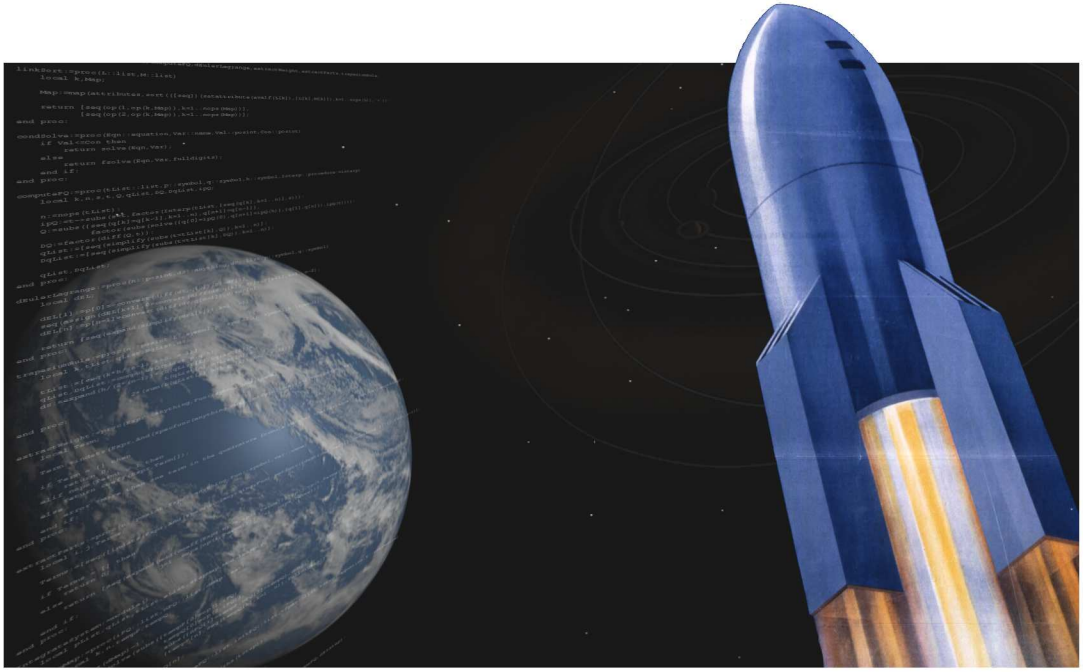
numerics of spacecraft dynamics

christian hellström



university of turku

NUMERICS OF SPACECRAFT DYNAMICS

by

CHRISTIAN HELLSTRÖM

IN PARTIAL FULFILMENT
OF THE REQUIREMENTS FOR THE DEGREE
DOCTOR OF PHILOSOPHY
(THEORETICAL PHYSICS)

UNIVERSITY OF TURKU
DEPARTMENT OF PHYSICS AND ASTRONOMY
FINLAND

December 2010

SUPERVISOR

Dr Seppo Mikkola
Department of Physics and Astronomy
University of Turku
Finland

REVIEWERS

Prof. Mikko Kaasalainen
Department of Mathematics
Tampere University of Technology
Finland

Dr Juha-Pekka Pellonpää
Department of Physics and Astronomy
University of Turku
Finland

EXAMINER

Prof. Benedict J. Leimkuhler
School of Mathematics
University of Edinburgh
United Kingdom

Research funded by the European Commission through *Marie Curie* Actions in the Astrodynamics Network (AstroNet) under contract number MRTN-CT-2006-035151, and conducted at Tuorla Observatory, University of Turku, Finland.

ISBN

978-951-29-4531-3 (printed) 978-951-29-4532-0 (electronic)

Printed by Uniprint Turku

Cover designed by Christian Hellström.
Cover image excerpt *Friede* from the original poster to the 1929 silent film *Frau im Mond* by Fritz Lang.
Courtesy of Friedrich-Wilhelm-Murnau-Stiftung, Wiesbaden, Germany.

Le temps est une invention du mouvement
Celui qui ne bouge pas ne voit pas le temps passer
—Amélie Nothomb

Contents

Introduction

On 21 July 1969 at 02:56 UTC the world watched in awe as Neil Armstrong pronounced the words "That's one small step for [a] man, one giant leap for mankind" from a quarter of a million miles from Earth. For millennia humans had marvelled at our home planet's nocturnal companion from afar, and now live images from its surface were being broadcast to over half a million viewers worldwide. We had split the atom, unravelled the structure of DNA, and now we had triumphantly set foot on a distant celestial body.

With the successful conclusion of the Apollo 17 mission in 1972 the last person to date to have walked on the lunar soil, Eugene Cernan, returned safely to Earth. Although all manned missions have been restricted to low Earth orbits since, the actual utilization of space really took off afterwards. Indeed, more than 3000 man-made objects have successfully been launched into outer space; these include spacecraft, space probes and telescopes, telecommunications satellites, and of course the International Space Station.

The space programme has led up to the development of (micro)computers, global telecommunications networks, worldwide geological and meteorological observations, and advanced (counter-)surveillance operations. Notable spin-offs also include joystick controllers, athletic shoes, novel water purification systems, smoke detectors, fire-resistant materials, lighter air tanks for firefighters, liquid crystal polymers, high-density batteries, laser angioplasty, and enriched baby food, to name but a few. Important contributions to the structural analysis of spacecraft and launch vehicles have, incidentally, been incorporated in modern aircraft and automobile safety design.

Unlike most scientific fields that amalgamate from the tireless endeavours of many a scholar who embarks on a noble quest in pursuit of knowledge, spaceflight has been conceived mainly by visionary novelists, film directors, and scientific revolutionaries. Indeed, probably the first documented reference to space travel goes back to Jules Verne's 1865 novel *De la Terre à la Lune*,

which featured the Columbiad space gun, after which the Apollo 11 command module Columbia was named. In 1901 H. G. Wells published a science fiction novel entitled *The First Men in the Moon*, which inspired the silent film *Le Voyage dans la Lune* by George Méliès, which premiered in 1902. In *Исследование мировых пространств реактивными приборами* (*The Exploration of Cosmic Space by Means of Reaction Devices*), which appeared in 1903, Konstantin Tsiolkovsky explored, inter alia, multi-stage rocket launchers to enable spaceflight, and concluded that the escape velocity could be achieved with rockets fuelled by liquid hydrogen and liquid oxygen. The rocket equation is usually attributed to him, although William Moore had already derived it back in 1813 in *A Treatise on the Motion of Rockets*; that fact has only been established fairly recently [79] though. Important experimental and theoretical contributions to rocketry and astronautics came in 1919 when Robert Goddard published *A Method of Reaching Extreme Altitudes*, and in 1923 with Hermann Oberth's book *Die Rakete zu den Planetenräumen*. During the production of the film *Frau im Mond*, which appeared in 1929, Hermann Oberth advised film director Fritz Lang on all scientific matters; it revealed, among other things, the countdown at launch. Nuclear propulsion systems were contemplated a year later by Robert Esnault-Pelterie in *L'Astronautique*, who also speculated on the possibility of interplanetary travel later on in his life.

With Wernher von Braun their bold ideas were realized, as the first successful suborbital flight of the V-2 rocket took place in 1944. The Saturn V launch vehicle that carried the crew of the Apollo 11 to the Moon was designed and constructed under his active supervision too. Before that milestone in the history of humanity transpired though, Sputnik entered the chronicles on 4 October 1957 as the first artificial satellite to reach outer space. It was followed by Yuri Gagarin, the first man in one full orbit around the Earth, on 12 April 1961.

From the conception through to the critical design review, actual construction, qualification, and operation of a spacecraft it usually takes many years of diligent labour. Consequently, it is of the utmost importance that at each stage of the process each individual involved is fully aware of the design requirements. In the initial phase the mission objectives, which are qualitative in nature, are translated into quantitative mission requirements that form the basis for the mission analysis that yields the design requirements. In all considerations it is obvious that the payload is the main design driver. A detailed overview of space systems engineering and case studies can be found in the book by Fortescue et al. [49].

Mission analysis, as well as both the on-board guidance and navigation systems, and the ground operations control centre, relies on the accurate simulation of the motion of the craft in space and time. Since space missions are extremely expensive and may take decades from design to decommissioning, it is essential that the numerical integration algorithms that determine these trajectories are accurate and stable over long periods of time. Geometric numerical integrators, which have been around for about twenty years, have the desired characteristics. Their use in space applications has been rather limited though, mainly because of financial and scientific-philosophical reasons. In the first place, it can take up to several years before software and the algorithms within are approved for use in space, which means that it is a costly procedure. New algorithms only tend to be introduced when it is absolutely necessary to do so, in accordance with the adage "If it ain't broke, don't fix it". In the second place, the literature on numerical integration algorithms can roughly be divided into two categories, namely applied and theoretical. The conventional path to space systems engineering prefers the applied approach, as the mathematical intricacies that pervade the more theoretically inclined studies quite often tend to elude the more practical philosophy of engineers.

The present dissertation *Numerics of Spacecraft Dynamics* is a humble attempt at opening up the field of geometric numerical integrators, and more specifically variational integrators, to a wider audience. The contents of this thesis are based on the following original research articles:

I C. Hellström, *Creating Variational Integrators with a Computer Algebra System*, Albanian Journal of Mathematics **4**:4 (2010), pp. 105–122.

II C. Hellström & S. Mikkola, *Explicit Algorithmic Regularization in the Few-Body Problem for Velocity-Dependent Perturbations*, Celestial Mechanics and Dynamical Astronomy, **106**:2 (2010), pp. 143–156.

III C. Hellström & S. Mikkola, *Universal Formulation of Quasi-Keplerian Motion, and Its Applications*, New Astronomy, **14**:7 (2009), pp. 607–614.

IV C. Hellström & S. Mikkola, *Satellite Attitude Dynamics and Estimation with the Implicit Midpoint Method*, New Astronomy, **14**:5 (2009), pp. 467–477.

The main results of article I have also been presented at international conferences, in particular:

- C. Hellström, *VarInt: Variational Integrator Design with Maple*, ISSAC 2010, International Symposium on Symbolic and Algebraic Computation, Munich, Germany, 25–28 July, 2010.
- C. Hellström, *VarInt – Variational Integrators with Maple*, ACA'10, Applications of Computer Algebra, Vlora, Albania, 24–27 June, 2010.

The first chapter serves as an expository discourse on dynamical systems theory with a focus on geometric mechanics. It introduces and reviews basic mathematical concepts in order to rephrase these on a higher level of abstraction, from which the differential-geometric scenery of the later chapters can be appreciated more readily. In the second chapter a new method to generate and analyse variational integrators to arbitrary order by means of a computer algebra system is presented. It comprises article I, and it forms the bridge between the theoretical ideas of Chapter 1 and the numerical analysis of prominent space applications in both the chapters that follow it. Chapter 3 introduces the fundamental equations of motion for artificial satellites in the sphere of influence of the Earth; these dynamical equations are to be examined with numerical integration algorithms in the final chapter. Details of the geometric structure of the N-body problem, and in particular the Kepler–Coulomb problem, for which $N = 2$, are described here too. In addition, an analytical (universal) solution is provided for a class of perturbations to the Kepler–Coulomb problem that preserve its integrability, namely the McIntosh–Cisneros–Zwanziger (MICZ) problem, which has been originally derived in III. Both Chapter 3 and 4 consist of the articles II, III and IV. These have been slightly adapted and extended to increase cohesion and avoid unnecessary repetitions. In particular, Chapter 4 showcases a novel numerical integration algorithm, proposed in II, the algorithmically regularized auxiliary-velocity algorithm, or ARAVA for short, that is specifically designed for orbital dynamics with dissipation. Examples of such non-conservative dynamical systems are artificial satellites in low-altitude orbits, such as the International Space Station, and spacecraft re-entering the Earth's atmosphere, for instance the now nearly defunct Space Shuttle. Furthermore, the application of the so-called implicit midpoint method to problem of the computation and estimation of the relative orientation of spacecraft in Earth orbits that does not require quaternions is demonstrated, which has previously been considered in IV.

A few notes on the notation and nomenclature used are in order here. In what follows the summation convention is adopted everywhere except in places where it is explicitly stated otherwise. Furthermore, all vector fields are assumed

to be complete, which ensures the existence and uniqueness of (local) solutions for all times, as assured by the Picard–Lindelöf theorem. For incomplete vector fields the concept of a flow box can be introduced alternatively, but we shall not dwell on such technicalities. Finally, all Lie groups and Lie algebras are finite-dimensional, and are taken over the field of the reals $\mathbb{R}$.

Chapter 1

Dynamics and Geometry

This introductory chapter serves as a review of dynamical systems and their underlying geometric structures. The standard Lagrangian and Hamiltonian formulations of classical mechanics are translated to a modern mathematical framework based on differential geometry and group theory that is known as geometric mechanics [1, 74, 106].

After an abstract and brief discussion of dynamical systems the Lagrangian formalism of classical mechanics is presented in Section 1.2.1, which forms the anacrusis to the Hamiltonian formulation of Section 1.2.2. In Chapter 2 we return to the Lagrangian formalism as the natural setting for variational integrators, though. Section 1.3 describes canonical transformations from a geometric point of view, together with the Hamilton–Jacobi equation and integrability.

1.1 Dynamical Systems

In its most general form a dynamical system consists of a set $\mathcal{X}$, a commutative monoid[1] $\mathbb{T}$, and a map $\varphi_t \colon \mathbb{T} \times \mathcal{X} \to \mathcal{X}$ with $t \in \mathbb{T}$ such that $\varphi_0 = \mathrm{Id}$ and $\varphi_s \circ \varphi_t = \varphi_t \circ \varphi_s = \varphi_{t+s}$. Quite often the set $\mathcal{X}$ is called the state space or phase space. The additive monoid $\mathbb{T}$ represents the notion of time. The requirement

[1] A monoid is a semi-group with an identity element; a group is an invertible monoid.

that $\mathbb{T}$ be a monoid ensures that dynamical systems can propagate in a specific direction of time. Commutativity in conjunction with additivity implies that time is relative. For time-reversible dynamical systems $\mathbb{T}$ is a group; motion in both directions of time is well-defined. The map φ_t describes the evolution over a time interval $[0, t] \subset \mathbb{T}$, and it is sometimes referred to as the evolution operator.

Examples of dynamical systems include continuous dynamical systems, discrete dynamical systems, and cellular automata. For a cellular automaton the state space $\mathcal{X}$ is a finite set, $\mathbb{T} = \mathbb{Z}$, and the evolution operator is the evolution function. For continuous as well as discrete dynamical systems $\mathcal{X}$ is a manifold. For the former class of dynamical systems $\mathbb{T} = \mathbb{R}$ and φ_t is a continuous function, whereas $\mathbb{T} = \mathbb{Z}$ and φ_t is simply a function for the latter. For both continuous and discrete dynamical systems the evolution operator defines the continuous and discrete flows respectively. The continuous (discrete) flow is generally defined implicitly by a set of differential (difference) equations. In particular, for $\mathcal{X} = \mathbb{R}^n$ and $\mathbb{T} = \mathbb{R}$, $\boldsymbol{x}(t) = \varphi_t(\boldsymbol{x}_0)$ is the solution of the initial-value problem

$$\begin{cases} \dfrac{\mathrm{d}\boldsymbol{x}(t)}{\mathrm{d}t} &= \boldsymbol{f}(\boldsymbol{x}(t)), \\ \boldsymbol{x}(0) &= \boldsymbol{x}_0, \end{cases} \tag{1.1a}$$

where the vector field $\boldsymbol{f} \in \mathcal{C}^1(\mathbb{R}^n, \mathbb{R}^n)$ is (locally) Lipschitz continuous. In a similar manner, the difference equation

$$\boldsymbol{x}_{k+1} = \boldsymbol{f}(\boldsymbol{x}_k), \tag{1.1b}$$

with some initial $\boldsymbol{x}_0 \in \mathbb{R}^n$, generates the discrete flow $\varphi_k = \boldsymbol{f}^k = \boldsymbol{f} \circ \boldsymbol{f} \circ \ldots \circ \boldsymbol{f}$, where $k \in \mathbb{Z}$. Henceforth we are concerned with discrete dynamical systems mainly as discretizations of continuous dynamical systems in the context of numerical simulations[2].

For some (initial) $x \in \mathcal{X}$, the set $O(x) = \{t \in \mathbb{T} \mid \varphi_t(x)\}$ is called the orbit through x. An orbit is periodic if there exists a $\tau > 0$, with $\tau \in \mathbb{T}$, such that $\varphi_\tau(x) = x$ for a non-trivial $O(x)$. The smallest τ satisfying the periodicity relation is known as the period. For $\mathbb{T} = \mathbb{Z}$ the designation cycle is to be preferred.

[2]The study of (lattice) difference equations is an emergent field of its own; we refer the interested reader to the forthcoming monograph *Discrete Integrable Dynamics* by Hietarinta, Joshi and Nijhoff, which is based on a series of lecture notes by Hietarinta [69].

A stationary point $\xi \in \mathcal{X}$ satisfies $\varphi_t(\xi) = \xi$ for all $t \in \mathbb{T}$. In continuous time a stationary point is often called an equilibrium, whereas it is known as a fixed point in discrete time.

Consider $\mathcal{S} \subset \mathcal{X}$. An invariant set is defined by the requirement that $\varphi_t(\mathcal{S}) \subset \mathcal{S}$ for all t. Hence, periodic orbits and stationary points are invariant sets.

Let $(\mathcal{X}, \mathbb{T}, \{\varphi_t\}_{t\in\mathbb{T}})$ be a continuous or discrete dynamical system. An orbit $O(x)$ is said to be homoclinic with respect to a stationary point ξ if it approaches the stationary point asymptotically in both directions of time, that is if

$$\lim_{t\to\pm\infty} \varphi_t(x) = \xi.$$

The set $W^{\mathrm{s}} \equiv \{x \in \mathcal{X} \mid \varphi_t(x) = \xi \text{ as } t \to \infty\}$ is aptly called the stable manifold, and $W^{\mathrm{u}} \equiv \{x \in \mathcal{X} \mid \varphi_t(x) = \xi \text{ as } t \to -\infty\}$ is known as the unstable manifold.

Similarly, a heteroclinic orbit is defined as a path in the state space that connects stationary points $\xi_1 \neq \xi_2$, such that

$$\lim_{t\to-\infty} \varphi_t(x) = \xi_1,$$

and

$$\lim_{t\to+\infty} \varphi_t(x) = \xi_2.$$

A heteroclinic orbit is therefore contained in the stable manifold of ξ_2 and the unstable manifold of ξ_1. A homoclinic orbit is obviously a heteroclinic orbit with $\xi_1 = \xi_2$.

Two dynamical systems $(\mathcal{X}, \mathbb{T}, \{\varphi_t\}_{t\in\mathbb{T}})$ and $(\mathcal{X}, \mathbb{T}, \{\psi_t\}_{t\in\mathbb{T}})$ are topologically equivalent if there exists a homeomorphism $h\colon \mathcal{X} \to \mathcal{X}$ that maps orbits of the former onto orbits of the latter whilst preserving the direction of time. For continuous (discrete) dynamical systems defined by the differential (difference) equations as in equation (1.1a) ((1.1b)) topological equivalence implies that the respective vector fields (maps) f and g are h-conjugate, that is $g = h \circ f \circ h^{-1}$. In the continuous case h is of course a diffeomorphism.

1.2 Geometric Mechanics

Geometric mechanics pertains to the intrinsic mathematical structure of classical dynamical systems by means of their symmetries and the herewith associated invariants. We shall only consider continuous dynamical systems here. Their discrete analogues are the topic of subsequent chapters.

1.2.1 Lagrangian Mechanics

The configuration manifold $\mathcal{Q}$ of a free dynamical system can be parameterized by the generalized coordinates q^α of all components that constitute the system, that is the number of degrees of freedom is equal to dimension of the configuration manifold. Here the covariant index α runs from 1 to $n = \dim \mathcal{Q}$. If the system is constrained holonomically, then the dimension of the configuration manifold is reduced by the number of holonomic constraint equations.

At any point $\boldsymbol{q}$ in the configuration manifold we can introduce the generalized velocity $\dot{\boldsymbol{q}}$, where the dot represents differentiation with respect to the time $t \in \mathbb{R}$. The generalized velocity is defined on the tangent space $\mathbf{T}_{\boldsymbol{q}}\mathcal{Q}$ at each point of the configuration manifold. For any smooth curve γ through $\boldsymbol{q} \in \mathcal{Q}$, the velocity at $\boldsymbol{q}$, that is the derivative of γ at $\boldsymbol{q}$, is a vector that lies in the tangent space $\mathbf{T}_{\boldsymbol{q}}\mathcal{Q}$. $\mathbf{T}_{\boldsymbol{q}}\mathcal{Q}$ obviously is an n-dimensional vector space. Moreover, the tangent bundle $\mathbf{T}\mathcal{Q}$ is endowed with a natural structure of a $2n$-dimensional manifold; the tangent spaces 'inherit' the differential structure from the base manifold $\mathcal{Q}$, as a coordinate chart for $\mathcal{Q}$ provides a local trivialization for $\mathbf{T}\mathcal{Q}$ [75, pp. 84–89].

The total (velocity) phase space is obtained by taking $\mathcal{Q}$ and all its tangent spaces $\mathbf{T}_{\boldsymbol{q}}\mathcal{Q}$ as follows:

$$\mathbf{T}\mathcal{Q} = \bigsqcup_{\boldsymbol{q}\in\mathcal{Q}} \mathbf{T}_{\boldsymbol{q}}\mathcal{Q},$$

which is known as the tangent bundle. It is an example of a fibre bundle, or more precisely a vector bundle, for there is a natural projection $\pi\colon \mathbf{T}\mathcal{Q} \to \mathcal{Q}$ with the property that its pre-image $\pi^{-1}(\{\boldsymbol{q}\}) \cong \mathbf{T}_{\boldsymbol{q}}\mathcal{Q}$ for any $\boldsymbol{q} \in \mathcal{Q}$, known as the fibre over $\boldsymbol{q}$, is a vector space. Locally the tangent bundle is homeomorphic to the Cartesian product $\mathcal{Q} \times \mathbf{T}_{\boldsymbol{q}}\mathcal{Q}$, which is the local trivialization. The entire tangent bundle is said to be trivial if and only if it can be written globally as the Cartesian product of two (topological) spaces. In general, the phase space is not trivial.

A (cross-)section of a fibre bundle, in particular the tangent bundle, is a continuous map $X\colon \mathcal{U} \to \mathbf{T}\mathcal{Q}$, where $\mathcal{U}$ is an open subset of $\mathcal{Q}$, such that $\pi \circ X = \mathrm{Id}_{\mathcal{U}}$, the identity on $\mathcal{U}$. Rephrased in perhaps more familiar terms, a section of the tangent bundle is a vector field on the configuration manifold. Equivalently, a vector field is a derivation of the ring of smooth functions $\mathcal{C}^\infty(\mathcal{Q}, \mathbb{R})$.

The structure group of the tangent bundle is the Lie group of invertible linear transformations $\mathrm{GL}(n, \mathbb{R})$. Indeed, consider the (open) neighbourhoods $\mathcal{U}_i$ and $\mathcal{U}_j$, such that $\mathcal{U}_i \cap \mathcal{U}_j \neq \emptyset$, and let $q \mapsto \boldsymbol{q}$ and $q' \mapsto \boldsymbol{q'}$ be their respective

coordinate systems on $\mathcal{Q}$. Any vector field $v \in \mathbf{T}_{\boldsymbol{p}}\mathcal{Q}$ with $\boldsymbol{p} \in \mathcal{U}_i \cap \mathcal{U}_j$ can be expressed locally as both

$$v = v^{\alpha} \frac{\partial}{\partial q^{\alpha}}\bigg|_{\boldsymbol{p}} \quad \text{and} \quad v = v'^{\alpha} \frac{\partial}{\partial q'^{\alpha}}\bigg|_{\boldsymbol{p}}.$$

These are clearly related by

$$v'^{\beta} = \frac{\partial q'^{\beta}}{\partial q^{\alpha}}\bigg|_{\boldsymbol{p}} v^{\alpha}.$$

For there to be a transition between both local representations, the matrix $G_{\alpha}^{\beta} \equiv \partial q'^{\beta} / \partial q^{\alpha}$ must be non-singular, that is $\mathbf{G} \in \mathrm{GL}(n, \mathbb{R})$.

Any time-dependent Lagrangian is a map $L\colon \mathbf{T}\mathcal{Q} \times \mathbb{R} \to \mathbb{R}$, and any time-independent Lagrangian is a function $L\colon \mathbf{T}\mathcal{Q} \to \mathbb{R}$. The dynamics are encoded in the famous Euler–Lagrange equations, which are a set of n first-order differential equations on the tangent bundle[3]:

$$\frac{\mathrm{d}}{\mathrm{d}t} \frac{\partial L}{\partial \dot{q}^{\alpha}} - \frac{\partial L}{\partial q^{\alpha}} = 0, \tag{1.3a}$$

supplied with another set of n first-order differential equations, which relate the points on the configuration manifold to their tangent spaces:

$$\dot{q}^{\alpha} = \frac{\mathrm{d}q^{\alpha}}{\mathrm{d}t}. \tag{1.3b}$$

Together these equations form a set of $2n$ first-order differential equations on $\mathbf{T}\mathcal{Q}$; on the configuration manifold $\mathcal{Q}$ the Euler–Lagrange equations are merely n second-order differential equations. The $2n$-dimensional structure of the tangent bundle becomes more apparent if these differential equations are written in a more concise form as

$$\frac{\mathrm{d}\zeta^{k}}{\mathrm{d}t} = f^{k}(\boldsymbol{\zeta}),$$

where $\boldsymbol{\zeta} \in \mathbf{T}\mathcal{Q}$ with coordinates $(q^1, \ldots, q^n, \dot{q}^1, \ldots, \dot{q}^n)$, and $f^k(\boldsymbol{\zeta})$ is the kth component of the vector field. The first n components reproduce equations (1.3b) and the last n equations yield the Euler–Lagrange equations (1.3a). Please note that both $\boldsymbol{\zeta}$ and $\boldsymbol{f}$ are merely defined on some local chart; we have to specify them on each chart of an atlas for the dynamical system under consideration to be well-defined on the entire tangent manifold.

[3] In fact, $\ddot{q}^{\alpha}$ is considered a coordinate function of $\mathbf{T}(\mathbf{T}\mathcal{Q}) = \mathbf{T}^2\mathcal{Q}$.

Stated as first-order differential equations on the tangent bundle, the Euler–Lagrange equations have a solution that is continuous and unique, as asserted by the Picard–Lindelöf theorem. This means that all trajectories on the tangent bundle are separated from each other; there passes but one trajectory through each point $\boldsymbol{\zeta} = (\boldsymbol{q}, \dot{\boldsymbol{q}})$.

The formalism described above depends on a specific coordinate system on charts of the tangent bundle, in which the diffeomorphism invariance of the Euler–Lagrange equations (1.3) is not apparent at all. It is however possible to rewrite the Euler–Lagrange equations without reference to any coordinate system. To that end, we have to introduce the notions of the Lagrangian one-form θ_L, which can be expressed locally as

$$\theta_L = \frac{\partial L}{\partial \dot{q}^\alpha} \mathrm{d}q^\alpha, \tag{1.4}$$

and the dynamical vector field

$$\Delta_L = \dot{q}^\alpha \frac{\partial}{\partial q^\alpha} + \ddot{q}^\alpha \frac{\partial}{\partial \dot{q}^\alpha}, \tag{1.5}$$

where the components $\ddot{q}^\alpha$ are to be written as functions of $\boldsymbol{q}$ and $\dot{\boldsymbol{q}}$. The dynamical vector field is simply a special case of a general vector field on $\mathbf{T}Q$,

$$\begin{aligned} X(\boldsymbol{q}, \dot{\boldsymbol{q}}) &= X^\alpha \frac{\partial}{\partial \zeta^\alpha} \\ &= X^\alpha_q \frac{\partial}{\partial q^\alpha} + X^\alpha_{\dot{q}} \frac{\partial}{\partial \dot{q}^\alpha}, \end{aligned} \tag{1.6}$$

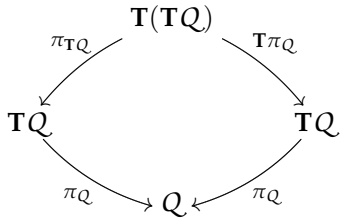
where both $\boldsymbol{X}_q$ and $\boldsymbol{X}_{\dot{q}}$ are functions on the tangent bundle. A vector field on Q is a section of the tangent bundle, $\mathbf{T}Q \xrightarrow{\pi_Q} Q$, which can be written as

$$X = X^\alpha_q(\boldsymbol{q}) \frac{\partial}{\partial q^\alpha},$$

on a chart $\boldsymbol{q} \in Q$, whereas a vector field on $\mathbf{T}Q$ is a slice through $\mathbf{T}(\mathbf{T}Q)$, where $\mathbf{T}(\mathbf{T}Q) \xrightarrow{\pi_{\mathbf{T}Q}} \mathbf{T}Q$, so that

$$X = X^\alpha_q(\boldsymbol{q}, \dot{\boldsymbol{q}}) \frac{\partial}{\partial q^\alpha} + X^\alpha_{\dot{q}}(\boldsymbol{q}, \dot{\boldsymbol{q}}) \frac{\partial}{\partial \dot{q}^\alpha}$$

in local coordinates $(\boldsymbol{q}, \dot{\boldsymbol{q}}) \in \mathbf{T}\mathcal{Q}$, as in equation 1.6. In addition, we have the projection $\mathbf{T}\pi_{\mathcal{Q}}$, such that the ('dual tangent rhombic') diagram



commutes. We shall often write π instead of $\pi_{\mathcal{Q}}$ whenever it is clear.

Apart from the dynamical vector field there are three special vector fields that we shall be concerned with in the sequel, namely (i) second-order vector fields X with $\mathbf{T}\pi_{\mathcal{Q}} \circ X = \mathrm{Id}_{\mathbf{T}\mathcal{Q}}$, for which $\boldsymbol{X}_q = \dot{\boldsymbol{q}}$ and $\boldsymbol{X}_{\dot{q}} = \boldsymbol{X}_{\dot{q}}(\boldsymbol{q}, \dot{\boldsymbol{q}})$ in equation (1.6); (ii) vertical vector fields, for which $\boldsymbol{X}_q$ vanishes identically[4]; and (iii) vector fields associated with point (coordinate) transformations, for which $\boldsymbol{X}_q = \boldsymbol{X}_q(\boldsymbol{q})$ and $X^{\alpha}_{\dot{q}} = \nabla X^{\alpha}_{q} \cdot \dot{\boldsymbol{q}}$. The dynamical vector field Δ_L is an example of a second-order vector field.

Before we can proceed, we have to introduce some mathematical nomenclature. The Lie derivative of a function f with respect to a vector field X is denoted by $\mathscr{L}_X f$ and it is defined by $\mathrm{d}f(X)$, where d is the exterior derivative; the exterior derivative reduces to the total derivative when it acts on zero-forms, which are simply functions. Equivalently, we may write the Lie derivative in terms of the interior product[5] as $\iota_X \mathrm{d}f$, or the natural pairing between duals $\langle \mathrm{d}f, X \rangle$. When applied to one-forms $\mathrm{d}f$, $\mathscr{L}_X \mathrm{d}f = \mathrm{d}(\mathscr{L}_X f)$.

[4] In the classical notation, a vector field X as in equation (1.6) is written as a vector $\boldsymbol{X} = (X^1_q, \ldots, X^n_q, X^1_{\dot{q}}, \ldots, X^n_{\dot{q}})$. The name 'vertical' vector field comes from the fact that these vectors are projected naturally to the null vector, that is $\pi_{\mathcal{Q}}(0, \ldots, 0, X^1_{\dot{q}}, \ldots, X^n_{\dot{q}}) = (0, \ldots, 0) \in \mathcal{Q}$.

[5] Generally, the interior product $\iota_X \colon \Omega^p(\mathcal{Q}) \to \Omega^{p-1}(\mathcal{Q})$ is a contraction of a differential p-form and a vector field $X \in \mathrm{Vec}(\mathcal{Q})$ [see e.g. 50, 56, 75, 129].

Now consider the Lie derivative of the Lagrangian one-form with respect to the dynamical vector field:

$$\begin{aligned}\mathscr{L}_{\Delta_L}\theta_L &= \left(\mathscr{L}_{\Delta_L}\frac{\partial L}{\partial \dot{q}^\alpha}\right)\mathrm{d}q^\alpha + \frac{\partial L}{\partial \dot{q}^\alpha}\mathrm{d}(\mathscr{L}_{\Delta_L}q^\alpha)\\ &= \left(\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial \dot{q}^\alpha}\right)\mathrm{d}q^\alpha + \frac{\partial L}{\partial \dot{q}^\alpha}\mathrm{d}\dot{q}^\alpha\\ &= \frac{\partial L}{\partial q^\alpha}\mathrm{d}q^\alpha + \frac{\partial L}{\partial \dot{q}^\alpha}\mathrm{d}\dot{q}^\alpha\\ &= \mathrm{d}L,\end{aligned}$$

where we have employed Cartan's ('magic') formula,

$$\mathscr{L}_X = \iota_X\mathrm{d} + \mathrm{d}\iota_X, \tag{1.7}$$

as well as the Euler–Lagrange equations (1.3). Hence, we can rewrite the Euler–Lagrange equations in a coordinate-*independent* way as

$$\iota_{\Delta_L}\omega_L = \mathrm{d}L, \tag{1.8}$$

where $\omega_L = -\mathrm{d}\theta_L$, the Lagrangian (symplectic) two-form. In coordinates,

$$\omega_L = \frac{\partial^2 L}{\partial \dot{q}^\alpha \partial q^\beta}\mathrm{d}q^\alpha \wedge \mathrm{d}q^\beta + \frac{\partial^2 L}{\partial \dot{q}^\alpha \partial \dot{q}^\beta}\mathrm{d}\dot{q}^\alpha \wedge \mathrm{d}q^\beta.$$

The flow generated by the dynamical vector field Δ_L is a Lagrangian symplectomorphism, that is $\Phi_t^*\omega_L = \omega_L$:

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\Phi_t^*\omega_L &= \Phi_t^*\mathscr{L}_{\Delta_L}\omega_L\\ &= \Phi_t^*\left(\iota_{\Delta_L}\mathrm{d}\omega_L + \mathrm{d}(\iota_{\Delta_L}\omega_L)\right)\\ &= \Phi_t^*\left(\iota_{\Delta_L}\mathrm{d}^2\theta_L + \mathrm{d}^2 L\right)\\ &= 0,\end{aligned}$$

as $\mathrm{d}^2 = 0$, and $\Phi_0 = \mathrm{Id}$. Please observe that all mathematical structures in equation (1.8) are intrinsic; although all these objects have been expressed in local representations, these expressions hold on all charts, and therefore they are valid globally.

Suppose that the Lagrangian L is invariant under a one-parameter group of diffeomorphisms on $\mathbf{T}Q$ with respect to a complete vector field X. The

infinitesimal transformation that X induces on L is traditionally written as $\delta L \equiv \mathscr{L}_X L$, and we have that

$$\begin{aligned}\mathscr{L}_X L &= \iota_X \mathrm{d}L \\ &= \iota_X \mathscr{L}_{\Delta_L} \theta_L \\ &= \iota_{[X,\Delta_L]} \theta_L + \mathscr{L}_{\Delta_L} \iota_X \theta_L,\end{aligned}$$

in which the first term vanishes for vertical vector fields [see 105, pp. 112–113]. In the derivation we have used the Euler–Lagrange equations (1.8) and the identity[6] $\iota_{[X,Y]} = \mathscr{L}_X \iota_Y - \iota_Y \mathscr{L}_X$. We note that the commutator of any vector field X that corresponds to point transformations and the dynamical vector field, that is $\delta\Delta_L \equiv [X, \Delta_L]$, yields a vertical vector field, as one can verify by direct computation. Hence, we arrive at Noether's theorem:

THEOREM 1 (Noether). *Let X be an infinitesimal generator of a one-parameter group of diffeomorphisms $\{t \in \mathbb{R} \mid \rho_t\}$ that consists of point transformations. If a Lagrangian L is infinitesimally invariant under ρ_ϵ with $\epsilon > 0$, then $\Gamma \equiv \iota_X \theta_L$ is a constant of motion.*

It is worth noting that Noether's theorem can be restated in terms of Lagrangian momentum maps [108]. We shall come to momentum maps in due course when we address symplectic reduction in Section 1.2.3, where they arise naturally in the Hamiltonian formalism.

1.2.2 HAMILTONIAN MECHANICS

In the Lagrangian formulation of classical mechanics dynamical systems are viewed as triples $(\mathbf{T}Q, \mathbb{R}, \{\Phi_t\}_{t\in\mathbb{R}})$. An alternative yet formally more structured approach to classical mechanics has been developed by Hamilton. The transition from the formalism devised by Euler and Lagrange to the Hamiltonian formalism is achieved by a Legendre transformation $\mathbb{F}L \colon \mathbf{T}Q \to \mathbf{T}^\star Q$,

$$\mathbb{F}L \colon (\boldsymbol{q}, \dot{\boldsymbol{q}}) \mapsto \left(\boldsymbol{q}, \frac{\partial L}{\partial \dot{\boldsymbol{q}}}(\boldsymbol{q}, \dot{\boldsymbol{q}})\right). \tag{1.9}$$

Technically, the Legendre transformation is a fibre derivative. It translates the Lagrangian $L(\boldsymbol{q}, \dot{\boldsymbol{q}}, t)$ with the generalized coordinates $\boldsymbol{q}$ and the generalized velocities $\dot{\boldsymbol{q}}$ as independent variables into the Hamiltonian function $H = H(\boldsymbol{q}, \boldsymbol{p}, t)$,

[6] This result follows from the Leibnitz rule for differential forms, $\mathscr{L}_X \iota_Y = \iota_{\mathscr{L}_X Y} + \iota_Y \mathscr{L}_X$, and the fact that $\mathscr{L}_X Y = [X, Y]$ for all vector fields X and Y.

where the generalized momenta are defined as

$$p_\alpha = \frac{\partial L}{\partial \dot{q}^\alpha}. \tag{1.10}$$

These generalized momenta are defined on the cotangent bundle $\mathbf{T}^\star Q$. The cotangent bundle can be constructed in a similar fashion as the tangent bundle by attaching the cotangent spaces at each point $\boldsymbol{q} \in Q$:

$$\mathbf{T}^\star Q = \bigsqcup_{\boldsymbol{q} \in Q} \mathbf{T}^\star_{\boldsymbol{q}} Q,$$

The fibres of the cotangent manifold are the cotangent spaces, which are the duals of the tangent spaces. They represent all allowed values of momentum at each point of the configuration manifold. Smooth cross-sections of the cotangent bundle are one-forms. Henceforward the canonical projection will be denoted by $\pi^\star$.

In terms of the generalized coordinates and the new generalized momenta, the Euler–Lagrange equations become

$$\begin{cases} \dfrac{\mathrm{d}p_\alpha}{\mathrm{d}t} = \dfrac{\partial L}{\partial q^\alpha}, \\ \dfrac{\mathrm{d}q^\alpha}{\mathrm{d}t} = \dot{q}^\alpha. \end{cases}$$

From the exact differential

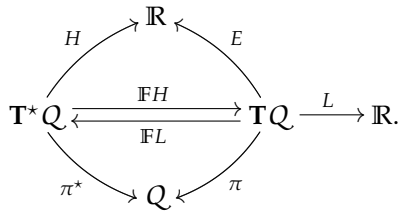
$$\mathrm{d}L = \frac{\partial L}{\partial q^\alpha}\mathrm{d}q^\alpha + \frac{\partial L}{\partial \dot{q}^\alpha}\mathrm{d}\dot{q}^\alpha + \frac{\partial L}{\partial t}\mathrm{d}t,$$

the definition of the canonical momenta (1.10), the Euler–Lagrange equations (1.3), and the identity $p_\alpha \mathrm{d}\dot{q}^\alpha = \mathrm{d}\left(p_\alpha \dot{q}^\alpha\right) - \dot{q}^\alpha \mathrm{d}p_\alpha$, we find that

$$\mathrm{d}\left(p_\alpha \dot{q}^\alpha - L\right) = -\dot{p}_\alpha \mathrm{d}q^\alpha + \dot{q}^\alpha \mathrm{d}p_\alpha - \frac{\partial L}{\partial t}\mathrm{d}t. \tag{1.11}$$

For the left-hand side to be the exact differential of some function that depends on $\boldsymbol{q}$, $\boldsymbol{p}$, and t rather than $\boldsymbol{q}$, $\dot{\boldsymbol{q}}$, and t, the generalized velocities have to be of the form $\dot{q}^\alpha = \dot{q}^\alpha\left(\boldsymbol{q}, \boldsymbol{p}, t\right)$. By inverting equation (1.10) we find the desired result, which may then be substituted into the Lagrangian $L = L\left(\boldsymbol{q}, \dot{\boldsymbol{q}}\left(\boldsymbol{q}, \boldsymbol{p}, t\right), t\right)$. Invertibility of $p_\alpha = p_\alpha\left(\boldsymbol{q}, \dot{\boldsymbol{q}}, t\right)$ at any point $\boldsymbol{q} \in \mathcal{Q}$ at all times is guaranteed if and only if $p_\alpha\left(\boldsymbol{q}, \dot{\boldsymbol{q}}, t\right)$ does not pass through an extremum, so that $\det\left(\partial p_\alpha / \partial \dot{q}^\beta\right) \neq 0$ or that the so-called Hessian $\det\left(\partial^2 L / \partial \dot{q}^\alpha \dot{q}^\beta\right)$ does not vanish. In that case the

Lagrangian L is said to be hyperregular. Generally, a Lagrangian is called *regular* if the fibre derivative is a local isomorphism, whereas it is *hyperregular* if the fibre derivative is a global isomorphism; for any hyperregular Lagrangian (Hamiltonian) the map $\mathbb{F}L\colon \mathbf{T}Q \to \mathbf{T}^\star Q$ ($\mathbb{F}H\colon \mathbf{T}^\star Q \to \mathbf{T}Q$) is a diffeomorphism, and $\mathbb{F}L = (\mathbb{F}H)^{-1}$. Provided that the Hamiltonian is hyperregular, the generalized momenta can be expressed as $p_\alpha = p_\alpha\left(\boldsymbol{q}, \dot{\boldsymbol{q}}, t\right)$. An additional Legendre transformation yields the Euler–Lagrange equations, which establishes the equivalence of both formalisms on the conditions that the Lagrangian $L\left(\boldsymbol{q}, \dot{\boldsymbol{q}}, t\right)$ and the Hamiltonian $H\left(\boldsymbol{q}, \boldsymbol{p}, t\right)$ are both hyperregular. In particular, we have the following commutative diagram:



The Hamiltonian function itself is an expression of the total energy of a conservative, autonomous dynamical system in terms of the canonical coordinates and momenta. Its counterpart in terms of the generalized coordinates and velocities is given by the energy function $E = (\mathbb{F}L)^* H$.

On the premise of a non-singular Hessian matrix,

$$H\left(\boldsymbol{q}, \boldsymbol{p}, t\right) = p_\alpha \dot{q}^\alpha\left(\boldsymbol{q}, \boldsymbol{p}, t\right) - L\left(\boldsymbol{q}, \dot{\boldsymbol{q}}\left(\boldsymbol{q}, \boldsymbol{p}, t\right), t\right) \tag{1.12}$$

is the Hamiltonian function. Its differential equals the left-hand side of equation (1.11), as desired. Upon comparison of the coefficients in

$$\begin{aligned} \mathrm{d}H = & \frac{\partial H}{\partial q^\alpha}\mathrm{d}q^\alpha + \frac{\partial H}{\partial p_\alpha}\mathrm{d}p_\alpha + \frac{\partial H}{\partial t}\mathrm{d}t \\ = & -\dot{p}_\alpha \mathrm{d}q^\alpha + \dot{q}^\alpha \mathrm{d}p_\alpha - \frac{\partial L}{\partial t}\mathrm{d}t, \end{aligned}$$

we find, on the one hand, the celebrated Hamilton's canonical equations,

$$\begin{cases} \dot{q}^\alpha = & \dfrac{\partial H}{\partial p_\alpha}, \\ \dot{p}^\alpha = -\dfrac{\partial H}{\partial q^\alpha}, \end{cases} \tag{1.13}$$

and, on the other hand, the relation

$$\frac{\partial H}{\partial t} = -\frac{\partial L}{\partial t}. \tag{1.14}$$

The solutions of Hamilton's canonical equations yield local expressions for the trajectories on the cotangent bundle. The cotangent bundle is the carrier manifold for the dynamics, and it is also known as the phase space.

Henceforth we consider autonomous dynamical systems. Hamilton's canonical equations (1.13) are $2n$ first-order differential equations on the cotangent bundle. These equations can also be written as

$$\dot{\boldsymbol{\eta}} = \mathbf{J}\nabla H(\boldsymbol{\eta}), \tag{1.15}$$

where $\boldsymbol{\eta} \in \mathbf{T}^\star Q$ of which the coordinates are $(\boldsymbol{q}, \boldsymbol{p})$, and

$$\mathbf{J} = \begin{pmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{I} & \mathbf{0} \end{pmatrix} \in \mathrm{Sp}\,(2n, \mathbb{R}).$$

Here $\mathrm{Sp}\,(2n, \mathbb{R}) = \left\{ \mathbf{A} \in \mathrm{GL}\,(2n, \mathbb{R}) \;\middle|\; \mathbf{A}^{\mathrm{T}}\mathbf{J}\mathbf{A} = \mathbf{J} \right\}$ denotes the symplectic (matrix) group, which in Cartan's classification of finite-dimensional Lie groups is written as $\mathrm{C}_n(\mathbb{R})$.

Yet another equivalent expression frequently found in the literature is

$$\dot{\boldsymbol{\eta}} = \{\boldsymbol{\eta}, H(\boldsymbol{\eta})\}, \tag{1.16}$$

where the Poisson bracket $\{\cdot, \cdot\}$ can be written locally as

$$\{f, g\} = \frac{\partial f}{\partial q^\alpha}\frac{\partial g}{\partial p_\alpha} - \frac{\partial f}{\partial p_\alpha}\frac{\partial g}{\partial q^\alpha}, \tag{1.17}$$

for $f, g \in \mathcal{C}^\infty(\mathbf{T}^\star Q, \mathbb{R})$. The Poisson bracket is related to the canonical symplectic form ω:

$$\{f, g\} = \omega(X_f, X_g) = \mathrm{d}f(X_g) \equiv \mathscr{L}_{X_g} f, \tag{1.18}$$

which endows the cotangent bundle with a symplectic structure. Here X_f and X_g are the (Hamiltonian) vector fields associated with the functions f and g respectively. Generally, a Hamiltonian dynamical system is a triple $(\mathcal{M}, \omega, H)$, where $(\mathcal{M}, \omega)$ is a symplectic manifold.

The canonical symplectic form $\omega = (\mathbb{F}H)^* \omega_L$ can be written locally as

$$\omega = \mathrm{d}q^\alpha \wedge \mathrm{d}p_\alpha. \tag{1.19}$$

In fact, the symplectic form of any symplectic manifold can be written in the standard form (1.19), as asserted by Darboux's theorem [28, 112]:

THEOREM 2 (Darboux). *Let $(\mathcal{M}, \omega)$ be a symplectic manifold, $\mathcal{U} \subset \mathcal{M}$ an open subset, and $(\boldsymbol{x}, \boldsymbol{y})$ coordinates on a chart of $\mathcal{U}$. The symplectic form is*

$$\omega|_{\mathcal{U}} = \mathrm{d}x^\alpha \wedge \mathrm{d}y_\alpha.$$

In general, a symplectic structure ω is (i) *bilinear*, that is $\omega \colon V \times V \to \mathbb{R}$, with V a finite-dimensional vector space; (ii) *skew-symmetric*, that is $\omega(v, w) = -\omega(w, v)$ for all $v, w \in V$; (iii) *non-degenerate*, which implies that $\omega(v, w) = 0$ if and only if $v = 0$ or $w = 0$; and (iv) *closed*, so that $\mathrm{d}\omega(v, w) = 0$ for all $v, w \in V$.

The symplectic structure is fundamental to the dynamics; both the canonical equations of motion and conserved quantities can be expressed in terms of it, as we shall see below. The existence and uniqueness of the Hamiltonian vector field[7] X_H is guaranteed by the fact that the symplectic form is non-degenerate. It is defined by the relation

$$\iota_{X_H}\omega = \mathrm{d}H, \tag{1.20}$$

which is equivalent to $X_H = \mathbf{J}\nabla H$; similar relations hold for all vector fields on the cotangent bundle. Equation (1.20) is the coordinate-independent form of Hamilton's canonical equations. However, in order to see the equivalence to the conventional form of the canonical equations (1.13), we have to introduce the (Hamiltonian) dynamical vector field, which is the counterpart of the Lagrangian dynamical vector field: $\Delta = (\mathbb{F}L)_* \Delta_L$; it is a section of the bundle $\mathbf{T}(\mathbf{T}\mathcal{M}) \to \mathbf{T}\mathcal{M}$ of a symplectic manifold $\mathcal{M}$. In coordinates $(\boldsymbol{q}, \boldsymbol{p})$ the dynamical vector field is

$$\Delta = \dot{q}^\alpha \frac{\partial}{\partial q^\alpha} + \dot{p}_\alpha \frac{\partial}{\partial p_\alpha}.$$

For $X_H = \Delta$ in equation (1.20) we retrieve Hamilton's canonical equations (1.13). In addition, the tautological form $\theta = (\mathbb{F}H)^* \theta_L$ can be written locally as

$$\theta = p_\alpha \mathrm{d}q^\alpha.$$

The symplectic two-form is obviously related to the tautological one-form by $\omega = -\mathrm{d}\theta$. In a similar manner as in the Lagrangian formalism, closedness of the canonical symplectic form ω implies that the flow Φ_t generated by the dynamical vector field is a symplectomorphism[8]. The condition that the symplectic form

[7] A vector field X on $\mathcal{M}$ that preserves ω is called a *symplectic* vector field, so that $\iota_X \omega$ is closed. A *Hamiltonian* vector field is a symplectic vector field for which $\iota_X \omega$ is exact, that is $\iota_X \omega = \mathrm{d}H$ for some $H \in \mathcal{C}^\infty(\mathcal{M}, \mathbb{R})$.

[8] A symplectomorphism f of a symplectic manifold $(\mathcal{M}, \omega)$ is a diffeomorphism $f \in \mathrm{Diff}(\mathcal{M})$ that preserves the symplectic form: $f^* \omega = \omega$.

be skew-symmetric in its arguments implies that the energy, that is Hamiltonian function, is conserved, for $\mathscr{L}_{X_H} H = \iota_{X_H} \mathrm{d}H = \iota_{X_H} \iota_{X_H} \omega = \omega(X_H, X_H) = 0$.

By definition, symplectomorphisms preserve the symplectic form of a $2n$-dimensional symplectic manifold, as well as all its pth exterior powers, where $p \leq n$. These $2p$-forms are known as Poincaré invariants. For $p = n$, we have that the $2n$-form $\omega^n = \omega \wedge \ldots \wedge \omega \neq 0$. Hence, the (normalized) form $\Omega = \omega^n / n!$ is a volume form, and it assigns an orientation to each fibre; it is known as the Liouville measure or the symplectic volume. For Hamiltonian dynamical systems this implies Liouville's theorem:

THEOREM 3 (Liouville). *The phase space volume Ω is preserved under the flow Φ_t generated by the Hamiltonian vector field X_H for all $t \in \mathbb{R}$.*

Even though the discussion has been rather abstract, our efforts will be rewarded. In numerical simulations the preservation of the symplectic structure is crucial. The canonical symplectic form encodes many important characteristics of the phase space, as it determines invariants, stability, spectral properties, such as the location and nature of bifurcations, and the precise location of KAM tori, which are essential to the study of chaos in dynamical systems. These facts will be explored in depth in the next chapters.

1.2.3 REDUCTION

The connection between conserved quantities and symmetries of a dynamical system, especially as expressed by Noether's theorem, is formalized in the concept of a momentum map[9]. Before we can appreciate the rich structure of momentum maps and how these give rise to what is known as symplectic reduction, we have to settle on some definitions and notation, as the literature on the subject tends to be quite varied.

Let $(\mathcal{M}, \omega)$ be a connected symplectic manifold, on which a Lie group G acts smoothly. The action of the Lie group G on the manifold $(\mathcal{M}, \omega)$ is a group homomorphism of G into the group of diffeomorphisms of $\mathcal{M}$:

$$\begin{aligned} \psi \colon G &\to \mathrm{Diff}(\mathcal{M}) \\ g &\mapsto \psi_g, \end{aligned}$$

that is $\psi_e = \mathrm{Id}_{\mathcal{M}}$ and $\psi_{gh} = \psi_g \circ \psi_h$ for all $g, h \in G$. Our definition of a group action is actually a *left* group action as opposed to a *right* group action; a left

[9] Some authors prefer the name 'moment map', which is in fact an incorrect translation of the French term 'application moment', as given by Souriau [147].

(right) group action acts on the manifold $\mathcal{M}$ by left (right) multiplication. Hence, a right group action leads to an anti-homomorphism of G into $\mathrm{Diff}(\mathcal{M})$. Since we only consider left group actions, and thus need not distinguish between either type of group action, we drop the designations 'left' and 'right' altogether. If $\mathcal{M}$ is a vector space and ψ_g is a linear map, then the action of G on $\mathcal{M}$ is simply a representation $\psi\colon G \to \mathrm{End}(\mathcal{M})$.

For any complete vector field X on $\mathcal{M}$, the map $t \mapsto \Phi_t = \exp tX$ obviously defines a smooth action of $\mathbb{R}$ on $\mathcal{M}$; it is nothing but the one-parameter group of diffeomorphisms generated by X.

The derivative at the identity e of the map $G \to \mathrm{Diff}(G)$ that sends each element $h \in G$ by conjugation with $g \in G$ to $g \cdot h \cdot g^{-1}$ is an invertible linear map from the Lie algebra $\mathfrak{g} = \mathrm{Lie}(G)$ to itself, and we denote it by $\mathrm{Ad}_g\colon \mathfrak{g} \to \mathfrak{g}$ for any $g \in G$. It is known as the adjoint action of G on $\mathfrak{g}$:

$$\begin{aligned} \mathrm{Ad}\colon G &\to \mathrm{Diff}(\mathfrak{g}) \subset \mathrm{Aut}(\mathfrak{g}) \\ g &\mapsto \mathrm{Ad}_g\,. \end{aligned}$$

In particular,

$$\mathrm{Ad}_g(\xi) = \frac{\mathrm{d}}{\mathrm{d}t} g \cdot \exp(t\xi) \cdot g^{-1}\bigg|_{t=0}.$$

What is more, we can define an action that is dual to the adjoint representation, called the coadjoint action. For Ad_g the dual Ad_g^* is determined from

$$\left\langle \mathrm{Ad}_g^*\, \Xi, \xi \right\rangle = \left\langle \Xi, \mathrm{Ad}_g\, \xi \right\rangle,$$

where $\Xi \in \mathfrak{g}^*$, the dual of $\mathfrak{g}$, and $\xi \in \mathfrak{g}$. The map $\mathrm{Ad}^*\colon G \to \mathrm{Diff}(\mathfrak{g}^*)$ sends g to $\mathrm{Ad}^*_{g^{-1}}$. These definitions ensure that both the adjoint and coadjoint actions are group homomorphisms.

