}^2}{m} + \frac{y_{i,1}^2}{I_1} + \frac{y_{i,2}^2}{I_2} + \frac{y_{i,3}^2}{I_3}\right)$$

is the sum of the translational and rotational kinetic energies of each rigid body. We consider an inter-body potential $V(Q,q)$ of the form

$$V(Q,q) \;=\; \sum_{1\le i<j\le N} V_{i,j}(Q_i, q_i, Q_j, q_j)$$

where the interaction between particle i and particle j has the form

$$\begin{aligned} V_{i,j} &= \varepsilon V_{i,j}^{short} + \varepsilon V_{i,j}^{dip} \\ V_{i,j}^{short} &= 4r_{i,j}^{-12} \qquad r_{i,j} = \|q_i - q_j\| \\ V_{i,j}^{dip} &= r_{i,j}^{-3}\mu_i^T\mu_j + 3r_{i,j}^{-5}\mu_i^T(q_i - q_j)\mu_j^T(q_j - q_i). \end{aligned}$$

Here, the term $V_{i,j}^{short}$ corresponds to a repulsive short range interaction, while $V_{i,j}^{dip}$ models the dipole interaction which depends on the dipole vector orientations $\mu_i \in \mathbb{R}^3$ given by

$$\mu_i = Q_i\overline{\mu_i}$$

for some fixed reference orientations $\overline{\mu_i} \in \mathbb{R}^3$.

The problem can be split as

$$\begin{array}{lcllcll} \dot y_i &=& \widehat{y_i} I^{-1} y_i & \qquad & \dot y_i &=& -\mathrm{rot}\,(Q_i^T \frac{\partial V}{\partial Q_i}) \\ \dot p_i &=& 0 & & \dot p_i &=& -\frac{\partial V}{\partial q_i} \\ \dot Q_i &=& Q\widehat{I^{-1}y_i} & & \dot Q_i &=& 0 \\ \dot q_i &=& \frac{p_i}{m} & & \dot q_i &=& 0 \end{array} \qquad i = 1 \ldots N.$$

where the operator rot is defined in (5.13). The vectors

$$v_i = \mathrm{rot}\,(Q_i^T\frac{\partial V}{\partial Q_i}) = \varepsilon\sum_{j\neq i} a_{ij} \qquad \text{and} \qquad w_i = \frac{\partial V}{\partial q_i} = \varepsilon\sum_{j\neq i} b_{ij} \tag{5.14}$$

can be computed straightforwardly using

$$\begin{aligned} a_{ij} &= \sum_{k=1}^{3} \begin{pmatrix} \alpha_{ijk2}Q_{ik3} - \alpha_{ijk3}Q_{ik2} \\ \alpha_{ijk3}Q_{ik1} - \alpha_{ijk1}Q_{ik3} \\ \alpha_{ijk1}Q_{ik2} - \alpha_{ijk2}Q_{ik1} \end{pmatrix}, \qquad \alpha_{ijkl} = \frac{\partial V_{i,j}}{\partial Q_{ikl}} = r_{i,j}^{-3}\overline{\mu}_{il}\mu_{jk} + 3r_{i,j}^{-5}\gamma_{ji}\overline{\mu}_{il}q_{ij}^{(k)}, \\ b_{ij} &= \Big(-48r_{ij}^{-14} - 3r_{i,j}^{-5}\mu_i^T\mu_j - 15r_{i,j}^{-7}\gamma_{ij}\gamma_{ji}\Big)q_{ij} + 3r_{i,j}^{-5}\gamma_{ji}\mu_i - 3r_{i,j}^{-5}\gamma_{ij}\mu_j, \end{aligned}$$

where we use the notations

$$q_{ij} = q_i - q_j \qquad \gamma_{ij} = \mu_i^T(q_i - q_j), \tag{5.15}$$

and $q_{ij}^{(k)}$ is the kth component of vector q_{ij}.

Computation of f^C Consider the vector fields f^A, g^A, f^B given by

$$f_i^A = \begin{pmatrix} \widehat{y_i}I^{-1}y_i \\ 0 \end{pmatrix} \qquad g_i^A = \begin{pmatrix} Q_i\widehat{I^{-1}y_i} \\ \frac{p_i}{m} \end{pmatrix} \qquad f_i^B = \begin{pmatrix} -\mathrm{rot}\,(Q_i^T\frac{\partial V}{\partial Q_i}) \\ -\frac{\partial V}{\partial q_i} \end{pmatrix} \qquad i = 1 \dots N.$$

Then, vector field f^C is given by

$$\begin{aligned} f_i^C(Q,q) &= \frac{\partial^2 f_i^A}{\partial(y,p)^2}(f^B, f^B) - 2\frac{\partial f_i^B}{\partial(Q,q)}\frac{\partial g^A}{\partial(y,p)}f^B \\ &= \begin{pmatrix} 2\widehat{v_i}I^{-1}v_i - 2\varepsilon\sum_{j\neq i} c_{i,j} \\ -2\varepsilon\sum_{j\neq i} d_{i,j} \end{pmatrix} \end{aligned}$$

where the vectors v_i are given in (5.14) and the vectors c_{ij}, d_{ij} can be computed as follows.

$$\begin{aligned} c_{ij} &= \sum_{k=1}^{3} \begin{pmatrix} \beta_{ijk2}Q_{ik3} - \beta_{ijk3}Q_{ik2} + \alpha_{ijk2}\widetilde{Q}_{ik3} - \alpha_{ijk3}\widetilde{Q}_{ik2} \\ \beta_{ijk3}Q_{ik1} - \beta_{ijk1}Q_{ik3} + \alpha_{ijk3}\widetilde{Q}_{ik1} - \alpha_{ijk1}\widetilde{Q}_{ik3} \\ \beta_{ijk1}Q_{ik2} - \beta_{ijk2}Q_{ik1} + \alpha_{ijk1}\widetilde{Q}_{ik2} - \alpha_{ijk2}\widetilde{Q}_{ik1} \end{pmatrix} \\ \beta_{ijkl} &= r_{i,j}^{-3}\overline{\mu}_{il}\widetilde{\mu}_{jk} - 3r_{i,j}^{-5}\widetilde{\mu}_j^T q_{ij}\overline{\mu}_{il}q_{ij}^{(k)} \\ &+ \frac{3}{m}r_{i,j}^{-5}\overline{\mu}_{il}\Big(\big(\mu_{jk} + 5r_{i,j}^{-2}\gamma_{ji}q_{ij}^{(k)}\big)q_{ij}^T w_{ji} + \mu_j^T w_{ji}q_{ij}^{(k)} - \gamma_{ji}w_{jik}\Big) \end{aligned}$$

$$\begin{aligned} d_{ij} &= 3r_{ij}^{-5}\Big(-\mu_i^T\widetilde{\mu}_j - \mu_j^T\widetilde{\mu}_i + 5r_{ij}^{-2}\gamma_{ij}\widetilde{\mu}_j^T q_{ij} - 5r_{ij}^{-2}\gamma_{ji}\widetilde{\mu}_i^T q_{ij} \\ &+ 5\frac{r_{ij}^{-2}}{m}(\gamma_{ji}w_{ji}^T\mu_i - \gamma_{ij}w_{ji}^T\mu_j) \\ &+ \frac{r_{ij}^{-2}}{m}q_{ij}^T w_{ji}((-224r_{ij}^{-9} - 5\mu_i^T\mu_j - 35r_{ij}^{-2}\gamma_{ij}\gamma_{ji}))\Big)q_{ij} \\ &+ 3r_{ij}^{-5}\Big(-\widetilde{\mu}_j^T q_{ij} + \frac{1}{m}\mu_j^T w_{ji} + 5\frac{r_{ij}^{-2}}{m}q_{ij}^T w_{ji}\gamma_{ji}\Big)\mu_i \\ &+ 3r_{ij}^{-5}\Big(-\widetilde{\mu}_i^T q_{ij} + \frac{1}{m}\mu_i^T w_{ji} - 5\frac{r_{ij}^{-2}}{m}q_{ij}^T w_{ji}\gamma_{ij}\Big)\mu_j \\ &+ 3r_{ij}^{-5}\gamma_{ji}\widetilde{\mu}_i - 3r_{ij}^{-5}\gamma_{ij}\widetilde{\mu}_j \\ &+ 3\frac{r_{ij}^{-5}}{m}\Big(16r_{ij}^{-9} + \mu_i^T\mu_j + 5r_{ij}^{-2}\gamma_{ij}\gamma_{ji}\Big)w_{ji} \end{aligned}$$

We have used the following notations

$$w_{ij} = w_i - w_j, \qquad \widetilde{Q}_i = Q_i \widehat{I^{-1} v_i}, \qquad \widetilde{\mu}_i = \widetilde{Q}_i \overline{\mu}_i$$

where v_i and w_i are given in (5.14) and $q_{ij}, q_{ij}^{(k)}, \gamma_{ij}$ are given in (5.15).

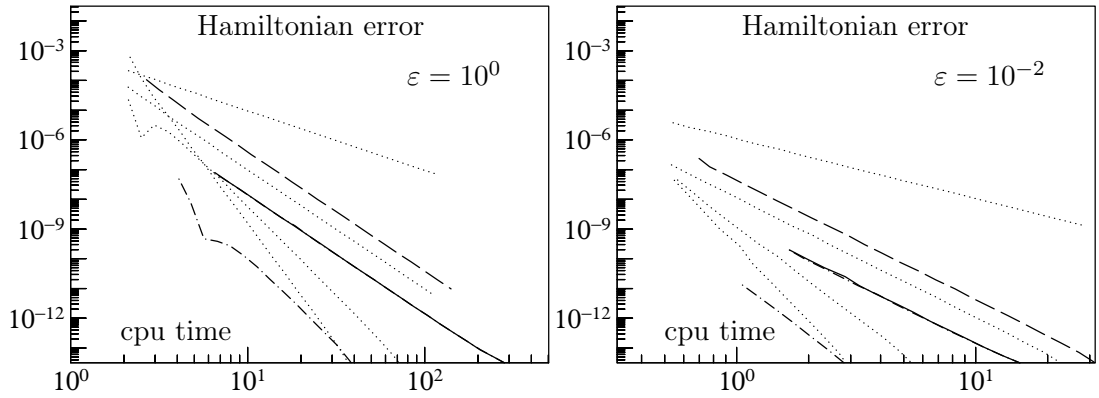


Figure 5.4: Molecular dynamics simulation: dipolar soft spheres. Comparison of Hamiltonian error versus cpu time for various splitting methods. Without processing: Strang splitting and composition methods (5.2) of orders $\mathcal{O}(h^p\varepsilon)$, $p = 2, 4, 6, 8$ (dotted lines), method (5.10) order $\mathcal{O}(h^{10}\varepsilon + h^4\varepsilon^2)$ (dashed lines). With processing: methods (5.5) and (5.11) of orders $\mathcal{O}(h^6\varepsilon + h^4\varepsilon^2)$ and $\mathcal{O}(h^6\varepsilon)$ (dashed-dotted lines), new method (Sect. 5.2) of order $\mathcal{O}(h^{10}\varepsilon + h^4\varepsilon^2)$ (solid line).

Numerical experiment We integrate in the interval $[0, 5]$. Similarly to the numerical experiments in [CFSZ08], we take 125 particles with initial positions $q_i(0)$ the points with integer coordinates of the cube $[0, 4]^3$ in $\mathbb{R}^3$, and initial velocities $p_i(0)$ zero. The angular momentum $y_i(0)$ and the orientations of each particle is chosen randomly with a quaternion with random components in $[-1, 1]$. We take $m = 1$, the moments of inertia of the rigid bodies are $I_1 = 0.345, I_2 = 0.653, I_3 = 1.0$ corresponding water molecules, and the fixed orientations for the dipoles are $\overline{\mu}_i = (0, 1, 1)^T$.

For this problem, the processed method (5.11) of [BCR01] is more efficient than the processed Takahashi–Imada. An explanation is the following. The cost of computing f^A is $\mathcal{O}(N)$, linear with the number of particles, whereas the cost of the torque f^B grows quadratically $\mathcal{O}(N^2)$, and thus dominates the cost. Our numerical experiments indicate that the computation of f^B together with f^C costs about 3 times the computation of f^B alone. Unlike previous problems considered in this chapter, this is far from negligible. Method (5.11) and the processed Takahashi–Imada method both require one evaluation of f^C per step. Method (5.11) requires 3 evaluations of f^B, thus the relative overcost due to the expensive evaluation of f^C is about $2/3 \approx 0.66\%$, whereas the processed Takahashi–Imada method requires one evaluation of f^B so the relative overcost is about 200%.

Chapter 6

Splitting methods with complex times for parabolic equations

Note: This chapter is identical to the article [CCDV08] in collaboration with F. Castella, P. Chartier and S. Descombes.
Similar results are derived independently by E. Hansen & A. Ostermann in [HO08b].

Although the numerical simulation of the heat equation in several space dimensions is now well understood, there remain a lot of challenges in the presence of an external source, *e.g.* for reaction-diffusion problems, or more generally for the complex Ginzburg-Landau equation.

Reaction-diffusion equations are mathematical models that describe how the population of one or several species distributed in space evolves under the action of two concurrent phenomena:

– "reactions" between species in which predators eat preys;

– diffusion which makes the species spread out in space.

Apart from biology and ecology, systems of this sort also appear in chemistry (hence the term reaction), geology and physics. From a mathematical point of view, they belong to the class of semi-linear parabolic partial differential equations and can be represented in the general form

$$\frac{\partial u}{\partial t} = D\Delta u + F(u),$$

where each component of the vector $u(x,t) \in \mathbb{R}^d$ represents the population of one species, D is the matrix of diffusion coefficients (often diagonal) and F accounts for all local interactions between species. The solutions of reaction-diffusion equations display a wide range of behaviours, like traveling waves and wave-like phenomena, or dissipative solitons.

The most simple reaction-diffusion equation occurs in chemistry and involves the concentration u of a single reactant in one spatial dimension,

$$\partial_t u = D\partial_x^2 u + F(u), \tag{6.1}$$

is also referred to as the KPP (Kolmogorov-Petrovsky-Piscounov [KPP37]) equation. If the reaction term vanishes, then the corresponding equation is the heat equation. Specific forms of this one-dimensional reaction-diffusion equation appear in the litterature:

– the choice $F(u) = u(1-u)$ yields Fisher's equation and is used to describe the spreading of biological populations;

– the choice $F(u) = u(1-u^2)$ describes Rayleigh-Benard convection;

– the choice $F(u) = u(1-u)(u-\alpha)$ with $0 < \alpha < 1$ arises in combustion theory and is referred to as Zeldovich equation.

Two-component (and more) systems are also possible and allow for a much larger range of possible phenomena than their one-component counterparts. They can be represented in the following form

$$\begin{pmatrix} \partial_t u_1 \\ \vdots \\ \partial_t u_d \end{pmatrix} = \begin{pmatrix} D_1 & & \\ & \ddots & \\ & & D_d \end{pmatrix} \begin{pmatrix} \Delta u_1 \\ \vdots \\ \Delta u_n \end{pmatrix} + \begin{pmatrix} F_1(u_1, \ldots, u_d) \\ \vdots \\ F_d(u_1, \ldots, u_d) \end{pmatrix}$$

Even more generally, the diffusion operator may involve a complex number: we shall denote $\operatorname{Re} z$ the real part, $\operatorname{Im} z$ the imaginary part, and $\arg z \in]-\pi, \pi]$ the argument of a complex number z. In this case, we obtain a complex Ginzburg-Landau equation with a polynomial non-linearity of the form

$$\frac{\partial u}{\partial t} = \delta \Delta u - \sum_{j=0}^{K} \mu_j |u|^{2j} u \tag{6.2}$$

where $K \geq 1$ is an integer, δ and μ_j, $j = 1 \ldots K$, are complex numbers with $\operatorname{Re} \delta > 0$, and $\operatorname{Re} \mu_K > 0$. For example, when $K = 1$, we obtain the well-known cubic Ginzburg-Landau equation [FT88], and when $K = 2$, the equation given by Fauve-Thual in [6] as a model of localized structures generated by subcritical instabilities.

When one wishes to approximate the solution of the above parabolic non-linear problem (6.1) or (6.2), a method of choice is based on operator-splitting: the idea is to split the abstract evolution equation (6.1) (or (6.2)) into two parts which can be solved explicitly or at least approximated efficiently.

For the sake of simplicity, let us illustrate the method on the linear case

$$\frac{\partial u}{\partial t} = \Delta u + V u, \tag{6.3}$$

where V is a linear operator, say $Vu = v(x)u$ with $v(x)$ a smooth function. Splitting methods basically rely on the identity

$$e^{h(\Delta+V)} = e^{h\Delta}\, e^{hV} + \mathcal{O}\left(h^2\right),$$

or on higher order approximations obtained by combining $e^{h\Delta}$ and e^{hV} in the appropriate fashion. Dividing time t into n time steps of size h (where $t = nh$), the above approximation indeed leads to the equality

$$u(t) = e^{t(\Delta+V)}\, u(0) = e^{nh(\Delta+V)}\, u(0) = \left(e^{h\Delta}\, e^{hV}\right)^n u(0) + \mathcal{O}(h).$$

The extension to the non-linear case is straightforward, replacing e^{hV} by the flow of a nonlinear differential equation.

For a positive stepsize h, the most simple numerical integrator is the Lie-Trotter splitting

$$e^{hV} e^{h\Delta} \tag{6.4}$$

which is an approximation of order 1 of the solution of (6.3), while the symmetric version

$$e^{h/2\,V} e^{h\Delta} e^{h/2\,V} \tag{6.5}$$

is referred to as the Strang splitting and is an approximation of order 2. For higher orders, one can consider general splitting methods of the form

$$e^{b_1 hV} e^{a_1 h\Delta} e^{b_2 hV} e^{a_2 h\Delta} \dots e^{b_s hV} e^{a_s h\Delta}. \tag{6.6}$$

The number of order conditions can be significantly reduced by imposing the symmetry $a_s = 0$ and $a_j = a_{s-j}, b_j = b_{s+1-j}$ whenever $1 \le j \le s$. It is interesting to note that raising the order can also be achieved by considering composition methods of the form

$$\Psi_h := \Phi_{\gamma_s h} \circ \dots \circ \Phi_{\gamma_1 h}, \tag{6.7}$$

where Φ_h is a low order approximation. Symmetry can even be obtained by imposing $\gamma_j = \gamma_{s+1-j}$ $(1 \le j \le s)$, and by choosing Φ_h symmetric. For instance, when Φ_h is the Strang splitting (6.5), this approach leads to

$$\Psi_h = e^{h\gamma_s/2\,V} e^{h\gamma_s\Delta} e^{h(\gamma_s+\gamma_{s-1})/2\,V} e^{h\gamma_{s-1}\Delta} \dots e^{h\gamma_1\Delta} e^{h\gamma_1/2\,V}.$$

However, achieving higher order is not as straightforward as it looks. A disappointing result indeed shows that all splitting methods (or composition methods) with real coefficients must have negative coefficients a_i and b_i in order to achieve order 3 or more. The existence of at least one negative coefficient was shown in [She89, SW92], and the existence of a negative coefficient for both operators was proved in [GK96]. An elegant geometric proof can be found in [BC05]. As a consequence, such splitting methods *cannot* be used when one operator, like Δ, is not time-reversible.

In order to circumvent this order-barrier, there are two possibilities. One can use a linear, convex combination (see [GRT02, GRT04, BDL06] for methods of order 3 and 4) or non-convex combination (see [Sch02, Des01] where an extrapolation procedure is exploited), of elementary splitting methods like (6.6). Another possibility is to consider splitting methods with *complex* coefficients a_i and b_i with positive real parts (see [Cha03] in celestial mechanics). In 1962/1963, Rosenbrock [Ros63] considered complex coefficients in a similar context.

In this article, we consider splitting methods of the form (6.7), and we derive new high-order methods using composition techniques originally developed for the geometric numerical integration of ordinary differential equations [HLW06]. The main advantages of this approach are the following:

– the splitting method inherits the stability property of exponential operators;

– we can replace the costly exponentials of the operators by cheap low order approximations without altering the overall order of accuracy;

– using complex coefficients allows to reduce the number of compositions needed to achieve any given order;

This paper is organized as follows. In Sect. 6.1, we derive new high-order splitting methods. In Sect. 6.2 we give a rigorous order estimate in the linear case, obtained as a direct consequence of the recent results by Hansen & Ostermann [HO08a]. Sect. 6.3 presents several numerical simulations, confirming the formally expected order of accuracy in the non-linear case.

6.1 Composition methods

Composition methods were mainly developed in the the 90's in the papers of Suzuki [Suz90], Yoshida [Yos90] and McLachlan [McL95] in the context of ordinary differential equations. In the classical theory, only real coefficients $\gamma_1, \ldots, \gamma_s$ were considered. In this section, we construct new composition methods, involving complex coefficients.

The idea is to compose with different stepsizes a basic one-step method Φ_h of low order of accuracy. Given the scalars $\gamma_1, \ldots, \gamma_s$, the corresponding composition method is defined as

$$\Psi_h = \Phi_{\gamma_s h} \circ \ldots \circ \Phi_{\gamma_1 h} \tag{6.8}$$

with stepsizes $\gamma_1 h, \gamma_2 h, \ldots, \gamma_s h$.

For the method Φ_h, the simplest choice is the symmetric Strang splitting

$$\Phi_h = e^{h/2\,V} e^{h\Delta} e^{h/2\,V}.$$

Then, we obtain the splitting method

$$\Psi_h = e^{h\gamma_s/2\,V} e^{h\gamma_s\Delta} e^{h(\gamma_s+\gamma_{s-1})/2\,V} e^{h\gamma_{s-1}\Delta} \ldots e^{h\gamma_1\Delta} e^{h\gamma_1/2\,V}$$

Another possibility, to avoid computing the exact flow $e^{h\Delta}$, is to replace it by a symmetric approximation, e.g. the Crank-Nicholson discretization (which is equivalent to the implicit midpoint rule for linear systems)

$$\Phi_h^M = \left(Id - \frac{h}{2}\Delta\right)^{-1} \left(Id + \frac{h}{2}\Delta\right).$$

One can also discretize the flow e^{hV} of the reaction, and consider the cheaper basic symmetric method

$$\Phi_h = \Phi_{h/2}^I \circ \Phi_h^M \circ \Phi_{h/2}^E \tag{6.9}$$

where Φ_h^E denotes the flow of the explicit Euler method $y_{n+1} = y_n + hf(y_n)$ and Φ_h^I denotes the flow of the implicit Euler method $y_{n+1} = y_n + hf(y_{n+1})$ for the approximation of the reaction. This is the Peaceman-Rachford formula [PJ55] originally developed for the heat equation, and extended to reaction-diffusion problems in [DR03].

Focusing back to composition methods, their construction relies on the following classical result in geometric integration.

Theorem 6.1.1 *(see [HLW06, Theorem II.4.1]) Let Φ_h be a method of (classical) order p. If*

$$\gamma_1 + \ldots + \gamma_s = 1 \qquad \gamma_1^{p+1} + \ldots + \gamma_s^{p+1} = 0, \tag{6.10}$$

then composition method (6.7) has (classical) order $p+1$.

Proof. The idea of proof is to show that if the basic method has order p,

$$\Phi_h(y) = \varphi_h(y) + C(y)h^{p+1} + \mathcal{O}(h^{p+2}),$$

where φ_h denotes the exact flow, then

$$\Phi_{\gamma_s h} \circ \ldots \circ \Phi_{\gamma_1 h}(y) = \varphi_h(y) + C(y)(\gamma_1^{p+1} + \ldots + \gamma_s^{p+1})h^{p+1} + \mathcal{O}(h^{p+2}).$$

□

6.1.1 Triple Jump composition methods with real coefficients

Using only real coefficients, equations (6.10) have no real solution for odd p, so the order increase is only possible for even p. In this case, the smallest s which allows for the existence of a solution is $s = 3$. If we impose symmetry, $\gamma_1 = \gamma_3$ and we obtain a unique solution

$$\gamma_1 = \gamma_3 = \frac{1}{2 - 2^{1/(p+1)}}, \qquad \gamma_2 = -\frac{2^{1/(p+1)}}{2 - 2^{1/(p+1)}}. \tag{6.11}$$

If the basic method Φ_h is symmetric of order p (p even), then we reach order $p + 1$, but due to the symmetry of the method the order is in fact $p + 2$. Now, this procedure can be repeated up to any order: we start with a symmetric method of order 2, we apply (6.11) with $p = 2$ to obtain order 4. With this new method, we repeat (6.11) with $p = 4$ to obtain a symmetric composition method of order 6 with 9 stages and so on.

