9-2015

Geometric Integrators with Application To Hamiltonian Systems

Hebatallah Jamil Al Sakaji

United Arab Emirates University

College of Science

Department of Mathematical Sciences

GEOMETRIC INTEGRATORS WITH APPLICATION TO HAMILTONIAN SYSTEMS

Hebatallah Jamil Al Sakaji

This thesis is submitted in partial fulfilment of the requirements for the degree of Master of Science in Mathematics

Under the Supervision of Dr. Anwar Hussein

September 2015

Declaration of Original Work

I, Hebatallah Jamil Alsakaji, the undersigned, a graduate student at the United Arab Emirates University (UAEU), and the author of this thesis entitled *"Geometric Integrators with Applications to Hamiltonian Systems"*, hereby, solemnly declare that this thesis is an original research work that has been done and prepared by me under the supervision of Dr. Anwar Hussein, in the College of Science at UAEU. This work has not previously been presented or published, or formed the basis for the award of any academic degree, diploma or a similar title at this or any other university. Any materials borrowed from other sources (whether published or unpublished) and relied upon or included in my thesis have been properly cited and acknowledged in accordance with appropriate academic conventions. I further declare that there is no potential conflict of interest with respect to the research, data collection, authorship, presentation and/or publication of this thesis.

Student's Signature _____ Date _____

Advisory Committee

1) Advisor: Dr. Anwar Hussein

Title: Assistant Professor

Department of Mathematical Sciences

College of Science

2) Co-advisor: Dr. Youssef EL-Khatib

Title: Associate Professor

Department of Mathematical Sciences

College of Science

3) Member (External Examiner): Prof. Hayder Akea

Title: Assistant Professor

Department of Applied Mathematics and Sciences

Institution: Abu Dhabi University

Approval of the Master Thesis

This Master Thesis is approved by the following Examining Committee Members:

1) Advisor (Committee Chair): Dr. Anwar Hussein

Title: Assistant Professor

Department of Mathematical Sciences

College of Science

Signature _____ Date _____

2) Member: Dr. Youssef EL-Khatib

Title: Associate Professor

Department of Mathematical Sciences

College of Science

Signature _____ Date _____

3) Member (External Examiner): Prof. Hayder Akea

Title: Assistant Professor

Department of Applied Mathematics and Sciences

Institution: Abu Dhabi University

Signature _____ Date _____

This Master Thesis is accepted by:

Dean of the College of Science: Dr. Ahmed Murad (Acting Dean)

Signature _____ Date _____

Dean of the College of Graduate Studies: Professor Nagi T. Wakim

Signature _____ Date _____

Copy _____ of _____

Abstract

Geometric numerical integration is a relatively new area of numerical analysis. The aim is to preserve the geometric properties of the flow of a differential equation such as symplecticity or reversibility. A conventional numerical integrator approximates the flow of the continuous-time equations using only the information about the vector field, ignoring the physical laws and the properties of the original trajectory. In this way, small inaccuracies accumulated over long periods of time will significantly diminish the operational lifespan of such discrete solutions. Geometric integrators, on the other hand, are built in a way that preserve the structure of continuous dynamics, so maintaining the qualitative behaviour of the exact flow even for long-time integration. The aim of this thesis is to design efficient geometric integrators for Hamiltonian systems and to illustrate their effectiveness. These methods are implicit for general (non-separable) Hamiltonian systems making them difficult to implement. However, We show that explicit integrators are possible in some cases. Both geometric and non-geometric integration methods are applied to several problems, then We do a comparison between these methods, in order to determine which of those quantities are preserved better by these methods. In particular, We develop explicit integrators for a special case of the restricted 3-body problem known as Hill's problem.

Keywords: Hamiltonian systems, geometric integrators, dynamical systems, Hill's problem, Splitting and Composition methods.

Title and Abstract (in Arabic)

تكامل هندسي مع تطبيق لأنظمة هاملتون

الملخص

التكامل العددي الهندسي هو مجال جديد نسبيا من التحليل العددي. والهدف من ذلك هو الحفاظ على الخصائص الهندسية للتدفق معادلة تفاضلية مثل الربط أو العودة إلى الوراء. تكامل العددي التقليدي الذي يقارب تدفق معادلات الزمن المتواصل فقط باستخدام المعلومات حول الحقل ناقلات، وتجاهل القوانين الفيزيائية وخصائص المسار الأصلي. وبهذه الطريقة سوف يتم حل المغالطات الصغيرة التي تراكمت على مدى فترات طويلة من الوقت تقلل إلى حد كبير من العمر التشغيلي لهذه الحلول منفصلة. تكامل الهندسية، من ناحية أخرى، يتم بناؤها بطريقة تحافظ على بنية ديناميكية مستمرة، وبالتالي الحفاظ على السلوك النوعي لتدفق المحدد حتى للاندماج منذ فترة طويلة. والهدف من هذه الرسالة هو تصميم تكامل هندسية فعالة لأنظمة هاملتون وتوضيح فعاليتها. هذه الأساليب هي ضمني لعام نظم (غير قابل للانفصال) هاملتون مما يجعل من الصعب تنفيذها. ومع ذلك، نظهر أن تكامل صريحة من الممكن في بعض الحالات. كما يمكن تطبق أساليب التكامل هندسية وغير هندسية للعديد من المشاكل، ثم نحن نفعل المقارنة بين هذه الأساليب، من أجل تحديد أي من تلك الكميات يتم الاحتفاظ بشكل أفضل من خلال هذه الأساليب. على وجه الخصوص، ونحن في وضع تكامل واضحة لحالة خاصة من تقييد مشكلة حركة الاجسام الثلاثة المعروفة باسم مسألة هيل.

الكلمات الرئيسية: نظم هاملتون، تكاملات هندسية، الأنظمة الديناميكية، مسألة هيل، التجزئة والتجميع.

Acknowledgements

This thesis is the consequence of a two-year study evolved by the contribution of many people and now I would like to express my gratitude to all the people supporting me from all the aspects for the period of my thesis. I would like to thank and express my deepest gratitude to Dr. Anwar Hussein, my advisor, for his help, guidance, understanding, encouragement and patience during my studies and preparation of this thesis. Furthermore, I would like to thank Prof.Hayder Akca and Dr.Youssef El-khatib for their support. I am also grateful to my family for their confidence in me and for their endless support.

Dedication

To my beloved parents and family

Table of Contents

List of Figures

Chapter 1: Introduction

The topic of this thesis is geometric integrators with applications to Hamiltonian systems. We will concentrate to give an introduction to the relatively new area of numerical analysis called geometric integration. This is an overview applied to a series of numerical methods, that aim to preserve the qualitative features of a diffrential equations.

During the past decade there has been an increasing interest in studying numerical methods that preserve certain properties of some differential equations. The reason is that some physical systems possess conserved quantities, and that the solutions of the systems also should contain these invariants.

In this thesis we are mainly concerned with symplectic geometric integration methods. We will give the definition of symplectic maps, and call any numerical scheme which induces a symplectic map as the symplectic numerical method. The results obtained in this thesis centers around solving some certain types of dynamical systems of Hamiltonian type using symplectic integration. More details on these topics can be found in several recent books [3, 7, 14], and survey articles [1, 2, 4, 8, 9, 12]. In particular, we solve a special case of the restricted three body problem by first splitting the Hamiltonian into Linear and nonlinear parts and solving each part separately. We then combine both parts using a symplectic method of order two.

The outline of this thesis is as follows. First, we describe Hamiltonian systems and the relationship with Lagrange's Equations. We then outline the main properties of Hamiltonian systems. Finally, we give a brief description of Poisson brackets and its properties. In chapter 2, we start by describing one step methods including an error analysis; we also illustrate the stability of such methods. We the describe in detail several popular numerical methods. In chapter 3, we discuss the

main properties of geometric integration. In chapter 4, we apply symplectic and non symplectic methods to three well known problems, namely, the pendulum, Kepler's problem, and Hill's problem.

Chapter 2: Hamiltonian Systems

In this chapter Hamiltonian systems and Poisson brackets will be introduced with their main properties.

2.1 Hamilton's Equations

Given a Lagrangian function $L(q,\dot{q})$, then Lagrange's equations of motion are given by

$$\frac{d}{dt}\nabla_{\dot{q}}L(q,\dot{q}) - \nabla_q L(q,\dot{q}) = 0, \tag{2.1}$$

where $q = (q_1, ..., q_n)^T$ is the position of a mechanical system, with n degrees of freedom. This equation can be reduced to a system of first-order equations, by introducing the conjugate momenta

$$p = \nabla_{\dot{q}}L(q,\dot{q}). \tag{2.2}$$

The relation defines a one to one map between p and q, for fixed q. We see that using equation (2.2), Lagrange's equations (2.1) can be reformulated as

$$\dot{p} = \nabla_q L(q,\dot{q}) \tag{2.3}$$

The Hamiltonian is defined by [7]

$$H(p,q) = p^T\dot{q} - L(q,\dot{q}). \tag{2.4}$$

2.2 Examples of Hamiltonians

Example 2.2.1 (The pendulum)**.** The mathematical pendulum is a system with one degree of freedom having the Hamiltonian

$$H(p,q) = \frac{1}{2}p^2 - \cos q.$$

Example 2.2.2 (Kepler's problem)**.** The Kepler problem considers the movement of a body around a center of gravity. It can be described by the Hamiltonian

$$H(p,q) = \frac{1}{2}\| p \|^2 - \frac{1}{\| q \|}$$

where $q = (q_1, q_2)$ and $p = (p_1, p_2)$.

Example 2.2.3 (Henon-Heiles problem)**.** The polynomial Hamiltonian in two degrees of freedom

$$H(p,q) = \frac{1}{2}(p_1^2 + p_2^2) + \frac{1}{2}(q_1^2 + q_2^2) + q_1^2 q_2 - \frac{1}{3}q_2^3$$

is a Hamiltonian differential equation that can have chaotic solutions.

Example 2.2.4 (The Restricted 3-Body Problem)**.** The restricted 3-body problem is defined by the Hamiltonian:

$$H = \frac{1}{2}\| q\|^2 - p^T K q - U$$

where

$$K = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix},$$

and U is the self-potential

$$U = \frac{\mu}{d_1} + \frac{1-\mu}{d_2}$$

The idea is to consider two bodies with masses μ and $1-\mu$ respectively, where $\mu \in [0, \frac{1}{2}$ with d_i the distance from the infinitesimal body to the i-th primary, or

$$d_1{}^2 = (p_1 - 1 + \mu)^2 + p_2{}^2, \qquad d_2{}^2 = (p_1 + \mu)^2 + p_2{}^2.$$

Definition 2.2.1. A Hamiltonian system is a system of second order differential equations of the form:

$$\dot{q} = H_p(t,q,p) \qquad \dot{p} = -H_q(t,q,p) \tag{2.5}$$

where the Hamiltonian $H = H(t,q,p)$ is a real valued function, the vectors $q = (q_1, ..., q_n)$ and $p = (p_1, ..., p_n)$ are called the position and momentum vectors, respectively, and t is the time. The variables q and p are said to be conjugate variables and the integer n is the number of degrees of freedom of the system.

Here we consider only autonomous Hamiltonian systems, i.e., time invariant systems of the form $H(q,p)$. In general, introduce the $2n$ vector z, the $2n \times 2n$ skew

symmetric matrix J, and the gradient ∇H by

$$z = \begin{bmatrix} q \\ p \end{bmatrix}, \qquad J = J_n = \begin{bmatrix} 0 & I \\ -I & 0 \end{bmatrix}, \qquad \nabla H = \begin{bmatrix} \frac{\partial H}{\partial z_1} \\ \vdots \\ \frac{\partial H}{\partial z_{2n}} \end{bmatrix}$$

where 0 is the $n \times n$ zero matrix and I is the $n \times n$ identity matrix. So the notation of the Hamiltonian system becomes

$$\dot{z} = J\nabla H(t,z). \tag{2.6}$$

An important class of Hamiltonians are those that are separable into kinetic and potential energy

$$H(p,q) = T(p) + V(q). \tag{2.7}$$

In this case the equations of motion take the form

$$\dot{q} = \nabla_p T(p) \qquad \dot{p} = -\nabla_q V(q) \tag{2.8}$$

Theorem 2.2.1. *Lagrange's equations are equivalent to Hamilton's equations [3]*

$$\dot{p}_k = -\frac{\partial H}{\partial q_k}(p,q), \quad \dot{q}_k = \frac{\partial H}{\partial p_k}(p,q), \quad k = 1,...,d.$$

Proof. The definitions for the momenta p and the Hamiltonian H imply that:

$$\frac{\partial H}{\partial p} = \dot{q}^T + p^T \frac{\partial \dot{q}}{\partial p} - \frac{\partial L}{\partial \dot{q}} \frac{\partial \dot{q}}{p} = \dot{q}^T$$

$$\frac{\partial H}{\partial q} = p^T \frac{\partial \dot{q}}{\partial q} - \frac{\partial L}{\partial q} - \frac{\partial L}{\partial \dot{q}} \frac{\partial \dot{q}}{\partial q} = -\frac{\partial L}{\partial q}$$

Therefore the Lagrange equations are equivalent to Hamilton's equations. □

2.3 Main Properties of Hamiltonian Systems

Consider the Hamiltonian system

$$\dot{p} = \nabla_q H(p,q), \qquad \dot{q} = -\nabla_p H(p,q) \tag{2.9}$$

with flow

$$\varphi_t : (p_\circ, q_\circ) \longrightarrow (p(t), q(t)). \tag{2.10}$$

We introduce the notion of a first integral.

Definition 2.3.1. A non-constant function $I(y)$ is a first integral of $\dot{y} = f(y)$ if

$$I'(y)f(y) = 0 \quad \forall\, y.$$

This is equivalent to the property that every solution $y(t)$ of $\dot{y} = f(y)$ satisfies $I(y(t)) = Const$.

Example 2.3.1 (First integral of the Lotka-Volterra Model)**.** For two species with $u(t)$ denoting the number of predators and $v(t)$ the number of prey, the Lotka-Volterra, models the growth of the two species with equations of motion given by

$$\dot{u} = u(v-2) \tag{2.11}$$

$$\dot{v} = v(1-u) \tag{2.12}$$

If we divide by each other and do separation of variables we get:

$$0 = \frac{1-u}{u}\dot{u} - \frac{v-2}{v}\dot{v} = \frac{d}{dt}I(u,v). \tag{2.13}$$

with the first integral

$$I(u,v) = \ln u - u + 2\ln v - v \tag{2.14}$$

Example 2.3.2 (First integral for the pendulum)**.** The equations of motion for the mathematical pendulum are given by

$$\dot{p} = -\sin q, \qquad \dot{q} = p$$

Now let

$$\frac{dH}{dt} = p\dot{p} + \sin q\dot{q} \tag{2.15}$$

substituting into the equations of motion we obtain

$$\frac{dH}{dt} = p(-\sin q) + (\sin q)p = 0$$

which shows that H is a first integral.

Definition 2.3.2 (Integrable Systems)**.** A Hamiltonian system is said to be completely integrable, if it has n first integrals (including the Hamiltonian itself), where n is the number of degrees of freedom. In mechanical systems, the first integrals are often familiar quantities.

Some of the important properties of the Hamiltonian $H(q,p)$ are listed below.