The group action ψ associated with a Lie group G induces a vector field on $\mathcal{M}$,

$$X^\xi = \frac{\mathrm{d}}{\mathrm{d}t} \psi_{\exp(t\xi)}\bigg|_{t=0}$$

for any $\xi \in \mathfrak{g}$, which is known as the infinitesimal generator. It is a Lie algebra homomorphism $\mathfrak{g} \to \mathrm{Vec}(\mathcal{M})$ with $\xi \mapsto X^\xi$.

Of particular importance to momentum maps are symplectic and Hamiltonian group actions. The action ψ is called symplectic if G acts by symplectomorphisms. In other words, $\psi\colon G \to \mathrm{Symp}(\mathcal{M}, \omega) \subset \mathrm{Diff}(\mathcal{M})$. A symplectic action is said to be Hamiltonian if the vector field generated by ψ is Hamiltonian,

which means that there is function $H^\xi\colon \mathcal{M} \to \mathbb{R}$, such that $\iota_{X^\xi}\omega = \mathrm{d}H^\xi$, where $X^\xi = X_{H^\xi}$ is the vector field generated by ψ.

The answer to the question what symplectic group actions are Hamiltonian leads us to the concept of a momentum map. An action ψ is Hamiltonian if there exists a map $\mu\colon \mathcal{M} \to \mathfrak{g}^*$, such that

$$\mathrm{d}\mu^\xi = \iota_{X^\xi}\omega, \tag{1.21}$$

where $\mu^\xi(\boldsymbol{m}) = \langle \mu(\boldsymbol{m}), \xi\rangle$, the component of μ along $\xi \in \mathfrak{g}$, and μ is equivariant[10] with respect to both the group action and the coadjoint action, that is $\mu \circ \psi_g = \mathrm{Ad}^*_{g^{-1}} \circ \mu$. The quadruple $(\mathcal{M}, \omega, G, \mu)$ is said to be a Hamiltonian G-space, and μ is called the *momentum map*.

We shall mainly be concerned with symplectic manifolds $(\mathbf{T}^\star Q, \omega)$ that are equipped with a tautological one-form θ that is related to the symplectic two-form in the usual way. If G acts on Q, then G can be lifted[11] naturally to a symplectic action on $(\mathbf{T}^\star Q, \omega)$, since the tautological one-form is invariant [28, pp. 11–12] under the lifted action because the canonical projection is G-equivariant; the cotangent lift is a group homomorphism from $\mathrm{Diff}(Q)$ onto $\mathrm{Symp}(\mathbf{T}^\star Q, \omega)$. This means that $\mathscr{L}_{X^\xi}\theta = 0$ for all $X^\xi \in \mathfrak{g}$. By Cartan's formula we obtain that $\mathrm{d}(\iota_{X^\xi}\theta) = -\iota_{X^\xi}\mathrm{d}\theta = \iota_{X^\xi}\omega$, and thus that $\mu^\xi = \iota_{X^\xi}\theta$.

We note that momentum maps are not unique, for they are determined up to constant functions on $\mathcal{M}$; $\mathcal{M}$ is of course assumed to be connected. For μ and ν momentum maps for the same group action this means that $\mu^\xi - \nu^\xi$ is a constant function on $\mathcal{M}$ for all $\xi \in \mathfrak{g}$, and that there is an element $\zeta \in \mathfrak{g}^*$ such that $\mu - \nu = \zeta$. Equivariance of momentum maps determines the constant by the coadjoint action of G on $\mathfrak{g}^*$ [68].

Suppose that there is a function $f\colon \mathcal{M} \to \mathbb{R}$ that is invariant under the action of a group G: $\mathscr{L}_{X^\xi} f = 0$ for any $\xi \in \mathfrak{g}$. By definition, $\mathscr{L}_{X^\xi} f = \iota_{X^\xi}\mathrm{d}f$, which we can also write as $\iota_{X^\xi}\iota_{X_f}\omega = -\iota_{X_f}\iota_{X^\xi}\omega$ because of the skew-symmetry of the symplectic form. We can recognize equation (1.21) immediately, so that

[10] A map $f\colon \mathcal{M} \to \mathcal{N}$ that commutes with the action of a (connected) group G is said to be equivariant when $f \circ \psi_g = \chi_g \circ f$ for all $g \in G$, where $\psi_g\colon \mathcal{M} \to \mathcal{M}$ and $\chi_g\colon \mathcal{N} \to \mathcal{N}$. If $f\colon V \to W$ is a linear function between vector spaces, and ψ and χ are representations on V and W respectively, then equivariance of the map f reduces to f being an intertwiner, or intertwining operator.

[11] For a diffeomorphism $f\colon \mathcal{M} \to \mathcal{N}$, the tangent lift $\mathbf{T}f\colon \mathbf{T}\mathcal{M} \to \mathbf{T}\mathcal{N}$ and its dual the cotangent lift $\mathbf{T}^\star f^{-1}\colon \mathbf{T}^\star\mathcal{M} \to \mathbf{T}^\star\mathcal{N}$ are bundle maps, that is they satisfy the relations $f \circ \pi_{\mathcal{M}} = \pi_{\mathcal{N}} \circ \mathbf{T}f$ and $f \circ \pi^\star_{\mathcal{M}} = \pi^\star_{\mathcal{N}} \circ \mathbf{T}^\star f^{-1}$ respectively. The appearance of the inverse of f in the definition of the cotangent lift is because the cotangent map $\mathbf{T}^\star f\colon \mathbf{T}^\star\mathcal{N} \to \mathbf{T}^\star\mathcal{M}$ covers f^{-1}; the tangent map and tangent lift are synonymous. For a group action $\psi_g\colon Q \to Q$ the tangent lift is simply defined as $\mathbf{T}\psi_g(\boldsymbol{q}, \dot{\boldsymbol{q}}) = (\psi_g(\boldsymbol{q}), \mathbf{T}_{\boldsymbol{q}}\psi_g(\dot{\boldsymbol{q}}))$, and its cotangent lift is given by $\mathbf{T}^\star\psi_g(\boldsymbol{q}, \boldsymbol{p}) = (\psi_g(\boldsymbol{q}), \mathbf{T}^\star_{\psi_g(\boldsymbol{q})}\psi_{g^{-1}}(\boldsymbol{p}))$, where $\psi_{g^{-1}} = (\psi_g)^{-1}$.

$\iota_{X_f}\iota_{X^\xi}\omega = \iota_{X_f}\mathrm{d}\mu^\xi = \mathscr{L}_{X_f}\mu^\xi$. Hence, μ is constant on all trajectories of the vector field of f. The one-parameter group of diffeomorphisms $\{t \in \mathbb{R} \mid \exp(tX_f)\}$ is a symmetry of the Hamiltonian G-space. In conclusion, we can phrase the essence of momentum maps in the geometric version of Noether's theorem:

THEOREM 4 (Noether). *Let $(\mathcal{M}, \omega, G, \mu)$ be a Hamiltonian G-space. If $f\colon \mathcal{M} \to \mathbb{R}$ is a G-invariant function, then μ is constant on the integral curves of the flow generated by X_f.*

A rather trivial example of a momentum map is the Hamiltonian H itself, which is $\mathbb{R}$-invariant. We can identify the dual $\mathbb{R}^\star \cong \mathbb{R}$ through the standard Euclidean inner product, and in particular $\mu^\xi = H \cdot \xi$ and $X^\xi = \xi X_H$. We return the construction of (non-trivial) momentum maps in Chapter 3.

In the context of geometric mechanics we mean by reduction a mathematical procedure that allows us to remove redundant degrees of freedom, and thus simplify the governing equations of motion because of symmetries present in the dynamical system. It is well known that the number of degrees of freedom reduces by the dimension of the symmetry that acts on a dynamical system, whereas the dimension of the phase space reduces by twice that value. A geometric formulation of that fact is contained in the Marsden–Weinstein–Meyer theorem, the statement and appreciation of which require the introduction of a few additional ideas.

The (group) orbit of a point $\boldsymbol{m} \in \mathcal{M}$ associated with a group action ψ is defined as $\mathrm{Orb}_G(\boldsymbol{m}) = \{g \in G \mid \psi_g(\boldsymbol{m})\} \subseteq \mathcal{M}$, which is not to be confused with the previous definition of an orbit, as given in Section 1.1. The isotropy group, or stabilizer, of ψ at $\boldsymbol{m}$ is $G_{\boldsymbol{m}} = \{g \in G \mid \psi_g(\boldsymbol{m}) = \boldsymbol{m}\} \subseteq G$. Similar definitions can be given for the adjoint and coadjoint actions, where the orbits are over the Lie algebra $\mathfrak{g}$ and its dual $\mathfrak{g}^*$ respectively. An action is said to be (i) *transitive*, if there is merely one orbit; (ii) *effective*, or *faithful*, if $g \mapsto \psi_g$ is injective; (iii) *free* if all stabilizers are trivial; (iv) *locally free* if all stabilizers are discrete; and (v) *proper* if all pre-images of compact sets are compact. Every free group action is naturally faithful.

The orbit space $\mathcal{M}/\!/G$ is the quotient $\mathcal{M}/\!\sim$, where $\sim$ denotes the orbit equivalence relation, that is if $\boldsymbol{m}$ and $\boldsymbol{n}$ are on the same orbit, then $\boldsymbol{m} \sim \boldsymbol{n}$. The orbit space is equipped with the standard quotient topology, that is $\mathcal{U} \subset \mathcal{M}/\!/G$ is open if and only if $\sigma^{-1}(\mathcal{U})$ is open, where $\sigma\colon \mathcal{M} \to \mathcal{M}/\!/G$ is the point-orbit projection, which maps each $\boldsymbol{m} \in \mathcal{M}$ to the orbit through $\boldsymbol{m}$. If the coadjoint action is trivial, then equivariance of the momentum map becomes invariance under the group action. In particular, we find that the pre-image of

the momentum map at the zero element $\mu^{-1}(0)$ is invariant under G, because $0 \in \mathfrak{g}^\star$ is a fixed point of the coadjoint action. By virtue of equivariance of the momentum map, $\mu^{-1}(\Xi)$ is in fact invariant for all coadjoint orbits $\Xi \in \mathfrak{g}^*$ because $\mathrm{Orb}_{G_\Xi}(\boldsymbol{m}) = \mathrm{Orb}_G(\boldsymbol{m}) \cap \mu^{-1}(\Xi)$. This means that if Ξ is a regular[12] value of the momentum map μ, then $\mu^{-1}(\Xi)$ is a submanifold of $\mathcal{M}$. If the coadjoint stabilizer G_Ξ acts freely and properly on $\mu^{-1}(\Xi)$, then the quotient $\mu^{-1}(\Xi)/G_\Xi$ is a manifold too. Note that the coadjoint stabilizer G_Ξ always acts locally freely on $\mu^{-1}(\Xi)$ for any regular value Ξ. If, in addition, it acts properly on $\mu^{-1}(\Xi)$, then the quotient $\mu^{-1}(\Xi)/G_\Xi$ is an orbifold rather than a manifold.

The reduction process is summarized in the famous Marsden–Weinstein–Meyer theorem [107, 117]:

THEOREM 5 (Marsden–Weinstein–Meyer). *Let $(\mathcal{M}, \omega, G, \mu)$ be a Hamiltonian G-space for a compact Lie group G, the coadjoint stabilizer of which G_Ξ acts freely and properly on $\mu^{-1}(\Xi)$. The orbit space $\mathcal{M}^\Xi \equiv \mathcal{M}/\!/G_\Xi = \mu^{-1}(\Xi)/G_\Xi$ is a symplectic manifold with a symplectic form given by $i^*\omega = \sigma^*\omega^\Xi$, where $i\colon \mu^{-1}(\Xi) \hookrightarrow \mathcal{M}$ is the inclusion map. Furthermore, $\sigma\colon \mu^{-1}(\Xi) \to \mathcal{M}^\Xi$ is a principal G_Ξ-bundle, and* $\dim \mathcal{M}^\Xi = \dim \mathcal{M} - 2 \dim G_\Xi$.

The symplectic manifold $(\mathcal{M}^\Xi, \omega^\Xi)$ is called the (symplectic) *reduction* of $(\mathcal{M}, \omega)$ at the level Ξ. If the Hamiltonian $H\colon \mathcal{M} \to \mathbb{R}$ of a Hamiltonian G-space is invariant under the action of G, then the reduced Hamiltonian can be calculated from $H^\Xi \circ \sigma = H \circ i$, which allows us to simplify $(\mathcal{M}, \omega, H)$ to $(\mathcal{M}^\Xi, \omega^\Xi, H^\Xi)$.

If the symmetry group of a Hamiltonian G-space is the product of compact connected Lie groups, then it is possible to perform the Marsden–Weinstein–Meyer reduction with respect to each factor. Specifically, if the group $G = G_1 \times G_2 \times \ldots \times G_K$ for some finite $K \in \mathbb{N}$, then obviously $\mathfrak{g} = \mathfrak{g}_1 \oplus \mathfrak{g}_2 \oplus \ldots \oplus \mathfrak{g}_K$ and $\mathfrak{g}^* = \mathfrak{g}_1^* \oplus \mathfrak{g}_2^* \oplus \ldots \oplus \mathfrak{g}_K^*$. Let $\mu_i\colon \mathcal{M} \to \mathfrak{g}_i^*$ be the momentum map with respect to the ith factor, where $i = 1, \ldots, K$. Equivariance of μ implies that μ_i is invariant under G_j with $i \neq j$. This allows us to compute the symplectic reduction one factor of the product at a time, a procedure that is known as *reduction in stages*. More information on reduction in stages and orbit reductions as opposed to point reductions can be found in the notes by Marsden et al. [109].

[12] A value Ξ is regular if and only if the symmetry algebra $\mathfrak{g}_{\boldsymbol{m}} = \mathrm{Lie}(G_{\boldsymbol{m}})$ is trivial for all $\boldsymbol{m} \in \mu^{-1}(\Xi)$.

1.3 Canonical Transformations

Yet another equivalent way of expressing the equations of motion in classical mechanics is by means of the Hamilton–Jacobi equation, which has its own merits, especially in respect of the connections between canonical transformations, integrability, and perturbation theory. In order to make the transition, we have to recast the definition of a symplectomorphism slightly. To that end, consider $(\mathcal{M}_1, \omega_1)$ and $(\mathcal{M}_2, \omega_2)$ to be two symplectic manifolds with $\dim \mathcal{M}_1 = \dim \mathcal{M}_2$, and $f\colon \mathcal{M}_1 \to \mathcal{M}_2$ a map between them. Usually we say that f is a symplectomorphism whenever $\omega_1 = f^*\omega_2$.

The product manifold $(\mathcal{M}_1 \times \mathcal{M}_2, \omega)$ has a symplectic form is given by $\omega = \pi_1^*\omega_1 - \pi_2^*\omega_2$, which is sometimes referred to as the twisted product form[13]. Here $\pi_i\colon \mathcal{M}_1 \times \mathcal{M}_2 \to \mathcal{M}_i$ denotes the projection onto $\mathcal{M}_i$ for $i = 1, 2$. Furthermore, define $i_f\colon \Gamma_f \hookrightarrow \mathcal{M}_1 \times \mathcal{M}_2$ to be the inclusion, where $\Gamma_f = \{\boldsymbol{m} \in \mathcal{M}_1 \mid (\boldsymbol{m}, f(\boldsymbol{m}))\}$ the graph of f. We can see that $\pi_1 \circ i_f = \pi_1|_{\Gamma_f}$, the restriction of the projection onto the first manifold in the product by the graph of the map f, and that $\pi_2 \circ i_f = f \circ \pi_1$. Consequently,

$$i_f^*\omega = (\,\pi_1|_{\Gamma_f})^*\,(\omega_1 - f^*\omega_2)\,.$$

Since $\pi_1 \circ i_f$ is clearly injective, we see that f is a symplectomorphism if and only if $i_f^*\omega = 0$. In that case, $\Gamma_f \subset \mathcal{M}_1 \times \mathcal{M}_2$ is said to be a Lagrangian[14] submanifold [1].

Assume that we pick a one-form θ, such that locally we find the familiar relation $\omega = -\mathrm{d}\theta$. For instance, $\theta = \pi_1^*\theta_1 - \pi_2^*\theta_2$, where $\omega_i = -\mathrm{d}\theta_i$, would do. For any symplectic diffeomorphism f we see that $i_f^*\mathrm{d}\theta = \mathrm{d}i_f^*\theta = 0$, which means that $i_f^*\theta$ must be closed. Locally we may write

$$i_f^*\theta = \mathrm{d}S, \tag{1.22}$$

for some $S\colon \Gamma_f \to \mathbb{R}$. The function S is known as the generating function. We stress that the generating function depends on the choice of θ, and that it is only defined on a local chart.

Suppose that $(\boldsymbol{Q}, \boldsymbol{P})$ and $(\boldsymbol{q}, \boldsymbol{p})$ are coordinates on $\mathcal{M}_1 = \mathbf{T}^\star Q_1$ and $\mathcal{M}_2 = \mathbf{T}^\star Q_2$ respectively. The relation $i_f^*\theta = \mathrm{d}S$ leaves us with four options to choose

[13] In fact, $\omega = \lambda_1\pi_1^*\omega_1 + \lambda_2\pi_2^*\omega_2$ is a symplectic form on $\mathcal{M}_1 \times \mathcal{M}_2$ for any $\lambda_1, \lambda_2 \in \mathbb{R} \setminus \{0\}$.

[14] The symplectic complement of a linear subspace $W \subset V$, where (V, ω) is a (finite-dimensional) symplectic vector space, is the subspace defined by $W^\omega = \{v \in V \mid \omega(v, w) = 0\ \forall w \in W\}$. W is called (i) *symplectic* if $W \cap W^\omega = \{0\}$, that is if $\omega|_{W\times W}$ is non-degenerate; (ii) *isotropic* if $W \subseteq W^\omega$, that is if $\omega|_{W\times W} = 0$; (iii) *co-isotropic* if $W^\omega \subseteq W$; and (iv) *Lagrangian* if $W = W^\omega$ is isotropic, so that $\dim W = \frac{1}{2}\dim V$.

our set of independent coordinates from, namely $(\boldsymbol{q}, \boldsymbol{Q})$, $(\boldsymbol{q}, \boldsymbol{P})$, $(\boldsymbol{p}, \boldsymbol{Q})$, or $(\boldsymbol{p}, \boldsymbol{P})$; these correspond to the four types of canonical transformations. The herewith related choices for the tautological one-forms are (i) $\theta_1 = P_\alpha \, \mathrm{d}Q^\alpha$ and $\theta_2 = p_\alpha \, \mathrm{d}q^\alpha$; (ii) $\theta_1 = -Q^\alpha \, \mathrm{d}P_\alpha$ and $\theta_2 = p_\alpha \, \mathrm{d}q^\alpha$; (iii) $\theta_1 = P_\alpha \, \mathrm{d}Q^\alpha$ and $\theta_2 = -q^\alpha \, \mathrm{d}p_\alpha$; and (iv) $\theta_1 = -Q^\alpha \, \mathrm{d}P_\alpha$ and $\theta_2 = -q^\alpha \, \mathrm{d}p_\alpha$.

The core of the Hamilton–Jacobi approach is deceptively simple: seek a canonical transformation that relates the original dynamical system to another one with trivial integral curves. In other words, we want to find a symplectomorphism f, such that $H \circ f = E$, a constant. Hence,

$$H\left(\boldsymbol{q}, \frac{\partial S}{\partial \boldsymbol{q}}\right) = E, \tag{1.23}$$

which is the time-independent Hamilton–Jacobi equation based on a canonical transformation of the first type. The generating function in the autonomous case is called Hamilton's *characteristic function*[15].

Time dependencies can be worked into the formalism as well by considering contact manifolds that are diffeomorphic to $\mathbf{T}^\star Q \times \mathbb{R}$. In a nutshell, a contact manifold is $(2n+1)$-dimensional manifold with a closed two-form of maximal rank $2n$. We do not go into the details of contact geometry here, as we have no need of it; we merely state the following result:

Theorem 6 (Cartan). *If $(\mathcal{M}, \omega)$ is a symplectic manifold and $H \colon \mathcal{M} \times \mathbb{R} \to \mathbb{R}$ a time-dependent Hamiltonian function, then $(\mathcal{M} \times \mathbb{R}, \tilde{\omega})$ is a contact manifold, where $\tilde{\omega} = \tilde{\pi}^* \omega + \mathrm{d}H \wedge \mathrm{d}t$ and $\tilde{\pi} \colon \mathcal{M} \times \mathbb{R} \to \mathcal{M}$ the projection onto the first factor. Furthermore, the dynamical vector field $\tilde{\Delta}$ is determined uniquely by the equation $\iota_{\tilde{\Delta}} \tilde{\omega} = 0$ and the normalization $\iota_{\tilde{\Delta}} \mathrm{d}t = 1$.*

For a contact manifold there is a one-form $\tilde{\theta}$ such that $\tilde{\theta} \wedge (\mathrm{d}\tilde{\theta})^n$ is a volume form and $\tilde{\omega} = -\mathrm{d}\tilde{\theta}$. In that case, the contact manifold is sometimes referred to as an exact contact manifold. The inquisitive reader may consult the literature for additional information [1, 28, 112].

1.3.1 Extended Phase Space

There is a way to incorporate time dependencies in the geometric formalism described so far, which goes back to Poincaré, and it has recently been developed further by Struckmeier [150, 151] in a more general context. The basic idea is to treat time as an additional canonical coordinate, that is $t(s) \equiv q^0(s)$, where

[15] Sometimes W instead of S is used in the time-independent Hamilton–Jacobi equation.

s is a superordinate evolution parameter of the dynamical system. As a result, there is an associated canonical momentum p_0, and the *extended phase space* is $\mathbf{T}^\star\bar{\mathcal{Q}} \equiv \mathbf{T}^\star(\mathcal{Q}\times\mathbb{R})$, which is $2(n+1)$-dimensional. In that way, we wish to transfer the contact geometry of a t-dependent dynamical system to a symplectic one, where all dynamical quantities are s-independent by construction. Please note that in doing so we introduce an additional degree of freedom to our dynamical system.

In complete analogy with the Hamiltonian function on the usual phase space, we introduce a Hamiltonian function $\bar{H}\colon \mathbf{T}^\star\bar{\mathcal{Q}}\to\mathbb{R}$ on the extended phase space. In order to relate a time-dependent Hamiltonian $H\colon \mathbf{T}^\star\mathcal{Q}\times\mathbb{R}$ to its equivalent on the extended phase space, we require that the contact form $\tilde{\theta}$, as given in Cartan's theorem, forms the basis for a symplectic structure on the extended phase space, which means that the extended phase space $(\mathbf{T}^\star\bar{\mathcal{Q}},\bar{\omega})$ is the *symplectization* of the contact manifold $(\mathbf{T}^\star\mathcal{Q}\times\mathbb{R},\tilde{\omega})$. As such, we have to equate the contact forms $\tilde{\theta}=\tilde{\pi}^*\theta - H\mathrm{d}t$ on $\mathbf{T}^\star\mathcal{Q}\times\mathbb{R}$ with $\tilde{\bar{\theta}}=\tilde{\bar{\pi}}^*\omega + p_0\mathrm{d}q^0 - \bar{H}\mathrm{d}s$ on $\mathbf{T}^\star\bar{\mathcal{Q}}\times\mathbb{R}$, where $\tilde{\bar{\pi}}\colon \mathbf{T}^\star\bar{\mathcal{Q}}\times\mathbb{R}\to\mathbf{T}^\star\mathcal{Q}$ is the projection onto the base phase space[16]. Since $t(s)=q^0$, we get that $\bar{H}\,\mathrm{d}s = (H+p_0)\,\mathrm{d}t$. If we denote $\kappa=\mathrm{d}t/\mathrm{d}s$, then we have the extended Hamiltonian

$$\bar{H}=\kappa(H+p_0). \tag{1.24}$$

For the dynamics to be equivalent on the extended phase space, we require—without loss of generality—that the dynamics evolves on the hypersurface defined by $\bar{H}=0$, which suggests that $p_0=-H\left(\boldsymbol{q}(s),\boldsymbol{p}(s),t(s)\right)$, the instantaneous value of the Hamiltonian. It is important to note that the extended Hamiltonian does not vanish identically; the $(2n+1)$-dimensional hypersurface $\bar{H}=0$ is the analogue of the $(2n-1)$-dimensional hypersurface defined by the relation $H=E$ in the autonomous case.

To be precise, for any compact hypersurface $\mathcal{H}\subseteq\mathcal{M}$, $\omega|_{\mathcal{H}}=-\mathrm{d}\theta'$, where θ' provides the contact structure for $\mathcal{H}$ [112, pp. 113–114]. In our case, $\mathcal{M}=\mathbf{T}^\star\bar{\mathcal{Q}}$ and $\mathcal{H}$ is the codimension-1 hypersurface onto which the t-dependent dynamics is constrained. For the extended phase space to be a symplectization we require the contact form θ' be equal to the contact form $\tilde{\theta}$. Indeed, $\bar{\omega}|_{\bar{H}=0}=-\mathrm{d}\tilde{\theta}$.

With these definitions and relations we can express the extended symplectic form $\bar{\omega}=\mathrm{d}q^\alpha\wedge\mathrm{d}p_\alpha$ as $\omega-\mathrm{d}t\wedge\mathrm{d}H(\boldsymbol{q}(s),\boldsymbol{p}(s),t(s))$. The extended dynamical vector field $\bar{\Delta}$ is obtained from the usual relation $\iota_{\bar{\Delta}}\bar{\omega}=\mathrm{d}\bar{H}$. Locally Hamilton's

[16] The defining relations for the dynamical vector field on $\mathbf{T}^\star\mathcal{Q}\times\mathbb{R}$ can be transferred directly to $\mathbf{T}^\star\bar{\mathcal{Q}}\times\mathbb{R}$, namely $\iota_{\tilde{\bar{\Delta}}}\tilde{\bar{\omega}}=0$ and $\iota_{\tilde{\bar{\Delta}}}\mathrm{d}s=1$.

canonical equations become

$$\begin{cases} \dfrac{\mathrm{d}q^\alpha}{\mathrm{d}s} &= \dfrac{\partial \bar{H}}{\partial p_\alpha}, \\ \dfrac{\mathrm{d}p_\alpha}{\mathrm{d}s} &= -\dfrac{\partial \bar{H}}{\partial q^\alpha}, \end{cases}$$

where now α runs from 0 to n. For $\alpha = 1, \ldots, n$, we obtain equations (1.13) with s as the independent variable, whereas for $\alpha = 0$, we retrieve the identity $\mathrm{d}t/\mathrm{d}s = \kappa$ together with the equation

$$\frac{\mathrm{d}H}{\mathrm{d}s}\left(\boldsymbol{q}(s), \boldsymbol{p}(s), t(s)\right) = \kappa \frac{\partial H}{\partial t}\left(\boldsymbol{q}(t), \boldsymbol{p}(t), t\right).$$

We draw attention to the fact that the aforementioned identity emphasizes that time is indeed a canonical coordinate on the extended phase space.

Canonical transformations $(q^0, \boldsymbol{q}, p_0, \boldsymbol{p}) \mapsto (Q^0, \boldsymbol{Q}, P_0, \boldsymbol{P})$ can be introduced in an identical manner on the extended phase space. These extended canonical transformations are able to generate transformations that link space and time non-trivially, such as the famous Lorentz–Poincaré transformations [150]. In order to have an absolute and globally defined rather than a relative notion of time that is intrinsic to the extended phase space, the transformed time must be independent of the canonical coordinates and momenta, that is $Q^0 \neq Q^0(\boldsymbol{q}, \boldsymbol{p})$. If in addition the transformed canonical coordinates and momenta as well as the transformed time do not depend explicitly on the instantaneous value of the Hamiltonian $H(\boldsymbol{q}(s), \boldsymbol{p}(s), t(s))$, then the extended phase space $\mathbf{T}^\star(Q \times \mathbb{R})$ decomposes as $\mathbf{T}^\star Q \times \mathbf{T}^\star \mathbb{R}$. In that case, the extended canonical transformations factorize as standard canonical transformations multiplied by time reparameterizations. An example of such a time transformation is presented in Chapter 4 when we take a look at the logarithmic Hamiltonian method.

1.3.2 Integrability

To close our cursory exposé of dynamics and its relation to (differential) geometry, we want to address the issue of integrability, as it leads us naturally to a few ideas that we require later on. Integrability—more precisely, integrability in the sense of Liouville—is closely related to separability of the Hamilton–Jacobi equation (1.23) in the sense of Stäckel. We explore separability for the MICZ problem in Appendix C, which is to be introduced in Chapter 3.

Consider a Hamiltonian dynamical system $(\mathcal{M}, \omega, H)$ with Φ_t the flow generated by X_H. As before, $\dim \mathcal{M} = 2n$. A function $f\colon \mathcal{M} \to \mathbb{R}$ lies in the

involution of the Hamiltonian H, that is $\{f, H\} = 0$, if and only if f is constant along integral curves of X_H, which implies that f is a constant of motion. Indeed,

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}(f \circ \Phi_t) &= \Phi_t^* \mathscr{L}_{X_H} f \\ &= \Phi_t^* \imath_{X_H} \mathrm{d}f \\ &= \Phi_t^* \imath_{X_H} \imath_{X_f} \omega \\ &= \Phi_t^* \omega(X_f, X_H) \\ &= \Phi_t^* \{f, H\} \\ &= 0.\end{aligned}$$

We call a Hamiltonian dynamical system (completely) *integrable* when it possesses $n = \frac{1}{2} \dim \mathcal{M}$ independent integrals of motion that are all pairwise in involution.

For a completely integrable $2n$-dimensional dynamical system $(\mathcal{M}, \omega, H)$ with n integrals of motion $\boldsymbol{f} = (f_1, \ldots, f_n)$, the level set $\boldsymbol{f}^{-1}(\boldsymbol{c})$ for any regular value $\boldsymbol{c} \in \mathcal{M}$ is a Lagrangian submanifold[17]. We can see that the connected components of the level set, denoted by $I_{\boldsymbol{c}}$, are homogeneous spaces by noting that $\Phi \colon \mathbb{R}^n \times I_{\boldsymbol{c}} \to I_{\boldsymbol{c}}$ with $\Phi = \Phi^1_{t_1} \circ \Phi^2_{t_2} \circ \ldots \circ \Phi^n_{t_n}$ defines a transitive group action on that level set. Here $\Phi^i_{t_i}$ is the flow generated by X_{f_i}. Hence, $I_{\boldsymbol{c}} \cong G/G_\ell$, where $G = \mathbb{R}^n$ and G_ℓ its stabilizer at $\ell \in I_{\boldsymbol{c}}$. Furthermore $\dim G_\ell = 0$, since $\dim I_{\boldsymbol{c}} = \dim G = n$. Therefore, G_ℓ is a discrete subgroup of G, and $G_\ell \cong \mathbb{Z}^k$ for some $0 \leq k \leq n$.

With these preliminaries we are finally in a position to appreciate the modern formulation of Liouville integrability, which is due to Arnold [5]:

THEOREM 7 (Arnold–Liouville). *Let $(\mathcal{M}, \omega, H)$ be a completely integrable dynamical system, and $\boldsymbol{f} = (f_1, \ldots, f_n)$ its integrals of motion. If the vector fields X_{f_i} for $i = 1, \ldots, n$ are complete on the level set $\boldsymbol{f}^{-1}(\boldsymbol{c})$ for any regular value $\boldsymbol{c}$, then the connected components of $\boldsymbol{f}^{-1}(\boldsymbol{c})$ are diffeomorphic to $\mathbb{R}^k \times \mathbb{T}^{n-k}$ with $0 \leq k \leq n$. In addition, the cotangent bundle of the compact component has coordinates $\boldsymbol{\phi} = (\phi_1, \ldots, \phi_n)$, and there are coordinates $\boldsymbol{J} = (J_1, \ldots, J_n)$, such that these form a Darboux chart $(\boldsymbol{\phi}, \boldsymbol{J})$.*

Here $\mathbb{T}^n = \mathbb{S}^1 \times \mathbb{S}^1 \times \ldots \times \mathbb{S}^1$, with n factors, denotes the n-torus. The $\boldsymbol{J}$ are known as the action coordinates, and the $\boldsymbol{\phi}$ are called the angle coordinates. It is obvious that complete integrability, as expressed by the Arnold–Liouville theorem, limits the allowed topologies of $\mathcal{M}$ severely.

[17] Indeed, its dimension is exactly n, and its tangent bundle is isotropic.

The standard manner in which the canonical transformation that achieves the transition to action–angle coordinates is obtained is as follows. Let $\gamma_\alpha(\boldsymbol{c})$ denote the homology cycle of the αth factor $\mathbb{S}^1$ of the n-torus for $i = 1, \ldots, n$ and with $\boldsymbol{c} \in \mathcal{M}$. We normalize the homology cycles such that the angle coordinates are defined modulo 2π:

$$\oint_{\gamma_\alpha(\boldsymbol{c})} \mathrm{d}\phi^\alpha = 2\pi.$$

With a canonical transformation of the second type for $(\boldsymbol{q}, \boldsymbol{p}) \mapsto (\boldsymbol{\phi}, \boldsymbol{J})$ we know that locally

$$p_\alpha = \frac{\partial S}{\partial q^\alpha}, \qquad \phi^\alpha = \frac{\partial S}{\partial J_\alpha}.$$

This leads us to the relation

$$J_\alpha = \frac{1}{2\pi} \oint_{\gamma_\alpha(\boldsymbol{c})} p_\beta \, \mathrm{d}q^\beta, \tag{1.25}$$

where the integrand is the local representation of $i^*(\theta)$ with $i\colon \boldsymbol{f}^{-1}(\boldsymbol{c}) \hookrightarrow \mathcal{M}$. In principle, these relations establish the connection between $(\boldsymbol{q}, \boldsymbol{p})$ and $(\boldsymbol{\phi}, \boldsymbol{J})$ for completely integrable dynamical systems. The map $\mathscr{A}\colon (\boldsymbol{q}, \boldsymbol{p}) \mapsto (\boldsymbol{\phi}, \boldsymbol{J})$ is a local diffeomorphism by the Arnold–Liouville theorem, but since it is a (local) bijection by construction, it turns out to be a global diffeomorphism. In fact, invertibility of the canonical transformation together with the periodicity in the angles allows us to write the original canonical variables as Fourier series in the action–angle variables.

The values of the action coordinates label the tori, and the values of the angle coordinate describe the actual dynamics on them. Because the tori are invariant manifolds, the transformed Hamiltonian $H \circ \mathscr{A}^{-1}$ cannot depend on the angles $\boldsymbol{\phi}$, so $H(\boldsymbol{\phi}, \boldsymbol{J}) = H(\boldsymbol{J})$. The equations of motion (1.13) can now be written as

$$\begin{cases} \dot{J}_\alpha = \quad 0, \\ \dot{\phi}^\alpha = -\nu^\alpha(\boldsymbol{J}). \end{cases} \tag{1.26}$$

Here $\nu^\alpha(\boldsymbol{J})$ denotes the frequency that determines the rate at which the αth homology cycle is traversed; the period of the motion on the homology cycle $\gamma_\alpha(\boldsymbol{c})$ is $2\pi/\nu^\alpha$. In conclusion, we see that the symplectic two-form is $\omega = \mathrm{d}\phi^\alpha \wedge \mathrm{d}J_\alpha$, and from the relation $\mathscr{A}_* X_H = X_{H \circ \mathscr{A}^{-1}}$ we obtain the dynamical vector field $\Delta = \nu^\alpha \partial / \partial \phi^\alpha$.

Chapter 2

Numerics

In the past couple of decades the field of numerical integration has seen two important developments: automatic differentiation [61] and geometric numerical integration [24, 66, 96, 115]. For generic initial-value problems automatic differentiation in combination with Taylor's method removes the necessity of calculating series expansions to arbitrary order either by hand or symbolically. We can in principle achieve arbitrarily high precision, because the order of the series expansions determines the order of the integrator, and not the time step used to discretize the expressions, as is the case of schemes based on finite differences. Thus, no truncation error is incurred.

For dynamical systems that can be formulated as Hamiltonian systems there exist so-called geometric numerical integrators. These integrators respect the fundamental geometric structure, that is to say they conserve the symplectic two-form, which underlies the dynamical evolution of the system. It has been common to design such geometric numerical integrators based on either previous knowledge of classical numerical integration algorithms, such as the (partitioned) Runge–Kutta methods, or (approximate) solutions to the Hamilton–Jacobi equation for transformations near the identity. Details, examples, and references can be found in the aforementioned books by Hairer et al. [66], and Leimkuhler and Reich [96].

There is, however, a different approach that bypasses many of the difficulties inherent in the design of higher-order versions of these geometric numerical integrators. It relies on the discretization of the action, from which one derives the numerical algorithms in a straightforward manner [108]. These variational integrators, as they are known throughout the literature, conserve the symplectic structure of the phase flow automatically. Any continuous symmetries present in the original system translate directly to the discretized version, and thus all (equivariant) momentum maps, or conserved quantities, are preserved infinitesimally.

So far, a general "plan to develop higher-order integrators based on [...] more accurate approximations to the action integral" [83] has been lacking, mainly because the manual effort to produce and analyse these higher-order variational integrators outweighs any apparent benefit. Inspired by the work of Gander and Gruntz [52] and more recently Gander [51] to create and analyse numerical algorithms, especially numerical integrators and quadrature rules, an approach to design variational integrators systematically based on approximations to the discrete action by means of a computer algebra system is covered here.

The fundamental concepts from discrete mechanics are reviewed briefly in Section 2.1, before we comment on the application of standard quadrature rules to variational integration in Section 2.2. The specifics of the quadrature formulas integrated in **VarInt** can be found in Appendix A. For more details and the source code of the Maple package named **VarInt**, please consult Appendix B.

2.1 Variational Integration

Consider an autonomous Lagrangian $L\colon \mathbf{T}Q \to \mathbb{R}$. Here and henceforth we assume that the generalized coordinates are at least $\mathcal{C}^2\left([a,b],\mathbb{R}\right)$, where $t \in [a,b]$. The corresponding action functional reads

$$S[L] = \int_a^b L\big(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t)\big)\,\mathrm{d}t, \tag{2.1}$$

from which the famous Euler–Lagrange equations are retrieved upon requiring stationarity of the action functional for fixed endpoints, that is $\delta S[L] = 0$ with $\delta\boldsymbol{q}(a) = \delta\boldsymbol{q}(b) = 0$.

Instead of deriving the Euler–Lagrange equations from the action and then discretizing the equations of motion, a different approach is used in the case of variational integrators. Here we discretize the action first by choosing an

appropriate quadrature formula, and then we derive the discrete version of the Euler–Lagrange equations, which are commonly known as the *discrete* Euler–Lagrange equations. The resulting integration algorithms preserve the differential-geometric structure of these dynamical systems *automatically*. Moreover, the order of the quadrature formula determines the order of the variational integrator [138].

2.1.1 Numerical Integration

To obtain a one-step numerical integration algorithm, we introduce a sequence of times $t_k = hk$, for $k = 0, \ldots, N$, at which the Lagrangian is to be evaluated. Here h denotes a sufficiently small time step. Furthermore, $\boldsymbol{q}_k \approx \boldsymbol{q}(t_k)$ and $\dot{\boldsymbol{q}}_k \approx \dot{\boldsymbol{q}}(t_k)$ for $k = 0, \ldots, N$. Consider the action between two consecutive points in time, say t_k and t_{k+1}. Since for a generic dynamical system with a certain Lagrangian we do not know the functional form of the solutions in advance, we choose an interpolating function, usually a polynomial, in accordance with the quadrature rule on the interval $[t_k, t_{k+1}]$. For a quadrature rule of arbitrary order we evaluate the Lagrangian at $(s+1) \geq 2$ distinct nodes, so that each time step is subdivided into s substeps t_k^i for $i = 0, \ldots, s$. Define $t_k^0 = t_k$ and $t_k^s = t_{k+1}$, and let $t_k^i - t_k^{i-1} = \gamma_i h > 0$ for $i = 1, \ldots, s$, such that $\sum_{i=1}^{s} \gamma_i = 1$. The action becomes a sum of the multi-point discrete Lagrangian L_d, which depends on the time step h:

$$\begin{aligned} S[L] &= \sum_{k=0}^{N-1} \int_{t_k}^{t_{k+1}} L\big(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t)\big)\, \mathrm{d}t \\ &\approx \sum_{k=0}^{N-1} L_\mathrm{d}\big(\boldsymbol{q}_k^0, \boldsymbol{q}_k^1, \ldots, \boldsymbol{q}_k^s\big) \\ &= \sum_{k=0}^{N-1} \sum_{i=1}^{s} L_\mathrm{d}^i\big(\boldsymbol{q}_k^{i-1}, \boldsymbol{q}_k^i\big), \end{aligned}$$

where $L_\mathrm{d}^i \colon Q \times Q \to \mathbb{R}$; it relates the multi-point discrete Lagrangian to its basic components defined on each segment of 'length' $\gamma_i h$. For generic (non-compositional) variational integrators these components depend on all $\boldsymbol{q}_k^i$ for $i = 0, \ldots, s$; a 'decomposition' into the L_d^i is to be understood in these instances as a formal definition only. Notice that the discrete state space $Q \times Q$ contains the same amount of information as the tangent bundle of the configuration manifold, for locally $\mathbf{T}Q \cong Q \times Q$.

The variation of the discrete Lagrangian is

$$\begin{aligned}
\delta S\left[L_{\mathrm{d}}\right] &\approx \sum_{k=0}^{N-1} \delta L_{\mathrm{d}}\left(\boldsymbol{q}_k^0, \boldsymbol{q}_k^1, \ldots, \boldsymbol{q}_k^s\right) \\
&= \sum_{k=0}^{N-1} \sum_{i=0}^{s} \frac{\partial L_{\mathrm{d}}^{[k]}}{\partial \boldsymbol{q}_k^i} \cdot \delta \boldsymbol{q}_k^i \\
&= \sum_{k=0}^{N-1} \left(\frac{\partial L_{\mathrm{d}}^{[k]}}{\partial \boldsymbol{q}_k} \cdot \delta \boldsymbol{q}_k + \sum_{i=1}^{s-1} \frac{\partial L_{\mathrm{d}}^{[k]}}{\partial \boldsymbol{q}_k^i} \cdot \delta \boldsymbol{q}_k^i + \frac{\partial L_{\mathrm{d}}^{[k]}}{\partial \boldsymbol{q}_{k+1}} \cdot \delta \boldsymbol{q}_{k+1} \right) \\
&= \sum_{k=1}^{N-1} \left(\boldsymbol{D}_0 L_{\mathrm{d}}^{[k]} \cdot \delta \boldsymbol{q}_k + \boldsymbol{D}_s L_{\mathrm{d}}^{[k-1]} \cdot \delta \boldsymbol{q}_k \right) + \sum_{k=0}^{N-1} \sum_{i=1}^{s-1} \boldsymbol{D}_i L_{\mathrm{d}}^{[k]} \cdot \delta \boldsymbol{q}_k^i,
\end{aligned}$$

where we have introduced $L_{\mathrm{d}}^{[k]}$ as a shorthand for $L_{\mathrm{d}}\left(\boldsymbol{q}_k^0, \boldsymbol{q}_k^1, \ldots, \boldsymbol{q}_k^s\right)$, and where $\boldsymbol{D}_i$ denotes the derivative with respect to the argument carrying the substep label i, that is $\boldsymbol{D}_i L_{\mathrm{d}}^{[k]} = \partial L_{\mathrm{d}}^{[k]} / \partial \boldsymbol{q}_k^i$. In the last line we have used the fact that $\delta \boldsymbol{q}_0 = \delta \boldsymbol{q}_N = 0$, as the endpoints are assumed fixed. Stationarity of the discrete action for arbitrary variations $\delta \boldsymbol{q}_k^i$ yields the discrete Euler–Lagrange equations:

$$\boldsymbol{D}_0 L_{\mathrm{d}}\left(\boldsymbol{q}_{k+1}^0, \boldsymbol{q}_{k+1}^1, \ldots, \boldsymbol{q}_{k+1}^s\right) + \boldsymbol{D}_s L_{\mathrm{d}}\left(\boldsymbol{q}_k^0, \boldsymbol{q}_k^1, \ldots, \boldsymbol{q}_k^s\right) = 0, \tag{2.3a}$$

$$\boldsymbol{D}_i L_{\mathrm{d}}\left(\boldsymbol{q}_k^0, \boldsymbol{q}_k^1, \ldots, \boldsymbol{q}_k^s\right) = 0, \qquad i = 1, \ldots, s-1. \tag{2.3b}$$

These equations determine the one-step (flow) map of the variational integrator: $\left(\boldsymbol{q}(t_k), \dot{\boldsymbol{q}}(t_k)\right) \mapsto \left(\boldsymbol{q}(t_{k+1}), \dot{\boldsymbol{q}}(t_{k+1})\right)$. These equations can also be written as

$$\boldsymbol{D}_i L_{\mathrm{d}}^i\left(\boldsymbol{q}_k^{i-1}, \boldsymbol{q}_k^i\right) + \boldsymbol{D}_i L_{\mathrm{d}}^{i+1}\left(\boldsymbol{q}_k^i, \boldsymbol{q}_k^{i+1}\right) = 0$$

for each of the components $i = 1, \ldots, s$. Please notice that for $i = s$ the last term on the left-hand side is $\boldsymbol{D}_0 L_{\mathrm{d}}^1\left(\boldsymbol{q}_{k+1}^0, \boldsymbol{q}_{k+1}^1\right) = \boldsymbol{D}_0 L_{\mathrm{d}}^{[k+1]}$ by virtue of the identity $\boldsymbol{q}_k^{i+s} = \boldsymbol{q}_{k+1}^i$.

It is common to write the one-step map in terms of the canonical coordinates and momenta on the cotangent bundle. To do so, we need to find a discrete analogue of the Legendre transformation[1], as given by (1.9). The discretized form

[1] Alternatively, we can use the Hamilton–Pontryagin principle [166, 167]:

$$\delta \left[\int \left(L(\boldsymbol{q}, \boldsymbol{v}) + \langle \boldsymbol{p}, \dot{\boldsymbol{q}} - \boldsymbol{v} \rangle \right) \mathrm{d}t \right] = 0,$$

where $(\boldsymbol{q}, \boldsymbol{v}, \boldsymbol{p}) \in \mathbf{T}\mathcal{Q} \oplus \mathbf{T}^\star\mathcal{Q}$ is the Whitney sum of the tangent and cotangent bundles, which is also known as the fibre product, and discretize the functional subsequently in complete analogy with the discrete variational principle.

of the Legendre transformation involves the endpoints of each time segment, $\mathbb{F}^{\pm}L_{\mathrm{d}}^{i}\colon Q\times Q\to \mathbf{T}^{\star}Q$:

$$\begin{aligned}\mathbb{F}^{+}L_{\mathrm{d}}^{i}\colon (\boldsymbol{q}_k^{i-1},\boldsymbol{q}_k^{i})&\mapsto (\boldsymbol{q}_k^{i},\boldsymbol{p}_k^{i})=\Big(\boldsymbol{q}_k^{i},\boldsymbol{D}_i L_{\mathrm{d}}^{i}(\boldsymbol{q}_k^{i-1},\boldsymbol{q}_k^{i})\Big),\\ \mathbb{F}^{-}L_{\mathrm{d}}^{i}\colon (\boldsymbol{q}_k^{i-1},\boldsymbol{q}_k^{i})&\mapsto (\boldsymbol{q}_k^{i-1},\boldsymbol{p}_k^{i-1})=\Big(\boldsymbol{q}_k^{i-1},-\boldsymbol{D}_{i-1} L_{\mathrm{d}}^{i}(\boldsymbol{q}_k^{i-1},\boldsymbol{q}_k^{i})\Big).\end{aligned}$$

In fact, the discrete Euler–Lagrange equations (2.3) can be written as

$$\mathbb{F}^{+}L_{\mathrm{d}}^{i}(q_k^{i-1},\boldsymbol{q}_k^{i})=\mathbb{F}^{-}L_{\mathrm{d}}^{i+1}(\boldsymbol{q}_k^{i},\boldsymbol{q}_k^{i+1}),$$

which implies that the canonical momenta are unique along any solution. Now the one-step map, written in canonical coordinates and momenta, is

$$\begin{aligned}\boldsymbol{p}_k^{i-1}&=-\boldsymbol{D}_{i-1}L_{\mathrm{d}}^{i}(\boldsymbol{q}_k^{i-1},\boldsymbol{q}_k^{i}),\\ \boldsymbol{p}_k^{i}&=\boldsymbol{D}_i L_{\mathrm{d}}^{i}(\boldsymbol{q}_k^{i-1},\boldsymbol{q}_k^{i}),\end{aligned}$$

for $i=1,\dots,s$, or equivalently,

$$\boldsymbol{p}_k=-\boldsymbol{D}_0 L_{\mathrm{d}}(\boldsymbol{q}_k^{0},\boldsymbol{q}_k^{1},\dots,\boldsymbol{q}_k^{s}), \tag{2.6a}$$

$$\boldsymbol{p}_{k+1}=\boldsymbol{D}_s L_{\mathrm{d}}(\boldsymbol{q}_k^{0},\boldsymbol{q}_k^{1},\dots,\boldsymbol{q}_k^{s}), \tag{2.6b}$$

$$\boldsymbol{D}_i L_{\mathrm{d}}(\boldsymbol{q}_k^{0},\boldsymbol{q}_k^{1},\dots,\boldsymbol{q}_k^{s})=0,\qquad i=1,\dots,s-1, \tag{2.6c}$$

where the last set of $(s-1)$ equations seems 'unaffected' by the Legendre transformation, and remains as in equation (2.3b). The reason for that is quite intuitive yet profound: on each time interval an interpolatory function approximates the Lagrangian function, so that the discrete Lagrangian becomes a piecewise smooth function. The momentum $\boldsymbol{p}_k^{i}$ computed with the discrete Legendre transformation $\mathbb{F}^{-}L_{\mathrm{d}}^{i+1}$ requires the interpolation function on the time segment $[t_k^{i},t_k^{i+1}]$, or data 'from the right' of t_k^{i}, whereas the same momentum calculated from the transformation $\mathbb{F}^{+}L_{\mathrm{d}}^{i}$ uses the interpolation function on $[t_k^{i-1},t_k^{i}]$, or values 'from the left' of t_k^{i}. Obviously, the intermediate momenta ($i=1,\dots,s-1$) are identical, because the interpolating function used is the same, so that its derivatives from the left and right coincide. In principle, the momenta at the endpoints of each time interval need not be related at all, for the interpolating function merely has to be equal in value in order to have a piecewise smooth discrete Lagrangian. However, the principle of stationary action relates the approximate (discrete) Lagrangian function to the (approximated) integral curves of the dynamical system, which in turn relates these momenta by means of the discrete Euler–Lagrange equations. Therefore, the momenta are

unique along any trajectory, and equations (2.3b) and (2.6c) are identical in both representations.

All integrators obtained in this way are structure-preserving, that is to say they preserve the symplectic structure of the flow. Indeed,

$$\begin{aligned}
\mathrm{d}L_{\mathrm{d}}^{[k]} &= \sum_{i=0}^{s} \boldsymbol{D}_i L_{\mathrm{d}}^{[k]} \cdot \mathrm{d}\boldsymbol{q}_k^i \\
&= \boldsymbol{D}_0 L_{\mathrm{d}}^{[k]} \cdot \mathrm{d}\boldsymbol{q}_k + \boldsymbol{D}_s L_{\mathrm{d}}^{[k]} \cdot \mathrm{d}\boldsymbol{q}_{k+1} + \sum_{i=1}^{s-1} \boldsymbol{D}_i L_{\mathrm{d}}^{[k]} \cdot \mathrm{d}\boldsymbol{q}_k^i \\
&= \boldsymbol{D}_0 L_{\mathrm{d}}^{[k]} \cdot \mathrm{d}\boldsymbol{q}_k - \boldsymbol{D}_0 L_{\mathrm{d}}^{[k+1]} \cdot \mathrm{d}\boldsymbol{q}_{k+1} \\
&= -\boldsymbol{p}_k \cdot \mathrm{d}\boldsymbol{q}_k + \boldsymbol{p}_{k+1} \cdot \mathrm{d}\boldsymbol{q}_{k+1},
\end{aligned}$$

from which it follows that $\mathrm{d}\boldsymbol{q}_k \wedge \mathrm{d}\boldsymbol{p}_k = \mathrm{d}\boldsymbol{q}_{k+1} \wedge \mathrm{d}\boldsymbol{p}_{k+1}$. The (discrete) Lagrangian flow conserves the (discrete) symplectic form as well as any momentum maps associated with (infinitesimal) invariances of the (discrete) action under symmetry operations, as shown by Marsden and West [108]. Please recall from Chapter 1 that the momentum map is to be calculated from $\mu \cdot \xi = \iota_{X^\xi}\theta$ for any group G that acts by the cotangent lift on $\mathbf{T}^\star Q$. In local coordinates $(\boldsymbol{q}, \boldsymbol{p}) \in \mathbf{T}^\star Q$ we have that $\mu \cdot \xi = \langle \boldsymbol{p}, \boldsymbol{X}^\xi(\boldsymbol{q}) \rangle$, so that the Lagrangian momentum map μ_L can be calculated from $\mu_L = (\mathbb{F}L)^* \mu$:

$$\mu_L \cdot \xi = \left\langle \frac{\partial L}{\partial \dot{\boldsymbol{q}}}, \boldsymbol{X}^\xi(\boldsymbol{q}) \right\rangle.$$

In the discrete framework we end up at each substep $i = 1, \ldots, s$ with

$$\begin{aligned}
\mu_{L_{\mathrm{d}}}^{i+} \cdot \xi &= \left\langle \boldsymbol{D}_i L_{\mathrm{d}}^i(\boldsymbol{q}_k^{i-1}, \boldsymbol{q}_k^i), \boldsymbol{X}^\xi(\boldsymbol{q}_k^i) \right\rangle, \\
\mu_{L_{\mathrm{d}}}^{i-1-} \cdot \xi &= \left\langle -\boldsymbol{D}_{i-1} L_{\mathrm{d}}^i(\boldsymbol{q}_k^{i-1}, \boldsymbol{q}_k^i), \boldsymbol{X}^\xi(\boldsymbol{q}_k^{i-1}) \right\rangle.
\end{aligned}$$

Now suppose that $L_{\mathrm{d}}(\boldsymbol{q}_k^0, \boldsymbol{q}_k^1, \ldots, \boldsymbol{q}_k^s)$ is invariant (for $k = 0, \ldots, N-1$) under the lifted action of $\psi_g \colon Q \to Q$ for $g = \exp(t\xi)$ with $t \in \mathbb{R}$, which means that

$$\sum_{i=1}^{s} L_{\mathrm{d}}^i(\psi_g(\boldsymbol{q}_k^{i-1}), \psi_g(\boldsymbol{q}_k^i)) = \sum_{i=1}^{s} L_{\mathrm{d}}^i(\boldsymbol{q}_k^{i-1}, \boldsymbol{q}_k^i).$$

By taking the derivative with respect to t and evaluating at $t = 0$, we find that

$$\begin{aligned} 0 &= \sum_{i=1}^{s} \boldsymbol{D}_{i-1} L_{\mathrm{d}}^{i}(\boldsymbol{q}_k^{i-1}, \boldsymbol{q}_k^{i}) \cdot \boldsymbol{X}^{\xi}(\boldsymbol{q}_k^{i-1}) + \boldsymbol{D}_i L_{\mathrm{d}}^{i}(\boldsymbol{q}_k^{i-1}, \boldsymbol{q}_k^{i}) \cdot \boldsymbol{X}^{\xi}(\boldsymbol{q}_k^{i}) \\ &= \sum_{i=1}^{s} \left(-\mu_{L_{\mathrm{d}}}^{i-1\,-} + \mu_{L_{\mathrm{d}}}^{i\,+} \right) \cdot \xi, \end{aligned}$$

from which it follows that $\mu_{L_{\mathrm{d}}}^{0\,-} = \mu_{L_{\mathrm{d}}}^{s\,+}$. All in all, we arrive at the discrete version of Noether's theorem:

THEOREM 8 (Marsden–West). *If the discrete Lagrangian L_d is (infinitesimally) invariant under the one-parameter group of transformations $\{t \in \mathbb{R} \mid \exp(t\xi)\}$ for all $\xi \in \mathfrak{g}$, then the associated discrete (Lagrangian) momentum map μ_{L_d} is a constant of motion.*

Hence, variational integrators conserve the symplectic two-form and momentum maps irrespective of the time step h. Obviously, the computed numerical solutions are only accurate whenever the time step is sufficiently small, which depends on the particulars of the problem under consideration. In addition, we note that invariance of the Lagrangian L under the lifted action of a group G does not necessarily imply (infinitesimal) invariance of its discrete counterpart L_{d} [see e.g. 66, p. 211].