These methods are originally due to Creutz & Gocksch [CG89], Forest [For89], Suzuki [Suz90], Yoshida [Yos90]. The name 'Triple Jump composition methods' was given in [HLW06, Example II.4.2]. However, since $\gamma_2 < 0$, these methods cannot be applied to non-reversible problems. In addition, the estimate $|\gamma_j| > 1$ implies a terrible zig-zag in the coefficients of the methods. Thus, this technique is not very efficient in the context of ordinary differential equations, and to reach high order it is much better to resort to general composition methods by directly solving the p-order conditions for large p.

6.1.2 Triple Jump composition methods with complex coefficients

For $s = 3$, equation (6.10) possesses $p + 1$ solutions in $\mathbb{C}$,

$$\gamma_1 = \gamma_3 = \frac{1}{2 - 2^{1/(p+1)} e^{2ik\pi/(p+1)}}, \qquad \gamma_2 = -\frac{2^{1/(p+1)} e^{2ik\pi/(p+1)}}{2 - 2^{1/(p+1)} e^{2ik\pi/(p+1)}}, \quad k = 0, \dots p.$$

The real solution with $k = 0$ is the one in (6.11), and the two conjugate solutions which minimize $|\gamma_1| + |\gamma_2| + |\gamma_3|$ are obtained for $k = \pm p/2$ (here p is even). It yields the solutions $(\gamma_1, \gamma_2, \gamma_3)$ and $(\overline{\gamma}_1, \overline{\gamma}_2, \overline{\gamma}_3)$ where

$$\gamma_1 = \gamma_3 = \frac{e^{i\pi/(p+1)}}{2e^{i\pi/(p+1)} + 2^{1/(p+1)}}, \qquad \gamma_2 = \frac{2^{1/(p+1)}}{2e^{i\pi/(p+1)} + 2^{1/(p+1)}}. \tag{6.12}$$

Notice that these two conjugate solutions also minimize the quantity $\max_{i=1,2,3} |\arg(\gamma_i)|$.

As a consequence, the method $\Phi_h^{[4]}$ of order 4 is defined as

$$\Phi_h^{[4]} = \Phi_{\gamma_1} \circ \Phi_{\gamma_2} \circ \Phi_{\gamma_1} \tag{6.13}$$

where $\gamma_1 = \gamma_3$ and γ_2 are given in (6.12) with $p = 2$, and requires three compositions.

Then, similarly to the approach with real coefficients, symmetric composition methods $\Phi_h^{[p]}$ of order p (p even) can be constructed by induction:

$$\Phi_h^{[2]} = \Phi_h, \qquad \Phi_h^{[p+2]} = \Phi_{\gamma_3 h}^{[p]} \circ \Phi_{\gamma_2 h}^{[p]} \circ \Phi_{\gamma_1 h}^{[p]} \quad \text{for } p \geq 2, \tag{6.14}$$

where $\gamma_1, \gamma_2, \gamma_3$ are given in (6.12). The method $\Phi_h^{[p]}$ requires $s = 3^{p/2-1}$ compositions of the basic method Φ_h.

We observe in Figure 6.2 that the quantity $\max_{i=1\dots s} |\arg(\gamma_i)|$ increases with the order p of the composition methods in (6.14). For the method (6.14) of order $p = 10$ this quantity is greater than $\pi/2$. Indeed, it possesses $s = 81$ compositions and the middle coefficient

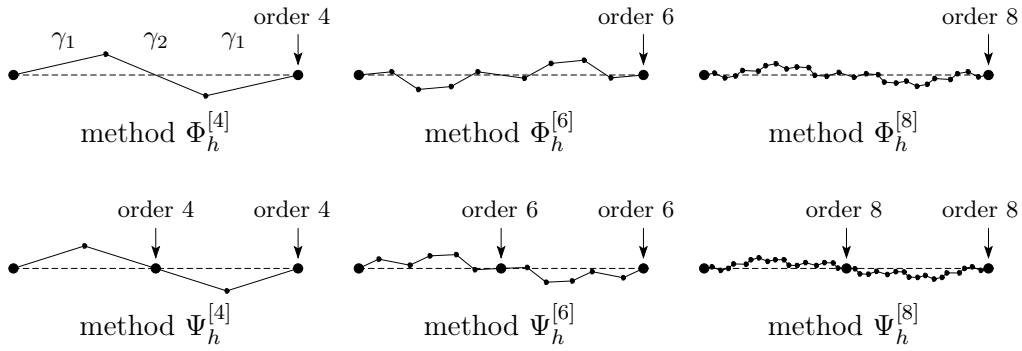


Figure 6.1: Diagrams of coefficients for compositions methods (6.15) and (6.16).

γ_{41} has a negative real part: $\mathrm{Re}\,(\gamma_{41}) \approx -5 \cdot 10^{-5} < 0$. Thus, this method cannot be used for non-reversible problems. An improvement to reduce the quantity $\max_{i=1\ldots s} |\arg(\gamma_i)|$ is to replace in (6.14) the coefficients $(\gamma_1, \gamma_2, \gamma_3)$ by $(\overline{\gamma}_1, \overline{\gamma}_2, \overline{\gamma}_3)$ alternatively, e.g.

$$\Phi_h^{[p+2]} = \Phi_{\gamma_3 h}^{[p]} \circ \Phi_{\gamma_2 h}^{[p]} \circ \Phi_{\gamma_1 h}^{[p]} \quad \text{if } p/2 \text{ odd}, \qquad \Phi_h^{[p+2]} = \Phi_{\overline{\gamma}_3 h}^{[p]} \circ \Phi_{\overline{\gamma}_2 h}^{[p]} \circ \Phi_{\overline{\gamma}_1 h}^{[p]} \quad \text{else.} \quad (6.15)$$

This yields a family of composition methods with $\max_{i=1\ldots s} |\arg(\gamma_i)| \leq \pi/2$ for $p = 2 \ldots 14$, as we can see in Figure 6.2.

Remark 6.1.2 Surprisingly, the sum of the moduli of coefficients $|\gamma_1| + |\gamma_2| + |\gamma_3| + \ldots$ involved in (6.8) in the considered family of composition methods is bounded as the order goes to infinity:

$$\prod_{k=1}^{\infty} \frac{2 + 2^{1/(2k+1)}}{|2e^{i\pi/(2k+1)} + 2^{1/(2k+1)}|} = \prod_{k=1}^{\infty} \left(1 + \frac{\pi^2}{36k^2} + \mathcal{O}\Big(\frac{1}{k^3}\Big)\right) < +\infty$$

This means that the length of the family of polygons in Figure 6.1 above is bounded (this limit is ≈ 1.315).

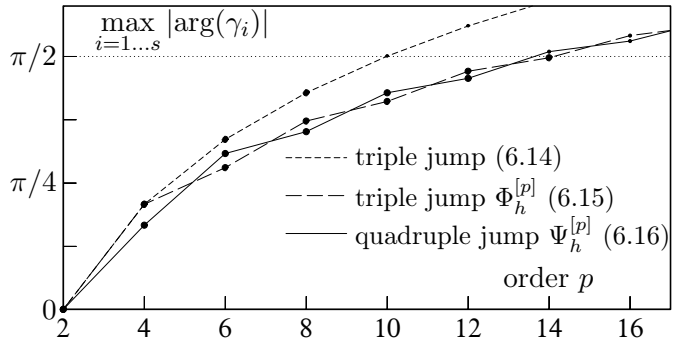


Figure 6.2: Values of $\max_{i=1\ldots s} |\arg\gamma_i|$ for various composition methods.

6.1.3 Quadruple Jump composition methods

For $s = 4$, imposing symmetry ($\gamma_1 = \gamma_4$, $\gamma_2 = \gamma_3$) equation (6.10) possesses p complex solutions

$$\gamma_1 = \overline{\gamma}_2 = \overline{\gamma}_3 = \gamma_4 = \frac{1}{2 - 2e^{ki\pi/(p+1)}}, \quad k = 1 \ldots p.$$

The two complex conjugate solutions with minimal sum of moduli (and also minimal $\max_{i=1\ldots4} |\arg(\gamma_i)|$) are obtained with $k = \pm p/2$, e.g.

$$\gamma_1 = \overline{\gamma}_2 = \overline{\gamma}_3 = \gamma_4 = \frac{1}{4} + i\frac{\sin(\pi/(p+1)}{4 + 4\cos(\pi/(p+1))}.$$

Hence, similarly to the triple jump composition methods in the previous section, for a symmetric basic method of order p, e.g. $\Phi_h^{[p]}$ in (6.15), we obtain a composition method

$$\Psi_h^{[p+2]} = \Phi_{\gamma_4 h}^{[p]} \circ \Phi_{\gamma_3 h}^{[p]} \circ \Phi_{\gamma_2 h}^{[p]} \circ \Phi_{\gamma_1 h}^{[p]} \tag{6.16}$$

of order $p + 2$.

The main advantage of this type of composition is that we obtain an accurate approximation of the solution in the middle as well (notice $\gamma_1 + \gamma_2 = 1/2$), namely

$$\Phi_{\gamma_2 h}^{[p]} \circ \Phi_{\gamma_1 h}^{[p]}.$$

Indeed, it can be checked that $(2\gamma_1, 2\gamma_2)$ and $(2\gamma_2, 2\gamma_1)$ are solutions of equation (6.10) with $s = 2$. This shows that

$$y_{n+1/2} = \Phi_{\gamma_2 h}^{[p]} \circ \Phi_{\gamma_1 h}^{[p]}(y_n) \tag{6.17}$$

yields an approximation of the solution at time $t = t_n + h/2$ with local error $\mathcal{O}(h^{p+2})$. Since this error is not propagated (it is only an inner stage), we obtain an approximation of order $p + 2$ not only for y_{n+1} at time $t_n + h$ but also for $y_{n+1/2}$ at time $t_n + h/2$.

We now give the details for the composition method of order 4

$$\Psi_h^{[4]} = \Phi_{\gamma_1 h} \circ \Phi_{\overline{\gamma}_1 h} \circ \Phi_{\overline{\gamma}_1 h} \circ \Phi_{\gamma_1 h}. \tag{6.18}$$

Algorithm 6.1.3 *Take a basic method for the solution of problem (6.1): either the Strang splitting with exponential maps (6.5),*

$$\Phi_h = e^{h/2\,V} e^{h\Delta} e^{h/2\,V}$$

or the Peaceman-Rachford formula (6.9),

$$\Phi_h = \Phi_{h/2}^I \circ \Phi_h^M \circ \Phi_{h/2}^E$$

One step of the "quadruple jump" composition method reads

$$\begin{aligned} y_{n+1/2} &= \Phi_{\overline{\gamma}_1\, h} \circ \Phi_{\gamma_1\, h}(y_n) \\ y_{n+1} &= \Phi_{\gamma_1\, h} \circ \Phi_{\overline{\gamma}_1\, h}(y_{n+1/2}) \end{aligned}$$

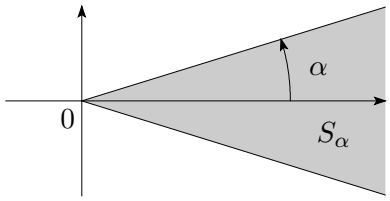
where $\gamma_1 = 1/4 + i\sqrt{3}/6$. This yields an approximation y_{n+1} of (formal) order 4 at time $t_n + h$, and an approximation $y_{n+1/2}$ of (formal) order 4 at time $t_n + h/2$.

6.2 Convergence analysis for unbounded operators

In this section, we introduce the notion of mα-dissipative operator and we describe the framework introduced in [HO08a]. We then show how it can be adapted in our situation to derive order results for linear parabolic equations.

Let α belong to $[0, \pi/2]$ and let us define the sector S_α in the complex plane by

$$S_\alpha = \{z \in \mathbb{C}, z = 0 \,\text{or}\, |\arg z| \leq \alpha\}.$$



Let H be a complex Hilbert space with scalar product denoted by $(\cdot,\cdot)$. We recall that a linear operator A with domain $D(A)$ dense in H is mα-dissipative if for all u in $D(A)$, $-(Au,u)$ belongs to S_α and if for all complex z which does not belong to S_α, $zId + A$ is an isomorphism from $D(A)$ to H.

A nice introduction to mα-accretive operators[1] can be found in [Cro05]. A mα-dissipative operator generates a C_0 semigroup on H and is a contraction operator from H to H. This framework is well adapted for our study since, on the one hand, the two linear operators defined in the introduction $A = \Delta$ and $A_c = c\Delta$ with $\operatorname{Re} c > 0$ with the same domain $H^2(\mathbb{R}^d)$ are respectively m0-dissipative and mα-dissipative with $\alpha = \arg c$, and, on the other hand, if A is a mα-dissipative operator and c is a complex number with $\beta = |\arg c|$ such that $\alpha + \beta \leq \pi/2$ then cA is a m$(\alpha+\beta)$-dissipative operator.

We now quote the main result of [HO08a]. Let X an arbitrary complex Banach space with norm $\|\cdot\|$, and consider $s+1$ linear unbounded operators L and A_j $(j = 1,\ldots,s)$, with $L = A_1 + \cdots A_s$ satisfying the following assumption :

Assumption 6.2.1 *The linear operators L and A_j $(j = 1,\ldots,s)$, generate a C_0 semigroup on X. Moreover there exist a real ω and s real numbers ω_j $(j = 1,\ldots,s)$ such that for $t \geq 0$ the operator L satisfies the following bound*

$$\left\|e^{tL}\right\| \leq e^{\omega t}, \tag{6.19}$$

and the operators A_j, $j = 1,\ldots,s$, satisfy the following bounds

$$\left\|e^{tA_j}\right\| \leq e^{\omega_j t}. \tag{6.20}$$

As mentioned previously this assumption is satisfied in the context of mα-dissipative operators.

The authors then introduce the function u defined for $t \geq 0$ and u_0 in X by

$$u(t) = e^{tL}u_0, \tag{6.21}$$

and a splitting method S of the form

$$S = \prod_{j=1}^{s} e^{\gamma_j h A_j}$$

where γ_j, $1 \leq j \leq s$, are nonnegative reals, and this splitting method is assumed to possess order p.

For an integer k they denote by E_k various compositions of the operators A_j $(j = 1,\ldots,s)$, that consist of exactly k factors and introduce the following assumption :

Assumption 6.2.2 *All expressions of the form $E_{p+1}u(t)$ are uniformly bounded on the interval $0 \leq t \leq T$ for some $T > 0$.*

[1] An operator B is said mα-accretive whenever $A = -B$ is mα-dissipative.

Under the previous assumptions they obtain the following theorem :

Theorem 6.2.3 *Let $T > 0$, $h > 0$ and n an integer, if assumptions 6.2.1 and 6.2.2 are valid, then*

$$\left\| \left(S^n - e^{nhL} \right) u_0 \right\| \leq Ch^p, \quad nh \leq T,$$

where the constant C can be chosen uniformly on bounded time intervals and, in particular, independent of n and h.

We now present the application of the previous theorem in our framework with H an Hilbert space. Since we are working with parabolic operators we only treat the case of mα-dissipative operators with $\alpha \leq \pi/2$.

Theorem 6.2.4 *Let $s+1$ linear unbounded operators L and A_j $(j = 1, \ldots, s)$. Let β_j $(1 \leq j \leq s)$, be complex numbers with positive real part and assume that $L = \beta_1 A_1 + \cdots + \beta_s A_s$. Moreover, assume that there exist η_j $(1 \leq j \leq s)$, real numbers such that $\beta_j A_j + \eta_j Id$ is mα_j-dissipative with α_j belonging to $[0, \pi/2($ and such that $L + \sum_{j=1}^{s} \eta_j Id$ generates a C_0 semigroup on H and satisfies* (6.19).

Let S an approximation of order p given by

$$S = \prod_{j=1}^{s} e^{h\beta_j A_j}.$$

If Assumption 6.2.2 with L and A_j $(1 \leq j \leq s)$ is satisfied, then

$$\left\| \left(S^n - e^{nhL} \right) u_0 \right\| \leq Ch^p, \quad nh \leq T,$$

where the constant C can be chosen uniformly on bounded time intervals and, in particular, independent of n and h.

Proof. This is a simple application of Theorem 6.2.3 with $L + \sum_{j=1}^{s} \eta_j Id$ and $\beta_j A_j + \eta_j Id$, $1 \leq j \leq s$. □

We now give two examples. Let V be a real bounded function of C^∞ class. The first example is obtained by taking for even j, $A_j = \Delta$ with domain $D(A_j) = H^2(\mathbb{R}^d)$, and for odd j, $A_j = V$. Since the operator Δ with domain $D(A) = H^2(\mathbb{R}^m)$ is mα-dissipative with $\alpha = 0$ and the second operator is a bounded operator, we have no limitation on the coefficients β. Since Assumption 6.2.2 is clearly satisfied, Theorem 6.2.4 applies. The second example is obtained by taking for even j, $A_j = c\Delta$ with $\operatorname{Re} c > 0$ (Ginzburg-Landau equation) with domain $D(A_j) = H^2(\mathbb{R}^m)$, and $A_j = V$ for odd j. In this case, either all the coefficients β_j are real for even j and no restrictions are imposed or we have to impose that $|\arg(c\beta_j)| \leq \pi/2$ for even j.

6.3 Numerical comparison of splitting methods

We consider the scalar equation in one-dimension

$$u_t = \Delta u + F(u)$$

where $F(u)$ is a non-linear reaction term. For the purpose of testing our methods, we take Fisher's potential

$$F(u) = u(1 - u).$$

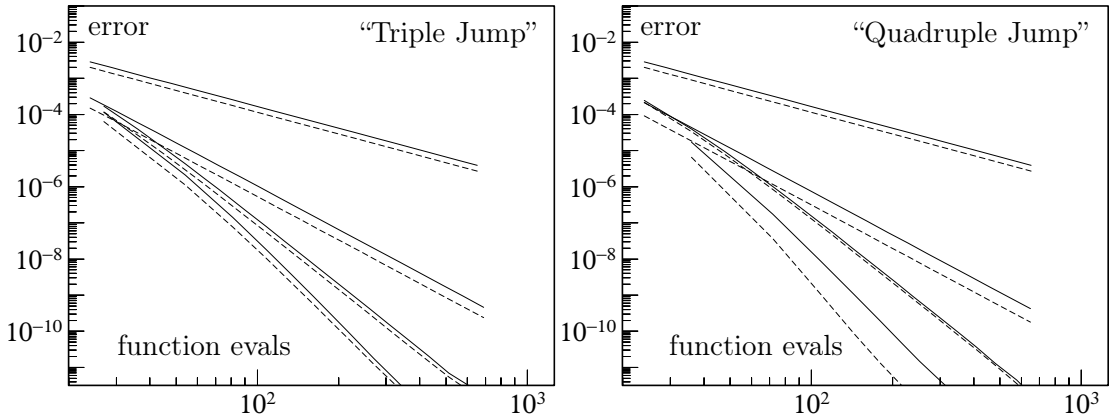


Figure 6.3: Plot: Error of composition methods versus number of evaluations of the basic method Φ_h. Strang splitting and "triple jump" composition methods $\Phi_h^{[p]}$, $p = 4, 6, 8$ in (6.15) (left picture) and "quadruple jump" composition methods $\Psi_h^{[p]}$, $p = 4, 6, 8$ in (6.16) (right picture). Solide lines: Basic method is the Strang splitting with exponential maps (6.5). Dashed lines: Pieceman-Rachford formula (6.9).

The differential equation

$$\frac{\partial u}{\partial z} = u(1-u), \qquad u(0) = u_0$$

can be solved analytically as

$$u(z) = u_0 + u_0(1-u_0)\frac{(e^z-1)}{1+u_0(e^z-1)},$$

which is well defined for small complex time z. For the numerical experiments, we search $u(x,t)$ as a periodic function on the interval $[0,1]$. After discretization in space, we arrive at the differential equation

$$\dot{u} = Au + F(u). \tag{6.22}$$

Vector $u(t)$ belongs to $\mathbb{R}^N$, and has the form

$$u(t) = (u^1(t), \ldots, u^N(t))$$

where $u^j(t)$ is an approximation on the space grid. The Laplacian Δ is approximated by the matrix A of size $N \times N$ given by

$$A = (N+1)^2 \begin{pmatrix} -2 & 1 & & & 1 \\ 1 & -2 & 1 & & \\ & 1 & -2 & 1 & \\ & & \ddots & \ddots & \ddots \\ 1 & & & 1 & -2 \end{pmatrix}.$$

Notice that the vector $F(u)$ is now defined by

$$F(u) = \big(u^1(1-u^1), \ldots, u^N(1-u^N)\big).$$

We take the C^∞ initial condition $u_0(x) = \sin(2\pi x)$, and consider a spatial discretization with $N = 100$ points. In Figure 6.3, we compare the accuracy of the composition methods introduced in this article ("triple" (6.15) and "quadruple" (6.16) jump compositions) on the

time interval $[0, T]$, where $T = 0.2$. We plot for many stepsizes the solution error at time T as a function of the number of evaluations of the basic method. As basic method, we consider (in solide lines) alternatively the Strang splitting (6.5) involving exponentials (i.e. exact flows)

$$\Phi_h = e^{h/2\,F} e^{h\Delta} e^{h/2\,F}$$

and (in dashed lines) the Pieceman-Rachford formula (6.9)

$$\Phi_h = \Phi^I_{h/2} \circ \Phi^M_h \circ \Phi^E_{h/2}.$$

The 'exact' solution is computed with a very small stepsize. We observe the expected orders (lines of slopes $2, 4, 6, 8$). Surprisingly, composition methods using the Pieceman-Rachford formula are slightly more accurate than the one using exponentials.

In Figure 6.4, we compare the "quadruple jump" composition method of order 4 with two extrapolation methods. We also give the results for the Strang splitting of order 2. We use the same initial data and parameters as before. The first extrapolation formula we consider is

$$\frac{4}{3}\Phi_{h/2} \circ \Phi_{h/2} - \frac{1}{3}\Phi_h \tag{6.23}$$

where for the basic method Φ_h, we take alternatively the Strang splitting with exponential maps (6.5), see left picture in Figure 6.4, and the Pieceman-Rachford formula (6.9), see right picture. However, as pointed out in [Sch02, Sect. 6], this scheme is not stable and does not converge in the second case (see dashed-dotted line in right picture). Another extrapolation method is considered in [Sch02] and taken from [Dia96],

$$\frac{45}{64}\Phi_{h/3} \circ \Phi_{h/3} \circ \Phi_{h/3} + \frac{1}{2}\Phi_{h/2} \circ \Phi_{h/2} - \frac{13}{64}\Phi_h. \tag{6.24}$$

Although the formal order of this method is 4, it is said in [Sch02] that the true order of convergence of this method is not clearly understood, and in the numerical experiments for linear problems in [Dia96], "the formal order is not reached ; the experimental precision is smaller than the theoretical precision, and the difference is smaller that 1".

Finally, for a fair comparison in Figure 6.4, it should be mentioned that computations using complex numbers are actually about four times more expensive than computations with reals numbers (because of the cost of a multiplication).