1. Conservation of the total energy: The Hamiltonian $H(p,q)$ is constant along the solutions of (2.9) and $H(p,q)$ is a first integral.

   Example 2.3.3. We show that the Hamiltonian is a constant of the motion:

$$\begin{aligned}\dot{H} &= \sum_{i=1}^{n} \frac{\partial H}{\partial q_i}\dot{q}_i + \sum_{i=1}^{n} \frac{\partial H}{\partial p_i}\dot{p}_i \\ &= \sum_{i=1}^{n} \frac{\partial H}{\partial q_i}\frac{\partial H}{\partial p_i} + \sum_{i=1}^{n} \frac{\partial H}{\partial p_i}(-\frac{\partial H}{\partial q_i}) = 0\end{aligned}$$

2. Conservation of the total linear and angular momentum.

   Example 2.3.4. We consider a system of of N particles interacting pairwise with potential forces depending on the distances of the particles. This is a Hamiltonian system with total energy

$$H(p,q) = \frac{1}{2}\sum_{i=1}^{N} \frac{1}{m_i} p_i^T p_i + \sum_{i=2}^{N}\sum_{j=1}^{i-1} V_{ij}(\parallel q_i - q_j \parallel).$$

   Here $q_i, p_i \in R^3$ represents the position and momentum of the ith particles of mass m_i, and $V_{ij}(r), (i > j)$ is the interacting potential between the ith and jth particle.

   The equations of motion are given by

$$\dot{q}_i = \frac{1}{m_i} p_i, \qquad \dot{p}_i = \sum_{j=1}^{N} v_{ij}(q_i - q_j)$$

   where for $i > j$, we have $v_{ij} = v_{ji} = -V_{ij}'(r_{ij})/r_{ij}$ with $r_{ij} = \parallel q_i - q_j \parallel$. The

conservation of the total linear and angular momentum

$$P = \sum_{i=1}^{N} p_i, \qquad L = \sum_{i=1}^{N} q_i \times p_i$$

is a consequence of the symmetry relation $v_{ij} = v_{ji}$. We can see that

$$\frac{d}{dt} \sum_{i=1}^{N} p_i = \sum_{i=1}^{N} \sum_{j=1}^{N} v_{ij}(q_i - q_j) = 0$$

$$\frac{d}{dt} \sum_{i=1}^{N} q_i \times p_i = \sum_{i=1}^{N} \frac{1}{m_i} p_i \times p_i + \sum_{i=1}^{N} \sum_{j=1}^{N} q_i \times v_{ij}(q_i - q_j) = 0$$

3. The flow φ_t (2.10) is a symplectic map, i.e.

$$\varphi_t'(y)^T J \varphi_t'(y) = J \quad \text{with} \quad J = \begin{pmatrix} 0 & I \\ -I & 0 \end{pmatrix}. \tag{2.16}$$

We discuss symplecticity in detail later.

2.4 Poisson Brackets

Consider Hamilton's equations

$$\dot{p} = \frac{\partial H}{\partial q} \tag{2.17}$$

$$\dot{q} = -\frac{\partial H}{\partial p}. \tag{2.18}$$

We can rewrite these equations in a simple form by using the Poisson brackets. So we need to define two functions f, g depending on the same variables (p,q). By definition of the Poisson brackets we have

$$\{f(p,q), g(p,q)\} = \frac{\partial f}{\partial p}\frac{\partial g}{\partial q} - \frac{\partial f}{\partial q}\frac{\partial g}{\partial p}$$

So Hamilton's equations become:

$$\dot{p} = \{q, H(p,q)\} \tag{2.19}$$

$$\dot{q} = \{p, H(p,q)\} \tag{2.20}$$

From above the equation appear symmetrical. So for any function p, q

$$\frac{d}{dt} f(p,q) = \frac{\partial f}{\partial p}\dot{p} + \frac{\partial f}{\partial q}\dot{q} = \frac{\partial f}{\partial p}\frac{\partial H}{\partial q} + \frac{\partial f}{\partial q}(-\frac{\partial H}{\partial p}) = \{f, H\}.$$

We can write it also with respect to time as:

$$\frac{d}{dt} f(p,q,t) = \{f, H\} + \frac{\partial f}{\partial t}$$

2.4.1 Properties of Poisson Brackets

The main properties are listed below.

1. $\{f,g\} = -\{g,f\}$ which implies that $\{f,f\} = 0$.
2. $\{f, g+h\} = \{f,g\} + \{f,h\}, \qquad \{f+g,h\} = \{f,h\} + \{g,h\}.$
3. From the chain rule for partial derivatives we can see that:

$$\{f, gh\} = g\{f,h\} + \{f,g\}h$$
$$\{fg, h\} = \{f,h\} + \{g,h\}.$$

4. An important property:

$$\{f, \{g,h\}\} + \{h, \{f,g\}\} + \{g, \{h,f\}\} = 0$$

Chapter 3: Numerical Methods

In any numerical study, we should first examine any geometric or structural properties of the differential equation or its flow. Then we design numerical methods which also have these structural properties and examine the consequences. All of which should encourage us to confront questions of phase space and degrees of freedom and think about the significance of local, global, and qualitative errors, and finally think about the tools and functions allowed in numerical analysis.

We now review some basic facts of the theory of numerical methods for the integration of systems of differential equations. We restrict our attention to *one-step* methods. General references on numerical integrators can be found in [3, 4, 11]. Here we are mainly interested in Hamiltonian problems though we point out that the methods we present can be applied to general systems of differential equations.

A numerical method when applied to an ordinary differential equation, approximates the solution of the problem, Consider the ODE:

$$y' = f(t,y), \quad t \geq t_0, \quad y(t_0) = y_0. \tag{3.1}$$

We assume that f is sufficiently well behaved and $y_0 \in \mathbb{R}^n$ is a given vector. We require that f obeys the Lipschitz condition

$$\|f(t,x) - f(t,y)\| \leq \lambda \|x - y\|, \ \forall x, y \in \mathbb{R}^n, \ t \geq t_0. \tag{3.2}$$

The real constant $\lambda > 0$ is independent of the choice of x and y and is known as the Lipschitz constant. If the Lipschitz condition is verified, then the system of ordinary differential equations has a unique solution.

3.1 One Step Methods

Given a discrete trajectory up to time t_n, $y_0, y_1, \ldots, y_n$, there are many ways of constructing the next approximation y_{n+1}. One common approach is to construct the next approximation from the l previous points $y_{n-l+1}, \ldots, y_n$ and then find the derivatives at these points $\dot{y}_{n-l+1}, \ldots, \dot{y}_n$, where $\dot{y}_i = f(y_i)$.
A linear multistep method is a linear recurrence relation,

$$\sum_{i=n-l+1}^{n+1} a_i y_i + \sum_{i=n-l+1}^{n+1} b_i \dot{y}_i = 0. \tag{3.3}$$

We will primarily be concerned with one-step method

$$y_{n+1} = \psi_h(y_n) \tag{3.4}$$

where ψ_h is generally non-linear and will depend on f and its derivatives.
Because one step method generate a mapping of phase space, there is a natural correspondence with the flow map [7].

3.1.1 Derivation of One Step Methods

One step method can be derived in many ways. One way is to apply the fundamental theorem of calculus by integrating both sides on a small interval $[t, t+h]$ obtaining

$$y(t+h) - y(t) = \int f(y(t+\tau))d\tau.$$

The right-hand side can then replaced by a suitable quadrature formula resulting in an approximation of the form

$$y(t+h) \approx y(t) + \sum b_i f(y(t+\tau_i))$$

with quadrature weights $\{b_i\}$ and quadrature points $\{\tau_i\}$ [7].

3.1.2 Error Analysis

Error analysis for a one-step method is concerned with:

- Local error: which is a comparison of the Taylor series with the true solutions and the numerical approximation.
- Global error: it is the accumulated error during the computation of a trajectory.

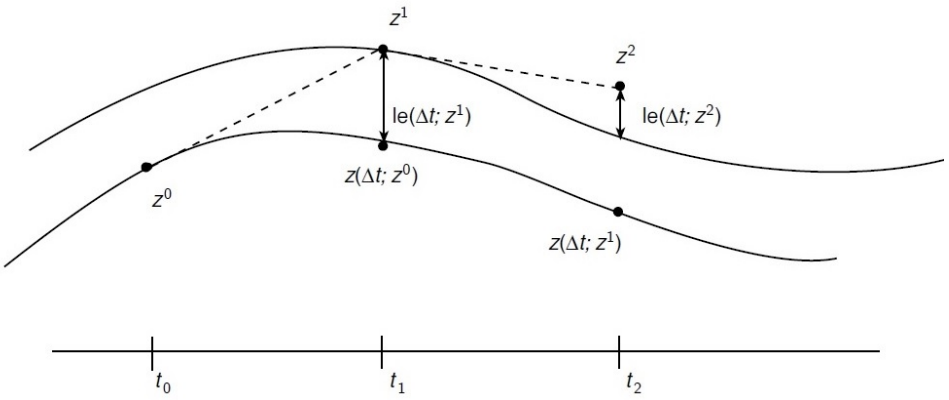


Figure 3.1: Local errors in numerical integration [7].

The term local error is used for the difference between the exact and approximate solution for a time-step h, starting at a point $\bar{z}$. Then by using Taylor expansion, we obtain:

$$\bar{\psi}_h(\bar{z}) - \phi_h(\bar{z}) = le(h;\bar{z}) = c_{p+1}(\bar{z})h^{p+1} + c_{p+2}(\bar{z})h^{p+2} + \dots \tag{3.5}$$

where p is termed the order of the method.

3.1.3 Convergence of One-Step Methods

Lemma 3.1.1 (Discrete Gronwall Lemma)**.** *Let a positive sequence* $\{y\}_{n=0}^{N}$ *satisfy*

$$y_{n+1} \leq a\, y_n + b$$

for constants $a, b > 0$. Then

$$y_n \leq \frac{b}{a-1}(a^n - 1) + a^n y_0 \ (\textit{for } a \neq 1) \textit{ and } y_n \leq nb + y_0 \ (\textit{for } a = 1)$$

3.1.4 Convergence of generalized One-Step Methods

Let the local error be given by: $le(h;\bar{y}) = \bar{\psi}_h(\bar{z}) - \phi_h(\bar{y})$, where $\bar{\psi}_h(\bar{z})$ is the numerical method, and $\phi_h(\bar{y})$ is the exact solution.
In general

$$||le(h;\bar{y})|| \leq c(y)h^{p+1}.$$

We further suppose that

$$||\bar{\psi}_h(u) - \bar{\psi}_h(v)|| \leq (1 + h\hat{L})||u - v||$$

where in general $\hat{L}$ is not the same as the Lipschitz constant for the vector field. Then:

$$\begin{aligned} \varepsilon_{n+1} &= y_{n+1} - y(t_{n+1}) \\ &= \bar{\psi}_h(y_n) - \phi_h(y(t_n)). \end{aligned} \tag{3.6}$$

We add and subtract $\bar{\psi}_h(y(t_n))$, which is the numerical solution at t_{n+1} obtained from the exact solution at time t_n.

$$\begin{aligned} ||\varepsilon_{n+1}|| &= ||\bar{\psi}_h(y_n) - \bar{\psi}_h(y(t_n)) + \bar{\psi}_h(y(t_n)) - \phi_h(y(t_n))|| \\ &\leq ||\bar{\psi}_h(y_n) - \bar{\psi}_h(z(t_n))|| + ||\bar{\psi}_h y(t_n)) - \phi_h(y(t_n))|| \\ &\leq (1 + h\hat{L})||y_n - y_{(}t_n)|| + c(y(t_n))h^{p+1} \\ &= (1 + h\hat{L})||e_n|| + c(y(t_n))h^{p+1} \\ &\leq (1 + h\hat{L})||\varepsilon_n|| + \bar{c}h^{p+1} \end{aligned} \tag{3.7}$$

Then by the Discrete Gronwall lemma, we can find the global error by:

$$\begin{aligned} ||\varepsilon_n|| &\leq \frac{\bar{c}h^{P+1}}{k-1}(k^n-1)+k^n||\varepsilon_o|| \\ &\leq h^p\frac{\bar{c}}{L}(\varepsilon^{T\hat{L}}-1) \end{aligned} \tag{3.8}$$

which implies that the global error satisfies $\max_{n=0,\ldots,N}||\varepsilon_n|| = O(h^p)$

3.2 Stability of Numerical Methods

For a linear system of ODEs, $\frac{dy}{dt} = Ay$, where A is a $n \times n$ matrix with a basis of eigenvectors, the general solution is given by

$$y(t) = \sum_{i=1}^{n} c_i e^{\lambda_i t}\xi_i$$

where λ_i's are the eigenvalues, and ξ_i's the corresponding eigenvectors. The stability is then determined by the eigenvalues. For example, if all the eigenvalues have negative real part, the origin is asymptotically stable. If the eigenvalues are in the left-half plane, the origin is Lyapunov stable.

- *A*-stable: a method is *A*-stable if the stability region includes the entire left plane. An *A*-stable method has the property that the origin is stable regardless of the stepsize.
- *L*-stable: *L*-stability is concerned with the asymptotic behaviour as $\mu \to \infty$.

A method is *L*-Stable if:

- it is *A*-stable
- $R(\mu) \to 0$ as $\mu \to \infty$.

3.3 Stiff Differential Equations

In a stiff differential equation the solution components evolve on very different timescales. This causes a problem as a numerical method, possibly chooses a step size for the most rapidly evolving component, even if its contribution to the solution is negligable. This leads to very small step sizes, highly inefficient computations and long waits for the user. The reason for this is an instability in the method, where a small error may grow rapidly with each step.

3.4 Explicit Euler Method

The most basic explicit method of integration known as Euler's method is given by

$$y_{n+1} = y_n + \Delta t f(y_n).$$

The quadrature rule used is just:

$$\int_0^{\Delta t} f(y(t+\tau))d\tau = \Delta t f(y(t)) + O(\Delta t^2)$$

This method is easy to use and each step is fast as no equations need to be evaluated and there is only one function evaluation per step. The Euler method is still used when f is hard to evaluate and there are a large number of simultaneous equations. Problems of this kind arise, for example, in weather forecasting. The main problem with this method is that there are often severe restrictions on the size of $h = \Delta t$.

Local Error

For the explicit Euler method, the error after one step of the method starting from the exact solution, is called the local error, is given as

$$e_{n+1} = (y(t_n) + hf(t_n, y(t_n))) - y(t_n + h).$$

By estimating the remainder term in the Taylor expansion of $y(t_n + h)$ at t_n, we can bound e_{n+1} by

$$||e_{n+1}|| \leq Ch^2 \; with \; C = \frac{1}{2} \max_{t_0 \leq t \leq T} ||y''(t)||$$

provided that the solution is twice continuously differentiable, which is the case if f is continuously differentiable.

3.4.1 Global Error

The global error will be the summation of local errors, and is given by:

$$|\sum_{i=1}^{n} e_{n+1}| \leq \sum_{i=1}^{n} \frac{h^2}{2} y''(t_i) = \frac{h}{2} y''(t), \; t_0 \leq t \leq 1$$

We note that the overall error is proportional to h, and to $\max |y''|$. We say that this is an order $O(h)$ or a first order method.