So far the description has been generic, in the sense that we do not have any additional information available apart from the Lagrangian and an as yet undefined quadrature rule to obtain numerical approximations. However, for almost-integrable systems, where we indeed have additional knowledge, namely the solutions of the integrable system themselves, we can insert that information into the discrete action to obtain more accurate solutions, as described by Farr [45].

2.1.2 COMPOSITION

For any one-step map $\Phi\colon \mathbb{R} \times \mathbf{T}^\star\mathcal{Q} \to \mathbf{T}^\star\mathcal{Q}$ of order p, we define the adjoint $\Phi^*\colon \mathbb{R} \times \mathbf{T}^\star\mathcal{Q} \to \mathbf{T}^\star\mathcal{Q}$ by

$$\Phi_h^* \circ \Phi_{-h} = \mathrm{Id}, \tag{2.8}$$

and it is of order p too. An integrator is called self-adjoint, or symmetric, if $\Phi_h^* = \Phi_h$. As a consequence, the order of any self-adjoint method is even.

It is possible to construct higher-order integrators from lower-order ones by composition in s stages. Consider a method Φ_h of order p and its adjoint Φ_h^*, together with $\{i = 1, \ldots, s \mid \alpha_i \in \mathbb{R}\}$ and $\{i = 1, \ldots, s \mid \beta_i \in \mathbb{R}\}$, such that

$$\sum_{i=1}^{s} \alpha_i + \beta_i = 1,$$
$$\sum_{i=1}^{s} \alpha_i^{p+1} + (-1)^p \beta_i^{p+1} = 0.$$

The map $\Psi_h = \Phi_{\alpha_s h} \circ \Phi^*_{\beta_s h} \circ \ldots \circ \Phi_{\alpha_1 h} \circ \Phi^*_{\beta_1 h}$ is at least of order $p+1$. For $\alpha_{s-i+1} = \beta_i$, with $i = 1, \ldots, s$, the composition is of course symmetric. In particular, the map $\Psi_h = \Phi_{h/2} \circ \Phi^*_{h/2}$ is symmetric, and it can be used to transform a first-order algorithm into a symmetric, second-order one. Furthermore, for any symmetric method Φ_h and $\{i = 1, \ldots, s \mid \gamma_i \in \mathbb{R}\}$ satisfying

$$\sum_{i=1}^{s} \gamma_i = 1,$$
$$\sum_{i=1}^{s} \gamma_i^{p+1} = 0,$$

the symmetric composition

$$\Psi_h = \Phi_{\gamma_s h} \circ \Phi_{\gamma_2 h} \circ \ldots \circ \Phi_{\gamma_1 h}, \tag{2.11}$$

where $\gamma_{s-i+1} = \gamma_i$ for $i = 1, \ldots, s$, is symmetric and of order at least $p+1$. Therefore, we can construct a higher-order (variational) integrator from a first-order one.

There is a systematic way due to Yoshida [165] and Suzuki [152] to obtain the composition coefficients for separable systems. The method relies on a re-interpretation of the Lie derivative as an operator, as evinced suggestively in equation (1.18) for functions on the phase space. We introduce an operator

$$\mathcal{L}_H = \{\cdot, H\},$$

which is commonly referred to as the Liouvillian, or Liouville operator[2]; it is to act on functions on the phase space, in particular canonical coordinates and

[2]The operatorial framework of classical mechanics can be viewed as a 'reverse-engineered' quantum-mechanical structure rooted in operators on Hilbert spaces, as formulated originally by Koopman [90] and von Neumann [131]. From that perspective the Liouvillian is often written as $\mathcal{L}_H \mapsto \mathrm{i}\mathcal{L}_H$, where the imaginary unit makes it a unitary operator. We can even retrofit classical mechanics with some of the mathematical intricacies of quantum mechanics and quantum field theory [39, 40, 57–59].

momenta. With it equation (1.16) can be written as

$$\dot{\boldsymbol{\eta}} = \mathcal{L}_H \boldsymbol{\eta},$$

which has the formal solution $\boldsymbol{\eta}(t) = \exp(t\mathcal{L}_H)\boldsymbol{\eta}(0)$. The exponential operator is sometimes referred to as the (classical) propagator.

For separable dynamical systems the Hamiltonian is $H(\boldsymbol{q}, \boldsymbol{p}) = T(\boldsymbol{p}) + V(\boldsymbol{q})$, so that the Liouvillian decomposes too: $\mathcal{L}_H = \mathcal{L}_T + \mathcal{L}_V$. The exponential of the sum of Liouvillian operators can be written as a product of the individual components of the Liouvillian with the Baker–Campbell–Hausdorff formula. The truncation of the factorized propagator determines the order of the composition algorithm. Obviously, we can decompose dynamical systems in many different ways[3] or we can arrange the terms in the operator product differently[4]. More details and the values of the composition coefficients for different orders can be found in Hairer et al. [66], pp. 150–158.

In creating higher-order integrators by means of composition, we have the freedom to choose the number of stages, and to some extent we can 'tune' the composition coefficients. The 'optimality' of these variables with respect to the performance of the algorithm depends on the specifics of the problem under consideration.

We have assumed the coefficients to be real, although there is in principle nothing that prevents us from using complex time steps, as proposed by Chambers [30]. It is important to note that composition methods use both forward and backward time steps, to which we shall come in more detail in the next subsection.

2.1.3 Quadrature or Composition?

At this point, the astute reader might have asked what the benefit is of variational integrators based on higher-order quadrature formulas. Indeed, higher-order variational integrators are generally implicit, whereas the concatenation of explicit lower-order integrators remains explicit at all orders, so that in terms of computational time the latter category would most likely outperform the former one.

For conservative dynamical systems the use of variational integrators can indeed be argued, though it should be noted that the variational formalism arises

[3] For instance, based on the different time scales of the individual components, or (non-)linearity or (non-)integrability of the separate terms, to name but a few.

[4] A propagator decomposition with identical time steps is sometimes referred to as a Lie–Trotter decomposition, whereas a symmetric product of operators is known as a Strang–Marchuk splitting.

naturally throughout mathematical physics, and it can of course be extended; we refer to Junge et al. [82], Kharevych et al. [86], Lew et al. [98, 99], and Marsden and West [108] for a few possibilities. The main advantage and actual utility of the variational construction of numerical integration algorithms lies in the fact that non-conservative forces can easily be included in a consistent way. N-body simulations in atomic and molecular physics, astrophysics, and chemistry are excellent candidates for these non-conservative variational integrators, as these simulations often become unstable under time reversal [46], so that higher-order geometric numerical integrators based on the symmetric composition of lower-order ones are not viable alternatives.

A force is a fibre-preserving map over the identity[5] $\boldsymbol{F}\colon \mathbf{T}\mathcal{Q} \to \mathbf{T}^\star\mathcal{Q}$, which reads $\boldsymbol{F}\colon (\boldsymbol{q}, \dot{\boldsymbol{q}}) \mapsto \big(\boldsymbol{q}, \boldsymbol{F}(\boldsymbol{q}, \dot{\boldsymbol{q}})\big)$ in coordinates. In order to include these non-conservative forces in the variational framework, we merely have to replace Hamilton's principle $\delta S\,[L] = 0$ by the so-called Lagrange–d'Alembert principle:

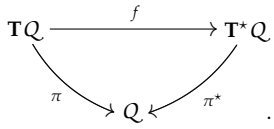
$$\sum_{k=0}^{N-1} \left[\delta \int_{t_k}^{t_{k+1}} L\big(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t)\big)\,\mathrm{d}t + \int_{t_k}^{t_{k+1}} \boldsymbol{F}\left(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t)\right) \cdot \delta\boldsymbol{q}(t)\,\mathrm{d}t \right] = 0. \tag{2.12}$$

As before, all integrals are approximated by a quadrature rule from $t \in [t_k, t_{k+1}]$, so that the Lagrange–d'Alembert principle becomes

$$\sum_{k=0}^{N-1} \left[\delta \underbrace{\sum_{i=0}^{s} L\Big(\boldsymbol{q}_{\mathrm{int}}\big(t_k^i\big), \dot{\boldsymbol{q}}_{\mathrm{int}}\big(t_k^i\big)\Big)}_{L_{\mathrm{d}}\left(\boldsymbol{q}_k^0, \boldsymbol{q}_k^1, \ldots, \boldsymbol{q}_k^s\right)} + \sum_{i=0}^{s} \underbrace{\boldsymbol{F}\Big(\boldsymbol{q}_{\mathrm{int}}\big(t_k^i\big), \dot{\boldsymbol{q}}_{\mathrm{int}}\big(t_k^i\big)\Big) \cdot \delta\boldsymbol{q}_{\mathrm{int}}\left(t_k^i\right)}_{\boldsymbol{f}_k^i \cdot \delta\boldsymbol{q}_k^i = \boldsymbol{f}_k^i\left(\boldsymbol{q}_k^0, \boldsymbol{q}_k^1, \ldots, \boldsymbol{q}_k^s\right) \cdot \delta\boldsymbol{q}_k^i} \right] = 0, \tag{2.13}$$

where we have written the discrete Lagrangian in terms of the interpolatory approximation of $\boldsymbol{q}(t)$ for $t \in [t_k, t_{k+1}]$: $\boldsymbol{q}_{\mathrm{int}}\big(t_k^i\big) = \boldsymbol{q}_{\mathrm{int}}\big(\boldsymbol{q}_k^0, \ldots, \boldsymbol{q}_k^s; t_k^i\big)$. It is worth

[5] A diffeomorphism $f\colon \mathbf{T}\mathcal{Q} \to \mathbf{T}^\star\mathcal{Q}$ is called a fibre-preserving map over the identity, or bundle map from $\mathbf{T}\mathcal{Q}$ to $\mathbf{T}^\star\mathcal{Q}$ over $\mathcal{Q}$, if $\pi^\star \circ f = \pi$, where π ($\pi^\star$) is the natural (canonical) projection, which takes (co)tangent vectors to the points at which they are attached:



mentioning that

$$\delta \boldsymbol{q}_{\text{int}}(t_k^i) = \sum_{j=0}^{s} \frac{\partial \boldsymbol{q}_{\text{int}}(t_k^i)}{\partial \boldsymbol{q}_k^j} \cdot \delta \boldsymbol{q}_k^j,$$

and that generally $\boldsymbol{q}_{\text{int}}(\boldsymbol{q}_k^0, \dots, \boldsymbol{q}_k^s; t_k^i) \neq \boldsymbol{q}_k^i$. Therefore,

$$\boldsymbol{f}_k^i(\boldsymbol{q}_k^0, \boldsymbol{q}_k^1, \dots, \boldsymbol{q}_k^s) = \sum_{j=0}^{s} \boldsymbol{F}\left(\boldsymbol{q}_{\text{int}}(t_k^j), \dot{\boldsymbol{q}}_{\text{int}}(t_k^j)\right) \cdot \frac{\partial \boldsymbol{q}_{\text{int}}(t_k^j)}{\partial \boldsymbol{q}_k^i}.$$

In a manner similar to the derivation of the discrete Euler–Lagrange equations (2.3), the *forced* discrete Euler–Lagrange equations can be shown to be

$$\boldsymbol{D}_0 L_{\text{d}}^{[k+1]} + \boldsymbol{f}_{k+1}^0 + \boldsymbol{D}_s L_{\text{d}}^{[k]} + \boldsymbol{f}_k^s = 0, \tag{2.14a}$$

$$\boldsymbol{D}_i L_{\text{d}}^{[k]} + \boldsymbol{f}_k^i = 0, \qquad i = 1, \dots, s-1. \tag{2.14b}$$

Again, it is possible to write the forced discrete Euler–Lagrange equations in terms of the canonical coordinates and momenta instead. To that end, we define the left and right discrete forces $\boldsymbol{F}_{\text{d}}^{i\pm} \colon Q \times Q \to \mathbb{R}$ respectively as $\boldsymbol{F}_{\text{d}}^{i-}(\boldsymbol{q}_k^{i-1}, \boldsymbol{q}_k^i)$ and $\boldsymbol{F}_{\text{d}}^{i+}(\boldsymbol{q}_k^{i-1}, \boldsymbol{q}_k^i)$, and we find that $\boldsymbol{f}_k^0 = \boldsymbol{F}_{\text{d}}^{1-}(\boldsymbol{q}_k^0, \boldsymbol{q}_k^1)$, $\boldsymbol{f}_k^s = \boldsymbol{F}_{\text{d}}^{s+}(\boldsymbol{q}_k^{s-1}, \boldsymbol{q}_k^s)$, and $\boldsymbol{f}_k^i = \boldsymbol{F}_{\text{d}}^{i+}(\boldsymbol{q}_k^{i-1}, \boldsymbol{q}_k^i) + \boldsymbol{F}_{\text{d}}^{i+1-}(\boldsymbol{q}_k^i, \boldsymbol{q}_k^{i+1})$ for $i = 1, \dots, s-1$. These, in turn, imply that

$$\int_{t_k}^{t_{k+1}} \boldsymbol{F}(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t)) \cdot \delta \boldsymbol{q}(t) \, \mathrm{d}t \approx \sum_{i=1}^{s} \left[\boldsymbol{F}_{\text{d}}^{i-}(\boldsymbol{q}_k^{i-1}, \boldsymbol{q}_k^i) \cdot \delta \boldsymbol{q}_k^{i-1} + \boldsymbol{F}_{\text{d}}^{i+}(\boldsymbol{q}_k^{i-1}, \boldsymbol{q}_k^i) \cdot \delta \boldsymbol{q}_k^i \right].$$

Consequently, we find that the equations (2.14) can be written as

$$\boldsymbol{D}_i L_{\text{d}}^i(\boldsymbol{q}_k^{i-1}, \boldsymbol{q}_k^i) + \boldsymbol{F}_{\text{d}}^{i+}(\boldsymbol{q}_k^{i-1}, \boldsymbol{q}_k^i) + \boldsymbol{D}_i L_{\text{d}}^{i+1}(\boldsymbol{q}_k^i, \boldsymbol{q}_k^{i+1}) + \boldsymbol{F}_{\text{d}}^{i+1-}(\boldsymbol{q}_k^i, \boldsymbol{q}_k^{i+1}) = 0$$

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

The appropriate discrete Legendre transformations for forced dynamical systems are

$$\begin{aligned}
\mathbb{F}^{f+} L_{\text{d}}^i &\colon (\boldsymbol{q}_k^{i-1}, \boldsymbol{q}_k^i) \mapsto (\boldsymbol{q}_k^i, \boldsymbol{p}_k^i) = \left(\boldsymbol{q}_k^i, \boldsymbol{D}_i L_{\text{d}}^i(\boldsymbol{q}_k^{i-1}, \boldsymbol{q}_k^i) + \boldsymbol{F}_{\text{d}}^{i+}\right), \\
\mathbb{F}^{f-} L_{\text{d}}^i &\colon (\boldsymbol{q}_k^{i-1}, \boldsymbol{q}_k^i) \mapsto (\boldsymbol{q}_k^{i-1}, \boldsymbol{p}_k^{i-1}) = \left(\boldsymbol{q}_k^{i-1}, -\boldsymbol{D}_{i-1} L_{\text{d}}^i(\boldsymbol{q}_k^{i-1}, \boldsymbol{q}_k^i) - \boldsymbol{F}_{\text{d}}^{i-}\right),
\end{aligned}$$

so that $\mathbb{F}^{f+} L_{\text{d}}^i(\boldsymbol{q}_k^{i-1}, \boldsymbol{q}_k^i) = \mathbb{F}^{f-} L_{\text{d}}^{i+1}(\boldsymbol{q}_k^i, \boldsymbol{q}_k^{i+1})$ with $i = 1, \dots, s$ as before. Now it is without any effort that we can derive the forced discrete Euler–Lagrange

equations on the cotangent bundle:

$$\boldsymbol{p}_k = -\boldsymbol{D}_0 L_\mathrm{d}\left(\boldsymbol{q}_k^0, \boldsymbol{q}_k^1, \ldots, \boldsymbol{q}_k^s\right) - \boldsymbol{f}_k^0, \tag{2.16a}$$

$$\boldsymbol{p}_{k+1} = \boldsymbol{D}_s L_\mathrm{d}\left(\boldsymbol{q}_k^0, \boldsymbol{q}_k^1, \ldots, \boldsymbol{q}_k^s\right) + \boldsymbol{f}_k^s, \tag{2.16b}$$

$$\boldsymbol{D}_i L_\mathrm{d}\left(\boldsymbol{q}_k^0, \boldsymbol{q}_k^1, \ldots, \boldsymbol{q}_k^s\right) + \boldsymbol{f}_k^i = 0, \qquad i = 1, \ldots, s-1. \tag{2.16c}$$

The functions $\boldsymbol{f}_k^i$ can be computed with the **VarInt**, so that we can easily generate higher-order variational integrators that include non-conservative forces in a 'variational' manner, that is in a way that respects the fundamental differential-geometric properties of any dynamical system.

2.2 Quadrature Formulas

In principle any integration formula can be used to approximate the discrete action, and thus generate a variational integrator, although some cautionary remarks are in order. First, autonomous dynamical systems, which are the ones considered here, are time-reversible, so in order to create variational integrators that respect this discrete symmetry, it is necessary to consider quadrature formulas that are 'symmetric', which means that the placement of the (interpolation) nodes must be symmetrical with respect to the midpoint of each time interval. This rules out the use of open Newton–Cotes and Radau integration formulas, for instance. Second, it is difficult to imagine how quadrature rules based on non-polynomial interpolation should be implemented for generic dynamical systems. Numerical integration based on rational functions [53], for example, either require the location of the poles in advance, or the integration weights cannot be computed explicitly for generic integrands. In the discrete formalism described so far the former requires the knowledge of contingent singularities of the (discrete) Lagrangian as functions of time, whereas the latter implies that these quadrature rules would have only limited applicability, if at all. Third, numerical integration methods that involve the derivatives of the integrand with respect to the independent variable, that is Turán [54, pp. 42–43] and Birkhoff quadrature formulas [103, Chap. 10], can be used as well, but they call for the time derivatives of the Lagrangian along the (numerical) solutions; it is essentially possible to compute these using either finite differences or automatic differentiation techniques, although that may be difficult and problem-dependent in practice.

Here we only consider time-independent Lagrangians. Time-dependent dynamical systems can be analysed similarly in the extended phase space formalism of Section 1.3.1. The Maple codes can be adapted rather easily to

accommodate non-autonomous Lagrangians, though we do not dwell on the details.

We wish to mention some notational issues. Because only autonomous dynamical systems are considered, it suffices to define the one-step discrete action on the interval $[0, h]$, where $h > 0$ is the time step. Hence, to make the notation somewhat more manageable in Maple, we have removed the 'step index' $k = 0, \ldots, N$ from all variables, as in actual implementations of these variational algorithms the step index is redundant, in the sense that it is translated to a function that returns the updated values of all variables. However, all variables still carry one index, namely the 'substep index' $i = 0, \ldots, s$. Henceforth we write the number of nodes as $n = s + 1$.

In order to transform any basic quadrature rule to an approximation of the action functional, it is important to notice that the independent variable is time, and that the coordinates and their derivatives with respect to time are approximated by polynomials of order $(n - 1)$. It is possible to design variational integrators based on non-polynomially fitted quadrature rules with the auxiliary module CreateVarInt. An example is given at the end of Section 2.3. Nevertheless, the order of *any* variational integrator is determined entirely by the order of the approximation of the discrete action.

2.3 Examples

The symplectic partitioned Runge–Kutta methods form a well-known class of variational integrators for conservative dynamical systems. For non-conservative systems probably the best studied example is the symplectic Newmark algorithm, as described in Kane et al. [83]. Beyond these the number of variational integrators is limited, mainly because the manual effort to generate these (higher-order) variational integrators is substantial.

VarInt is a library that enables anyone with a Maple distribution to create and analyse new variational integrators with ease. The module **VarInt** has four main procedures: VarInt, CreateVarInt, ExtractAlgorithm, and IntegrateSystem, which provides basic functionality for the prompt numerical analysis of one-dimensional problems. VarInt computes the (forced) discrete Euler–Lagrange equations. In order to obtain an actual recipe that allows us to compute the discrete flow efficiently, we have to manipulate the expressions returned by VarInt, which depends highly on the functional form of the Lagrangian, and is

hence best done interactively. For separable Lagrangians

$$L(\boldsymbol{q}, \dot{\boldsymbol{q}}) = T(\dot{\boldsymbol{q}}) - V(\boldsymbol{q}), \tag{2.17}$$

with $T\colon \mathbf{T}Q \to \mathbb{R}$ a quadratic kinetic energy function and $V\colon Q \to \mathbb{R}$ the potential energy, an ancillary procedure ExtractAlgorithm is included in **VarInt**. It aids in the extraction of such a one-step map, even for dynamical systems with generic non-conservative forces. It greatly enhances the development of variational integrators for non-conservative forces up to arbitrary order, which has only been touched upon scantily thus far.

The module CreateVarInt is similar in design as VarInt with the significant difference that the approximation to the discrete action can be supplied manually by specifying the nodes, the weights of the numerical integration formula, and the interpolation procedure, which is polynomial by default. CreateVarInt therefore extends VarInt by allowing new quadrature rules to be defined and the creation of non-polynomially fitted variational integration algorithms.

To see the full scope of **VarInt**, we shall first look at simple problems. Consider a two-point Newton–Cotes approximation of the action and a non-conservative Rayleigh force. We can obtain the discrete Euler–Lagrange equations with VarInt as follows:

```
1 > restart;                                          #clear memory
2 > with(VarInt):                                     #load VarInt
3 > dEL1:=VarInt(2,L,F,NewtonCotes,p,q,h);            #obtain dEL equations.
```

Since we have not yet specified the functional forms of the Lagrangian and the Rayleigh force, the expressions Maple returns are fully implicit. To obtain a more applicable representation of the two-point variational Newton–Cotes integrator, we define a separable Lagrangian (2.17), and extract the algorithm:

```
4 > L:=(q,Dq)->1/2*M*Dq^2-V(q):                       #define Lagrangian
5 > dEL2:=VarInt(2,L,F,NewtonCotes,p,q,h):            #obtain dEL equations
6 > ExtractAlgorithm(dEL2,p,q,V,F);                   #obtain algorithm.
```

Here M is the mass. In the case of vectorial coordinates and momenta, M has to be interpreted as the mass matrix. The one-step map $(\boldsymbol{q}_0, \boldsymbol{p}_0) \mapsto (\boldsymbol{q}_1, \boldsymbol{p}_1)$ reads

$$\boldsymbol{q}_1 = \boldsymbol{q}_0 + h\frac{\boldsymbol{p}_0}{M} - \frac{h^2}{2M}\left[\nabla V(\boldsymbol{q}_0) - \boldsymbol{F}\left(\boldsymbol{q}_0, \frac{\boldsymbol{q}_1 - \boldsymbol{q}_0}{h}\right)\right],$$

$$\boldsymbol{p}_1 = \boldsymbol{p}_0 - \frac{h}{2}\left[\nabla V(\boldsymbol{q}_0) + \nabla V(\boldsymbol{q}_1) - \boldsymbol{F}\left(\boldsymbol{q}_0, \frac{\boldsymbol{q}_1 - \boldsymbol{q}_0}{h}\right) - \boldsymbol{F}\left(\boldsymbol{q}_1, \frac{\boldsymbol{q}_1 - \boldsymbol{q}_0}{h}\right)\right],$$

The algorithm is implicit for generic $\boldsymbol{F}$. For conservative dynamical systems the algorithm reduces to the the famous second-order Störmer–Verlet[6] algorithm, which is sometimes referred to as the leapfrog:

$$\boldsymbol{q}_1 = \boldsymbol{q}_0 + h\frac{\boldsymbol{p}_0}{M} - \frac{h^2}{2M}\nabla V(\boldsymbol{q}_0), \tag{2.19a}$$

$$\boldsymbol{p}_1 = \boldsymbol{p}_0 - \frac{h}{2}\left[\nabla V(\boldsymbol{q}_0) + \nabla V(\boldsymbol{q}_1)\right]. \tag{2.19b}$$

It is clearly explicit. It can be obtained in the active Maple worksheet in several ways:

```
> eval(%,F=0);                                  #alternative 1
> F:=(q,Dq)->0:                                 #alternative 2
> dEL3:=VarInt(2,L,F,NewtonCotes,p,q,h);        #alternative 2 (cont'd)
> ExtractAlgorithm(dEL3,p,q,V,F);               #alternative 2 (cont'd)
> dEL4:=VarInt(2,L,0,NewtonCotes,p,q,h);        #alternative 3
> ExtractAlgorithm(dEL3,p,q,V,F);               #alternative 3 (cont'd).
```

As it happens, the Newton–Cotes, Romberg, Gauss–Lobatto, and Clenshaw–Curtis quadrature rules with two nodes are identical, so that their variational integrators are the same.

Incidentally, for three node points the Newton–Cotes, Gauss–Lobatto, and Clenshaw–Curtis quadrature formulas coincide; the integration rule is simply Simpson's formula. The Maple code

```
> dEL5:=VarInt(3,L,0,GaussLobatto,p,q,h);       #obtain dEL equations
> ExtractAlgorithm(dEL5,p,q,V,F);               #obtain algorithm
```

results in the fourth-order algorithm for conservative dynamical systems reported by Farr and Bertschinger [46]:

$$\boldsymbol{q}_1 = \boldsymbol{q}_0 + \frac{h}{2}\frac{\boldsymbol{p}_0}{M} - \frac{h^2}{24M}\left[2\nabla V(\boldsymbol{q}_0) + \nabla V(\boldsymbol{q}_1)\right], \tag{2.20a}$$

$$\boldsymbol{q}_2 = \boldsymbol{q}_0 + h\frac{\boldsymbol{p}_0}{M} - \frac{h^2}{6M}\left[\nabla V(\boldsymbol{q}_0) + 2\nabla V(\boldsymbol{q}_1)\right], \tag{2.20b}$$

$$\boldsymbol{p}_2 = \boldsymbol{p}_0 - \frac{h}{6}\left[\nabla V(\boldsymbol{q}_0) + 4\nabla V(\boldsymbol{q}_1) + \nabla V(\boldsymbol{q}_2)\right]. \tag{2.20c}$$

Equation (2.20a) has to be solved iteratively for generic (non-linear) potentials. Equations (2.20b) and (2.20c) are clearly explicit.

[6] Strictly speaking, the name 'Störmer' is written 'Størmer' in Norwegian, but we adhere to the established orthography to avoid unnecessary confusion.

For four nodes these three families of quadrature rules lead to different variational integrators. The variational Newton–Cotes integrator, which is based on Simpson's $\frac{3}{8}$ rule, is easily found to be

$$\begin{aligned}
\boldsymbol{q}_1 &= \boldsymbol{q}_0 + \frac{h}{3}\frac{\boldsymbol{p}_0}{M} - \frac{h^2}{648M}\left[27\nabla V(\boldsymbol{q}_0) + 14\nabla V(\boldsymbol{q}_1) - 5\nabla V(\boldsymbol{q}_2)\right], \\
\boldsymbol{q}_2 &= \boldsymbol{q}_0 + \frac{2h}{3}\frac{\boldsymbol{p}_0}{M} - \frac{h^2}{324M}\left[27\nabla V(\boldsymbol{q}_0) + 38\nabla V(\boldsymbol{q}_1) + 7\nabla V(\boldsymbol{q}_2)\right], \\
\boldsymbol{q}_3 &= \boldsymbol{q}_0 + h\frac{\boldsymbol{p}_0}{M} - \frac{h^2}{8M}\left[\nabla V(\boldsymbol{q}_0) + 2\nabla V(\boldsymbol{q}_1) + \nabla V(\boldsymbol{q}_2)\right], \\
\boldsymbol{p}_3 &= \boldsymbol{p}_0 - \frac{h}{8}\left[\nabla V(\boldsymbol{q}_0) + 3\nabla V(\boldsymbol{q}_1) + 3\nabla V(\boldsymbol{q}_2) + \nabla V(\boldsymbol{q}_3)\right].
\end{aligned}$$

Similarly, the variational Clenshaw–Curtis integrator with four nodes is

$$\begin{aligned}
\boldsymbol{q}_1 &= \boldsymbol{q}_0 + \frac{3h}{14}\frac{\boldsymbol{p}_0}{M} - \frac{h^2}{13440M}\left[320\nabla V(\boldsymbol{q}_0) + 259\nabla V(\boldsymbol{q}_1) - 21\nabla V(\boldsymbol{q}_2)\right], \\
\boldsymbol{q}_2 &= \boldsymbol{q}_0 + \frac{6h}{7}\frac{\boldsymbol{p}_0}{M} - \frac{h^2}{13440M}\left[1280\nabla V(\boldsymbol{q}_0) + 3339\nabla V(\boldsymbol{q}_1) + 259\nabla V(\boldsymbol{q}_2)\right], \\
\boldsymbol{q}_3 &= \boldsymbol{q}_0 + \frac{15h}{14}\frac{\boldsymbol{p}_0}{M} - \frac{h^2}{84M}\left[10\nabla V(\boldsymbol{q}_0) + 28\nabla V(\boldsymbol{q}_1) + 7\nabla V(\boldsymbol{q}_2)\right], \\
\boldsymbol{p}_3 &= \boldsymbol{p}_0 - \frac{h}{18}\left[2\nabla V(\boldsymbol{q}_0) + 7\nabla V(\boldsymbol{q}_1) + 7\nabla V(\boldsymbol{q}_2) + 2\nabla V(\boldsymbol{q}_3)\right].
\end{aligned}$$

The variational integrators that derive from the Gauss–Lobatto quadrature rules correspond to the well-known Lobatto IIIA/IIIB algorithms, and their forms can be found in the literature.

All variational Gauss–Legendre integrators are equal to the Gauss collocation methods. As an example, the Gauss–Legendre variational integrator with two nodes is easily found to be

$$\boldsymbol{q}_1 = \boldsymbol{q}_0 + h\frac{\boldsymbol{p}_0}{m} - \frac{h^2}{12m}\left[c_-\nabla V\left(\boldsymbol{q}_+\right) + c_+\nabla V\left(\boldsymbol{q}_-\right)\right], \tag{2.23a}$$

$$\boldsymbol{p}_1 = \boldsymbol{p}_0 - \frac{h}{2}\left[\nabla V\left(\boldsymbol{q}_+\right) + \nabla V\left(\boldsymbol{q}_-\right)\right], \tag{2.23b}$$

where we have defined $\boldsymbol{q}_\pm = \frac{1}{2}\left(\boldsymbol{q}_0 + \boldsymbol{q}_1\right) \pm \frac{1}{6}\sqrt{3}\left(\boldsymbol{q}_0 - \boldsymbol{q}_1\right)$, and $c_\pm = 3 \pm \sqrt{3}$. The Gauss–Legendre and the Chebyshev quadrature formulas with two nodes happen to coincide, so that their variational integrators (2.23) are identical.

Last, we take a look at the CreateVarInt module. The syntax is slightly different from VarInt, as one can see below:

```
15 > Digits:=16:                                    #numerical precision
16 > x:=0.5904158239150231:                         #positive node
17 > w:=0.9964248649058515:                         #weight
18 > etc:=L,0,p,q,h:                                #shorthand
19 > CreateVarInt(-1..1,[-x,x],[w,w],etc):          #obtain dEL equations
20 > ExtractAlgorithm(%,p,q,V);                     #obtain algorithm.
```

The first argument is the range over which the nodes are defined, so that the nodes, supplied as a list as the second argument to CreateVarInt, can be transformed appropriately. The third argument is the list of weights associated with these nodes. The fourth through to the eighth argument are the Lagrangian function, the Rayleigh function, and the labels for the canonical momenta, canonical coordinates and the time step, respectively. The ninth argument is optional, and it is not shown here; it takes the handle of an interpolation procedure, which must have the same syntax as the built-in procedures for data interpolation, as specified in the documentation of the CurveFitting package. If the ninth argument is omitted, the standard polynomial interpolation procedure PolynomialInterpolation (also known as interp) is used internally.

For instance, consider a custom yet naive implementation of polynomial interpolation:

```
21 > Poly:=proc(xdata,ydata,z)                      #custom interpolation
22     local c,n,Eqs,Var,Fun;
23     n:=nops(xdata):
24     Fun:=x->add(c[k]*x^(k-1),k=1..n):
25     Eqs:={seq(Fun(xdata[m])=ydata[m],m=1..n)}:
26     Var:={seq(c[m],m=1..n)}:
27     assign(solve(Eqs,Var)):
28     collect(factor(Fun(z)),z);
29   end proc:
30 > etc:=L,0,p,q,h,Poly;                           #shorthand
31 > CreateVarInt(-1..1,[-1,1],[1,1],etc):         #obtain dEL equations
32 > ExtractAlgorithm(%,p,q,V);                     #obtain algorithm.
```

The code obviously yields the Störmer–Verlet algorithm (2.19). The values for the nodes and weights shown on lines 16 and 17 are such that the underlying quadrature rule integrates any linear combinations of the set $\{e^{\pm\nu x}, x\,e^{\pm\nu x}\}$ with $\nu = 1$ exactly on the interval $[-1,1]$. Recently, non-polynomially fitted quadrature rules have moved increasingly to the centre of attention [16, 18, 77], especially exponentially fitted ones for numerical integration algorithms for ordinary differential equations [17]. The idea is to translate the philosophy behind Gaussian integration formulas, that is that they integrate polynomials exactly, to non-polynomial functions, in particular exponentials and trigonometric functions, based on the formalism developed by Ixaru [76]. That leads to a set

of non-linear conditions, from which the nodes and weights can be computed (numerically). Unfortunately, the nodes and corresponding weights for these exponentially fitted quadrature rules are not determined uniquely.

The optional argument enables us to provide alternative interpolation routines, which can be practical both as a diagnostic tool and as an interface to create new variational integrators that are designed for specific dynamical systems. Quadrature rules based on rational interpolation [7], for instance, might be of use in the simulations of dynamical systems with singularities, such as N-body problems in astrophysics and molecular dynamics.

2.4 Numerical Performance

To illustrate the performance of some of the variational integrators that we can generate with **VarInt**, we look at a simple test case. Consider a particle with unit mass in one dimension moving under the influence of the potential $V(q) = q^2\left(q^2 - 1\right)$. The corresponding Lagrangian is

$$L(q, \dot{q}) = \frac{1}{2}\dot{q}^2 - V(q).$$

In addition, let there be a dissipative force $F(q, \dot{q}) = -\epsilon\dot{q}$, where $\epsilon \geq 0$.

We first look at the case for which $\epsilon = 0$, that is the conservative case. The maximum relative energy errors for several variational integrators with $h = 0.25$ over a time span of 1000 units are shown in Table 2.1, where we have set $q\left(0\right) = 1$ and $\dot{q}\left(0\right) = 0$ initially. It is clear that these variational integrators do not conserve the energy exactly, but the errors remain bounded *at all times*.

	$n = 2$	$n = 3$	$n = 4$
Gauss–Legendre	$8.70888744 \cdot 10^{-2}$	$5.52211493 \cdot 10^{-4}$	$1.01220402 \cdot 10^{-5}$
Gauss–Lobatto	$1.17754867 \cdot 10^{-1}$	$2.33293558 \cdot 10^{-3}$	$2.68473932 \cdot 10^{-5}$
Fejér (1st)	$8.22761193 \cdot 10^{-2}$	$9.56875975 \cdot 10^{-4}$	$1.47025385 \cdot 10^{-4}$
Fejér (2nd)	$8.82216383 \cdot 10^{-2}$	$4.55277144 \cdot 10^{-4}$	$2.73505976 \cdot 10^{-4}$
Fejér (3nd)	$8.83051038 \cdot 10^{-2}$	$2.58435463 \cdot 10^{-3}$	$2.76487029 \cdot 10^{-4}$
Fejér (4th)	$8.82075954 \cdot 10^{-2}$	$2.59877552 \cdot 10^{-3}$	$2.76757734 \cdot 10^{-4}$
Chebyshev	$8.70888744 \cdot 10^{-2}$	$4.55277144 \cdot 10^{-4}$	$8.14513001 \cdot 10^{-6}$

Table 2.1: Maximum relative energy errors for the quartic oscillator.

For the numerical demonstration of the quartic oscillator with dissipation we choose $\epsilon = 10^{-2}$. The dissipation causes the particle to settle down in one of the minima of the potential, located at $q = \pm\frac{1}{2}\sqrt{2}$, so that the final energy of the particle is $E = -\frac{1}{4}$. Initially $q(0) = 0$ and $q(0) = \frac{1}{10}$. As a benchmark we have used the high-fidelity Maple dsolve[taylorseries] integrator.

The absolute energy errors are shown in Figure 2.1 for a selection of variational integrators with four nodes. It is clear that for this particular problem Fejér's first rule gives the best variational integrator in terms of accuracy. All variational integrators shown converge to the correct solution, as anticipated. These algorithms exhibit the correct energy behaviour, that is they do not dissipate artificially, as non-variational integrators typically do [83].

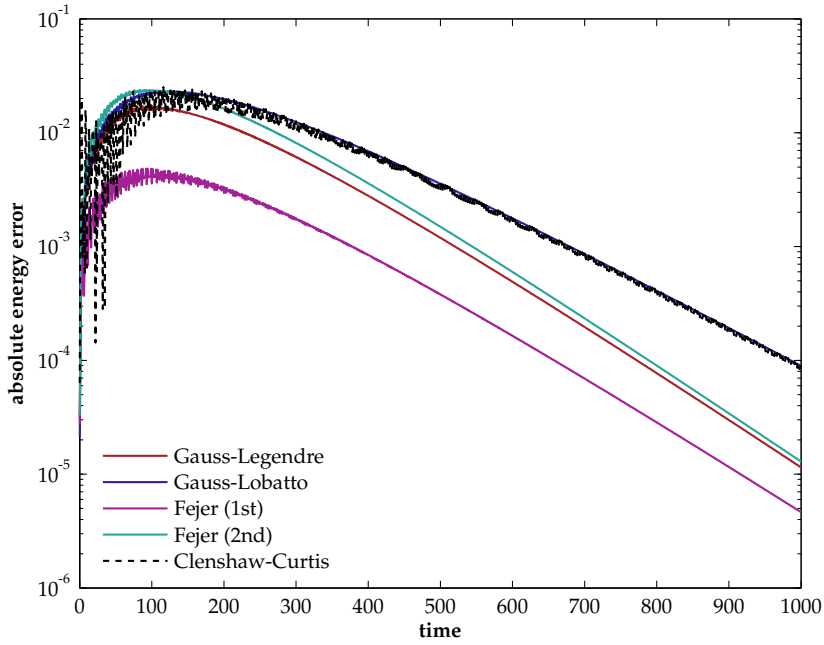


FIGURE 2.1: Absolute energy errors of some variational integrators with four nodes for the quartic oscillator with dissipation.

CHAPTER 3

Astrodynamics

The centrepiece of celestial mechanics and astrodynamics, which deals with the dynamics of spacecraft and artificial satellites under the influence of celestial bodies, is the Newtonian N-body problem. It describes the gravitational interaction between $N \geq 2$ particles. Here we shall finally savour the fruits of our labours, and cast the N-body problem in a geometric form in Section 3.1 to apply the Marsden–Weinstein–Meyer reduction from Chapter 1 to it. Since we shall mainly be concerned with space applications, and since there is a preponderance of man-made satellites and spacecraft in orbits around the Earth, we shall, as customary in space flight mechanics, restrict our attention mainly to the N-body problem with $N = 2$, or the Kepler–Coulomb problem. For some recent developments in astrodynamics, such as low-energy transfers that rely on the exploitation of invariant manifolds [89] and weak stability boundaries [15] in the three-body problem and beyond, we refer to the literature.

Orbital mechanics is based on the Kepler–Coulomb problem, which we describe in Section 3.2. The Hamilton–Jacobi equation for the Kepler–Coulomb problem is known to be separable in four coordinate systems only: spherical, parabolic, elliptic and spheroconical [35]. Adding a generic perturbation generally destroys the integrability and symmetry of the original system. However, McIntosh and Cisneros [113], and Zwanziger [168], who focussed solely

on its quantum-mechanical implications, discovered that the added potential of a self-dual Dirac (magnetic) monopole preserves the symmetries of the Kepler–Coulomb problem, and hence that it remains integrable. The McIntosh–Cisneros–Zwanziger (MICZ) problem has been solved formally by Bates [13] using Souriau's regularization technique [35, Chap. 5]. Although Bates' solution is simple and elegant, it lacks a clear physical interpretation; third-order derivatives of the coordinates with respect to the (reparameterized) time are required.

We review the classical MICZ problem from a universal variable point of view, as outlined previously by Caballero and Elipe [27]. We address its relevance to astrodynamics in the next chapter. In Section 3.2.1 we briefly review the symmetries of the classical MICZ problem in relation to the Kepler–Coulomb problem. The similarities between these systems enable us to write down an analytical solution that is valid for any type of orbit, as shown in Section 3.2.2. Following Deprit [41], the MICZ problem is commonly denominated in the astrodynamical literature quasi-Keplerian for reasons that will become apparent below. Section 3.2.3 comprises a discussion of the quasi-Keplerian orbital elements and Delaunay variables, from which other orbital representations may be obtained. We have endeavoured to keep the discussion as general as possible, so that our results can be applied to related physical systems, such as for example Rydberg atoms in the presence of magnetic monopoles.

The Kepler–Coulomb problem and its relative the MICZ problem are both highly stylized; they describe the classical interaction of point masses. In actual applications the shape, and mass distributions of the primaries about which spacecraft orbit affect the motion too. Orbital perturbations that are most prominent in the sphere of influence of the Earth are presented in Section 3.2.5.

An integral part of the on-board guidance and navigation software of any spacecraft or artificial satellite is the attitude and orbit control system (AOCS). It controls the orbital parameters and relative orientation of the craft. In practical applications of artificial satellites the instantaneous position and velocity in orbit and the relative orientation, or attitude, with respect to the Earth's frame of reference are equally important; the relative orientation of a craft in orbit is essential to the operational status of the craft, with regard to its thermal equilibrium, and the alignment of solar arrays and propulsion modules for instance, and also to the mission itself, especially with regard to telecommunication devices and the use of scientific equipment. The attitude of a satellite is determined by its orbit, shape, and mass distribution. The attitude dynamics and its geometric struc-

ture are discussed in Section 3.3, whereupon we introduce the gravity-gradient torque in Section 3.3.5.

3.1 *N*-Body Problem

Let $(\boldsymbol{q}_i, \boldsymbol{p}_i)$ denote the coordinates on the ith copy $\mathbf{T}^\star\mathbb{R}^3 \cong \mathbb{R}^6$ of the entire phase space $\mathcal{M} = \left(\mathbf{T}^\star\mathbb{R}^3\right)^N \cong \mathbb{R}^{6N}$ of the ith particle with mass m_i, where $i = 1, \ldots, N$. The N-body Hamiltonian is

$$H = \sum_{i=1}^{N} \frac{1}{2m_i} \|\boldsymbol{p}_i\|^2 - G \sum_{i<j} \frac{m_i m_j}{\|\boldsymbol{q}_i - \boldsymbol{q}_j\|}, \tag{3.1}$$

where G is Newton's gravitational constant, and $\|\cdot\|^2 = \langle \cdot, \cdot \rangle$ denotes the standard Euclidean (ℓ^2) norm. The Hamiltonian is clearly invariant under the lifted action[1] of the three-dimensional Euclidean group $\mathrm{E}(3, \mathbb{R}) = \mathbb{R}^3 \rtimes \mathrm{O}(3, \mathbb{R})$; it acts on any $\mathbb{R}^3$ factor by $((\boldsymbol{a}, \mathbf{A}), \boldsymbol{q}) \mapsto \mathbf{A}\boldsymbol{q} + \boldsymbol{a}$ for all $(\boldsymbol{a}, \mathbf{A}) \in \mathrm{E}(3, \mathbb{R})$ and $\boldsymbol{q} \in \mathbb{R}^3$. The subgroup that preserves the orientation of all Euclidean isometries is called the special Euclidean group $\mathrm{SE}(3, \mathbb{R}) = \mathbb{R}^3 \rtimes \mathrm{SO}(3, \mathbb{R})$, where $\mathrm{O}(3, \mathbb{R}) \cong \mathrm{SO}(3, \mathbb{R}) \times \mathbb{Z}_2$.

Before we can reduce the phase space for the N-body problem, we have to take a closer look at the momentum maps associated with the individual (connected) components of the $\mathrm{E}(3, \mathbb{R})$ symmetry, namely the induced $\mathbb{R}^3$ and $\mathrm{SO}(3, \mathbb{R})$ actions. The infinitesimal generator of the lifted $\mathbb{R}^3$ action is

$$X^\xi = \xi^\alpha \frac{\partial}{\partial q^\alpha},$$

which we can also write as $\boldsymbol{X}^\xi = (\boldsymbol{\xi}, \mathbf{0})$. We obtain from equation (1.21) that

$$\frac{\partial \mu^\xi}{\partial p_\alpha} = \xi^\alpha, \qquad \frac{\partial \mu^\xi}{\partial q^\alpha} = 0.$$

These equations lead us to the conclusion that $\mu^\xi = \boldsymbol{p} \cdot \boldsymbol{\xi}$, which means that the sought-after momentum map $\mu \colon \mathbb{R}^3 \times (\mathbb{R}^3)^\star \to (\mathbb{R}^3)^\star$ is $(\boldsymbol{q}, \boldsymbol{p}) \mapsto \boldsymbol{p}$, the (linear) momentum. We can easily convince ourselves that the N-body momentum map is the total momentum of the system.

[1] The lifted action $\mathrm{E}(3, \mathbb{R}) \times \mathbf{T}^\star\mathbb{R}^3 \mapsto \mathbf{T}^\star\mathbb{R}^3$ is given by $((\boldsymbol{a}, \mathbf{A}), (\boldsymbol{q}, \boldsymbol{p})) \mapsto (\mathbf{A}\boldsymbol{q} + \boldsymbol{a}, \boldsymbol{p}\mathbf{A}^\top)$. Indeed, consider a finite-dimensional base manifold Q with coordinates $\boldsymbol{q}$, and a map $f \colon Q \to Q$. If $(\boldsymbol{q}, \dot{\boldsymbol{q}})$ and $(\boldsymbol{q}', \dot{\boldsymbol{q}}')$ are tangent-lifted coordinates, and $(\boldsymbol{q}, \boldsymbol{p})$ and $(\boldsymbol{q}, \boldsymbol{p}')$ represent cotangent-lifted coordinates, then $\dot{\boldsymbol{q}}' = \mathrm{D}f(\boldsymbol{q})\dot{\boldsymbol{q}}$ and $\boldsymbol{p}' = \boldsymbol{p}(\mathrm{D}f(\boldsymbol{q}))^{-1}$ as long as $(\boldsymbol{q}', \boldsymbol{p}') = (\mathbf{T}^\star f)^{-1}(\boldsymbol{q}, \boldsymbol{p})$.

To obtain the momentum map for the lifted action of $\mathrm{SO}(3,\mathbb{R})$, we note that we may identify $\mathfrak{so}(3)$ with $\mathbb{R}^3$ by what is sometimes referred to in the literature as the hat ($\widehat{\ }$) map. As the Lie algebra of the matrix Lie group $\mathrm{SO}(3,\mathbb{R})$, $\mathfrak{so}(n) = \left\{\mathbf{A} \in \mathrm{GL}(n,\mathbb{R}) \;\middle|\; \mathbf{A} + \mathbf{A}^\top = \mathbf{0}\right\}$, so that the linear isomorphism[2] is

$$\begin{aligned} \mathbb{R}^3 &\to \mathbf{T}_e \mathrm{SO}(3,\mathbb{R}) \\ \boldsymbol{a} = (a^1, a^2, a^3) \mapsto \mathbf{A} \equiv \widehat{\boldsymbol{a}} &= \begin{pmatrix} 0 & -a^3 & a^2 \\ a^3 & 0 & -a^1 \\ -a^2 & a^1 & 0 \end{pmatrix} \in \mathfrak{so}(3), \end{aligned}$$

with the Lie bracket $[\cdot,\cdot]$, the commutator, replaced by the exterior product of vectors in $\mathbb{R}^3$, that is $\boldsymbol{a} \wedge \boldsymbol{b} \mapsto \left[\widehat{\boldsymbol{a}}, \widehat{\boldsymbol{b}}\right] = [\mathbf{A}, \mathbf{B}] = \mathbf{AB} - \mathbf{BA}$. Here obviously the group identity $e = \mathbf{I}$. The standard inner product is given by $\boldsymbol{a} \cdot \boldsymbol{b} = -\frac{1}{2} \operatorname{Tr} \mathbf{AB}$, the Killing form, which defines a metric on $\mathfrak{so}(3)$, so that the identification is in fact an isometry. As before, we have $\mathbb{R}^\star \cong \mathbb{R}$ via the inner product. All in all, we end up with $\mathfrak{so}(3)^\star \cong \mathbb{R}^3$.

As before, we compute the infinitesimal generator to be

$$\boldsymbol{X}^\xi = (\xi \boldsymbol{q}, \xi \boldsymbol{p}) = (\boldsymbol{\omega} \wedge \boldsymbol{q}, \boldsymbol{\omega} \wedge \boldsymbol{p})$$

for $\xi = \widehat{\boldsymbol{\omega}} \in \mathfrak{so}(3)$. Consequently,

$$\frac{\partial \mu^\xi}{\partial \boldsymbol{p}} = \xi \boldsymbol{q}, \qquad \frac{\partial \mu^\xi}{\partial \boldsymbol{q}} = -\xi \boldsymbol{p},$$

which can be solved by $\mu^\xi = \xi \boldsymbol{q} \cdot \boldsymbol{p} = (\boldsymbol{\omega} \wedge \boldsymbol{q}) \cdot \boldsymbol{p} = (\boldsymbol{q} \wedge \boldsymbol{p}) \cdot \boldsymbol{\omega}$. The angular momentum $\boldsymbol{L} \equiv \boldsymbol{q} \wedge \boldsymbol{p}$ is the momentum map, and the total angular momentum is the desired momentum map for the whole N-body system.

We can now turn our attention to the symplectic reduction of the N-body problem, which corresponds to the conventional physicists' transformation to the centre-of-mass coordinates. The Marsden–Weinstein–Meyer theorem formalizes the reductive aspects of this well-known transformation. We take as the phase space the collision-free manifold

$$\mathcal{N} \equiv \left\{ ((\boldsymbol{q}_1, \boldsymbol{p}_1), \ldots, (\boldsymbol{q}_N, \boldsymbol{p}_N)) \in \left(\mathbf{T}^\star \mathbb{R}^3\right)^N \;\middle|\; \boldsymbol{q}_i \neq \boldsymbol{q}_j \text{ for } i \neq j \right\}.$$

[2]The isomorphism becomes even more apparent when we look at the algebraic structure of $\mathfrak{so}(n)$: $[e_i, e_j] = \epsilon_{ij}{}^k e_k$ for the basis elements e_i with $i = 1, \ldots, \dim\left(\mathfrak{so}(n)\right) = \frac{1}{2} n(n-1)$. Here $\epsilon_{ij}{}^k$ denotes the permutation symbol.

It is equipped with the standard N-body symplectic form

$$\omega = \sum_{i=1}^{N} \mathrm{d}\boldsymbol{q}_i \wedge \mathrm{d}\boldsymbol{p}_i.$$

We could start out with the entire phase space $\left(\mathbf{T}^\star\mathbb{R}^3\right)^N$, but we would end up with an orbifold rather than a manifold. The downside of our collision-free construction is that the initial and reduced manifolds will not be closed; the corresponding (reduced) Hamiltonian vector field is incomplete.

For convenience let $\mathcal{R} \equiv (\mathbb{R}^3)^N$. We fix a $\boldsymbol{P} \in \mathcal{R}^\star \cong \mathcal{R}$, such that $\sum_{i=1}^N \boldsymbol{p}_i = \boldsymbol{P}$. For any level $\boldsymbol{P}$ the space $\mu^{-1}(\boldsymbol{P})/\mathbb{R}^3$ identifies all points that differ by an overall translation of all the canonical coordinates of the N particles at a constant total momentum $\boldsymbol{P}$. A possible set of coordinates on the reduced base manifold is given by the relative coordinates $\boldsymbol{r}_{ij} = \boldsymbol{q}_i - \boldsymbol{q}_j$ for $i < j$, and we have that $\dim\left(\mathcal{N}/\!/\mathbb{R}^3\right) = 6N - 6$. Different sets of coordinates, such as the well-known Jacobi coordinates, can of course be used instead.