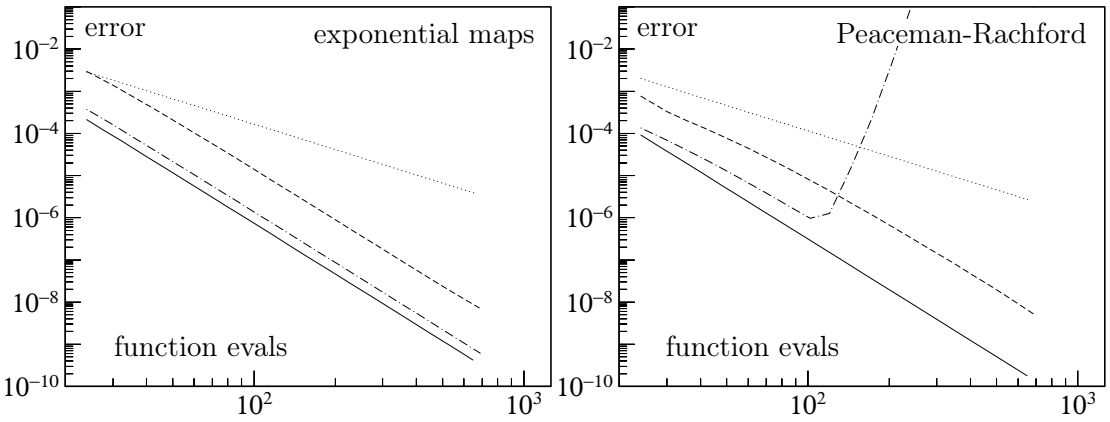


Figure 6.4: Plot: Error versus number of evaluations of the basic method Φ_h. Strang splitting (dotted lines), and "quadruple jump" composition method $\Psi^{[4]}_h$ (6.16) (solid line), extrapolation method (6.23) (dashed-dotted line), extrapolation method (6.24) (dashed lines). Left picture: Basic method is the Strang splitting with exponential maps (6.5). Right picture: Pieceman-Rachford formula (6.9).

Conclusion

We have constructed new high-order compositions methods and splitting methods using complex coefficients for parabolic linear and non-linear parabolic partial differential equations. Based on the results of Hansen & Osterman [HO08a], a convergence analysis is provided in the linear case.

Notice that it is also possible to construct high-order splitting methods involving complex coefficients for only one operator. For instance, the following splitting method is symmetric and of order 4,

$$e^{b_1 hV} e^{a_1 A} e^{b_2 hV} e^{a_2 A} e^{b_3 hV} e^{a_2 A} e^{b_2 hV} e^{a_1 A} e^{b_1 hV} \tag{6.25}$$

where $b_1 = 1/10 - i/30$, $b_2 = 4/15 + 2i/15$, $b_3 = 4/15 - i/5$ are complex, and $a_1 = a_2 = a_3 = a_4 = 1/4$ are all reals. This type of splitting method is of great interest in the case where one operator has its eigenvalues close to the imaginary axis, e.g. the Ginzburg-Landau equation (6.2) with large $|\arg \alpha|$ (close to $\pi/2$).

A systematic study of optimal composition methods (i.e. methods with optimal error constants) is out of the scope of this paper and will be the subject of a future article by the same authors. It requires the resolution in $\mathbb{C}$ of the polynomial systems of order conditions for composition methods and splitting methods. Also, a theoretical analysis in the case of a non-linear source is in preparation.

Appendix A

Maple script for the modified moments of inertia

Using a symbolic manipulation package like MAPLE, the functions in Table 3.1 can be computed formally by comparing the Taylor series of the exact solution of (3.1), recursively with the series expansion of the DMV algorithm applied with modified moments of inertia. A MAPLE script for this computation is the following:

```
# MAPLE SCRIPT
> with(linalg): Order := 8:
# Modified moments of inertia
> s := 1+h^2*s3+h^4*s5+h^6*s7:
> d := h^2*d3+h^4*d5+h^6*d7:
> i1mod := 1/(s/i1+d): i2mod := 1/(s/i2+d): i3mod := 1/(s/i3+d):
# SERIES EXPANSION OF THE NUMERICAL SOLUTION (DMV)
> e1 := 0: e2 := 0: e3 := 0:
> for i from 1 to 13 do
>   alpha := series(1+e1^2+e2^2+e3^2,h);
>   e1 := series(alpha*h/2*y1/i1mod+(i2mod-i3mod)/i1mod*e2*e3,h);
>   e2 := series(alpha*h/2*y2/i2mod+(i3mod-i1mod)/i2mod*e3*e1,h);
>   e3 := series(alpha*h/2*y3/i3mod+(i1mod-i2mod)/i3mod*e1*e2,h);
> od:
> y1dmv := series(y1+4/h/alpha*(i2mod-i3mod)*e2*e3,h):
# Cayley transform
> Id := matrix(3,3,[1,0,0,0,1,0,0,0,1]):
> ehat := matrix(3,3,[0,-e3,e2,e3,0,-e1,-e2,e1,0]):
> Qdmv := evalm((Id+ehat)&*inverse((Id-ehat))):
# SERIES EXPANSION OF THE EXACT SOLUTION
> fncy := y1(t),y2(t),y3(t):
> fncQ := q11(t),q12(t),q13(t),q21(t),q22(t),q23(t),q31(t),q32(t),q33(t):
> Qexact := matrix(3,3,[fncQ]):
# Equations of motion
> eqy := diff(y1(t),t)=y2(t)*y3(t)*(i2-i3)/i2/i3,
>        diff(y2(t),t)=y1(t)*y3(t)*(i3-i1)/i1/i3,
>        diff(y3(t),t)=y1(t)*y2(t)*(i1-i2)/i2/i1:
> W := matrix(3,3,[0, -y3(t)/i3, y2(t)/i2,
>                  y3(t)/i3, 0, -y1(t)/i1,
>                  -y2(t)/i2, y1(t)/i1, 0]):
> QW := evalm(Qexact&*W):
> eqQ := seq(seq(diff(Qexact[i,j],t)=QW[i,j],j=1..3),i=1..3):
# Initial condition
> init := q11(0)=1,q12(0)=0,q13(0)=0,
>         q21(0)=0,q22(0)=1,q23(0)=0,
>         q31(0)=0,q32(0)=0,q33(0)=1,
>         y1(0)=y1,y2(0)=y2,y3(0)=y3:
# Exact solution
> assign(dsolve({eqy,eqQ,init},{fncy,fncQ},type=series)):
> y1exact := subs(t=h,y1(t)):
> Qexact := simplify(subs(t=h,matrix(3,3,[fncQ]))):
# LOCAL ERROR
> erry := simplify(series(y1exact-y1dmv,h)):
> errQ := simplify(series(Qexact[3,2]-Qdmv[3,2],h)):
```

```
# COMPUTATION OF s3,d3,s5,d5,...
> sol := proc(coeff,err,n)
> solve(convert(series(err,h,n),polynom),coeff):
> end:
> s3 := sol(s3,erry,4): d3 := sol(d3,errQ,4):
> s5 := sol(s5,erry,6): d5 := sol(d5,errQ,6):
> s7 := sol(s7,erry,8): d7 := sol(d7,errQ,8):
# DECOMPOSITION as polynomials in H(y) and C(y)
> C:=(y1^2+y2^2+y3^2)/2: H:=(y1^2/i1+y2^2/i2+y3^2/i3)/2:
> decomp := proc(expr,vars)
> solve({subs({y1=1,y2=0,y3=0},expr),subs({y1=0,y2=1,y3=0},expr),
>        subs({y1=0,y2=0,y3=1},expr),subs({y1=1,y2=1,y3=1},expr)},vars);
> end:
> decomp(s3=a1*C+a2*H,{a1,a2});
       /       i2 i3 + i1 i3 + i1 i2       i2 + i1 + i3\
      { a2 = - ---------------------, a1 = ------------ }
       \            3 i1 i2 i3              6 i1 i2 i3 /
> decomp(s5=a1*C^2+a2*C*H+a3*H^2,{a1,a2,a3});
```

Appendix B

Fortran code for the Preprocessed Discrete Moser–Veselov algorithm of order 10

```
      SUBROUTINE DMV10 (AM,Q,POTENP,H,NSTEP,RPAR,IPAR)
C-----------------------------------------------------------------------
C            PREPROCESSED DISCRETE MOSER-VESELOV ALGORITHM
C-----------------------------------------------------------------------
C  PREPROCESSED DISCRETE MOSER-VESELOV ALGORITHM OF ORDER 10 FOR THE
C  NUMERICAL SOLUTION OF THE EQUATIONS OF MOTION OF THE FREE RIGID BODY.
C  ORTHOGONAL MATRICES ARE REPRESENTED BY QUATERNIONS.
C  THE CODE IS READY TO INCLUDE AN EXERNAL POTENTIAL (SYMMETRIC STRANG
C  SPLITTING OF ORDER 2).
C-----------------------------------------------------------------------
C     AUTHORS: ERNST HAIRER (1) AND GILLES VILMART (1)(2)
C                (1) UNIVERSITE DE GENEVE, DEPT. DE MATHEMATIQUES
C                    2-4 RUE DU LIEVRE, CASE POSTALE 64
C                    CH-1211 GENEVE 4, SWITZERLAND
C                (2) IRISA/INRIA RENNES, PROJET IPSO
C                    CAMPUS DE BEAULIEU, F-35042 RENNES CEDEX, FRANCE
C     E-MAILS: Ernst.Hairer@math.unige.ch
C              Gilles.Vilmart@math.unige.ch
C
C  THIS CODE IS DESCRIBED IN:
C    E. HAIRER AND G. VILMART, PREPROCESSED DISCRETE MOSER-VESELOV
C    ALGORITHM FOR THE FULL DYNAMICS OF A RIGID BODY
C    J. Phys. A: Math. Gen. 39 (2006) 13225-13235.
C    http://stacks.iop.org/0305-4470/39/13225
C
C     VERSION: AUGUST 29, 2006
C     (latest correction of a small bug: December 28, 2007)
C-----------------------------------------------------------------------
C     INPUT PARAMETERS
C     ----------------
C     AM(I)      INITIAL ANGULAR MOMENTUM (I=1,2,3)
C     Q(I)       INITIAL QUATERNION FOR ORTHOGONAL MATRIX (I=1,2,3,4)
C     H          STEP SIZE
C     NSTEP      NUMBER OF STEPS
C     POTENP     NAME (EXTERNAL) OF SUBROUTINE FOR AN EXTERNAL POTENTIAL
C                   SUBROUTINE POTENP(Q,POTP,RPAR,IPAR)
C                   DIMENSION Q(4),POTP(3)
C                   POTP(1)=...   ETC.
C     RPAR,IPAR  REAL AND INTEGER PARAMETERS (OR PARAMETER ARRAYS) WHICH
C                CAN BE USED FOR COMMUNICATION BETWEEN SUBROUTINES
C     RPAR(11), RPAR(12), RPAR(13) ARE THE THREE MOMENTS OF INERTIA OF
C                THE RIGID BODY
C
C     OUTPUT PARAMETERS
C     -----------------
```

```
C     AM(I)       SOLUTION (ANGULAR MOMENTUM) AT ENDPOINT
C     Q(I)        SOLUTION (QUATERNION) AT ENDPOINT
C-----------------------------------------------------------------------
      PARAMETER (MSPLIT=1)
C-----------------------------------------------------------------------
      IMPLICIT REAL*8 (A-H,O-Z)
      DOUBLE PRECISION Q(4),AM(3),POTP(3)
      DIMENSION IPAR(20),RPAR(20)
      EPS=ABS(1.D-15*H)
      HA=H/2.0D0
      HB=H/MSPLIT
C
      HD=HB/2.0D0
      HC=4.0D0/HB
      AI1=RPAR(11)
      AI2=RPAR(12)
      AI3=RPAR(13)
C---      CONSTANT COEFFICIENTS FOR THE MODIFIED MOMENTS OF INERTIA
      XI1=1.0D0/AI1+1.0D0/AI2+1.0D0/AI3
      XI2=1.0D0/AI1**2+1.0D0/AI2**2+1.0D0/AI3**2
      XI3=1.0D0/AI1**3+1.0D0/AI2**3+1.0D0/AI3**3
      XI4=1.0D0/AI1**4+1.0D0/AI2**4+1.0D0/AI3**4
      XDET=AI1*AI2*AI3
      XDET2=XDET**2
      XDET3=XDET**3
      XDET4=XDET2**2
      XS1=AI1+AI2+AI3
      XS2=AI1**2+AI2**2+AI3**2
      XS3=AI1**3+AI2**3+AI3**3
      XS4=AI1**4+AI2**4+AI3**4
      XSI=(AI1+AI2)/AI3+(AI2+AI3)/AI1+(AI3+AI1)/AI2
      XSI12=(AI1+AI2)/AI3**2+(AI2+AI3)/AI1**2+(AI3+AI1)/AI2**2
      XSI21=(AI1**2+AI2**2)/AI3+(AI2**2+AI3**2)/AI1+(AI3**2+AI1**2)/AI2
      XSI13=(AI1+AI2)/AI3**3+(AI2+AI3)/AI1**3+(AI3+AI1)/AI2**3
      XSI31=(AI1**3+AI2**3)/AI3+(AI2**3+AI3**3)/AI1+(AI3**3+AI1**3)/AI2
      XSI22=(AI1**2+AI2**2)/AI3**2+(AI2**2+AI3**2)/AI1**2
     &      +(AI3**2+AI1**2)/AI2**2
C
      S3C1=XS1/(6.0D0*XDET)
      S3C2=-XI1/3.0D0
      D3C1=-1.0D0/(3.0D0*XDET)
      D3C2=S3C1
C
      S5C1=(XS2/XDET-XI1)/(30.0D0*XDET)
      S5C2=(1.0D0-XSI)/(30.0D0*XDET)
      S5C3=(3.0D0*XS1/XDET+2.0D0*XI2)/60.0D0
      D5C1=-XS1/XDET2/60.0D0
      D5C2=XI1/XDET/10.0D0-XS2/XDET2/60.0D0
      D5C3=-(9.0D0+XSI)/XDET/60.0D0
C
      S7C1=(4.0D0*XDET+17.0D0*XS3-15.0D0*XDET*XSI)/(2520.0D0*XDET3)
      S7C2=(9.0D0*XS1+10.0D0*XDET*XI2-6.0D0*XSI21)/(420.0D0*XDET2)
      S7C3=((6.0D0*XSI12-1.0D2*XI1)*XDET+53.0D0*XS2)/(2520.0D0*XDET2)
      S7C4=(15.0D0-XDET*XI3-2.0D0*XSI)/(630.0D0*XDET)
      D7C1=(34.0D0*XDET*XI1-19.0D0*XS2)/(2520.0D0*XDET3)
      D7C2=(XS3+2.0D0*XDET*XSI-85.0D0*XDET)/(1260.0D0*XDET3)
      D7C3=(47.0D0*XS1+13.0D0*XSI21-38.0D0*XDET*XI2)/(2520.0D0*XDET2)
      D7C4=(9.0D0*XDET*XI1+XDET*XSI12-11.0D0*XS2)/(1260.0D0*XDET2)
C
      S9C1=(62.0D0*XS4-94.0D0*XDET*XSI21
     &     +66.0D0*XDET2*XI2+81.0D0*XDET*XS1)/(45360.0D0*XDET4)
      S9C2=(-77.0D0*XSI31+75.0D0*XDET*XSI12
     &     +214.0D0*XS2-240.0D0*XDET*XI1)/(22680.0D0*XDET3)
      S9C3=(26.0D0*XDET*XSI22+55.0D0*XS3+204.0D0*XDET
     &     -50.0D0*XDET2*XI3-59.0D0*XDET*XSI)/(7560.0D0*XDET3)
      S9C4=(137.0D0*XDET*XI2-XDET*XSI13
     &     +3.0D0*XS1-69.0D0*XSI21)/(11340.0D0*XDET2)
      S9C5=(2.0D0*XDET2*XI4+5.0D0*XDET*XSI12
     &     -171.0D0*XDET*XI1+159.0D0*XS2)/(45360.0D0*XDET2)
      D9C1=(60.0D0*XSI*XDET-61.0D0*XS3-247.0D0*XDET)/(45360.0D0*XDET4)
```

```
      D9C2=(54.0D0*XDET*XS1-XS4+218.0D0*XDET*XSI21
     &     -426.0D0*XDET2*XI2)/(45360.0D0*XDET4)
      D9C3=(125.0D0*XDET*XI1-5.0D0*XSI31-130.0D0*XS2
     &     +4.0D0*XDET*XSI12)/(7560.0D0*XDET3)
      D9C4=(67.0D0*XS3-735.0D0*XDET-15.0D0*XDET*XSI22
     &     +87.0D0*XDET*XSI+34.0D0*XDET2*XI3)/(22680.0D0*XDET3)
      D9C5=(165.0D0*XS1-XDET*XSI13-9.0D0*XSI21
     &     -145.0D0*XDET*XI2)/(45360.0D0*XDET2)
c---
      CALL POTENP(Q,POTP,RPAR,IPAR)
      AM1=AM(1)-HA*POTP(1)
      AM2=AM(2)-HA*POTP(2)
      AM3=AM(3)-HA*POTP(3)
c
      DO ISTEP=1,NSTEP
C ---      COMPUTATION OF THE MODIFIED MOMENTS OF INERTIA
        HAM0=0.5D0*HB**2*(AM1**2/AI1+AM2**2/AI2+AM3**2/AI3)
        ANOR0=0.5D0*HB**2*(AM1**2+AM2**2+AM3**2)
        HAM2=HAM0**2
        ANOR2=ANOR0**2
        HAM3=HAM0**3
        ANOR3=ANOR0**3
        HAM4=HAM2**2
        ANOR4=ANOR2**2
        ANORHAM=HAM0*ANOR0
        ANOR2HAM=HAM0*ANOR2
        ANORHAM2=HAM2*ANOR0
        ANOR3HAM=HAM0*ANOR3
        ANOR2HAM2=HAM2*ANOR2
        ANORHAM3=HAM3*ANOR0
        CSS=1.0D0+(S3C1*ANOR0+S3C2*HAM0)
     &     +(S5C1*ANOR2+S5C2*ANORHAM+S5C3*HAM2)
     &     +(S7C1*ANOR3+S7C2*ANOR2HAM+S7C3*ANORHAM2+S7C4*HAM3)
     &     +(S9C1*ANOR4+S9C2*ANOR3HAM+S9C3*ANOR2HAM2
     &     +S9C4*ANORHAM3+S9C5*HAM4)
        CDD=(D3C1*ANOR0+D3C2*HAM0)
     &     +(D5C1*ANOR2+D5C2*ANORHAM+D5C3*HAM2)
     &     +(D7C1*ANOR3+D7C2*ANOR2HAM+D7C3*ANORHAM2+D7C4*HAM3)
     &     +(D9C1*ANOR4+D9C2*ANOR3HAM+D9C3*ANOR2HAM2
     &     +D9C4*ANORHAM3+D9C5*HAM4)
        AI1MODI=CSS/AI1+CDD
        AI2MODI=CSS/AI2+CDD
        AI3MODI=CSS/AI3+CDD
        AI1MOD=1.0D0/AI1MODI
        AI2MOD=1.0D0/AI2MODI
        AI3MOD=1.0D0/AI3MODI
        FAD1=AI2MOD-AI3MOD
        FAD2=AI3MOD-AI1MOD
        FAD3=AI1MOD-AI2MOD
        FAC1=FAD1*AI1MODI
        FAC2=FAD2*AI2MODI
        FAC3=FAD3*AI3MODI
        DO ISPLIT=1,MSPLIT
C ---      SOLVE FOR INTERNAL STAGE
          AM1I=AM1*HD*AI1MODI
          AM2I=AM2*HD*AI2MODI
          AM3I=AM3*HD*AI3MODI
          CM1=AM1I+FAC1*AM2I*AM3I
          CM2=AM2I+FAC2*CM1*AM3I
          CM3=AM3I+FAC3*CM1*CM2
          DO I=1,50
            CM1B=CM1
            CM2B=CM2
            CM3B=CM3
            CALPHA=1+CM1**2+CM2**2+CM3**2
            CM1=CALPHA*AM1I+FAC1*CM2*CM3
            CM2=CALPHA*AM2I+FAC2*CM1*CM3
            CM3=CALPHA*AM3I+FAC3*CM1*CM2
            ERR=ABS(CM1B-CM1)+ABS(CM2B-CM2)+ABS(CM3B-CM3)
            IF (ERR.LT.EPS) GOTO 22
```

```
          END DO
 22       CONTINUE
C ---      UPDATE Q
          Q0=Q(1)
          Q1=Q(2)
          Q2=Q(3)
          Q3=Q(4)
          Q(1)=Q0-CM1*Q1-CM2*Q2-CM3*Q3
          Q(2)=Q1+CM1*Q0+CM3*Q2-CM2*Q3
          Q(3)=Q2+CM2*Q0+CM1*Q3-CM3*Q1
          Q(4)=Q3+CM3*Q0+CM2*Q1-CM1*Q2
C ---       UPDATE M
          CALPHA=HC/CALPHA
          AM1=AM1+FAD1*CM2*CM3*CALPHA
          AM2=AM2+FAD2*CM1*CM3*CALPHA
          AM3=AM3+FAD3*CM1*CM2*CALPHA
        END DO
C ---       PROJECTION
        QUAT=1.0D0/SQRT(Q(1)**2+Q(2)**2+Q(3)**2+Q(4)**2)
        Q(1)=Q(1)*QUAT
        Q(2)=Q(2)*QUAT
        Q(3)=Q(3)*QUAT
        Q(4)=Q(4)*QUAT
C
        CALL POTENP(Q,POTP,RPAR,IPAR)
        AM1=AM1-H*POTP(1)
        AM2=AM2-H*POTP(2)
        AM3=AM3-H*POTP(3)
      END DO
      AM(1)=AM1+HA*POTP(1)
      AM(2)=AM2+HA*POTP(2)
      AM(3)=AM3+HA*POTP(3)
      RETURN
      END

C * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C --- DRIVER FOR DMV10 FOR THE MOTION OF A RIGID BODY
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
        include 'dmv10.f'
        IMPLICIT DOUBLE PRECISION (A-H,O-Z)
        DIMENSION Q(4),AM(3),RPAR(20),IPAR(20)
        REAL TIME0,TIME1
        EXTERNAL POTENP
C --- INITIAL VALUES
C ANGULAR MOMENTUM
        AM(1)=1.8D0
        AM(2)=0.4D0
        AM(3)=-0.9D0
C QUATERNION
        Q(1)=1.0D0
        Q(2)=0.0D0
        Q(3)=0.0D0
        Q(4)=0.0D0
C --- MOMENTS OF INERTIA
        AI1=0.6D0
        AI2=0.8D0
        AI3=1.0D0
        RPAR(11)=AI1
        RPAR(12)=AI2
        RPAR(13)=AI3
C ---
        H=0.01
        XEND=10.0D0
        NSTEP=XEND/H
        H=XEND/NSTEP
        WRITE (6,*) 'XEND=',XEND,' H=',H,' NSTEP=',NSTEP
C ---
        DO I=1,10
          RPAR(I)=0.0D0
          IPAR(I)=0
```

```
        END DO
C
        WRITE (6,*) '--- INITIAL CONDITION'
        WRITE (6,*) '  AM    ',AM(1),AM(2),AM(3)
        WRITE (6,*) '  QQ    ',Q(1),Q(2),Q(3),Q(4)
        CALL HAMIL(Q,AM,HAM0,RPAR,IPAR)
        CALL CPU_TIME(TIME0)
C ---
        CALL DMV10 (AM,Q,POTENP,H,NSTEP,RPAR,IPAR)
C ---
        CALL CPU_TIME(TIME1)
        CALL HAMIL(Q,AM,HAM1,RPAR,IPAR)
        WRITE (6,*) '--- SOLUTION AT ENDPOINT'
        WRITE (6,*) '  AM    ',AM(1),AM(2),AM(3)
        WRITE (6,*) '  QQ    ',Q(1),Q(2),Q(3),Q(4)
        WRITE (6,*) 'ERR HAM=',(HAM1-HAM0)/HAM0,' TIME=',TIME1-TIME0
        STOP
        END
C
        SUBROUTINE HAMIL (Q,AM,HAM,RPAR,IPAR)
        IMPLICIT REAL*8 (A-H,O-Z)
        DIMENSION Q(4),AM(3)
        DIMENSION IPAR(*),RPAR(*)
        CALL POTEN(Q,POT,RPAR,IPAR)
        HAM=AM(1)**2/RPAR(11)+AM(2)**2/RPAR(12)+AM(3)**2/RPAR(13)
        HAM=HAM/2.0D0+POT
        RETURN
        END
c
        SUBROUTINE POTEN(Q,POT,RPAR,IPAR)
        IMPLICIT REAL*8 (A-H,O-Z)
        DIMENSION Q(4)
        DIMENSION IPAR(*),RPAR(*)
        POT=Q(1)**2-Q(2)**2-Q(3)**2+Q(4)**2
        POT=0.0D0
        RETURN
        END
c
        SUBROUTINE POTENP(Q,POTP,RPAR,IPAR)
        IMPLICIT REAL*8 (A-H,O-Z)
        DIMENSION Q(4),POTP(3)
        DIMENSION IPAR(*),RPAR(*)
        POTP(1)=-2*(Q(1)*Q(2)+Q(3)*Q(4))
        POTP(2)=-2*(Q(1)*Q(3)-Q(2)*Q(4))
        POTP(1)=0.0d0
        POTP(2)=0.0d0
        POTP(3)=0.0d0
        RETURN
        END
```

Appendix C

Exact resolution of the two–body Kepler problem

In several recent publications [MN02, MN04, Koz07], conservative approximations for the exact numerical resolution of the 2-body Kepler problem in $\mathbb{R}^3$ are proposed,

$$\dot{q} = p, \quad \dot{p} = -\frac{K}{|q|^3} q, \qquad q(t_0) = q_0, \qquad p(t_0) = p_0. \tag{C.1}$$

for some constant $K > 0$. These discretizations exactly conserve all first integrals, the angular momentum, the Hamiltonian and Runge-Lenz vector, and are based on the Kustaanheimo–Stiefel transformation (1965), introduced in [KS65], which links the Kepler problems to the four-dimensional harmonic oscillator.