There are therefore two restrictions on h, it must be small both for accuracy at each stage and for stability to stop the errors growing. Stiffness arises when the restriction on h for stability is much more severe than the restriction for accuracy.

3.4.2 Stability and Instability

A numerical method to solve a differential equation is unstable if the errors it makes grow more rapidly than the underlying solution. If the errors decay then it is stable.

To check the Stability for Explicit Euler method if $h > 0$ and $\lambda = p + iq$ then $|1 + h\lambda| < 1$ implies that

$$(1 + hp)^2 + (hq)^2 < 1$$

$$2hp + h^2 p^2 + h^2 q^2 < 0$$

$$2p + hp^2 + hq^2 < 0$$

so that the method is stable if (p,q) lies in the circle of radius $\frac{1}{h}$ as shown in Figure 1. In this figure the shaded region shows the values of p and q for which the numerical method is stable. Recall that the original differential equation is stable provided that p lies in the half-plane $p < 0$. The shaded region only occupies a fraction of this half-plane, although the size of the shaded region increases as $h \to 0$. Thus, for a fixed value of h the numerical method will only have errors which do not grow if the eigenvalues of A are severely constrained

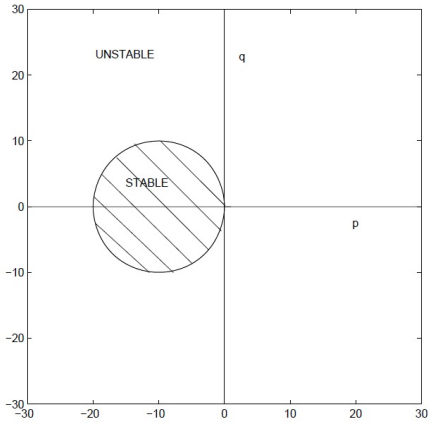


Figure 3.2: Stability region for the Explicit Euler method

3.5 Implicit Euler Method

For non-stiff equations we can use implicit Euler method with fixed point iteration.

$$y_{n+1} = y_n + h_n f(y_{n+1})$$

In the Implicit Euler method, y_{n+1} is not given as an explicit expression. Instead, y_{n+1} is given as the solution of an equation. If function F(y, t) is non-linear in y, the equation is a non-linear equation. But in case of a stiff differential equation for which certain numerical methods for solving the equation are numerically unstable, unless the stepsize is taken to be extremely small, we can use implicit Euler with Newton iteration:

$$y_{n+1} = y_n + hf(y_{n+1}) \tag{3.9}$$

$$y_{n+1} - y_n - hf(y_{n+1}) = 0 \tag{3.10}$$

$$y_{n+1}^{k+1} = y_{n+1}^{k} + \delta y_{n+1} \tag{3.11}$$

where k is the iteration counter. These iterations are performed at every integration step.

Now to see when should we use fixed point iteration or Newton iteration we should check the contractivity, i.e. if we take the linear equation $\dot{y} = \lambda y$ then the contractivity will be $|\varphi^{'}(u)| = |h\lambda| < 1$.

The Implicit Euler method has the same order of error of the Explicit Euler method. i.e. the global error is proportional to h. If we now apply this to the equation $\dot{y} = Ay$ we have

$$y_{n+1} = y_n + hAy_{n+1} \tag{3.12}$$

This is a linear system which we need to invert to give:

$$y_{n+1} = (I - hA)^{-1} y_n \tag{3.13}$$

Now if the eigenvalues of A are λ_j, those of $(I - hA)^{-1}$ are $(1 - h\lambda_j)^{-1}$ with the same eigenvectors.

Thus the contribution to y_n in the direction of ϕ_j involves:

$$|1 - h\lambda_j|^{1-n} \tag{3.14}$$

Now let $\lambda_j = p + iq$, it follows that

$$|1 - h\lambda_j|^{-2} = \frac{1}{(1-hp)^2 + (hq)^2} \tag{3.15}$$

Therefore the error decay and the method is stable if

$$\frac{1}{(1-hp)^2+(hq)^2} < 1$$

If

$$1 < (1-hp)^2+(hq)^2$$

or

$$0 < h(p^2+q^2)-2p$$

the stability region is shaded in Figure 1. This picture is in complete contrast to the one that we obtained for the Explicit Euler method. The stability region is now very large and certainly includes the half-plane $p < 0$. Thus any errors in the numerical method will be rapidly damped out. Unfortunately the numerical solution can decay even if $p \geq 0$, so that neutral or growing terms in the underlying solution can be damped out as well. This is a source of (potential) long term error, especially in Hamiltonian problems [7]. Implicit Euler method has order one this means the local truncation error

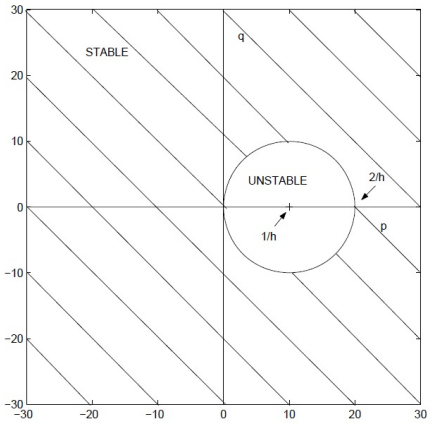


Figure 3.3: Stability region for the Implicit Euler method

(defined as the error made in one step) is $o(h^2)$ the error at a specific time t is $O(h)$

Global error $\rightarrow E_N(h) = O(h)$

In general, for a first order method we reduce the time step h by a factor of 2 the global error $E_N(h) = O(h)$ decreases only by a factor of 2 we like to have higher order method.

3.6 Symplectic Euler Method

$$p_{n+1} = p_n - hH_q(p_{n+1}, q_n) \qquad \text{or} \qquad p_{n+1} = p_n - hH_q(p_n, q_{n+1})$$

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

are symplectic methods of order 1. The method is stable and can be used for non-separable systems [3]. This method will be discussed in detail later.

3.7 Trapezoidal Rule

$$y_{n+1} = y_n + h[f(y_n) + f(y_{n+1})]$$

The trapezoidal rule is based on :

$$\int_0^h f(y(t+\tau))d\tau = \frac{1}{2}h[f(y(t)) + f(y(t+h))] + O(h^3)$$

The trapezoidal method is an implicit method. This method is also symmetric, i.e. if you know y_n and you want to find y_{n+1} with step-size h then this is the same method for finding y_n given y_{n+1} with step-size $-h$, This property is important for finding approximations to the solutions equations such as $\ddot{y} + y = 0$ which are the same both forwards and backwards in time.

The Local Truncation Error:

$$\varepsilon_h = O(h^3)$$

The Global Error $E_N(h) = O(h^2)$ is of second order.

3.8 Implicit Midpoint Method

The implicit midpoint rule is a symmetric Runge-Kutta method closely

related to the trapezium rule. It is given by

$$y_{n+1} = y_n + hf\left(\frac{y_n + y_{n+1}}{2}\right).$$

The quadrature rule is defined as:

$$\int_0^h f(y(t+\tau))d\tau = h\left[f(\frac{y(t)+f(y(t+h)}{2})\right] + O(h^3)$$

When applied to the linear ODE $\dot{y} = Ay$ the method gives exactly the same sequence of iterates as the trapezium rule. Thus its stability properties are identical to the Trapezium Rule and hence are optimal. Like the Trapezium Rule the global error of the Implicit Midpoint Rule varies as h^2 and a function solve is required to find y_{n+1}.
The Implicit Midpoint rule is the simplest example of a sequence of implicit Runge-Kutta methods called Gauss-Legendre methods. The solution by using this method will be very accurate for long times with excellent stability, but regardless of cost. Implicit midpoint is a symplectic method of order 2.

3.9 Störmer-Verlet Method

The Störmer-Verlet Method is explicit for separable Hamiltonian problems. The method is given by

$$\begin{aligned} p_{n+1/2} &= p_n - \frac{h}{2}H_q(p_{n+1/2}, q_n) \\ q_{n+1} &= q_n + \frac{h}{2}(H_p(p_{n+1/2}, q_n) + H_p(p_{n+1/2}, q_{n+1})) \\ p_{n+1} &= p_{n+1/2} - \frac{h}{2}H_q(p_{n+1/2}, q_{n+1}) \end{aligned} \tag{3.16}$$

and also it can be written as [3]:

$$\begin{aligned} q_{n+1/2} &= q_n + \frac{h}{2} H_q(p_n, q_{n+1/2}) \\ p_{n+1} &= p_n + \frac{h}{2}(H_p(p_n, q_{n+1/2}) + H_p(p_{n+1}, q_{n+1/2})) \\ q_{n+1} &= q_{n+1/2} - \frac{h}{2} H_q(p_{n+1}, q_{n+1/2}) \end{aligned} \tag{3.17}$$

We will discuss this method in detail later.

3.10 Runge-Kutta Methods

Definition 3.10.1. Let a_{ij}, b_i $(i, j = 1, ..., s)$ be real numbers and let $c_i = \sum_{j=1}^{s} a_{ij}$. An s-stage Runge-Kutta method is given by

$$k_i = f(t_o + c_i h, y_o + h \sum_{j=1}^{s} a_{ij} k_j), i = 1, ..., s$$

$$y_1 = y_0 + h \sum_{i=1}^{s} b_i k_i.$$

In this case we will have a full matrix with a non zero coefficients, so the slopes k_i can no longer be computed explicitly and even sometimes do not exist [3].

In a Butcher table the coefficients are usually displayed as follows: The

Table 3.1: Runge-Kutta method

c_1	a_{11}	.	.	.	a_{1s}
.	.				.
.	.				.
.	.				.
c_s	a_{s1}	.	.	.	a_{ss}
	b_1	.	.	.	b_s

number of stages s and the constant coefficients $\{b_i\}$, $\{a_{ij}\}$ completely characterize a Runge-Kutta method. In general, such a method is implicit and leads to a nonlinear system in the s internal stage variables y_n.

Definition 3.10.2. A Runge-Kutta method (or a general one-step method) has order p, if for all sufficiently regular problems $\dot{y} = f(t,y)$, $y(t_o) = y_o$, the local error $y_1 - y(t_o + h)$ satisfies

$$y_1 - y(t_o + h) = O(h^{p+1}) \text{ as } h \to 0.$$

To check the order of a Runge-Kutta method one has to compute the Taylor series expansions of $y(t_o + h)$ and y_1 around $h = 0$.
The algebraic conditions for the coefficients for orders 1,2 and 3:

1. $\sum_i b_i = 1$ for order 1
2. $\sum_i b_i c_i = \frac{1}{2}$ and $\sum_i b_i c_i^2 = \frac{1}{3}$ for order 2
3. $\sum_{i,j} b_i a_{ij} c_j = \frac{1}{6}$ for order 3.

The Runge-Kutta method of order 4 is given by the following equations:

$$y_{n+1} = y_n + \frac{1}{6}h(k_1 + 2k_2 + 2k_3 + k_4)$$

$$t_{n+1} = t_n + h$$

where the y_{n+1} is the approximation of $y(t_{n+1})$ and

$$\begin{aligned} k_1 &= f(t_n, y_n) \\ k_2 &= f(t_n + \frac{1}{2}h, y_n + \frac{1}{2}hk_1) \\ k_3 &= f(t_n + \frac{1}{2}h, y_n + \frac{1}{2}hk_2) \\ k_4 &= f(t_n + \frac{1}{2}h, y_n + \frac{1}{2}hk_3) \end{aligned} \tag{3.18}$$

It can be shown that there is a value C which depends on f in a complex

Table 3.2: Runge-Kutta method of order 4

$$
\begin{array}{c|cccc}
0 & 0 & 0 & 0 & 0 \\
\frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 \\
\frac{1}{2} & 0 & \frac{1}{2} & 0 & 0 \\
1 & 0 & 0 & 1 & 0 \\
\hline
 & \frac{1}{6} & \frac{1}{3} & \frac{1}{3} & \frac{1}{6}
\end{array}
$$

way such that a local truncation error $E = |y(t+h) - y_{n+1}|$ is bounded by:

$$E \leq Ch^5.$$

Over a large number of steps these errors accumulate as before to give a global error ε of the form:

$$\varepsilon \sim (e^{LT} - 1)h^4.$$

The error is proportional to h^4; hence the name of an order 4 method. The error for a given h is much smaller than for the explicit Euler method.
Runge-Kutta is very famous and widely favoured because:

- It is a one step method.
- It is easy to start and code.
- It is easy to use and no equation needs to be solved at each stage.
- It is highly accurate for moderate h values.
- The method is stable.

However, this method has some disadvantages. When the function is hard to evaluate it will be difficult to evaluate it 4 times each iteration, errors accumulate rapidly as t increases and for some kinds of problems it can not be used unless h is very small.

3.11 Partitioned Runge-Kutta Methods

We consider differential equations in the partitioned form:

$$\dot{y} = f(y,z), \qquad \dot{z} = g(y,z)$$

where y and z may be vectors of different dimensions.

The idea is to take two different Runge-Kutta methods, and to treat the y-variables with the first method (a_{ij}, b_i), and the z-variables with the second method $(\hat{a}_{ij}, \hat{b}_i)$.

Definition 3.11.1. Let b_i, a_{ij} and $\hat{b}_i, \hat{a}_{ij}$ be the coefficients of two Runge-Kutta methods. A partitioned Runge-Kutta method for the solution of the above equation is given by [3]:

$$k_i = f(y_o + h\sum_{j=1}^{s} a_{ij}k_i, z_o + h\sum_{j=1}^{s} \hat{a}_{ij}l_j)$$

$$l_i = g(y_o + h\sum_{j=1}^{s} a_{ij}k_i, z_o + h\sum_{j=1}^{s} \hat{a}_{ij}l_j)$$

$$y_1 = y_o + h\sum_{i=1}^{s} b_i k_i, \qquad z_1 = z_o + h\sum_{i=1}^{s} \hat{b}_i l_i$$

Methods of this type have originally been proposed by Hofer (1976) and Griepentrog (1978). Their importance for Hamiltonian systems has been discovered only very recently.

An interesting example is the symplectic Euler method, where the implicit Euler method $b_1 = 1$, $a_{11} = 1$ is combined with the explicit Euler method $\hat{b}_1 = 1$, $\hat{a}_{11} = 0$. The Stormer-Verlet method can be written in the previous form by the coefficients given in Table(3).

Table 3.3: Stormer-Verlet as a partitioned Runge-Kutta method

0	0	0
1	$\frac{1}{2}$	$\frac{1}{2}$
	$\frac{1}{2}$	$\frac{1}{2}$

$\frac{1}{2}$	$\frac{1}{2}$	0
$\frac{1}{2}$	$\frac{1}{2}$	0
	$\frac{1}{2}$	$\frac{1}{2}$

3.12 Splitting and Composition Methods

These methods can exploit the natural decomposition of Hamiltonian systems and have been used with great success in studies of the solar system and of molecular dynamics (see for example [10]). The main idea behind splitting methods is to decompose the discrete flow ψ_h as a composition of simpler flows:

$$\psi_h = \psi_{1,h} \circ \psi_{2,h} \circ \psi_{3,h} \dots$$

where each of the sub-flows is chosen such that each represent a simpler integration of the original. A geometrical perspective on this approach is to find useful geometric properties of each preserved under combination, symplecticity is just such a property, but we often seek to preserve reversibility and other structures. Suppose that a differential equation takes the form:

$$\frac{du}{dt} = f = f_1 + f_2$$

Here the functions f_1 and f_2 may will represent different physical processes in which case there is a natural decomposition of the problem (say into terms to kinetic and potential energy).