Please observe that the connected component $\mathrm{SE}\,(3,\mathbb{R})$ does not act freely on $\mathcal{N}$, as the dimension of its stabilizer $G_{\boldsymbol{n}}$ for $\boldsymbol{n} = (\boldsymbol{q},\boldsymbol{p}) \in \mathcal{N}$, defined by the elements $(\boldsymbol{a},\mathbf{A}) \in \mathrm{SE}\,(3,\mathbb{R})$, such that

$$\begin{cases} \mathbf{A}\boldsymbol{q} + \boldsymbol{a} &= \boldsymbol{q}, \\ \mathbf{A}\boldsymbol{p} &= \boldsymbol{p}, \end{cases}$$

is at least one if and only if the N particles are collinear, or at syzygy, and the momenta all lie on a line through the origin that is parallel to the line connecting the particles. For the action of the whole group $\mathrm{E}\,(3,\mathbb{R})$ the requirements for a non-trivial stabilizer are that the $\boldsymbol{q}_i$ lie in a plane W and that the $\boldsymbol{p}_i$ are constrained to the plane W_0 through the origin with $W_0 \parallel W$. In particular, N-body configurations with a net angular momentum $\boldsymbol{L} \in \mathcal{R}^\star$ equal to zero are enclosed within. Please note that for $N = 2$, the use of the collision-free phase space already precludes the situation with net zero angular momentum.

If, however, we restrict our attention to the open (dense) submanifold of $\mathcal{N}$ with a trivial stabilizer, then we may still apply the Marsden–Weinstein–Meyer theorem [68]. Since the coadjoint isotropy group is $G_{\boldsymbol{0}} = \mathrm{O}\,(3,\mathbb{R})$, the reduced phase space has a dimension of $6N - 12$ for $N \geq 3$. The coordinates $\boldsymbol{r}_{ij}$ on $\mathcal{N}_1 \equiv \mathcal{N}/\!/\mathbb{R}^3$ now become $\|\boldsymbol{r}_{ij}\|^2$ on $\mathcal{N}_2 \equiv \mathcal{N}_1/\!/\mathrm{O}\,(3,\mathbb{R})$. There are $\binom{N}{2}$ such coordinates, so that for $N \geq 5$ these are not all independent.

For a non-zero total angular momentum $\boldsymbol{L} \in \mathcal{R}^\star$, $G_{\boldsymbol{L}} = \mathrm{SO}\,(2,\mathbb{R}) \cong \mathbb{S}^1$, since only rotations in the plane perpendicular to $\boldsymbol{L}$ leave the total angular momentum

invariant. Consequently, we wind up with a $(6N-10)$-dimensional reduced phase space for $N \geq 2$. Coordinates on $\mathcal{N}_2$ are known as shape coordinates. The computation of suitable shape coordinates for the N-body problem with $N \geq 3$ is actually quite intricate [see e.g. 102]. Additional thoughts on the reduction of the N-body problem can be found in the article by Lin and Marsden [101].

3.2 Kepler–Coulomb Problem

For $N = 2$ we have the option of taking the origin of our coordinate system to coincide with centre of mass of the two bodies, for which naturally $\boldsymbol{P} = \boldsymbol{0}$. We neglect the mass of the satellite particle because $M \equiv m_1 + m_2 \gg m_2$ for any astrodynamical purposes, so that the centre of mass coincides with the frame of reference of the primary, that is the particle with the largest mass. The corresponding Kepler–Coulomb Hamiltonian $H\colon \mathcal{N}_1 \to \mathbb{R}$ is

$$H(\boldsymbol{q},\boldsymbol{p}) = \frac{1}{2}\|\boldsymbol{p}\|^2 - \frac{k}{\|\boldsymbol{q}\|}, \tag{3.2}$$

in which we have absorbed all constants in $k \in \mathbb{R}$. We can actually compute the local shape coordinates on $\mathcal{N}_2$ rather easily: the quotient map $\sigma\colon \mathcal{N}_1 \to \mathcal{N}_2$ is given by $(\boldsymbol{q},\boldsymbol{p}) \mapsto (r,\theta) = (\|\boldsymbol{q}\|, \boldsymbol{p}\cdot\boldsymbol{q}/\|\boldsymbol{q}\|)$. Hence, we see that $\mathcal{N}_2 \cong \mathbb{R}^+ \times \mathbb{R} = \big\{(r,\theta) \in \mathbb{R}^2 \,\big|\, r > 0\big\}$. The Marsden–Weinstein–Meyer theorem prescribes the relations $i^*\omega = \sigma^*\omega^{\boldsymbol{L}}$ and $H^{\boldsymbol{L}} \circ \sigma = H \circ i$ with $i\colon \mu^{-1}(\boldsymbol{L}) \hookrightarrow \mathcal{N}_1$, which enable us to see that $\omega^{\boldsymbol{L}} = \mathrm{d}r \wedge \mathrm{d}\theta$ and that the fully reduced Hamiltonian $H^{\boldsymbol{L}}\colon \mathcal{N}_2 \to \mathbb{R}$ is

$$H^{\boldsymbol{L}}(r,\theta) = \frac{r^2\theta^2 + \|\boldsymbol{L}\|^2}{2r^2} - \frac{k}{r}.$$

The Kepler–Coulomb problem (3.2) has, in addition to the linear and angular momenta, a different integral of motion, namely the Laplace–Runge–Lenz vector $\boldsymbol{R} = \boldsymbol{p} \wedge \boldsymbol{L} - k\boldsymbol{q}/q$, for which we have that $\boldsymbol{L}\cdot\boldsymbol{R} = 0$; the Kepler–Coulomb problem is said to be maximally super-integrable. To understand the symmetry associated with this remarkable quantity, we compute the Poisson brackets:

$$\big\{L_\alpha, L_\beta\big\} = \epsilon_{\alpha\beta}{}^{\gamma} L_\gamma, \quad \big\{L_\alpha, R_\beta\big\} = \epsilon_{\alpha\beta}{}^{\gamma} R_\gamma, \quad \big\{R_\alpha, R_\beta\big\} = -2H\epsilon_{\alpha\beta}{}^{\gamma} L_\gamma.$$

These equations show that there are three (algebraically) distinct domains: two open regions $H < 0$ and $H > 0$, and their common boundary $H = 0$. The open regions correspond to the elliptic (bounded) and hyperbolic (unbounded) types of motion, whereas the boundary represents all parabolic motions. For $H > 0$ we can scale the Laplace–Runge–Lenz vector by a factor $2H$ to obtain an

$\mathfrak{o}(3,1)$ (Lorentz) algebra, and for $H < 0$ we may scale that very vector by $-2H$ so we are left with an $\mathfrak{o}(4)$ algebra. Since we are mainly concerned with the ramifications for bounded orbits, we shall consider the case $H < 0$ from here on.

We define the scaled Laplace–Runge–Lenz vector as

$$\boldsymbol{R} = \frac{1}{\sqrt{2|H|}} \left(\boldsymbol{p} \wedge \boldsymbol{L} - k \frac{\boldsymbol{q}}{q} \right), \tag{3.3}$$

which we shall now simply refer to as the Laplace–Runge–Lenz vector. To see the structure of the $\mathfrak{o}(4)$ algebra more clearly, we define $\boldsymbol{A}^{\pm} = \frac{1}{2}(\boldsymbol{L} \pm \boldsymbol{R})$, and calculate the Poisson brackets to be

$$\left\{ A^+_\alpha, A^+_\beta \right\} = \epsilon_{\alpha\beta}{}^\gamma A^+_\gamma, \quad \left\{ A^-_\alpha, A^-_\beta \right\} = \epsilon_{\alpha\beta}{}^\gamma A^-_\gamma, \quad \left\{ A^+_\alpha, A^-_\beta \right\} = 0,$$

with $\|\boldsymbol{A}^+\| = \|\boldsymbol{A}^-\|$. It shows that $\mathfrak{o}(4) \cong \mathfrak{o}(3) \oplus \mathfrak{o}(3)$, for which its connected component is $\mathfrak{so}(4)$. The momentum map associated with the diagonal $\mathrm{SO}(4,\mathbb{R})$ action is [37, p. 57]

$$\begin{aligned} \mu \colon \mathcal{N}_1 &\to \mathfrak{so}(4)^\star \\ (\boldsymbol{q}, \boldsymbol{p}) &\mapsto (\boldsymbol{A}^+ + \boldsymbol{A}^-, \boldsymbol{A}^+ - \boldsymbol{A}^-) = (\boldsymbol{L}, \boldsymbol{R}). \end{aligned} \tag{3.4}$$

The origin of an $\mathrm{SO}(4,\mathbb{R})$ symmetry group in a three-dimensional dynamical system might seem quite obscure at first sight. The Kepler–Coulomb problem in three-dimensional Euclidean space is related to the (constrained) harmonic oscillator in four dimensions by the Kustaanheimo–Stiefel transformation [92], or the Duru–Kleinert transformation [43], as it is known in the context of atomic path integrals. The transformation linearizes and regularizes the equations of motion; it is a generalization of the Euler and Levi-Cività transformations in one and two spatial dimensions respectively. Moreover, Moser [128] has shown that the flow of the n-dimensional (regularized) Kepler–Coulomb problem is equivalent to the geodesic flow of Riemannian spaces of constant curvature, the n-spheres $\mathbb{S}^n$; these spaces are topologically equivalent to the fixed-energy surfaces of the Kepler–Coulomb problem[3]. Moser's construction involves a stereographic projection $\mathbb{S}^3 \to \mathbb{R}^3$ that is extended to the cotangent bundle of the three-sphere $\mathbf{T}^\star\mathbb{S}^3$; all collision states are mapped to the north pole of

[3]More generally, the phase space of the regularized n-dimensional Kepler–Coulomb problem is symplectomorphic to a minimal coadjoint orbit of $\mathrm{SO}(2, n+1)$, the double cover of the conformal group of the Minkowski space in $n+1$ dimensions. The equivalence of the Moser and Kustaanheimo–Stiefel transformations are the consequence of the accidental (double cover) homomorphism $\mathrm{SU}(2,2) \to \mathrm{SO}(2,4)$ [91]. More details on the conformal structure can be found in Cordani [35], Chap. 6, and Guillemin and Sternberg [62], Chap. 3.

the three-sphere. Stated differently, the $\mathrm{SO}(4,\mathbb{R})$ symmetry arises because the Kepler–Coulomb problem in three dimensions is equivalent to a free particle in four dimensions that is constrained to move on the three-sphere, which has an obvious $\mathrm{SO}(4,\mathbb{R})$ symmetry.

Moser's mathematical construction had already been studied before in a quantum-mechanical setting by Fock [48] and Bargmann [10]. The cases with $H \geq 0$ have been covered by Belbruno [14]. The answer to the question whether there are any geodesic-equivalent flows for the N-body problem with $N \geq 3$ is negative unless the total angular momentum is zero and the Hamiltonian is strictly positive [111].

3.2.1 McIntosh–Cisneros–Zwanziger Problem

The Hamiltonian of the McIntosh–Cisneros–Zwanziger (MICZ) system is

$$H(\boldsymbol{q},\boldsymbol{p}) = \frac{1}{2}\|\boldsymbol{p}\|^2 - \frac{k}{\|\boldsymbol{q}\|} + \frac{\lambda^2}{2\|\boldsymbol{q}\|^2}, \tag{3.5}$$

where $k \in \mathbb{R}$. We shall occasionally write $p = \|\boldsymbol{p}\|$ and $q = \|\boldsymbol{q}\|$ whenever appropriate. The monopole potential is actually derived from a vector potential[4] $\boldsymbol{A}$, such that the magnetic field $\boldsymbol{B} = \nabla \wedge \boldsymbol{A} = \lambda \boldsymbol{q}/\|\boldsymbol{q}\|^3$ for $\boldsymbol{q} \in \mathbb{R}^3 \setminus \{0\}$. This in turn means that in going from the Kepler–Coulomb to the MICZ problem, the canonical momenta are transformed via a minimal substitution: $\boldsymbol{p} \mapsto \boldsymbol{p} - \boldsymbol{A}$. The parameter λ can be viewed as a deformation parameter; for $\lambda = 0$ we recover the well-known Kepler–Coulomb problem. The associated one-parameter family of differential equations is Hamiltonian at each value of λ, as shown by Bates [13]. These Hamiltonian deformations are distinct for different values of the deformation parameter. We require that $\lambda^2 \in \mathbb{R}$ in order to allow for both positive and negative (real) values.

The phase space of the classical MICZ problem $\mathcal{M} = \mathbf{T}^\star\big(\mathbb{R}^3 \setminus \{\mathbf{0}\}\big)$ is endowed with the Poisson structure

$$\begin{aligned} \{q^\alpha, q^\beta\} &= 0, \qquad \{q^\alpha, p_\beta\} = \delta^\alpha_\beta, \\ \{p_\alpha, p_\beta\} &= \lambda \epsilon_{\alpha\beta\gamma} q^\gamma / \|\boldsymbol{q}\|^3, \end{aligned}$$

[4] The original vector potential for Dirac's magnetic monopole [see e.g. 56, p. 165] for $\boldsymbol{q} = (q^1, q^2, q^3)$ is

$$\boldsymbol{A} = \frac{\lambda}{q}\frac{1}{q^3 - q}\left(-q^2, q^1, 0\right)$$

up to a gauge function. The field strength is calculated from $F = \mathrm{d}A$, where $A = \langle \boldsymbol{A}, \mathrm{d}\boldsymbol{q}\rangle$: $F = \lambda/\|\boldsymbol{q}\|^3 \left(q^1\,\mathrm{d}q^2 \wedge \mathrm{d}q^3 + q^2\,\mathrm{d}q^3 \wedge \mathrm{d}q^1 + q^3\,\mathrm{d}q^1 \wedge \mathrm{d}q^2\right)$. Observe that $\mathrm{d}F = 4\pi\lambda\delta(\boldsymbol{q})\,\mathrm{d}q^1 \wedge \mathrm{d}q^2 \wedge \mathrm{d}q^3$.

and the herewith related canonical two-form

$$\omega = \mathrm{d}q^\alpha \wedge \mathrm{d}p_\alpha + \frac{\lambda}{2\|\boldsymbol{q}\|^3}\epsilon_{\alpha\beta\gamma}q^\alpha \mathrm{d}q^\beta \wedge \mathrm{d}q^\gamma. \tag{3.6}$$

The first term in equation (3.6) is simply the symplectic two-form of the Kepler–Coulomb system. The additional dynamics due to the monopole potential is incorporated in the second term. Physically, λ is the magnetic 'charge' of the monopole. Mathematically, the deformation parameter can be interpreted as the De Rham cohomology class of the symplectic form[5].

Since the Hamiltonian function is independent of time it is a conserved quantity. A simple calculation reveals that because of rotational invariance,

$$\boldsymbol{L} = \boldsymbol{q} \wedge \boldsymbol{p} - \lambda\frac{\boldsymbol{q}}{q} \tag{3.7}$$

is conserved as well. In analogy with the Kepler–Coulomb problem there is an additional conserved quantity:

$$\boldsymbol{R} = \frac{1}{\sqrt{2|H|}}\left(\boldsymbol{p} \wedge \boldsymbol{L} - k\frac{\boldsymbol{q}}{q}\right), \tag{3.8}$$

which corresponds to the Laplace–Runge–Lenz vector (3.3), where now obviously $\boldsymbol{L}$ from equation (3.7) has to be substituted. Under the Poisson bracket the Laplace–Runge–Lenz vector yields the $\mathfrak{so}(4)$ algebra for $H < 0$, that is, the Lie algebra of the four-dimensional rotation group, and $\mathfrak{so}(3,1)$ for $H > 0$, which is the Lie algebra of the conformal group in four-dimensions. These are the same symmetries of the Kepler–Coulomb problem. In fact, $\boldsymbol{L} \cdot \boldsymbol{q}/q = -\lambda$, which shows that the solutions to Hamilton's equations for the equation (3.5) are conic sections, where the opening angle $\vartheta = \arccos \lambda/L$. Similarly we have that

$$\begin{aligned}\boldsymbol{q} \cdot (\boldsymbol{p} \wedge \boldsymbol{L}) &= \sqrt{2|H|}\,(\boldsymbol{q} \cdot \boldsymbol{R}) - q \\ &= L^2 - \lambda^2,\end{aligned}$$

so that the motion lies in the plane.

[5]In fact, the symplectic form and the field strength lie in the same De Rham cohomology class $[\omega] = [F]$ with $H^2_{\mathrm{dR}}(\mathcal{M}, \mathbb{R}) \cong \mathbb{R}$, since the first term in equation (3.6) is closed, and hence exact. The presence of a magnetic monopole is indicated by the fact that symplectic form ω (3.6) is not closed everywhere: $\mathrm{d}\omega \neq 0$; it is closed almost everywhere.

3.2.2 Universal Solution

Because the motion is planar, it is convenient to use polar coordinates $\boldsymbol{q} = (r, \theta)$ in the orbital plane:

$$H = \frac{1}{2}\left(p_r^2 + \frac{p_\theta^2}{r^2}\right) - \frac{k}{r} + \frac{\lambda^2}{2r^2}. \tag{3.9}$$

In polar coordinates Hamilton's equations for the Hamiltonian (3.9) simply read

$$\begin{cases} \dot{r} &= p_r, & \dot{p}_r &= \dfrac{p_\theta^2 + \lambda^2}{r^3} - \dfrac{k}{r^2}, \\ \dot{\theta} &= \dfrac{p_\theta}{r^2}, & \dot{p}_\theta &= 0. \end{cases} \tag{3.10}$$

It is customary to reparameterize the time t with a so-called Sundman transformation

$$\mathrm{d}t = r\mathrm{d}s. \tag{3.11}$$

Let the prime ($'$) denote the derivative with respect to the independent (universal) variable s, that is, $f'(s) = \mathrm{d}f(s)/\mathrm{d}s = r\dot{f}(t)$ for any $f \in \mathcal{C}^1(\mathbb{R}, \mathbb{R})$ by virtue of the chain rule and the transformation (3.11). Therefore, we find that

$$\begin{aligned} r'' &= (rp_r)' \\ &= r\frac{\mathrm{d}(rp_r)}{\mathrm{d}t} \\ &= r\left(p_r^2 + \frac{p_\theta^2 + \lambda^2}{r^2} - \frac{k}{r}\right). \end{aligned} \tag{3.12}$$

Since the Hamiltonian is constant, we have the identity

$$2H_0 - \left(p_r^2 + \frac{p_\theta^2 + \lambda^2}{r^2} - \frac{2k}{r}\right) = 0,$$

where H_0 is the initial (numerical) value of the Hamiltonian. Now we can add this expression to the bracketed expression of the equation (3.12) and obtain

$$r'' = 2H_0 r + k.$$

Hence, we arrive at the set of differential equations:

$$\begin{cases} r'' &= 2H_0 r + k, \\ \theta' &= p_\theta / r, \\ t' &= r, \end{cases} \tag{3.13}$$

which are to be supplied with initial conditions. The deformation parameter λ does not appear explicitly in these equations; it only resides in the value H_0.

Let r_0, r_0' and r_0'' denote the initial values of r, $r' = \boldsymbol{q} \cdot \boldsymbol{p}$ and $r'' = 2H_0 r + k$ respectively. The system (3.13) is then easily solved in terms of Stumpff functions by

$$\begin{cases} r(s) &= r_0 + r_0' s\, \mathrm{c}_1\left(-2H_0 s^2\right) + r_0'' s^2\, \mathrm{c}_2\left(-2H_0 s^2\right), \\ \theta(s) &= p_\theta \xi(s), \\ t(s) &= t_0 + r_0 s + r_0' s^2\, \mathrm{c}_2\left(-2H_0 s^2\right) + r_0'' s^3\, \mathrm{c}_3\left(-2H_0 s^2\right), \end{cases} \tag{3.14}$$

where we have defined

$$\xi(s) = \int_0^s \frac{\mathrm{d}\varphi}{r(\varphi)}. \tag{3.15}$$

Using the universal Stumpff functions and the definition $\beta = -2H_0$, the solution to $r(s)$ and $t(s)$ may be written more concisely as

$$r(s) = \; r_0 + r_0'\, \mathrm{U}_1(s;\beta) + r_0''\, \mathrm{U}_2(s;\beta), \tag{3.16a}$$

$$t(s) = t_0 + r_0 s + r_0'\, \mathrm{U}_2(s;\beta) + r_0''\, \mathrm{U}_3(s;\beta), \tag{3.16b}$$

where we set $t_0 = 0$ and the solution for the angle $\theta(s)$ is as before. The actual integration of equation (3.15), and therefore the solution of the polar angle as a function of the universal variable, which is completely absent in the discussion of Caballero and Elipe [27] yet necessary in order to convert the results to different coordinate systems, is straightforward if we rewrite the expression for $r(s)$ in terms of trigonometric functions with equations (D.2) and subsequently use the substitution $s = \tan\frac{1}{2}\sqrt{-2H_0}\varphi$ or $\varphi = 2\arctan s/\sqrt{-2H_0}$. In order to retain universality of the solution, we use the functions defined in Appendix D to obtain after some manipulations that

$$\xi(s) = \frac{s\, \mathrm{at}_1\left[(p_\theta^2 + \lambda^2)s^2\, \mathrm{tg}_1^2\left(-\frac{1}{2}H_0 s^2\right)\right]}{r_0 + \frac{1}{2}r_0' s\, \mathrm{tg}_1\left(-\frac{1}{2}H_0 s^2\right)} \tag{3.16c}$$

We note that these solutions are indeed 'universal', for they are well-defined irrespective of the sign of H_0, that is to say for any type of orbit.

3.2.3 Orbital Elements

In many practical situations we often encounter elliptic orbits, for which $H_0 < 0$. In these instances we require a formulation in terms of Keplerian, or orbital, elements to gain more insight into the geometry of the problem.

Because the deformation parameter does not appear explicitly in the equations of motion and thus their solutions except for the angle, the orbital elements can be found from the Keplerian case ($\lambda = 0$) by bearing in mind that now we need to use the quasi-Keplerian energy H_0, which includes the contribution from the inverse-square potential, instead of the Keplerian orbital energy. All in all, the semi-major axis a, eccentricity e, the mean anomaly $\mathscr{M}$, and the eccentric anomaly $\mathscr{E}$ can be defined in the usual way, that is

$$\begin{aligned} a &= -\frac{k}{2H_0}, \\ e^2 &= \left(1 - \frac{r}{a}\right)^2 + \frac{(r')^2}{ak}, \\ \mathscr{M} &= nt, \\ \mathscr{E} - e\sin\mathscr{E} &= \mathscr{M}, \end{aligned}$$

where the last equation is Kepler's equation. The mean motion is denoted by $n = \sqrt{k/a^3}$, which is related to the orbital period T by $T = 2\pi/n$. The eccentric anomaly is related to the true anomaly ν by

$$\cos\nu = \frac{\cos\mathscr{E} - e}{1 - e\cos\mathscr{E}}.$$

As usual, the distance can be expressed as $r = a(1 - e\cos\mathscr{E})$. Specifically we have that $k = GM$ with G Newton's gravitational constant and M is the mass of the primary; the mass of the artificial satellite is presumed to be negligible. The inclination of the orbit i is of course the angle between the orbital angular momentum $\boldsymbol{L}$ and the axis from the South Pole through the centre of the Earth in the direction of the North Pole, the polar axis. The argument of the perigee ω and the longitude of the ascending node Ω, or node for short, can be defined similarly using the geocentric-equatorial coordinate system, which can be found in the standard references on astrodynamics, such as Bate et al. [12], Danby [38], and Vallado [157]; the (angular) orbital elements are drawn in Figure 3.1 in the geocentric-equatorial coordinate system. These remarkable similarities enable us to translate Kepler's laws to the MICZ case.

There seems to be no formal difference at all in the definitions of the classical orbital elements; we merely have to substitute the numerical value of the MICZ Hamiltonian for the initial conditions. The polar angle θ, however, differs from its Keplerian counterpart. In order to understand the difference more clearly, let

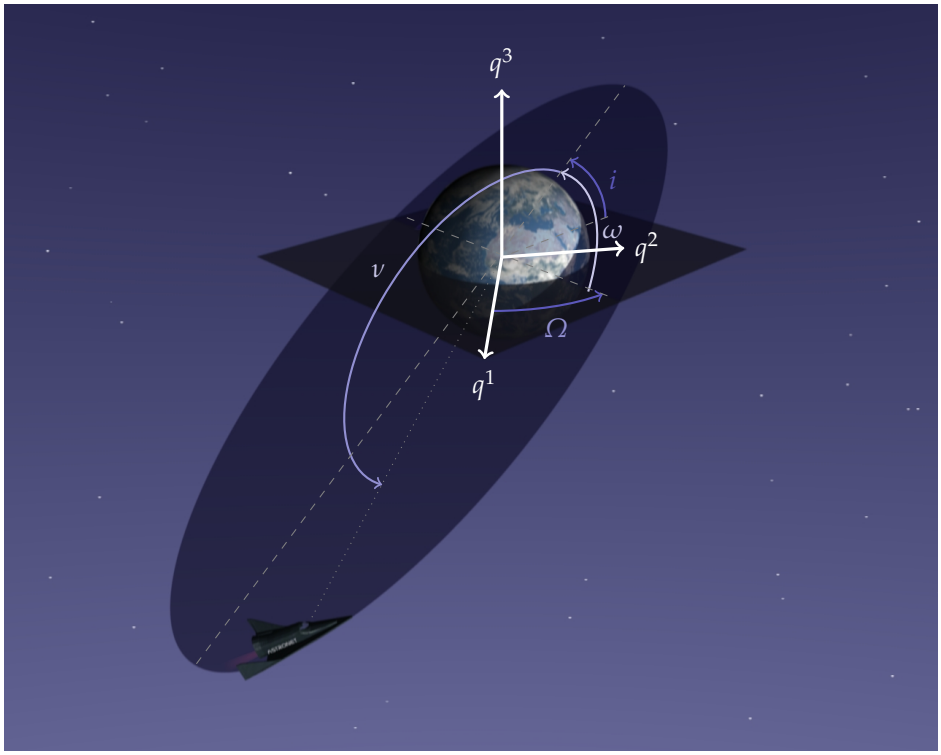


FIGURE 3.1: Graphical representation of the orbital elements in the geocentric-equatorial coordinate system. The q^1 axis points towards the vernal equinox at a particular epoch (Υ), the q^2 axis lies 90° east in the equatorial plane, and the q^3 axis passes through the North Pole. As an aside, we note that the vernal equinox occurs at the ascending node of the Sun; it does not coincide exactly with the intersection of the ecliptic and the equator since the ecliptic describes the mean path of the Sun.

us introduce the canonical transformation $p_\theta \mapsto p_\psi = \sqrt{p_\theta^2 + \lambda^2}$, of which

$$S(\boldsymbol{q}, \boldsymbol{p}) = r p_r + \psi p_\psi \tag{3.18}$$

is the generating function, such that

$$\boldsymbol{q} = \frac{\partial S}{\partial \boldsymbol{p}}, \qquad \boldsymbol{p} = \frac{\partial S}{\partial \boldsymbol{q}},$$

with $\boldsymbol{q} = (r, \psi)$ and $\boldsymbol{p} = (p_r, p_\psi)$. Here ψ represents the transformed polar angle induced by the transformation. For $\lambda = 0$ the canonical transformation is obviously the trivial one. Note that $p_\psi^2 = L^2$, as one may verify from equation (3.7).

From the expression of the generating function (3.18) and the equation for ψ we find that

$$\psi = \frac{p_\theta}{\sqrt{p_\theta^2 + \lambda^2}}. \tag{3.19}$$

Consequently, the true anomaly ν, which is often used rather than the mean anomaly $\mathscr{M}$, is related to the polar angle θ by

$$\theta = \psi \nu. \tag{3.20}$$

We note that this relation can be derived from the definition of the true anomaly in terms of the eccentricity and the eccentric anomaly as well, but we refrain from doing so here.

Before turning our attention to the canonical Delaunay variables, it is useful to look at the action–angle coordinates in polar coordinates still. The MICZ problem seems but a variation, or better, an extension, of its Keplerian original, yet there is a subtle difference. We start from the time-independent Hamilton–Jacobi equation (1.23). In polar coordinates it reads

$$\frac{1}{2}\left[\left(\frac{\partial S_r}{\partial r}\right)^2 + \frac{1}{r^2}\left(\frac{\partial S_\theta}{\partial r}\right)^2\right] + V(r) = E$$

for any central force problem, that is for any potential $V = V(r)$. Here we have split Hamilton's characteristic function $S = S_r + S_\theta$. Multiplication by $2r^2$

separates the r and θ parts, so that their respective characteristic functions can be integrated to give

$$S_r(r) = \int_0^r \sqrt{2\left(E - V(\rho)\right) - \frac{\ell^2}{\rho^2}}\,\mathrm{d}\rho,$$
$$S_\theta(\theta) = \ell\theta,$$

where $\ell = p_\theta = \sqrt{L^2 - \lambda^2}$ is the (constant) angular momentum.

The action variables are then calculated from equation (1.25):

$$J_\alpha = \frac{1}{2\pi}\oint p_\alpha\,\mathrm{d}q^\alpha,$$

where the integration is over all possible values of q^α for fixed energies E; no summation is implied in the integrand. Notice that $p_\alpha = \partial S/\partial q^\alpha$. In particular, we have that

$$J_r = -\sqrt{J_\theta^2 + \lambda^2} + k\sqrt{-\frac{1}{2E}}$$

and

$$J_\theta = \ell$$

for the MICZ potential, where now $E = H_0$. The Hamiltonian becomes in terms of these action variables

$$H(\boldsymbol{J}) = -\frac{k^2}{2\left(J_r + \sqrt{J_\theta^2 + \lambda^2}\right)^2}. \tag{3.21}$$

By plugging these results into Hamilton's characteristic function we obtain the angle variables ϕ^α by differentiation, more precisely,

$$\phi^\alpha = \frac{\partial S(\boldsymbol{J})}{\partial J_\alpha}.$$

It is, however, more interesting at this point to look at the frequencies $\boldsymbol{\nu}$ associated with the generalized coordinates,

$$\nu^\alpha = \dot{\phi}^\alpha = \frac{\partial H(\boldsymbol{J})}{\partial J_\alpha},$$

or more specifically their ratio:

$$\frac{\nu_r}{\nu_\theta} = \frac{\sqrt{\ell^2 + \lambda^2}}{\ell}. \tag{3.22}$$

It is obvious that for $\lambda = 0$ the frequencies are commensurate. For $\lambda \neq 0$ the frequencies are generally not commensurate, corresponding to bounded yet non-closed orbits. Hence, the motion on the two-torus $\mathbb{T}^2$ is not periodic but quasi-periodic for the MICZ problem, as it is a completely integrable system. The Kepler–Coulomb problem is special in the sense that its orbits are indeed periodic, that is closed on $\mathbb{T}^2$. For $\ell > 0$ the motion is even degenerate, which is indicative of the conservation of the Laplace–Runge–Lenz vector. The motion in the phase space translates to precessing orbits in reality. The 'failure' of closure of the orbits can be measured by

$$\begin{aligned} |\delta\theta| &= 2\pi \left(\frac{\nu_r}{\nu_\theta} - 1 \right) \\ &= \pi \frac{\lambda^2}{\ell^2} + \mathcal{O}\left(\lambda^4\right). \end{aligned} \tag{3.23}$$

3.2.4 Action–Angle Coordinates

In order to arrive at the Delaunay elements, it is necessary to go to spherical coordinates. In spherical coordinates $\boldsymbol{q} = (r, \theta, \phi)$ the Hamiltonian of the MICZ problem is

$$H = \frac{1}{2} \left(p_r^2 + \frac{p_\theta^2 + \lambda^2}{r^2} + \frac{p_\phi^2}{r^2 \sin^2 \theta} \right) - \frac{k}{r}. \tag{3.24}$$

The aforementioned canonical transformation $p_\theta \mapsto p_\psi = \sqrt{p_\theta^2 + \lambda^2}$ has a generating function

$$S(\boldsymbol{q}, \boldsymbol{p}) = r p_r + \psi p_\psi + \phi p_\phi \tag{3.25}$$

that transforms the Hamiltonian effortlessly into

$$H = \frac{1}{2} \left(p_r^2 + \frac{p_\psi^2}{r^2} + \frac{p_\phi^2}{r^2 \sin^2 \theta} \right) - \frac{k}{r}, \tag{3.26}$$

which, again, is nothing but the Keplerian Hamiltonian in spherical coordinates. The corresponding Hamilton–Jacobi equation remains separable; in Appendix C we show that the classical MICZ Hamiltonian (3.5) is separable in spherical and spheroconical coordinates only. It is then straightforward to derive the

action–angle coordinates on $\mathbb{T}^3$, the three-torus; see for example Cordani [35], Sec. 3.2, or Chang and Marsden [31] for a geometric derivation. The result for the well-known Delaunay variables is

$$\begin{aligned} l &= \mathscr{M}, & \mathcal{L} &= \sqrt{ka}, \\ g &= \omega, & \mathcal{G} &= \sqrt{ka\,(1-e^2)}, \\ h &= \Omega, & \mathcal{H} &= \cos i \sqrt{ka\,(1-e^2)}. \end{aligned}$$

We note that here, as well as in the case of the orbital elements, (i) both the argument of perigee ω and the node Ω are not defined for equatorial orbits ($i = 0$); and (ii) the argument of perigee ω remains undefined for circular orbits ($e = 0$). Evidently, different orbital representations, such as for instance the singularity-free modified equinoctial coordinates or the related canonical Poincaré variables, can be derived from the classical orbital elements and the Delaunay elements without much ado [see e.g. 71].

3.2.5 Perturbations

The orbital motion of an artificial satellite in a low-altitude orbit around the Earth can be attributed to several forces. The primary contribution is due to the Newtonian gravitational attraction, which assumes the Earth and the satellite to be spherically symmetric bodies. The fact that the Earth is in fact not spherically symmetric is demonstrated clearly in Figure 3.2, where the geoid, the gravitational equipotential surface of the Earth, is superimposed in colour on the actual surface of the Earth. It can be modelled by a position-dependent perturbative force, which is derived from the so-called *geopotential*

$$V(\boldsymbol{r}) = -\frac{k}{r}\left\{1 + \sum_{l=2}^{\infty}\sum_{m=0}^{l}\left(\frac{r_\oplus}{r}\right)^l P_{l,m}(\sin\theta)\left[C_{l,m}\cos m\phi + S_{l,m}\sin m\phi\right]\right\}.$$

Here, $\boldsymbol{r} = (r, \theta, \phi)$ are the spherical coordinates of the artificial satellite in a geocentric frame of reference, $r_\oplus$ is the mean equatorial radius of the Earth, and $P_{l,m} = (-1)^m P_l^m(x)$ denote the associated Legendre functions of degree l and order m at $x \in \mathbb{R}$ with $-1 \leq x \leq 1$. It is assumed that the centre of mass coincides with the centre of the Earth, which is certainly a very reasonable approximation for artificial satellites. The coefficients $C_{l,m}$ and $S_{l,m}$ have been determined from measurements up to high degrees and orders [see e.g. 139]. In the astrodynamical literature we often find the definition $J_l = -C_{l,0}$ for the

zonal harmonics, that is terms having $m = 0$. Sectorial harmonics occur at values $l = m$, and tesseral harmonics if $l \neq m$, and $l \neq 0$ and $m \neq 0$.

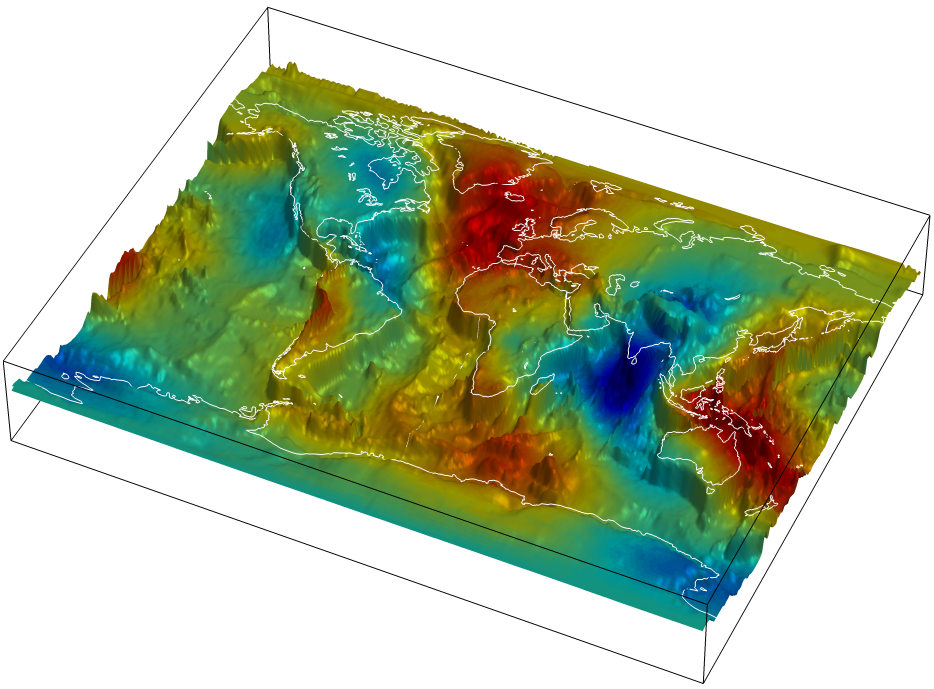


FIGURE 3.2: Approximate geoid as calculated from the data of the Earth Gravitational Model EGM2008 [139], and the topography of the Earth's surface. The elevation runs from $-407\,\mathrm{m}$ below to $8,752\,\mathrm{m}$ above mean sea level [156]. The geoid height is the distance above the reference ellipsoid of the Earth, the World Geodetic System WGS 84 [132] ellipsoid, which is also used for the Global Positioning System (GPS). The geoid height ranges from about $-105\,\mathrm{m}$ (blue) to about $85\,\mathrm{m}$ (red). Coastlines are shown in white.

These spherical harmonics are the (angular) eigenfunctions of Laplace's equation in spherical coordinates, in which we expand the Earth's gravitational field. Formally, the convergence of the infinite sum depends on the values of all coefficients $C_{l,m}$ and $S_{l,m}$. These coefficients are however obviously not known up to arbitrary order and degree. In practice we truncate the infinite sum, so that convergence is not an issue anymore.

Apart from the J_2 term in the geopotential, which is due to the oblateness of the Earth, atmospheric drag affects the orbital motion of artificial satellites near the Earth most. Rayleigh's formula for the drag force is often used as a model for the atmospheric drag of artificial satellites:

$$\boldsymbol{F}_D = -\tfrac{1}{2} C_D A \rho u^2 \hat{\boldsymbol{u}}. \tag{3.27}$$

The drag coefficient C_D measures the resistance of the satellite; it depends on the precise geometry of the satellite. The (local) density of the atmosphere is indicated by ρ, which is generally very difficult to determine, for it depends on many factors. A is the orthographic projection of the satellite on the plane perpendicular to the orbital motion, that is the cross-sectional area, which obviously depends on the attitude of the satellite. The unit velocity of the satellite relative to the atmosphere is denoted by $\hat{\boldsymbol{u}} = \boldsymbol{u}/\|\boldsymbol{u}\|$, where $\boldsymbol{u} = \boldsymbol{v} - \boldsymbol{\omega}_\oplus \wedge \boldsymbol{r}$ with $\boldsymbol{\omega}_\oplus$ the angular velocity of the Earth.

A simplistic yet useful approximation for the atmospheric density as a function of the altitude h_{sat} is given by the barometric formula:

$$\rho = \rho_0 \exp\left(-\beta h_{\text{sat}}\right),$$

where ρ_0 is the density of the air at the surface of the Earth, and β is called the barometric coefficient, which is in principle different for each layer of the Earth's atmosphere. For a comprehensive overview of effects on the local density of the atmosphere we refer to Vallado [157] and references therein.

Third-body perturbations, solar radiation pressure, tidal effects, and the Earth's albedo and infrared radiation all contribute to a lesser extent to the orbital motion of low-altitude satellites. We shall ignore these perturbations henceforth.

Spacecraft entering the Earth's atmosphere from orbit move at hypersonic speeds, which causes enormous aerodynamic heating. Space capsules, such as the ones used in for example the Apollo missions, often have blunt shapes to create shock layers, which dissipate most of the excess of energy to the surrounding air. Space Shuttle orbiters and similar spacecraft require thermal protection systems to guard their interiors from the intense heat. In both cases the flight path and attitude control of the craft are crucial to the success of the re-entry of the craft. In the case of winged spacecraft there is, in addition to the drag force, the lift force perpendicular to the relative velocity between the spacecraft and the atmosphere due to the specific shape of the wings of the spacecraft (aerofoil). Its magnitude is given by

$$F_L = \tfrac{1}{2} C_L S \rho u^2, \tag{3.28}$$

where C_L is the lift coefficient of the craft at a certain angle of attack, and S is the planform area of the craft. Generally, both C_D and C_L depend on the angle of attack, the Reynolds number and the Mach number, but for our purposes it suffices to assume both coefficients to be constant.

3.3 Rigid Body Motion

We shall now focus on the attitude dynamics of spacecraft. In Section 3.3.1 Euler's equations are presented, from which the torque-free motion of a rigid body is derived in Section 3.3.2. An analytical solution in terms of Jacobi's elliptic functions is given; it constitutes the benchmark for the numerical solutions obtained with the implicit midpoint method, to be discussed in depth in Chapter 4. The geometric structure of the torque-free rigid body motion is reviewed briefly in Section 3.3.4.

3.3.1 Euler's Equations

In order to describe the motion of a satellite completely, we specify two complementary frames of reference. First, a non-inertial frame that co-orbits with the satellite, which we call the body frame $\mathcal{B}$, determines the relative orientation with respect to its pivot, the origin of the frame. In it, all coordinates on the body are fixed, so that $\mathcal{B}$ is not adequate to describe the time evolution. Second, an inertial frame, the space frame $\mathcal{S}$, accounts fully for the dynamics. Its origin is chosen to coincide with the centre of the Earth.

In $\mathcal{B}$ the moment of inertia tensor I is symmetric and time-independent, so that there is an orthogonal transformation that diagonalizes it. The coordinate system in which $\mathsf{I} = \operatorname{diag}(I_1, I_2, I_3)$ is known as the principal-axes system. In addition, we assume that $I_1 > I_2 > I_3$. For $\boldsymbol{\omega} = (\omega_1, \omega_2, \omega_3) \in \mathcal{B}$, the angular velocity vector and the instantaneous rate of rotation of the satellite are encoded in Euler's equations [see e.g. 81]:

$$\boldsymbol{N} = \boldsymbol{\omega} \wedge (\mathsf{I}\boldsymbol{\omega}) + \mathsf{I}\dot{\boldsymbol{\omega}}, \tag{3.29}$$

where $\boldsymbol{N} = (N_1, N_2, N_3)$ is the torque applied to the satellite. Henceforth the dependence of the angular velocity on time is understood.

3.3.2 The Euler–Poinsot Problem

In the case of torque-free motion the components of equation (3.29) read

$$\begin{cases} I_1\dot{\omega}_1 &= (I_2 - I_3)\omega_2\omega_3, \\ I_2\dot{\omega}_2 &= (I_3 - I_1)\omega_3\omega_1, \\ I_3\dot{\omega}_3 &= (I_1 - I_2)\omega_1\omega_2, \end{cases} \tag{3.30}$$

which are to be supplied with initial conditions for $\boldsymbol{\omega}$. This system of first-order non-linear differential equations is known as the Euler–Poinsot problem. Let us write $\boldsymbol{L} = \mathsf{I}\boldsymbol{\omega}$ for the angular momentum in the body frame. In terms of the body angular momentum the square of the angular momentum is $L^2 \equiv \|\boldsymbol{L}\|^2 = (\mathsf{I}\boldsymbol{\omega}) \cdot (\mathsf{I}\boldsymbol{\omega})$, and the kinetic energy is $T = \frac{1}{2}\boldsymbol{L} \cdot (\mathsf{I}^{-1}\boldsymbol{L})$ are constants of motion.

Euler's equations (3.30) are known to have an analytical solution in terms of the Jacobian elliptic functions $\operatorname{sn}(u \mid m)$, $\operatorname{cn}(u \mid m)$ and $\operatorname{dn}(u \mid m)$, where $u \in \mathbb{R}$ is the argument, and the parameter lies in the range $0 < m < 1$. The solution reads

$$\begin{cases} \omega_1(t) &= \quad \alpha \operatorname{cn}(p(t - t_0) \mid m), \\ \omega_2(t) &= -\beta \operatorname{sn}(p(t - t_0) \mid m), \\ \omega_3(t) &= \quad \gamma \operatorname{dn}(p(t - t_0) \mid m), \end{cases} \tag{3.31}$$

where we opt to set $t_0 = 0$, so that necessarily $\omega_2(0) = 0$. The remaining five parameters are given by, for instance,

$$\begin{cases} \alpha^2 &= \dfrac{L^2 - 2I_3T}{I_1(I_1 - I_3)}, \\ \beta^2 &= \dfrac{L^2 - 2I_3T}{I_2(I_2 - I_3)}, \\ \gamma^2 &= \dfrac{2I_1T - L^2}{I_3(I_1 - I_3)}, \\ p^2 &= \dfrac{(I_2 - I_3)(2I_1T - L^2)}{I_1I_2I_3}, \\ m^2 &= \dfrac{(I_1 - I_2)(L^2 - 2I_3T)}{(I_2 - I_3)(2I_1T - L^2)}, \end{cases} \tag{3.32}$$

with $I_1 > L^2/2T > I_3$, in which we have incorporated the initial conditions by using the constants of motion L^2 and T at any convenient time, say $t = 0$. For an alternate definition of the parameters in terms of the initial conditions see for example Armitage and Eberlein [4], pp. 339–342.

The form of the solution depends on the magnitude of $L^2/2T$ in relation to the components of I. If $I_2 > L^2/2T > I_3$ then $0 < m < 1$, which lies in the

principal range of the elliptic parameter. However, for $I_1 > L^2/2T > I_2$ we have that $m > 1$, in which case we may use the following identities [2, p. 573] to scale the parameter inside the principal range:

$$\begin{aligned}\sqrt{m}\,\mathrm{sn}\,(u \mid m) &= \mathrm{sn}\left(\sqrt{m}u \mid m^{-1}\right),\\ \mathrm{cn}\,(u \mid m) &= \mathrm{dn}\left(\sqrt{m}u \mid m^{-1}\right),\\ \mathrm{dn}\,(u \mid m) &= \mathrm{cn}\left(\sqrt{m}u \mid m^{-1}\right).\end{aligned}$$

Finally, for $I_2 = L^2/2T$, corresponding to the situation in which $m = 1$, the solution simplifies slightly, as $\mathrm{sn}\,(u \mid 1) = \tanh u$, $\mathrm{cn}\,(u \mid 1) = \mathrm{dn}\,(u \mid 1) = \operatorname{sech} u$ and β^2 reduces to $4T^2/L^2$.

We note that the signs assigned to the individual components of the angular velocity vector are arbitrary in the sense that the only requirement is that the product $\alpha\beta\gamma < 0$. We have chosen the negative sign in the second component because $I_3 - I_2 < 0$ on the right-hand side of the middle equation in (3.30).

3.3.3 Body and Space Coordinates

So far we have only looked at the rotational motion in the $\mathcal{B}$. In order to find the attitude with respect to the space frame, it is necessary to transform the coordinates in the body frame to the space frame. Let $\boldsymbol{e}^{(1)}$, $\boldsymbol{e}^{(2)}$, $\boldsymbol{e}^{(3)}$ denote the unit vectors along the principal axes in the stationary frame $\mathcal{S}$. First, let us assume that there are no external forces that act on the body and that the origin of $\mathcal{B}$ remains stationary relative to $\mathcal{S}$, in which instance there are no external torques either. Then we may choose the coordinate systems $\mathcal{B}$ and $\mathcal{S}$ to coincide at $t = 0$, so that the matrix

$$\mathbf{M} = \begin{pmatrix} \vdots & \vdots & \vdots \\ \boldsymbol{e}^{(1)} & \boldsymbol{e}^{(2)} & \boldsymbol{e}^{(3)} \\ \vdots & \vdots & \vdots \end{pmatrix}. \tag{3.33}$$

transforms the coordinates from the body frame to the space frame. The column entries are calculated from

$$\dot{\boldsymbol{e}}^{(i)} = -\boldsymbol{\omega} \wedge \boldsymbol{e}^{(i)}, \quad i = 1, 2, 3, \tag{3.34}$$

which states that the body, along with its principal axes, rotates relative to the space frame with angular velocity $\boldsymbol{\omega}$. The direction cosine matrix $\mathbf{M} \in \mathrm{SO}\,(3, \mathbb{R})$,

so that it has three degrees of freedom. In the literature we often see the matrix defined in terms of Euler (roll, pitch and yaw) angles or quaternions; see for instance Hairer et al. [66], and Marsden and Ratiu [106] for more details, or Arribas et al. [6], who use quaternions to derive the Hamiltonian formalism for rigid bodies with external torques. From Euler's equations (3.30) and the kinematic equations for the principal axes (3.34) the attitude dynamics can be computed. Notice that the motion is constrained to the sphere $\mathbb{S}^2$ with radius $\|\boldsymbol{L}_{\mathcal{S}}\|$, since $\boldsymbol{L}_{\mathcal{S}}$ is conserved in $\mathcal{S}$ in the absence of external torques due to the fact that $\boldsymbol{N} \equiv \dot{\boldsymbol{L}}_{\mathcal{S}} = 0$. The subscript $\mathcal{S}$ indicates that the quantity is represented in the space coordinates. In the body system the body angular momentum is generally not conserved; it is only conserved provided that $\boldsymbol{\omega}$ and $\boldsymbol{L}$ are parallel.

Now we presume that the body coordinate system is moving relative to the frame frozen in space with velocity $\boldsymbol{v}(t) = \dot{\boldsymbol{x}}(t)$ as a consequence of external forces. The torque induced by the external force affects the rotational motion within $\mathcal{B}$, whereas the force itself alters the motion of the body coordinate system as a whole. Furthermore, we demand that the rotational motion decouples from the translational motion. Henceforth $\boldsymbol{e}^{(1)}$ points towards the orbital motion, $\boldsymbol{e}^{(2)}$ in the direction opposite to the normal of the orbital plane, and $\boldsymbol{e}^{(3)}$ towards the centre of the Earth, as shown schematically in Figure 3.3. The body coordinate system defined in this particular way is right-handed. The radial unit vector in the space frame and $\boldsymbol{e}^{(3)}$ are anti-parallel, and so are the angular momentum vector in the space frame and $\boldsymbol{e}^{(2)}$. The equations for the principal axes in $\mathcal{S}$ then read

$$\dot{\boldsymbol{e}}^{(i)} = -\boldsymbol{\omega} \wedge \boldsymbol{e}^{(i)} + \boldsymbol{v}^{(i)}, \quad i = 1, 2, 3, \tag{3.35}$$

in which $\boldsymbol{v}^{(i)}$ describes the rate of change due to the translational motion of the ith principal axis in the inert frame of reference $\mathcal{S}$. In the case of a satellite moving with angular velocity $\boldsymbol{\Omega}$ in orbit around the Earth we have that the velocity $\boldsymbol{v}^{(i)} = -\boldsymbol{\Omega} \wedge \boldsymbol{e}^{(i)}$.

The transformation from the body frame to the space frame is now accomplished by an orientation-preserving map of the Euclidean space, that is a translation $\boldsymbol{x}(t) \in \mathbb{R}^3$ of the body frame with respect to the space frame and a rotation $\mathbf{M} \in \mathrm{SO}(3, \mathbb{R})$ of the body frame relative to its reference position:

$$\boldsymbol{r} \mapsto \mathbf{M}(t)\boldsymbol{r} + \boldsymbol{x}(t) \tag{3.36}$$

for $\boldsymbol{r} \in \mathcal{B}$.

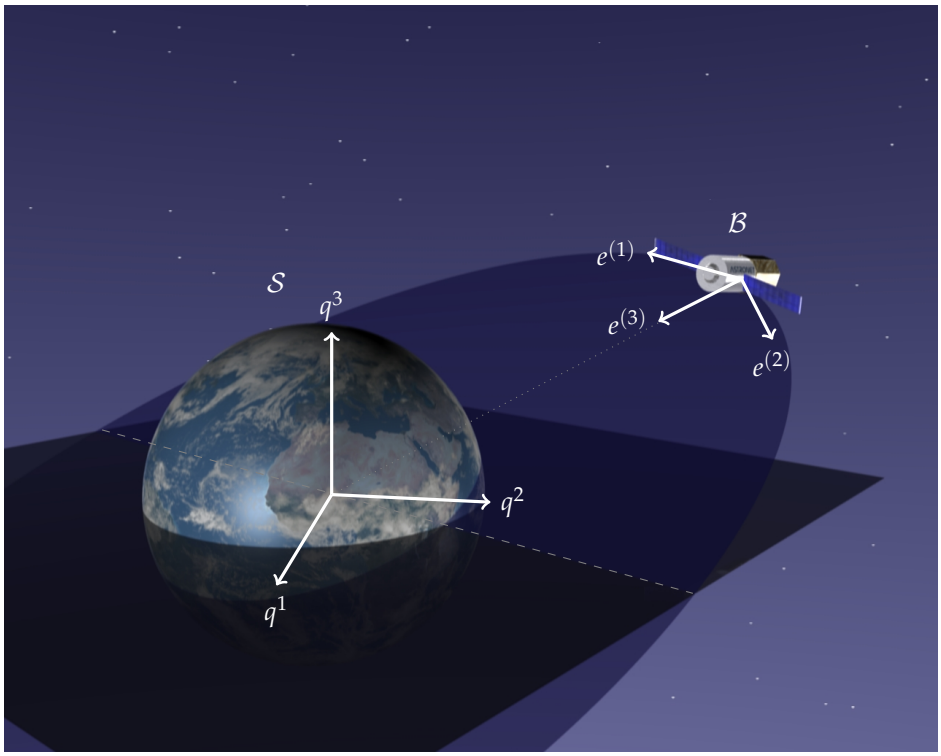


FIGURE 3.3: Graphical representation of the body and space coordinate systems $\mathcal{B}$ and $\mathcal{S}$.

3.3.4 REPRISE OF THE EULER–POINSOT PROBLEM

Euler's equations (3.30) are actually the result of a reduction process from a Hamiltonian dynamical system constrained to the Lie group $\mathrm{SO}(3,\mathbb{R})$ [106], and they describe what is known as a Lie–Poisson dynamical system. We do not wish to go into the full details of Lie–Poisson mechanics but merely state that it formalizes the by now familiar Hamiltonian formulation of classical mechanics. In fact, Lie–Poisson mechanics can even be generalized further by what is known as Nambu mechanics [130], but we shall have no need of such an abstraction.

The key difference between Hamiltonian and Lie–Poisson dynamical systems is the carrier space of the dynamics; Poisson manifolds appear as a natural generalization of symplectic manifolds. In short, a Poisson manifold is a smooth

manifold $\mathcal{P}$ that is equipped with a Poisson structure, that is a Lie algebra of $\mathcal{C}^\infty(\mathcal{P},\mathbb{R})$ under the Poisson bracket. In other words, $\mathcal{P}$ has an anti-symmetric bilinear map $\{\cdot,\cdot\}:\mathcal{C}^\infty(\mathcal{P},\mathbb{R})\times\mathcal{C}^\infty(\mathcal{P},\mathbb{R})\to\mathcal{C}^\infty(\mathcal{P},\mathbb{R})$ that satisfies the Jacobi identity and is a derivation on the ring $\mathcal{C}^\infty(\mathcal{P},\mathbb{R})$ for its first argument. It is obvious that the Poisson bracket from equation (1.16) is such a Poisson bracket on the function space $\mathcal{C}^\infty(\mathbf{T}^\star Q,\mathbb{R})$. Any symplectic manifold is therefore a Poisson manifold.