However, an efficient implementation of the exact solution of the 2-body Kepler problem (C.1) using this KS transformation does not seem documented in the literature[1].

Algorithm C.0.1 (Exact solution of Kepler problem (C.1), elliptic trajectory)

1. Compute the scalars

$$r = |q_0|, \qquad P = \frac{|p_0|^2}{2}, \qquad M = q_0^T p_0, \qquad A = \frac{K}{r} - P > 0, \qquad \lambda = \sqrt{2A}.$$

2. Compute x from the implicit relation

$$f(x) = K \arctan(x) + \frac{a_1 x + a_2 x^2 - a_1 x^3}{(1+x^2)^2} + a_3 = 0 \tag{C.2}$$

   where

$$a_1 = r(A - P), \qquad a_2 = 2\lambda M, \qquad a_3 = -A\lambda h/2.$$

   This can be done with a few Newton iterations $x_{n+1} = x_n - f(x_n)/f'(x_n)$.

3. The solution of (C.1) at time $t = t_0 + h$ is given by updating q_0 and p_0 (compensated summation can be applied, see Sect. 3.5.2):

$$q_1 = q_0 + (d_1 q_0 + d_2 p_0), \qquad p_1 = p_0 + (d_3 q_0 + d_4 p_0), \tag{C.3}$$

[1] In fact, the algorithm proposed in [Koz07] does not give the solution of the Kepler problem (C.1) after a fixed time of integration h. It gives the exact solution after a time which is not known *a priori* and depends on the initial conditions.

where

$$d_1 = -s^2(1 + P/A), \qquad d_2 = (cs\lambda r + s^2 M)/A,$$
$$d_3 = -cs\lambda(1 + P/A)/|q_1|, \qquad d_4 = rd_1/|q_1|$$
$$|q_1| = r + rs^2(P/A - 1) + cs\lambda M/A,$$

and we use

$$c = (1 - x^2)/(1 + x^2) \qquad s = 2x/(1 + x^2). \tag{C.4}$$

Proposition C.0.2 *For stepsize h smaller than half a period of elliptic motion (similar formulas hold for parabolic or hyperbolic trajectories), Algorithm C.0.1 yields the exact solution of the 2-body Kepler problem (C.1). Also, the nonlinear equation (C.2) processes a unique solution.*

Proof. Using the Kustaanheimo-Stiefel transformation $(q, p) \in \mathbb{R}^3 \times \mathbb{R}^3 \mapsto (Q, P) \in \mathbb{R}^4 \times \mathbb{R}^4$ as introduced in [KS65], the Kepler problem (C.1) can be transformed into the 4-dimensional harmonic oscillator with a transformed time $t(s)$:

$$\frac{\partial Q(s)}{\partial s} = \frac{1}{4} P(s) \qquad \frac{\partial P(s)}{\partial s} = -2A\, Q(s) \qquad \frac{\partial t(s)}{\partial s} = |Q(s)|^2,$$

where we use the notations of [Koz07]. The derivation of the exact solution of this system as a function of $\cos(\lambda s)$, and $\sin(\lambda s)$ is straightforward, and all formulas can be expressed in terms of (q, p) in (C.3). Then, the main idea is to put $x = \tan(\lambda s_0/2)$ where s_0 is defined implicitly by $t(s_0) - t_0 = h$. This allows to express the quantities $c = \cos(\lambda s_0)$, and $s = \sin(\lambda s_0)$ as rational functions of x (C.4), and relation $t(s_0) - t_0 = h$ is equivalent to (C.2). Since the solution of $t(s_0) - t_0 = h$ is unique ($t(s)$ is an increasing function), the non linear equation (C.2) possesses a unique solution and the algorithm is valid as long as x is well defined, i.e. $\lambda s_0 < \pi$, or equivalently the stepsize h is lower than half a period. □

Annexe D

Résumé de la thèse en français

Le sujet de la thèse est l'étude et la construction de méthodes numériques géométriques pour les équations différentielles, c'est-à-dire qui préservent des propriétés géométriques du flot exact, notamment la symétrie, la symplecticité des systèmes hamiltoniens, la conservation d'intégrales premières, la structure de Poisson, etc. Ce mémoire s'articule en trois parties étroitement liées.

Dans la première partie (Chapitres 1, 2 et 3), on introduit une nouvelle approche de construction d'intégrateurs numériques géométriques d'ordre élevé, en s'inspirant de la théorie des équations différentielles modifiées (backward error analysis). Le cas des méthodes développables en B-séries est spécifiquement analysé, et on introduit une nouvelle loi de composition sur les B-séries. L'efficacité de cette approche est illustrée par la construction d'un nouvel intégrateur géométrique d'ordre élevé pour les équations du mouvement d'un corps rigide. On obtient également une méthode numérique précise pour le calcul de points conjugués pour les géodésiques du corps rigide.

Dans la seconde partie (Chapitre 4), on étudie dans quelle mesure les excellentes performances des méthodes symplectiques, pour l'intégration à long terme en astronomie et en dynamique moléculaire, persistent pour les problèmes de contrôle optimal. On discute également l'extension de la théorie des équations modifiées aux problèmes de contrôle optimal.

La dernière partie (Chapitres 5 et 6) est dédiée aux méthodes de pas fractionnaire (splitting en anglais). Dans le même esprit que les équations modifiées, on considère des méthodes de splitting pour les systèmes hamiltoniens perturbés, qui font intervenir des potentiels modifiés. On termine par la construction de méthodes de splitting d'ordre élevé avec coefficients complexes pour les équations aux dérivées partielles paraboliques, notamment les problèmes de réaction-diffusion en chimie.

Chapitre 1 En s'inspirant de la théorie des équations modifiées (backward error analysis), on présente une nouvelle approche pour la construction d'intégrateurs géométriques d'ordre élevé pour les équations différentielles ordinaires. On l'appelle 'intégrateur à champs de vecteurs modifiés', car on modifie le champ de vecteurs avant d'appliquer la méthode. Cette approche est illustrée avec la règle du point milieu appliquée à la dynamique complète des équations du corps rigide libre. On accorde une attention particulière aux méthodes développables en B-séries pour lesquelles des formules explicites pour les équations différentielles modifiées sont données. Une nouvelle loi de composition sur les B-séries, appelée loi de substitution, est présentée.

Chapitre 2 On présente la structure algébrique commune de deux lois de composition sur les B-séries : la composition de Butcher, qui correspond à la composition du flot des intégrateurs, et la loi de substitution introduite au chapitre précédent, qui correspond à la composition de champs de vecteurs de B-séries. Les structures d'algèbre de Hopf sur les arbres racinés sont un objet bien étudié, particulièrement en combinatoire, et sont caractérisées essentiellement par une loi de coproduit. Il est bien connu que la première loi de composition correspond au produit de convolution sur l'algèbre de Hopf d'arbres de Connes & Kreimer pour la renormalisation en théorie des champs quantiques. Il a été démontré récemment que la loi de substitution peut être vue comme un coproduit, permettant de construire une nouvelle algèbre de Hopf d'arbres. On explique leur relation algébrique du point de vue de l'intégration numérique géométrique.

Chapitre 3 Comme application de l'idée des intégrateurs des champs de vecteurs modifiés, on construit un intégrateur efficace d'ordre élevé pour les équations du mouvement du corps rigide libre. L'algorithme "Discrete Moser-Veselov" est une discrétisation intégrable des équations du mouvement. Il est symplectique et réversible en temps et il conserve toutes les intégrales premières du système. Son seul défaut est son faible ordre de convergence. On présente une modification de cet algorithme jusqu'à un ordre arbitrairement élevé qui a un surcoût négligeable mais améliore considérablement la précision. On étudie également l'accumulation des erreurs d'arrondi au cours du temps et on explique comment la réduire. Enfin, on propose une modification qui permet de calculer le champ tangent, pour le calcul de points conjugués sur les géodésiques de corps rigides.

Chapitre 4 Pour des problèmes de contrôle optimal, le principe du maximum de Pontryagin donne les conditions nécessaires d'optimalité sous la forme d'une équation différentielle hamiltonienne. Pour son intégration numérique, les méthodes symplectiques sont un choix naturel. On étudie dans quelle mesure les excellentes performances des intégrateurs symplectiques pour l'intégration à long terme en astronomie et en dynamique moléculaire, s'étendent aux problèmes de contrôle optimal. Les expériences numériques et l'analyse rétrograde montrent que, pour des problèmes en petite dimension et une trajectoire proche d'une valeur critique de l'hamiltonien, les intégrateurs symplectiques ont un net avantage. On illustre cela avec les cas Martinet en géométrie sous-riemannienne. Pour des problèmes comme le transfert orbital d'un satellite ou le contrôle d'un corps rigide sous-marin, un tel avantage n'est pas observé. Le système hamiltonien est un problème aux deux bouts et l'intervalle de temps n'est en général pas assez grand pour que les intégrateurs symplectiques puissent bénéficier de leur préservation structurelle du flot. On discute également l'extension éventuelle de la théorie de l'analyse rétrograde pour les intégrateurs symplectiques aux problèmes de contrôle optimal.

Chapitre 5 On étudie des méthodes de splitting pour des systèmes hamiltoniens qui utilisent des potentiels modifiés contenant des crochets de Lie. On montre que cette approche, initialement développée pour des équations différentielles d'ordre 2 (par exemple, problèmes à N corps en coordonnées de Jacobi), peut être aussi appliquée avec succès à des problèmes de corps rigide asymétrique avec un potentiel externe. On illustre cela avec le corps pesant asymétrique, un modèle de satellites, et une simulation de dynamique moléculaire avec des sphères molles dipolaires. On construit également un nouveau processeur pour la méthode de Takahashi-Imada (une modification de Störmer-Verlet) pour atteindre l'ordre $\mathcal{O}(h^{10}\varepsilon + h^4\varepsilon^2)$ pour des systèmes hamiltoniens perturbés, où h est le pas de temps et ε est la taille de la perturbation. Il se révèle efficace dans de nombreuses situations.

Chapitre 6 Le dernier chapitre est dédié aux méthodes de splitting avec coefficients complexes pour les équations aux dérivées partielles paraboliques linéaires et non-linéaires. Il est connu que toute méthode de splitting avec des coefficients réels d'ordre strictement plus grand que 2 possède nécessairement des coefficients négatifs. Ainsi, ces méthodes avec coefficients réels ne peuvent pas être utilisées lorsqu'un opérateur comme le laplacien Δ n'est pas réversible en temps et ne peut pas être résolu pour des temps négatifs. Pour contourner cette barrière d'ordre, on construit de nouvelles méthodes de splitting d'ordre élevé avec coefficients complexes, en s'appuyant sur les techniques de composition initialement développées pour l'intégration géométrique des équations différentielles ordinaires. On donne une justification théorique de l'ordre de convergence des méthodes introduites dons le cas linéaire avec des opérateurs exponentiels. Nos expériences numériques montrent que l'ordre est bien celui attendu, en particulier dans le cas d'une source non-linéaire, et aussi avec la discrétisation de Peaceman-Rachford comme ingrédient de base.

D.1 Intégration numérique géométrique

Le domaine de l'intégration géométrique étant très vaste, on présente dans cette section les aspects les plus importants de l'intégration numérique géométrique des équations différentielles ordinaires (voir les ouvrages [SSC94, LR04, HLW06]) pour la compréhension de ce travail de thèse. Ces idées sont illustrées par les exemples du problème de Kepler, le problème à trois corps en mécanique céleste, et le problème du pendule asymétrique.

Considérons un système d'équations différentielles[1],

$$\dot{y} = f(y), \qquad y(0) = y_0, \tag{D.1}$$

où $f(y)$ est un champ de vecteurs suffisamment différentiable. La première et la plus simple des méthodes numériques est due à Euler [Eul68] en 1768,

$$y_{n+1} = y_n + hf(y_n).$$

Étant donnée une longueur de pas h, on calcule récursivement des approximations y_1, y_2, $y_3, \ldots$ des valeurs $y(h), y(2h), y(3h), \ldots$ de la solution. On l'appelle méthode d'Euler explicite car le calcul de y_{n+1} est effectué explicitement avec une évaluation du champ de vecteurs f en supposant le vecteur y_n déjà connu. A l'inverse, la méthode d'Euler implicite

$$y_{n+1} = y_n + hf(y_{n+1})$$

requiert la résolution numérique d'un système non linéaire d'équations à chaque pas.

Flot exact On définit le flot (exact) φ_t de l'équation différentielle (D.1) pendant le temps t comme l'application qui, à tout point y_0 de l'espace des phases associe la valeur $y(t)$ de la solution de l'équation différentielle avec la valeur initiale $y(0) = y_0$. Autrement dit,

$$\varphi_t(y_0) = y(t) \quad \text{si} \quad y(0) = y_0.$$

On appelle méthode numérique à un pas une application Φ_h qui approche le flot pour un temps h de l'équation différentielle (D.1).

[1] Il est à noter q'un système non autonome $\dot{y} = f(t, y)$ peut être mis sous cette forme en considérant l'équation supplémentaire $\dot{t} = 1$.

Définition D.1.1 *Une méthode numérique $y_{n+1} = \Phi_h(y_n)$ est d'ordre p pour le problème (D.1) si l'erreur locale satisfait*

$$\Phi_h(y) - \varphi_h(y) = \mathcal{O}(h^{p+1}).$$

On peut facilement vérifier que les méthodes d'Euler explicite, implicite et symplectique sont d'ordre 1, en comparant leur développement en série de Taylor avec celui du flot exact.

Pour atteindre un ordre de précision plus élevé, Runge [Run95] et Heun [Heu00] ont construit il y a plus d'un siècle des méthodes qui comportent plusieurs pas de la méthode d'Euler, et Kutta [Kut01] a ensuite introduit les "méthodes de Runge-Kutta" sous leur forme générale. Par exemple, la méthode

$$\begin{aligned} Y_1 &= y_n \qquad\qquad Y_2 = y_n + \frac{h}{2} f(Y_1) \\ Y_3 &= y_n + \frac{h}{2} f(Y_2) \qquad Y_4 = y_n + h f(Y_3) \\ y_{n+1} &= y_n + \frac{h}{6}\big(f(Y_1) + 2f(Y_2) + 2f(Y_3) + f(Y_4)\big) \end{aligned} \tag{D.2}$$

est connue sous le nom de 'La' méthode de Runge-Kutta d'ordre 4 (bien qu'il y ait une infinité de choix possibles). L'obtention des conditions d'ordre des méthodes de Runge-Kutta devint très élégante avec la théorie des arbres et des B-séries, initiée par J. C. Butcher dans les années 1963-72 [But63, But64a, But64b, But69, But72].

Méthodes développables en B-séries Les B-séries ont été introduites par Hairer & Wanner [HW74]. La série de Taylor de la solution exacte (D.1) avec valeur initiale $y(0) = y$ s'écrit

$$y(h) = y + hf(y) + \frac{h^2}{2!} f'(y)f(y) + \frac{h^3}{3!}\Big(f''(f(y), f(y)) + f'(y)f'(y)f(y)\Big) + \dots$$

En effet, $\dot y = f(y)$, $\ddot y = f'(y)\dot y = f'(y)f(y)$, etc. Les méthodes de B-séries sont des intégrateurs numériques $y_{n+1} = \Phi_h(y_n)$ dont la série de Taylor possède la même structure, mais avec des coefficients réels $a(\tau)$:

$$\Phi_h(y) = y + ha(\bullet)f(y) + h^2 a(\begin{smallmatrix}\bullet\\|\\\bullet\end{smallmatrix})f'(y)f(y) + h^3\Big(\frac{a(\begin{smallmatrix}&\bullet\\\diagup&\diagdown\\\bullet&&\bullet\end{smallmatrix})}{2} f''(f(y), f(y)) + a(\begin{smallmatrix}\bullet\\|\\\bullet\\|\\\bullet\end{smallmatrix})f'(y)f'(y)f(y)\Big) + \dots$$

où les coefficients $a(\tau)$ sont définis pour tous les arbres racinés et caractérisent l'intégrateur. Les méthodes de B-séries incluent non seulement toutes les méthodes de Runge-Kutta mais aussi les méthodes de séries de Taylor, la méthode à un pas sous-jacente aux intégrateurs linéaires multi-pas, etc (voir [HLW06, Chap. XIV]).

Pour des classes particulières d'équations différentielles, il est essentiel, pour obtenir un bon comportement qualitatif des solutions, d'utiliser des intégrateurs numériques qui préservent certaines propriétés géométriques du flot exact.

Exemple : la preuve historique par Newton de la seconde loi de Kepler Le problème de Kepler décrit le mouvement de deux corps s'attirant mutuellement, par exemple une planète tournant autour du Soleil. Il s'écrit comme une équation différentielle

$$\dot q = p, \qquad \dot p = f(q) = -\frac{q}{\|q\|^3}, \tag{D.3}$$

où $q = (q_1, q_2)$ et $p = (p_1, p_2)$ représentent la position et le moment de la planète relativement au Soleil. Comme nous allons le voir, ce problème possède de nombreuses propriétés géométriques, en particulier, il est hamiltonien. La seconde loi de Kepler stipule que le moment angulaire, défini par

$$\det(q, p) = q_1 p_2 - q_2 p_1$$

est une intégrale première, c'est-à-dire une quantité conservée le long de toute solution du système d'équations différentielles (D.3). Bien sûr, ce résultat peut être vérifié par simple dérivation. En 1687, Newton donna dans le 'Theorema 1' de son *Principia* [New87] une élégante preuve géométrique de ce résultat. Étonnement, sa preuve repose sur un intégrateur géométrique : la méthode d'Euler symplectique, qui est étroitement liée à la méthode de Störmer–Verlet, un intégrateur aujourd'hui largement utilisé en dynamique moléculaire pour son excellent comportement. Dans le livre de Ostermann & Wanner [OW08], on peut trouver une description des grandes découvertes de Newton, en fait obtenues par des raisonnement très géométriques.

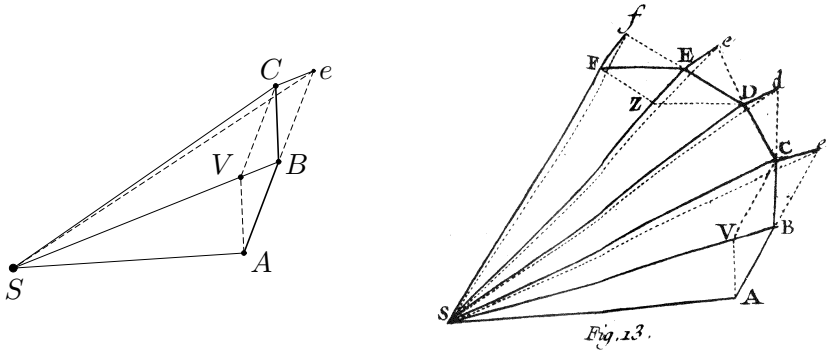


FIG. D.1 – Reproduction du *Principa* de Newton (à droite)

La preuve de Newton La preuve historique s'appuie sur la discrétisation suivante de l'équation différentielle (D.3)

$$q_{n+1} = q_n + h p_n, \qquad p_{n+1} = p_n + h f(q_{n+1}),$$

qui est connue aujourd'hui sous le nom de méthode d'Euler symplectique et peut s'interpréter de la manière suivante. Considérons le dessin 1 de Newton, où S représente le Soleil, et soient $A = q_{n-1}$, $B = q_n$, $C = q_{n+1}$, $D = q_{n+2}$, etc. Pendant le premier pas de temps, la planète se déplace de A à B en ligne droite, sans force externe, avec une vitesse constante p_{n-1}. Au point B, une force ponctuelle $f(q_n)$ est appliquée, où la vitesse est légèrement modifiée en direction du Soleil. Pendant le second pas de temps, le corps se déplace en C avec une vitesse constante p_n et ainsi de suite. Un calcul direct montre que le schéma ci-dessus implique la discrétisation naturelle suivante

$$q_{n+1} - 2q_n + q_{n+1} = h^2 f(q_n). \tag{D.4}$$

En fait, si l'on considère (D.4) avec l'approximation de vitesse plus précise $p_n = (q_{n+1} - q_{n-1})/(2h)$, on obtient la célèbre méthode de Störmer–Verlet, ou saute-mouton (leap frog en anglais), voir plus loin une formulation à un pas équivalente.

Revenons maintenant à la preuve géométrique de Newton. La diagonale (BV) du parallélogramme $ABCV$ est orientée vers le Soleil S car

$$\overrightarrow{BV} = \overrightarrow{BC} - \overrightarrow{AB} = (q_{n+1} - q_n) - (q_n - q_{n-1}) = h^2 f(q_n) = Const \cdot q_n.$$

On note qu'en l'absence de force, la planète aurait continué son déplacement en ligne droite de B vers e à vitesse constante, ainsi $CVBe$ est un parallélogramme. Ensuite, les triangles SAB et SBe ont une base de même longueur ($\overrightarrow{AB} = \overrightarrow{Be}$) et la même hauteur, et donc la même aire. De même, les triangles SBC et SBe avec la base commune SB et la même hauteur ont la même aire. Ainsi, les triangles SAB et SBC ont la même aire :

$$\det(q_{n-1}, q_n - q_{n-1}) = \det(q_n, q_{n+1} - q_n).$$

De la même manière, tous les triangles SAB, SBC, SCD, etc, ont la même aire. En remplaçant p_n en fonction de $q_n, q_{n+1}, \ldots$, on obtient que le moment angulaire $\det(q_n, p_n) = \det(q_{n-1}, p_{n-1})$ est conservé exactement par la discrétisation (D.4) (aussi bien pour Euler symplectique que pour Störmer–Verlet). On conclut que le mouvement d'un corps, soumis à une force centripède, satisfait la seconde loi de Kepler. □

D.1.1 Systèmes hamiltoniens et intégrateurs symplectiques

Une des classes de problèmes les plus importantes en intégration numérique géométrique est la classe des systèmes hamiltoniens, voir l'article introductif [Hai05]. Il s'agit d'équations différentielles de la forme

$$\dot{p} = -H_q(p, q), \qquad \dot{q} = H_p(p, q)$$

où $H(p, q)$ est une fonction scalaire représentant l'énergie totale du système, les vecteurs q et p de dimension d représentent la position et le moment, et d est le nombre de degrés de liberté. Ici, H_p et H_q sont les vecteurs des dérivées partielles. Le système peut se réécrire matriciellement sous la forme (D.1)

$$\dot{y} = J^{-1}\nabla H(y) \qquad \text{avec} \quad J = \begin{pmatrix} 0 & Id \\ -Id & 0 \end{pmatrix}, \tag{D.5}$$

où le vecteur $y = (p, q)^T$ est de dimension $2d$ dans l'espace des phases, et Id est la matrice identité de taille d. Par exemple, le problème de Kepler (D.3) est un système hamiltonien à $d = 2$ degrés de liberté, avec $H(p, q) = p^T p/2 + 1/\|q\|$.