The most direct form of splitting methods decompose the equation into two problems:

$$\frac{du_1}{dt} = f_1 \qquad \text{and} \qquad \frac{du_2}{dt} = f_2$$

chosen such that these two problems can be integrated in closed form to give explicity computable flows $\psi_1(t)$ and $\psi_2(t)$. We denote $\psi_{i,h}$ the result of applying the corresponding continuous flows $\psi_i(t)$ over a time t. A simple (first order) splitting is then

given by the Lie-Trotter formula

$$\psi_n = \psi_{1,h} \circ \psi_{2,h}$$

Suppose the original problem has Hamiltonian $H = H_1 + H_2$ then we can say the composition of two problems with respect to Hamiltonian H_1 and H_2. The differential equation corresponding to each Hamiltonian leads to an evolutionary map $\psi_i(t)$ of the form described above.

The Lie-Trotter splitting introduces local errors proportional to h^2 at each step and more accurate decomposition is the Strang splitting given by

$$\psi_n = \psi_{1,\frac{h}{2}} \circ \psi_{2,h} \circ \psi_{1,\frac{h}{2}}$$

This splitting method has a local error proportional to h^3.

Example 3.12.1. Suppose that a Hamiltonian system has a Hamiltonian which can be expressed as a combination of kinetic energy and a potential energy term as follows:

$$H(u) = H_1(u) + H_2(u) = T(p) + T(p)$$

$$\frac{dp}{dt} = -\frac{\partial H_2}{\partial q}, \quad \frac{dq}{dt} = \frac{\partial H_1}{\partial p}$$

This splitting of H is usually referred to as a separable or P-Q splitting, we immediately have that

$$\psi_{1,h} = I - hH_{2,q} \quad and \quad \psi_{2,h} = I + hH_{1,p}$$

Where I represents the identity mapping. Applying the Lie-Trotter formula directly to

the splitting gives the symplectic Euler method

$$p_{n+1} = p_n - hH_{2,q}(q_n) \quad and \quad q_{n+1} = q_n + hH_{1,p}(p_{n+1}).$$

3.13 Backward Error Analysis

Consider an ordinary differential equation

$$\dot{y} = f(y)$$

and a numerical method $\phi_h(y)$ which produces the approximations $y_o, y_1, y_2, \ldots$
A forward error analysis consists of the study of the errors $y_1 - \varphi_h(y_o)$ (local error) and $y_n - \varphi_{nh}(y_o)$ (Global error) in the solution space. The idea of the backward analysis is to search for a modified differential equation $\dot{\tilde{y}} = f_h(\tilde{y})$ of the form

$$\dot{\tilde{y}} = f(\tilde{y}) + hf_2(\tilde{y}) + h^2 f_3(\tilde{y}) + \ldots \tag{3.19}$$

such that $y_n = \tilde{y}(nh)$ considering the difference of the vector field $f(y)$ and $f_h(y)$. This will give a good indication of the behaviour of the numerical solution and the global error with the full awareness of the converges issues.
For computation of the modified equation $\dot{y} = f(y)$ we put $y := \tilde{y}(t)$ for a fixed t and we expand the solution into Taylor series

$$\tilde{y}(t+h) = y + h(f(y) + hf_2(y) + h^2 f_3(y) + ...) + \frac{h^2}{2!}(f'(y) + hf_2'(y) + \ldots)(f(y) + hf_2(y) + \ldots) + \ldots \tag{3.20}$$

Then we assume that the numerical method $\phi_h(y)$ can be expanded as

$$\phi_h(y) = y + hf(y) + h^2 j_2(y) + h^3 j_3(y) + \ldots \tag{3.21}$$

where the functions $j_i(y)$ are composed of $f(y)$ and its derivatives. For example, the explicit Euler method gives $j_i(y) = 0$ for all $i \geq 2$. In order to get $\tilde{y}(nh) = y_n$ for all n, we must have $\tilde{y}(t+h) = \phi_h(y)$. By comparing like powers of h in expressions (3.20) and (3.21) will give us a recurrence relation for the functions $f_i(y)$

$$\begin{aligned} f_2(y) &= j_2(y) - \frac{1}{2!} f' f(y) \\ f_3(y) &= j_3(y) - \frac{1}{3!}(f''(f,f)(y) + f' f' f(y)) - \frac{1}{2!}(f' f_2(y) + f_2'(y) + f(y)) \end{aligned} \tag{3.22}$$

Theorem 3.13.1 ([3]). *Suppose that the method $y_{n+1} = \phi_h(y_n)$ is of order p*

$$\phi_h(y) = \varphi_h(y) + h^{p+1}\delta_{p+1}(y) + O(h^{p+2})$$

where $\varphi_t(y)$ denote the exact flow of of $\dot{y} = f(y)$ and $h^{p+1}\delta_{p+1}(y)$ the leading term of the local truncation error. The modified equation then satisfies

$$\dot{\tilde{y}} = f(\tilde{y}) + h^p f_{p+1}(\tilde{y}) + h^{p+1} f_{p+2}(\tilde{y}) + \ldots, \quad \tilde{y}(0) = y_o$$

with $f_{p+1}(y) = \delta_{p+1}(y)$.

Proof. The construction of the functions f_i shows that $f_i(y) = 0$ for $2 \leq i \leq p$ if and only if $\phi_h(y) - \varphi_h(y) = O(h^{p+1})$. A first application of the modified equation (1) is the existence of an asymptotic of the global error .Indeed, by the nonlinear variation of constants formula, the difference between its solution $\tilde{y}(t)$ and the solution $y(t)$ of $\dot{y} = f(y)$ satisfies

$$\tilde{y}(t) - y(t) = h^p e_p(t) + h^{p+1} e_{p+1}(t) + \ldots.$$

Since $y_n = \tilde{y}(nh) + O(h^N)$ for the solution of a truncated modified equation, this proves the existence of an asymptotic expansion in powers of h for the global error $y_n - y(nh)$.

□

Example 3.13.1. Consider the Lotka-Volterra equations

$$\dot{q} = q(p-1), \qquad \dot{p} = p(2-q). \tag{3.23}$$

Now we apply explicit Euler method, and the symplectic Euler method,both with constant step size $h = 0.1$. The first term of the modified equations are

a.

$$\dot{q} = q(p-1) - \frac{h}{2}q(p^2 - pq + 1) + O(h^2)$$

$$\dot{p} = -p(q-2) - \frac{h}{2}p(q^2 - pq - 3q + 4) + O(h^2)$$

b.

$$\dot{q} = q(p-1) - \frac{h}{2}q(p^2 + pq - 4p + 1) + O(h^2)$$

$$\dot{p} = -p(q-2) - \frac{h}{2}p(q^2 - pq - 5q + 4) + O(h^2)$$

Chapter 4: Geometric Integration

Geometric integration is the numerical integration of a differential equation with the goal of preserving certain geometric properties; Thee geometric properties to be preserved are, usually, preservation of energy, momentum, angular momentum, phase space volume, symmetries, time-reversal symmetry, symplectic structure and dissipation. To see this, consider the initial value problem

$$\dot{y} = f(y), \qquad y(0) = y_0 \tag{4.1}$$

The aim is to compute the solution of (4.1) as efficiently as possible. The type of method used, its order, local error and choice of time steps are all tailored to this end [11] By exploiting the structure of the problem, the application of a geometric integrator would allow us to fix a larger than normal time step and compute very long orbits without compromising the emerging phase portrait. Here we concentrate on Hamiltonian systems and on methods that preserve their symplectic structure, first integrals, symmetries, or phase-space volume. Note that no method can preserve both energy and symplecticity at the same time (in general).

Theorem 4.0.2 (Conservation of Linear Invariants)**.** *All explicit and implicit Runge-Kutta methods conserve linear invariants. Partitioned Runge-Kutta method conserve linear invariants if $b_i = \hat{b}_i$ for all i, or if the invariants depends only on p or only on q [3].*

Proof. Let $I(y) = d^T y$ with a constant vector d, so that $d^T f(y) = 0$ for all y. In the

case of Runge-Kutta method we thus have $d^T k_i = 0$, and consequently

$$d^T y_1 = d^T y_o + h d^T (\sum_{i=1}^{s} b_i k_i) = d^T y_o.$$

The statement for partitioned methods is proved similarly. □

Now we will consider differential equations of the form

$$\dot{Y} = A(Y)Y \tag{4.2}$$

where Y can be a vector or a matrix.

Theorem 4.0.3. *If $A(Y)$ is skew-symmetric for all Y (i.e. $A^T = -A$), then the quadratic function $I(Y) = Y^T Y$ is an invariant. In particular if the initial value Y_0 consists of orthonormal columns ($i.e. Y_0^T Y_0 = I$), then the columns of the solution Y(t) of equation (4.2) remains orthonormal for all t.*

Proof. The derivative of $I(Y)$ is $I'(Y)H = Y^T H + H^T Y$. Thus, we have

$$I'(Y)f(Y) = I'(Y)(A(Y)Y) = Y^T A(Y)Y + Y^T A(Y)^T Y$$

for all Y which vanishes because $A(Y)$ is skew-symmetric. □

Example 4.0.2 (Rigid Body [3])**.** The motion of a free rigid body, whose center of mass is at the origin, is described by the Euler equations

$$\begin{aligned} \dot{y}_1 &= a_1 y_2 y_3, \qquad a_1 = (I_2 - I_3)/(I_2 I_3) \\ \dot{y}_2 &= a_2 y_3 y_1, \qquad a_2 = (I_3 - I_1)/(I_3 I_1) \\ \dot{y}_3 &= a_3 y_1 y_2, \qquad a_3 = (I_1 - I_2)/(I_1 I_2) \end{aligned} \tag{4.3}$$

where the vector $y = (y_1, y_2, y_3)^T$ represents the angular momentum in the body frame, and I_1, I_2, I_3 are the principle moments of inertia. This problem can be can be written as:

$$\begin{bmatrix} \dot{y}_1 \\ \dot{y}_2 \\ \dot{y}_3 \end{bmatrix} = \begin{bmatrix} 0 & y_3/I_3 & -y_2/I_2 \\ -y_3/I_3 & 0 & y_1/I_1 \\ y_2/I_2 & -y_1/I_1 & 0 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} \tag{4.4}$$

which is of the form (4.2), with a skew-symmetric matrix $A(Y)$. By theorem (2) $y_1^2 + y_2^2 + y_3^2$ is an invariant and a second quadratic invariant is

$$H(y_1, y_2, y_3) = \frac{1}{2}(\frac{y_1^2}{I_1} + \frac{y_2^2}{I_2} + \frac{y_3^2}{I_3})$$

which represents the kinetic energy.

In [3], the authors applied the implicit midpoint rule and the explicit Euler method to the rigid body problem which illustrates very well the advantages of a geometric integrator in comparison with a classical integrator. As shown in Figure (4.1) below the sphere will be represented with some of the solutions of corresponding to $I_1 = 2$, $I_2 = 1$ and $I_3 = 2/3$. They lie on the intersection of the sphere with the ellipsoid given by $H(y_1, y_2, y_3) = Const$. The implicit midpoint rule was applied (30 steps) with step size $h = 0.3$ and initial value $y_0 = (\cos(1.1), 0, \sin(1.1))^T$. It stays exactly on a solution curve. This follows from the fact that the implicit midpoint rule preserves quadratic invariants exactly.

On the right side of the picture from figure (1), the explicit Euler method was used for 320 steps with $h = 0.05$ and with the same initial value. As the authors conclude in [3], we can see that the numerical solution has a wrong qualitative behaviour (it should lie

on a closed curve). The numerical solution even drifts away from the sphere.

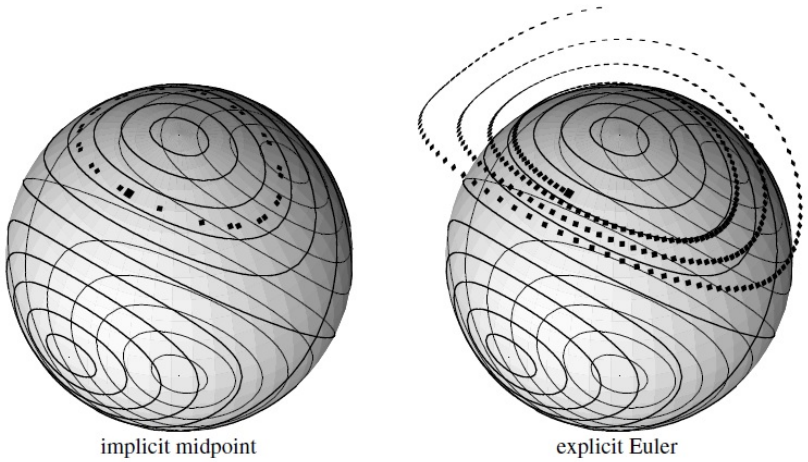


Figure 4.1: Solution of the Euler equations for the rigid body [3].

4.1 Symmetricity and Reversibility

In this section we will talk about symmetric methods and how they play an important role in geometric integration of differential equations. We will discuss also reversible differential equations and reversible maps, and then explain the relation between symmetric integrators and reversible differential equations.

4.1.1 Reversible Differential Equations and Maps

Conservative mechanical systems have the property that inverting the initial direction of the velocity vector and keeping the initial position does not change the solution trajectory, it only inverts the direction of motion. Such systems are said to be reversible.

Definition 4.1.1. Let ρ be an invertible linear transformation in the phase space of $\dot{y} = f(y)$. This differential equation and the vector field $f(y)$ are called ρ-reversible if

$$\rho f(y) = -f(\rho y) \quad \forall y. \tag{4.5}$$

This property is illustrated in the picture on the left side of figure(4.2). For

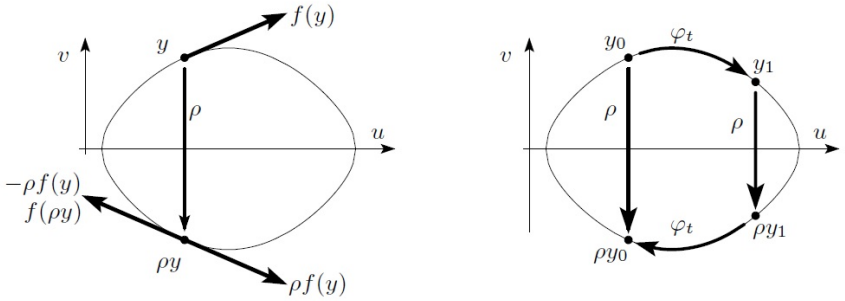


Figure 4.2: Reversible vector field and reversible map [3]

ρ-reversible differential equations, the exact flow $\varphi_t(y)$ satisfies:

$$\rho \circ \varphi_t = \varphi_{-t} \circ \rho = \varphi_t^{-1} \circ \rho, \tag{4.6}$$

as seen in figure (4.2), on the the right side. The identity is a consequence of the group property $\varphi_t \circ \varphi_s = \varphi_{t+s}$ and the left identity follows from:

$$\frac{d}{dt}(\rho \circ \rho_t)(y) = \rho f(\varphi_t(y)) = -f((\rho \circ \varphi_t)(y))$$

$$\frac{d}{dt}(\varphi_{-t} \circ \rho)(y) = -f((\varphi_{-t} \circ \rho)(y)),$$

because expression (4.6) satisfy the same differential equation with the same initial value, $(\rho \circ \varphi_o)(y) = (\varphi_o \circ \rho)(y) = \rho y$. Now we can get a new definition from equation(4.6).