We can take a shortcut by looking at Hamilton's canonical equations as expressed in equation (1.15): $\dot{\boldsymbol{\eta}}=\mathbf{J}\nabla H(\boldsymbol{\eta})$, where $\boldsymbol{\eta}\in\mathcal{M}$, a $2n$-dimensional symplectic manifold $(\mathcal{M},\omega)$ with $\omega=\frac{1}{2}\mathrm{d}\boldsymbol{\eta}\wedge\mathbf{J}\mathrm{d}\boldsymbol{\eta}$. If we replace the symplectic matrix $\mathbf{J}$ with a coordinate dependent skew-symmetric matrix $\mathbf{J}(\boldsymbol{\eta})$, then we end up with an example of Lie–Poisson system on a Poisson manifold, for which

$$\dot{\boldsymbol{\eta}}=\mathbf{J}(\boldsymbol{\eta})\nabla H(\boldsymbol{\eta}), \tag{3.37}$$

as we shall see below.

Indeed, for any Lie algebra $\mathfrak{g}$ its dual $\mathfrak{g}^\star$ defines a Poisson structure with respect to the Lie–Poisson bracket

$$\{f,g\}_\pm(\Xi)=\pm\left\langle\Xi,\left[\frac{\delta f}{\delta\Xi},\frac{\delta g}{\delta\Xi}\right]\right\rangle. \tag{3.38}$$

Here $\Xi\in\mathfrak{g}^\star$, $f,g\in\mathcal{C}^\infty(\mathfrak{g}^\star,\mathbb{R})$, and $\delta f/\delta\Xi\in\mathfrak{g}$ denotes the functional derivative of f at Ξ. Recall that the adjoint representation of $\mathfrak{g}$ is a linear map $\mathrm{ad}:\mathfrak{g}\to\mathrm{End}(\mathfrak{g})$, $\xi\mapsto\mathrm{ad}_\xi$, which is defined as $\mathrm{ad}_\xi(\eta)=[\xi,\eta]$, the Lie bracket, for all $\xi,\eta\in\mathfrak{g}$. Let $\dim\mathfrak{g}$ be r, and $e_1,\dots,e_r$ denote a basis for $\mathfrak{g}$. In this basis the Lie bracket becomes $[e_i,e_j]=c_{ij}^k e_k$, where the c_{ij}^k are known as the structure constants. We write $e^1,\dots,e^r$ for the corresponding dual basis elements of $\mathfrak{g}^\star$. Now we are able write the Lie–Poisson bracket (3.38) as

$$\{f,g\}_\pm(\Xi)=\pm c_{ij}^k\Xi_k\frac{\delta f}{\delta\Xi_i}\frac{\delta g}{\delta\Xi_j}, \tag{3.39}$$

where $\Xi=\Xi_i e^i$. Lie–Poisson brackets result from canonical Poisson brackets on the phase space $\mathbf{T}^\star G$ related to a certain Lie group G. In fact, any Lie–Poisson bracket can be obtained from a Poisson bracket by reduction via a momentum map [106].

In finite dimensions we can identify $\mathfrak{g}\cong\mathbb{R}^n$ and $\mathfrak{g}^\star\cong\mathbb{R}^n$, so that $\delta/\delta\Xi$ is $\partial/\partial\boldsymbol{\eta}=\nabla$, and the above construction boils down to the Lie–Poisson bracket

$$\{f,g\}_\pm(\boldsymbol{\eta})=\pm\left\langle\nabla f(\boldsymbol{\eta}),\mathbf{J}(\boldsymbol{\eta})\nabla g(\boldsymbol{\eta})\right\rangle, \tag{3.40}$$

where $\mathbf{J}(\boldsymbol{\eta})$ is a skew-symmetric matrix, which is a consequence of the fact that the Lie bracket $[\cdot,\cdot]$ is anti-symmetric in its arguments. $\mathbf{J}(\boldsymbol{\eta})$ is sometimes referred to as the structure matrix.

A numerical integrator is called a Lie–Poisson integrator if its discrete flow is a Poisson map, which depends of course on the structure matrix in question. An algorithm can only be a Lie–Poisson integrator for a specific class of structure matrices. Lower-order Lie–Poisson integrators can generally not be combined to form higher-order methods as in the case of symplectic integrators. For a review of Lie–Poisson integrators we refer the reader to Karasözen [84] and references therein. Before we proceed with the discussion of the Lie–Poisson structure of the Euler–Poinsot problem, we note that Lie–Poisson dynamical systems have a variational structure too. For more details on variational Lie–Poisson integration algorithms we refer to the literature [22, 29, 104].

We shall now demonstrate that the Euler–Poinsot problem (3.30) can be brought into a Lie–Poisson form (3.37). The Lie group that leaves the angular-momentum sphere invariant is $G = \mathrm{SO}(3,\mathbb{R})$. As before, we may identify $\mathfrak{so}(3)$ with $\mathbb{R}^3$, and a natural pairing between $\mathfrak{so}(3)^\star$ and $\mathfrak{so}(3)$ is then given by the inner product. All in all, the relevant Lie–Poisson bracket with respect to $\boldsymbol{L}$ becomes

$$\{f,g\}_{\pm}(\boldsymbol{L}) = \pm\boldsymbol{L}\cdot\big(\nabla f(\boldsymbol{L})\wedge\nabla g(\boldsymbol{L})\big),$$

so that the time evolution of any real scalar function f is given by $\dot{f} = \{f,\mathscr{H}\}_{\pm}$ with $\mathscr{H}\colon G \to \mathbb{R}$ the Lie–Poisson Hamiltonian. For Euler's equations the Lie–Poisson bracket carries a minus sign, as we show below.

Let us define the Lagrangian $\mathscr{L} = \frac{1}{2}\boldsymbol{L}\cdot\boldsymbol{L}$, which is one-half of the space angular momentum, and the (free) Hamiltonian by $\mathscr{H} = \frac{1}{2}\boldsymbol{L}\cdot(\mathsf{I}^{-1}\boldsymbol{L})$. Euler's equations (3.30) can be written as

$$\dot{\boldsymbol{L}} = -\nabla\mathscr{L}\wedge\nabla\mathscr{H}. \tag{3.41}$$

Alternatively, by taking the inner product with $\boldsymbol{L}$ we find that

$$\dot{\mathscr{L}} = -\boldsymbol{L}\cdot\big(\nabla\mathscr{L}\wedge\nabla\mathscr{H}\big) = 0,$$

which indeed confirms that the correct Lie–Poisson bracket bears a minus sign for the Euler–Poinsot problem. Furthermore, it shows that in addition to the trivial Casimir $\mathscr{H}$, $\mathscr{L}$ is a Casimir function too. Similarly, we write for the Euler–Poinsot problem

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{pmatrix} L_1\\ L_2\\ L_3\end{pmatrix} = \begin{pmatrix} 0 & -L_3 & L_2\\ L_3 & 0 & -L_1\\ -L_2 & L_1 & 0\end{pmatrix}\begin{pmatrix} L_1/I_1\\ L_2/I_2\\ L_3/I_3\end{pmatrix}. \tag{3.42}$$

Any Poisson manifold can be foliated by a collection of symplectic submanifolds, or leaves as they are called in this context; these leaves are the coadjoint orbits [106]. On these leaves any Casimir function C, for which $\{C, f\}_{\pm} = 0$ for any f, is constant. The symplectic leaves are therefore spheres with radius $\|\boldsymbol{L}\|$ centred at the origin.

3.3.5 Gravity-Gradient Torque

The gravitational force of the Earth applied to the centre of mass of the satellite does not exert a torque. The gravitational force does, however, create a torque around the centre of mass of the satellite, since the force is generally not uniform over the dimensions of the craft. This torque is known as the gravity-gradient torque and it is especially important in the attitude dynamics for low-altitude orbits. Here we tacitly assume the Earth to be spherical, so that the non-uniformity of the gravitational force over the satellite is not the result of local variations in the Earth's gravitational field but entirely due to the fact that the various parts of the satellite can be at slightly different distances from the centre of the Earth. Such irregularities in the geopotential can of course be taken into account as well, but for our present discussion they are negligible.

Let k be the gravitational parameter of the Earth, m the mass of the satellite, $\boldsymbol{r}_{\mathrm{CM}}$ the position vector from the Earth's core to the centre of mass of the satellite, and $\boldsymbol{r}_{\mathrm{sat}}$ the position vector from the centre of mass of the satellite to a point on the satellite. Furthermore, we define $\boldsymbol{r} = \boldsymbol{r}_{\mathrm{CM}} + \boldsymbol{r}_{\mathrm{sat}}$. Please recall that the radial unit vector $\boldsymbol{r}/\|\boldsymbol{r}\|$ is opposite to $\boldsymbol{e}^{(3)}$. The gravitational force on a point on the satellite with an infinitesimal mass element $\mathrm{d}m$ at $\boldsymbol{r}$ is

$$\begin{aligned} \mathrm{d}\boldsymbol{F} &= -\frac{k\boldsymbol{r}\,\mathrm{d}m}{\|\boldsymbol{r}\|^3} \\ &= -\frac{k\,\mathrm{d}m\,(\boldsymbol{r}_{\mathrm{CM}} + \boldsymbol{r}_{\mathrm{sat}})}{\|\boldsymbol{r}_{\mathrm{CM}} + \boldsymbol{r}_{\mathrm{sat}}\|^3}. \end{aligned} \tag{3.43}$$

The gravity-gradient torque about the centre of mass of the satellite thus becomes

$$\begin{aligned} \boldsymbol{N} &= \int \boldsymbol{r}_{\mathrm{sat}} \wedge \mathrm{d}\boldsymbol{F} \\ &= -k \int \frac{\boldsymbol{r}_{\mathrm{sat}} \wedge \boldsymbol{r}_{\mathrm{CM}}}{\|\boldsymbol{r}_{\mathrm{CM}} + \boldsymbol{r}_{\mathrm{sat}}\|^3}\,\mathrm{d}m. \end{aligned}$$

Since clearly $\|\boldsymbol{r}_{\mathrm{CM}}\| \gg \boldsymbol{r}_{\mathrm{sat}}$, we expand the denominator in the integrand up to second order in $\|\boldsymbol{r}_{\mathrm{sat}}/\boldsymbol{r}_{\mathrm{CM}}\|^2$:

$$\frac{1}{\|\boldsymbol{r}_{\mathrm{CM}}+\boldsymbol{r}_{\mathrm{sat}}\|^3} \simeq \frac{1}{r_{\mathrm{CM}}^3}\left[1-3\frac{\boldsymbol{r}_{\mathrm{sat}}\cdot\boldsymbol{r}_{\mathrm{CM}}}{r_{\mathrm{CM}}^2}\right],$$

where $r_{\mathrm{CM}} = \|\boldsymbol{r}_{\mathrm{CM}}\|$. Because $\int \boldsymbol{r}_{\mathrm{sat}}\,\mathrm{d}m = 0$, we end up with

$$\boldsymbol{N} = -\frac{3k}{r_{\mathrm{CM}}^5}\,\boldsymbol{r}_{\mathrm{CM}} \wedge \int \boldsymbol{r}_{\mathrm{sat}}\,(\boldsymbol{r}_{\mathrm{sat}}\cdot\boldsymbol{r}_{\mathrm{CM}})\,\mathrm{d}m.$$

After some simple manipulations and the introduction of the moment of inertia tensor $\mathsf{I} = \int \left(r_{\mathrm{sat}}^2 1 - \boldsymbol{r}_{\mathrm{sat}}\boldsymbol{r}_{\mathrm{sat}}\right)\,\mathrm{d}m$, in which 1 is the identity tensor, $r_{\mathrm{sat}} = \|\boldsymbol{r}_{\mathrm{sat}}\|$ and $\boldsymbol{r}_{\mathrm{sat}}\boldsymbol{r}_{\mathrm{sat}}$ is the dyadic product, we are left with the required expression for the gravity-gradient torque about the centre of mass of the satellite:

$$\begin{aligned}\boldsymbol{N} &= \frac{3k}{r_{\mathrm{CM}}^5}\,\boldsymbol{r}_{\mathrm{CM}} \wedge (\mathsf{I}\boldsymbol{r}_{\mathrm{CM}}) \\ &= \frac{3k}{r_{\mathrm{CM}}^3}\,\boldsymbol{e}^{(3)} \wedge \left(\mathsf{I}\boldsymbol{e}^{(3)}\right).\end{aligned} \tag{3.44}$$

For satellites in circular orbits $\boldsymbol{\Omega} = -n\boldsymbol{e}^{(2)}$, where $n = \sqrt{k/r_{\mathrm{CM}}^3}$ is the mean motion. The attitude dynamics follows from Euler's equations in the orbital body system

$$\left\{\begin{array}{rcl} I_1\dot{\omega}_1 - (I_2-I_3)\omega_2\omega_3 &=& -3n^2(I_2-I_3)e_2^{(3)}e_3^{(3)}, \\ I_2\dot{\omega}_2 - (I_3-I_1)\omega_1\omega_3 &=& -3n^2(I_3-I_1)e_1^{(3)}e_3^{(3)}, \\ I_3\dot{\omega}_3 - (I_1-I_2)\omega_1\omega_2 &=& -3n^2(I_1-I_2)e_1^{(3)}e_2^{(3)}, \end{array}\right. \tag{3.45}$$

or with $i = 1,2,3$,

$$I_i\dot{\omega}_i = \sum_{j,k=1}^{3}\epsilon^{ijk}\left[I_j\left(\omega_j\omega_k - 3n^2e_j^{(3)}e_k^{(3)}\right)\right], \tag{3.46}$$

for short, as well as the equations for the principal axes in the inertial frame of reference

$$\left\{\begin{array}{rcl} \dot{\boldsymbol{e}}^{(1)} &=& -\boldsymbol{\omega}\wedge\boldsymbol{e}^{(1)} + n\boldsymbol{e}^{(3)}, \\ \dot{\boldsymbol{e}}^{(2)} &=& -\boldsymbol{\omega}\wedge\boldsymbol{e}^{(2)}, \\ \dot{\boldsymbol{e}}^{(3)} &=& -\boldsymbol{\omega}\wedge\boldsymbol{e}^{(3)} - n\boldsymbol{e}^{(1)}. \end{array}\right. \tag{3.47}$$

We note that for elliptical orbits the mean motion does not represent the instantaneous angular rate of the satellite in orbit.

Realizing that $\boldsymbol{e}^{(1)} = \boldsymbol{e}^{(2)} \wedge \boldsymbol{e}^{(3)}$, we can reduce the coupled system (3.47) by dropping the differential equation for $\boldsymbol{e}^{(1)}$:

$$\begin{cases} \dot{\boldsymbol{e}}^{(2)} &= -\boldsymbol{\omega} \wedge \boldsymbol{e}^{(2)}, \\ \dot{\boldsymbol{e}}^{(3)} &= -\left[\boldsymbol{\omega} + n\boldsymbol{e}^{(2)}\right] \wedge \boldsymbol{e}^{(3)}. \end{cases} \tag{3.48}$$

Once $\boldsymbol{e}^{(2)}$ and $\boldsymbol{e}^{(3)}$ are known, $\boldsymbol{e}^{(1)}$ can easily be computed from the exterior product. From a numerical point of view it is important to note that the inner products

$$\boldsymbol{e}^{(i)} \cdot \boldsymbol{e}^{(j)} = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases} \tag{3.49}$$

are integrals of motion for equations (3.48) but not for (3.47). We can easily verify that the time derivatives of the geometric products (3.49) are generally non-zero for the system (3.47), whereas the expressions are identically zero for the reduced equations. This means that the reduced equations are numerically stable, and they are to be preferred; the orthogonality of the attitude matrix is preserved by the reduced system only.

Due to the gravity-gradient torque the kinetic energy and the square of the angular momentum are not conserved anymore:

$$\dot{T} = -3n^2 \sum_{i,j,k=1}^{3} \epsilon^{ijk} \omega_i I_j e_j^{(3)} e_k^{(3)},$$

$$\dot{L}^2 = -6n^2 \sum_{i,j,k=1}^{3} \epsilon^{ijk} I_i \omega_i I_j e_j^{(3)} e_k^{(3)}.$$

We have the following Jacobian integral of motion

$$E = \tfrac{1}{2}\boldsymbol{\omega} \cdot (\mathsf{I}\boldsymbol{\omega}) + n\boldsymbol{e}^{(2)} \cdot (\mathsf{I}\boldsymbol{\omega}) + \tfrac{3}{2}n^2 \boldsymbol{e}^{(3)} \cdot \left(\mathsf{I}\boldsymbol{e}^{(3)}\right), \tag{3.50}$$

for circular orbits instead. It represents the energy in a co-rotating frame, where the second term is nothing but the second component of the angular momentum L_2, that is the component perpendicular to the orbital plane, multiplied by the mean motion.

We end our presentation of astrodynamics by mentioning that the non-linear system (3.45) and (3.47) is not of the form (3.37), and hence does not constitute a Lie–Poisson system.

Chapter 4

Applications and Algorithms

Now that we have established and explored the differential-geometric structure of the equations of motion of artificial satellites and spacecraft in the previous chapters, it is time to turn our attention and efforts to applications and the algorithms to solve these numerically. We shall pursue three main lines in this chapter.

First, we describe three prominent applications in celestial mechanics and astrodynamics of quasi-Keplerian motion, for which we have derived the universal solution in Chapter 3: the relativistic precession of the apsides, the numerical integration of perturbed Kepler–Coulomb problems with a generalized leapfrog, and the averaged motion of Earth-orbiting satellites with the J_2 perturbation.

Second, a new algorithm is presented for the numerical integration of second-order ordinary differential equations with perturbations that depend on the first derivative of the dependent variables with respect to the independent variable; it is especially designed for few-body problems with velocity-dependent perturbations. The algorithm can be used within extrapolation methods for enhanced accuracy, and it is fully explicit, which makes it a competitive alternative to standard discretization methods.

Third, we describe the application of the implicit midpoint integrator to the problem of attitude dynamics for low-altitude satellites without the use of

quaternions. Initially, we consider the satellite to rotate without external torques applied to it. We compare the numerical solution with the exact solution in terms of Jacobi's elliptic functions, as given in Chapter 3. Afterwards we include the gravity-gradient torque, in which case the implicit midpoint integrator proves to be a fast, simple and accurate method. Higher-order versions of the implicit midpoint scheme are compared to Gauss–Legendre Runge–Kutta methods in terms of accuracy and CPU time. Finally, we investigate the performance of a parameter-adaptive Kalman filter based on the implicit midpoint integrator for the determination of the principal moments of inertia through observations.

4.1 Quasi-Keplerian Motion

The MICZ problem provides a template for several problems in physics that range from the motion of charged particles around (hypothetical) magnetic monopoles to the precession of planets due to general-relativistic corrections to Newtonian gravity. In Sections 4.1.1 through to 4.1.3 we describe three such applications of quasi-Keplerian motion from the realms of celestial mechanics and astrodynamics.

4.1.1 General-Relativistic Precession

The observation that an inverse-square potential gives rise to precession of the orbit is a well-established fact. In fact, the perihelion precession of the planet Mercury was one of three tests put forward by Einstein to examine the validity of the general theory of relativity experimentally, apart from the predicted deflection of light by the Sun due to the curvature of space-time near massive bodies, and the gravitational redshift of electromagnetic waves. The precession of the apsides of any satellite in its broadest sense, can be attributed to lowest-order by an inverse-square potential added to the Newtonian one [144, Sec. 11.9]:

$$V(\boldsymbol{q}) = -\frac{GM}{\|\boldsymbol{q}\|}\left(1 - \frac{3GM}{c^2\|\boldsymbol{q}\|}\right), \tag{4.1}$$

which obviously corresponds to the MICZ potential if we take $k = GM$ and $\lambda^2 = -6G^2M^2/c^2$. With these identifications the relativistic precession of the periapsis can be approximated by the quasi-Keplerian motion. The precession angle can be calculated to lowest order from equation (3.23):

$$|\delta\theta| = \frac{6\pi GM}{c^2 a(1-e^2)}, \tag{4.2}$$

where we have used that $L^2 = \ell^2 + \lambda^2 = ak(1-e^2)$. Indeed, equation (4.2) is to lowest order in perfect agreement with the general-relativistic expression [124, p. 1110], since $a_{\mathrm{K}} = a + \mathcal{O}\left(\lambda^2\right)$ and $e_{\mathrm{K}}^2 = e^2 + \mathcal{O}\left(\lambda^2\right)$. The subscripts K distinguish the classical Keplerian definitions from the ones defined in Section 3.2.3. The fact that the quasi-Keplerian motion mimics the lowest-order general-relativistic contribution to Newtonian gravity enables us to use the solution described in Section 3.2.2.

4.1.2 Numerics in the Extended Phase Space

Consider a Hamiltonian H that consists of a Keplerian part K and a coordinate-dependent perturbation R, that is,

$$\begin{aligned} H(\boldsymbol{q},\boldsymbol{p}) &= \frac{1}{2}\|\boldsymbol{p}\|^2 - \frac{k}{\|\boldsymbol{q}\|} - R(\boldsymbol{q}) \\ &\equiv K - R. \end{aligned} \tag{4.3}$$

Since a generic perturbation generally destroys the integrability of the Kepler–Coulomb Hamiltonian, one could solve Hamilton's equations numerically with the Störmer–Verlet algorithm (2.19) for instance, which implements the natural separation of the Hamiltonian (4.3) into K and R automatically:

$$\begin{aligned} \boldsymbol{q}_{k+1/2} &= \boldsymbol{q}_n + \frac{h}{2}\frac{\partial H}{\partial \boldsymbol{p}}(\boldsymbol{p}_k, \boldsymbol{q}_k) \\ &= \boldsymbol{q}_k + \frac{h}{2}\boldsymbol{p}_k, \end{aligned} \tag{4.4a}$$

$$\begin{aligned} \boldsymbol{p}_{k+1} &= \boldsymbol{p}_k - h\frac{\partial H}{\partial \boldsymbol{q}}(\boldsymbol{p}_k, \boldsymbol{q}_{k+1/2}) \\ &= \boldsymbol{p}_k + h\left[k\frac{\boldsymbol{q}_{k+1/2}}{q_{n+1/2}^3} + \frac{\partial R}{\partial \boldsymbol{q}}(\boldsymbol{q}_{k+1/2})\right] \end{aligned} \tag{4.4b}$$

$$\begin{aligned} \boldsymbol{q}_{k+1} &= \boldsymbol{q}_{k+1/2} + \frac{h}{2}\frac{\partial H}{\partial \boldsymbol{p}}(\boldsymbol{p}_{k+1}, \boldsymbol{q}_{k+1/2}) \\ &= \boldsymbol{q}_{k+1/2} + \frac{h}{2}\boldsymbol{p}_{k+1}. \end{aligned} \tag{4.4c}$$

The coordinates and momenta are calculated at steps $k = 0, \ldots, N-1$ with a time step h. In the generalized leapfrog scheme the Keplerian part and the perturbation are to be split completely, which means that they are integrated separately. Because the perturbation is independent of the momenta by assumption,

it only affects the momentum updates; initially the coordinates and momenta are both advanced half an integration step according to the unperturbed Keplerian solution, then the full contribution to the momenta due to the perturbation is added, whereupon the coordinates and momenta are updated once more over half a step.

Whether a particular Hamiltonian system can easily be integrated numerically depends on the characteristics of the problem under consideration. Apt choices for so-called time transformations are sometimes known to aid computations. In fact, there is a class of time transformations that preserve the integrability of the Keplerian part, which is transformed into a MICZ Hamiltonian, as suggested first by Mikkola [118]. The idea is to move to the extended phase space of Section 1.3.1.

Let the time depend on some parameter s, so $t = t(s)$, or more generally

$$\mathrm{d}t = g(\bar{\boldsymbol{q}}, \bar{\boldsymbol{p}})\mathrm{d}s,$$

where $\bar{\boldsymbol{q}} = (q^0, \boldsymbol{q})$ and $\bar{\boldsymbol{p}} = (p_0, \boldsymbol{p})$. The choice of a particular time step function g is crucial to the success of the integrator. For instance, it is known that the so-called logarithmic Hamiltonian produces the exact (regularized) trajectories for the Kepler–Coulomb problem [122, 123, 140]:

Theorem 9 (Preto–Tremaine–Mikkola–Tanikawa). *For a separable Hamiltonian function $H = T(\boldsymbol{p}) + V(\boldsymbol{q})$, let the time step function be*

$$g(\boldsymbol{q}, \bar{\boldsymbol{p}}) = \frac{f\big(T_B(\bar{\boldsymbol{p}})\big) - f\big(-V(\boldsymbol{q})\big)}{T_B(\bar{\boldsymbol{p}}) + V(\boldsymbol{q})},$$

where $f\colon \mathbb{R} \to \mathbb{R}$ is a smooth function, and $T_B(\bar{\boldsymbol{p}}) \equiv T(\boldsymbol{p}) + p_0$. If $H = K$ and $f = \ln$, then the Störmer–Verlet algorithm (4.4) *for the extended Hamiltonian $\bar{H} = g(H + p_0) \sim \ln\big(\|\boldsymbol{p}\|^2 + 2p_0\big) + \ln\|\boldsymbol{q}\|$ conserves the diagonal $\mathrm{SO}(4, \mathbb{R})$ momentum map* (3.4) *infinitesimally.*

We shall now assume that $g = g(\boldsymbol{q}, t)$, which signifies that it is a function of the generalized coordinates of the extended phase space, as time is considered a coordinate. Such a time step function amounts to an extended Hamiltonian $\bar{H} = g(\boldsymbol{q}, t)(K + p_0) - g(\boldsymbol{q}, t)R \equiv \bar{K} - \bar{R}$, as is clear from equation (1.24). Since $\bar{R} = \bar{R}(\boldsymbol{q}, t)$, any time step function can easily be integrated numerically, as it only contributes to the momentum updates. It is therefore essential to find an appropriate time step function for $\bar{K}$.

In order to return to the physical-time phase space, we can use the inverse time transformation g^{-1}. However, since $K_B \equiv K + p_0$ is constant over one time step, say $\bar{K} = \epsilon$, we find that

$$\begin{aligned} K'_B &= g^{-1}(\bar{K} - \epsilon) \\ &= K_B - \epsilon/g \\ &= K + p_0 - \epsilon/g. \end{aligned} \tag{4.5}$$

Specifically, for the time step function [118]

$$g(\boldsymbol{q}) = \frac{\|\boldsymbol{q}\|^2}{C_0 + C_1\|\boldsymbol{q}\| + C_2\|\boldsymbol{q}\|^2}, \tag{4.6}$$

where $C_i \in \mathbb{R}$ for $i = 0, 1, 2$, we find that

$$K'_B = \frac{1}{2}\|\boldsymbol{p}\|^2 - \frac{k - C_1\epsilon}{\|\boldsymbol{q}\|} - \frac{C_2\epsilon}{\|\boldsymbol{q}\|^2} - C_0\epsilon + p_0. \tag{4.7}$$

The constant terms can be dropped without altering the dynamics. All in all, we are left with an MICZ Hamiltonian, where $k \mapsto k - C_1\epsilon$ and $\lambda^2 \mapsto -2C_2\epsilon$, which is integrable, so that one may use the generalized leapfrog algorithm to compute any momentum-independent perturbations to the exact solution efficiently. The values of the parameters C_0, C_1 and C_2 can be tuned for optimal performance, that is to say computational time and/or accuracy.

4.1.3 Physical Geodesy

Because the Earth is not spherically symmetric, low-altitude spacecraft in orbit around the Earth experience an asymmetric force that causes both secular and periodic changes in the orbital elements. The main contribution is the so-called J_2 term, which is due to the oblateness of the Earth and of the order 10^{-3}. By averaging over one full orbit we can eliminate periodic effects, and study the secular contributions, which for the J_2 term cause apsidal rotation, that is a rotation of the orbit itself in the orbital plane, a steady change in the mean anomaly, and nodal regression. The latter can be understood intuitively: the additional attraction of the equatorial bulge due to the flattening of the Earth at the poles pulls the spacecraft towards the equatorial plane, thereby inducing a net torque about the line of nodes, which tends to draw the orbital plane towards the equator. The change in angular momentum is parallel to the torque, so that the precession is parallel to the line of nodes and perpendicular to the

angular momentum vector. The line of nodes rotates opposite to the motion of the craft, hence the name nodal regression.

The averaged potential of the J_2 perturbation is

$$\bar{R} = -\frac{kJ_2}{8a}\left(\frac{r_\oplus}{a}\right)^2 \frac{3\cos 2i + 1}{(1-e^2)^{3/2}}, \tag{4.8}$$

where $r_\oplus$ is the mean equatorial radius of the Earth, $k = GM_\oplus$ the standard gravitational parameter with $M_\oplus$ the mass of the Earth, and where the orbital elements are all Keplerian. For more details we refer to Kaula [85], p. 39 or Vallado [157], p. 604. The rates of change for the secular variations of the orbital elements can be calculated from Lagrange's planetary equations

$$\dot{c}_k = -\sum_{n=1}^{6} \{c_k, c_n\} \frac{\partial R}{\partial c_n},$$

where k runs from 1 to 6, $\{c_k\}$ is an appropriate set of orbital parameters, and R is an arbitrary perturbation potential. For $R = \bar{R}$ we find that the secular rates of change for ω, Ω and $\mathscr{M} = \mathscr{M}_0 + nt$ are [157, pp. 606–610]

$$\dot{\omega} = \frac{3nJ_2r_\oplus^2}{8p^2}\left(5\cos 2i + 3\right), \tag{4.9a}$$

$$\dot{\Omega} = -\frac{3nJ_2r_\oplus^2}{2p^2}\cos i, \tag{4.9b}$$

$$\dot{M}_0 = \frac{3nJ_2r_\oplus^2\sqrt{1-e^2}}{8p^2}\left(3\cos 2i + 1\right), \tag{4.9c}$$

respectively. Here we set $n^2a^3 = k$ and $p = a(1-e^2)$.

We shall now show that a modified quasi-Keplerian potential can be used to model the averaged J_2 effects. First, we have seen that the $1/\|\boldsymbol{q}\|^2$ term in the potential of the MICZ problem gives rise to a precession of the apsides in the orbital plane, which has a rate of $\dot{\omega}$. Second, a variation in the effective mass δk induces a change in the mean motion n and thus mean anomaly $\mathscr{M}$:

$$\begin{aligned}\mathscr{M} + \delta\mathscr{M} &= \sqrt{\frac{k+\delta k}{a^3}}t \\ &= nt\left(1 + \frac{\delta k}{2k}\right) + \mathcal{O}\left((\delta k)^2\right).\end{aligned}$$

Therefore, $2k\dot{\mathscr{M}}_0 = n\delta k$. Third, a rotation around the axis from the South Pole to the North Pole, the polar axis, corresponds to a rotation of the line of nodes.

The component of the angular momentum along the axis of rotation L_3 remains constant. Moreover, it lies in the involution of any Hamiltonian with a central force potential $H(\boldsymbol{q},\boldsymbol{p}) = \frac{1}{2}\|\boldsymbol{p}\|^2 + V(\|\boldsymbol{q}\|)$, that is for which $\{H, L_3\} = 0$, so that it is possible to include a contribution proportional to the polar angular momentum, which accounts for the rotation of the line of nodes.

All in all, the perturbation can be modelled by a potential of the form

$$V = -\frac{k+\delta k}{r} + \frac{\lambda^2}{2r^2} + \alpha L_3, \tag{4.10}$$

which upon inspection of equations (4.9) shows that

$$\begin{aligned}
\delta k &= 2k\dot{\mathscr{M}}_0/n \\
&= \frac{3kJ_2r_\oplus^2}{4a^2(1-e^2)^{3/2}}\left(3\cos 2i + 1\right), &(4.11a)\\
\lambda^2 &= 2k\dot{\omega}p/n \\
&= \frac{3kJ_2r_\oplus^2}{4a(1-e^2)}\left(5\cos 2i + 3\right), &(4.11b)\\
\alpha &= \dot{\Omega} \\
&= \frac{3\sqrt{k}J_2r_\oplus^2}{2a^{7/2}(1-e^2)^2}\cos i &(4.11c)
\end{aligned}$$

For any given set of initial conditions these are merely numerical values that characterize the orbit. We emphasize, again, that these are Keplerian orbital elements, though we do not append any subscripts K to minimize the number of subscripts. This demonstrates that the two-body problem with an averaged perturbation due to the oblateness of the Earth is equivalent to the MICZ problem combined with an infinitesimal rotation around the polar axis. The fact that the constructed Hamiltonian is separable enables us to compute the secular motion of the J_2 problem efficiently: first, we use the exact solution of the quasi-Keplerian motion by calculating the quasi-Keplerian motion, and subsequently we rotate the solution around the z-axis. Because the polar angular momentum lies in the involution of H, the order of these operations is irrelevant.

4.2 Numerics of Orbital Dynamics

The extrapolation method by Gragg [60], and Bulirsch and Stoer [25] is for a large class of ordinary differential equations among the most efficient high-precision integrators available. In astrodynamics and celestial mechanics we

are mainly concerned with generic perturbations to N-body systems, such as for instance velocity-dependent forces. In addition to the computational difficulties associated with these perturbations, singular forces appear due to the gravitational attraction of the bodies, which vary with the inverse square of the distance between these bodies. Accurate and efficient integration of the equations of motion is important, especially during close encounters, where the relative energies of the bodies are high, or in the design of trajectories for spacecraft and artificial satellites, where margins for errors are generally small. Algorithmic regularization, as expressed in Theorem 9, eliminates the computational complications due to the singularity in the forces. However, it relies on the Störmer–Verlet algorithm (2.19), which is intended for conservative dynamical systems only. The implicit and generalized midpoint methods have been shown to be successful, though computationally expensive for strongly perturbed problems [67, 120, 121]; for velocity-dependent Rayleigh forces virtually all variational integrators are implicit.

We propose a time-reversible algorithmic regularization scheme that remains explicit even for velocity-dependent forces. The method suggested here is to be part of an extrapolation method, such as the Gragg–Bulirsch–Stoer (GBS) method, in order to attain high accuracy. In Section 4.2.1 we describe the basic algorithm, which we have dubbed the auxiliary-velocity algorithm (AVA). The algorithmic regularization procedure is then adapted to the AVA in Section 4.2.2, where we consider it in the context of the two-body problem. Section 4.2.3 discloses the extension of the algorithmically regularized auxiliary-velocity algorithm (ARAVA) to the N-body problem. Here we also discuss its performance in several illustrative cases from astrodynamics and celestial mechanics: atmospheric friction for low-altitude artificial satellites, the two-dimensional atmospheric re-entry problem, and black-hole binaries. In the latter case we modify the basic ARAVA slightly to incorporate the spin of the primary black hole. We conclude by summarizing the main results, and comparing our results to previous studies on the subject.

4.2.1 Auxiliary-Velocity Algorithm

Inspired by Newton's second law of motion we consider the autonomous initial-value problem

$$\begin{cases} \ddot{\boldsymbol{x}} &= \boldsymbol{f}\left(\boldsymbol{x}, \dot{\boldsymbol{x}}\right), \\ \boldsymbol{x}(0) &= \boldsymbol{x}_0, \\ \dot{\boldsymbol{x}}(0) &= \dot{\boldsymbol{x}}_0, \end{cases} \tag{4.12}$$

where the single (double) overdot indicates the first (second) derivative with respect to time t, $\boldsymbol{x} \in \mathbb{R}^n$, $\dot{\boldsymbol{x}} \in \mathbb{R}^n$, and $\boldsymbol{f} \colon \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^n$ is assumed to be sufficiently smooth, specifically Lipschitz continuous. The second-order ordinary differential equation describing the time evolution of the initial-value problem is explicit, which allows us to write it as a coupled system of first-order ordinary differential equations by defining $\dot{\boldsymbol{x}} = \boldsymbol{v}$:

$$\begin{cases} \dot{\boldsymbol{x}} &= \boldsymbol{v}, \\ \dot{\boldsymbol{v}} &= \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{v}), \end{cases} \tag{4.13}$$

with appropriate initial conditions for $\boldsymbol{x}(0)$ and $\boldsymbol{v}(0)$. By writing $\boldsymbol{z} = (\boldsymbol{x}, \boldsymbol{v})$ and $\boldsymbol{g}(\boldsymbol{z}) = (\boldsymbol{v}, \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{v}))$ we arrive at the equivalent first-order system $\dot{\boldsymbol{z}} = \boldsymbol{g}(\boldsymbol{z})$.

The leapfrog is a standard choice for the numerical integration of initial-value problems, either on its own or within the Gragg–Bulirsch–Stoer (GBS) extrapolation algorithm for second-order ordinary differential equations, as it is explicit, time-reversible, and of second order in the step size h: the asymptotic expansion of its global discretization error contains only terms with even powers of the step size. Since $\boldsymbol{v}$ appears in the vector field, it is however not possible to construct such an explicit leapfrog algorithm for the first-order system (4.13) or its second-order equivalent (4.12). Instead, the implicit midpoint method, which we shall describe below from equation (4.28) onward in more detail, has been suggested [119]:

$$\boldsymbol{x}_{k+\frac{1}{2}} = \boldsymbol{x}_k + \frac{h}{2}\boldsymbol{v}_k, \tag{4.14a}$$

$$\boldsymbol{v}_{k+1} = \boldsymbol{v}_k + h\boldsymbol{f}\left(\boldsymbol{x}_{k+\frac{1}{2}}, \tfrac{1}{2}(\boldsymbol{v}_k + \boldsymbol{v}_{k+1})\right), \tag{4.14b}$$

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_{k+\frac{1}{2}} + \frac{h}{2}\boldsymbol{v}_{k+1}. \tag{4.14c}$$

For a generic non-linear vector field the value $\boldsymbol{v}_{k+1}$ has to be solved numerically with iterative procedures. For highly non-linear and/or computationally expensive vector fields the updates in the velocities are obviously the bottleneck in the performance of the numerical integration of the initial-value problem.

At the core of the auxiliary-velocity algorithm lies the extended space of variables consisting of the position $\boldsymbol{x}$, the velocity $\boldsymbol{v}$, and the auxiliary velocity

$\boldsymbol{w}$. The integration with the AVA from time t_k to time t_{k+1} proceeds as follows:

$$\boldsymbol{x}_{k+\frac{1}{2}} = \boldsymbol{x}_k + \frac{h}{2}\boldsymbol{v}_k, \tag{4.15a}$$

$$\boldsymbol{w}_{k+\frac{1}{2}} = \boldsymbol{w}_k + \frac{h}{2}\boldsymbol{f}\left(\boldsymbol{x}_{k+\frac{1}{2}}, \boldsymbol{v}_k\right), \tag{4.15b}$$

$$\boldsymbol{v}_{k+1} = \boldsymbol{v}_k + h\boldsymbol{f}\left(\boldsymbol{x}_{k+\frac{1}{2}}, \boldsymbol{w}_{k+\frac{1}{2}}\right), \tag{4.15c}$$

$$\boldsymbol{w}_{k+1} = \boldsymbol{w}_{k+\frac{1}{2}} + \frac{h}{2}\boldsymbol{f}\left(\boldsymbol{x}_{k+\frac{1}{2}}, \boldsymbol{v}_{k+1}\right), \tag{4.15d}$$

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_{k+\frac{1}{2}} + \frac{h}{2}\boldsymbol{v}_{k+1}, \tag{4.15e}$$

where initially we set $\boldsymbol{v}_0 = \boldsymbol{w}_0$. In the extended space of variables the algorithm is explicit and time-reversible, or symmetric. Notice that the (implicit) velocity update for the implicit midpoint (4.14b) has been replaced by a construction resembling the basic leapfrog in the auxiliary-velocity algorithm (4.15b)–(4.15d); the intermediate auxiliary velocity agrees to first order in the step size with the mean velocity over one step.

Let Φ_h denote the one-step map of a generic symmetric integrator of order p, that is $\boldsymbol{z}_{k+1} = \boldsymbol{z}_k + \Phi_h(\boldsymbol{z}_k)$. In the case of the AVA we have that $\boldsymbol{z} = (\boldsymbol{x}, \boldsymbol{v}, \boldsymbol{w})$. We can of course increase the accuracy of any symmetric integrator by at least one order by taking the composition of s iterations of the basic integrator with modified step sizes as in equation (2.11). The resulting higher-order integrator is a symmetric integrator too. Therefore, we can increase the order of the AVA by taking several copies of the AVA with different step sizes. It is important to note that for odd p there are no real solutions to the equations for the coefficients, so that we have to take either p even, or consider complex time steps.

As an example, the coefficients for the symmetric fourth-order integrator obtained from concatenation of three steps of a second-order symmetric algorithm are $\gamma_1 = \frac{1}{6}(4 + 2\sqrt[3]{2} + \sqrt[3]{4})$, $\gamma_2 = -\frac{1}{3}(1 + \sqrt[3]{2})^2$, and $\gamma_3 = \gamma_1$, as found by Yoshida [165]. A list of values for the coefficients of a symmetric sixth-order

algorithm, which we shall use below, is given by

$$\begin{aligned}
\gamma_1 &= 0.78451361047755726381949763,\\
\gamma_2 &= 0.23557321335935813368479318,\\
\gamma_3 &= -1.17767998417887100694641568,\\
\gamma_4 &= 1.31518632068391121888424973,\\
\gamma_5 &= -1.17767998417887100694641568,\\
\gamma_6 &= 0.23557321335935813368479318,\\
\gamma_7 &= 0.78451361047755726381949763,
\end{aligned}$$

Clearly the cost of the increase of the order is the number of stages: a fourth-order composition algorithm requires at least three stages, a sixth-order integrator at least seven stages, and so on. The 'optimal' values of the γs have been calculated for different problems under different optimality criteria, such as for example accuracy, the number of force evaluations, and CPU time. These and related considerations are collected in Hairer et al. [66], Chap. 5.

The Gragg–Bulirsch–Stoer (GBS) extrapolation method [25, 60] is often used for high-precision numerical integration of initial-value problems. In a nutshell, the GBS extrapolation method consists of a numerical integrator, and an interpolation as well as an extrapolation procedure. The numerical integrator computes a finite sequence of approximations to the initial-value problem at different step sizes: $\boldsymbol{z}(h_i)$. An interpolation algorithm is applied to the sequence of approximations, such that interpolating function $\hat{\boldsymbol{z}}(h_i)$ is equal to $\boldsymbol{z}(h_i)$. By means of an extrapolation procedure this interpolating function is calculated for the limiting case of zero step size, that is to say the exact solution $\boldsymbol{z}(0) = \lim_{h\to 0} \boldsymbol{z}(h)$ is approximated by $\hat{\boldsymbol{z}}(0)$.

The standard implementation of the GBS method requires the discretization method to have an asymptotic expansion of the global error that proceeds with even powers of the step size, as for each additional Richardson extrapolation the accuracy is increased by two orders in the step size instead of merely one. A sufficient condition is that the finite-difference scheme is time-reversible. Since both the implicit midpoint method and the leapfrog, including the auxiliary-velocity algorithm, are second-order symmetric integrators, they may be used inside the GBS extrapolation method[1]. The GBS extrapolation method requires

[1]We call attention to the fact that in general the extrapolation procedure does not preserve the differential-geometric structure. As a matter of fact, if a numerical algorithm is both symmetric and symplectic, and of order $2p$, $p \geq 1$, then polynomial extrapolation methods can be constructed that conserve the symplectic form up to order $4p + 1$, as shown by Blanes et al. [19].

smooth vector fields, which can be achieved by the Sundman transformation considered in the next section.

Before we continue with the algorithmic regularization procedure, we note that for conservative dynamical systems variational integrators are to be preferred in long-term simulations, as they preserve both the symplectic structure and any momentum maps associated with symmetry actions. We have shown in Chapter 2 that non-conservative forces can be added easily in the discrete variational formalism. Nevertheless, these integrators are not as efficient inside the GBS extrapolation algorithm as the one we propose, since they are generally implicit. We emphasize that for conservative dynamical systems the AVA reduces to the basic Störmer–Verlet algorithm, which is indeed symplectic. Even so, the AVA is principally designed to be an efficient alternative to existing integrators for non-conservative dynamical systems, and it is ideally suited inside the GBS extrapolation algorithm.

4.2.2 Algorithmic Regularization

Here we apply the algorithmic regularization procedure for the two-body problem, as described by Preto and Tremaine [140], and Mikkola and Tanikawa [122, 123], to the auxiliary-velocity algorithm. The reduced equations of motion (per unit of mass) for the perturbed two-body problem read

$$\ddot{\boldsymbol{r}} = -\frac{k\boldsymbol{r}}{r^3} + \epsilon \boldsymbol{a}\left(t, \boldsymbol{r}, \boldsymbol{v}\right), \tag{4.16}$$

where k is the gravitational parameter, $r = \|\boldsymbol{r}\|$, the relative position of the two bodies, $\boldsymbol{v}$ is the relative velocity, ϵ is the perturbation parameter, and $\boldsymbol{a}$ is a general perturbing acceleration. For the sake of generality we allow for the possibility that the acceleration depends on time. The advantage of the algorithmic regularization procedure is that it produces the exact trajectory with an $\mathcal{O}(h^2)$ error in the time for $\epsilon = 0$; close approaches are regularized, and thus the algorithm is not merely an adaptive step size algorithm [65, 72], or an algorithm with variable time steps and regularization built-in [94].

Before proceeding we introduce some definitions to facilitate the algorithmic regularization of the auxiliary-velocity algorithm. Define the kinetic energy $T = T(\boldsymbol{v}) = \frac{1}{2}v^2$, the negative value of the two-body potential energy $U = U(\boldsymbol{r}) = k/r$, which is commonly known as the force function, the Keplerian binding energy $B = U - T$, and the Newtonian two-body acceleration $\boldsymbol{A}(\boldsymbol{r}) = \nabla U(\boldsymbol{r})$.

The algorithmic regularization is accomplished by the time transformation

$$\mathrm{d}s = U(\boldsymbol{r})\mathrm{d}t, \tag{4.17}$$

which yields the derivatives with respect to the reparameterized time s, denoted by the prime,

$$\boldsymbol{r}' = \boldsymbol{v}/T_B, \tag{4.18a}$$
$$\boldsymbol{v}' = \left[\boldsymbol{A}(\boldsymbol{r}) + \epsilon\boldsymbol{a}\left(t, \boldsymbol{r}, \boldsymbol{v}\right)\right]/U, \tag{4.18b}$$
$$\boldsymbol{w}' = \left[\boldsymbol{A}(\boldsymbol{r}) + \epsilon\boldsymbol{a}\left(t, \boldsymbol{r}, \boldsymbol{w}\right)\right]/U. \tag{4.18c}$$

Here $T_B \equiv T + B = U$ along the orbit. The distinction between T_B and U in the transformed equations of motion is to ensure separability of the unperturbed problem, that is for $\epsilon = 0$, as in that case the right-hand side of equation (4.18a) can only depend on $\boldsymbol{v}$, and the right-hand sides of equations (4.18b) and (4.18c) can only depend $\boldsymbol{r}$. The equation for the auxiliary velocity (4.18c) is clearly identical to the equation for the ordinary velocity (4.18b) but with $\boldsymbol{v}$ replaced by $\boldsymbol{w}$ everywhere. The transformation affects the binding energy as well for it depends implicitly on time through both $\boldsymbol{r}$ and $\boldsymbol{v}$. Hence, B transforms accordingly,

$$B' = -\epsilon\boldsymbol{v}\cdot\boldsymbol{a}\left(t, \boldsymbol{r}, \boldsymbol{v}\right)/U.$$

Finally, we mention the scaling of the time step h under the transformation (4.17): $\bar{h} = h/T_B = h/U$ with $\bar{h}$ the 'time-transformed' step size used in the regularized numerical computations.

On the whole, we can write the update maps concisely as $\boldsymbol{X}(h)$ and $\boldsymbol{V}(h)$, where

$$\boldsymbol{X}(h)\colon \boldsymbol{r}_k \mapsto \boldsymbol{r}_{k+1},$$
$$\boldsymbol{V}(h)\colon \boldsymbol{v}_k \mapsto \boldsymbol{v}_{k+1}.$$

In particular, we have for the auxiliary-velocity algorithm that the map $\boldsymbol{X}(h)$ is to be implemented as

$$\boldsymbol{X}(h)\colon \left[\begin{array}{rcl} T_B & = & \frac{1}{2}v^2 + B, \\ \bar{h} & = & h/T_B, \\ t & \mapsto & t + \bar{h}, \\ \boldsymbol{r} & \mapsto & \boldsymbol{r} + \bar{h}\boldsymbol{v}, \end{array}\right. \tag{4.19a}$$

and $\boldsymbol{V}(h)$ as

$$\boldsymbol{V}(h): \left[\begin{array}{rcl} U & = & k/r, \\ \bar{h} & = & h/U, \\ \boldsymbol{w} & \mapsto & \boldsymbol{w} + \frac{1}{2}\bar{h}\left[\boldsymbol{A}(\boldsymbol{r}) + \epsilon \boldsymbol{a}\left(t, \boldsymbol{r}, \boldsymbol{v}\right)\right], \\ \boldsymbol{v} & \mapsto & \boldsymbol{v} + \bar{h}\left[\boldsymbol{A}(\boldsymbol{r}) + \epsilon \boldsymbol{a}\left(t, \boldsymbol{r}, \boldsymbol{w}\right)\right], \\ B & \mapsto & B - \epsilon \bar{h} \boldsymbol{w} \cdot \boldsymbol{a}\left(t, \boldsymbol{r}, \boldsymbol{w}\right), \\ \boldsymbol{w} & \mapsto & \boldsymbol{w} + \frac{1}{2}\bar{h}\left[\boldsymbol{A}(\boldsymbol{r}) + \epsilon \boldsymbol{a}\left(t, \boldsymbol{r}, \boldsymbol{v}\right)\right]. \end{array}\right. \tag{4.19b}$$

Notice that the unperturbed Keplerian acceleration $\boldsymbol{A}(\boldsymbol{r})$ is evaluated only once in each iteration, whereas the perturbation is computed three times. Furthermore, if the perturbation separates into a velocity-dependent and a velocity-independent part, then, again, the part that does not depend on the velocity is to be evaluated only once.

Let $\boldsymbol{Y}(h)$ be the composition $\boldsymbol{X}(h/2) \circ \boldsymbol{V}(h) \circ \boldsymbol{X}(h/2)$, that is one full iteration of the regularized auxiliary-velocity algorithm with step size h. The numerical integration of the perturbed two-body problem (4.16) from time t_k to $t_{k+m} = t_k + mh$ with $m \in \mathbb{N}$ is simply a sequence of m subsequent leapfrog iterations, that is

$$\begin{aligned} [\boldsymbol{Y}(h)]^k &= \boldsymbol{Y}(h) \circ \ldots \circ \boldsymbol{Y}(h) \\ &= \boldsymbol{X}(h/2) \circ [\boldsymbol{V}(h) \circ \boldsymbol{X}(h)]^{k-1} \circ \boldsymbol{V}(h)\boldsymbol{X}(h/2) \\ &= \boldsymbol{X}(h/2)\boldsymbol{V}(h) \circ [\boldsymbol{X}(h) \circ \boldsymbol{V}(h)]^{k-1} \boldsymbol{X}(h/2). \end{aligned}$$

Here we have simplified the expression by using the relation $\boldsymbol{X}(h/2) \circ \boldsymbol{X}(h/2) = \boldsymbol{X}(h)$. Henceforth we shall refer to the algorithmically regularized auxiliary-velocity algorithm, defined by the map (4.19), by its acronym: ARAVA.

4.2.3 Applications

It is obvious that the basic AVA (4.15) is valid for any ordinary differential equation in which the vector field depends on functions of the dependent variable and their derivatives with respect to the independent variable. The regularized two-body version of Section 4.2.2 is designed for the two-body problem, but here we extend that result to the general N-body problem, where $N \geq 2$. These results are obviously valid for both the gravitational N-body problem and its counterpart in classical molecular dynamics. Furthermore, we discuss the application of the ARAVA to several practical problems, such as

the motion of artificial satellites in low Earth orbits, the atmospheric re-entry problem, and the dynamics of satellites near black holes.

The utility of the ARAVA is of course not limited to these demonstrative cases. Realistic physical systems with perturbations that depend on the derivatives of the dependent variable with respect to the independent variable are abundant. Typically, the dependent variable is the position, and the independent one is time, such as in our examples and for instance the motion of dust particles [see e.g. 137]. For planetary landing vehicles and rockets, the mass is a dependent (time-varying) variable too. Its derivative with respect to time, the mass (flow) rate, multiplied by the relative exhaust velocity gives the thrust. Its inclusion in the ARAVA formalism requires it to be written as part of an augmented velocity vector, as described in the section on the dynamics of black-hole binaries. In that case the ARAVA can be used to compute the desired trajectory numerically, once an (optimal) control law has been determined. For a discussion of the one-dimensional lunar soft-lander problem, where the optimal control law simply consists of switching the thruster on at full throttle until touchdown, upon fulfilment of the control criterion, we refer to Meditch [116] and Kirk [88], pp. 427–428.

N-Body Formulation

The two-body ARAVA translates directly to its more general N-body version. As usual, let $\boldsymbol{r}_{ij} = \boldsymbol{r}_i - \boldsymbol{r}_j$ denote the relative distance between the bodies labelled i and j with $i \neq j$. We define the total kinetic energy as

$$T = \frac{1}{2}\sum_{i=1}^{N} m_i \|\boldsymbol{v}_i\|^2 = \frac{1}{2}\bar{V}^2, \tag{4.20}$$

where we have introduced the shorthand notation $\bar{\boldsymbol{V}} = (\sqrt{m_1}\boldsymbol{v}_1, \ldots, \sqrt{m_N}\boldsymbol{v}_N)$. Similarly, we define $\boldsymbol{R} = (\boldsymbol{r}_1, \ldots, \boldsymbol{r}_N)$ and $\boldsymbol{V} = \dot{\boldsymbol{R}} = (\boldsymbol{v}_1, \ldots, \boldsymbol{v}_N)$. The N-body force function is simply

$$U = \frac{1}{2}\sum_{i \neq j} \frac{G m_i m_j}{r_{ij}}, \tag{4.21}$$

where G is Newton's gravitational constant, and m_i is the mass of the ith body. As before, the N-body binding energy $B = U - T$. The factor of one-half appears in order to avoid double-counting of contributions to the gravitational potential.