Les systèmes hamiltoniens possèdent les deux propriétés fondamentales suivantes :

Conservation de l'énergie L'énergie $H(y) = H(p, q)$ est constante le long de toute solution de l'équation différentielle. On dit que c'est une intégrale première du système. On peut le montrer facilement par dérivation : $\frac{d}{dt}H(y(t)) = 0$.

Symplecticité La matrice jacobienne du flot φ_t des dérivées par rapport à y du système hamiltonien (D.5) vérifie l'identité matricielle (Poincaré [Poi92])

$$\varphi_t'(y)^T J \varphi_t'(y) = J.$$

En fait, cette propriété caractérise les systèmes hamiltoniens [HLW06, Theorem VI.2.8]. Elle implique la préservation du volume ($|\det \Phi_h'(y)| = 1$) et est équivalent en dimension $d = 1$, voir [HLW06, Sect. VI.2].

Ceci motive la définition suivante :

Définition D.1.2 *Un intégrateur numérique $y_{n+1} = \Phi_h(y_n)$ est symplectique pour un système hamiltonien (D.5) si la matrice jacobienne du flot numérique satisfait*

$$\Phi_h'(y)^T J \Phi_h'(y) = J$$

pour tout pas de temps h (suffisamment petit).

Malheureusement, un intégrateur numérique ne peut pas être simultanément symplectique et préserver l'énergie exactement, sinon il se réduit à une transformation du temps du flot exact. Ce résultat est dû à Ge & Marsden [GM88] et une preuve algébrique est donnée par Chartier, Faou & Murua [CFM06]. Cependant, un intégrateur symplectique conserve par définition $d(2d-1)$ invariants, et nous verrons par la suite que sous certaines hypothèses, il préserve bien l'énergie des systèmes hamiltoniens sur des temps exponentiellement longs.

On commence par quelques exemples de méthodes symplectiques.

La règle du point milieu Un des intégrateurs symplectiques les plus simples est la règle du point milieu

$$y_{n+1} = y_n + hf\big(\frac{y_n + y_{n+1}}{2}\big).$$

C'est une méthode de Runge-Kutta à deux étages et donc une méthode de B-séries.

Les deux intégrateurs suivants ne sont pas des méthodes de B-séries mais de P-séries, une extension naturelle aux systèmes partitionnés, faisant intervenir des arbres bicolores.

Méthode d'Euler symplectique En combinant les méthodes d'Euler explicite et implicite, on obtient deux méthodes adjointes (désignées sous le même nom),

$$\left\{ \begin{array}{rcl} p_{n+1} &=& p_n - hH_q(p_{n+1}, q_n) \\ q_{n+1} &=& q_n + hH_p(p_{n+1}, q_n) \end{array} \right. \qquad \text{et} \qquad \left\{ \begin{array}{rcl} p_{n+1} &=& p_n - hH_q(p_n, q_{n+1}) \\ q_{n+1} &=& q_n + hH_p(p_n, q_{n+1}) \end{array} \right. .$$

Méthode de Störmer–Verlet En composant un demi-pas de chaque méthode d'Euler symplectique, on obtient

$$\begin{array}{rcl} p_{n+1/2} &=& p_n - \dfrac{h}{2} H_q(p_{n+1/2}, q_n) \\ q_{n+1} &=& q_n + \dfrac{h}{2}\Big(H_p(p_{n+1/2}, q_n) + H_p(p_{n+1/2}, q_{n+1})\Big) \\ p_{n+1} &=& p_{n+1/2} - \dfrac{h}{2} H_q(p_{n+1/2}, q_{n+1}) \end{array}$$

Ces méthodes apparaissent déjà dans la preuve géométrique de la seconde loi de Kepler par Newton, présentée au début de cette introduction. Pour les hamiltoniens séparables, $H(q,p) = p^T p/2 + U(q)$ on peut montrer que cette méthode est la formulation à un pas de la discrétisation équivalente (D.4) où $f(q) = -\nabla U(q)$, avec l'approximation de vitesse $p_n = (q_{n+1} - q_{n-1})/(2h)$. On peut noter que les méthodes d'Euler symplectique et de Störmer–Verlet sont des schémas explicites lorsque l'hamiltonien est séparable.

Intégrateurs symétriques On peut montrer que la règle du point milieu et la méthode de Störmer–Verlet sont symétriques, c'est-à-dire

$$\Phi_h \circ \Phi_{-h}(y) = y \qquad \text{ou de manière équivalente} \qquad \Phi_{-h}^{-1}(y) = \Phi_h(y).$$

On peut le vérifier en observant que la substitution $y_n \leftrightarrow y_{n+1}$ et $h \leftrightarrow -h$ ne modifie pas les formules. Ces deux intégrateurs sont donc d'ordre 2, car une méthode symétrique a toujours un ordre de convergence pair [HLW06, Theorem II.3.2].

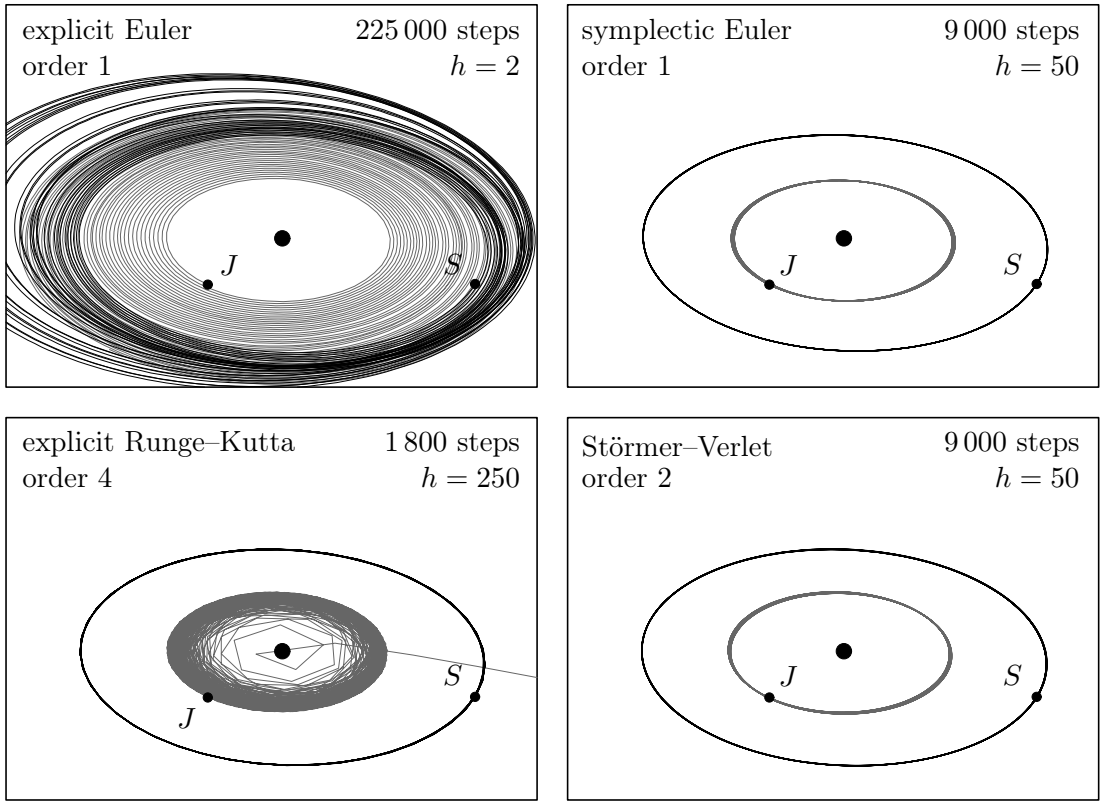


FIG. D.2 – Intégrateurs symplectiques et non symplectiques pour le système Soleil-Jupiter-Saturn (grands pas de temps).

Expérience numérique : le problème à trois corps On considère le problème à trois corps (Soleil-Jupiter-Saturne) qui est un système hamiltonien avec

$$H(p,q) = \frac{1}{2}\sum_{i=0}^{2}\frac{1}{m_i}p_i^T p_i - G\sum_{i=1}^{2}\sum_{j=0}^{i-1}\frac{m_i m_j}{\|q_i - q_j\|}.$$

On prend les conditions initiales $q_i(0), p_i(0)$ dans $\mathbb{R}^3$, la constante G et les masses m_i dans [HLW06, Table I.2.2]. À ce système, on applique la méthode d'Euler explicite avec le pas de temps $h = 2$ jours, la méthode d'Euler symplectique et la méthode de Störmer–Verlet avec le pas bien plus grand $h = 50$, toutes sur une période de 450 000 jours. On donne aussi le résultat pour la méthode de Runge-Kutta explicite (D.2) d'ordre de convergence 4, et donc avec une longueur de pas plus grande $h = 250$. Dans la figure D.2, on observe que les méthodes d'Euler symplectique et de Störmer–Verlet présentent toutes deux un comportement correct. Pour la méthode d'Euler explicite, on observe que les planètes sont éjectées en spirales avec une énergie croissante, alors que pour la méthode Runge-Kutta (D.2) Jupiter s'écrase dans le soleil avant d'être éjectée à son tour. Il est à noter que les méthodes d'Euler symplectique et de Störmer–Verlet présenteraient encore un comportement correct avec le plus grand pas de temps $h = 250$.

Dans l'expérience suivante (Figure D.3), on étudie la conservation de l'énergie. On observe que l'erreur dans l'énergie croît linéairement avec le temps pour les méthodes non symplectiques (Euler explicite et Runge-Kutta d'ordre 4). La justification de cette croissance linéaire en temps est immédiate, en utilisant le fait que le flot exact φ_h conserve

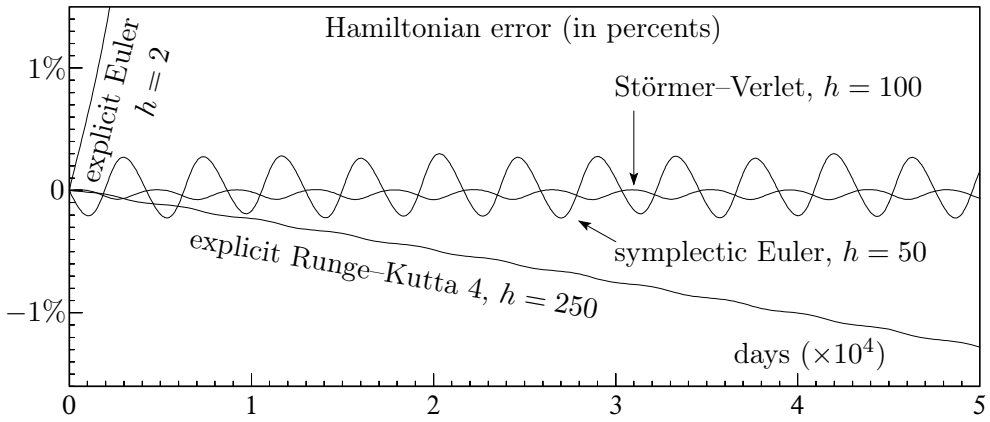


FIG. D.3 – Conservation de l'énergie pour le système à trois corps Soleil-Jupiter-Saturn

l'hamiltonien on a

$$H(y_{n+1}) - H(y_n) = H(y_{n+1}) - H(\varphi_h(y_n)) = \mathcal{O}(h^{p+1}),$$

car $y_{n+1} = \varphi_h(y_n) + \mathcal{O}(h^{p+1})$. En sommant cette estimation pour $n = 0$ à $N-1$, on obtient la borne linéaire

$$H(y_N) - H(y_0) = \mathcal{O}(th^p),$$

où $t = Nh$ et p est l'ordre de la méthode.

En revanche, l'erreur dans l'énergie reste bornée et petite pour les intégrateurs symplectiques (Euler syplectique et Störmer–Verlet),

$$H(y_N) - H(y_0) = \mathcal{O}(h^p).$$

L'explication théorique de ce comportement est due à Benettin & Giorgilli [BG94] et Tang [Tan94], voir [HLW06, Sect. IX.8]. Elle s'obtient en utilisant la théorie de l'analyse rétrograde (backward error analysis).

D.1.2 L'analyse rétrograde

Considérons un système d'équations différentielles ordinaires (D.1) $\dot{y} = f(y)$, et un intégrateur numérique

$$y_{n+1} = \Phi_h(y_n).$$

L'idée de l'analyse rétrograde est de chercher et d'étudier une équation différentielle modifiée

$$\dot{z} = \widetilde{f}_h(z) = f(z) + hf_2(z) + h^2 f_3(z) + \ldots, \qquad z(0) = y_0, \tag{D.6}$$

qui soit une série formelle en puissance de la longueur de pas h, telle que la solution numérique $\{y_n\}$ soit formellement égale à la solution exacte de (D.6),

$$y_n = z(nh) \qquad \text{pour} \quad n = 0, 1, 2, \ldots,$$

c'est-à-dire (le dessin du haut de la Figure D.5)

$$\Phi_{f,h}(y) = \varphi_{\widetilde{f}_h,h}(y), \tag{D.7}$$

où $\varphi_{\widetilde{f}_h,h}$ désigne le flot exact de (D.6).

L'idée de l'analyse rétrograde à été introduite par Wilkinson (1960) dans le contexte de l'algèbre linéaire numérique. Elle ne fut pas appliquée à l'intégration des équations différentielles ordinaires avant que l'on s'intéresse au comportement en temps longs des solutions numériques. Ruth [Rut83] utilisa l'idée de l'analyse rétrograde pour motiver l'utilisation d'intégrateurs symplectiques pour les systèmes hamiltoniens. En fait, si on applique une méthode symplectique à un système hamiltonien $\dot y = J^{-1}\nabla H(y)$, l'équation différentielle modifiée (D.6) est aussi hamiltonienne,

$$\dot z = \widetilde f_h(z) = J^{-1}\nabla \widetilde H(z), \qquad \widetilde H(z) = H(z) + hH_2(z) + h^2H_3(z) + \dots .$$

Ceci permet de transférer des propriétés des systèmes hamiltoniens perturbés (par exemple : conservation de l'énergie, théorie KAM pour les systèmes intégrables) vers les intégrateurs symplectiques. Rapidement, il est apparu que ce type de raisonnement ne se restreint pas aux systèmes hamiltoniens mais que ces techniques s'étendent aux équations différentielles réversibles, aux systèmes de Poisson, aux problèmes à divergence nulle, etc.

Une analyse rigoureuse à été développée dans les années quatre-vingt-dix[2]. On a le théorème fondamental suivant qui justifie rigoureusement l'utilisation des méthodes symplectiques et qui est dû à Benettin & Giorgilli [BG94] et Tang [Tan94], voir [HLW06, Sect. IX.8].

Théoreme D.1.3 *Considérons un système hamiltonien (D.5) avec $H : U \to \mathbb{R}$ analytique et une méthode de B-série (ou P-série) $y_{n+1} = \Phi_h(y_n)$ d'ordre p appliquée avec un pas constant[3] h. On suppose*

- *l'intégrateur est symplectique pour tout système hamiltonien $\dot y = J^{-1}\nabla H(y)$;*
- *et la solution numérique reste contenue dans un compact.*

Alors, pour $t_n = nh$ et $h \to 0$ on a

$$\begin{aligned} \widetilde H(y_n) &= \widetilde H(y_0) + \mathcal{O}(e^{-\gamma/(\omega h)}) \\ H(y_n) &= H(y_0) + \mathcal{O}(h^p) \end{aligned}$$

sur des intervalles exponentiellement longs $nh \le e^{\gamma/(\omega h)}$, où $\gamma > 0$ dépend uniquement de la méthode et $\omega > 0$ est relié à la constante de Lipschitz (plus haute fréquence) de l'équation différentielle.

Cela signifie que pour un pas h assez petit, l'énergie est bien conservée et bornée par $\mathcal{O}(h^p)$ sur des intervalles de temps exponentiellement longs. L'idée principale de la preuve est que la solution numérique $\{y_n\}$ étant (formellement) la solution exacte du système hamiltonien perturbé via l'analyse rétrograde (D.1.2), la solution numérique conserve (formellement) exactement l'hamiltonien modifié $H_h(z)$. Comme cet hamiltonien modifié est une petite perturbation de taille $\mathcal{O}(h^p)$ de l'hamiltonien original $H(y)$, l'hamiltonien original est bien conservé.

La série formelle dans l'équation différentielle modifiée (D.6) ne converge pas en général (sauf pour des problèmes linéaires), ceci rend l'analyse rigoureuse très technique ; on est obligé de tronquer les séries de manière à ce que l'erreur soit la plus petite possible. On peut montrer que, si l'on tronque la série (D.6) après le terme de taille $\mathcal{O}(h^{N(h)})$ où $N(h) = \mathcal{O}(1/h)$, on obtient l'erreur de troncature exponentiellement petite qui apparaît dans le Théorème D.1.3.

[2] nonante pour les Suisses

[3] Le comportement excellent des intégrateurs symplectiques est en général perdu avec des pas variables, voir [HLW06, Sect. VIII.2]. Ici, on considère un pas constant h.

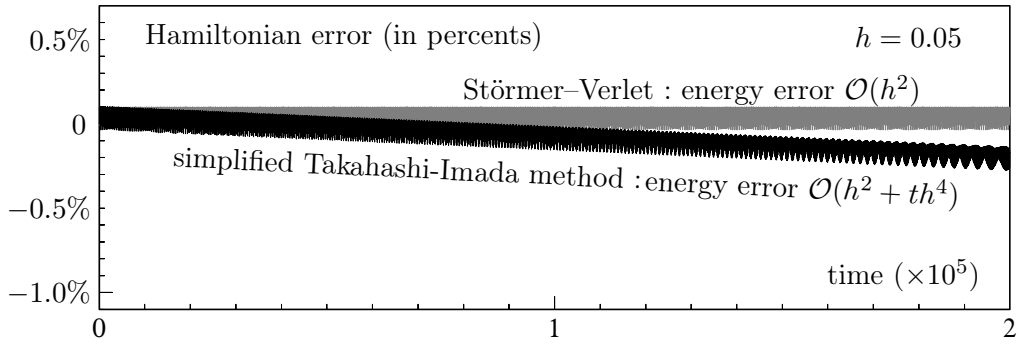


FIG. D.4 – Erreur hamiltonienne le long de la solution numérique du pendule asymétrique. Ce contre-exemple montre que la symplecticité seule ne suffit pas pour une bonne conservation à long terme. Il est tiré de [HMS08].

Néanmoins, les résultats de conservation de l'énergie obtenus avec l'analyse rétrograde décrits précédemment, ne s'appliquent PAS aux équations différentielles hautement oscillantes ou aux problèmes en dimension infinie (équations aux dérivées partielles), car la conclusion du Théorème D.1.3 devient caduque pour $\omega \to \infty$.

Remarque D.1.4 *Il n'y a pas seulement les méthodes symplectiques qui ont un bon comportement en temps long. Par exemple, la règle du trapèze*

$$y_{n+1} = y_n + \frac{h}{2}\big(f(y_n) + f(y_{n+1})\big)$$

n'est pas symplectique, mais elle est conjuguée à la règle du point milieu qui, elle, est symplectique. En effet, il existe une application χ_h, qui est une perturbation $\mathcal{O}(h^2)$ de l'identité, telle que

$$\Phi_h^{trap} = (\chi_h)^{-1} \circ \Phi_h^{point\ milieu} \circ \chi_h.$$

Ainsi, après n pas de la méthode, on a $(\Phi_h^{trap})^n = (\chi_h)^{-1} \circ (\Phi_h^{midpoint})^n \circ \chi_h$ et la règle du trapèze a le même comportement en temps long que la règle du point milieu, symplectique. On appelle cela la symplecticité conjuguée ([Sto88], voir [HLW06, Sect. VI.8]).

Remark D.1.5 *Il existe des résultats de conservation similaires pour les méthodes de B-séries (ou P-séries) symétriques appliquées à des systèmes intégrables réversibles (par exemple le problème de Kepler) ou des systèmes intégrables réversibles perturbés (comme le problème à trois corps Soleil-Jupiter-Saturn), voir [HLW06, Chap. XI]. D'un point de vue pratique, la propriété de symétrie est en en général plus facile à obtenir que la symplecticité d'un intégrateur numérique.*

Pendule asymétrique Pour illustrer les difficultés que peut rencontrer une méthode symplectique, on termine cette section par l'exemple du pendule asymétrique proposé dans [FHP04], qui est un système hamiltonien à un seul degré de liberté, avec

$$H(p, q) = p^2/2 - \cos q + 0.2 \sin(2q).$$

On considère la condition initiale $q(0) = 0$, $p(0) = 2.5$. La vitesse initiale est suffisamment grande pour que le pendule fasse des tours, et la vitesse reste positive au cours du temps. Ainsi, la symétrie $p \leftrightarrow -p$ n'a pas d'influence sur la solution numérique, et la perturbation

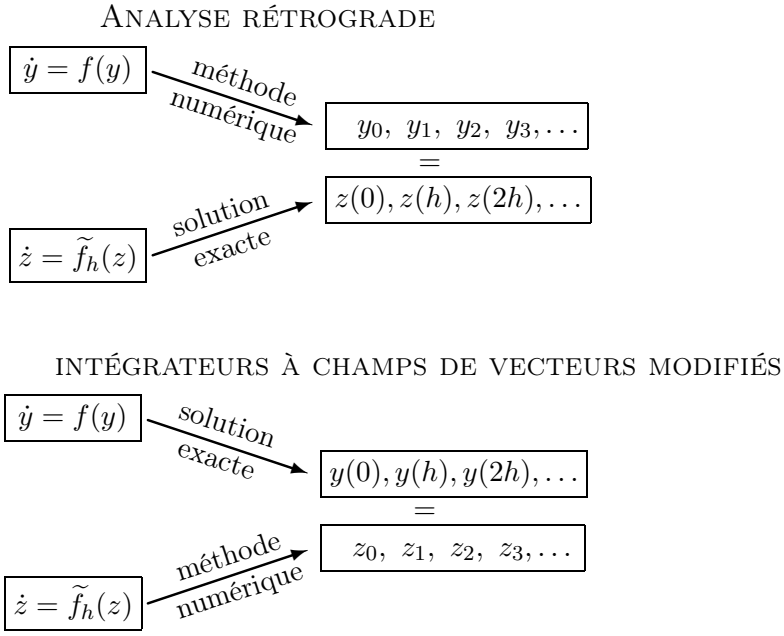


Fig. D.5 – L'analyse rétrograde comparée aux intégrateurs à champs de vecteurs modifiés

$+0.2\sin(2q)$ dans l'hamiltonien détruit la symétrie $q \leftrightarrow -q$. La Remarque D.1.5 pour les méthodes symétriques ne s'applique donc pas pour ce problème. Pour la méthode de Störmer–Verlet (voir Figure D.4 avec le pas $h = 0.05$), l'énergie est bien conservée au cours du temps, et c'est une conséquence directe du Théorème D.1.3. On pourrait opposer que la solution ne reste pas contenue dans un compact, car $q(t)$ croit indéfiniment pendant que le pendule fait des tours. Néanmoins, l'angle du pendule est en fait défini modulo 2π, ainsi, l'espace des phases naturel pour (p, q) est le cylindre $\mathbb{R} \times [0, 2\pi]$, et la solution est en fait périodique.