Definition 4.1.2. A map $\phi(y)$ is called ρ-reversible if

$$\rho \circ \phi = \phi^{-1} \circ \rho$$

Example 4.1.1. An important example is the partitioned system:

$$\dot{u} = f(u,v) \qquad \dot{v} = g(u,v), \tag{4.7}$$

where $f(u,-v) = -f(u,v)$ and $g(u,-v) = g(u,v)$. Here the transformation ρ is given by $\rho(u,v) = (u,-v)$. If we call a vector field or a map reversible (without specifying the transformation ρ), we mean that it is ρ-reversible with this particular ρ.

It's always true that $\ddot{u} = g(u)$ is a reversible system when $\dot{u} = v$ and $\dot{v} = g(u)$.

Definition 4.1.3. A numerical one-step method ϕ_h is called symmetric or time reversible, if it satisfies

$$\phi_h \circ \phi_{-h} = id$$

or equivalently $\phi_h = \phi_{-h}^{-1}$. With the definition of the adjoint method ($\phi_h^* = \phi_{-h}^{-1}$) the condition for symmetry reads $\phi_h = \phi_h^*$. A method $y_1 = \phi_h(y_o)$ is symmetric if exchanging $y_o \leftrightarrow y_1$ and $h \leftrightarrow -h$ leaves the method unaltered.

Theorem 4.1.1. *If a numerical method, applied to a ρ-reversible differential equation satisfies*

$$\rho \circ \phi_h = \phi_{-h} \circ \rho \tag{4.8}$$

then the numerical flow ϕ_f is a ρ-reversible map if and only if ϕ_h is a symmetric method.

Proof. As a consequence of equation (4.8), the numerical flow ϕ_h is ρ-reversible if and only if $\phi_{-h} \circ \rho = \phi_h^{-1} \circ \rho$. Since ρ is an invertible transformation, this is equivalent to the symmetry of the method ϕ_h. □

It is also true that a symmetric method is ρ-reversible if and only if the condition (4.8) holds, that is automatically satisfied by most numerical methods.
Let us discuss the validity of (4.8) for the different classes of methods [3]:

- Runge-Kutta methods (explicit or implicit) satisfies (4.8) if (4.5) holds. In particular, the proof for the explicit Euler method $\phi_h(y_o) = y_o + hf(y_o)$ is:

$$\begin{aligned} \rho \circ \phi_h(y_0) &= \rho(y_0 + hf(y_0)) \\ &= \rho(y_0) + \rho hf(y_0) \\ &= \rho(y_0) - hf\rho(y_0) \\ &= \phi_{-h}(\rho y_0) \end{aligned} \tag{4.9}$$

- Partitioned Runge-Kutta Methods applied to a partitioned system satisfy the condition (4.8) if $\rho(u,v) = (\rho_1(u), \rho_2(v))$ with invertible ρ_1 and ρ_2. The proof is the same as for Runge-Kutta methods. Notice that the mapping $\rho(u,v) = (u,-v)$ is of this special form.

- Composition methods. If two methods ϕ_h and ψ_h satisfy (4.8), then so does the adjoint ϕ_h^* and the composition $\phi_h \circ \psi_h$. Consequently, the composition methods, which compose a basic method ϕ_h and its adjoint with different step sizes, have the property (20) provided the basic method ϕ_h has it.

- Splitting methods are based on a splitting $\dot{y} = f^{[1]}(y) + f^{[2]}(y)$ of the differential equation. If both vector fields, $f^{[1]}(y)$ and $f^{[2]}(y)$, satisfy (4.5), then their exact flows $\varphi_h^{[1]}$ and $\varphi_h^{[2]}$ satisfy (4.6). In this situation, the splitting method has the property (4.8).

4.1.2 Symmetric Methods

The explicit and implicit Euler methods are not symmetric methods.
The following methods are symmetric:

- Midpoint Rule:

$$y_{n+1} = y_n + hf\left(\frac{y_{n+1} + y_n}{2}\right). \tag{4.10}$$

By exchanging $h \leftrightarrow -h$ and $(n+1) \leftrightarrow (n)$ we will get:

$$\begin{aligned} y_n &= y_{n+1} - hf\left(\frac{y_n + y_{n+1}}{2}\right) \\ y_{n+1} &= y_n + hf\left(\frac{y_{n+1} + y_n}{2}\right) \end{aligned} \tag{4.11}$$

So midpoint rule is a symmetric method.

- Trapezoidal Rule:

$$y_{n+1} = y_n + \frac{h}{2}\left((y_n) + f(y_{n+1})\right) \tag{4.12}$$

By exchanging $h \leftrightarrow -h$ and $(n+1) \leftrightarrow (n)$ we will get:

$$\begin{aligned} y_n &= y_{n+1} - \frac{h}{2}\left((y_{n+1}) + f(y_n)\right) \\ y_{n+1} &= y_n + \frac{h}{2}\left(f(y_n) + f(y_{n+1})\right) \end{aligned} \tag{4.13}$$

So trapezoidal rule is a symmetric integrator.

Theorem 4.1.2. *The Störmer-Verlet Method given by*

$$\begin{aligned} p_{n+1/2} &= p_n - \frac{h}{2} H_q(p_{n+1/2}, q_n) \\ q_{n+1} &= q_n + \frac{h}{2}(H_p(p_{n+1/2}, q_n) + H_p(p_{n+1/2}, q_{n+1})) \\ p_{n+1} &= p_{n+1/2} - \frac{h}{2} H_q(p_{n+1/2}, q_{n+1}) \end{aligned} \tag{4.14}$$

is a symmetric method.

Proof. By exchanging $(n) \leftrightarrow (n+1)$ and $h \leftrightarrow -h$ we get:

$$\begin{aligned} p_{n+1/2} &= p_{n+1} + \frac{h}{2} H_q(p_{n+1/2}, q_{n+1}) \\ q_n &= q_{n+1} - \frac{h}{2}(H_p(p_{n+1/2}, q_{n+1}) + H_p(p_{n+1/2}, q_n)) \\ p_n &= p_{n+1/2} + \frac{h}{2} H_q(p_{n+1/2}, q_n) \end{aligned} \tag{4.15}$$

then we have

$$\begin{aligned} p_{n+1} &= p_{n+1/2} - \frac{h}{2} H_q(p_{n+1/2}, q_{n+1}) \\ q_{n+1} &= q_n + \frac{h}{2}(H_p(p_{n+1/2}, q_n) + H_p(p_{n+1/2}, q_{n+1})) \\ p_{n+1/2} &= p_n - \frac{h}{2} H_q(p_{n+1/2}, q_n) \end{aligned} \tag{4.16}$$

□

Theorem 4.1.3. *The adjoint method of an s-stage Runge-Kutta method is again an s-stage Runge-Kutta method. Its coefficients are given by*

$$a^*_{ij} = b_{s+1-j} - a_{s+1-i,s+1-j}, \qquad b^*_i = b_{s+1-i}. \tag{4.17}$$

If

$$a_{s+1-i,s+1-j} + a_{ij} = b_j \; \forall \; i, j \tag{4.18}$$

then the Runge-Kutta method is symmetric.[3]

Proof. Exchanging $y_o \leftrightarrow y_1$ and $h \leftrightarrow -h$ in the Runge-Kutta formula yields

$$k_i = f(y_o + h \sum_{j=1}^{s} (b_j - a_{ij}) k_j), \qquad y_1 = y_o + h \sum_{i=1}^{s} b_i k_i. \tag{4.19}$$

Since the values $\sum_{j=1}^{s}(b_j - a_{ij} = 1 - c_i$ appear in the reverse order, we replace k_i by k_{s+1-i} in (28) and then we substitute all indices i and j by $s+1-i$ and $s+1-j$, respectively, this proves (4.17).The assumption (4.18) implies $a^*_{ij} = a_{ij}$ and $b^*_i = b_i$, so that $\phi^*_h = \phi_h$. □

Explicit Runge-Kutta methods cannot fulfill condition (4.18) with $i = j$ and no explicit Runge-Kutta can be symmetric. Now, consider diagonally implicit Runge-Kutta method for which $a_{ij} = 0$ for $i < j$, but with diagonal elements that can be non-zero. In this case condition (4.18) becomes:

$$a_{ij} = b_j = b_{s+1-j} \qquad \text{for } i > j, \quad a_{jj} + a_{s+1-j,s+1-j} = b_j \tag{4.20}$$

The Runge-Kutta table of such a method is thus of the form (for $s = 5$), with $a_{33} =$

c_1	a_{11}				
c_2	b_1	a_{22}			
c_3	b_1	b_2	a_{33}		
$1-c_2$	b_1	b_2	b_3	a_{44}	
$1-c_1$	b_1	b_2	b_3	b_2	a_{55}
	b_1	b_2	b_3	b_2	b_1

$b_3/2$, $a_{44} = b_2 - a_{22}$ and $a_{55} = b_1 - a_{11}$. If one of the b_i, vanishes, then the corresponding stage does not influence the numerical result. This stage can therefore be suppressed, so that the is equivalent to one with fewer stages.

4.2 Symplectic Transformation

The basic objects to be studied are two-dimensional parallelograms in $\mathbb{R}^{2d}$. We suppose the parallelogram to be spanned by two vectors

$$\xi = \begin{bmatrix} \xi^p \\ \xi^q \end{bmatrix}, \eta = \begin{bmatrix} \eta^p \\ \eta^q \end{bmatrix} \text{ in the } (p,q) \text{ space } (\xi^p, \xi^q, \eta^p, \eta^q \in R^d) \text{ as}$$

$$P = \{t\xi + s\eta \quad | 0 \le t \le 1, 0 \le s \le 1\} \tag{4.21}$$

In the case $d = 1$ we consider the oriented area:

or.area(P)= $\det \begin{bmatrix} \xi^p & \eta^p \\ \xi^q & \eta^q \end{bmatrix} = \xi^p \eta^q - \xi^q \eta^p$

as seen on the left picture of figure (4.3). In higher dimension, we replace this by the sum of the projections of P onto the coordinates planes (p_i, q_i), i.e. by

$$\omega(\xi,\eta) := \sum_{i=1}^{d} det \begin{bmatrix} \xi_i^p & \eta_i^p \\ \xi_i^q & \eta_i^q \end{bmatrix} = \sum_{i=1}^{d} (\xi_i^p \eta_i^q - \xi_i^q \eta_i^p) \tag{4.22}$$

This defines a bilinear map acting on vectors of $\mathbb{R}^{2d}$, which will play a central role for Hamiltonian systems. In matrix notation,this map has the form

$$\omega(\xi,\eta) = \xi * TJ\eta \quad \text{with} \quad J = \begin{bmatrix} 0 & I \\ -I & 0 \end{bmatrix} \tag{4.23}$$

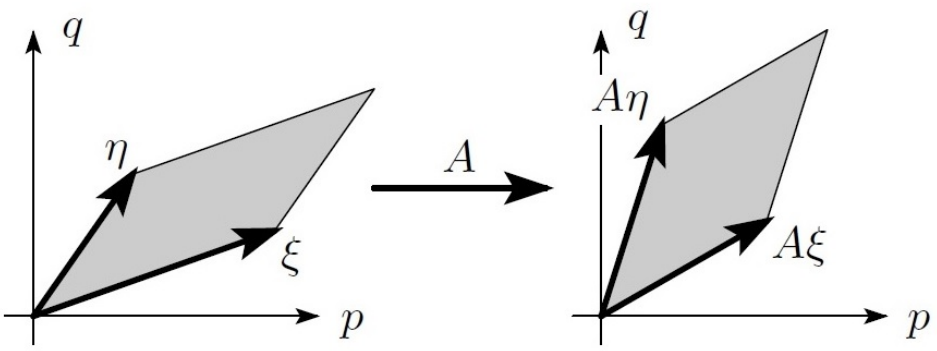


Figure 4.3: Symplecticity (area preservation) of a linear mapping

Definition 4.2.1. A linear mapping $A : \mathbb{R}^{2d} \to \mathbb{R}^{2d}$ is called symplectic if

$$A^T JA = J$$

or equivalently, if $\omega(A\xi, A\eta) = \omega(\xi,\eta)$, $\forall \xi, \eta \in \mathbb{R}^{2d}$.

In the case $d = 1$, where the expression $\omega(\xi, \eta)$ represents the area of the parallelogram P, symplecticity of a linear mapping A is therefore the area preservation of A, as we can see in figure (4.3). In the general case $d > 1$, symplecticity means that the sum of the oriented areas of the projections of P onto (p_i, q_i) is the same as that for the transformed parallelograms $A(P)$.

We now turn our attention to nonlinear mappings. Differentiable functions can be locally approximated by linear mappings. This justifies the following definition.[3]

Definition 4.2.2. A differentiale map $g : U \to \mathbb{R}^{2d}$ (where $U \subset \mathbb{R}^{2d}$ is an open set) is called symplectic if the Jacobian matrix $g'(p,q)$ is everywhere symplectic, i.e., if:

$$g'(p,q)^T J g'(p,q) = J \qquad \text{or} \qquad \omega(g'(p,q)\xi . g'(p,q)\eta) = \omega(\xi, \eta)$$

Theorem 4.2.1. *Let $H(p,q)$ be a twice continuously differentiable function on $U \subset \mathbb{R}^{2d}$, then for each fixed t, the flow φ_t is a symplectic transformation whenever it is defined [3].*

Theorem 4.2.2. *The symplectic Euler methods*

$$p_{n+1} = p_n - hH_q(p_{n+1}, q_n) \qquad or \qquad p_{n+1} = p_n - hH_q(p_n, q_{n+1})$$

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

are symplectic method.