Again, the time transformation for the algorithmic regularization is

$$\mathrm{d}s = U\mathrm{d}t, \tag{4.22}$$

where now U depends on all the relative coordinates of the N bodies. The total specific force on each body is the sum of the resultant gravitational acceleration due to the attraction of $(N-1)$ bodies, and a velocity-dependent acceleration:

$$\begin{aligned}\boldsymbol{F}_i &= -\sum_{\substack{j=1\\ i\neq j}}^{N} \frac{Gm_j}{r_{ij}^2}\frac{\boldsymbol{r}_{ij}}{r_{ij}} + \epsilon \boldsymbol{a}\left(t, \boldsymbol{R}, \boldsymbol{v}_i\right) \\ &\equiv \boldsymbol{G}_i(\boldsymbol{R}) + \epsilon \boldsymbol{a}\left(t, \boldsymbol{R}, \boldsymbol{v}_i\right)\end{aligned} \tag{4.23}$$

for all bodies $i = 1, \ldots, N$. Furthermore, we define $\boldsymbol{G}(\boldsymbol{R}) = (\boldsymbol{G}_1, \ldots, \boldsymbol{G}_N)$ and the specific N-body perturbing force as

$$\boldsymbol{P}\left(t, \boldsymbol{R}, \boldsymbol{V}\right) = \left(\boldsymbol{a}\left(t, \boldsymbol{R}, \boldsymbol{v}_1\right), \ldots, \boldsymbol{a}\left(t, \boldsymbol{R}, \boldsymbol{v}_N\right)\right)$$

The coordinate and velocity update maps for the ARAVA then become

$$\boldsymbol{X}(h)\colon \left[\begin{array}{rcl} T_B &=& \frac{1}{2}\bar{V}^2 + B, \\ \bar{h} &=& h/T_B, \\ t &\mapsto& t + \bar{h}, \\ \boldsymbol{R} &\mapsto& \boldsymbol{R} + \bar{h}\boldsymbol{V}, \end{array}\right. \tag{4.24a}$$

and

$$\boldsymbol{V}(h)\colon \left[\begin{array}{rcl} U &=& \dfrac{1}{2}\displaystyle\sum_{i\neq j}\frac{Gm_im_j}{r_{ij}}, \\ \bar{h} &=& h/U, \\ \boldsymbol{W} &\mapsto& \boldsymbol{W} + \frac{1}{2}\bar{h}\left[\boldsymbol{G}(\boldsymbol{R}) + \epsilon\boldsymbol{P}\left(t, \boldsymbol{R}, \boldsymbol{V}\right)\right], \\ \boldsymbol{V} &\mapsto& \boldsymbol{V} + \bar{h}\left[\boldsymbol{G}(\boldsymbol{R}) + \epsilon\boldsymbol{P}\left(t, \boldsymbol{R}, \boldsymbol{W}\right)\right], \\ B &\mapsto& B - \epsilon\bar{h}\boldsymbol{W}\cdot\boldsymbol{P}\left(t, \boldsymbol{R}, \boldsymbol{W}\right), \\ \boldsymbol{W} &\mapsto& \boldsymbol{W} + \frac{1}{2}\bar{h}\left[\boldsymbol{G}(\boldsymbol{R}) + \epsilon\boldsymbol{P}\left(t, \boldsymbol{R}, \boldsymbol{V}\right)\right], \end{array}\right. \tag{4.24b}$$

respectively. Again, at $t = 0$ we set $\boldsymbol{V}(0) = \boldsymbol{W}(0)$, which clearly generalizes the basic two-body formulation from Section 4.2.2. We note that for $\epsilon = 0$ the N-body ARAVA and the integrator described by Mikkola and Tanikawa [122] coincide.

We have ascertained the correctness of our implementation of the ARAVA for an arbitrary number of bodies through randomized trial runs. For all numerical

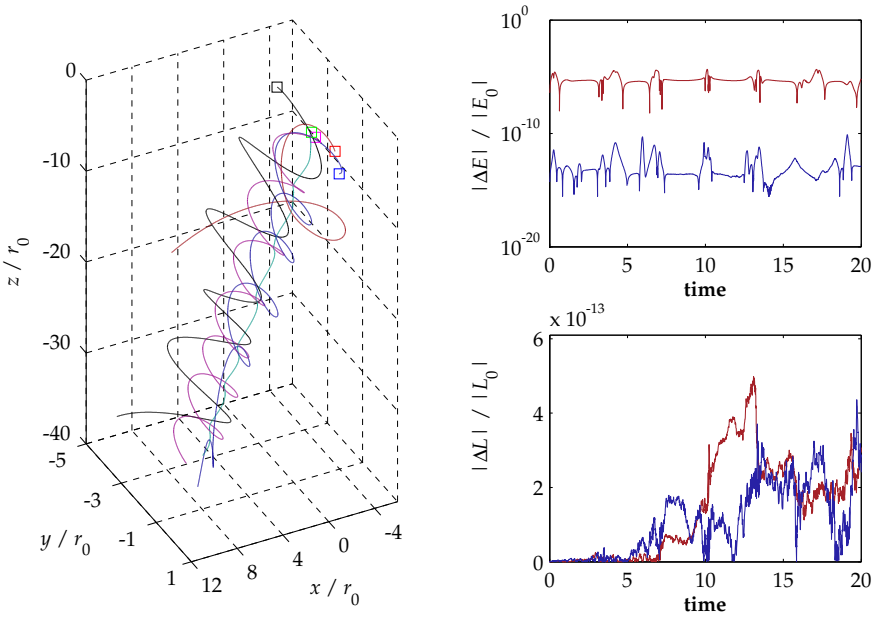


FIGURE 4.1: On the left, a visualization of the numerical computation with ARAVA of the trajectories of the five bodies from their initial positions, indicated by the squares. On the right, the performance of the ARAVA in the five-body problem: the relative errors in the total energy (top) and total angular momentum (bottom) as a function of the time for the standard second-order algorithm (blue) and the sixth-order symmetric composition method (red).

computations we have set $G = 1$ for convenience. In each run we have chosen the N masses at random from the unit interval, which physically means that all masses are defined in proportion to the largest mass. The N bodies have then been positioned at random within a sphere of an arbitrary radius r_0 with random initial velocities. The results of such a run for $N = 5$ with close encounters are shown in Figure 4.1 for an integration step size of $h = 0.1$. In the left panel of Figure 4.1 the numerically calculated trajectories of the five bodies from their initial positions are drawn; the right panels show the accuracy of the second- (blue) and sixth-order (red) ARAVA in terms of the relative errors in the total energy and total angular momentum.

Artificial Satellites in Low Earth Orbits

For our next numerical simulation we have injected a medium-sized satellite with a mass $m = 500\,\mathrm{kg}$, drag coefficient $C_D = 2.2$, and cross-sectional area $A = 2.5\,\mathrm{m}^2$ into a nearly circular equatorial orbit with a 7024 km semi-major axis, which corresponds to an altitude of approximately 650 km. The eccentricity of the orbit is $e = 0.04$ and the inclination is $i = 1^\circ$. At the surface of the Earth we take the standard atmospheric density $\rho_0 = 1.3\,\mathrm{kg\,m}^{-3}$, and at an altitude of 650 km we choose a nominal density of $10^{-13}\,\mathrm{kg\,m}^{-3}$, which yields a barometric coefficient $\beta = 4.7 \cdot 10^{-2}\,\mathrm{km}^{-1}$.

Since the drag force is a non-conservative (dissipative) force, the work exerted by the drag force along the orbit of a the satellite is a strictly monotonic increasing function of time, which causes the total energy of the satellite to decrease as time passes. The energy decrease is responsible for the reduction of the size and eccentricity of the orbit, which leads to a downward spiralling motion of the satellite towards the Earth. Therefore, the change in the total energy of the satellite is a good measure of the effect of the perturbation on the orbit and the numerical accuracy, especially since the ARAVA preserves the total two-body energy up to round-off errors in the absence of any perturbations.

We have compared the AVA and ARAVA to an Adams–Bashforth–Moulton (ABM) algorithm with an absolute tolerance of 10^{-10} and a relative tolerance of 10^{-8}, which serves as a benchmark. Numerous numerical experiments with different numbers of harmonics in the geopotential (up to 36) indicate that both the AVA and ARAVA demonstrate the correct behaviour, that is the energy decreases steadily with approximately the right amount. There is however some artificial dissipation due to the fact that the inclusion of non-conservative forces in the AVA and ARAVA does not respect the differential-geometric structure of the dynamical system, in other words the algorithms are not variational anymore for non-conservative forces. In Figure 4.2 we have drawn the relative energy errors for the AVA and ARAVA ($h = 1\,\mathrm{min}$), and the classical Runge–Kutta method ($h = 2\,\mathrm{min}$) for a 12×12 geopotential. It is clear that the standard Runge–Kutta method suffers from the same problems with regards to the energy behaviour, as already noted by Marsden and West [108] for a different dynamical system. For computations over longer periods of time a variational integrator might actually be preferred, although that generally comes at the price of being implicit.

Important to note is that a quarter of all vector field evaluations are due to the coordinate-dependent acceleration (geopotential) for the ARAVA, which

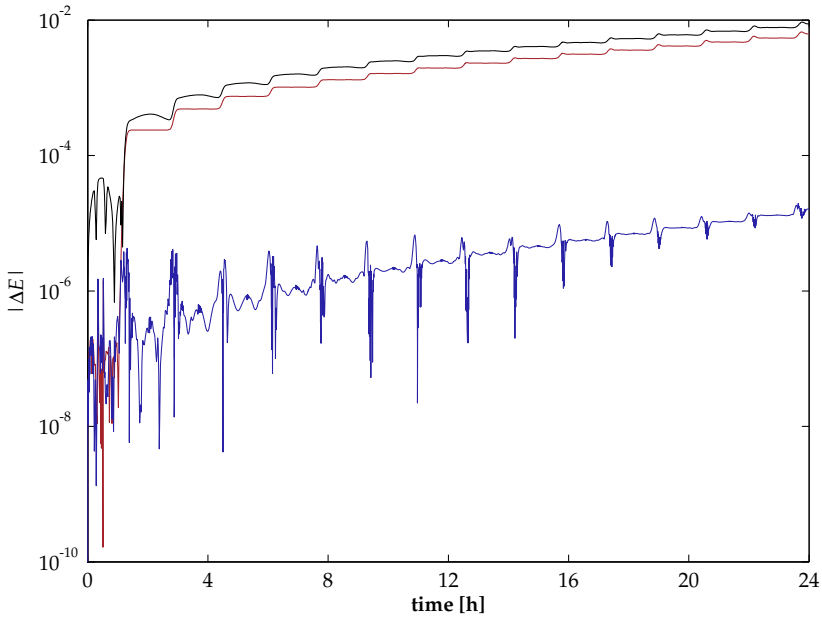


FIGURE 4.2: Relative energy errors for the second-order ARAVA (red) and AVA (blue) with $h = 60\,\mathrm{s}$ and the fourth-order classical Runge–Kutta method (black) with $h = 120\,\mathrm{s}$.

uses only one such an evaluation per time step. Higher order terms in the geopotential thus affect only one-quarter of all the vector field evaluations for the ARAVA, whereas they affect all vector field evaluations of most standard integrators, such as those of the Runge–Kutta family and multi-step methods. Hence, the inclusion of higher order terms in the geopotential increases the overhead significantly more for algorithms different from the ARAVA. As an example, the ARAVA is already faster than the standard Runge–Kutta algorithm for a relatively modest 2×2 geopotential at the same accuracy. Interestingly, for a generic geopotential the ARAVA outperforms the AVA with exactly the same step size on average by 5–10% measured in CPU time.

Spacecraft Re-Entry

We shall now consider the two-dimensional re-entry problem [42, Chap. 2], that is we neglect any cross-track deviations. Furthermore, we assume the atmosphere to be stationary during the re-entry phase, which means that the rotation of the Earth and any winds are ignored: $\boldsymbol{v} = \boldsymbol{u}$. Recently, Mititelu [125] obtained analytical solutions for the planar atmospheric re-entry problem under similar simplifying assumptions. Though the solution obtained can be valuable in the actual design and quick re-evaluation of the relevant flight parameters, the solution is highly non-trivial yet limited in its scope. Instead of focussing on the theoretical framework, we concentrate on the performance of the numerical integrator, which can easily be adapted to more realistic scenarios, such as the three-dimensional atmospheric entry problem. A brief description of the three-dimensional re-entry problem with attitude control can be found in for instance Stengel [148], pp. 265–270.

For our numerical demonstration we have taken a craft with mass $m = 60000\,\mathrm{kg}$ flying at Mach 25 at an altitude of 120 km with an initial flight-path angle of -7°. Furthermore, we have set $F_L/F_D = 9.5$, and $\beta = 0.12\,\mathrm{km}^{-1}$. We note that the lift-to-drag ratio is in fact rather large, so we can expect several clearly visible skip regions, where the spacecraft rebounds from the atmosphere to decelerate the craft; we might expect numerical difficulties associated with perturbations that exceed the gravitational attraction by several orders of magnitude.

We have used the sixth-order ARAVA with a step size $h = 0.1\,\mathrm{s}$. The altitude during re-entry is shown in Figure 4.3, where it is clear that the lift causes the craft to bounce off the atmosphere several times during the re-entry. The difference between the solution of the ARAVA and a high-fidelity ABM algorithm with an absolute error tolerance of 10^{-14} and a relative error tolerance of 10^{-12} is plotted in Figure 4.4. During the final (approach) phase of the re-entry, the error tends to grow as a result of the accumulation of truncation and round-off errors, which tend to create fluctuations in the perturbation. Since the local density at an altitude of 20 km is roughly $0.1\,\mathrm{kg\,m}^{-3}$, these fluctuations are more easily amplified, whereas they tend to be suppressed during the initial (critical) phase of the re-entry, where the density of the atmosphere is only a fraction of that value. Notice, however, that the overall altitude error still remains well below 1 m for most of the time during the re-entry.

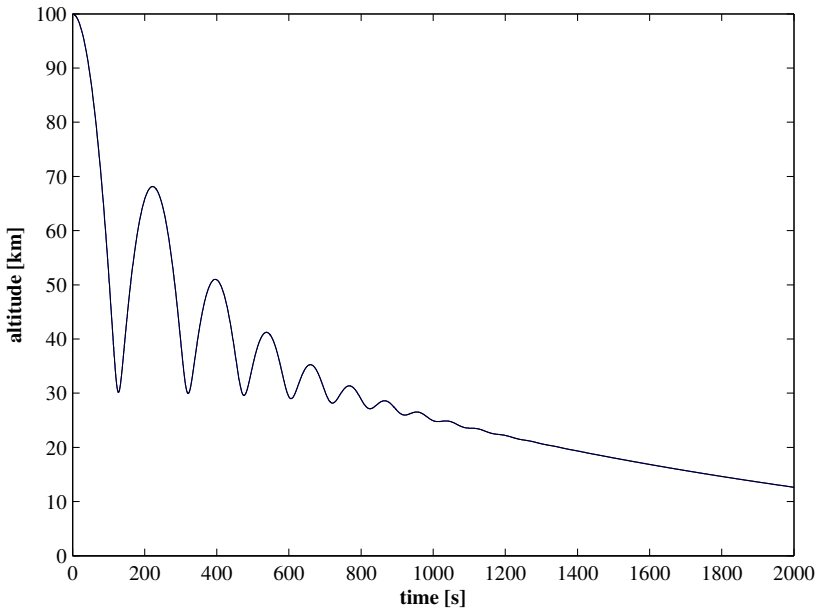


FIGURE 4.3: Altitude versus time for the planar atmospheric re-entry problem: several skip regions are clearly visible, where the lift temporarily dominates the gravitational attraction.

BLACK-HOLE BINARIES

Instead of the standard implementation of the GBS extrapolation method with its modified midpoint integrator, a generalized midpoint based on the leapfrog algorithm, specifically designed for second-order differential equations has previously been suggested for the numerical study of the motion of stars around black holes [120, 121]. Here we discuss the application of the ARAVA in conjunction with the GBS extrapolation scheme with rational polynomial interpolation and Richardson extrapolation, and demonstrate that it is a highly competitive alternative.

Let $\boldsymbol{x}$ be the relative separation vector between the (primary) black hole with mass m_1 and a satellite with mass m_2 in its vicinity, $\boldsymbol{v}$ its derivative with respect to time, and $\boldsymbol{s}$ the spin vector of the primary. The general-relativistic force per

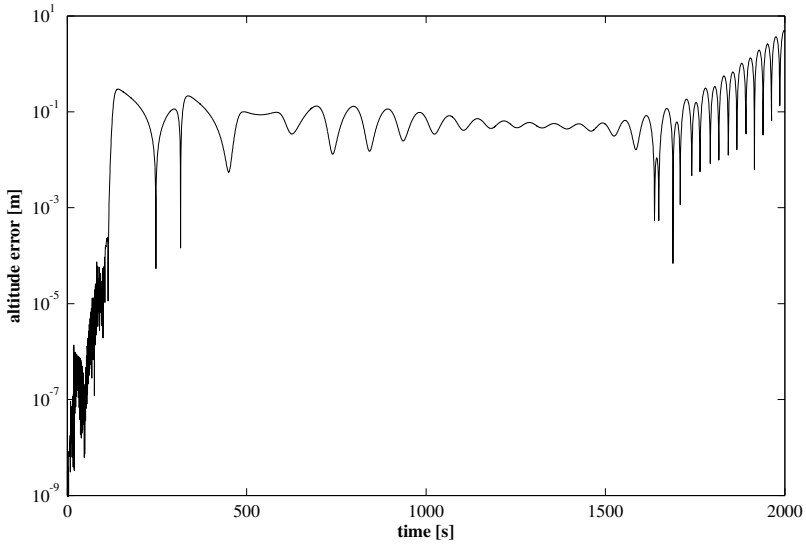


FIGURE 4.4: An estimation of the absolute error in the numerical computation of the altitude with a sixth-order ARAVA for $h = 0.1\,\mathrm{s}$.

unit mass in the parameterized post-Newtonian formalism is given by [87, 146]

$$\boldsymbol{a} = \boldsymbol{a}_0 + c^{-2}\boldsymbol{a}_2 + c^{-3}\boldsymbol{a}_3 + c^{-4}\boldsymbol{a}_4 + c^{-5}\boldsymbol{a}_5, \tag{4.25}$$

where $\boldsymbol{a}_0$ is the Newtonian contribution, and c denotes the speed of light in vacuum. The post-Newtonian acceleration $\boldsymbol{a}_2 = \boldsymbol{a}_2(\boldsymbol{x}, \boldsymbol{v})$ is the main (PPN1) contribution, $\boldsymbol{a}_3 = \boldsymbol{a}_3(\boldsymbol{x}, \boldsymbol{v}, \boldsymbol{s})$ describes the spin–orbit interaction [11], and $\boldsymbol{a}_4 = \boldsymbol{a}_4(\boldsymbol{x}, \boldsymbol{v}, \boldsymbol{s})$ can be decomposed into a (PPN2) term that depends only on $\boldsymbol{x}$ and $\boldsymbol{v}$, and a quadrupole term that depends on all three variables, as it is induced by the rotation of the black hole [164]. Finally, $\boldsymbol{a}_5 = \boldsymbol{a}_5(\boldsymbol{x}, \boldsymbol{v})$ is the radiative contribution, which is generally referred to as the PPN2.5 term. All in all, the equations of motion can be written concisely as

$$\begin{cases} \dot{\boldsymbol{x}} &= \boldsymbol{v}, \\ \dot{\boldsymbol{v}} &= \boldsymbol{a}(\boldsymbol{x}, \boldsymbol{v}, \boldsymbol{s}), \\ \dot{\boldsymbol{s}} &= \boldsymbol{b}(\boldsymbol{x}, \boldsymbol{v}, \boldsymbol{s}). \end{cases} \tag{4.26}$$

The first-order differential equation for the spin, $\dot{\boldsymbol{s}} = \boldsymbol{b}(\boldsymbol{x}, \boldsymbol{v}, \boldsymbol{s})$, accounts for the rigid rotation of the black hole. Herein the spin of the satellite is assumed to be negligible, that is $m_1 \gg m_2$.

In order to be able to use the ARAVA as discretization scheme within the GBS extrapolation method, we have to rewrite the equations of motion for the black-hole binary system (4.26) as

$$\begin{cases} \dot{\boldsymbol{x}} &= \boldsymbol{v}, \\ \dot{\boldsymbol{v}}_{\text{aug}} &= \boldsymbol{a}_{\text{aug}}\left(\boldsymbol{x}, \boldsymbol{v}_{\text{aug}}\right), \end{cases} \tag{4.27}$$

where we have introduced the augmented 'velocity' vector $\boldsymbol{v}_{\text{aug}} = (\boldsymbol{v}, \boldsymbol{s})$, and $\boldsymbol{a}_{\text{aug}}\left(\boldsymbol{x}, \boldsymbol{v}_{\text{aug}}\right) = \left(\boldsymbol{a}\left(\boldsymbol{x}, \boldsymbol{v}_{\text{aug}}\right), \boldsymbol{b}\left(\boldsymbol{x}, \boldsymbol{v}_{\text{aug}}\right)\right)$ the augmented vector field. Associated with the augmented generalized velocity is the augmented auxiliary variable $\boldsymbol{w}_{\text{aug}} = (\boldsymbol{w}, \boldsymbol{\sigma})$, where $\boldsymbol{\sigma}$ is the auxiliary spin. It is clear from equations (4.27) that the coordinate update map for the ARAVA remains unaltered, whereas the velocity update map now includes both the velocity and the spin.

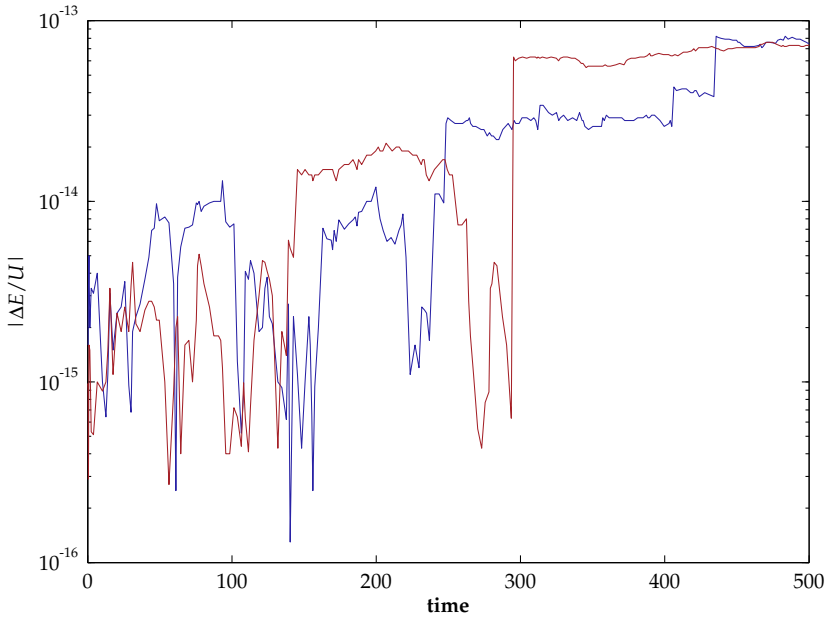


FIGURE 4.5: Accuracy of the GBS extrapolation algorithm with the ARAVA (red) and generalized midpoint method (blue) for the black-hole binary system.

We have compared the performance of the generalized midpoint method of Mikkola and Merritt [120, 121] with the ARAVA in Figures 4.5 and 4.6. In

Figure 4.5 the relative error in the total energy is displayed as a function of time. For the numerical computations we have taken $m_1 = 0.96$, $m_2 = 0.04$, $\boldsymbol{x}(0) = (0.50, 0.25, 1.00)$, $\boldsymbol{v}(0) = (0.20, -1.00, 0.45)$ and $\boldsymbol{s}(0) = (0.0, 0.0, 0.6)$. The accuracy for the ARAVA is comparable to that of the generalized midpoint method. Figure 4.6 shows the relative CPU times of the ARAVA with respect to the generalized midpoint method as a function of the inverse perturbation parameter, c^2.

For relatively strong perturbations ($\epsilon \geq 10^{-5}$) the GBS extrapolation method combined with the ARAVA is approximately twice as fast as the same extrapolation algorithm with the generalized midpoint method. Moderate perturbations ($\epsilon \approx 10^{-6}$) tip the balance in favour of the GBS extrapolation method with the generalized midpoint method instead of the ARAVA. Previous studies [120] suggest that for weak perturbations ($\epsilon \leq 10^{-6}$), the implicit midpoint method has a slight computational advantage over the generalized midpoint method and thus the ARAVA, as the implicit midpoint method requires approximately three or less iterations, and thus three or less vector field evaluations per step to solve the augmented velocity update, whereas the ARAVA always requires one evaluation per step for the unperturbed (Newtonian) augmented acceleration, and three evaluations per step for the perturbed (post-Newtonian) augmented acceleration.

4.3 Numerics of Attitude Dynamics

To close, we now take a look at the attitude dynamics of artificial satellites and spacecraft in low-altitude orbits around the Earth. Most geometrically oriented studies have mainly focussed on the numerical integration of Euler's equations for the torque-free rigid body [8, 100, 114, 142], as it is a prototype of a Lie–Poisson system. Numerical Lie–Poisson integration dates back twenty years to the works of Ge and Marsden [55], and Channell and Scovel [32]. Standard approaches to the numerical integration of the Hamiltonian equations for torque-free rigid bodies include constraint algorithms, such as RATTLE and SHAKE, the discrete Moser–Veselov algorithm, and splitting methods; an overview can be found in the book by Hairer et al. [66]. Some work has been done on rigid bodies with control and internal torques as well [44, 73, 141]. Touma and Wisdom [155] studied the motion of mutually interacting rigid bodies in the Serret–Andoyer formalism [20, 63] with their eyes on planetary systems. Similarly, Lee et al. [93] looked at the full-body problem and numerical Lie–Poisson integration. The (time-transformed) implicit midpoint integrator was used by Palmer et al. [136]

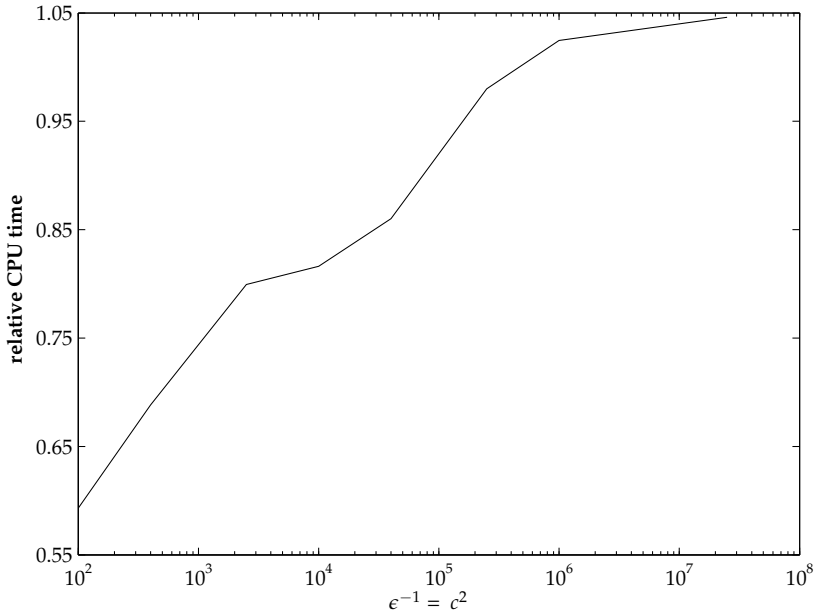


FIGURE 4.6: Average computation times of the ARAVA relative to generalized midpoint for different values of c^2, the inverse of the perturbation parameter.

in a similar context, although their approach relied fully on quaternions, and they did not look at attitude estimation.

We shall come to the numerical aspects of the implicit midpoint method relevant to the problem at hand in Sections 4.3.1 to 4.3.3. Aspects of attitude estimation with the implicit midpoint method are to be found in Section 4.4.

4.3.1 The Implicit Midpoint Method

For the numerical integration of the autonomous differential equations $\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x})$ the implicit midpoint method with step size h can be used:

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + h\boldsymbol{f}\left(\frac{\boldsymbol{x}_{k+1} + \boldsymbol{x}_k}{2}\right), \tag{4.28}$$

which is equivalent to

$$\bar{\boldsymbol{x}} = \boldsymbol{x}_k + \frac{h}{2} \boldsymbol{f}(\bar{\boldsymbol{x}}), \tag{4.29}$$

where $\bar{\boldsymbol{x}}$ is the mean value: $2\bar{\boldsymbol{x}} = \boldsymbol{x}_k + \boldsymbol{x}_{k+1}$. For non-linear vector functions $\boldsymbol{f}$ one can seldom solve $\bar{\boldsymbol{x}}$ explicitly from equation (4.29), in which cases one has to take refuge to numerical methods, such as for example iterative schemes. It is a variational integrator, and it can in fact be computed with **VarInt** using the CreateVarInt module:

```
> restart;                                    #clear memory
> with(VarInt):                               #load VarInt
> etc:=1,L,F,p,q,h:                           #shorthand
> CreateVarInt(-1..1,[-1,0],[0,2],etc);       #alternative 1
> CreateVarInt(-1..1,[0,1],[2,0],etc);        #alternative 2.
```

The implicit midpoint method is an *almost* Lie–Poisson integrator, as it conserves the Lie–Poisson structure up to second order in the step size h [8]. Lie–Poisson integrators can be constructed for any Hamiltonian $\mathscr{H}: \mathcal{P} \to \mathbb{R}$ that can be split into several components, $\mathscr{H}(\boldsymbol{\eta}) = \sum_{k=1}^{K} \mathscr{H}_k(\eta_k)$ with $K = \dim \mathcal{P}$. The individual vector fields $X_{\mathscr{H}_k} = \mathbf{J}(\boldsymbol{\eta}) \nabla \mathscr{H}_k(\eta_k)$ can be integrated separately. Any symplectic integrator with the discrete flow $\Phi(h)$ becomes an explicit first-order Lie–Poisson integrator with respect to the step size h for the composition of the discrete flows of the individual components $\Phi_1(h) \circ \ldots \circ \Phi_K(h)$. A reversible second-order Lie–Poisson integrator is given by the symmetric composition $\Phi_1(h/2) \circ \ldots \circ \Phi_K(h) \circ \ldots \circ \Phi_1(h/2)$, as shown by McLachlan [114] and Reich [142] independently. For Euler's equations (3.41) we have already seen that $\mathscr{H}_k = \frac{1}{2} L_k^2 / I_k$, so that L_k is constant for each k:

$$\begin{array}{lllllllll}
i=1: & \dot{L}_1 & = & 0, & \dot{L}_2 & = & L_1 L_3 / I_1, & \dot{L}_3 & = & -L_1 L_2 / I_1, \\
i=2: & \dot{L}_1 & = & -L_2 L_3 / I_2, & \dot{L}_2 & = & 0, & \dot{L}_3 & = & L_1 L_2 / I_2, \\
i=3: & \dot{L}_1 & = & L_2 L_3 / I_3, & \dot{L}_2 & = & -L_1 L_3 / I_3, & \dot{L}_3 & = & 0,
\end{array}$$

where $\mathbf{J}(\boldsymbol{L})$ is as in equation (3.42). Similar equations hold for each of the principal axes (3.34), where now the Hamiltonian function of the ith principal axis is $\mathscr{H}_k^{(i)} = \frac{1}{2} \left(e_k^{(i)} \right)^2$ for $i = 1, 2, 3$, and the structure matrix is $\mathbf{J}(-\boldsymbol{\omega})$.

For the Euler–Poinsot problem the angular velocity is constrained to the intersection of the kinetic-energy surface and the angular-momentum surface. In angular momentum space these constitute an ellipsoid and a sphere respectively, whereas in angular velocity space they are both ellipsoids. In Figure 4.7 we have drawn the exact solution in terms of Jacobi's elliptic functions (see Section 3.3.2) for $\mathsf{I} = \operatorname{diag}(2, 1, \frac{2}{3})\,\mathrm{kg\,m^2}$ and $\boldsymbol{\omega}(0) = (0.4535, 0, 0.891)\,\mathrm{rad\,s^{-1}}$ in black. The

elliptic functions have been computed to quad precision with the arithmetic-geometric mean method, described in Abramowitz and Stegun [2], p. 577. From the plot it is clear that the angular velocity stays on the intersection of the kinetic energy surface and the angular momentum surface, which proves the validity and accuracy of the method. The kinetic energy and the square of the angular momentum are conserved exactly by construction. We have double-checked these results with a 64th-order numerical Taylor series comparison with the aid of the computer algebra system Maple and the freely available TAYLOR package [80] with extended precision; all results are in agreement to within machine (double) precision.

The results for the second-order implicit midpoint method (4.36) with step size $h = 0.1\,\mathrm{s}$ are shown in Figure 4.7 as well. The implicit midpoint method conserves exactly the kinetic energy and angular momentum, and therefore reproduces the exact trajectory. Furthermore, the orthogonality of the matrix (3.33) is preserved throughout.

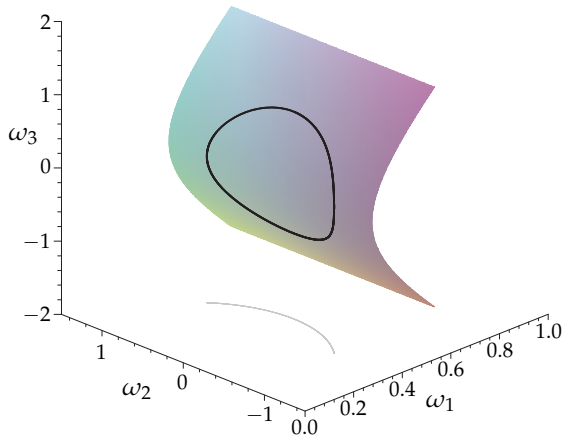


FIGURE 4.7: Exact and numerical solutions to equations (3.30) with $I_1 = 2$, $I_2 = 1$, $I_3 = \frac{2}{3}\,\mathrm{kg\,m^2}$ and $\boldsymbol{\omega}(0) = (0.4535, 0, 0.891)\,\mathrm{rad\,s^{-1}}$. Both the analytical and numerical trajectory in angular velocity space (black) remain on the intersection of the kinetic energy surface and the angular momentum surface. The projections of the trajectories onto the (ω_1, ω_2)-plane (grey) are also displayed. The units of all axes are radians per second.

The phase error of the implicit midpoint method along the solution can be corrected for, as argued by Palmer et al. [136]. In the absence of external torques

the true time step τ is related to the integrator time step h by

$$\tau = \prod_{i=1}^{3} \frac{\bar{\omega}_i}{\omega_i} h. \tag{4.30}$$

Notice that the product is ill-defined whenever at least one component of the angular velocity becomes smaller than the computational precision. This problem is partially balanced by the fact that $\bar{\boldsymbol{\omega}}$ is roughly equal to $\boldsymbol{\omega}$ for small enough step sizes. One generally expects these corrections to be minimal. Moreover, if only one of the factors in the triple product changes its sign, then the time increment does so too, which could temporarily lead to a slight backward motion in time.

We have plotted the time scaling factor $\chi \equiv \prod_{i=1}^{3} \bar{\omega}_i / \omega_i$ as a function of the iteration number in the integration in Figure 4.8. We have used the same initial conditions as before, and an integrator step size of $h = 0.5\,\mathrm{s}$. The graph is generic in the sense that for all initial values the overall shape of the curve remains similar, although the positions and heights of the peaks may vary. The dashed line represents the actual average over all values of χ, whereas the dotted line is the mean for the restricted values $0 \leq \chi \leq 2$. As is clear from the graph, χ remains close to its mean of unity but 'diverges' at multiple instances in both the positive and negative direction. This leads to precipitous shifts forwards and backwards in time. The time transformation (4.30) is therefore impractical for our purposes, and we discard it henceforth.

4.3.2 Higher-Order Methods

Instead of a mere second-order implicit midpoint method, where $\boldsymbol{x}_k \mapsto \boldsymbol{x}_{k+1} = \Phi_h^{\mathrm{IM}}(\boldsymbol{x}_k)$ according to equation (4.28), we often require a higher-order integrator. Since the implicit midpoint method is a symplectic and time-reversible algorithm, higher-order methods can be constructed rather easily by the concatenation of several implicit midpoint methods with different time steps, as we have already seen before. In fact, the composition of several implicit midpoint methods yields the so-called diagonally implicit Runge–Kutta methods, for which $a_{ij} = 0$ for $i < j$, as in equation (4.31) below. Any symplectic diagonally implicit Runge–Kutta method with $b_i \neq 0$ is equivalent to the composition $\Phi_{b_s h}^{\mathrm{IM}} \circ \ldots \circ \Phi_{b_2 h}^{\mathrm{IM}} \circ \Phi_{b_1 h}^{\mathrm{IM}}$ [66, p. 148]. This composition is symmetric as well.

Alternatively, higher-order Runge–Kutta integrators that conserve quadratic invariants may be used. Here there are two options: either one can use the Gauss–Legendre methods, which are all implicit and of order $2s$, so that one

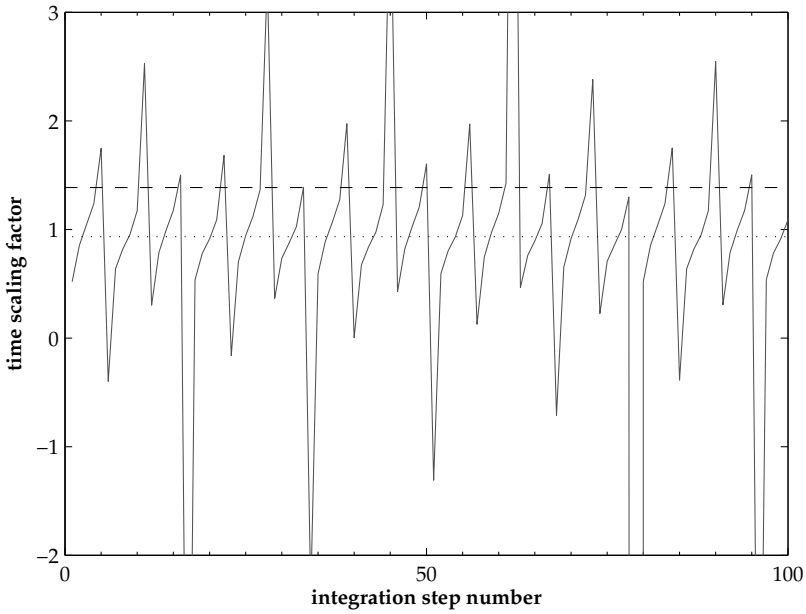


FIGURE 4.8: Evolution of the time scaling factor in the time-transformed implicit midpoint scheme. On the horizontal axes we have taken the integration step number, which is related to the time by equation (4.30). The dashed line is the mean value for all calculated values of the time scaling factor. The dotted line represents the average value of the time scaling function, where 'divergent' values larger than two and smaller than zero have been discarded.

obviously needs $s > 1$, or one can construct higher-order methods by exploiting the W-transformation [66, p. 235].

The implicit midpoint rule is a second-order case of the more general implicit s-stage Runge–Kutta methods:

$$\begin{aligned} \boldsymbol{k}_i &= \boldsymbol{f}\left(t_k + \boldsymbol{c}_i h, \boldsymbol{x}_k + h\sum_{j=1}^{s} a_{ij}\boldsymbol{k}_j\right), \\ \boldsymbol{x}_{k+1} &= \boldsymbol{x}_k + h\sum_{i=1}^{s} b_i \boldsymbol{k}_i. \end{aligned} \tag{4.31}$$

For the implicit midpoint rule $a_{11} = c_1 = 1/2$ and $b_1 = 1$. It is well known that Runge–Kutta methods conserve quadratic invariants if these are symplectic, in other words if their coefficients satisfy $b_i a_{ij} + b_j a_{ji} = b_i b_j$ for $i, j = 1, \ldots, s$ [66, p. 192].

The coefficients of s-stage Runge–Kutta methods are often written conveniently in Butcher tables:

$$\begin{array}{c|ccc} c_1 & a_{11} & \ldots & a_{1s} \\ \ldots & \ldots & & \ldots \\ c_s & a_{s1} & \ldots & a_{ss} \\ \hline & b_1 & \ldots & b_s \end{array}.$$

The implicit midpoint method is a one-stage Gauss–Legendre method of second order. The Hammer–Hollingsworth method is a two-stage fourth-order Runge–Kutta Gauss–Legendre algorithm, and its Butcher table reads:

$$\begin{array}{c|cc} \frac{1}{2} - \frac{1}{2\sqrt{3}} & \frac{1}{4} & \frac{1}{4} - \frac{1}{2\sqrt{3}} \\ \frac{1}{2} + \frac{1}{2\sqrt{3}} & \frac{1}{4} + \frac{1}{2\sqrt{3}} & \frac{1}{4} \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}. \tag{4.32}$$

The sixth-order Runge–Kutta Gauss–Legendre method is characterized by

$$\begin{array}{c|ccc} \frac{1}{2} - \frac{\sqrt{3}}{2\sqrt{5}} & \frac{5}{36} & \frac{2}{9} - \frac{1}{\sqrt{15}} & \frac{5}{36} - \frac{1}{6\sqrt{5}} \\ \frac{1}{2} & \frac{5}{36} + \frac{\sqrt{15}}{24} & \frac{2}{9} & \frac{5}{36} - \frac{\sqrt{15}}{24} \\ \frac{1}{2} + \frac{\sqrt{3}}{2\sqrt{5}} & \frac{5}{36} + \frac{1}{6\sqrt{5}} & \frac{2}{9} + \frac{1}{\sqrt{15}} & \frac{5}{36} \\ \hline & \frac{5}{18} & \frac{4}{9} & \frac{5}{18} \end{array}. \tag{4.33}$$

For an overview of the coefficients for the Gauss–Legendre methods up to tenth order we refer to the pioneering work of Butcher [26]. All these algorithms preserve the orthogonality of the attitude matrix $\mathbf{M}$.

4.3.3 Attitude Propagation

Before moving on to the issues related to attitude estimation, we pause to recast equation (3.46) into a slightly more manageable form:

$$\dot{\boldsymbol{\omega}} = \boldsymbol{\alpha}(\boldsymbol{\omega}) + \boldsymbol{\beta}\big(\boldsymbol{e}^{(3)}\big), \tag{4.34}$$

where the components of the vector functions on the right-hand side are defined as

$$\alpha_i(\boldsymbol{\varphi}) = \sum_{j,k=1}^{3} \epsilon^{ijk} I_j \varphi_j \varphi_k / I_i,$$

and $\beta_i(\boldsymbol{\varphi}) = -3n^2\alpha_i(\boldsymbol{\varphi})$. A one-step propagation with the implicit midpoint method for the systems (4.34) and (3.48) can now be written as

$$\left\{\begin{array}{rcl} \bar{\boldsymbol{\omega}} & = & \boldsymbol{\omega}_k + \dfrac{h}{2}\left[\boldsymbol{\alpha}\left(\bar{\boldsymbol{\omega}}\right) + \boldsymbol{\beta}\left(\bar{\boldsymbol{e}}^{(3)}\right)\right], \\ \bar{\boldsymbol{e}}^{(2)} & = & \boldsymbol{e}_k^{(2)} - \dfrac{h}{2}\bar{\boldsymbol{\omega}} \wedge \bar{\boldsymbol{e}}^{(2)}, \\ \bar{\boldsymbol{e}}^{(3)} & = & \boldsymbol{e}_k^{(3)} - \dfrac{h}{2}\left[\bar{\boldsymbol{\omega}} + n\bar{\boldsymbol{e}}^{(2)}\right] \wedge \bar{\boldsymbol{e}}^{(3)}, \end{array}\right. \tag{4.35}$$

from which we solve the mean values $\bar{\boldsymbol{\omega}}$, $\bar{\boldsymbol{e}}^{(2)}$, and $\bar{\boldsymbol{e}}^{(3)}$ numerically, and after which the updated values are found from the expressions $\boldsymbol{\omega}_{k+1} = 2\bar{\boldsymbol{\omega}} - \boldsymbol{\omega}_k$, $\boldsymbol{e}_{k+1}^{(2)} = 2\bar{\boldsymbol{e}}^{(2)} - \boldsymbol{e}_k^{(2)}$, and similarly for $\boldsymbol{e}^{(3)}$. The equations for $\bar{\boldsymbol{e}}^{(2)}$ and $\bar{\boldsymbol{e}}^{(3)}$ can be solved explicitly once $\bar{\boldsymbol{\omega}}$ is known, for they are linear. The equation for $\bar{\boldsymbol{\omega}}$, however, cannot be solved analytically. Here one's only recourse are numerical techniques. Numerical experiments with equations (4.35) have shown that fixed-point iterations are to be preferred to methods based on Newton–Raphson iteration, since the decomposition of the Jacobian matrix requires more computational effort than direct fixed-point iterations.

Even so, we can rewrite equations (4.35) in such a way that the iterative solutions can be found with less computational operations by scaling the variables by the factor $h/2$. We define

$$\boldsymbol{\vartheta} = \frac{h}{2}\boldsymbol{\omega}, \quad \boldsymbol{\delta} = \frac{h}{2}\boldsymbol{e}^{(2)}, \quad \boldsymbol{\varepsilon} = \frac{h}{2}\boldsymbol{e}^{(3)},$$

so that

$$\left\{\begin{array}{rcl} \bar{\boldsymbol{\vartheta}} & = & \boldsymbol{\vartheta}_k + \left[\boldsymbol{\alpha}\left(\bar{\boldsymbol{\vartheta}}\right) + \boldsymbol{\beta}\left(\bar{\boldsymbol{\varepsilon}}\right)\right], \\ \bar{\boldsymbol{\delta}} & = & \boldsymbol{\delta}_k - \bar{\boldsymbol{\vartheta}} \wedge \bar{\boldsymbol{\delta}}, \\ \bar{\boldsymbol{\varepsilon}} & = & \boldsymbol{\varepsilon}_k - \left[\bar{\boldsymbol{\vartheta}} + n\bar{\boldsymbol{\delta}}\right] \wedge \bar{\boldsymbol{\varepsilon}}. \end{array}\right. \tag{4.36}$$

4.4 Numerics of Attitude Estimation

We shall now describe how the attitude of a satellite can be estimated from observations with an extended Kalman filter [148, 157]. Because the equations of motion are highly non-linear, a linearized Kalman filter is not appropriate; several numerical experiments have established that the estimated solution diverges within seconds after the initialization, so that either one has to restart the estimator time and again, or one has to look at alternatives, such as the extended Kalman filter, which lends itself perfectly for parameter-adaptive filtering as well.

4.4.1 Extended Kalman Filter

Since we are dealing with a continuous, non-linear dynamical system and sampled observations, we use a standard, hybrid formulation that reflects the continuous/discrete duality. The state of the dynamical system and the covariance are propagated numerically with the implicit midpoint integrator, whereas their updates are calculated using a discrete-time approach. Let $\boldsymbol{x}$ denote the state of the n-dimensional dynamical system, which is governed by

$$\begin{cases} \dot{\boldsymbol{x}}(t) &= \boldsymbol{f}\left(\boldsymbol{x}(t), t\right) \\ \boldsymbol{x}(0) &= \boldsymbol{x}_0 \end{cases} \tag{4.37}$$

and relates to the m-dimensional observational data through the observation equations:

$$\boldsymbol{z}(t) = \boldsymbol{h}\left(\boldsymbol{x}(t), t\right) + \boldsymbol{n}(t).$$

Here we have assumed that the noise $\boldsymbol{n}(t)$ is additive, which means that it decouples from the observations themselves. Kalman filters are known to be optimal estimators for Gaussian noise, whereas for non-Gaussian noise they are only suboptimal. For a more general setting that includes random input and control variables we refer to the aforementioned literature and references therein.

The observation function $\boldsymbol{h}$ determines which variables are experimentally accessible, and how the output relates to the observations $\boldsymbol{z}$. The dimension of $\boldsymbol{h}$ is in principle not related to the dimension of the dynamical system at all.

In extended Kalman filters the error propagation is assumed to be linear. Hence, in accordance with common nomenclature, let $\mathbf{F}(t)$ be the Jacobian matrix of $\boldsymbol{f}(t)$ and $\mathbf{H}(t)$ the linearization of the observation function $\boldsymbol{h}(t)$ with respect to the state $\boldsymbol{x}$ around some reference solution. The observation matrix $\mathbf{H}(t)$ accounts for the conversion between the state outputs and the observations. For a generic initial-value problem (4.37), where $\boldsymbol{x} \in \mathbb{R}^n$ and $\boldsymbol{f}$ is a differentiable vector field, the linearized equation about any solution, which may be the trivial solution without loss of generality, is $\mathrm{d}\dot{\boldsymbol{x}}(t) = \mathbf{F}(t)\mathrm{d}\boldsymbol{x}(t)$. The Jacobian matrix is to be evaluated on the reference solution. The error state-transition matrix can be calculated by solving the initial-value problem and

$$\begin{cases} \dot{\boldsymbol{\Phi}}(t, t_0) &= \mathbf{F}(t)\boldsymbol{\Phi}(t, t_0) \\ \boldsymbol{\Phi}(t_0, t_0) &= \mathbf{I} \end{cases} \tag{4.38}$$

simultaneously. The state-transition matrix is simply the Jacobian matrix for the transformation $\boldsymbol{x}(t_0) \mapsto \boldsymbol{x}(t)$, which are related through the flow. In particular,

$\mathbf{\Phi}(t, t_0)$ propagates the initial differential state $\mathrm{d}\boldsymbol{x}(t_0)$ to $\mathrm{d}\boldsymbol{x}(t) = \mathbf{\Phi}(t, t_0)\mathrm{d}\boldsymbol{x}(t_0)$. The set $\{t \in \mathbb{R} \mid \mathbf{\Phi}(t, t_0)\}$ is a one-parameter group of diffeomorphisms of $\mathbb{R}^n$, as it inherits its properties from the exact, non-linear flow through the process of linearization. As is clear from the system (4.38), whenever the Jacobian matrix is independent of time,

$$\mathbf{\Phi}(t, t_0) = \exp\left[\mathbf{F}(t - t_0)\right]. \tag{4.39}$$

Let $\mathbf{Q}$ denote the process noise matrix, and $\mathbf{R}$ the measurement residual matrix. The best estimates of the state and covariance matrix $\mathbf{P}$ are calculated from [148, 157]

$$\begin{aligned}
\boldsymbol{x}_k^- &= \Phi_h^N\left(\boldsymbol{x}_{k-1}^+, t\right) \\
\mathbf{P}_k^- &= \mathbf{\Phi}_{k-1}\mathbf{P}_{k-1}^+\mathbf{\Phi}_{k-1}^T + \mathbf{Q}_{k-1} \\
\mathbf{K}_k &= \mathbf{P}_k^-\mathbf{H}_k\left[\mathbf{H}_k\mathbf{P}_k^-\mathbf{H}_k^T + \mathbf{R}_k\right]^{-1} \\
\boldsymbol{x}_k^+ &= \boldsymbol{x}_k^- + \mathbf{K}_k\left[\boldsymbol{z}_k - \mathbf{H}_k\boldsymbol{x}_k^-\right] \\
\mathbf{P}_k^+ &= \mathbf{P}_k^- - \mathbf{K}_k\mathbf{H}_k\mathbf{P}_k^-,
\end{aligned} \tag{4.40}$$

where the superscript minus (plus) signs represent the values prior to (after) the observations, and where we have suppressed the dependencies of all variables and matrices on time. The first equation propagates the state with a numerical integrator with step size h from the previous to the current observation time in N iterations. Convergence of the filter depends on the integrator step size; ideally, $N \geq 1$, so that the step size is smaller than the time between observations. The second equation advances the covariance matrix with the error state-transition matrix $\mathbf{\Phi}$. The third equation computes the Kalman gain matrix, which is then used in the fourth and fifth equations to compute the new estimations of the state vector and covariance matrix. The most time-consuming steps in the estimation process are the state and covariance predictions with the numerical integrator; these are the most critical as well, since any error in the propagation affects the quality of the estimation, and hence the convergence. At each estimation, that is for all times of the observations, both the error state-transition matrix and the observation matrix have to be recalculated.

In our case of interest the vector fields do not depend on time explicitly, and hence neither does the Jacobian matrix. This means that we can approximate the error state-transition matrix by its truncated Taylor series from equation (4.39) at each moment in the estimation process. For the Jacobian matrix we

need the partial derivatives of the vector field $\boldsymbol{f}$ with respect to the state vector $\boldsymbol{x} = \left(\boldsymbol{\omega}, \boldsymbol{e}^{(2)}, \boldsymbol{e}^{(3)}\right)$. The non-zero components of the partial derivatives are

$$
\begin{aligned}
\frac{\partial \omega_i}{\partial \omega_j} &= \sum_{k,l=1}^{3} \epsilon^{ikl} I_k \left(\delta_k^j \omega_l + \delta_l^j \omega_k\right) / I_i, \\
\frac{\partial \omega_i}{\partial e_j^{(3)}} &= -3n^2 \sum_{k,l=1}^{3} \epsilon^{ikl} I_k \left(\delta_k^j e_l^{(3)} + \delta_l^j e_k^{(3)}\right) / I_i, \\
\frac{\partial e_i^{(2)}}{\partial \omega_j} &= \sum_{k,l=1}^{3} \epsilon^{ikl} \delta_k^j e_l^{(2)}, \\
\frac{\partial e_i^{(2)}}{\partial e_j^{(2)}} &= \sum_{k,l=1}^{3} \epsilon^{ikl} \omega_k \delta_l^j, \\
\frac{\partial e_i^{(3)}}{\partial \omega_j} &= \sum_{k,l=1}^{3} \epsilon^{ikl} \delta_k^j e_l^{(3)}, \\
\frac{\partial e_i^{(3)}}{\partial e_j^{(2)}} &= n \sum_{k,l=1}^{3} \epsilon^{ikl} \delta_k^j e_l^{(3)}, \\
\frac{\partial e_i^{(3)}}{\partial e_j^{(3)}} &= \sum_{k,l=1}^{3} \epsilon^{ikl} \left[\omega_k + n e_k^{(2)}\right] \delta_l^j.
\end{aligned}
\tag{4.41}
$$

The method described above is adequate when observations yield the instantaneous orientation of the principal axes with respect to some initial reference frame. These measurements, which are referred to as absolute measurements as they are based on the actual position of the spacecraft in orbit, come from attitude sensors, such as star, Sun or Earth sensors, and magnetometers. If, for instance, one measures the angular velocity and two of the three principal axes, then the observation matrix $\mathbf{H}$ becomes the identity matrix, which remains constant throughout the estimation procedure. Relative measurements with for example gyroscopes can often achieve a higher accuracy, although the observation matrix becomes more complicated.