A l'inverse, pour la méthode de Takahashi-Imada simplifiée, l'énergie n'est pas bien conservée, et on observe un biais linéaire dans la Figure D.4 (même pas de temps $h = 0.05$). Cette méthode possède le même ordre de convergence 2 que Störmer–Verlet. Il s'agit d'une modification où le champs $f(q)$ est remplacé par $f(q + h^2/12\, f(q))$ dans la définition. La motivation pour cela est que la nouvelle méthode est d'ordre effectif 4, c'est-à-dire qu'il existe un changement de coordonnées χ_h tel que $\chi_h^{-1} \circ \Phi_h \circ \chi_h$ est d'ordre 4. Le concept d'ordre effectif a été introduit initialement par Butcher [But69] dans le contexte des méthodes de Runge-Kutta. La méthode de Takahashi-Imada simplifiée est non symplectique en général (et ne vérifie donc pas les hypothèses du Théorème D.1.3), mais c'est encore une méthode de B-série symétrique qui préserve le volume. Le flot numérique est alors symplectique pour le problème du pendule, car c'est un problème à un seul degré de liberté. Ce contre-exemple est tiré de [HMS08], et l'explication de la non conservation de l'énergie est que l'hamiltonien modifié n'est pas globalement défini sur le cylindre : l'intégrale sur une période le long de la solution exacte de la fonction coefficient $H_4(q, p)$ dans l'hamiltonien modifié est non nulle. Ce contre-exemple simple illustre que la symplecticité seule n'est pas suffisante pour assurer une bonne conservation à long terme de l'énergie.

D.2 Principaux résultats obtenus

On décrit ici, chapitre par chapitre, les principales idées des nouveaux résultats présentés dans cette thèse.

Chapitre 1 : Intégrateurs à champ de vecteurs modifié

L'analyse rétrograde est un outil théorique qui donne de nombreuses informations sur le comportement à long terme des intégrateurs numériques géométriques. Nous allons montrer que simplement en échangeant les rôles de la "solution numérique" et de la "solution exacte" (cf. les deux schémas de la Figure D.5), on obtient un nouveau moyen de construire des intégrateurs d'ordre élevé, qui préservent les propriétés géométriques et seront utiles pour l'intégration à long terme.

Précisément, on considère comme précédemment un problème d'équation différentielle à valeur initiale (D.1) et un intégrateur numérique. Mais à présent, on cherche une équation différentielle modifiée $\dot{z} = \widetilde{f}_h(z)$, toujours de la forme (D.6), de sorte que la solution numérique $\{z_n\}$ de la méthode appliquée avec le pas h à l'équation modifiée $\dot{z} = \widetilde{f}_h(z)$ soit formellement égale à la solution exacte

$$\Phi_{\widetilde{f}_h,h}(y) = \varphi_{f,h}(y) \tag{D.8}$$

du problème initial (D.1), c'est-à-dire

$$z_n = y(nh) \qquad \text{pour} \quad n = 0, 1, 2, \ldots,$$

(voir le schéma du bas dans la Figure D.5). Remarquons que cette équation modifiée est différente de celle considéré précédemment pour l'analyse rétrograde. Cependant, en raison de sa grande similarité avec l'analyse rétrograde, tous les résultats théoriques et pratiques s'étendent dans ce nouveau contexte. L'équation différentielle modifiée est encore une série asymptotique en générale divergente, et ses troncatures héritent des propriétés géométriques du flot exact lorsqu'un intégrateur adapté est appliqué. Les fonctions coefficients $f_j(z)$ peuvent être calculées récursivement à l'aide d'un logiciel de calcul formel comme MAPLE. Pour cela, on développe chacun des membres de l'équation $z(t+h) = \Phi_{\widetilde{f}_h,h}(z(t))$ en séries en puissances de h, et on compare les coefficients. Une fois que quelques fonctions $f_j(z)$ sont connues, on obtient l'algorithme suivant.

Algorithme D.2.1 (intégrateurs à champ de vecteurs modifié)
Considérons la troncature

$$\dot{z} = f_h^{[r]}(z) = f(z) + hf_2(z) + \cdots + h^{r-1}f_r(z) \tag{D.9}$$

de l'équation différentielle modifiée correspondant à l'intégrateur $\Phi_{f,h}(y)$. Alors,

$$z_{n+1} = \Psi_{f,h}(z_n) := \Phi_{f_h^{[r]},h}(z_n)$$

définit une méthode numérique d'ordre r qui approche la solution de (D.1). On l'appelle intégrateur à champ de vecteurs modifié, car le champ de vecteurs $f(y)$ de (D.1) est modifié en $f_h^{[r]}$ avant que l'intégrateur de départ soit appliqué.

Cette approche est une alternative pour construire des intégrateurs numériques d'ordre élevé. Les approches classiques sont les méthodes multi-pas, les méthode de Runge-Kutta, de séries de Taylor, l'extrapolation, la composition, et les méthodes de splitting. Elle est

d'un grand intérêt dans le contexte de l'intégration numérique géométrique car, comme pour l'analyse rétrograde, l'équation différentielle modifiée hérite des propriétés structurelles de (D.1) si un intégrateur adapté est appliqué.

Peu de méthodes déjà connues peuvent s'inscrire dans le cadre des intégrateurs à champ de vecteurs modifié, bien que non construits de cette manière. Les plus importantes sont les méthodes de fonctions génératrices introduites par Feng [Fen86]. Ce sont des intégrateurs symplectiques d'ordre élevé pour les systèmes hamiltoniens, obtenus en appliquant une méthode symplectique de base à un système hamiltonien modifié. L'hamiltonien correspondant est solution formelle d'une équation aux dérivées partielles de Hamilton-Jacobi. L'approche générale de l'algorithm D.2.1 introduite est décrite dans [CHV07b].

Chapitres 1–2 : Analyse pour les B-séries : une loi de susbstitution

La plupart des méthodes numériques a un flot développable en B-séries (notamment les méthodes de Runge–Kutta), comme introduit et étudié dans [HW74], voir [HLW06, Chap. III].

Soit $T = \{\bullet, \begin{smallmatrix}\bullet\\|\\\bullet\end{smallmatrix}, \begin{smallmatrix}\bullet\\|\\\bullet\\|\\\bullet\end{smallmatrix}, \begin{smallmatrix}\bullet\\|\\\bullet\\/\!\backslash\\\bullet\;\bullet\end{smallmatrix}, \ldots\}$ l'ensemble des arbres racinés, et soit $\emptyset$ l'arbre vide. Pour $\tau_1, \ldots, \tau_m \in T$, on note $\tau = [\tau_1, \ldots, \tau_m]$ l'arbre obtenu en rattachant les racines de τ_1, $\ldots$, τ_m à un nouveau noeud, qui devient la racine de l'arbre τ. Les différentielles élémentaires $F_f(\tau)$ sont définies par récurrence par

$$F_f(\bullet)(y) = f(y), \qquad F_f(\tau)(y) = f^{(m)}(y)\big(F_f(\tau_1)(y), \ldots, F_f(\tau_m)(y)\big). \tag{D.10}$$

Étant donnés des coefficients réels $a(\emptyset)$ et $a(\tau), \tau \in T$, une B-série est une série de la forme

$$\begin{aligned} B(f,a) &= a(\emptyset)\, Id + \sum_{\tau \in T} \frac{h^{|\tau|}}{\sigma(\tau)} a(\tau) F_f(\tau) \\ &= a(\emptyset)\, Id + h a(\bullet)\, f + h^2 a(\begin{smallmatrix}\bullet\\|\\\bullet\end{smallmatrix})\, f'f + h^3 + h^3 a(\begin{smallmatrix}\bullet\\|\\\bullet\\/\!\backslash\\\bullet\;\bullet\end{smallmatrix})\, f''(f,f) + \ldots, \end{aligned}$$

où Id représente l'application identité $Id\,(y) = y$ et les scalaires $\sigma(\tau)$ sont des coefficients de normalisation connus. La série de Taylor de la solution exacte de (1.1) peut s'écrire comme une B-série $y(h) = B(f,e)(y_0)$ avec des coefficients $e(\tau)$. Le flot $y_{n+1} = \Phi_{f,h}(y_n)$ d'une méthode de Runge-Kutta est de la forme $\Phi_{f,h} = B(f,a)$ où $a(\tau)$ dépend exclusivement des coefficients de la méthode (voir [HLW06, Chap. III] pour plus de détails).

Dans la perspective d'unifier la théorie des intégrateurs à champs de vecteurs modifiés avec l'analyse rétrograde, on définit (D.6) comme l'équation différentielle modifiée donnée par

$$\Phi_{\widetilde{f}_h,h}(y) = \Psi_{f,h}(y) \tag{D.11}$$

où Φ et Ψ sont deux intégrateurs numériques développables en B-séries $\Phi_{f,h} = B(f,a)$ et $\Psi_{f,h} = B(f,c)$. Pour $\Psi_{f,h}(y) = \varphi_{f,h}(y)$ (flot exact), on retrouve la formule (D.8) pour les intégrateurs à champs de vecteurs modifiés, tandis que pour $\Phi_{\widetilde{f}_h,h}(y) = \varphi_{\widetilde{f}_h,h}(y)$ on obtient l'équation (D.7) pour l'analyse rétrograde.

En termes de B-séries, l'équation (D.11) s'écrit $B(\widetilde{f}_h, a) = B(f,c)$. En calculant récursivement les premiers coefficients de (1.2), on se convainc rapidement que ce sont des combinaisons linéaires de différentielles élémentaires et que $\widetilde{f}_h(y) = h^{-1}B(f,b)(y)$ avec des coefficients $b(\tau)$ à déterminer (on remarque que l'on a nécessairement $b(\emptyset) = 0$). Ceci motive le théorème suivant, introduit dans [CHV05].

Théoreme D.2.2 *Pour $b(\emptyset) = 0$, le champ de vecteurs $h^{-1}B(f,b)$ inséré dans $B(\cdot,a)$ donne une B-séries*

$$B\big(h^{-1}B(f,b),a\big) = B(f,b\star a).$$

On a $(b\star a)(\emptyset) = a(\emptyset)$, et les valeurs des premiers coefficients sont données dans la Table 1.2 plus loin. Une formule générale pour $(b\star a)(\tau)$ est donnée dans (1.27) en Sect. 1.4.

Éléments de la preuve. On calcule à présent à la main les coefficients de la loi de substitution pour les premiers arbres, jusqu'à l'ordre 3. On considère une B-série

$$\begin{aligned} B(g,a)(y) &= a(\emptyset)y + ha(\bullet)g(y) + h^2a(\bullet\!\!-\!\!\bullet)g'(y)g(y) + \frac{h^3}{2}a(\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet)g''(y)(g(y),g(y)) \\ &\quad + h^3a(\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet)g'(y)g'(y)g(y) + \dots \end{aligned} \tag{D.12}$$

où le champ de vecteurs g est lui-même remplacé par une B-série $g = h^{-1}B(f,b)$. En développant chaque terme de la série individuellement, et en omettant l'argument (y), on obtient

$$hg = hb(\bullet)f + h^2b(\bullet\!\!-\!\!\bullet)f'f + \frac{h^3}{2}b(\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet)f''(f,f) + h^3b(\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet)f'f'f + \dots$$

$$\begin{aligned} h^2g'g &= h^2(b(\bullet)f + hb(\bullet\!\!-\!\!\bullet)f'f + \dots)'(b(\bullet)f + hb(\bullet\!\!-\!\!\bullet)f'f + \dots) \\ &= h^2b(\bullet)^2f'f + 2h^3b(\bullet)b(\bullet\!\!-\!\!\bullet)f'f'f + h^3b(\bullet\!\!-\!\!\bullet)b(\bullet)f''(f,f) + \dots \end{aligned}$$

$$\begin{aligned} \frac{h^3}{2}g''(g,g) &= \frac{h^3}{2}(b(\bullet)f + \dots)''(b(\bullet)f + \dots, b(\bullet)f + \dots) \\ &= \frac{h^3}{2}b(\bullet)^3f''(f,f) + \dots \end{aligned}$$

$$\begin{aligned} h^3g'g'g &= h^3(b(\bullet)f + \dots)'(b(\bullet)f + \dots)'(b(\bullet)f + \dots) \\ &= h^3b(\bullet)^3f'f'f + \dots \end{aligned}$$

On substitue ensuite les expressions de hg, $h^2g'g$, $h^3g'g'g$, $\frac{h^3}{2}g''(g,g)$ dans (D.12), et on regroupe les termes en hf, $h^2f'f$, $h^3f'f'f$, $\frac{h^3}{2}f''(f,f)$. Cela donne

$$\begin{aligned} B(g,a)(y) &= a(\emptyset) + ha(\bullet)b(\bullet)f + h^2\Big(a(\bullet)b(\bullet\!\!-\!\!\bullet) + a(\bullet\!\!-\!\!\bullet)b(\bullet)^2\Big)f'f \\ &\quad + \frac{h^3}{2}\Big(a(\bullet)b(\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet) + 2a(\bullet\!\!-\!\!\bullet)b(\bullet)b(\bullet\!\!-\!\!\bullet) + a(\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet)b(\bullet)^3\Big)f''(f,f) \\ &\quad + h^3\Big(a(\bullet)b(\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet) + 2a(\bullet\!\!-\!\!\bullet)b(\bullet)b(\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet) + a(\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet)b(\bullet)^3\Big)f'f'f + \dots \\ &= B(f,b\star a)(y) \end{aligned}$$

On obtient ainsi les premiers coefficients de la loi de substitution :

$$\begin{aligned} (b\star a)(\emptyset) &= a(\emptyset) \\ (b\star a)(\bullet) &= a(\bullet)b(\bullet) \\ (b\star a)(\bullet\!\!-\!\!\bullet) &= a(\bullet)b(\bullet\!\!-\!\!\bullet) + a(\bullet\!\!-\!\!\bullet)b(\bullet)^2 \\ (b\star a)(\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet) &= a(\bullet)b(\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet) + 2a(\bullet\!\!-\!\!\bullet)b(\bullet)b(\bullet\!\!-\!\!\bullet) + a(\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet)b(\bullet)^3 \\ (b\star a)(\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet) &= a(\bullet)b(\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet) + 2a(\bullet\!\!-\!\!\bullet)b(\bullet)b(\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet) + a(\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet\!\!-\!\!\bullet)b(\bullet)^3 \end{aligned} \tag{D.13}$$

La détermination de l'équation modifiée définie par (D.11), c'est-à-dire des coefficients $b(\tau)$ pour $a(\tau)$ et $c(\tau)$ donnés, dans la relation

$$B\big(h^{-1}B(f,b),a\big) = B(f,c),$$

revient à résoudre en $b(\tau)$ le système algébrique d'équations

$$(b \star a)(\tau) = c(\tau) \qquad \text{pour} \quad \tau \in T. \tag{D.14}$$

On remarque que

$$(b \star a)(\tau) = a(\bullet)b(\tau) + \cdots + a(\tau)b(\bullet)^{|\tau|},$$

où les trois petits points ne comportent que des arbres d'ordres strictement plus petits que $|\tau|$. Par conséquent, pour des intégrateurs consistants $\Phi_{f,h} = B(f,a)$ et $\Psi_{f,h} = B(f,c)$, pour lesquels $a(\emptyset) = a(\bullet) = 1$ et $c(\emptyset) = c(\bullet) = 1$, les coefficients $b(\tau)$ peuvent être calculés récursivement à partir de (D.14). De cette manière, le calcul des champs de vecteurs $f_j(y)$ dans l'équation différentielle modifiée (D.6) ou (D.9) se réduit au calcul de coefficients réels.

Intégrateurs à champs de vecteurs modifiés. Dans ce cas, $\Psi_{f,h}$ dans (D.11) est le h-flot exact qui est une B-série avec des coefficients $e(\tau)$. Par conséquent, les coefficients $b(\tau)$ de l'équation différentielle modifiée pour l'intégrateur $\Phi_{f,h} = B(f,a)$ s'obtiennent à partir de

$$(b \star a)(\tau) = e(\tau) \qquad \text{pour} \quad \tau \in T.$$

Analyse rétrograde. L'équation différentielle modifiée pour une méthode $\Psi_{f,h} = B(f,c)$ est obtenue en prenant $\Phi_{f,h}$ égal au flot exact. Ses coefficients $b(\tau)$ sont alors donnés par la relation

$$(b \star e)(\tau) = c(\tau) \qquad \text{pour} \quad \tau \in T.$$

Une loi de groupe sur les B-séries Les B-séries $h^{-1}B(f,b)$ correspondant aux applications $b : T \cup \{\emptyset\} \to \mathbb{R}$ avec $b(\emptyset) = 0$ représentent des champs de vecteurs. Le produit $b \star a$ définit une structure de groupe sur l'ensemble $\big\{c : T \cup \{\emptyset\} \to \mathbb{R}\,;\, c(\emptyset) = 0, c(\bullet) = 1\big\}$ représentant de tels champs de vecteurs. Son élément unité est donné par $c(\bullet) = 1$ et $c(\tau) = 0$ pour $|\tau| > 1$, il correspond au champ de vecteurs d'origine construit à partir de $f(y)$.

Dans le chapitre 2, on étudie plus avant les propriétés algébriques de la loi de substitution sur les B-séries. On présente la structure algébrique commune de deux lois de composition sur les B-séries : la composition du groupe de Butcher, qui correspond à la composition du flot des intégrateurs, et la loi de substitution introduite au chapitre précédent qui correspond à la composition des champs de vecteurs de B-séries.

Les structures d'algèbres de Hopf sur les arbres racinés sont maintenant bien étudiées, particulièrement en combinatoire, et sont essentiellement caractérisées par une loi de coproduit. Il est bien connu que la première loi de composition correspond au produit de convolution sur l'algèbre de Hopf d'arbres de Connes & Kreimer pour la renormalisation dans la théorie des champs quantiques. Il a été démontré récemment par Calaque, Ebrahimi-Fard & Manchon [CEFM08], dans le contexte de l'algèbre combinatoire, que la loi de substitution sur les B-séries permet de définir un nouveau coproduit Δ_{CEM} qui permet la construction d'une nouvelle algèbre de Hopf sur les arbres, par exemple (comparer avec (D.13))

$\Delta_{CEM}($  $) =$  $\otimes \bullet + 2 \bullet$ $\otimes$ $+ \bullet^3 \otimes$

On montre que la nouvelle loi de substitution ainsi construite est compatible avec la composition standard des B-séries,

$$B(f,a)\Big(B(f,b)(y)\Big) = B(f, b\cdot a)(y).$$

Par exemple, on a la distributivité

$$b \star (a\cdot c) = (b\star a)\cdot(b\star c).$$

On montre également que le sous-groupe des B-séries symplectiques (pour la composition standard des B-séries) est en bijection naturelle via l'analyse rétrograde avec le sous-groupe des B-séries hamiltoniennes muni de la loi de substitution.

Enfin, on explique l'extension de cette théorie aux méthodes d'intégrations partitionnées (P-séries). C'est particulièrement important dans le contexte des intégrateurs symplectiques.

Chapitre 3 : Un intégrateur d'ordre élevé pour le mouvement d'un corps rigide

Pour illustrer l'efficacité que peuvent avoir les intégrateurs à champs de vecteurs modifiés, on considère les équations du mouvement d'un corps rigide,

$$\dot y = \widehat y\, I^{-1} y, \qquad \dot Q = Q\,\widehat{I^{-1}y}, \qquad \text{où} \qquad \widehat a = \begin{pmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{pmatrix} \tag{D.15}$$

pour un vecteur $a = (a_1, a_2, a_3)^T$. Ici, $I = \operatorname{diag}(I_1, I_2, I_3)$ est la matrice des moments d'inertie, y est le vecteur du moment angulaire, et Q est la matrice orthogonale qui décrit la rotation relativement à un système de coordonnées fixé. Comme intégrateur numérique, on choisit l'algorithme Discrete Moser–Veselov (DMV) [MV91],

$$\widehat y_{n+1} = \Omega_n\, \widehat y_n\, \Omega_n^T, \qquad Q_{n+1} = Q_n\, \Omega_n^T, \tag{D.16}$$

où la matrice orthogonale Ω_n est donnée par

$$\Omega_n^T D - D\,\Omega_n = h\,\widehat y_n.$$

Ici, la matrice diagonale $D = \operatorname{diag}(d_1, d_2, d_3)$ est déterminée par $d_1 + d_2 = I_3$, $d_2 + d_3 = I_1$, et $d_3 + d_1 = I_2$. Cet algorithme est un excellent intégrateur géométrique et partage de nombreuses propriétés géométriques avec le flot exact. Il est symplectique, il préserve exactement l'hamiltonien, le casimir et le moment angulaire Qy (dans le référentiel fixé), et il préserve l'orthogonalité de Q. Son seul inconvénient est son faible ordre de convergence deux.

La technique des intégrateurs à champs de vecteurs modifiés ne peut pas être appliquée directement pour augmenter l'ordre de la méthode car l'algorithme (D.16) n'est pas défini pour des problèmes généraux (D.1). Il est cependant défini pour I_j arbitraires et par conséquent on cherche des moments d'inertie modifiés $\widetilde I_j$ tels que l'algorithme DMV appliqué avec $\widetilde I_j$ donne la solution exacte de (D.15). Il est démontré dans [HV06] que cela est possible avec

$$\frac{1}{\widetilde I_j} \;=\; \frac{1}{I_j}\Big(1 + h^2 s_3(y_n) + h^4 s_5(y_n) + \cdots\Big) + h^2 d_3(y_n) + h^4 d_5(y_n) + \cdots\ . \tag{D.17}$$

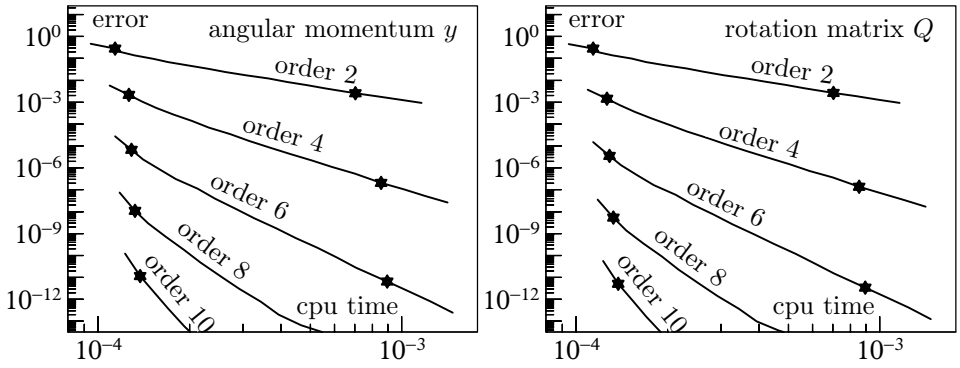


FIG. D.6 – Diagramme travail-précision pour l'intégrateur DMV (ordre 2) et les versions modifiées d'ordre 4, 6, 8 et 10.