Proof. Consider the Hamiltonian problem

$$\begin{aligned} \dot{p} &= -H_q(p,q) \\ \dot{q} &= H_p(p,q). \end{aligned} \tag{4.24}$$

We consider symplectic Euler is given by

$$\begin{aligned} p_{n+1} &= p_n - hH_q(p_{n+1}, q_n) \\ q_{n+1} &= q_n + hH_p(p_{n+1}, q_n) \end{aligned} \tag{4.25}$$

Differentiating (34) with respect to (p_n, q_n),

$$\begin{aligned} \frac{\partial p_{n+1}}{\partial p_n} &= \frac{\partial p_n}{\partial p_n} - h\left(H_{qq}\frac{\partial q_n}{\partial p_n} + H_{qp}\frac{\partial p_{n+1}}{\partial p_n}\right) = I - hH_{qp}\frac{\partial p_{n+1}}{\partial p_n} \\ \frac{\partial p_{n+1}}{\partial q_n} &= \frac{\partial p_n}{\partial q_n} - h\left(H_{qq}\frac{\partial q_n}{\partial q_n} + H_{qp}\frac{\partial p_{n+1}}{\partial q_n}\right) = -hH_{qq} - hH_{qp}\frac{\partial p_{n+1}}{\partial q_n} \\ \frac{\partial q_{n+1}}{\partial p_n} &= \frac{\partial q_n}{\partial p_n} + h\left(H_{pq}\frac{\partial q_n}{\partial p_n} + H_{pp}\frac{\partial p_{n+1}}{\partial p_n}\right) = hH_{pp}\frac{\partial p_{n+1}}{\partial p_n} \\ \frac{\partial q_{n+1}}{\partial q_n} &= \frac{\partial q_n}{\partial q_n} + h\left(H_{pq}\frac{\partial q_n}{\partial q_n} + H_{pp}\frac{\partial p_{n+1}}{\partial q_n}\right) = I + hH_{pq} + hH_{pp}\frac{\partial p_{n+1}}{\partial q_n} \end{aligned}$$

Noting that $H_{qp} = H_{pq}^T$, we can write the above in matrix form as follows

$$\begin{bmatrix} I + hH_{qp} & 0 \\ -hH_{pp} & I \end{bmatrix} \begin{bmatrix} \frac{\partial p_{n+1}}{\partial p_n} & \frac{\partial p_{n+1}}{\partial q_n} \\ \frac{\partial q_{n+1}}{\partial p_n} & \frac{\partial q_{n+1}}{\partial q_n} \end{bmatrix} = \begin{bmatrix} I & -hH_{qq} \\ 0 & I + H_{qp}^T \end{bmatrix} \tag{4.26}$$

and thus

$$\begin{bmatrix} \frac{\partial p_{n+1}}{\partial p_n} & \frac{\partial p_{n+1}}{\partial q_n} \\ \frac{\partial q_{n+1}}{\partial p_n} & \frac{\partial q_{n+1}}{\partial q_n} \end{bmatrix} = \begin{bmatrix} I + hH_{qp} & 0 \\ -hH_{pp} & I \end{bmatrix}^{-1} \begin{bmatrix} I & -hH_{qq} \\ 0 & I + H_{qp}^T \end{bmatrix} \tag{4.27}$$

Next we need to show that

$$\begin{bmatrix} \frac{\partial p_{n+1}}{\partial p_n} & \frac{\partial p_{n+1}}{\partial q_n} \\ \frac{\partial q_{n+1}}{\partial p_n} & \frac{\partial q_{n+1}}{\partial q_n} \end{bmatrix}^T J \begin{bmatrix} \frac{\partial p_{n+1}}{\partial p_n} & \frac{\partial p_{n+1}}{\partial q_n} \\ \frac{\partial q_{n+1}}{\partial p_n} & \frac{\partial q_{n+1}}{\partial q_n} \end{bmatrix} = J$$

where $J = \begin{bmatrix} 0 & I \\ -I & 0 \end{bmatrix}$ i.e.

$$\begin{bmatrix} I & -hH_{qq} \\ 0 & I+H_{qp}^T \end{bmatrix}^T \begin{bmatrix} I+hH_{qp} & 0 \\ -hH_{pp} & I \end{bmatrix}^{-T} J \begin{bmatrix} I+hH_{qp} & 0 \\ -hH_{pp} & I \end{bmatrix}^{-1} \begin{bmatrix} I & -hH_{qq} \\ 0 & I+H_{qp}^T \end{bmatrix} = J$$

which is equivalent to showing that

$$\begin{bmatrix} I+hH_{qp} & 0 \\ -hH_{pp} & I \end{bmatrix} J^{-1} \begin{bmatrix} I+hH_{qp} & 0 \\ -hH_{pp} & I \end{bmatrix}^T = \begin{bmatrix} I & -hH_{qq} \\ 0 & I+H_{qp}^T \end{bmatrix} J^{-1} \begin{bmatrix} I & -hH_{qq} \\ 0 & I+H_{qp}^T \end{bmatrix}^T.$$

Because H_{pp} and H_{qq} are symmetric matrices, and using $J^{-1} = -J$ the last expression can be written as

$$\begin{bmatrix} I+hH_{qp} & 0 \\ -hH_{pp} & I \end{bmatrix} J \begin{bmatrix} I+hH_{qp}^T & -hH_{pp} \\ 0 & I \end{bmatrix} = \begin{bmatrix} I & -hH_{qq} \\ 0 & I+H_{qp}^T \end{bmatrix} J \begin{bmatrix} I & 0 \\ -hH_{qq} & I+H_{qp} \end{bmatrix}$$

which is fairly easy to verify. □

Theorem 4.2.3. *The implicit midpoint rule*

$$y_{n+1} = y_n + hJ^{-1}\nabla H(y_{n+1}+y_n)/2) \tag{4.28}$$

is a symplectic method.

Proof. let $y_{n+1} = \phi_n(y_n)$ we need to show that

$$\phi_n'^T J \phi_n' = J$$

By differentiating 4.28 we get:

$$\phi_n' = \frac{\partial y_{n+1}}{\partial y_n} = I + hJ^{-1}\nabla^2 H\left(\frac{y_{n+1}+y_n}{2}\right)\left(\frac{1}{2}\right)\left(\frac{\partial y_{n+1}}{\partial y_n}+1\right)$$

$$\phi_n' = \frac{\partial y_{n+1}}{\partial y_n} = \left(I - \frac{h}{2}J^{-1}\nabla^2 H\right)^{-1}\left(I + \frac{h}{2}J^{-1}\nabla^2 H\right)$$

So $\phi_n'^T J \phi_n' = J$ means:

$$\left(I + \frac{h}{2}J^{-1}\nabla^2 H\right) J \left(I + \frac{h}{2}J^{-1}\nabla^2 H\right)^T = \left(I - \frac{h}{2}J^{-1}\nabla^2 H\right) J \left(I - \frac{h}{2}J^{-1}\nabla^2 H\right)^T$$

Since H is symmetric so $(\nabla^2 H)^T = \nabla^2 H$. And $(J^{-1})^T = -J^{-1} = J$.

We get

$$\left(I + \frac{h}{2}J^{-1}\nabla^2 H\right)^T = I - \frac{h}{2}\nabla^2 H J^{-1}$$

$$\left(I - \frac{h}{2}J^{-1}\nabla^2 H\right)^T = I + \frac{h}{2}\nabla^2 H J^{-1}$$

Then

$$\left(IJ + \frac{h}{2}J^{-1}\nabla^2 HJ\right)\left(I - \frac{h}{2}\nabla^2 HJ^{-1}\right) = \left(IJ - \frac{h}{2}J^{-1}\nabla^2 HJ\right)\left(I + \frac{h}{2}\nabla^2 HJ^{-1}\right)$$

$$\begin{aligned} &J - \frac{h}{2}J\nabla^2 HJ^{-1} + \frac{h}{2}J^{-1}\nabla^2 HJ - \frac{h^2}{4}J^{-1}\nabla^2 HJ\nabla^2 HJ^{-1} \\ &= J + \frac{h}{2}J\nabla^2 HJ^{-1} - \frac{h}{2}J^{-1}\nabla^2 HJ - \frac{h^2}{4}J^{-1}\nabla^2 HJ\nabla^2 HJ^{-1} \end{aligned}$$

And we find

$$hJ\nabla^2 HJ^{-1} = hJ^{-1}\nabla^2 HJ$$

$$-J\nabla^2 HJ = -J\nabla^2 HJ$$

So the implicit midpoint rule is symplectic. □

Theorem 4.2.4. *The Störmer-Verlet method is a sympletic method of order 2.*

4.3 Conservation of the Hamiltonian

We know that the Hamiltonian $H(p,q)$ is constant along the exact solutions of the Hamiltonian system. Since the local error of the rth order integrator is of size $O(h^{r+1})$, now if we have

$$H(p_{n+1},q_{n+1}) - H(p_n,q_n) = O(h^{r+1})$$

By taking the summation of these error, we obtain

$$H(p_n,q_n) - H(p_0,q_0) = O(nh^{r+1} = O(th^r), \quad t = nh$$

Since no cancellation of errors can be expected for general integrators. For symplectic integrators, however, we have the much more favourable estimate:

$$H(p_n,q_n) - H(p_0,q_0) = O(h^r)\ for \quad nh \leq T \tag{4.29}$$

With an extremely large T, provided that the numerical solution stays in a compact set. We will use the error in the Hamiltonian equation 4.29 in chapter 5.

Chapter 5: Examples

In this chapter we will numerically solve three well known problems using symplectic methods as well as the classical methods. We perform a comparison between the methods and then decide which of these methods give the better solution. For the case when the Hamiltonian is non-separable we will use a splitting method.

5.1 The Mathematical Pendulum

A classic example is the pendulum, it is a two-dimensional system with phase space $\mathbb{R}^2$ having the Hamiltonian

$$H(p,q) = \frac{1}{2}p^2 - cosq, \tag{5.1}$$

so that the equations of motion are

$$\dot{p} = -\sin q, \quad \dot{q} = p \tag{5.2}$$

where p is the angular momentum and q is the angle of the pendulum. Here are some

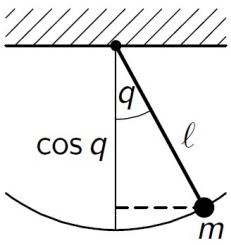


Figure 5.1: Mathematical Pendulum

of the properties of the pendulum:

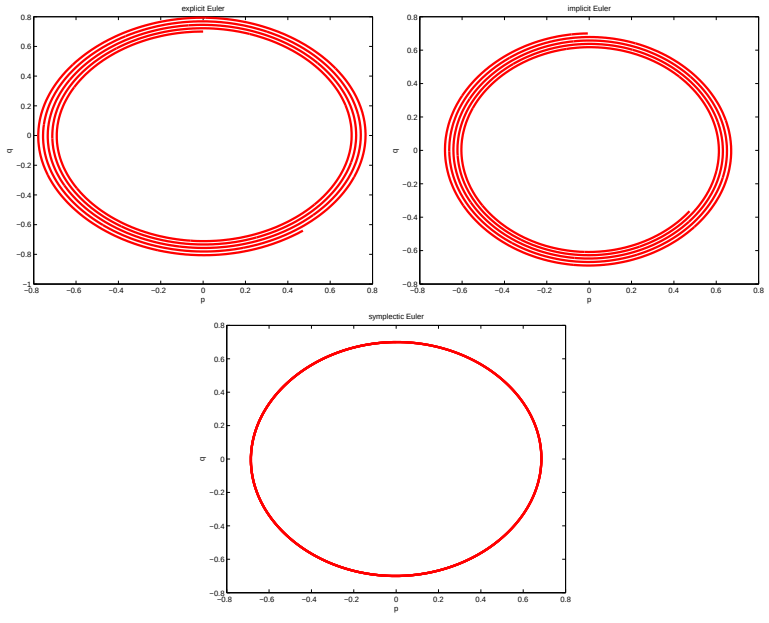
- It conserves the total energy H. That is, its flow stays on the level sets are curves

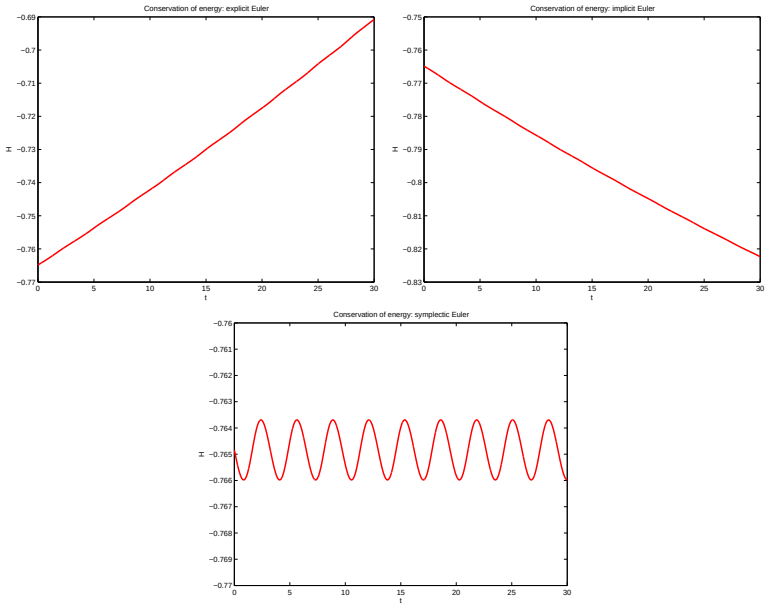
in the plane.

- Being a Hamiltonian system, its flow is symplectic. For two-dimensional systems, this is equivalent to being area-preserving.
- The symmetry, $(p,q) \mapsto (-p,-q)$, maps the system into itself; while the reversing symmetry $(p,q) \mapsto (-p,q)$ maps the vector field into minus itself.

Because this is such a simple system, preserving any of these three properties gives a geometric integrator with good long-time behaviour for almost all initial conditions.

We apply the above numerical methods to the pendulum equations. We consider the intial conditions $(0,1)$ with time step $h = 0.1$. Observe that the numerical solutions of the explicit Euler and the implicit Euler methods (see figs) spiral outwards or inwards. The symplectic Euler shows the correct qualitative behaviour of the problem. The energy (Hamiltonian) even though is not exactly conserved by symplectic Euler, oscillates in certain limits, while both the explicit and implicit Euler go away from the value of the hamiltonian as is we expected.





5.2 Kepler's Problem

Kepler's problem describes the motion in a plane of a material point that is attracted towards the origin with a force inversely proportional to the distance squared. The Hamiltonian of the system is [3, 13]

$$H(p,q) = \frac{1}{2}p^T p - \frac{1}{\|q\|}, \tag{5.3}$$

or equivalently

$$H(p_1,p_2,q_1,q_2) = \frac{1}{2}(p_1^2+p_2^2) - \frac{1}{\sqrt{q_1^2+q_2^2}}, \tag{5.4}$$

where the first term of the Hamiltonian is the kinetic energy and the second the potential. The equations of motion are

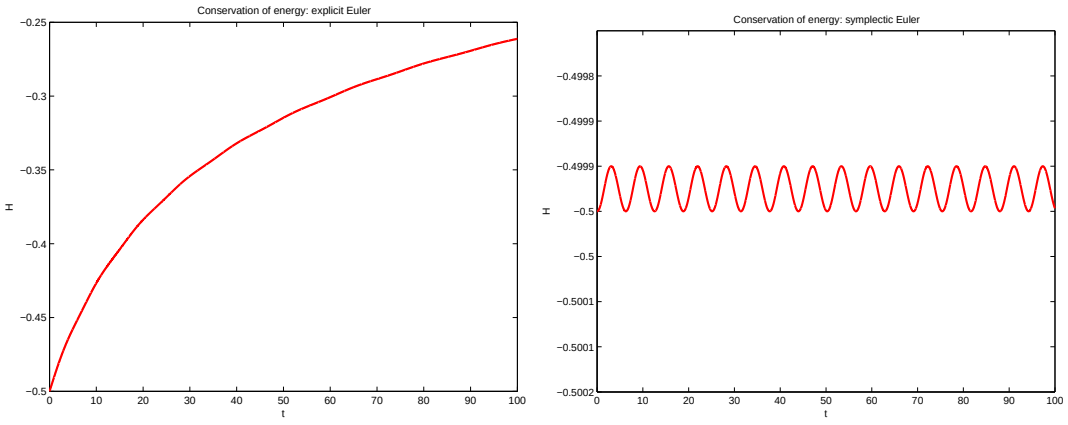
$$\dot{p}_i = -\frac{q_i}{(q_1^2+q_2^2)^{3/2}}, \quad q_i = p_i, \quad i = 1,2. \tag{5.5}$$

The trajectory in the plane is a closed curve, given by an ellipse, i.e., the solution is periodic. Since the problem is autonomous, the Hamiltonian (energy) H is a conserved

quantity. For this we choose the initial conditions $p = (0,1)^T$ and $q = (1,0)^T$, with time step $h = 0.01$. The behaviour of the energy when we apply the three different methods is different. As in the previous example, the energy although is not exactly conserved by symplectic Euler, oscillates in certain limits, even in the long term. However with explicit Euler, it diverges without a limit, going away from the value of the Hamiltonian. We remark that the angular momentum $L(p,q)$ given by

$$L(p,q) = q_1 p_2 - q_2 p_1, \tag{5.6}$$

is exactly conserved by the symplectic Euler.