Alternatively, we may calculate the state update estimates with the implicit midpoint method directly, that is, without the error state-transition matrix,

which is customary in basic differential correction algorithms:

$$\begin{cases} \mathrm{d}\bar{\boldsymbol{\vartheta}} &= \mathrm{d}\boldsymbol{\vartheta}_k + \left[\mathrm{d}\boldsymbol{\alpha}\left(\bar{\boldsymbol{\vartheta}}\right) + \mathrm{d}\boldsymbol{\beta}\left(\bar{\boldsymbol{\varepsilon}}\right)\right] \\ \mathrm{d}\bar{\boldsymbol{\delta}} &= \mathrm{d}\boldsymbol{\delta}_k - \mathrm{d}\bar{\boldsymbol{\vartheta}} \wedge \bar{\boldsymbol{\delta}} - \bar{\boldsymbol{\vartheta}} \wedge \mathrm{d}\bar{\boldsymbol{\delta}} \\ \mathrm{d}\bar{\boldsymbol{\varepsilon}} &= \mathrm{d}\boldsymbol{\varepsilon}_k - \left[\mathrm{d}\bar{\boldsymbol{\vartheta}} + n\mathrm{d}\bar{\boldsymbol{\delta}}\right] \wedge \bar{\boldsymbol{\varepsilon}} - \left[\bar{\boldsymbol{\vartheta}} + n\bar{\boldsymbol{\delta}}\right] \wedge \mathrm{d}\bar{\boldsymbol{\varepsilon}}, \end{cases} \tag{4.42}$$

where

$$\mathrm{d}\alpha_i(\boldsymbol{\varphi}) = \sum_{j,k=1}^{3} \epsilon^{ijk} I_j \left(\varphi_k \mathrm{d}\varphi_j + \varphi_j \mathrm{d}\varphi_k\right) / I_i, \tag{4.43}$$

and $\mathrm{d}\beta_i(\boldsymbol{\varphi}) = -3n^2 \mathrm{d}\alpha_i(\boldsymbol{\varphi})$. The final values for the differentials are then to be computed from these mean values as before. These linearized equations (4.42) can be solved either exactly or by simultaneous iteration with equations (4.36).

We now consider the attitude dynamics with the gravity-gradient torque. For our computations we have used $\mathsf{I} = \mathrm{diag}\,(160, 142, 36)\ \mathrm{kg\,m^2}$, $r = 1.1 r_\oplus$, where $r_\oplus$ is the mean Earth radius, and $\boldsymbol{\omega}(0) = (0.1978, 0, 12, 0.26)\ \mathrm{rad\,s^{-1}}$, which corresponds to a medium-sized satellite in a low Earth orbit with an approximate altitude of 640 km. For the initial orientation of the principal axes we have taken a set of Cartesian unit vectors for simplicity. We have calculated the solutions with the implicit midpoint method with $h = 0.25\,\mathrm{s}$, the TAYLOR package with quad precision, and Maple's built-in numerical Taylor series method (dsolve[taylorseries]) of order 64, of which the latter served as a point of reference for the accuracy. Interestingly, the TAYLOR integrator fails to preserve the orthogonality of the attitude matrix $\mathbf{M}$. The implicit midpoint method preserves orthogonality, as expected, and produces the correct solutions to within $\mathcal{O}(h^2)$, although there is the expected systematic error in the phase. These are translated to a secular and periodic contribution, as visible in Figure 4.9, where we have drawn the differences at discrete times between the 'exact' and the numerical solution of the implicit midpoint algorithm (with $h = 0.01\,\mathrm{s}$) in the angular velocity space $\Delta\boldsymbol{\omega}$ on the left, and its projection onto the $(\Delta\omega_1, \Delta\omega_2)$-plane on the right. The difference 'spirals' outwards from the origin to create a saddle-shaped cloud. It is this outward motion that gives rise to the secular contribution to the absolute error. The periodic part is due to the fact that the motion in angular velocity space is quasi-periodic.

Integrations with time steps smaller than one-tenth of a second have not shown to improve the accuracy significantly, which peaks at approximately $\mathcal{O}\left(10^{-6}\right)$. For larger step sizes higher-order versions are recommended, as they reduce the error substantially. In all computations shown the co-orbiting energy (3.50) is conserved to within machine precision.

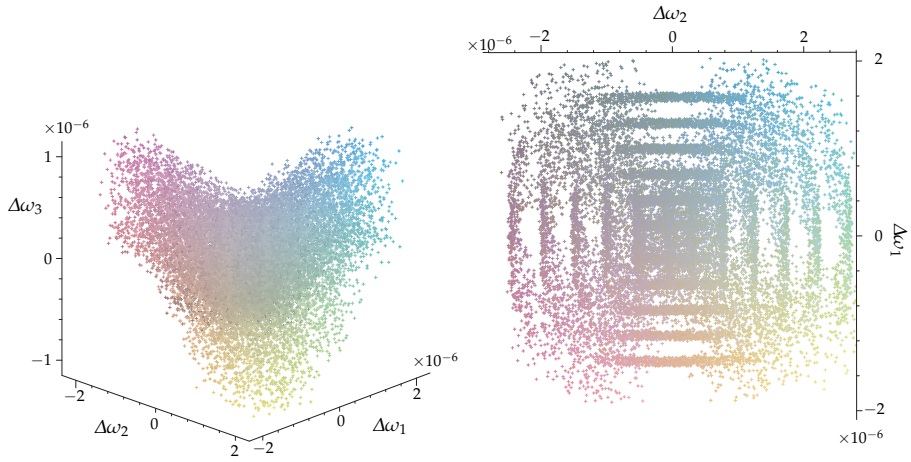


FIGURE 4.9: Difference in the angular velocity between the solution obtained from the 64th-order numerical Taylor series method and the implicit midpoint method with step size $h = 0.01\,\mathrm{s}$ (left), and its projection onto the $(\Delta\omega_1, \Delta\omega_2)$-plane (right). The differences are in units of radians per second.

As mentioned, higher-order Runge–Kutta Gauss–Legendre methods can be used instead. These behave similarly to the higher-order symmetric compositions of the implicit midpoint method in regard to accuracy. In order to compare the computational performance of the implicit midpoint compositions and the Gauss–Legendre methods of orders four and six, we have drawn the relative CPU time overhead with respect to the second-order implicit midpoint method for different time steps in Figure 4.10. All computations were carried out on an Linux laptop with an Intel® Pentium® M 1.4 GHz processor with 512 MB RAM. The fourth- and sixth-order compositions of the implicit midpoint method, and the fourth- and sixth-order Gauss–Legendre methods are denoted by IM-4 and IM-6, and GL-4 and GL-6, respectively. A CPU time overhead $p \geq 0$ means that the computation takes approximately $p + 1$ times longer than the computation with the second order implicit midpoint method, which has $p = 0$. The mean values for the CPU time overhead of the implicit midpoint methods of orders four and six, and the Gauss–Legendre methods of orders four and six are 1.52, 1.80, 3.01 and 3.24, respectively. Therefore, the implicit midpoint method of orders four and six outperform their Gauss–Legendre counterparts by approximately 10% computation time. Since we have found no differences in

accuracy, we recommend the use of the implicit midpoint integrator based on its computational performance.

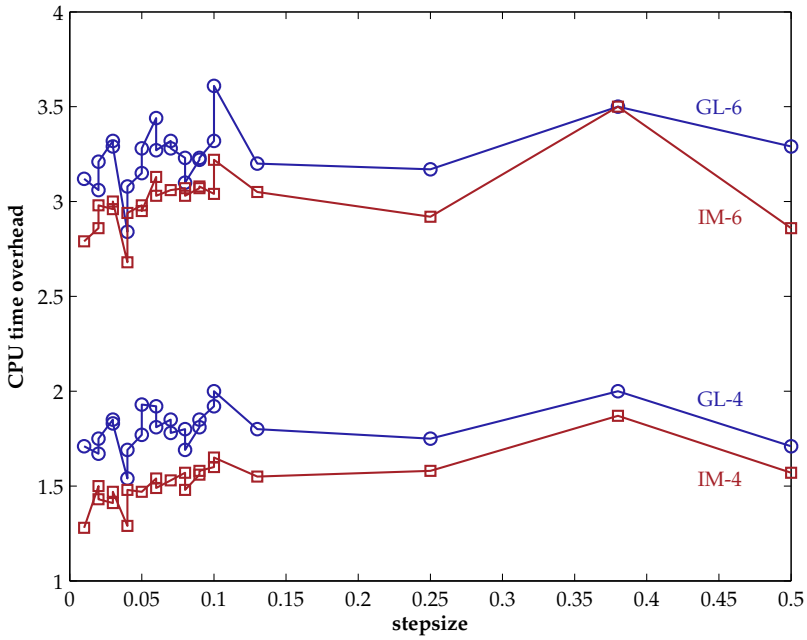


FIGURE 4.10: Normalized CPU time overheads relative to the second-order implicit midpoint method as a function of the step size.

For the estimation of the attitude we have chosen a second-order implicit midpoint scheme to be used in the extended Kalman filter. The observational data are based on a high-accuracy computation with Maple, to which normally distributed noise has been added to model measurement bias and uncertainties in the process of observation. The initial conditions for the Maple calculations have been set to $\boldsymbol{\omega}(0) = (0.2006, -0.012, 0.025)\ \mathrm{rad\,s^{-1}}$, and, again, unit vectors for the principal axes. Furthermore, we have used that $\mathsf{I} = \mathrm{diag}\,(98, 82, 44)\,\mathrm{kg\,m^2}$ and $r = 1.2r_{\oplus}$. The variance of the noise for the angular velocity components has been set to $\sigma_{\omega}^2 = 0.05\,\mathrm{rad^2\,s^{-2}}$, which amount to an accuracy of approximately three degrees per second, and for the principal axes $\sigma_e^2 = 0.25\,\mathrm{m^2}$. The time between consecutive observations is one second, and we use two intermediate

integration steps, that is $h = 0.5\,\mathrm{s}$. The simulated data consist of the angular velocity vector, and the orientation of the second and third principal axes with regards to their initial placement. In Figure 4.11 we have plotted the attitude estimation errors in degrees, these are the differences between the exact and estimated values. Here the estimated attitude matrix has been converted to conventional roll, pitch and yaw angles using an $\boldsymbol{e}^{(3)}$-$\boldsymbol{e}^{(2)}$-$\boldsymbol{e}^{(1)}$ rotation sequence. The graphs show that the attitude estimation errors are bounded to within a few degrees of their near-zero averages. Even for long simulations the extended Kalman filter does not diverge. Strictly speaking, it does not converge either—the covariance matrix does not become zero—, but at least its errors stay bounded; the standard deviations of these errors lie inside the measurement uncertainties.

These estimation errors can be reduced by increasing the number of observations. We have found that an improvement of approximately one degree can be achieved by using observations that are merely 0.1 seconds apart. More integration steps have not shown to have a profound effect on the performance.

4.4.2 Parameter-Adaptive Extended Kalman Filter

The extended Kalman filter described above is appropriate for the estimation of the attitude of a low Earth-orbiting satellite. The performance of the filter depends on the exact knowledge of the mean motion and the principal moments of inertia. Either one has to determine these separately, or one can estimate these parameters simultaneously. Actual convergence of these so-called parameter-adaptive filters depends on many factors: the number of uncertain parameters, their magnitudes of uncertainty, the functional dependence of the observables on the uncertain parameters, the quality of output measurements, and the actual knowledge of system inputs [148, p. 393].

The aforementioned filter can be transformed into a parameter-adaptive filter by augmenting it with the parameters as follows. The state vector is enlarged by the p parameters: $\tilde{\boldsymbol{x}} = (\boldsymbol{x}, \boldsymbol{p})$, so that the equations of motion become

$$\frac{\mathrm{d}}{\mathrm{d}t}\tilde{\boldsymbol{x}} = \tilde{\boldsymbol{f}}\,(\tilde{\boldsymbol{x}}, t)\,,$$

where the $(n + p)$-dimensional augmented vector field $\tilde{\boldsymbol{f}}$ accounts for the dynamics as well as the time evolution of the parameters. For constant parameters, such as in the case of the mean motion or principal moments of inertia, the appropriate augmented vector field is $\tilde{\boldsymbol{f}} = (\boldsymbol{f}, \boldsymbol{0})$. If these parameters cannot be observed directly, then the observation matrix for the augmented system simply

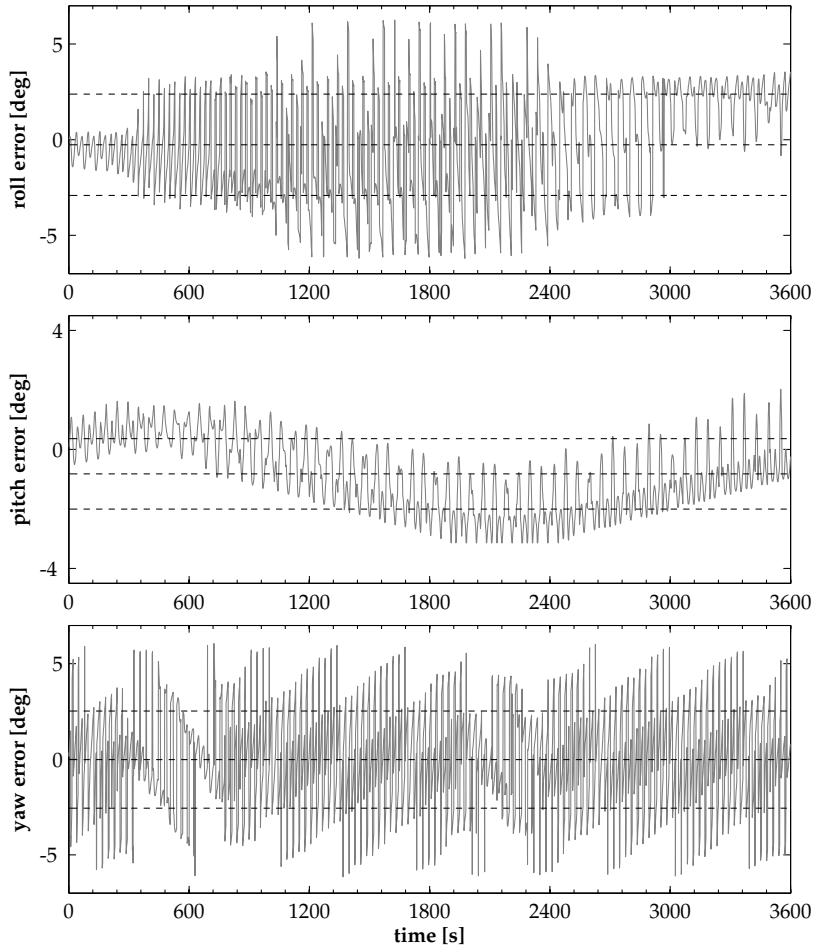


FIGURE 4.11: Evolution of the errors in the Euler angles. The dashed lines are the mean values of the estimation errors, and the dotted lines show the standard deviation of the estimation errors.

becomes $\tilde{\mathbf{H}} = \left(\mathbf{H}\ \mathbf{0}_{k\times p}\right)$, which is obviously an $m \times (n+p)$ matrix, as it maps the new $(n+p)$-dimensional state vector $\tilde{\boldsymbol{x}}$ to the observational variables $\boldsymbol{z}$.

In the case of constant parameters, the Jacobian matrix becomes

$$\tilde{\mathbf{F}}(t) = \begin{pmatrix} \mathbf{F}(t) & \partial \boldsymbol{f}/\partial \boldsymbol{p} \\ \mathbf{0}_{p\times n} & \mathbf{0}_{p\times p} \end{pmatrix},$$

where $\mathbf{F}(t) = \partial \boldsymbol{f}/\partial \boldsymbol{x}$ is the $n \times n$ Jacobian matrix of the dynamical system. The $n \times p$ matrix $\partial \boldsymbol{f}/\partial \boldsymbol{p}$ denotes the Jacobian matrix of the dynamical vector field with respect to the parameters. For $\boldsymbol{p} = (I_1, I_2, I_3)$ we find that the components

$$\frac{\partial f_i}{\partial p_j} = \sum_{k,l=1}^{3} \epsilon^{ikl} \left(\omega_k \omega_l - 3n^2 e_k^{(3)} e_l^{(3)}\right) \left(\delta_k^j - I_k / I_i \delta_i^j\right) / I_i, \tag{4.44}$$

if $i = 1, 2, 3$, and 0 otherwise, for the reduced equations of motion (3.45) and (3.48) with $\boldsymbol{x} = \left(\boldsymbol{\omega}, \boldsymbol{e}^{(2)}, \boldsymbol{e}^{(3)}\right)$.

The error-state transition matrix is $(n+p) \times (n+p)$-dimensional, as it propagates both the original differential state vector and the errors in the parameters. Similarly, the covariance matrix and the process noise matrix become $(n+p) \times (n+p)$-dimensional, the Kalman gain is an $(n+p) \times m$ matrix, and the measurement residual matrix remains $(m \times m)$-dimensional. The combined parameter–state estimate is computed as in equations (4.40) but with all but the observation vector and the measurement residual matrix replaced by their augmented analogues.

We note that if the parameters are not constants, and hence they satisfy non-trivial dynamical equations, then obviously the bottom right $p \times p$ matrix is non-zero, and may become time-dependent.

We have examined the performance of the parameter-adaptive estimation with the implicit midpoint integrator. In our simulations we have estimated the attitude as before with additional estimation of the principal moments of inertia. We have taken the previous simulated observations and assumed for the parameter estimates $\mathsf{I} = \mathrm{diag}\,(100, 81.5, 42)\,\mathrm{kg\,m^2}$.

As one can see from Figure 4.12, the parameter estimation is not accurate: the parameters have not converged to their actual values. In fact, there is a residual in the trace of the error covariance matrix, which indicates that at the end of the simulation the estimates have not converged perfectly. Incidentally, the accuracy of the attitude matrix is affected marginally by the parallel parameter estimation; their errors still remain within a few degrees. Neither decreasing the integration step size, nor increasing the order of the integrator or the number of

observations improves the quality of the parameter estimation. More accurate initial estimates for the moments of inertia indeed decrease the errors of the final estimates, yet they do not yield the desired values. We therefore suspect that the non-linear dependence of the observables on the principal moments of inertia is the main contributor to the failure of the parameter estimation, in line with observations by Crassidis et al. [36].

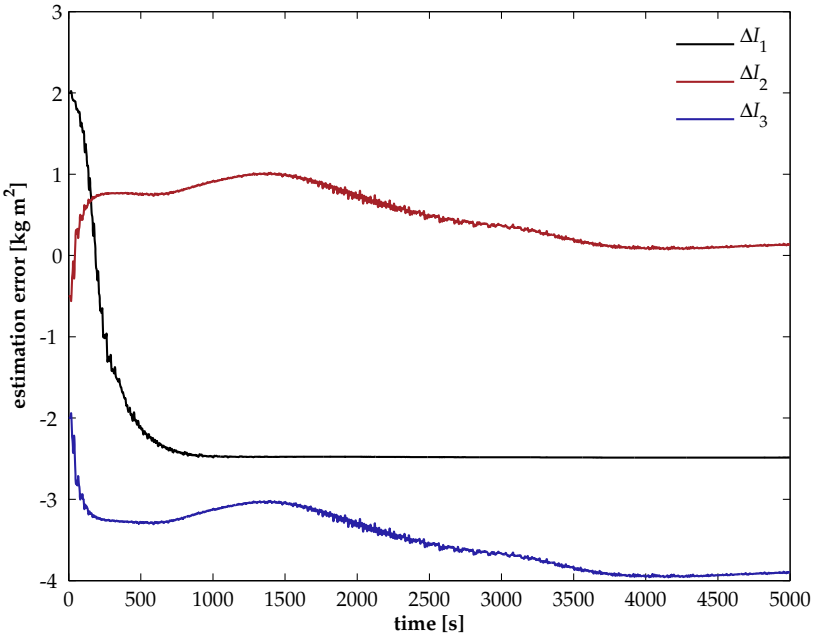


FIGURE 4.12: Errors in the estimation of the principal moments of inertia.

Conclusion and Outlook

As man liberated himself from the bounds of the Earth, he sent forth ever more sophisticated machines to explore the vast void around him. At the core of such a spacecraft or artificial satellite lies complex software that controls and guides it through the heavens on its way to knowledge. In order for such a grand voyage to succeed it is of the essence that the numerics inside that determines the position in space and through time as well as the orientation of the craft with respect to the Earth is accurate, fast, and reliable; without the correct location in space and time the spacecraft cannot fulfil its mission, and without the correct orientation it is unable to relay its findings back to the space operations centre.

Numerical algorithms to calculate the orbital and attitude motion of a satellite that satisfy the requirements mentioned before are geometric numerical integrators. These geometric numerical algorithms can be considered from a variational point of view. We have examined these variational integrators in depth in the treatise at issue in the context of spacecraft dynamics. A quartet of contributions may be identified.

First, it is a well-established fact that simulations of (non-linear) dynamical systems, both with non-conservative forces and without, benefit greatly from the preservation of their geometric structures, especially over long time spans as compared to the characteristic time scales of the systems at hand. Variational integrators, and more generally geometric numerical integrators, are ideally suited for such simulations. The discrete variational formalism is both mathematically natural and computationally practical, as demonstrated by **VarInt**, a library developed for the computer algebra system Maple, with which it is possible to explore and design variational integrators for Lagrangian/Hamiltonian dynamical systems systematically and to arbitrary order. Some of these variational integrators correspond to well-known classes of geometric numerical algorithms, such as the symplectic partitioned Runge–Kutta methods, notably the Gauss collocation and Lobatto IIIA/IIIB methods. Few variational integrators have

previously been reported that lie outside of the standard classification, although the discrete variational formalism is certainly not restricted to it. With the aid of **VarInt** one can venture beyond the geometric numerical algorithms one usually encounters. The computed discrete flow maps can be either general, and serve as templates for generic problems, as demonstrated by numerous examples, or optimized for a specific problem thanks to the symbolic capabilities of a computer algebra system. In addition, we have shown that the performance of these variational integrators depends highly on the particulars of the quadrature formula used to approximate the action functional.

Secondy, to the attitude dynamics and estimation problem we have applied the implicit midpoint method, also a variational integrator, in a new way, namely one that does not require the use of quaternions. In the case of the torque-free motion, the standard implicit midpoint method reproduces the correct solution up to a slight error in the phase. In more realistic scenarios, where the gravity-gradient torque has to be included for satellites in low Earth orbits, the implicit midpoint method has proven to be in remarkable agreement with the 'exact' solution obtained by a high-order numerical Taylor series method with Maple. We have argued that higher-order methods formed by the symmetric compositions of the implicit midpoint method are to be preferred to higher-order Gauss–Legendre methods, as the accuracy is comparable but the former surmounts the latter by 10% in computation time. Given the ease of implementation of the implicit midpoint method of arbitrary order, we believe that the implicit midpoint integrator is an attractive method for the study of the gravity-gradient attitude dynamics; its implementation for non-circular orbits is straightforward, although the expressions become more involved.

The implicit midpoint integrator has also proven to be a good choice for the estimation of the attitude with an extended Kalman filter. The errors of the relative rotation angles are bounded to within a few degrees and lie within the measurement errors. Extended Kalman filters, however, do not provide a solid framework for the combined estimation of attitude and the principal moments of inertia for low Earth-orbiting satellites; convergence to the correct values of the moments of inertia cannot be guaranteed. Our results indicate that the functional dependence of all (output) variables on the moments of inertia is the reason for its failure to determine both the attitude and the parameters correctly.

Third, we have proposed a novel numerical integrator, the algorithmically regularized auxiliary-velocity algorithm, or ARAVA for short, that is both fast and accurate, either on its own or in combination with the Gragg–Bulirsch–Stoer extrapolation method whenever higher accuracy is required. Its main

advantage lies in the fact that is remains explicit, even for velocity-dependent perturbations, yet it requires only one evaluation of the unperturbed force per step, which can reduce the computational time considerably. Both in terms of computational time and accuracy it is on par with conventional algorithms. Because of its symmetry higher-order integrators can easily be constructed by the symmetric composition of the basic second-order algorithm. The algorithm does not require the storage of values from previous iterations because of its resemblance to the standard leapfrog algorithm, which makes it suitable for problems in which memory requirements are more stringent. The algorithm is based on the logarithmic Hamiltonian method, which is exact for the Kepler–Coulomb problem and regularizes the motion, in contrast to many adaptive algorithms in the extended phase space for the N-body problem.

Inside the Gragg–Bulirsch–Stoer extrapolation method the algorithmically regularized auxiliary-velocity algorithm is at least as accurate as standard discretization methods for second-order differential equations with velocity-dependent perturbations, such as the implicit and generalized midpoint methods. At best, the algorithm cuts the computational expense in half compared to the generalized midpoint method. For relatively strong perturbations the proposed method is more efficient than the generalized midpoint method, which may be preferred for weakly perturbed systems, where the regularized auxiliary-velocity algorithm loses its advantage. We emphasize, however, that the performance of our algorithm within the Gragg–Bulirsch–Stoer extrapolation scheme depends on the specifics of the implementation, such as for instance the actual step size control and whether one uses rational or polynomial interpolation. Nevertheless, the algorithmically regularized auxiliary-velocity algorithm seems to be a promising numerical integrator.

Fourth, an analytical solution to the McIntosh–Cisneros–Zwanziger problem, an integrable deformation of the Kepler–Coulomb problem, for any orbit type has been derived. The solution extends and completes the one by Caballero and Elipe [27], for which we have suggested several potentially interesting space applications.

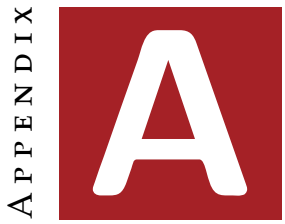
Finally, we suggest some prospective research opportunities based on the contents of this doctoral dissertation. With the development of the library **VarInt** the number of variational integrators available is considerable. Extensive numerical analysis is required to understand each of these families of variational integrators, and to establish their prime applications together with of course their limitations.

The computer algebra aided design of variation integrators with **VarInt** can be made to encompass a wider range of problems that exhibit a variational structure. Possible extensions include (i) Turán–Birkhoff quadrature formulas; (ii) quadrature rules that are exact for non-polynomial integrands, which might have advantages for N-body problems in molecular physics and celestial mechanics, where the singular terms in the discrete Lagrangian might benefit from approximations in terms of rational functions instead of polynomials; (iii) variational force-gradient algorithms [135]; (iv) prolongation-collocation variational integrators [97]; (v) time-reversible splitting techniques in order to accommodate the design of variational integrators for dynamical systems with multiple time scales [78, 95]; (vi) Lie–Poisson and Nambu dynamical systems; (vii) the asynchronous (and multi-symplectic [23, 126]) formalism(s) for partial differential equations; and (viii) an interface for the discrete mechanics and optimal control (DMOC) programme [134], which differs fundamentally from the standard optimal-control framework, as studied by Chyba et al. [33] in a geometric setting. We note that Turán–Birkhoff quadrature formulas as discretizations of the action functional yield derivatives of the force with respect to time, which are usually deemed unphysical. A possible avenue of investigation, however, is the creation of hybrid variational/automatic differentiation (VAD) algorithms, where the underlying algorithms are variational, but where the derivatives of the forces are computed with an automatic differentiation procedure. Thus, the benefits of variational integrators can be maintained while at the same time the power of automatic differentiation is utilized. Conservative forces can in fact be computed from the Lagrangian or Hamiltonian with automatic differentiation too. On the level of numerical software library design, such a technique could provide a clean and simple user interface, where only the Lagrangian or Hamiltonian, the initial values, and the integration step size are required to initiate the numerical computations.

Turning our attention to space applications once more, we can pose the intriguing question whether there are any higher-order variational integrators in the extended phase space that are exact for the Kepler–Coulomb problem, based on for example the Gauss–Lobatto or Clenshaw–Curtis quadrature rules. Additionally, the enquiry into the existence of exact numerical integrators for integrable problems in astrodynamics [see e.g. 110], such as for instance the motion of a particle in the gravitational field of two fixed centres, which may serve as a basis for the J_2 [159–161] (and J_3 [162, 163]) dynamics of spacecraft in Earth orbits, or any non-spherically symmetric primary as a matter of fact, could be well worth pursuing.

An efficient geometric attitude–orbit propagator and estimator is of great practical interest. Geometric (Bayesian) estimation offers some clear advantages [158]. Since the full dynamics of any spacecraft in orbit around the Earth has both slow (secular and (quasi-)periodic) and fast ((quasi-)periodic) components, advanced splitting techniques might have to be developed and employed to exploit the various time scales of the dynamical system. Many other applications may be considered as well, such as for instance the development of geometric numerical integration algorithms for the relative motion of a collection of spacecraft, which is of some importance to conceived future formation flight missions.

Ultimately, the differential geometry of dynamical systems offers a vantage point, from which the field of geometric numerical integration becomes clearer. With the computer algebra aided design of variational integrators it is hoped that the field becomes more accessible, in order that many more may come to enjoy the prolific landscape.



Quadrature Rules

Here we describe the symmetric quadrature formulas supported by the library **VarInt** in detail; more information on the Maple codes is available in Appendix B. Sections A.1 and A.2 introduce equidistant numerical integration methods, especially the closed Newton–Cotes quadrature rules, and an adaptation of Romberg's method to variational integration. Gaussian quadrature rules can be found in Section A.3, which includes the Gauss–Legendre, Fejér, Clenshaw–Curtis, and Gauss–Lobatto formulas. In Section A.4 Chebyshev's equal-weight formula is discussed briefly. We conclude with the tanh-sinh, or double exponential, quadrature formula in Section A.5.

A.1 Newton–Cotes Quadrature

The closed Newton–Cotes quadrature formulas approximate definite integrals by approximating the integrand $f\colon \mathbb{R} \to \mathbb{R}$ with an interpolating polynomial evaluated at the node points $x_k = a + kh$, where $k = 0, \ldots, n$, and the step size

$h = (b-a)/(n-1)$:

$$\int_a^b f(x)\,\mathrm{d}x \approx \int_a^b \left\{ \sum_{k=0}^{s} f(x_k)\pi_k(x) \right\} \mathrm{d}x$$
$$= \sum_{k=0}^{s} f(x_k) \underbrace{\int_a^b \pi_k(x)\,\mathrm{d}x}_{w_k}.$$

Here $\{\pi_k(x)\}$ is a polynomial basis, and w_k are known as the weights; these weights are usually calculated by integration of Lagrange polynomials, although one is in principle free to select any polynomial basis for the interpolation.

A.2 Romberg Quadrature

Another family of classical integration formulas with equidistant nodes is the one due to Romberg. Romberg quadrature distinguishes itself from Newton–Cotes quadrature in that it always uses the same basic two-point approximation, the composite trapezium rule, yet recursively by inserting nodes at the centres of all (sub)intervals. The essence of Romberg quadrature is that a Richardson extrapolation procedure [143] is applied to the composite trapezium rule to obtain higher-order approximations to the integral under evaluation.

It is important to note that the composite trapezium rule leads to a continuous approximation of the integrand, yet its derivative with respect to the independent variable is discontinuous at each node. Hence, the naive implementation of Romberg quadrature seems impossible to generate variational integrators, as we require that $\boldsymbol{q} \in \mathcal{C}^1([t_k, t_{k+1}], \mathbb{R})$ for $k = 0, \ldots, N-1$. Nevertheless, we can still use a 'modified' trapezium rule and Richardson extrapolation in conjunction with a sufficiently smooth interpolating function, at the cost of losing the adaptivity of the algorithm. Again, the composite trapezium rule is used as a basic approximation, though now the interpolating function is not piecewise linear but rather it is chosen such that both $\boldsymbol{q}$ and $\dot{\boldsymbol{q}}$ are well-defined at each node.

A.3 Gauss Quadrature

A class of n-point quadrature rules that integrate up to $(2n-1)$st-degree polynomials exactly by evaluating a weighted sum of function values are the Gaus-

sian ones:

$$\int_a^b f(x)\omega(x)\,\mathrm{d}x = \sum_{k=1}^{n} w_k f(x_k) + R_n, \tag{A.1}$$

where the 'optimal' values for the weights w_k depend on the placement of the nodes x_k along the interval $[a, b]$. The integrand is assumed to be sufficiently smooth, specifically it is a $\mathcal{C}^{2n}([-1, 1], \mathbb{R})$ function. R_n denotes the remainder for a Gaussian integration formula with n nodes [149, pp. 180–181],

$$R_n = \frac{f^{(2n)}(\xi)}{(2n)!} \int_a^b \omega(x)\phi_n^2(x)\,\mathrm{d}x,$$

where $a < \xi < b$, and $\phi_n(x)$ is the related nth degree orthogonal polynomial. As usual, $\omega(x)$ denotes a positive weight function appearing in the integrand. In the case of interest for variational integrators the nodes are placed symmetrically over a finite interval, for which $[-1, 1]$ is commonly used. For an integral over an arbitrary but finite interval $[a, b]$ the linear transformation $x \mapsto \frac{1}{2}(b-a)\,x + \frac{1}{2}(a+b)$ can be used. An overview of the various quadrature formulas of the Gauss family can be found in Abramowitz and Stegun [2], Chap. 25. Here we discuss the quadrature rules based on the Legendre, Chebyshev and Lobatto nodes. The nodes for the Gauss–Radau quadrature formulas are not distributed symmetrically across the interval of integration, so that they cannot be used for the design of variational integrators for autonomous dynamical systems.

A.3.1 Gauss–Legendre Quadrature

In Gauss–Legendre quadrature formulas the weight function $\omega(x) = 1$, which is known as the Legendre weight function. The nodes x_k with $k = 1, \ldots, n$ for the n-point Gauss–Legendre quadrature formulas are the zeros of the Legendre polynomials $P_n(x)$. The corresponding weights are given by

$$w_k = \frac{2}{1 - x_k^2} \frac{1}{[P_n'(x_k)]^2}, \tag{A.2}$$

where the prime indicates the derivative with respect to the argument. It is important to note that the zeros of the Legendre polynomials come in pairs, so that the quadrature rule is symmetric about the origin. Furthermore, the zeros lie in the interval $(-1, 1)$, that is they do not include the endpoints.

The fact that the endpoints of the integration interval do not appear explicitly in the quadrature formula means that it is necessary to 'include' the endpoints

by means of extrapolation; the values of the coordinates and their derivatives are indeed specified at one such a point for initial-value problems. The idea is to interpolate the coordinates with an $(n-1)$st degree polynomial through the interior points ($i = 1, \ldots, s-1$), as before, and extrapolate to the endpoints of the integration interval ($i = 0$ and $i = s$). It is then possible to express the first ($i = 1$) and last ($i = s-1$) of the interior points in terms of the remaining interior points and the endpoints. In that way, the endpoints can be included in accordance with the quadrature nodes. Although polynomial extrapolation is notorious for being very inaccurate outside the interval of the interpolation, we assume that the time step h is sufficiently small to overcome the issues associated herewith.

A.3.2 Fejér Quadrature

The formulas due to Fejér [47] are based on Chebyshev polynomials of the first and second kind; they are the analogues of the classical Gauss–Chebyshev quadratures rules, which are defined with respect to non-trivial weight functions. The nodes for the integration rules based on the Chebyshev polynomials of the first and second kind are

$$x_k^{(1)} = \cos\theta_k^{(1)}, \qquad \theta_k^{(1)} = \frac{2k-1}{n}\frac{\pi}{2}, \tag{A.3}$$

and

$$x_k^{(2)} = \cos\theta_k^{(2)}, \qquad \theta_k^{(2)} = \frac{k}{n+1}\pi, \tag{A.4}$$

respectively. The weights are

$$w_k^{(1)} = \frac{2}{n}\left[1 - 2\sum_{j=1}^{\lfloor n/2\rfloor}\frac{\cos\left(2j\theta_k^{(1)}\right)}{4j^2-1}\right], \tag{A.5}$$

and

$$w_k^{(2)} = \frac{4\sin\theta_k^{(2)}}{n+1}\sum_{j=1}^{\lfloor (n+1)/2\rfloor}\frac{\sin\left((2j-1)\,\theta_k^{(2)}\right)}{2j-1}, \tag{A.6}$$

where $\theta_k^{(1)}$ and $\theta_k^{(2)}$ are as before.

Alternatively, the zeros of the nth-degree Chebyshev polynomial of the third kind $V_n(x)$ can be used,

$$x_k^{(3)} = \cos\theta_k^{(3)}, \qquad \theta_k^{(3)} = \frac{2k-1}{2n+1}\pi, \tag{A.7}$$

as well as those of the nth-degree Chebyshev polynomial of the fourth kind $W_n(x)$,

$$x_k^{(4)} = \cos\theta_k^{(4)}, \qquad \theta_k^{(4)} = \frac{2k}{2n+1}\pi. \tag{A.8}$$

The corresponding weights are

$$w_k^{(3)} = \frac{4\sin\theta_k^{(3)}}{n+\frac{1}{2}} \sum_{j=1}^{\lfloor (n+1)/2 \rfloor} \frac{\sin\left((2j-1)\,\theta_k^{(3)}\right)}{2j-1}, \tag{A.9}$$

and

$$w_k^{(4)} = \frac{4\sin\theta_k^{(4)}}{n+\frac{1}{2}} \sum_{j=1}^{\lfloor (n+1)/2 \rfloor} \frac{\sin\left((2j-1)\,\theta_k^{(4)}\right)}{2j-1}, \tag{A.10}$$

respectively, as shown by Notaris [133].

Related to Fejér quadrature formulas is the one by Clenshaw and Curtis [34], which is nothing but Fejér's second rule with the nodes -1 and 1 added. Define

$$\theta_k = \frac{k-1}{n-1}\pi, \qquad k = 1, \ldots, n.$$

The Clenshaw–Curtis nodes are then simply $x_k = \cos\theta_k$, and the associated weights are given by

$$w_k = \frac{c_k}{n}\left[1 - 2 \sum_{j=1}^{\lfloor (n+1)/2 \rfloor} {}^{*} \frac{\cos 2j\theta_k}{4j^2-1}\right], \tag{A.11}$$

where $c_k = 2 - \delta_{0,k \bmod n}$, and $\sum{}^{*}$ signifies that the last term in the sum should be halved.

A.3.3 Gauss–Lobatto Quadrature

Additional Gaussian integration formulas that include both endpoints are the Gauss–Lobatto ones:

$$\int_{-1}^{-1} f(x)\,\mathrm{d}x \approx \frac{2}{n\,(n-1)}\left[f\,(-1)+f\,(1)\right]+\sum_{k=2}^{n-1} w_k f(x_k). \tag{A.12}$$

The interior nodes are the zeros of the derivative of the Legendre polynomials, that is they satisfy $P'_{n-1}(x)=0$, and the interior weights can be calculated to be

$$w_k = \frac{2}{n\,(n-1)}\frac{1}{\left[P_{n-1}(x_k)\right]^2}. \tag{A.13}$$

A.4 Chebyshev Quadrature

Somewhat related to the quadrature formulas of the Gaussian type is the equal-weight integration formula by Chebyshev:

$$\int_{-1}^{-1} f(x)\,\mathrm{d}x \approx \frac{2}{n}\sum_{k=1}^{n} f(x_k). \tag{A.14}$$

The nodes are the solutions to the equation $G_n(x)=0$, where $G_n(x)$ is the polynomial part [70] of

$$F_n(x) = x^n \exp\left[\frac{n}{2}\int_{-1}^{1}\ln\left(1-\frac{t}{x}\right)\mathrm{d}t\right]. \tag{A.15}$$

The integral inside the exponential can be calculated easily:

$$\int_{-1}^{1}\ln\left(1-\frac{t}{x}\right)\mathrm{d}t = -2+(1+x)\ln\left(1+\frac{1}{x}\right)+(1-x)\ln\left(1-\frac{1}{x}\right).$$

The zeros of $G_n(x)$ are known to be real only for $n\leq 7$ and $n=9$. Hence, the use of Chebyshev quadrature is restricted to these values.

A.5 Takahasi–Mori Quadrature

For the numerical computation of integrals over infinite intervals $(-\infty,\infty)$ the composite trapezium rule is noted for its excellent performance in terms of

accuracy and efficiency compared to quadrature formulas with the same density of sampling points [153]. For any analytical function g that vanishes at infinity

$$\int_{-\infty}^{\infty} g(x)\,\mathrm{d}x \approx v \sum_{k=-\infty}^{\infty} g\,(kv),$$

where in practice the infinite sum itself converges often quite rapidly. We can take advantage of the performance of the trapezium rule by applying a variable transformation $x \mapsto \varphi(t)$ to integrals over finite intervals:

$$\begin{aligned}\int_{-1}^{1} f(x)\,\mathrm{d}x &= \int_{-\infty}^{\infty} f\left(\varphi(t)\right)\varphi'(t)\,\mathrm{d}t \\ &\approx v \sum_{k=-\infty}^{\infty} f\,(kv)\,\varphi'\,(kv).\end{aligned}$$

The method proposed by Schwartz [145] involves the transformation $\varphi(t) = \tanh t$, for which the resulting quadrature formula has an asymptotic error of $\mathcal{O}\left(\exp\left[-c\sqrt{M}\right]\right)$ with $v = \pi/\sqrt{M}$ [64], where M denotes the number of function evaluations, and $c \in \mathbb{R}$ depends on the integrand and the particular variable transformation. For $\varphi(t) = \operatorname{erf} t$ the error is $\mathcal{O}\left(\exp\left[-c\sqrt[3]{M^2}\right]\right)$ asymptotically. More details on these and other variable transformations can be found in the review article by Mori [127]. In fact, for all functions $f \in H^p\,(D)$, $1 < p \leq \infty$, the Hardy spaces on the unit disc $D = \{z \in \mathbb{C} \mid |z| < 1\}$, Andersson [3] has shown that the bound on the asymptotic error of any quadrature formula is $\mathcal{O}\left(M^{1-1/(2p)} \exp\left[-c\sqrt{M}\right]\right)$.

The double exponential quadrature formula dates back to the work by Takahasi and Mori [154], who improved on the transformation method by Schwartz. Their integration rule accelerates the convergence of one-dimensional integrals by introducing a suitable variable transformation that results in the double exponential decay of the integrand: $\varphi(t) = \tanh\left(\frac{\pi}{2}\sinh t\right)$. Rather than looking at functions that belong to the Hardy classes $H^p\,(D)$ with $p > 1$, we can focus on the more modest class of integrable functions over $(-1, 1)$, possibly with algebraic or logarithmic singularities at the endpoints ± 1, and a finite number of singularities outside the interval of integration. The asymptotic error of the quadrature formula behaves as $\mathcal{O}\left(\exp\left[-cM/\ln M\right]\right)$; the constant c is related to the location of the singularities of the integrand after the application of the variable transformation. The optimal value of v is

$$v = \frac{2}{M}\ln 2dM,$$

where d is the distance between the real axis and the nearest singularity of the integrand after the variable transformation has been applied; the transformed integrand is thus regular in the strip $|\Im(z)| < d$. In case the original function $f(z)$ with $z \in \mathbb{C}$ has only a singularity at $z = \infty$, we easily compute that $d = \frac{\pi}{2}$. At the optimal step v the nodes in the interval $(-1, 1)$ tend to cluster near the boundaries, especially for small M.

The Takahasi–Mori, or tanh-sinh, formula,

$$\int_{-1}^{1} f(x)\,\mathrm{d}x \approx v\frac{\pi}{2}\sum_{k=-n}^{n} f\left(\tanh\left(\frac{\pi}{2}\sinh kv\right)\right)\frac{\cosh kv}{\cosh^2\left(\frac{\pi}{2}\sinh kv\right)}, \tag{A.16}$$

has been shown to be fast and accurate in high-precision experimental mathematics [9]; in practice we often choose v adaptively. Recently, Borwein and Ye [21] have shown that the Takahasi–Mori quadrature formula converges quadratically for all integrands $f \in H^2(D)$ in the limit of $M \to \infty$.

All these transformed quadrature formulas based on the trapezium rule have exponential decay of the asymptotic error, which basically means that halving the step size roughly doubles the number of correct digits. Note, however, that the quadrature formulas are not exact for polynomials, in contrast to the Gaussian quadrature formulas.

Appendix B

Maple Codes

The package **VarInt** is a library for Maple with which variational integrators can be designed and analysed. The code has been split into its three main components for convenience: VarInt, ExtractAlgorithm and IntegrateSystem. **VarInt** has been created with Maple 11, and it is compatible with all modern Maple releases. An up-to-date version of **VarInt**, including installation instructions, documentation, and an interactive Maple help library with examples, can be obtained from the author or from the Maplesoft Application Centre.

The module VarInt (Code I) consists of the main procedure to compute variational integrators based on built-in quadrature formulas, and a submodule named CreateVarInt (lines 103–148), which provides an interface for user-defined quadrature formulas and interpolation procedures. For separable Lagrangians $L(\boldsymbol{q}, \dot{\boldsymbol{q}}) = T(\dot{\boldsymbol{q}}) - V(\boldsymbol{q})$, where T and V are the (quadratic) kinetic and potential energies, respectively, the module ExtractAlgorithm (Code II) can be used to obtain both explicit and implicit expressions of variational algorithms from the discrete Euler–Lagrange equations; the algorithm extraction module works for both conservative and non-conservative dynamical systems. In addition, it is possible to analyse the variational integrators numerically for one-dimensional dynamical systems with IntegrateSystem (Code III).

Code I: Maple Code for the VarInt module.

```
VarInt:=module()
  export ModuleApply, ExtractAlgorithm, IntegrateSystem, CreateVarInt;
  option package;
  description "Variational Integrator Design with Maple";
  local linkSort, condSolve, computePQ, dEulerLagrange,
        extractWeight, extractParts, trapeziumRule;

  linkSort:=proc(L::list,M::list)
    description "sort a list algebraically in non-descending order "
                "and sort a linked list with its permutation cycle";
    local k,Map;

    Map:=map(attributes, sort([seq])
             (setattribute(evalf(L[k]),[L[k],M[k]]),k=1..nops(L)),`<`):

    return [seq(op(1,op(k,Map)),k=1..nops(Map))],
           [seq(op(2,op(k,Map)),k=1..nops(Map))];
  end proc:

  condSolve:=proc(Eqn::equation,Var::name,Val::posint,Con::posint)
    description "solve an equation algebraically or numerically";

    if Val<=Con then
      return solve(Eqn,Var);
    else
      return fsolve(Eqn,Var,fulldigits);
    end if:
  end proc:

  computePQ:=proc(tList::list,p::symbol,q::symbol,h::symbol,
                  Interp::procedure:=interp)
    description "extrapolate and substitute interval boundaries";
    local k,n,s,t,Q,qList,DQ,DqList,ipQ;

    n:=nops(tList):
    ipQ:=t->subs(s=t,factor(Interp(tList,[seq(q[k],k=1..n)],s))):
    Q:=subs({seq(q[k]=q[k-1],k=1..n),q[n+1]=q[n-1]},
        factor(subs(solve({q[0]=ipQ(0),q[n+1]=ipQ(h)},
                          {q[1],q[n]}),ipQ(t)))):
    DQ:=factor(diff(Q,t)):
    qList:=[seq(simplify(subs(t=tList[k],Q)),k=1..n)]:
    DqList:=[seq(simplify(subs(t=tList[k],DQ)),k=1..n)]:

    qList,DqList;
  end proc:

  dEulerLagrange:=proc(n::posint,dS::anything,dF::list,p::symbol,q::symbol)
    description "calculate the discrete Euler-Lagrange equations";
    local dEL;

    dEL[1]:=p[0]=-convert(diff(dS,q[0]),D)-dF[1]:
    seq(assign(dEL[k+1],0=convert(diff(dS,q[k]),D))+dF[k+1],k=1..n-2):
    dEL[n]:=p[n-1]=convert(diff(dS,q[n-1]),D)+dF[n]:

    return [seq(expand(simplify(dEL[k])),k=1..n)];
  end proc:
```

```
57    trapeziumRule:=proc(n::posint,L::symbol,p::symbol,q::symbol,h::symbol)
58      description "approximate an action with the trapezium rule";
59      local k,tList,qList,DqList,dS;
60
61      tList:=[seq(k*h/(n-1),k=0..n-1)]:
62      qList,DqList:=computePQ(tList,p,q,h):
63      dS:=expand(h/(2*(n-1))*(L(qList[1],DqList[1])+
64                              L(qList[n],DqList[n])+
65                 2*(add(L(qList[k],DqList[k]),k=2..n-1))));
66    end proc:
67
68    extractWeight:=proc(Expr::anything,Fun::symbol,Var::name)
69      description "extract a weight from a quadrature formula";
70      local Term;
71
72      Term:=indets(Expr,And(specfunc(anything,Fun),
73                            patfunc(identical(Var),anything))):
74
75      if Term = {} then
76        return 0;
77      elif nops(Term) = 1 then
78        return coeff(Expr,Term[]);
79      else
80        error "More than one term in the quadrature formula "
81              "matches the variable ",
82              Fun(Var,anything);
83      end if:
84    end proc:
85
86    extractParts:=proc(n::posint,Expr::anything,Fun::symbol,Var::name)
87      description "extract equal-time portions from a quadrature formula";
88      local i,j,Terms;
89
90      Terms:=[seq([indets(Expr,And(specfunc(anything,Fun),
91                                   patfunc(identical(Var[i]),anything)))[]],
92                                   i=0..n-1)]:
93
94      if Terms = [] then
95        return 0;
96      else
97        return [seq(expand(add(coeff(Expr,op(j,op(i,Terms)))*
98                                     op(j,op(i,Terms)),
99                               j=1..nops(op(i,Terms)))),i=1..nops(Terms))];
100     end if:
101   end proc:
102
```

```
103   CreateVarInt:=module()
104     description "create variational integrators based on user-defined "
105                 "nodes, weights and the interpolation function";
106     local a,b,k,n,minL,maxL,nLst,wLst,tList,qList,DqList,
107           dS,dF,ModuleApply;
108
109     ModuleApply:=proc(r::range,nList::list,wList::list,
110                       L::symbol,F::anything,p::symbol,q::symbol,h::symbol,
111                       iProc::procedure:=interp)
112       a:=op(1,r):
113       b:=op(2,r):
114       n:=nops(nList):
115
116       minL:=evalf(min(nList[])):
117       maxL:=evalf(max(nList[])):
118
119       if minL < a and maxL > b then
120         print("Error [VarInt: CreateVarInt]: node list out of bounds");
121         return;
122       elif minL < a then
123         print("Error [VarInt: CreateVarInt]: lowest node ", minL,
124               " lies below the specified lower bound, ", a);
125         return;
126       elif maxL > b then
127         print("Error [VarInt: CreateVarInt]: highest node ", maxL,
128               " exceeds the specified upper bound ", b);
129         return;
130       end if:
131
132       if not n = nops(wList) then
133         print("Error [VarInt: CreateVarInt]: number of elements in "
134               "the node list does not match the number of elements "
135               "in the weight list");
136         return;
137       end if:
138
139       nLst,wLst:=linkSort(nList,wList):
140       tList:=[seq(h/(2*(a-b))*(a+b-2*nLst[k])+h/2,k=1..n)]:
141       qList,DqList:=computePQ(tList,p,q,h,iProc):
142       dS:=h/2*expand(add(wLst[k]*L(qList[k],DqList[k]),k=1..n)):
143       dF:=[seq(expand(h/2*add(wLst[k]*F(qList[k],DqList[k])*
144             factor(diff(qList[k],q[l-1])),k=1..n)),l=1..n)]:
145
146       dEulerLagrange(n,dS,dF,p,q,h);
147     end proc:
148   end module:
149
150   ModuleApply:=proc(n::posint,L::symbol,F::anything,Quad::name,
151                     p::symbol,q::symbol,h::symbol)
152     description "compute a variational integrator based on "
153                 "built-in quadrature rules";
154     local i,j,k,l,m,s,t,x,y,z,nList,tList,wList,qList,DqList,
155           dS,dF,Method,preMult,cF,cG,tempL,rTab,delta,theta;
156
157     Method:={NewtonCotes,Romberg,Chebyshev,GaussLegendre,
158              GaussLobatto,Fejer1,Fejer2,Fejer3,Fejer4,
159              ClenshawCurtis,TakahasiMori};
160
```

```
161       if not Quad in Method then
162         print("Error [VarInt: Quadrature]: quadrature ",Quad," invalid");
163         print("Please select one of the following:");
164         print(NewtonCotes," closed Newton-Cotes quadrature");
165         print(Romberg," Romberg quadrature");
166         print(Chebyshev," Chebyshev quadrature");
167         print(GaussLegendre," Gauss-Legendre quadrature");
168         print(GaussLobatto," Gauss-Lobatto quadrature");
169         print(Fejer1," Fejer (1st kind) quadrature");
170         print(Fejer2," Fejer (2nd kind) quadrature");
171         print(Fejer3," Fejer-Notaris (3rd kind) quadrature");
172         print(Fejer4," Fejer-Notaris (4th kind) quadrature");
173         print(ClenshawCurtis," Clenshaw-Curtis quadrature");
174         print(TakahasiMori," Takahasi-Mori (double exponential) quadrature");
175         return;
176       end if:
177
178       if Quad in Method and n<2 then
179         print("Error [VarInt: Nodes]: number of nodes for ",Quad,
180               " must be at least 2");
181         return;
182       end if:
183
184       if Quad=Romberg and not type(log(n-1)/log(2),nonnegint) then
185         print("Error [VarInt: Nodes]: number of nodes for ",Quad,
186               " must be 2, 3, 5, 9, 17, 33, ... = 1 + 2^N,"
187               " with N an integer");
188         return;
189       end if:
190
191       if Quad=TakahasiMori and ( n<3 or type(n,even) ) then
192         print("Error [VarInt: Nodes]: number of nodes for ",Quad,
193               " must be an odd integer greater than or equal to 3");
194         return;
195       end if:
196
197       if Quad in Method minus {NewtonCotes,Romberg} then
198         if Quad=GaussLegendre then
199           nList:=[condSolve(orthopoly[P](n,t)=0,t,n,4)]:
200           wList:=[seq(2/((1-nList[k]^2)*
201                   subs(s=nList[k],diff(orthopoly[P](n,s),s))^2),k=1..n)]:
202           preMult:=1:
203         elif Quad=GaussLobatto then
204           nList:=[-1,condSolve(diff(orthopoly[P](n-1,t),t)=0,t,n,4),1]:
205           wList:=[2/(n*(n-1)),
206                   seq(2/(n*(n-1)*orthopoly[P](n-1,nList[k])^2),k=2..n-1),
207                   2/(n*(n-1))]:
208           preMult:=1:
209         elif Quad=Fejer1 then
210           theta:=k->(2*k-1)*Pi/(2*n):
211           nList:=[seq(cos(theta(k)),k=1..n)]:
212           wList:=[seq(2/n*(1-2*add(cos(2*l*theta(k))/(4*l^2-1),
213                                    l=1..floor(n/2))),k=1..n)]:
214           preMult:=1:
215         elif Quad=Fejer2 then
216           theta:=k->k*Pi/(n+1):
217           nList:=[seq(cos(theta(k)),k=1..n)]:
218           wList:=[seq(4*sin(theta(k))/(n+1)*
219                   add(sin((2*l-1)*theta(k))/(2*l-1),
220                   l=1..floor(n/2+1/2)),k=1..n)]:
221           preMult:=1:
```

```
222         elif Quad=Fejer3 then
223           theta:=k->(2*k-1)*Pi/(2*n+1):
224           nList:=[seq(cos(theta(k)),k=1..n)]:
225           wList:=[seq(4*sin(theta(k))/(n+1/2)*
226                   add(sin((2*l-1)*theta(k))/(2*l-1),
227                   l=1..floor(n/2+1/2)),k=1..n)]:
228           preMult:=1:
229         elif Quad=Fejer4 then
230           theta:=k->2*k*Pi/(2*n+1):
231           nList:=[seq(cos(theta(k)),k=1..n)]:
232           wList:=[seq(4*sin(theta(k))/(n+1/2)*
233                   add(sin((2*l-1)*theta(k))/(2*l-1),
234                   l=1..floor(n/2+1/2)),k=1..n)]:
235           preMult:=1:
236         elif Quad=ClenshawCurtis then
237           delta:=(i,j)->table(symmetric,identity)[i,j]:
238           theta:=k->k*Pi/(n-1):
239           nList:=[seq(cos(theta(k)),k=0..n-1)]:
240           wList:=[seq((2-delta(0,`mod`(k,n-1)))*
241                   (1-2*(add(cos(2*l*theta(k))*
242                   (1-1/2*delta(l,floor((1/2)*n-1/2)))/(4*l^2-1),
243                   l=1..floor(n/2-1/2))))/(n-1),k=0..n-1)]:
244           preMult:=1:
245         elif Quad=Chebyshev then
246           cF:=(l,z)->convert(series(exp(l/2*(-2+ln(1-z)*(1-1/z)+
247                                   ln(1+z)*(1+1/z))),z=0,l+2),polynom):
248           cG:=(m,x)->factor(subs(y=1/x,x^m*cF(m,y))):
249           nList:=[condSolve(cG(n,t)=0,t,n,3)]:
250           wList:=[seq(1/n,k=1..n)]:
251           preMult:=1:
252         elif Quad=TakahasiMori then
253           m:=n/2-1/2:
254           l:=evalf(2*ln(Pi*n)/n):
255           nList:=[seq(evalf(tanh(Pi/2*sinh(k*l))),k=-m..m)]:
256           wList:=[seq(evalf(Pi/2*cosh(k*l)/cosh(Pi/2*sinh(k*l))^2),k=-m..m)]:
257           preMult:=l:
258         end if:
259
260         nList,wList:=linkSort(nList,wList):
261         tList:=[seq(h/2*nList[k]+h/2,k=1..n)]:
262         qList,DqList:=computePQ(tList,p,q,h):
263         dS:=h/2*expand(add(wList[k]*preMult*L(qList[k],DqList[k]),k=1..n)):
264         dF:=[seq(expand(h/2*add(wList[k]*preMult*F(qList[k],DqList[k])*
265               factor(diff(qList[k],q[l-1])),k=1..n)),l=1..n)]:
266       elif Quad=NewtonCotes then
267         tList:=[seq(k*h/(n-1),k=0..n-1)]:
268         qList,DqList:=computePQ(tList,p,q,h):
269         dS:=expand(int(interp(tList,
270                               [seq(tempL(qList[k],DqList[k]),k=1..n)],t),
271                               t=0..h)):
272         wList:=[seq(extractWeight(dS,tempL,q[k]),k=0..n-1)]:
273         dS:=expand(eval(dS,tempL=L)):
274         dF:=[seq(expand(wList[k]*F(qList[k],DqList[k])),k=1..n)]:
```

```
275       elif Quad=Romberg then
276         tList:=[seq(k*h/(n-1),k=0..n-1)]:
277         qList,DqList:=computePQ(tList,p,q,h):
278
279         m:=log(n-1)/log(2):
280
281         for i from 0 to m do
282           rTab[0,i]:=subs({seq(q[k]=q[k*2^(m-i)],k=1..2^i)},
283                           trapeziumRule(1+2^i,tempL,p,q,h)):
284         end do:
285
286         for j to m do
287           for i from j to m do
288             rTab[j,i]:=(4^j*rTab[j-1,i]-rTab[j-1,i-1])/(4^j-1):
289           end do:
290         end do:
291
292         dS:=expand(eval(rTab[m,m],tempL=L)):
293         dF:=eval(extractParts(n,rTab[m,m],tempL,q),tempL=F):
294       end if:
295
296       return dEulerLagrange(n,dS,dF,p,q,h);
297
298     end proc:
299   end module:
```