Les expressions $s_k(y)$ et $d_k(y)$ peuvent être calculées avec un logiciel de calcul formel de la même manière que l'on obtient les équations différentielles modifiées. Les premiers termes sont

$$\begin{aligned} s_3(y_n) &= -\frac{1}{3}\Big(\frac{1}{I_1}+\frac{1}{I_2}+\frac{1}{I_3}\Big)\,H(y_n) + \frac{I_1+I_2+I_3}{6\,I_1\,I_2\,I_3}\,C(y_n), \\ d_3(y_n) &= \frac{I_1+I_2+I_3}{6\,I_1\,I_2\,I_3}\,H(y_n) - \frac{1}{3\,I_1\,I_2\,I_3}\,C(y_n), \end{aligned}$$

où

$$C(y) = \frac{1}{2}\Big(y_1^2+y_2^2+y_3^2\Big) \qquad \text{et} \qquad H(y) = \frac{1}{2}\Big(\frac{y_1^2}{I_1}+\frac{y_2^2}{I_2}+\frac{y_3^2}{I_3}\Big)$$

sont le casimir et l'Hamiltonien du système. L'interprétation physique de ce résultat est la suivante : après une perturbation adaptée de la forme du corps rigide, l'application de l'algorithme DMV donne le mouvement exact du corps rigide de départ. En tronquant les séries dans (D.17) après le terme h^{2r-2} on obtient l'algorithme DMV à champs modifiés d'ordre $2r$.

Expérience numérique On considère un corps rigide asymétrique avec les moments d'inertie $I_1 = 0.6$, $I_2 = 0.8$, et $I_3 = 1.0$ sur l'intervalle $[0, 10]$. Les valeurs initiales sont $y(0) = (1.8, 0.4, -0.9)^T$ et $Q(0)$ est la matrice identité. L'implémentation de l'algorithme DMV à champs modifiés est faite à l'aide de quaternions, comme expliqué dans [HV06]. Bien que $H(y)$ et $C(y)$ soient constants le long de la solution numérique, on recalcule les valeurs de $\widetilde{I}_j$ à chaque pas pour simuler la présence d'un potentiel externe.

On applique l'algorithme DMV et ses extensions d'ordre 4, 6, 8, et 10 avec de nombreux pas différents, et on trace dans la Figure D.6 l'erreur globale au temps final comme une fonction du temps de calcul CPU. Les temps d'exécution sont une moyenne sur 1000 expériences. Les symboles donnent les valeur obtenues avec les longueurs de pas $h = 0.1$ et $h = 0.01$, respectivement.

Les figures illustrent bien les ordres de convergence espérés (l'ordre p correspond à une droite de pente $-p$). Le plus intéressant est le fait qu'une grande précision est obtenue plus ou moins gratuitement. Considérons les résultats obtenus avec les pas $h = 0.1$. L'erreur pour l'algorithme DMV (ordre 2) est supérieure à 20%. Pour un très faible surcoût, la modification d'ordre 10 fournit une précision de plus de 11 chiffres avec le même pas de temps.

Points conjugués sur les géodésiques de corps rigides Dans [BF07], le lieu conjugué (l'ensemble des points conjugués) des géodésiques de corps rigides est étudié dans le cas où deux moments d'inertie sont égaux (par exemple $I_2 = I_3$), et le cas général asymétrique est actuellement étudié dans [BF07].

Dans la perspective de calculer des points conjugués sur des géodésiques de corps rigides, on donne un algorithme précis pour le calcul de dérivées du flot par rapport aux conditions initiales. Il s'agit du champ tangent

$$\frac{\partial y(t)}{\partial y_0}, \qquad \frac{\partial Q(t)}{\partial y_0}.$$

On montre que les dérivées de $Q(t)$ peuvent être aisément approchées sous la forme

$$\frac{\partial Q_n}{\partial y_{0,j}} = Q_n \widehat{a}_{n,j}, \quad j = 1, 2, 3$$

où les $\widehat{a}_{n,j}$ sont des matrices antisymétriques. Ensuite, les points conjugués sont simplement obtenus lorsque la matrice 3×3 dont les colonnes sont les vecteurs $a_{n,j}$ devient singulière.

L'idée de l'algorithme est de dériver par rapport aux conditions initiales la discrétisation d'ordre élevé des équations du mouvement données par l'algorithme DMV modifié. On montre que cela peut être implémenté efficacement.

Chapitre 4 : Le rôle des intégrateurs symplectiques en contrôle optimal

On considère un problème de contrôle optimal de la forme

$$(P) \begin{cases} \text{Min } \Phi(x(1)), \\ \dot{x}(t) = f(x(t), u(t)), \quad t \in (0,1), \\ x(0) = x^0, \end{cases}$$

où $f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ et $\Phi : \mathbb{R}^n \to \mathbb{R}$ sont deux fonctions lisses (C^∞), et on suppose, pour simplifier, que l'état $x(t)$ et la fonction de contrôle $u(t)$ sont continues. Les conditions nécessaires d'optimalité données par le principe du maximum de Pontryagin, un outil majeur en contrôle optimal (voir par exemple [Eva83, MS82]), sont les suivantes. Il existe une fonction co-état $p : (0,1) \to \mathbb{R}^n$ telle que la solution (P) est solution du problème aux deux bouts,

$$(OC) \begin{cases} \dot{x}(t) = H_p(x(t), p(t), u(t)) \\ \dot{p}(t) = -H_x(x(t), p(t), u(t)) \\ H(x(t), p(t), u(t)) = \min_{\alpha \in A} H(x(t), p(t), \alpha) \\ x(0) = x^0, \quad p(1) = \Phi'(x(1)). \end{cases}$$

pour $t \in (0,1)$, où la fonction hamiltonienne $H : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ est définie par

$$H(x, p, u) = p^T f(x, u).$$

De plus, l'hamiltonien est conservé, c'est-à-dire $H(x(t), p(t), u(t))$ est constant le long de la solution de (OC).

Le travail de [Hag00, BLV06] montre qu'appliquer une discrétisation de Runge-Kutta directement au problème de contrôle optimal (P) est équivalent à appliquer une méthode de Runge-Kutta partitionnée symplectique au système hamiltonien issu de la formulation de Pontryagin du problème de contrôle optimal.

Par exemple, on considère la *méthode d'Euler explicite* avec $h = \frac{1}{N}$, et $x(t_k) \approx x_k,\ t_k = kh$:

$$\begin{cases} \text{Min } \Phi(x_N), \\ x_{k+1} = x_k + hf(x_k, \overline{u}_k), \quad k = 0, \dots N-1 \\ x_0 = x^0. \end{cases}$$

En introduisant des multiplicateurs de Lagrange, cette discrétisation est équivalente à appliquer une méthode de Runge-Kutta partitionnée symplectique, ici la *méthode d'Euler symplectique* :

$$\begin{cases} x_{k+1} = x_k + hf(x_k, \overline{u}_k), \\ p_{k+1} = p_k - hp_{k+1}^T f_x(x_k, \overline{u}_k), \\ 0 = p_{k+1}^T f_u(x_k, \overline{u}_k), \text{ i.e } \overline{u}_k = \varphi(x_k, p_{k+1}), \\ x_0 = x^0, \quad p_N = \Phi'(x_N). \end{cases}$$

avec $k = 0, \dots N-1$. Ils démontrent que ceci est vrai pour toute discrétisation de Runge-Kutta.

L'objet du Chapitre 4 est d'étudier dans quelle mesure les excellentes performances des intégrateurs symplectiques pour l'intégration à long terme en astronomie et en dynamique moléculaire, s'étendent aux problèmes de contrôle optimal. On étudie d'abord le cas Martinet en géométrie sous-riemannienne. Après élimination du contrôle en utilisant le principe du maximum de Pontryagin, on aboutit à l'hamiltonien

$$H(q,p) = \frac{1}{2}\left(\left(p_x + p_z\frac{y^2}{2}\right)^2 + \frac{p_y^2}{(1+\beta x)^2}\right).$$

où $q = (x, y, z)^T$ est l'état, et $p = (p_x, p_y, p_z)^T$ est l'état adjoint. La dynamique intéressante se trouve dans l'espace à deux dimensions de coordonnées (y, p_y). En utilisant la théorie de l'analyse rétrograde, on montre que les intégrateurs symplectiques ont un net avantage pour le cas Martinet où $\beta = 0$ (voir Figure D.7) et aussi une perturbation non intégrable ($\beta = -10^{-4}$) (voir Figure D.8), même si l'intégration à long terme n'est pas centrale ici.

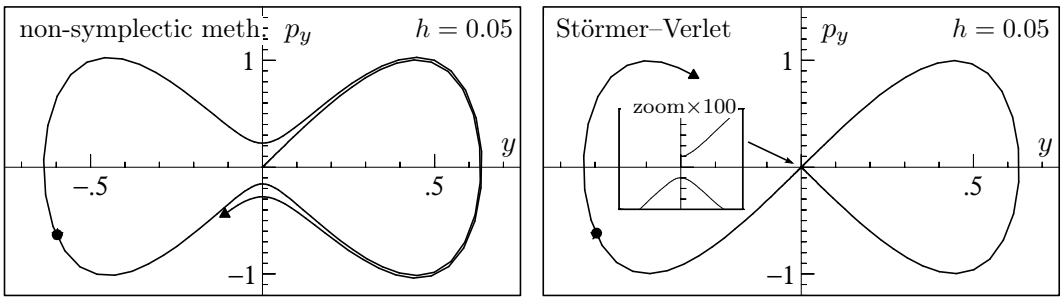


FIG. D.7 – Portrait de phase dans le plan (y, p_y) pour le cas plan $\beta = 0$.

Néanmoins, pour des problèmes comme le transfert orbital d'un satellite ou le contrôle d'un corps rigide sous-marin, un tel avantage ne peut être observé. Le système hamiltonien est un problème aux deux bouts et l'intervalle de temps n'est en général pas assez long pour que les intégrateurs symplectiques puissent bénéficier de leur préservation structurelle du flot.

L'analyse rétrograde pour des problèmes de contrôle optimal ? On discute également la possibilité d'étendre la théorie de l'analyse rétrograde et des équations modifiées aux intégrateurs symplectiques pour le contrôle optimal. On montre que cela est possible

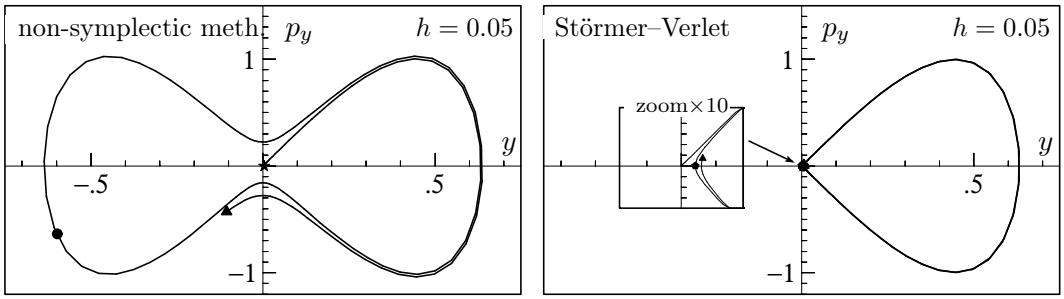


FIG. D.8 – Portrait de phase dans le plan (y, p_y) pour le cas non intégrable $\beta = -10^{-4}$.

pour les problèmes de contrôle optimal linéaires quadratiques, avec un état $x(t) \in \mathbb{R}^n$, et un contrôle $u(t) \in \mathbb{R}^m$,

$$\begin{cases} \text{Min } \frac{1}{2}\int_0^1 (x^T Z x + u^T S u)dt, \\ \dot{x} = Ax + Bu \\ x(0) \text{ donné} \end{cases}$$

où $A, Z \in \mathbb{R}^{n\times n}$, avec Z symétrique, $S \in \mathbb{R}^{m\times m}$ est symétrique définie positive, et $B \in \mathbb{R}^{n\times m}$ est de rang m. Précisément, on montre que la solution numérique obtenue en appliquant une méthode symplectique (par exemple une méthode de Runge-Kutta partitionnée symplectique) au système Hamiltonien dans la formulation de Pontryagin est formellement égale à la solution exacte d'un problème de contrôle modifié. En général, ce problème de contrôle perturbé n'est plus un problème de contrôle optimal mais peut être interprété comme un problème de contrôle stationnaire ou un problème de contrôle min-max, ou l'on a éventuellement ajouté des fonctions de contrôle supplémentaires. Par exemple, considérons la règle du point milieu pour le problème de contrôle optimal

$$\begin{cases} \text{Min } \frac{1}{2}\int_0^1 (x_1^2 + u_1^2)dt, \\ \dot{x}_1 = x_2 \\ \dot{x}_2 = -x_1 + u_1 \\ x_1(0), x_2(0) \text{ donnés} \end{cases}$$

La solution numérique peut être interprétée comme la solution exacte du problème de contrôle perturbé suivant où l'on a ajouté une variable de contrôle additionnelle u_2.

$$\begin{cases} \min\limits_{u_1}\max\limits_{u_2} \dfrac{1}{2}\displaystyle\int_0^1 \big(x_1^2 + u_1^2 - h^2 u_2^2 + \frac{h^2}{12}(-2x_1^2 - x_2^2) + \ldots\big)dt, \\ \dot{x}_1 = x_2 + \dfrac{h^2}{\sqrt{12}}u_2 - \dfrac{h^2}{12}x_2 + \ldots \\ \dot{x}_2 = -x_1 + u_1 - \dfrac{h^2}{12}u_1 + \ldots \\ x_1(0), x_2(0) \text{ donnés} \end{cases}$$

Ce résultat a une interprétation en théorie des jeux où le premier joueur contrôle u_1 pour minimiser la fonction de coût, tandis que u_2 essaye de maximiser le coût.

Chapitre 5 : Méthode de splitting avec potentiels modifiés

Dans le Chapitre 5, on considère des méthodes de splitting pour les systèmes hamiltoniens perturbés de la forme $H = H^A + H^B$, voir la présentation dans [MQ02]. Le champ de vecteurs $f(x) = J^{-1}\nabla H(x)$ est séparé en $f(x) = A(x) + B(x)$, et on suppose que les

flots des champs de vecteurs $A = J^{-1}\nabla H^A$ et $B = J^{-1}\nabla H^B$ peuvent être approchés efficacement soit exactement ou avec une grande précision. Une approche standard pour ce type de problèmes est de considérer des méthodes de splitting de la forme

$$e^{a_m hA} e^{b_m hB} e^{a_{m-1} hA} e^{b_{m-1} hB} \cdots e^{a_1 hA} e^{b_1 hB}$$

où e^{hA} et e^{hB} sont les flots associés à A et B. Dans le contexte de l'intégration géométrique, ce type d'intégrateur est d'un grand intérêt, car il préserve les propriétés qualitatives de la solution exacte. En effet, lorsque A et B sont des champs de vecteurs hamiltoniens, tous les flots $e^{a_i hA}$ et $e^{b_i hB}$ sont symplectiques, et la méthode de splitting qui en résulte est symplectique en tant que composition de flots symplectiques. Ceci garantit la bonne conservation de l'énergie sur les temps longs.

Pour réduire le nombre de compositions, et donc le coût de calcul, une amélioration substantielle est de considérer des méthodes avec processeur. Pour réduire le nombre d'évaluations à chaque pas d'intégration, l'idée des processeurs, initialement introduits par Butcher [But69] dans le contexte des méthodes de Runge-Kutta, est de considérer des compositions de la forme

$$e^P e^{hK} e^{-P}$$

où e^{hK} s'appelle le noyau et doit être peu coûteux, et l'ordre de $e^P e^{hK} e^{-P}$, appelé ordre effectif, est plus grand que celui de e^{hK}. En utilisant un pas de temps h constant, après N pas, on obtient $e^P (e^{hK})^N e^{-P}$. D'abord, on applique le processeur (ou correcteur) e^{-P}, ensuite e^{hK} une fois par pas, et le post-processeur e^P est évalué seulement lorsque un résultat est désiré.

Une analyse générale des méthodes de splitting symplectique avec processeur est donnée dans [BCR99].

En pratique, le principal outil pour l'obtention des contions d'ordre des méthodes de splitting est la formule de Baker-Campell-Hausdorff (BCH) (voir par exemple [HLW06, Sect. III.4.2]) qui implique que l'erreur locale pour ces méthodes est formellement une combinaison linéaire de crochets de Lie dans l'algèbre de lie engendrée par les champs de vecteurs A et B.

Maintenant, on suppose que le champs de vecteurs B est une petite perturbation du champ de vecteurs A,

$$B = \mathcal{O}(\varepsilon)$$

où ε est un petit paramètre. Dans ce cas, les crochets de Lie impliquant peu de B sont dominants et devraient donc être supprimés en priorité pour réduire l'erreur de la méthode. Par exemple, $[A,[A,B]] = \mathcal{O}(\varepsilon)$ est dominant comparé à $[B,[B,A]] = \mathcal{O}(\varepsilon^2)$. L'idée des processeurs a été utilisée pour l'intégration symplectique de systèmes hamiltoniens presque intégrables dans [WHT96, McL96].

Ces méthodes sont appelées 'méthodes de Runge-Kutta Nyström' dans [BCR01], car elles ont été introduites dans le cadre des équations différentielles du second ordre $\ddot{x} = f(x)$. Cependant, cette classe de méthodes ne s'applique pas seulement aux équations différentielles du second ordre, comme le problème à N corps avec coordonnées de Jacobi étudié dans [WH91].

La principale contribution de ce chapitre est la construction d'un nouveau processeur pour la méthode de Takahashi–Imada (une modification [Row91, TI86] du splitting de Stang),

$$e^{\frac{h}{2}B - \frac{h^3}{48}C} e^{hA} e^{\frac{h}{2}B - \frac{h^3}{48}C}.$$

pour atteindre l'ordre $\mathcal{O}(h^{10}\varepsilon + h^4\varepsilon^2)$. On montre également que cette classe de méthodes peut être appliquée avec succès aux problèmes de corps rigide asymétrique avec un potentiel externe :

- le corps pesant asymétrique (potentiel externe linéaire) ;
- un simulation de satellite (potentiel externe quadratique) ;
- une simulation en dynamique moléculaire : c'est un problème à N corps où N molécules d'eau sont modélisées comme des corps rigides asymétriques et interagissent comme des sphères molles magnétiques dipolaires.

Les expériences numériques montrent que cette méthode est très efficace pour ε petit, lorsque le coût d'évaluation du champ de vecteurs $C = [B, [B, A]]$ cumulé avec avec celui de B est petit comparé au coût d'évaluation de A et B seuls.

Chapitre 6 : Méthodes de splitting avec des coefficients complexes pour les équations paraboliques

Le dernier chapitre est dédié aux méthodes de splitting avec coefficients complexes pour les équations aux dérivées partielles paraboliques linéaires et non-linéaires. La résolution numérique de l'équation de la chaleur en plusieurs dimensions d'espace est maintenant bien connue mais il reste de nombreux défis en présence d'une source externe, par exemple pour les équations de réaction-diffusion, ou plus généralement pour l'équation complexe de Ginzburg-Landau. D'un point de vue mathématique, ils appartiennent à la classe des équations aux dérivées partielles semi-linéaires paraboliques et peuvent être représentés sous la forme générale

$$\frac{\partial u}{\partial t} = D\Delta u + F(u),$$

où chaque composante du vecteur $u(x,t) \in \mathbb{R}^d$ représente la population d'une espèce, D est la matrice des coefficients de diffusion (souvent diagonale) et F correspond à toutes les interactions locales entre espèces. Les solutions des équations de réaction-diffusion présentent un large éventail de comportements, comme les ondes voyageuses, les phénomènes d'ondes, et les solitons dissipatifs. Considérons le cas linéaire

$$\frac{\partial u}{\partial t} = \Delta u + Vu, \tag{D.18}$$

où V est un opérateur linéaire, par exemple $Vu = v(x)u$ avec $v(x)$ une fonction lisse.

L'idée naturelle des méthodes de splitting consiste à combiner les exponentielles d'opérateurs $e^{t\Delta}$ et e^{tV} de telle manière que l'approximation obtenue soit aussi précise que possible. La généralisation au cas non-linéaire est immédiate en remplaçant e^{tV} par le flot de l'équation différentielle associée.

Pour un pas $h > 0$, l'intégrateur numérique le plus simple est la méthode de Lie-Trotter

$$e^{hV}e^{h\Delta} \tag{D.19}$$

qui est une approximation d'ordre 1 de la solution de (D.18), et la version symétrique

$$e^{h/2\,V}e^{h\Delta}e^{h/2\,V} \tag{D.20}$$

est connue sous le nom de splitting de Strang et est une approximation d'ordre 2. Pour des ordres plus élevés, on peut considérer des méthodes de splittting de la forme

$$e^{b_1 hV}e^{a_1 h\Delta}e^{b_2 hV}e^{a_2 h\Delta}\dots e^{b_s hV}e^{a_s h\Delta}. \tag{D.21}$$

Cependant, construire des méthodes d'ordre supérieur à la méthode de Strang (D.20) n'est pas aisé. Un résultat négatif montre que toute méthode de splitting (ou méthode de composition) avec des coefficients réels d'ordre strictement plus grand que 2 possède nécessairement un coefficient négatif à la fois pour a_i et b_i. Ainsi, de telles méthodes de splitting avec coefficients réels ne peuvent pas être utilisées lorsque un opérateur comme Δ n'est pas réversible en temps et ne peut donc pas être résolu pour des temps négatifs. L'existence d'au moins un coefficient négatif est montrée dans [She89, SW92], et le fait que cela concerne les deux opérateurs est prouvée dans [GK96]. Une élégante preuve géométrique est proposée dans [BC05].

Pour contourner cette barrière d'ordre, il y a deux possibilités. On peut utiliser une combinaison linéaire non-convexe de produits de type (D.21), qui peuvent être obtenus par une technique d'extrapolation [Sch02, Des01]. Une autre possibilité est de considérer des méthodes de splitting avec des coefficients complexes de partie réelle positive. L'idée d'utiliser des coefficients complexes dans des méthodes numériques n'est pas nouvelle. Déjà en 1962/1963, Rosenbrock [Ros63] considéra des coefficients complexes pour ses méthodes. Dans [GRT02, GRT04], cette idée est exploitée pour atteindre les ordres 3 et 4 en utilisant une combinaison convexe de produit d'exponentielles avec coefficients complexes. Concernant les méthodes de splitting, cette liberté supplémentaire nous permet de construire des méthodes d'ordre élevé.

Il est intéressant de remarquer qu'on peut aussi augmenter l'ordre en considérant des méthodes de composition de la forme

$$\Phi_{\gamma_s h} \circ \ldots \circ \Phi_{\gamma_1 h} \tag{D.22}$$

où pour des méthodes symétriques $\gamma_j = \gamma_{s+1-j}$, et Φ_h est une méthode symétrique, par exemple le Strang splitting (D.20) d'ordre 2. On obtient alors la méthode de splitting

$$\Psi_h = e^{h\gamma_s/2\,V} e^{h\gamma_s \Delta} e^{h(\gamma_s+\gamma_{s-1})/2\,V} e^{h\gamma_{s-1}\Delta} \ldots e^{h\gamma_1\Delta} e^{h\gamma_1/2\,V}.$$

L'avantage de cette approche avec des méthodes de composition est que l'on peut remplacer la méthodes de Strang avec des exponentielles (D.20) par une discrétisation symétrique, par exemple

$$\Phi_h = \Phi^I_{h/2} \circ \Phi^M_h \circ \Phi^E_{h/2}$$

où Φ^E_h est le flot de la méthode d'Euler explicite $y_{n+1} = y_n + hf(y_n)$ et Φ^I_h est le flot de la méthode d'Euler implicite $y_{n+1} = y_n + hf(y_{n+1})$ pour approcher la réaction, et Φ^M_h est la discrétisation de Crank-Nicholson (équivalente à la règle du point milieu pour des problèmes linéaires)

$$\Phi^M_h = \left(Id - \frac{h}{2}\Delta\right)^{-1} \left(Id + \frac{h}{2}\Delta\right).$$

On appelle cela la formule de Peaceman-Rachford [PJ55] développée à l'origine pour l'équation de la chaleur, et étendue aux problèmes de réaction-diffusion dans [DR03].