5.3 The Restricted 3-Body Problem

This is a special case of the 3-body problem. A detailed derivation of this problem can be found in [13]. The idea is to consider two bodies with masses μ and $1-\mu$ respectively where $\mu \in [0, \frac{1}{2}]$. Both bodies are moving under the influence of their mutual gravitational attraction in a circular orbit. The heavier of these two bodies is called the primary and the other one the secondary. To complete the picture, a third body is introduced with a negligible mass that will not affect the motion of the primary and the secondary bodies moving under the influence of their gravity. The Restricted 3-body problem has the Hamiltonian:

$$H(p,q) = \frac{1}{2}\|p\|^2 - q^T J p - \frac{\mu}{r_1} - \frac{1-\mu}{r_2} \tag{5.7}$$

where $q, p \in \mathbb{R}^2$ represent the position and momentum,

$$K = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$$

and r_i the distance from the infinitesimal body to the ith primary,

$$r_1^2 = (q_1 - 1 + \mu)^2 + q_2^2, \quad r_2^2 = (q_1 + \mu)^2 + q_2^2$$

To simplify matters, we consider a limit case of the restricted 3-body problem known as **Hill's lunar problem**. This problem was introduced by G. Hill in [6] as an approximation of the Moon-Earth-Sun system, in order to study the motion of the Moon under the influence of the Earth and perturbed by a distant Sun. A full derivation of this problem can be found in [15]. Hill's problem is defined by the (normalised) Hamiltonian

$$H(p,q) = \frac{1}{2}\|p\|^2 - q^T K p - \frac{1}{\|q\|} + \frac{1}{2} q^T A q \tag{5.8}$$

where

$$A = \begin{bmatrix} -2 & 0 \\ 0 & 1 \end{bmatrix}. \tag{5.9}$$

The equations of motion are given by

$$\begin{aligned} \dot{p} &= Kp - Aq - \frac{q}{\|q\|^3} = -H_q \\ \dot{q} &= p + Kq = H_p \end{aligned} \tag{5.10}$$

where $q = (q_1, q_2)^T$ and $p = (p_1, p_2)^T$. Notice that unlike the restricted problem, the normalized form of HillâĂŹs equations are parameterless

5.3.1 Application of Symmetric and Non-Symmetric Methods

The Runge-Kutta method of order 4

To Apply the Runge-Kutta algorithm to Hill's problem, put $y = (q, p)^T$ and $f = (H_p, -H_q)^T$. This gives

$$\begin{aligned} k_1 &= hf(y_n) \\ k_2 &= hf(y_n + \frac{1}{2}k_1) \\ k_3 &= hf(y_n + \frac{1}{2}k_2) \\ k_4 &= hf(y_n + \frac{1}{2}k_3) \end{aligned} \tag{5.11}$$

The updated position is then obtained from

$$y_{n+1} = y_n + \frac{1}{6}(k_1 + k_2 + k_3 + k_4). \tag{5.12}$$

Symplectic Euler Method

The general symplectic Euler method can be chosen as [3]

$$\begin{aligned} q_{n+1} &= q_n + hH_p(q_n, p_{n+1}) \\ P_{n+1} &= p_n - hH_q(q_n, p_{n+1}) \end{aligned} \tag{5.13}$$

Substituting the equations of motion (5.10) we obtain

$$\begin{aligned} p_{n+1} &= p_n - h\left(K^T p_{n+1} + Aq_n + \frac{q_n}{\|q\|^3}\right) \\ q_{n+1} &= q_n + h(p_{n+1} + Kq_n) \end{aligned} \tag{5.14}$$

where the matrices K and A are defined in (5.9) above. Notice that at this stage, algorithm (5.14) is implicit. However, It can be made explicit by determining p_{n+1} as

follows:

$$\begin{aligned} p_{n+1} &= p_n - hK^T p_{n+1} + h\left(+Aq_n \frac{q_n}{\|q\|^3}\right) \\ \Rightarrow p_{n+1} + hK^T p_{n+1} &= p_n - h\left(+Aq_n \frac{q_n}{\|q\|^3}\right) \\ \Rightarrow \left(I + hK^T\right) p_{n+1} &= p_n - h\left(Aq_n \frac{q_n}{\|q\|^3}\right) \\ \Rightarrow p_{n+1} &= \left(I + hK^T\right)^{-1} \left[p_n - h\left(Aq_n \frac{q_n}{\|q\|^3}\right)\right] \end{aligned} \tag{5.15}$$

The inverse of $\left(I + hK^T\right)$ is non-singular and is given by

$$\left(I + hK^T\right)^{-1} = \frac{1}{1+h^2} \begin{bmatrix} 1 & h \\ -h & 1 \end{bmatrix}. \tag{5.16}$$

Störmer-Verlet Method

We show that the Störmer-Verlet Method for Hill's problem can be expressed explicitly. The Störmer-Verlet algorithm is given by [3]

$$\begin{aligned} p_{n+1/2} &= p_n - \frac{h}{2} H_q(q_n, p_{n+1/2}) \\ q_{n+1} &= q_n + \frac{h}{2}\left[H_p(q_n, p_{n+1/2}) + H_p(q_{n+1}, p_{n+1/2})\right] \\ p_{n+1} &= p_{n+1/2} - \frac{h}{2} H_q(q_{n+1}, p_{n+1/2}) \end{aligned} \tag{5.17}$$

We proceed as follows. Substituting equation (5.10) into (5.17), we can determine the half step $p_{n+1/2}$ as follows:

$$\begin{aligned} p_{n+1/2} &= p_n - \frac{h}{2}\left(K^T p_{n+1/2} + Aq_n + \frac{q_n}{\|q_n\|^3}\right) \\ \Rightarrow p_{n+1/2} + \frac{h}{2}K^T p_{n+1/2} + &= p_n - \frac{h}{2}\left(Aq_n + \frac{q_n}{\|q_n\|^3}\right) \\ \Rightarrow \left(I + \frac{h}{2}K^T\right) p_{n+1/2} &= p_n - \frac{h}{2}\left(Aq_n + \frac{q_n}{\|q_n\|^3}\right) \\ \Rightarrow p_{n+1/2} &= \left(I + \frac{h}{2}K^T\right)^{-1}\left[p_n - \frac{h}{2}\left(Aq_n + \frac{q_n}{\|q_n\|^3}\right)\right] \end{aligned} \tag{5.18}$$

Similarly, we can determine the full step q_{n+1} as follows:

$$\begin{aligned} q_{n+1} &= q_n + \frac{h}{2}\left(p_{n+1/2} + Kq_n + p_{n+1/2} + Kq_{n+1}\right) \\ \Rightarrow q_{n+1} - \frac{h}{2}Kq_{n+1} &= q_n + \frac{h}{2}\left(2p_{n+1/2} + Kq_n\right) \\ \Rightarrow \left(I - \frac{h}{2}K\right) q_{n+1} &= \left(I + \frac{h}{2}K\right) q_n + hp_{n+1/2} \\ \Rightarrow q_{n+1} &= \left(I - \frac{h}{2}K\right)^{-1}\left[\left(I + \frac{h}{2}K\right) q_n + hp_{n+1/2}\right] \end{aligned} \tag{5.19}$$

Finally, collecting the results of (5.18) and (5.19), the method described in (5.17) can be written as

$$\begin{aligned} p_{n+1/2} &= \left(I + \frac{h}{2}K^T\right)^{-1}\left[p_n - \frac{h}{2}\left(Aq_n + \frac{q_n}{\|q_n\|^3}\right)\right] \\ q_{n+1} &= \left(I - \frac{h}{2}K\right)^{-1}\left[\left(I + \frac{h}{2}K\right) q_n + hp_{n+1/2}\right] \\ p_{n+1} &= p_{n+1/2} - \frac{h}{2}\left(K^T p_{n+1/2} + Aq_{n+1} + \frac{q_{n+1}}{\|q_{n+1}\|^3}\right) \end{aligned} \tag{5.20}$$

Here the non-singular matrices $\left(I + \frac{h}{2}K^T\right)$ and $\left(I - \frac{h}{2}K\right)$ are given by

$$\left(I + \frac{h}{2}K^T\right)^{-1} = \left(I - \frac{h}{2}K\right)^{-1} = \frac{1}{4+h^2}\begin{bmatrix} 2 & h \\ -h & 2 \end{bmatrix}$$

The method outlined in (5.20) above is the Störmer-Verlet for Hill's problem. Notice that the method is explicit making for easy implementation.

Splitting the Hamiltonian

The Hamiltonian of Hill's problem $H(p,q)$ defined by equation (5.8) can be efficiently split into linear and nonlinear parts

$$\begin{aligned} H^{[1]} &= \frac{1}{2}\|p\|^2 - q^T K p + \frac{1}{2} q^T A q \\ H^{[2]} &= -\frac{1}{\|q\|} \end{aligned} \tag{5.21}$$

with equations of motion

$$H^{[1]} \quad : \quad \begin{pmatrix} \dot{p} \\ \dot{q} \end{pmatrix} = \Omega \begin{pmatrix} p \\ q \end{pmatrix} \tag{5.22}$$

$$H^{[2]} \quad : \quad \begin{pmatrix} \dot{p} \\ \dot{q} \end{pmatrix} = \begin{pmatrix} \frac{q}{\|q\|^3} \\ 0 \end{pmatrix} \tag{5.23}$$

where

$$\Omega = \begin{pmatrix} K & -A \\ I & K \end{pmatrix} = \begin{pmatrix} 0 & 1 & 2 & 0 \\ -1 & 0 & 0 & -1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & -1 & 0 \end{pmatrix}$$

The equations of motion can be solved exactly to give

$$
\begin{aligned}
H^{[1]} \quad : \quad \begin{pmatrix} p \\ q \end{pmatrix} &= \exp(\Omega t) \begin{pmatrix} p_0 \\ q_0 \end{pmatrix} \\
&= \begin{pmatrix} 2-\cos t & 3t-2\sin t & 3t-\sin t & 1-\cos t \\ -\sin t & 2\cos t-1 & \cos t-1 & -\sin t \\ \sin t & 2-2\cos t & 2-\cos t & \sin t \\ 2\cos t-2 & 4\sin t-3t & 2\sin t-3t & 2\cos t-1 \end{pmatrix} \begin{pmatrix} p_0 \\ q_0 \end{pmatrix} \\
\text{and} \qquad H^{[2]} \quad : \quad \begin{pmatrix} p \\ q \end{pmatrix} &= \begin{pmatrix} p_0 - t\frac{q_0}{\|q_0\|^3} \\ 0 \end{pmatrix}
\end{aligned}
\tag{5.24}
$$

Denoting the flows of these two systems by $\varphi_t^{[1]}$ and $\varphi_t^{[2]}$, we solve the system using the second order composition with $h = \Delta t$

$$
\Phi_h = \varphi_{h/2}^{[1]} \circ \varphi_h^{[2]} \circ \varphi_{h/2}^{[1]} \tag{5.25}
$$

which is the Strang splitting giving the algorithm

$$
\begin{aligned}
(p_{n+1/2}, q_{n+1/2}) &= \exp(\frac{h}{2}\Omega)(p_n, q_n), \\
\bar{p}_{n+1/2} &= p_{n+1/2} - h\frac{q_{n+1/2}}{\|q_{n+1/2}\|^3}, \\
(p_{n+1}, q_{n+1}) &= \exp(\frac{h}{2}\Omega)(\bar{p}_{n+1/2}, q_{n+1/2}).
\end{aligned}
\tag{5.26}
$$

The method is equivalent to the Störmer-Verlet scheme (5.17). It is symplectic and symmetric and, as can be seen, explicit. At each step, the term $\exp(\frac{h}{2}\Omega)$ can be reused, so we only need one evaluation at the first step. Using the symmetric composition (5.25), we can easily construct the higher order symmetric composition methods which is beyond the limits of this thesis.

5.4 Numerical Experiments

To evaluate the quality of our numerical solutions for Hill's problem, we look to the Hamiltonian (energy) $H(p,q)$ which is constant along exact solutions. For symplectic integrators, it can be shown [3] that $H(p_n,q_n - H(p_0,q_0) = \mathcal{O}(h^r)$, where r is the order of the integrator. To this end we measure the performance using the relative error in energy given by

$$E = \frac{|H(p_n,q_n - H(p_0,q_0)|}{|H(p_0,q_0)|} \tag{5.27}$$

The initial conditions we use are $p_0 = (0, 0.509080)$ and $q_0 = (0.50908, 0)$, obtained from [5] and which are known to generate stable Hill orbits of family type g'. In all our experiments we integrate from $t = 0$ to $t = 100$, and we use step-sizes ranging from $h = 0.01$ to h = 10^{-5}. In figure 5.2, we plot the relative error in energy as a function

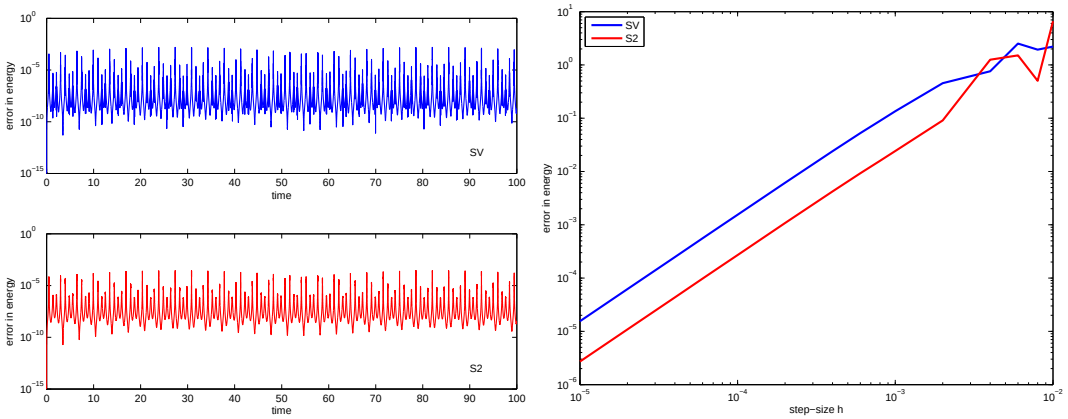


Figure 5.2: Relative error in energy versus time (left) for $h = 10^{-4}$, and the maximum relative energy versus a range of h (right), both for the second order splitting (S1) and the standard Störmer-Verlet (SV).

of time for the standard Störmer-Verlet (SV) (5.20) and the second order composition (S2) (5.25) for $h = 10^{-4}$. We also plot the maximum relative energy over the full range of h. It is clear that S2 has a better performance in terms of precision though we observe from figure 5.3 that is only slightly faster. Here we emphasise that we are not particularly interested in the overall evaluation time for a particular method, but rather in the differences in evaluation time between methods. Also for $h < 10^{-4}$ we

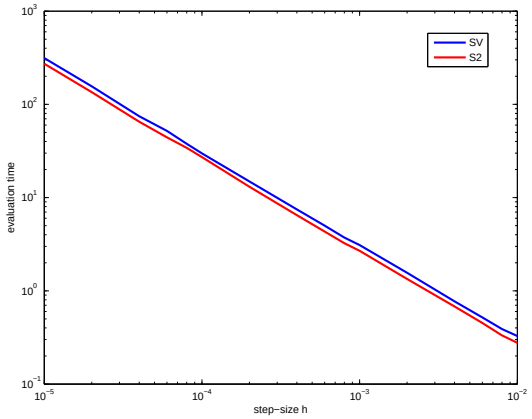


Figure 5.3: Evaluation time versus a range of h for the second order composition (S2) and the standard Störmer-Verlet (SV).

found no real improvement. To complete our comparison, we attempted to include the results of the standard fourth-order Runge-Kutta method. However, The method failed to compute the correct orbits despite using a variable step-size and a very low relative tolerance. This shows that one has to be careful when selecting appropriate methods to integrate Hill's problem.

Chapter 6: Conclusion

In this thesis we applied numerical integration methods to Hamiltonian systems. We showed that some of the classic well known integrators, such as explicit Euler and Runge-Kutta, do not produce good numerical results in the long term in the sense that the trajectories are distorted and the hamiltonian function isnâĂŹt conserved. By contrast we showed that geometric integrators such as the popular Symplectic Euler and the Stormer-Verlet methods when applied to Hamiltonian systems, give accurate results over a long period of time as our experimants showed. Amongst several examples we considered, a procedure for explicitly integrating Hill's equations was presented using a Strang splitting method. Our computational results reveal that new the proposed method is efficient and simple to implement. Using this idea it is anticipated that higher order symplectic methods could be constructed by composition techniques with the midpoint rule and St*ö*rmer-Verlet as the base methods. This class of methods could be developed to efficiently solve many problems involving dynamical systems in general.

Bibliography

[1] C. J. Budd and A. Iserles. Geometric integration: Numerical solution of differential equations on manifolds. *Phil. Trans. R. Soc. Lond. A*, 357:945–956, 1999.

[2] C.J. Budd and M. D. Piggott. Geometric integration and its applications. In *in Handbook of numerical analysis*, pages 35–139. North-Holland, 2000.

[3] E. Hairer, C. Lubich, and G. Wanner. *Geometric Numerical Integration*, volume 31 of *Springer Series in Computational Mathematics*. Springer, 2001.

[4] E. Hairer, C. Lubich, and G. Wanner. Geometric numerical integration illustrated by the stormer-verlet method. *Acta Numerica*, pages 399–450, 2003.

[5] M. Hénon. Numerical exploration of the restricted problem V. Hill's case: periodic orbits and their stability. *Astron. & Astrophys*, 1:223–238, 1969.

[6] G. W. Hill. Researches in the Lunar Theory. *Amer. Jour. of Mathematics*, 1(1):5–26, 1878.

[7] B. Leimkuhler and S. Reich. *Simulating Hamiltonian Dynamics*. Cambridge University Press, 2005.

[8] A Lew, JE Marsden, M Ortiz, and M West. Variational time integrators. *International Journal for Numerical Methods in Engineering*, 60(1):153–212, 2004.

[9] Daniel W Markiewicz. Survey on symplectic integrators. *Preprint Univ. California at Berkeley, Spring*, 1999.

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

[11] R. I. McLachlan and G. R. W. Quispel. Geometric integrators for odes. *Journal of Physics A: Mathematical and General*, 39(19):5251–5286, 2006.

[12] Robert I McLachlan and G Reinout W Quispel. Splitting methods. *Acta Numerica*, 11:341–434, 2002.

[13] K. R. Meyer, G. R. Hall, and D. Offin. *Introduction To Hamiltonian Dynamical Systems And The N-body Problem*. Springer, 1992.

[14] G Moore. Dynamical systems and numerical analysis. *Proceedings of the Edinburgh Mathematical Society (Series 2)*, 41(01):213–216, 1998.

[15] V. Szebehely. *Theory of Orbits*. Academic Press Inc., 1967.

Appendix

```
function Y = EulerExplicit(odefun,tspan,y0)
%
%   The function implements the forward Euler method of order 1.
%
h = diff(tspan);
if any(sign(h(1))*h <= 0)
  error('Entries of TSPAN are not in order.')
end
y0 = y0(:);   % Make a column vector.
neq = length(y0);
N = length(tspan);
Y = zeros(neq,N);
Y(:,1) = y0;
for i = 1:N-1
  Y(:,i+1) = Y(:,i) + h(i)*feval(odefun,Y(:,i));
end

function Y = EulerImplicit(odefun,tspan,y0)
%
%   The function implements the Implicit Euler method of order 1.
%
global yold h
h = tspan(2)-tspan(1);
y0 = y0(:);   % Make a column vector.
neq = length(y0);
N = length(tspan);
Y = zeros(neq,N);
yold = y0;
for i = 1:N-1
  yold = mynewton(@Jfun_pend,yold);
  Y(:,i+1) = yold;
end
return

function [q,p] = SymplecticEuler(dqdt,dpdt,tspan,q0,p0)
%
%  Symplectic Euler function, explicit method.
%  solves Hamilton's equations dq/dt = T'(p), dp/dt = - V'(q)
%
neq = length(q0);
N = length(tspan);
```

```
h = diff(tspan);
q = zeros(neq,N); q(:,1)=q0;
p = zeros(neq,N); p(:,1)=p0;
for i = 1:N-1
    p(:,i+1) = p(:,i) + h(i) * feval(dpdt,q(:,i));
    q(:,i+1) = q(:,i) + h(i) * feval(dqdt,p(:,i+1));
end


function [G,dG] = Jfun_pend(x)
global yold h
G1=x(1)-yold(1)+h*sin(x(2));
G2=x(2)-yold(2)-h*x(1);
G=[G1;G2];
dG1dx1= 1;
dG1dx2 = h*cos(x(2));
dG2dx1 = -h;
dG2dx2 = 1;
dG = [dG1dx1 dG1dx2;dG2dx1 dG2dx2];

function x = mynewton(Jfun,x0,xtol,ftol,maxit,verbose,varargin)
%
% newtonSys  Newton's method for systems of nonlinear equations.
%
if nargin < 3 | isempty(xtol),    xtol = 5e-5;  end
if nargin < 4 | isempty(ftol),    ftol = 5e-5;  end
if nargin < 5 | isempty(maxit),   maxit = 15;   end
if nargin < 6 | isempty(verbose), verbose = 0;  end
xeps = max(xtol,5*eps);  feps = max(ftol,5*eps);    %  Smallest tols are 5*eps
if verbose, fprintf('\nNewton iterations\n  k     norm(f)      norm(dx)\n'); end
x = x0;  k = 0;        %  Initial guess and current number of iterations
while k <= maxit
  k = k + 1;
  [f,J] = feval(Jfun,x);   %   Returns Jacobian matrix and f vector
  dx = J\f;
  x = x - dx;
  if verbose, fprintf('%3d %12.3e %12.3e\n',k,norm(f),norm(dx));  end
  if ( norm(f) < feps ) | ( norm(dx) < xeps ),  return;   end
end
warning(sprintf('Solution not found within tolerance after %d iterations\n',k));

function F = pend(y)
p=y(1); q=y(2);
F = [-sin(q);p];

function out = pend_qt(p)
p=p(:);
```

```
out = p;

function out = pend_pt(q)
q=q(:);
out = -sin(q);

function F = kepler(y)
p1=y(1); p2=y(2); q1=y(3); q2=y(4);
F = [-q1/(q1^2+q2^2)^(3/2);-q2/(q1^2+q2^2)^(3/2);p1;p2];

function out = kepler_qt(p)
p1=p(1); p2=p(2);
out = [p1;p2];

function out = kepler_pt(q)
q=q(:);
out = -q./(q'*q).^(3/2);

%run_pend.m
clear
close all
% Time span.
t=0:.01:30;
% Initial conditions.
p0=0; q0=0.7; y0=[p0,q0];
% Euler explicit
y1 = EulerExplicit(@pend,t,y0);
p2=y1(1,:); q2=y1(2,:);
H1 = 1/2*p2.^2-cos(q2);
% Euler implicit
y2 = EulerImplicit(@Jfun_pend,t,y0);
p3=y2(1,:); q3=y2(2,:);
H2 = 1/2*p3.^2-cos(q3);
% Euler symplectic
[q1 p1]=EulerSymplectic(@pend_qt,@pend_pt,t,q0,p0);
H3 = 1/2*p1.^2-cos(q1);
% plot results
plot(p2,q2,'r','linewidth',3)
xlabel('p')
ylabel('q')
title('explicit Euler')
print -depsc p1.eps
figure
plot(p3(2:length(p3)),q3(2:length(q3)),'r','linewidth',3)
xlabel('p')
ylabel('q')
title('implicit Euler')
```

```
print -depsc p2.eps
figure
plot(p1,q1,'r','linewidth',3)
xlabel('p')
ylabel('q')
title('symplectic Euler')
print -depsc p3.eps
figure
plot(t,H1,'r','linewidth',2)
xlabel('t')
ylabel('H')
title('Conservation of energy: explicit Euler')
print -depsc h1.eps
figure
plot(t(2:length(t)),H2(2:length(H2)),'r','linewidth',2)
xlabel('t')
ylabel('H')
title('Conservation of energy: implicit Euler')
print -depsc h2.eps
figure
plot(t,H3,'r','linewidth',2)
axis([0 30 -0.77 -0.76])
xlabel('t')
ylabel('H')
title('Conservation of energy: symplectic Euler')
print -depsc h3.eps

% run_kepler
clear
close all
% Time span.
t=0:.01:100;
% Initial conditions.
p0=[0,1]; q0=[1,0]; y0=[p0,q0];
% Euler explicit
y = EulerExplicit(@kepler,t,y0);
p1=y(1,:); p2=y(2,:); q1=y(3,:); q2=y(4,:);
H1 = 1/2*(p1.^2+p2.^2)-1./(q1.^2+q2.^2).^0.5;
% Euler symplectic
[v u]=EulerSymplectic(@kepler_qt,@kepler_pt,t,q0,p0);
u1=u(1,:); u2=u(2,:); v1=v(1,:); v2=v(2,:);
H2 = 1/2*(u1.^2+u2.^2)-1./(v1.^2+v2.^2).^0.5;
% plot results
figure
plot(t,H1,'r','linewidth',2)
xlabel('t')
ylabel('H')
```

```
title('Conservation of energy: explicit Euler')
print -depsc h1_kep.eps
figure
plot(t,H2,'r','linewidth',2)
axis([0 30 -0.77 -0.76])
xlabel('t')
ylabel('H')
title('Conservation of energy: symplectic Euler')
axis([0 100 -0.5002 -0.4998])
print -depsc h2_kep.eps

% Runge-Kutta applied to Hill's problem
function rk45 = rungekutta45(eta0,h,tmax)
% Matlab's ODE45
global J A
tic;
%t = [0:h:tmax];
etai = eta0;
Hi =1/2*norm(etai(1:2))^2 - etai(3:4)'*J*etai(1:2)+1/2*etai(3:4)'*A*etai(3:4) - 1

%Integration
options = odeset('RelTol',1e-22,'AbsTol',[1e-22 1e-22 1e-22 1e-22]);
[t,eta] = ode45('phi',[0 tmax],eta0,options);
H = zeros(length(t),1);
E = zeros(length(t),1);
for j=1:length(t)
    H(j)=1/2*norm(eta(j,1:2))^2 - eta(j,3:4)*J*eta(j,1:2)'+1/2*eta(j,3:4)*A*eta(j
         - 1/norm(eta(j,3:4));
    E(j)= abs((Hi-H(j))/Hi);
end
Emax = max(abs(E));
inttime = toc;
rk45 = [t eta H E];
end

% Hill-Stormer
clear
close all
tic
% Parameter set
J = [0 1;-1 0];
A = [-2 0;0 1];
I = eye(2);

% Initial data at t=0 (set A):

p=[0,0.509080];
```

```
q=[0.50908,0];

% Perform the Stormer-Verlet method
tmax=30;                              % the time limit
h=10^(-3);                               % the step size
tspan=0:h:tmax;                       % the number of steps
NT = length(tspan);
neq = length(p);
P = zeros(neq,NT);
Q = zeros(neq,NT);
H = zeros(NT,1);
H(1)=0;
L = [2 h; -h 2];
P(:,1)=p; Q(:,1)=q;
H(1) = 1/2*norm(P(:,1))^2 - Q(:,1)'*J*P(:,1) - Q(1,1)^2 + 1/2*Q(2,1)^2 - 1/norm(Q
for i = 1:NT-1
    Ph = 2/(4+h^2)*L * ( P(:,i) - h/2 * (A*Q(:,i) + Q(:,i)/norm(Q(:,i))^3) );
    Q(:,i+1) = 2/(4+h^2)*L * ( (I+h/2*J)*Q(:,i) + h*Ph );
    P(:,i+1) = (I-h/2*J')*Ph - h/2*(A*Q(:,i+1) + Q(:,i+1)/norm(Q(:,i+1))^3);

    % evaluate Hamiltonian
    H(i+1) = 1/2*norm(P(:,i+1))^2 - Q(:,i+1)'*J*P(:,i+1) -Q(1,i+1)^2 + 1/2*Q(2,i+
             - 1/norm(Q(:,i+1));
    HE(i+1) = abs((H(1)-H(i+1)))/H(1);
end
toc


% strang splitting
function S2 = Splitting2nd(eta0,h,tmax)
%2nd order splitting
global J A L
tic;
t = [0:h:tmax];
etai = eta0;
H = zeros(length(t),1);
E = zeros(length(t),1);
%etao = zeros(length(t),4);
Hi =1/2*norm(etai(1:2))^2 - etai(3:4)'*J*etai(1:2)+1/2*etai(3:4)'*A*etai(3:4)
      - 1/norm(etai(3:4));
v = expm(h/2*L);
%Integration
for i=1:length(t)
    eta1 = phi1(etai,v);
    eta2 = phi2(eta1,h);
    etai = phi1(eta2,v);
    H(i) = 1/2*norm(etai(1:2))^2 - etai(3:4)'*J*etai(1:2)+1/2*etai(3:4)'*A*etai(3
```

```
            - 1/norm(etai(3:4));
    E(i)= abs(Hi-H(i))/abs(Hi);
%    etao(i,:) = etai;
end
Emax = max(E);
Eav = mean(E);
inttime = toc;
S2 = [h Emax Eav inttime];
%S2 = [t' etao H E];
end

function F = phi1(eta,v)
F = v*eta;
end

function F = phi2(eta,t)
F=[eta(1:2)-t*eta(3:4)/norm(eta(3:4))^3;eta(3:4)];
end
```