La contribution de ce chapitre est la suivante. On considère des méthodes de splitting de la forme (D.21), et on construit de nouvelles méthodes d'ordre élevé en utilisant les techniques de composition (D.22) initialement développées pour l'intégration numérique géométrique des équations différentielles ordinaires [HLW06]. On donne une justification théorique de l'ordre de convergence des méthodes introduites dans le cas linéaire pour des opérateurs exponentiels en s'appuyant sur le récent résultat de convergence de Hansen & Ostermann [HO08a]. Les principaux avantages de cette approche sont les suivants :

– la méthode de splitting hérite des propriétés de stabilité des opérateurs exponentiels ;

– on peut remplacer les coûteux opérateurs exponentiels par des approximations moins coûteuses d'ordre faible sans altérer l'ordre de convergence de la méthode de splitting obtenue.

Nos simulations numériques montrent que l'ordre de convergence est celui attendu en particulier dans le cas d'un terme de source non-linéaire et pour la discrétisation de Peaceman-Rachford.

Bibliography

[ABCK97] A. Agrachev, B. Bonnard, M. Chyba, and I. Kupka. Sub-Riemannian sphere in Martinet flat case. *ESAIM/COCV (Control, Optimisation and Calculus of Variations)*, 2:377–448, 1997.

[BC05] S. Blanes and F. Casas. On the necessity of negative coefficients for operator splitting schemes of order higher than two. *Appl. Num. Math.*, 54:23–37, 2005.

[BCF01] G. Benettin, A. M. Cherubini, and F. Fassò. A changing-chart symplectic algorithm for rigid bodies and other Hamiltonian systems on manifolds. *SIAM J. Sci. Comput.*, 23:1189–1203, 2001.

[BCK99] B. Bonnard, M. Chyba, and I. Kupka. Non integrable geodesics in SR-Martinet geometry. *Proceedings of the 1997 Summer Research Institute on Differential Geometry and Control, University of Colorado, Boulder. - Providence R.I.*, 64:119–134, 1999.

[BCMR02] A. M. Bloch, P. E. Crouch, J. E. Marsden, and T. S. Ratiu. The symmetric representation of the rigid body equations and their discretization. *Nonlinearity*, 15:1309–1341, 2002.

[BCR99] S. Blanes, F. Casas, and J. Ros. Symplectic integrators with processing: a general study. *SIAM J. Sci. Comput.*, 21:149–161, 1999.

[BCR00] S. Blanes, F. Casas, and J. Ros. Processing symplectic methods for near-integrable Hamiltonian systems. *Celestial Mech. Dynam. Astronom.*, 77:17–35, 2000.

[BCR01] S. Blanes, F. Casas, and J. Ros. High-order Runge-Kutta-Nyström geometric methods with processing. *Appl. Num. Math.*, 39:245–259, 2001.

[BDL06] A. Bandrauk, E. Dehghanian, and H. Lu. Complex integration steps in decomposition of quantum exponential evolution operators. *Chem. Phys. Lett.*, 419:346–350, 2006.

[BF07] L. Bates and F. Fassò. The conjugate locus for the Euler top. i. The axisymmetric case. *International Mathematical Forum*, 2(43):2109–2139, 2007.

[BG94] G. Benettin and A. Giorgilli. On the Hamiltonian interpolation of near to the identity symplectic mappings with application to symplectic integration algorithms. *J. Statist. Phys.*, 74:1117–1143, 1994.

[BH69] A. E. Bryson and Y.-C. Ho. *Applied Optimal Control*. Blaisdell, Waltham, MA, 1969.

[BLT99] B. Bonnard, G. Launay, and E. Trelat. The transcendence needed to compute the sphere and wave front in martinet sr-geometry. *J. Math. Sci., New York*, 103(6):688–708, 1999.

[BLV06] J. F. Bonnans and J. Laurent-Varin. Computation of order conditions for symplectic partitioned Runge-Kutta schemes with application to optimal control. *Numer. Math.*, 103(1):1–10, 2006.

[Bro37] D. Brouwer. On the accumulation of errors in numerical integration. *Astronomical Journal*, 46:149–153, 1937.

[Bro00] Ch. Brouder. Runge-Kutta methods and renormalization. *Euro. Phys. J. C*, 12:521–534, 2000.

[Bro04] Ch. Brouder. Trees, renormalization and differential equations. *BIT*, 44(3):425–438, 2004.

[BT01] B. Bonnard and E. Trelat. On the role of abnormal minimizers in sr-geometry. *Ann. Fac. Sci. Toulouse*, 6(X, 3):405–491, 2001.

[But63] J. C. Butcher. Coefficients for the study of Runge-Kutta integration processes. *J. Austral. Math. Soc.*, 3:185–201, 1963.

[But64a] J. C. Butcher. Implicit Runge-Kutta processes. *Math. Comput.*, 18:50–64, 1964.

[But64b] J. C. Butcher. Integration processes based on Radau quadrature formulas. *Math. Comput.*, 18:233–244, 1964.

[But69] J. C. Butcher. The effective order of Runge-Kutta methods. In J. Ll. Morris, editor, *Proceedings of Conference on the Numerical Solution of Differential Equations*, volume 109 of *Lecture Notes in Math.*, pages 133–139, 1969.

[But72] J. C. Butcher. An algebraic theory of integration methods. *Math. Comput.*, 26:79–106, 1972.

[CCDV08] F. Castella, P. Chartier, S. Descombes, and G. Vilmart. Splitting methods with complex times for parabolic equations. *Submitted*, 2008.

[CEFM08] D. Calaque, K. Ebrahimi-Fard, and D. Manchon. Two Hopf algebras of trees interacting. *Submitted*, 2008.

[CFM06] P. Chartier, E. Faou, and A. Murua. An algebraic approach to invariant preserving integrators: the case of quadratic and Hamiltonian invariants. *Numer. Math.*, 103:575–590, 2006.

[CFSZ08] E. Celledoni, F. Fassò, N. Säfström, and A. Zanna. The exact computation of the free rigid body motion and its use in splitting methods. *SIAM J. Sci. Comp.*, 30(4):2084–2112, 2008.

[CG89] M. Creutz and A. Gocksch. Higher-order hybrid Monte Carlo algorithms. *Phys. Rev. Lett.*, 63:9–12, 1989.

[Cha03] J. E. Chambers. Symplectic integrators with complex time steps. *Astron. J.*, 126:1119–1126, 2003.

[CHSC08a] M. Chyba, T. Haberkorn, R.N. Smith, and S.K. Choi. Autonomous underwater vehicles: Development and implementation of time and energy efficient trajectories. *Ship Technology Research*, 55(2):36–48, 2008.

[CHSC08b] M. Chyba, T. Haberkorn, R.N. Smith, and S.K. Choi. Design and implementation of time efficient trajectories for underwater vehicles. *IEEE Journal of Ocean Engineering*, 2008. (doi:10.1016/j.oceaneng.2007.07.007).

[CHV05] P. Chartier, E. Hairer, and G. Vilmart. A substitution law for B-series vector fields. *INRIA Report, No. 5498*, 2005.

[CHV06] M. Chyba, E. Hairer, and G. Vilmart. Symplectic integrators in sub-Riemannian geometry: the Martinet case. *INRIA Report, No. 6017*, 2006.

[CHV07a] P. Chartier, E. Hairer, and G. Vilmart. Modified differential equations. *ESAIM: Proc.*, 21:16–20, 2007.

[CHV07b] P. Chartier, E. Hairer, and G. Vilmart. Numerical integrators based on modified differential equations. *Math. Comp.*, 76:1941–1953, 2007.

[CHV08] M. Chyba, E. Hairer, and G. Vilmart. The role of symplectic integrators in optimal control. *Optimal Control, Applications and Methods*, 2008. (doi:10.1002/oca.855).

[CK98] A. Connes and D. Kreimer. Hopf algebras, renormalization and noncommutative geometry. *Commun. Math. Phys.*, 199(1):203–242, 1998.

[CK00] A. Connes and D. Kreimer. Renormalization in quantum field theory and the Riemann–Hilbert problem. I. the Hopf algebra structure of graphs and the main theorem. *Commun. Math. Phys.*, 210(1):249–273, 2000.

[CL98] P. Chartier and E. Lapôtre. Reversible B-series. *INRIA Report, No. 1221*, 1998.

[CM98] A. Connes and H. Moscovici. Hopf algebras, cyclic cohomology and the transverse index theorem. *Comm. Math. Phys.*, 198(1):199–246, 1998.

[CMSS94] M. P. Calvo, A. Murua, and J. M. Sanz-Serna. Modified equations for odes. *Contemporary Mathematics*, 172:63–74, 1994.

[Cro05] M. Crouzeix. Approximation of parabolic equations, 2005. Lecture notes available at `http://perso.univ-rennes1.fr/michel.crouzeix/`.

[CS90] P. J. Channell and J. C. Scovel. Symplectic integration of Hamiltonian systems. *Nonlinearity*, 3:231–259, 1990.

[CS06] E. Celledoni and N. Säfström. Efficient time-symmetric simulation of torqued rigid bodies using Jacobi elliptic functions. *J. Phys. A*, 39:5463–5478, 2006.

[CSS94] M. P. Calvo and J. M. Sanz-Serna. Canonical B-series. *Numer. Math.*, 67:161–175, 1994.

[Des01] S. Descombes. Convergence of a splitting method of high order for reaction-diffusion systems. *Math. Comp.*, 70(236):1481–1501 (electronic), 2001.

[Dia96] B.O. Dia. *Méthodes de directions alternées d'ordre élevé en temps*. PhD thesis, Université Claude Bernard Lyon 1., 1996.

[DLM97] A. Dullweber, B. Leimkuhler, and R. McLachlan. Symplectic splitting methods for rigid body molecular dynamics. *J. Chem. Phys. 107 No.*, 15:5840–5851, 1997.

[DR03] S. Descombes and M. Ribot. Convergence of the Peaceman-Rachford approximation for reaction-diffusion systems. *Numer. Math.*, 95(3):503–525, 2003.

[Dür86] A. Dür. Möbius functions, incidence algebras and power series representations. In *Lecture Notes in Math.*, volume 1202. Springer-Verlag, 1986.

[Eul68] L. Euler. *Institutionum Calculi Integralis*. Opera omnia, 1768.

[Eva83] L. C. Evans. An introduction to mathematical optimal control theory, version 0.1. *Lecture notes available at* `http://math.berkeley.edu/~evans/control.course.pdf`, 1983.

[Fas03] F. Fassò. Comparison of splitting algorithm for the rigid body. *J. Comput. Phys.*, 189:527–538, 2003.

[Fen86] K. Feng. Difference schemes for Hamiltonian formalism and symplectic geometry. *J. Comp. Math.*, 4:279–289, 1986.

[FGS05] F. Fassò, A. Giacobbe, and N. Sansonetto. Periodic flows, rank-two Poisson structures, and nonholonomic mechanics. *Regular and Chaotic Dynamics*, 10(3):267–284, 2005.

[FHP04] E. Faou, E. Hairer, and T.-L. Pham. Energy conservation with non-symplectic methods: examples and counter-examples. *BIT*, 44:699–709, 2004.

[For89] E. Forest. Canonical integrators as tracking codes. *AIP Conference Proceedings*, 184:1106–1136, 1989.

[FT88] S. Fauve and O. Thual. Localized structures generated by subcritical instabilities. *J. Phys. France*, 49:1829–1833, 1988.

[FWQW89] K. Feng, H. M. Wu, M.-Z. Qin, and D. L. Wang. Construction of canonical difference schemes for Hamiltonian formalism via generating functions. *J. Comp. Math.*, 7:71–96, 1989.

[GB04] V. Guibout and A. Bloch. A discrete maximum principle for solving optimal control problems. *43rd IEEE Conference on Decision and Control, Atlantis, Paradise Island, Bahamas*, 2:1806–1811, 2004.

[GK96] D. Goldman and T. J. Kaper. nth-order operator splitting schemes and nonreversible systems. *SIAM J. Numer. Anal.*, 33:349–367, 1996.

[GM88] Z. Ge and J. E. Marsden. Lie-Poisson Hamilton-Jacobi theory and Lie-Poisson integrators. *Phys. Lett. A*, 133:134–139, 1988.

[GRT02] Z. Gegechkori, J. Rogava, and M. Tsiklauri. High degree precision decomposition method for the evolution problem with an operator under a split form. *M2AN Math. Model. Numer. Anal.*, 36(4):693–704, 2002.

[GRT04] Zurab Gegechkori, Jemal Rogava, and Mikheil Tsiklauri. The fourth order accuracy decomposition scheme for an evolution problem. *M2AN Math. Model. Numer. Anal.*, 38(4):707–722, 2004.

[Hag99] W. Hager. Optcon_xrk, Fortran software for solving unconstrained control problems using explicit Runge-Kutta discretizations. `http://www.math.ufl.edu/~hager/papers/optcon_xrk`, 1999.

[Hag00] W. Hager. Runge-Kutta methods in optimal control and the transformed adjoint system. *Numer. Math.*, 87(2):247–282, 2000.

[Hai94] E. Hairer. Backward analysis of numerical integrators and symplectic methods. *Annals of Numerical Mathematics*, 1:107–132, 1994.

[Hai99] E. Hairer. Backward error analysis for multistep methods. *Numer. Math.*, 84:199–232, 1999.

[Hai05] E. Hairer. Long-time energy conservation of numerical integrators. *Found. of Comput. Math.*, pages 162–180, 2005.

[Hen62] P. Henrici. *Discrete Variable Methods in Ordinary Differential Equations*. John Wiley & Sons Inc., New York, 1962.

[Heu00] K. Heun. Neue Methode zur approximativen Integration der differentialgleichungen einer unabhängigen veränderlichen. *Zeitschr. für Math. u. Phys.*, 45:23–38, 1900.

[Hig93] N. J. Higham. The accuracy of floating point summation. *SIAM J. Sci. Comput.*, 14:783–799, 1993.

[HLW06] E. Hairer, C. Lubich, and G. Wanner. *Geometric Numerical Integration. Structure-Preserving Algorithms for Ordinary Differential Equations*. Springer Series in Computational Mathematics 31. Springer-Verlag, Berlin, second edition, 2006.

[HMR08] E. Hairer, R.I. McLachlan, and A. Razakarivony. Achieving Brouwer's law with implicit Runge-Kutta methods. *BIT*, 48:231–243, 2008.

[HMS08] E. Hairer, R. I. McLachlan, and R. D. Skeel. On energy conservation of the simplified Takahashi–Imada method. *To appear in M2AN Math. Model. Numer. Anal.*, 2008.

[HO08a] E. Hansen and A. Ostermann. Exponential splitting for unbounded operators. *To appear in Math. Comp.*, 2008.

[HO08b] E. Hansen and A. Ostermann. High order splitting methods for analytic semigroups exist. *Submitted*, 2008.

[HV06] E. Hairer and G. Vilmart. Preprocessed Discrete Moser-Veselov algorithm for the full dynamics of the rigid body. *J. Phys. A*, 39:13225–13235, 2006.

[HW74] E. Hairer and G. Wanner. On the Butcher group and general multi-value methods. *Computing*, 13:1–15, 1974.

[Jac50] C. G. J. Jacobi. Sur la rotation d'un corps. *Journal für die reine und angewandte Matematik (Journal de Crelle)*, 39:293–350, 1850. (lu dans la séance du 30 juillet 1849 à l'académie des sciences de Paris).

[Kah65] W. Kahan. Further remarks on reducing truncation errors. *Comm. ACM*, 8:40, 1965.

[Kos94] P.-V. Koseleff. *Formal calculus for Lie methods in Hamiltonian mechanics.* PhD thesis, Lawrence Berkeley Laboratory LBID-2030, 1994. Tech. Report UC-405, Berkeley, CA.

[Koz07] R. Kozlov. Conservative discretizations of the Kepler motion. *J. Phys. A*, 40:4529–4539, 2007.

[KPP37] A.N. Kolmogoroff, I.G. Petrovsky, and N.S. Piscounoff. Etude de l'équation de la diffusion avec croissance de la quantité de matière et son application a un problème biologique. *Bulletin de l'Université d'état de Moscou, Série Internationale Section A Mathématiques et Mécanique*, 1:1–25, 1937.

[KS65] P. Kustaanheimo and E. Stiefel. Perturbation theory of Kepler motion based on spinor regularization. *J. Reine Angew. Math.*, 218:204–19, 1965.

[Kut01] W. Kutta. Beitrag zur näherungsweisen Integration totaler Differentialgleichungen. *Zeitschr. für Math. u. Phys.*, 46:435–453, 1901.

[LLM06] M. Leok, T. Lee, and N. H. McClamroch. Attitude maneuvers of a rigid spacecraft in a circular orbit. In *Proceedings of the IEEE American Control Conference*, pages 1742–1747, 2006.

[LR01] J. Laskar and P. Robutel. High order symplectic integrators for perturbed Hamiltonian systems. *Celest. Mech.*, 80:39–62, 2001.

[LR04] B. Leimkuhler and S. Reich. *Simulating Hamiltonian Dynamics.* Cambridge Monographs on Applied and Computational Mathematics 14. Cambridge University Press, Cambridge, 2004.

[LS96] D. Lewis and J. C. Simo. Conserving algorithms for the n-dimensional rigid body. *Fields Inst. Commun.*, 10:121–139, 1996.

[McL95] R. I. McLachlan. Composition methods in the presence of small parameters. *BIT*, 35:258–268, 1995.

[McL96] R. I. McLachlan. More on symplectic integrators. In J. E. Marsden, G. W. Patrick, and W. F. Shadwick, editors, *Integration Algorithms and Classical Mechanics*, volume 10, pages 141–149. Amer. Math. Soc., Providence, R. I., 1996.

[McL07] Robert I. McLachlan. A new implementation of symplectic Runge-Kutta methods. *SIAM J. Sci. Comput.*, 29(4):1637–1649, 2007.

[Mey99] K. R. Meyer. Periodic solutions of the N-body problem. In *Lecture Notes in Math. 1719.* Springer-Verlag, 1999.

[Mit00] J. Wm. Mitchell. *A simplified variation of parameters solution for the motion of an arbitrarily torqued mass asymmetric rigid body.* PhD thesis, University of Cincinnati, 2000.

[MKW08] H. Munthe-Kaas and W. Wright. On the Hopf algebraic structure of Lie group integrators. *Found. Comput. Math.*, 8(2):227–257, 2008.

[MN02] Y. Minesaki and Y. Nakamura. A new discretization of the Kepler motion which conserves the Runge-Lenz vector. *Phys. Lett. A*, 306:127–133, 2002.

[MN04] Y. Minesaki and Y. Nakamura. A new conservative numerical integration algorithm for the three-dimensional Kepler motion based on the Kustaanheimo–Stiefel regularization theory. *Phys. Lett. A*, 324:282–92, 2004.

[Møl65] O. Møller. Quasi double-precision in floating point addition. *BIT*, 5:251–255, 1965.

[MQ02] R. I. McLachlan and G. R. W. Quispel. Splitting methods. *Acta Numerica*, 11:341–434, 2002.

[MS82] J. Macki and A. Strauss. *Introduction to optimal control theory.* Springer-Verlag, New York, 1982. Undergraduate Texts in Mathematics.

[Mur94] A. Murua. *Métodos simplécticos desarrollables en P-series.* PhD thesis, Univ. Valladolid, 1994.

[Mur06] A. Murua. The Hopf algebra of rooted trees, free Lie algebras, and Lie series. *Found. Comput. Math.*, 6(4):387–426, 2006.

[MV91] J. Moser and A. P. Veselov. Discrete versions of some classical integrable systems and factorization of matrix polynomials. *Comm. Math. Phys.*, 139:217–243, 1991.

[MW01] J. E. Marsden and M. West. Discrete mechanics and variational integrators. *Acta Numerica*, 10:1–158, 2001.

[MZ05] R. I. McLachlan and A. Zanna. The discrete Moser–Veselov algorithm for the free rigid body, revisited. *Found. Comput. Math.*, 5:87–123, 2005.

[New87] I. Newton. *Philosophiae Naturalis Principia Mathematica.* Londini, 1687.

[OMF03] I.P. Omelyan, I.M. Mryglod, and R. Folk. Symplectic analytically integrable decomposition algorithms: classification, derivation, and application to molecular dynamics, quantum and celestial mechanics simulations. *Comput. Phys. Comm.*, 151:272–314, 2003.

[OW08] A. Ostermann and G. Wanner. *Geometry by its history.* Undergraduate Texts in Mathematics. Springer-Verlag, in preparation, 2008.

[PJ55] D. W. Peaceman and H. H. Rachford Jr. The numerical solution of parabolic and elliptic differential equations. *J. Soc. Indust. Appl. Math.*, 3:28–41, 1955.

[Poi92] H. Poincaré. *Les Méthodes Nouvelles de la Mécanique Céleste Tome I.* Gauthier-Villars, Paris, 1892.

[Ros63] H. H. Rosenbrock. Some general implicit processes for the numerical solution of differential equations. *Comput. J.*, 5:329–330, 1962/1963.

[Row91] G. Rowlands. A numerical algorithm for Hamiltonian systems. *J. Comput. Phys.*, 97:235–239, 1991.

[Run95] C. Runge. Ueber die numerische Auflösung von Differentialgleichungen. *Math. Ann.*, 46:167–178, 1895.

[Rut83] R. D. Ruth. A canonical integration technique. *IEEE Trans. Nuclear Science*, NS-30:2669–2671, 1983.

[Sch02] M. Schatzman. Numerical integration of reaction-diffusion systems. *Numer. Algorithms*, 31(1-4):247–269, 2002. Numerical methods for ordinary differential equations (Auckland, 2001).

[She89] Q. Sheng. Solving linear partial differential equations by exponential splitting. *IMA J. Numer. Anal.*, 9:199–212, 1989.

[SSC94] J. M. Sanz-Serna and M. P. Calvo. *Numerical Hamiltonian Problems*. Chapman & Hall, London, 1994.

[Sto88] D. Stoffer. On reversible and canonical integration methods. Technical Report SAM-Report No. 88-05, ETH-Zürich, 1988.

[Suz90] M. Suzuki. Fractal decomposition of exponential operators with applications to many-body theories and Monte Carlo simulations. *Phys. Lett. A*, 146:319–323, 1990.

[SW92] G. J. Sussman and J. Wisdom. Chaotic evolution of the solar system. *Science*, 257:56–62, 1992.

[Tan94] Y.-F. Tang. Formal energy of a symplectic scheme for Hamiltonian systems and its applications I. *Computers Math. Applic.*, 27:31–39, 1994.

[TI86] M. Takahashi and M. Imada. Monte Carlo calculation of quantum systems. II. Higher order correction. *J. Phys. Soc. Jpn.*, 53:3765–3769, 1986.

[Vil08] G. Vilmart. Reducing round-off errors in rigid body dynamics. *J. Comput. Phys.*, 227:7083–7088, 2008.

[vZS07a] R. van Zon and J. Schofield. Numerical implementation of the exact dynamics of free rigid bodies. *J. Comput. Phys.*, 225(1):145–164, 2007.

[vZS07b] R. van Zon and J. Schofield. Symplectic algorithms for simulations of rigid body systems using the exact solution of free motion. *Phys. Rev. E*, 75:056701, 2007.

[WH91] J. Wisdom and M. Holman. Symplectic maps for the n-body problem. *Astron. J.*, 102:1528–1538, 1991.

[WHT96] J. Wisdom, M. Holman, and J. Touma. Symplectic correctors. In J. E. Marsden, G. W. Patrick, and W. F. Shadwick, editors, *Integration Algorithms and Classical Mechanics*, pages 217–244. Amer. Math. Soc., Providence R. I., 1996.

[WM97] J. M. Wendlandt and J. E. Marsden. Mechanical integrators derived from a discrete variational principle. *Physica D*, 106:223–246, 1997.

[Yos90] H. Yoshida. Construction of higher order symplectic integrators. *Phys. Lett. A*, 150:262–268, 1990.

[Zan05] A. Zanna. A note on the implicit midpoint rule and the Euler equations for the rigid body. *Private communication*, 2005.

List of Figures

List of Tables