CODE II: Maple Code for the ExtractAlgorithm module.

```
1   ExtractAlgorithm:=module()
2     description "extract the one-step map from the discrete "
3                 "Euler-Lagrange equations for separable Lagrangians";
4     local i,k,n,dSum,extAlg,mCurry,extractVar,plugEqn,recSub,ModuleApply;
5
6     mCurry:=proc(p)
7       description "modified curry procedure";
8
9       subs(['_p'=p,'_pX'=_rest],()->_p(_pX,args));
10    end proc:
11
12    extractVar:=proc(Eqn::equation,Var::name,VFun::symbol,FFun::anything)
13      description "extract variable from (implicit) equation";
14      local lhsVFun,lhsFFun,lhsCoeff,tmpEqn;
15
16      tmpEqn:=expand(isolate(Eqn,Var)):
17      lhsVFun:=select(has,lhs(tmpEqn),VFun):
18      if lhsVFun=NULL then
19        lhsVFun:=0:
20      end if:
21      if FFun=0 then
22        lhsFFun:=0:
23      else
24        lhsFFun:=select(has,lhs(tmpEqn),FFun):
25        if lhsFFun=NULL then
26          lhsFFun:=0:
27        end if:
28      end if:
29      lhsCoeff:=frontend(coeff,[lhs(tmpEqn),Var]):
30      return expand(simplify((tmpEqn-lhsVFun-lhsFFun)/lhsCoeff));
31    end proc:
32
```

```
33    plugEqn:=proc(Expr::equation,Eqns)
34      description "substitute an equation into the rhs of an expression";
35
36      lhs(Expr)=expand(simplify(frontend(mCurry(subs,{Eqns}),[rhs(Expr)])));
37    end proc:
38
39    recSub:=proc(Expr::equation,Eqns)
40      description "substitute a sequence of equations recursively";
41      local n;
42
43      n:=nops(Eqns);
44      if n = 1 then
45        return plugEqn(Expr,Eqns[n]);
46      else
47        return recSub(plugEqn(Expr,Eqns[n]),Eqns[1..n-1]);
48      end if:
49    end proc:
50
51    ModuleApply:=proc(dEL::list,p::symbol,q::symbol,V::symbol,F::anything:=0)
52      n:=nops(dEL):
53      extAlg[n]:=expand(add(dEL[k],k=2..n)-dEL[1]+p[0]):
54      extAlg[n-1]:=extractVar(dEL[n-1],q[n-1],V,F):
55
56      if n=3 then
57        extAlg[n-2]:=extractVar(plugEqn(dEL[n-2],extAlg[n-1]),q[n-2],V,F):
58      elif n>=4 then
59        dSum:=[seq(add(dEL[k],k=1..i),i=1..n-2)]:
60
61        for i from n-2 by -1 to 1 do
62          extAlg[i]:=extractVar(recSub(dSum[i],
63                      [seq(extAlg[k],k=i+1..n-1)]),q[i],V,F):
64        end do:
65      end if:
66
67      for i from 2 by 1 to n do
68        extAlg[i]:=recSub(extAlg[i],[seq(extAlg[k],k=1..i-1)]):
69      end do:
70
71      [seq(extAlg[k], k = 1 .. n)];
72    end proc:
73  end module:
```

CODE III: Maple Code for the IntegrateSystem module.

```
IntegrateSystem:=module()
  description "numerically integrate with a variational integrator";
  local pList,qList,tList,eList,stepMap,ModuleApply;

  stepMap:=proc(iPQ::list,nPQ::list,dMap::set)
    local k,n,tempP,tempQ;

    n:=nops(dMap)-1:
    assign(fsolve(subs({tempP[0]=op(1,iPQ),
                        tempQ[0]=op(2,iPQ)},
                       subs({nPQ[1]=tempP,nPQ[2]=tempQ},dMap)),
                      {tempP[n],seq(tempQ[k],k=1..n)},fulldigits));

    return tempP[n],tempQ[n];
  end proc:

  ModuleApply:=proc(DEL::list,namePQ::list,initPQ::list,
                    tSpan::list,tStep::equation,E::procedure)
    local k,N,initE,dEL,dELstep;

    N:=ceil((op(2,tSpan)-op(1,tSpan))/rhs(tStep)):
    pList:=table([0=op(1,initPQ)]):
    qList:=table([0=op(2,initPQ)]):
    initE:=E(op(1,initPQ),op(2,initPQ)):
    eList:=table([0=initE]):
    tList:=[seq(op(1,tSpan)+k*rhs(tStep),k=0..N)]:
    dEL:={seq(DEL[k],k=1..nops(DEL))}:
    dELstep:=subs(tStep,dEL):

    for k to N do
      pList[k],qList[k]:=stepMap([pList[k-1],qList[k-1]],namePQ,dELstep):
      eList[k]:=E(pList[k],qList[k]):
    end do:

    return tList,[seq(pList[k],k=0..N)],
                 [seq(qList[k],k=0..N)],
                 [seq(eList[k],k=0..N)]:
  end proc:
end module:
```

Appendix C

Integrability of the MICZ Problem

The introduction of an additional term that varies with the inverse of the square of the distance destroys the separability of the Hamilton–Jacobi equation for the Kepler–Coulomb problem, as we show below. Since for $\lambda = 0$ in equation (3.5) we have the Kepler–Coulomb problem, which is separable in spherical, parabolic, elliptic and spheroconical coordinates, the MICZ problem can be separable in these four coordinate systems at most.

The answer to the question whether an autonomous, orthogonal Hamiltonian

$$H(\boldsymbol{q}, \boldsymbol{p}) = g^{\alpha}(\boldsymbol{q}) p_{\alpha}^2 + V(\boldsymbol{q}) \tag{C.1}$$

admits a separable solution in a particular natural orthogonal coordinate system is contained in Stäckel's theorem [35, p. 162]:

Theorem 10 (Stäckel). *The Hamilton–Jacobi equation for the Hamiltonian* (C.1) *is separable if and only if there is a matrix* $\mathbf{S}(\boldsymbol{q})$*, the Stäckel matrix, such that the kth row is a function of* q_k *only, and if there is a vector* $\boldsymbol{v}$*, the Stäckel vector, such that the kth element is only a function of* q_k*. The Stäckel matrix and vector satisfy*

$$g^{\alpha}(\boldsymbol{q}) S_{\alpha\beta} = \delta_{1\beta}, \qquad g^{\alpha}(\boldsymbol{q}) v_{\alpha} = V(\boldsymbol{q}). \tag{C.2}$$

Here $\boldsymbol{g}(\boldsymbol{q})$ is the diagonal of the metric tensor, or equivalently, each component is the square of the individual scale factor. Observe that any additional

coordinate-dependent terms to the potential do not alter the Stäckel matrix $\mathbf{S}(\boldsymbol{q})$. Hence, it suffices to check whether there exists a Stäckel vector that, when each of its elements multiplied by the corresponding metric factor in the associated canonical momentum term, and added together, gives back the potential.

C.1 Spherical Coordinates

We have already indicated in Section 3.2.4 that the Hamilton–Jacobi equation for the MICZ problem remains separable in spherical coordinates through the canonical transformation that relates the MICZ problem with the Kepler–Coulomb problem. For completeness we prove the separability in spherical coordinates again, but now with Stäckel's theorem.

The transition from Cartesian (x, y, z) to spherical coordinates (r, θ, ϕ) is achieved by the substitution

$$\begin{cases} x &= r\sin\theta\cos\phi, \\ y &= r\sin\theta\sin\phi, \\ z &= r\cos\theta, \end{cases} \tag{C.3}$$

where the Euclidean distance $\rho = \sqrt{x^2+y^2+z^2} = r$. Here $r > 0$, $0 \leq \theta \leq \pi$, and $0 \leq \phi < 2\pi$. The diagonal of the metric tensor is known to be

$$g = \left(1, \frac{1}{r^2}, \frac{1}{r^2\sin^2\theta}\right), \tag{C.4}$$

and the potential is

$$V(r) = -\frac{k}{r} + \frac{\lambda^2}{2r^2}.$$

Nevertheless, consider a potential that depends only on the distance, that is to say, $V = V(r)$. Hence, the components of the Stäckel vector for θ and ϕ are zero. It is then easy to see that $\boldsymbol{v} = (V(r), 0, 0)$ is the desired Stäckel vector, which concludes the proof of separability of the MICZ problem in spherical coordinates.

C.2 Parabolic Coordinates

Parabolic coordinates (σ, τ, ϕ) are related to Cartesian ones through

$$\begin{cases} x &= \sigma\tau\cos\phi, \\ y &= \sigma\tau\sin\phi, \\ z &= \frac{1}{2}(\sigma^2-\tau^2), \end{cases} \tag{C.5}$$

with $\rho = \frac{1}{2}(\sigma^2 + \tau^2)$, $\sigma \geq 0$, $\tau \geq 0$ and $0 \leq \phi < 2\pi$. The diagonal of the metric tensor reads

$$g = \left(\frac{1}{\sigma^2 + \tau^2}, \frac{1}{\sigma^2 + \tau^2}, \frac{1}{\sigma^2 \tau^2} \right). \tag{C.6}$$

The MICZ potential becomes in parabolic coordinates

$$V(\sigma, \tau) = -\frac{2k}{\sigma^2 + \tau^2} + \frac{2\lambda^2}{(\sigma^2 + \tau^2)^2} \tag{C.7}$$

Now suppose that there is a Stäckel vector $\boldsymbol{v} = (v_1(\sigma), v_2(\tau), 0)$. The potential and the scale factors are all symmetric in σ and τ, so without loss of generality $v_1(\sigma) = f(\sigma)$ and $v_2(\tau) = f(\tau)$ for some yet to be determined function f. Necessarily,

$$\frac{f(\sigma) + f(\tau)}{\sigma^2 + \tau^2} = -\frac{2k}{\sigma^2 + \tau^2} + \frac{2\lambda^2}{(\sigma^2 + \tau^2)^2}. \tag{C.8}$$

Equivalently,

$$(f(\sigma) + f(\tau)) \left(\sigma^2 + \tau^2 \right) = -2k \left(\sigma^2 + \tau^2 \right) + 2\lambda^2, \tag{C.9}$$

from which it follows that

$$f(\sigma) = -f(\tau) - 2k + \frac{2\lambda^2}{\sigma^2 + \tau^2}, \tag{C.10}$$

which for $\lambda \neq 0$ depends on both σ and τ, and hence contradicts our assumption that there exists a Stäckel vector. Please observe that for $\lambda = 0$, in which case the equation for the unknown function f is $f(\sigma) = -f(\tau) - 2k$, the constant function $f(\sigma) = f(\tau) = -k$ satisfies the condition (C.10). We conclude that the MICZ Hamiltonian for non-zero λ is not separable in parabolic coordinates.

Alternatively, observe that in general both $\boldsymbol{g}(\boldsymbol{q})$ and the potential $V(\boldsymbol{q})$ are infinitely differentiable, so that the components of the Stäckel vector must be infinitely differentiable too. Taking derivatives with respect to the coordinates $\boldsymbol{q}$ is therefore well-defined. In particular, take the derivative of equation (C.9) with respect to σ and τ, removing the unknown function f entirely, to arrive at

$$0 = \frac{16\sigma\tau\lambda^2}{(\sigma^2 + \tau^2)^3}. \tag{C.11}$$

This equality must be satisfied for arbitrary σ and τ, which proves that the MICZ Hamiltonian is not separable in parabolic coordinates for $\lambda \neq 0$, as it contradicts our assumption that there exists a Stäckel vector for any value of λ.

C.3 Elliptic Coordinates

Elliptic coordinates (ζ, χ, ϕ) are defined by

$$\begin{cases} x &= \sqrt{(\zeta^2-1)(1-\chi^2)}\cos\phi, \\ y &= \sqrt{(\zeta^2-1)(1-\chi^2)}\sin\phi, \\ z &= \zeta\chi+1, \end{cases} \tag{C.12}$$

with $\rho = \zeta + \chi$. Here we have that $\zeta \geq 1$, $-1 \leq \chi \leq 1$ and $0 \leq \phi < 2\pi$. Now we have that

$$g = \left(\frac{\zeta^2-1}{\zeta^2+\chi^2}, \frac{1-\chi^2}{\zeta^2+\chi^2}, \frac{1}{(\zeta^2-1)(1-\chi^2)} \right). \tag{C.13}$$

The MICZ potential becomes in parabolic coordinates

$$V(\zeta,\chi) = -\frac{k}{\zeta+\chi} + \frac{\lambda^2}{2(\zeta+\chi)^2} \tag{C.14}$$

Again, we assume that there is a Stäckel vector $v = (v_1(\zeta), v_2(\chi), 0)$. By antisymmetry in ζ and χ we can write $v_1(\zeta) = f(\zeta)$ and $v_2(\chi) = -f(\chi)$. The Stäckel condition yields

$$f(\zeta)(\zeta^2-1) - f(\chi)(\chi^2-1) = -\frac{(\zeta-\chi)\left(2k(\zeta+\chi)-\lambda^2\right)}{2(\zeta+\chi)} \tag{C.15}$$

In complete analogy with the proof for parabolic coordinates we can remove the dependency on the unknown function by differentiating with respect to both ζ and χ once, which leads to the expression

$$0 = \frac{(\zeta-\chi)\lambda^2}{(\zeta+\chi)^2}. \tag{C.16}$$

Generally, $\zeta \neq \chi$, so that λ must be equal to zero for the equality to hold. Thus, for $\lambda \neq 0$ there does not exist a Stäckel vector in elliptic coordinates.

C.4 Spheroconical Coordinates

Spheroconical coordinates, or conical coordinates, (r, μ, ν) are given by

$$\begin{cases} x &= \frac{r}{b}\sqrt{(b^2+\mu^2)(b^2-\nu^2)}, \\ y &= \frac{r}{ab}\mu\nu, \\ z &= \frac{r}{a}\sqrt{(a^2-\mu^2)(a^2+\nu^2)}, \end{cases} \tag{C.17}$$

where the parameters a and b satisfy $a^2 + b^2 = 1$. Furthermore, $\mu^2 \leq a^2$ and $\nu^2 \leq b^2$. The Euclidean distance is obviously $\rho = r$. The components of $\boldsymbol{g}$ are

$$g_r = 1, \tag{C.18a}$$

$$g_\mu = \frac{(b^2 + \mu^2)(a^2 - \mu^2)}{r^2(\mu^2 + \nu^2)} \tag{C.18b}$$

$$g_\nu = \frac{(b^2 - \nu^2)(a^2 + \nu^2)}{r^2(\mu^2 + \nu^2)}, \tag{C.18c}$$

respectively. It is then obvious that any potential of the form $V = V(r)$ leads to a separable Hamiltonian in both spherical and spheroconical coordinates.

APPENDIX D

Stumpff Functions

Here we summarize some basic facts about the Stumpff functions, which we have used throughout our discussion. The Stumpff functions are defined for all $z \in \mathbb{C}$ and $n \in \mathbb{N}$ by the series [38, 157]:

$$\mathrm{c}_n(z) = \sum_{k=0}^{\infty} \frac{(-1)^k z^k}{(n+2k)!}. \tag{D.1}$$

Their derivatives with respect to the argument z can be expressed as

$$\frac{\mathrm{d}}{\mathrm{d}z}\mathrm{c}_n(z) = \frac{1}{2}\left(n\,\mathrm{c}_{n+2}(z) - \mathrm{c}_{n+1}(z)\right),$$

or

$$2z\frac{\mathrm{d}}{\mathrm{d}z}\mathrm{c}_{n+1}(z) = \mathrm{c}_n(z) - (n+1)\,\mathrm{c}_{n+1}(z).$$

Stumpff functions can be related to one another through the following recurrence relations:

$$z\,\mathrm{c}_{n+2}(z) = \frac{1}{n!} - \mathrm{c}_n(z),$$
$$z\left((n+1)\,\mathrm{c}_{n+3}(z) - \mathrm{c}_{n+2}(z)\right) = \mathrm{c}_n(z) - (n+1)\,\mathrm{c}_{n+1}(z).$$

Specifically, we have for the first values of n that

$$\mathrm{c}_0\left(z^2\right) = \cos z, \tag{D.2a}$$

$$\mathrm{c}_1\left(z^2\right) = \frac{\sin z}{z}, \tag{D.2b}$$

$$\mathrm{c}_2\left(z^2\right) = \frac{1-\cos z}{z^2}, \tag{D.2c}$$

and

$$\mathrm{c}_3\left(z^2\right) = \frac{z-\sin z}{z^3}. \tag{D.2d}$$

The universal, or associated, Stumpff functions are defined as

$$\mathrm{U}_n\left(z;\alpha\right) = z^n\,\mathrm{c}_n\left(\alpha z^2\right). \tag{D.3}$$

It is now easy to verify that these universal Stumpff functions satisfy the differential equations

$$\frac{\mathrm{d}^2}{\mathrm{d}z^2}\,\mathrm{U}_{n+2}\left(z;\alpha\right) = \frac{z^n}{n!} - \alpha\,\mathrm{U}_{n+2}\left(z;\alpha\right),$$
$$\frac{\mathrm{d}}{\mathrm{d}z}\,\mathrm{U}_{n+1}\left(z;\alpha\right) = \mathrm{U}_n\left(z;\alpha\right),$$

and the recurrence relation

$$\mathrm{U}_n\left(z;\alpha\right) = \frac{z^n}{n!} - \alpha\,\mathrm{U}_{n+2}\left(z;\alpha\right),$$

which follows directly from the definitions and the recurrence relations of the Stumpff functions $\mathrm{c}_n\left(z\right)$. The integral along any rectifiable curve $\gamma \subset \mathbb{C}$, that is, an arc with finite length, is particularly simple:

$$\int_\gamma \mathrm{U}_n\left(z;\alpha\right)\mathrm{d}z = \mathrm{U}_{n+1}\left(z;\alpha\right) + c, \tag{D.4}$$

with $c \in \mathbb{C}$ an integration constant.

Somewhat related to the Stumpff functions is the function

$$\begin{aligned}\mathrm{tg}_1(z^2) &= \frac{c_1(z^2)}{c_0(z^2)} \\ &= \frac{\tan z}{z},\end{aligned} \tag{D.5}$$

which is well-defined for $z \to 0$. The $\mathrm{tg}_{2n+1}(z)$ are related to one another by $\mathrm{tg}_{2n+1}(z) = 1/(2n+1) + z\,\mathrm{tg}_{2n+3}(z)$ for $n \in \mathbb{N}$, from which it follows that

$$\mathrm{tg}_{2n+1}(z) = \begin{cases} \displaystyle\sum_{k=1}^{\infty} \frac{(-1)^{k-1} 2^{2k} (2^{2k}-1) B_{2k} z^{k-1}}{(2k)!}, & n = 0, \\ \displaystyle\sum_{k=1}^{\infty} \frac{(-1)^{k+n-1} 2^{2k} (2^{2k}-1) B_{2k} z^{k-n-1}}{(2k)!} - \sum_{k=1}^{n} \frac{z^{k-n-1}}{(2k-1)}, & n > 1, \end{cases}$$

where B_k denote the Bernoulli numbers. The first term in the Laurent series is the term $z^{-n}\,\mathrm{tg}_1(z)$. The second term can be rewritten in terms of the generalized hypergeometric function ${}_pF_q\left(a_1, \ldots, a_p; b_1, \ldots, b_q; z\right)$ [2, Chap. 15]:

$$\sum_{k=1}^{n} \frac{z^{k-n-1}}{(2k+1)} = z^{n-\frac{1}{2}}\,\mathrm{at}_1(-z) - \frac{{}_2F_1\left(1, \frac{1}{2}+n; \frac{3}{2}+n; z\right)}{2n+1},$$

where ${}_2F_1\left(1, \frac{1}{2}; \frac{3}{2}; z\right) = \mathrm{at}_1(-z)$. Here we have already introduced the first of the collection of functions

$$\begin{aligned} \mathrm{at}_{2n+1}(z) &\equiv \sum_{k=0}^{\infty} \frac{(-1)^{k+n} z^k}{2(k+n)+1} \\ &= \frac{1}{2}(-1)^n \Phi\left(-z, 1, n+\tfrac{1}{2}\right), \end{aligned} \tag{D.6}$$

for $n \in \mathbb{N}$ and $|z| \leq 1$. Here Φ denotes the Lerch transcendent:

$$\Phi(z, s, a) \equiv \sum_{k=0}^{\infty} \frac{z^k}{(a+k)^s},$$

which is valid for $|z| < 1$ and $-a \notin \mathbb{N}$. These functions obey the recurrence relation $\mathrm{at}_{2n+1}(z) = 1/n - z\,\mathrm{at}_{2n+3}(z)$. Of particular interest is the first of these functions, namely

$$\mathrm{at}_1(z^2) = \frac{\arctan z}{z}. \tag{D.7}$$

References

[1] R. Abraham and J.E. Marsden, *Foundations of Mechanics*, AMS Chelsea Publishing, Providence, RI, 2008.

[2] M. Abramowitz and I.A. Stegun (Eds.), *Handbook of Mathematical Functions*, Dover Publications, Mineola, NY, 1972.

[3] J.E. Andersson, *Optimal Quadrature of H^p Functions*, Mathematische Zeitschrift, 172 (1980), pp. 55–62.

[4] J.V. Armitage and W.F. Eberlein, *Elliptic Functions*, Vol. 67 of *London Mathematical Society Student Texts*, Cambridge University Press, Cambridge, UK, 2006.

[5] V.I. Arnold, *Mathematical Methods of Classical Mechanics*, Vol. 60 of *Graduate Texts in Mathematics*, Springer, New York, NY, 1989.

[6] M. Arribas, A. Elipe, and M. Palacios, *Quaternions and the Rotation of a Rigid Body*, Celestial Mechanics and Dynamical Astronomy, 96 (2006), pp. 239–251.

[7] W. van Assche and I. Vanherwegen, *Quadrature Formulas Based on Rational Interpolation*, Mathematics of Computation, 61 (1993), pp. 765–783.

[8] M.A. Austin, P.S. Krishnaprasad, and L.-S. Wang, *Almost Poisson Integration of Rigid Body Systems*, Journal of Computational Physics, 107 (1993), pp. 105–117.

[9] D.H. Bailey, K. Jeyabalan, and X.S. Li, *A Comparison of Three High-Precision Quadrature Schemes*, Experimental Mathematics, 14 (2005), pp. 317–329.

[10] V. Bargmann, *Zur Theorie des Wasserstoffatoms: Bemerkungen zur gleichnamigen Arbeit von V. Fock*, Zeitschrift für Physik, 99 (1936), pp. 576–582.

[11] B.M. Barker and R.F. O'Connell, *The Gravitational Interaction: Spin, Rotation, and Quantum Effects. A Review*, General Relativity and Gravitation, 11 (1979), pp. 149–175.

[12] R.R. Bate, D.D. Mueller, and J.E. White, *Fundamentals of Astrodynamics*, Dover Publications, Mineola, NY, 1971.

[13] L. Bates, *Symmetry Preserving Deformations of the Kepler Problem*, Reports on Mathematical Physics, 26 (1988), pp. 413–428.

[14] E. Belbruno, *Two-Body Motion under the Inverse Square Central Force and Equivalent Geodesic Flows*, Celestial Mechanics and Dynamical Astronomy, 15 (1977), pp. 467–476.

[15] ——, *Capture Dynamics and Chaotic Motions in Celestial Mechanics*, Princeton University Press, Princeton, NJ, 2004.

[16] G. vanden Berghe and M. van Daele, *Trigonometric Polynomial or Exponential Fitting Approach?*, Journal of Computational and Applied Mathematics, 233 (2009), pp. 969–979.

[17] ——, *Symplectic Exponentially-Fitted Four-Stage Runge–Kutta Methods of the Gauss Type*, Numerical Algorithms, (2010).

[18] G. vanden Berghe, M. van Daele, and H. vande Vyver, *Exponentially Fitted Quadrature Rules of the Gauss Type for Oscillatory Integrands*, Applied Numerical Mathematics, 53 (2005), pp. 509–526.

[19] S. Blanes, F. Casas, and J. Ros, *Extrapolation of Symplectic Integrators*, Celestial Mechanics and Dynamical Astronomy, 75 (1999), pp. 149–161.

[20] A. Bloch, P. Gurfil, and K.-Y. Lum, *The Serret–Andoyer Formalism in Rigid-Body Dynamics: II. Geometry, Stabilization, and Control*, Regular and Chaotic Dynamics, 12 (2007), pp. 426–447.

[21] J.M. Borwein and L. Ye, *Quadratic Convergence of the* tanh-sinh *Quadrature Rule*, D-Drive preprint #342 (2006). http://locutus.cs.dal.ca:8088/archive/00000342.

[22] N. Bou-Rabee and J.E. Marsden, *Hamilton–Pontryagin Integrators on Lie Groups Part I: Introduction and Structure-Preserving Properties*, Foundations of Computational Mathematics, 9 (2009), pp. 197–219.

[23] T.J. Bridges and S. Reich, *Numerical Methods for Hamiltonian PDEs*, Journal of Physics A: Mathematical and General, 39 (2006), pp. 5287–5320.

[24] C.J. Budd and M.D. Piggott, *Geometric Integration and its Applications*, Handbook of Numerical Analysis, 11 (2003), pp. 35–139.

[25] R. Bulirsch and J. Stoer, *Numerical Treatment of Ordinary Differential Equations by Extrapolation Methods*, Numerische Mathematik, 8 (1966), pp. 1–13.

[26] J.C. Butcher, *Implicit Runge–Kutta Processes*, Mathematics of Computation, 18 (1964), pp. 50–64.

[27] J.A. Caballero and A. Elipe, *Universal Solution for Motions in a Central Force Field*, Astronomical and Astrophysical Transactions, 19 (2001), pp. 869–874.

[28] A. Cannas da Silva, *Lectures on Symplectic Geometry*, Lecture Notes in Mathematics, Springer, Berlin, Heidelberg, New York, 2001.

[29] H. Cendra, J.E. Marsden, S. Pekarsky, and T.S. Ratiu, *Variational Principles for Lie–Poisson and Hamilton–Poincaré Equations*, Moscow Mathematical Journal, 3 (2003), pp. 833–867.

[30] J.E. Chambers, *Symplectic Integrators with Complex Time Steps*, The Astronomical Journal, 126 (2003), pp. 1119–1126.

[31] D.E. Chang and J.E. Marsden, *Geometric Derivation of the Delaunay Variables and Geometric Phases*, Celestial Mechanics and Dynamical Astronomy, 86 (2003), pp. 185–208.

[32] P.J. Channell and J.C. Scovel, *Integrators for Lie–Poisson Dynamical Systems*, Physica D, 50 (1991), pp. 80–88.

[33] M. Chyba, E. Hairer, and G. Vilmart, *The Role of Symplectic Integrators in Optimal Control*, Optimal Control Applications and Methods, 30 (2009), pp. 367–382.

[34] C.W. Clenshaw and A.R. Curtis, *A Method for Numerical Integration on an Automatic Computer*, Numerische Mathematik, 2 (1960), pp. 197–205.

[35] B. Cordani, *The Kepler Problem. Group Theoretical Aspects, Regularization and Quantization, with Application to the Study of Perturbations*, Vol. 29 of *Progress in Mathematical Physics*, Birkhäuser, Basel, 2003.

[36] J.L. Crassidis, F.L. Markley, and Y. Cheng, *Survey of Nonlinear Attitude Estimation Methods*, Journal of Guidance, Control, and Dynamics, 30 (2007), pp. 12–28.

[37] R.H. Cushman and L.M. Bates, *Global Aspects of Classical Integrable Systems*, Birkhäuser, Basel, 2004.

[38] J.M.A. Danby, *Fundamentals of Celestial Mechanics*, Willmann–Bell, Richmond, VA, 1988.

[39] E. Deotto, E. Gozzi, and D. Mauro, *Hilbert Space Structure in Classical Mechanics I*, Journal of Mathematical Physics, 44 (2003), pp. 5902–5936.

[40] ——, *Hilbert Space Structure in Classical Mechanics II*, Journal of Mathematical Physics, 44 (2003), pp. 5937–5957.

[41] A. Deprit, *The Eliminations of Parallax in Satellite Theory*, Celestial Mechanics and Dynamical Astronomy, 24 (1981), pp. 111–153.

[42] R.S. Dunning, The Orbital Mechanics of Flight Mechanics, Technical Report NASA SP-325, National Aeronautics and Space Administration, 1973.

[43] H. Duru and H. Kleinert, *Solution of the Path Integral for the H-Atom*, Physics Letters B, 84 (1979), pp. 185–188.

[44] A. Elipe and V. Lanchares, *Exact Solution of a Triaxial Gyrostat with One Rotor*, Celestial Mechanics and Dynamical Astronomy, 101 (2008), pp. 49–68.

[45] W.M. Farr, *Variational Integrators for Almost-Integrable Systems*, Celestial Mechanics and Dynamical Astronomy, 103 (2009), pp. 105–118.

[46] W.M. Farr and E. Bertschinger, *Variational Integrators for the Gravitational N-Body Problem*, The Astrophysical Journal, 663 (2007), pp. 1420–1433.

[47] L. Fejér, *Mechanische Quadraturen mit positiven Cotesschen Zahlen*, Mathematische Zeitschrift, 37 (1933), pp. 287–309.

[48] V.A. Fock, *Zur Theorie des Wasserstoffatoms*, Zeitschrift für Physik, 98 (1935), pp. 145–154.

[49] P. Fortescue, J. Stark, and G. Swinerd (Eds.), *Spacecraft Systems Engineering*, John Wiley and Sons, Chichester, UK, 2007.

[50] T. Frankel, *Geometry of Physics. An Introduction*, Cambridge University Press, Cambridge, UK, 2003.

[51] W. Gander, *Generating Numerical Algorithms Using a Computer Algebra System*, BIT Numerical Mathematics, 46 (2006), pp. 491–504.

[52] W. Gander and D. Gruntz, *Derivation of Numerical Methods Using Computer Algebra*, SIAM Review, 41 (1999), pp. 577–593.

[53] W. Gautschi, *The Use of Rational Functions in Numerical Quadrature*, Journal of Computational and Applied Mathematics, 133 (2001), pp. 111–126.

[54] ——, *Orthogonal Polynomials and Special Functions. Computation and Applications*, Vol. 1883 of *Lecture Notes in Mathematics*, Springer, Berlin, Heidelberg, 2006.

[55] Z. Ge and J.E. Marsden, *Lie–Poisson Hamilton–Jacobi Theory and Lie–Poisson Integrators*, Physics Letters A, 133 (1988), pp. 134–139.

[56] M. Göckeler and T. Schücker, *Differential Geometry, Gauge Theories, and Gravity*, Cambridge Monographs on Mathematical Physics, Cambridge University Press, Cambridge, UK, 1999.

[57] E. Gozzi, *Hidden BRS Invariance in Classical Mechanics I*, Physics Letters B, 201 (1988), pp. 525–528.

[58] ——, *Functional Techniques in Classical Mechanics*, Nuclear Physics B - Proceedings Supplements, 104 (2002), pp. 243–246.

[59] E. Gozzi, M. Reuter, and W.D. Thacker, *Hidden BRS Invariance in Classical Mechanics II*, Physical Review D, 40 (1989), pp. 3363–3377.

[60] W. Gragg, *On Extrapolation Algorithms for Ordinary Initial Value Problems*, SIAM Journal on Numerical Analysis, 2 (1965), pp. 384–403.

[61] A. Griewank and A. Walther, *Evaluating Derivatives. Principles and Techniques of Algorithmic Differentiation*, Vol. 105 of *Applied Mathematics*, Society for Industrial and Applied Mathematics, Philadelphia, PA, 2008.

[62] V. Guillemin and S. Sternberg, *Variations on a Theme by Kepler*, Vol. 42 of *Colloquium Publications*, American Mathematical Society, Providence, RI, 2006.

[63] P. Gurfil, A. Elipe, W. Tangren, and M. Efroimsky, *The Serret–Andoyer Formalism in Rigid-Body Dynamics: I. Symmetries and Perturbations*, Regular and Chaotic Dynamics, 12 (2007), pp. 389–425.

[64] S. Haber, *The* tanh *Rule for Numerical Integration*, SIAM Journal on Numerical Analysis, 14 (1977), pp. 668–685.

[65] E. Hairer and D. Stoffer, *Reversible Long-Term Integration with Variable Step Sizes*, SIAM Journal on Scientific Computing, 18 (1997), pp. 257–269.

[66] E. Hairer, C. Lubich, and G. Wanner, *Geometric Numerical Integration. Structure-Preserving Algorithms for Ordinary Differential Equations*, Vol. 31 of *Springer Series in Computational Mathematics*, Springer, Berlin, Heidelberg, New York, 2006.

[67] S. Harfst, A. Gualandris, D. Merritt, and S. Mikkola, *A Hybrid N-Body Code Incorporating Algorithmic Regularization and Post-Newtonian Forces*, Monthly Notices of the Royal Astronomical Society, 389 (2008), pp. 2–12.

[68] G.J. Heckman and P. Hochs, *Geometry of the Momentum Map*, Lecture Notes MRI Spring School, University of Utrecht, the Netherlands, 2004.

[69] J. Hietarinta, *Discrete Integrable Dynamics*, Lecture Notes UFYS3015, University of Turku, Finland, 2010.

[70] F.B. Hildebrand, *Introduction to Numerical Analysis*, Dover Publications, Mineola, NY, 1987.

[71] G.R. Hintz, *Survey of Orbit Element Sets*, Journal of Guidance, Control, and Dynamics, 31 (2008), pp. 785–790.

[72] T. Holder, B. Leimkuhler, and S. Reich, *Explicit Variable Step-Size and Time-Reversible Integration*, Applied Numerical Mathematics, 39 (2001), pp. 367–377.

[73] D.D. Holm and J.E. Marsden, *The Rotor and the Pendulum*, In: P. Donato, C. Duval, J. Elhadad, and G.M. Tuynman (Eds.), *Symplectic Geometry and Mathematical Physics*, Progress in Mathematics, pp. 189–203, Birkhäuser, Boston, MA, 1991.

[74] D.D. Holm, T. Schmah, and C. Stoica, *Geometric Mechanics and Symmetry. From Finite to Infinite Dimensions*, Vol. 12 of *Oxford Texts in Applied and Engineering Mathematics*, Oxford University Press, Oxford, UK, 2009.

[75] C.J. Isham, *Modern Differential Geometry for Physicists*, Vol. 61 of *World Scientific Lecture Notes in Physics*, World Scientific, Singapore, 1999.

[76] L.G. Ixaru, *Operations on Oscillatory Functions*, Computer Physics Communications, 105 (1997), pp. 1–19.

[77] L.G. Ixaru and B. Paternoster, *A Gauss Quadrature Rule for Oscillatory Integrands*, Computer Physics Communications, 133 (2001), pp. 177–188.

[78] Z. Jia and B. Leimkuhler, *Geometric Integrators for Multiple Time-Scale Simulation*, Journal of Physics A: Mathematical and General, 39 (2006), pp. 5379–5404.

[79] W. Johnson, *Comments and Commentary on William Moore's* A Treatise on the Motion of Rockets and an Essay on Naval Gunnery, International Journal of Impact Engineering, 16 (1995), pp. 499–521.

[80] A. Jorba and M. Zou, *A Software Package for the Numerical Integration of ODEs by means of High-Order Taylor Methods*, 2004. http://www.maia.ub.es/~angel/taylor.

[81] J.V. José and E.J. Saletan, *Classical Dynamics. A Contemporary Approach*, Cambridge University Press, New York, NY, 2002.

[82] O. Junge, J.E. Marsden, and S. Ober-Blöbaum, *Discrete Mechanics and Optimal Control*, In: *Proceedings of the 16th IFAC Conference on Decision and Control*, 2005.

[83] C. Kane, J.E. Marsden, M. Ortiz, and M. West, *Variational Integrators and the Newmark Algorithm for Conservative and Dissipative Mechanical Systems*, International Journal for Numerical Methods in Engineering, 49 (2000), pp. 1295–1325.

[84] B. Karasözen, *Poisson Integrators*, Mathematical and Computer Modelling, 40 (2004), pp. 1225–1244.

[85] W.M. Kaula, *Theory of Satellite Geodesy. Applications of Satellites to Geodesy*, Dover Publications, Mineola, NY, 2000.

[86] L. Kharevych, Y. Weiwei, Y. Tong, E. Kanso, J.E. Marsden, P. Schröder, and M. Desbrun, *Geometric, Variational Integrators for Computer Animation*, In: D. Fellner and S. Spencer (Eds.), *Proceedings of the 2006 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pp. 43–51, Eurographics Association, Aire-la-Ville, Switzerland, 2006.

[87] L.E. Kidder, *Coalescing Binary Systems of Compact Objects to (Post)$\frac{5}{2}$-Newtonian Order. V. Spin Effects*, Physical Review D, 52 (1995), pp. 821–847.

[88] D.E. Kirk, *Optimal Control Theory. An Introduction*, Dover Publications, Mineola, NY, 2004.

[89] W.S. Koon, M.W. Lo, J.E. Marsden, and S.D. Ross, *Dynamical Systems, the Three-Body Problem, and Space Mission Design*, Marsden Books, 2008.

[90] B.O. Koopman, *Hamiltonian Systems and Transformation in Hilbert Space*, Proceedings of the National Academy of Sciences, 17 (1931), pp. 315–318.

[91] M. Kummer, *On the Regularization of the Kepler Problem*, Communications in Mathematical Physics, 84 (1982), pp. 133–152.

[92] P. Kustaanheimo and E. Stiefel, *Perturbation Theory of Kepler Motion Based on Spinor Regularization*, Journal für die reine und angewandte Mathematik, 218 (1965), pp. 204–219.

[93] T. Lee, M. Leok, and N.H. McClamroch, *Lie Group Variational Integrators for the Full Body Problem in Orbital Mechanics*, Celestial Mechanics and Dynamical Astronomy, 98 (2007), pp. 121–144.

[94] B. Leimkuhler, *Reversible Adaptive Regularization: Perturbed Kepler Motion and Classical Atomic Trajectories*, Philosophical Transactions: Mathematical, Physical and Engineering Sciences, 357 (1999), pp. 1101–1133.

[95] ——, *An Efficient Multiple Time-Scale Reversible Integrator for the Gravitational N-Body Problem*, Applied Numerical Mathematics, 43 (2002), pp. 175–190.

[96] B. Leimkuhler and S. Reich, *Simulating Hamiltonian Dynamics*, Vol. 14 of *Cambridge Monographs on Applied and Computational Mathematics*, Cambridge University Press, Cambridge, UK, 2005.

[97] M. Leok and T. Shingel, *Prolongation-Collocation Variational Integrators*, arXiv:1101.1995 (2011). http://arxiv.org/abs/1101.1995.

[98] A. Lew, J.E. Marsden, M. Ortiz, and M. West, *Asynchronous Variational Integrators*, Archive for Rational Mechanics and Analysis, 167 (2003), pp. 85–146.

[99] ——, *Variational Time Integrators*, International Journal for Numerical Methods in Engineering, 60 (2004), pp. 153–212.

[100] S. Li and M. Qin, *Lie–Poisson Integration for Rigid Body Dynamics*, Computers and Mathematics with Applications, 30 (1995), pp. 105–118.

[101] F.J. Lin and J.E. Marsden, *Symplectic Reduction and Topology for Applications in Classical Molecular Dynamics*, Journal of Mathematical Physics, 33 (1992), pp. 1281–1294.

[102] R.G. Littlejohn and M. Reinsch, *Internal or Shape Coordinates in the N-Body Problem*, Physical Review A, 52 (1995), pp. 2035–2051.

[103] G.G. Lorentz, K. Jetter, and S.D. Riemenschneider, *Birkhoff Interpolation*, Vol. 19 of *Encyclopedia of Mathematics and Its Applications*, Cambridge University Press, Cambridge, UK, 1984.

[104] Z. Ma and C.W. Rowley, *Lie–Poisson Integrators: a Hamiltonian, Variational Approach*, International Journal for Numerical Methods in Engineering, 82 (2010), pp. 1609–1644.

[105] G. Marmo, E.J. Saletan, A. Simoni, and B. Vitale, *Dynamical Systems. A Differential Geometric Approach to Symmetry and Reduction*, John Wiley and Sons, Chichester, UK, 1985.

[106] J.E. Marsden and T.S. Ratiu, *Introduction to Mechanics and Symmetry. A Basic Exposition of Classical Mechanical Systems*, Vol. 17 of *Texts in Applied Mathematics*, Springer, New York, NY, 2002.

[107] J.E. Marsden and A. Weinstein, *Reduction of Symplectic Manifolds with Symmetry*, Reports on Mathematical Physics, 5 (1974), pp. 121–130.

[108] J.E. Marsden and M. West, *Discrete Mechanics and Variational Integrators*, Acta Numerica, 10 (2001), pp. 357–514.

[109] J.E. Marsden, G. Misiołek, J.-P. Ortega, M. Perlmutter, and T.S. Ratiu, *Hamiltonian Reduction by Stages*, Vol. 1913 of *Lecture Notes in Mathematics*, AMS Chelsea Publishing, Berlin, Heidelberg, New York, 2007.

[110] D. ó Mathúna, *Integrable Systems in Celestial Mechanics*, Vol. 51 of *Progress in Mathematical Physics*, Birkhäuser, Boston, MA, 2008.

[111] C. McCord, K.R. Meyer, and D. Offin, *Are Hamiltonian Flows Geodesic Flows?*, Transactions of the American Mathematical Society, 355 (2003), pp. 1237–1250.

[112] D. McDuff and D. Salamon, *Introduction to Symplectic Topology*, Vol. 22 of *Oxford Mathematical Monographs*, Oxford University Press, Oxford, UK, 1999.

[113] H.V. McIntosh and A. Cisneros, *Degeneracy in the Presence of a Magnetic Monopole*, Journal of Mathematical Physics, 11 (1970), pp. 896–916.

[114] R.I. McLachlan, *Explicit Lie–Poisson Integration and the Euler Equations*, Physical Review Letters, 71 (1993), pp. 3043–3046.

[115] R.I. McLachlan and G.R.W. Quispel, *Geometric Integrators for ODEs*, Journal of Physics A: Mathematical and General, 39 (2006), pp. 5251–5285.

[116] J.S. Meditch, *On the Problem of Optimal Thrust for a Lunar Soft Landing*, IEEE Transactions on Automatic Control, 9 (1964), pp. 477–484.

[117] K.R. Meyer, *Symmetries and Integrals in Mechanics*, In: M.M. Peixoto (Ed.), *Proceedings of the 1971 Symposium on Dynamical Systems*, pp. 259–272, Academic Press, New York, NY, 1973.

[118] S. Mikkola, *Efficient Symplectic Integration of Satellite orbits*, Celestial Mechanics and Dynamical Astronomy, 74 (1999), pp. 278–285.

[119] S. Mikkola and S. Aarseth, *A Time-Transformed Leapfrog Scheme*, Celestial Mechanics and Dynamical Astronomy, 84 (2002), pp. 343–354.

[120] S. Mikkola and D. Merritt, *Algorithmic Regularization with Velocity-Dependent Forces*, Monthly Notices of the Royal Astronomical Society, 372 (2006), pp. 219–223.

[121] ——, *Implementing Few-Body Algorithmic Regularization with Post-Newtonian Terms*, The Astronomical Journal, 135 (2008), pp. 2398–2405.

[122] S. Mikkola and K. Tanikawa, *Explicit Symplectic Algorithms for Time-Transformed Hamiltonians*, Celestial Mechanics and Dynamical Astronomy, 74 (1999), pp. 287–295.

[123] ——, *Algorithmic Regularization of the Few-Body Problem*, Monthly Notices of the Royal Astronomical Society, 310 (1999), pp. 745–749.

[124] C.W. Misner, K.S. Thorne, and J.A. Wheeler, *Gravitation*, W.H. Freeman, New York, NY, 1973.

[125] G. Mititelu, *Analytical Solutions for the Equations of Motion of a Space Vehicle During the Atmospheric Re-Entry Phase on a 2-D Trajectory*, Celestial Mechanics and Dynamical Astronomy, 103 (2009), pp. 327–342.

[126] B.E. Moore and S. Reich, *Multi-Symplectic Integration Methods for Hamiltonian PDEs*, Future Generation Computer Systems, 19 (2003), pp. 395–402.

[127] M. Mori, *Quadrature Formulas Obtained by Variable Transformation and the DE-Rule*, Journal of Computational and Applied Mathematics, 12-13 (1985), pp. 119–130.

[128] J.K. Moser, *Regularization of Kepler's Problem and the Averaging Method on a Manifold*, Communications on Pure and Applied Mathematics, 23 (1970), pp. 609–636.

[129] M. Nakahara, *Geometry, Topology and Physics*, Graduate Student Series in Physics, Institute of Physics Publishing, Bristol, UK, 2000.

[130] Y. Nambu, *Generalized Hamiltonian Dynamics*, Physical Review D, 7 (1973), pp. 2405–2412.

[131] J. von Neumann, *Zur Operatorenmethode in der Klassischen Mechanik*, Annals of Mathematics, 33 (1932), pp. 587–642.

[132] NIMA, Department of Defense World Geodetic System 1984. Its Definition and Relationships with Local Geodetic Systems, Technical Report TR8350.2, National Imagery and Mapping Agency, 2004.

[133] S.E. Notaris, *Interpolatory Quadrature Formulae with Chebyshev Abscissae of the Third or Fourth Kind*, Journal of Computational and Applied Mathematics, 81 (1997), pp. 83–99.

[134] S. Ober-Blöbaum, *Discrete Mechanics and Optimal Control*, PhD thesis, University of Paderborn, Paderborn, Germany, 2008.

[135] I.P. Omelyan, I.M. Mryglod, and R. Folk, *Construction of High-Order Force-Gradient Algorithms for Integration of Motion in Classical and Quantum Systems*, Physical Review E, 66 (2002), pp. 026701-1–026701-21.

[136] P.L. Palmer, S. Mikkola, and Y. Hashida, *A Simple High Accuracy Integrator for Spacecraft Attitude Systems*, AIAA Guidance, Navigation, and Control Conference and Exhibit, (2004).

[137] P. Pástor, J. Klačka, and L. Kómar, *Motion of Dust in Mean Motion Resonances with Planets*, Celestial Mechanics and Dynamical Astronomy, 103 (2009), pp. 343–364.

[138] G.W. Patrick and C. Cuell, *Error Analysis of Variational Integrators of Unconstrained Lagrangian Systems*, Numerische Mathematik, 113 (2009), pp. 243–264.

[139] N.K. Pavlis, S.A. Holmes, S.C. Kenyon, and J.K. Factor, *An Earth Gravitational Model to Degree 2160: EGM2008*, presented at the 2008 General Assembly of the European Geosciences Union, Vienna, Austria, April 13–18, 2008.

[140] M. Preto and S. Tremaine, *A Class of Symplectic Integrators with Adaptive Time Step for Separable Hamiltonian Systems*, The Astronomical Journal, 118 (1999), pp. 2532–2541.

[141] M. Puta, *On the Dynamics of the Rigid Body with a Single Rotor and an Internal Torque*, Reports on Mathematical Physics, 32 (1993), pp. 343–349.

[142] S. Reich, Numerical Integration of the Generalized Euler Equation, Technical Report TR 93-20, University of British Columbia, 1993.

[143] C. Richardson and J. Gaunt, *The Deferred Approach to the Limit*, Transactions of the Royal Society of London, 226 (1927), pp. 300–361.

[144] W. Rindler, *Relativity – Special, General and Cosmological*, Oxford University Press, Oxford, UK, 2001.

[145] C. Schwartz, *Numerical Integration of Analytic Functions*, Journal of Computational Physics, 4 (1969), pp. 19–29.

[146] M.H. Soffel, *Relativity in Astrometry. Celestial Mechanics and Geodesy*, Springer, Berlin, Heidelberg, 1989.

[147] J.-M. Souriau, *Géométrie de l'Espace des Phases, Calcul des Variations et Mécanique Quantique*, Faculté des Sciences de Marseille, France, 1965.

[148] R.F. Stengel, *Optimal Control and Estimation*, Dover Publications, Mineola, NY, 1994.

[149] J. Stoer and R. Bulirsch, *Introduction to Numerical Analysis*, Vol. 12 of *Texts in Applied Mathematics*, Springer, New York, NY, 2002.

[150] J. Struckmeier, *Hamiltonian Dynamics on the Symplectic Extended Phase Space for Autonomous and Non-Autonomous Systems*, Journal of Physics A: Mathematical and General, 38 (2005), pp. 1257–1278.

[151] ——, *Extended Hamilton–Lagrange Formalism and Its Application to Feynman's Path Integral for Relativistic Quantum Physics*, International Journal of Modern Physics E, 18 (2009), pp. 79–108.

[152] M. Suzuki, *Fractal Decomposition of Exponential Operators with Applications to Many-Body Theories and Monte Carlo Simulations*, Physics Letters A, 146 (1990), pp. 319–323.

[153] H. Takahasi and M. Mori, *Error Estimation in the Numerical Integration of Analytic Functions*, Report of the Computer Centre, University of Tokyo, 3 (1970), pp. 41–108.

[154] ——, *Double Exponential Formulas for Numerical Integration*, Publications of the Research Institute for Mathematical Sciences, 9 (1974), pp. 721–741.

[155] J. Touma and J. Wisdom, *Lie–Poisson Integrators for Rigid Body Dynamics in the Solar System*, The Astronomical Journal, 107 (1994), pp. 1189–1202.

[156] USGS, *Global 30 Arc Second Elevation Data Set*, Data Available from the United States Geological Survey, 1996.

[157] D.A. Vallado, *Fundamentals of Astrodynamics and Applications*, Vol. 21 of *Space Technology Library*, Microcosm, Hawthorne, CA, 2007.

[158] J.M. Valpiani and P.L. Palmer, *Nonlinear Geometric Estimation for Satellite Attitude*, Journal of Guidance, Control, and Dynamics, 31 (2008), pp. 835–848.

[159] J.P. Vinti, *New Method of Solution for Unretarded Satellite Orbits*, Journal of Research of the National Bureau of Standards, 63B (1959), pp. 105–116.

[160] ——, *Theory of an Accurate Intermediary Orbit for Satellite Astronomy*, Journal of Research of the National Bureau of Standards, 65B (1961), pp. 169–201.

[161] ——, *Zonal Harmonic Perturbations of an Accurate Reference Orbit of an Artificial Satellite*, Journal of Research of the National Bureau of Standards, 67B (1963), pp. 191–222.

[162] ——, *Invariant Properties of the Spheroidal Potential of an Oblate Planet*, Journal of Research of the National Bureau of Standards, 70B (1966), pp. 1–16.

[163] ——, *Inclusion of the Third Zonal Harmonic in an Accurate Reference Orbit of an Artificial Satellite*, Journal of Research of the National Bureau of Standards, 70B (1966), pp. 17–46.

[164] C.M. Will, *Testing the General Relativistic "No-Hair" Theorems Using the Galactic Center Black Hole Sagittarius A**, Astrophysical Journal Letters, 674 (2008), pp. L25–L28.

[165] H. Yoshida, *Construction of Higher Order Symplectic Integrators*, Physics Letters A, 150 (1990), pp. 262–268.

[166] H. Yoshimura and J.E. Marsden, *Dirac Structures in Lagrangian Mechanics Part I: Implicit Lagrangian Systems*, Journal of Geometry and Physics, 57 (2006), pp. 133–156.

[167] ——, *Dirac Structures in Lagrangian Mechanics Part II: Variational Structures*, Journal of Geometry and Physics, 57 (2006), pp. 209–250.

[168] D. Zwanziger, *Exactly Soluble Nonrelativistic Model of Particles with Both Electric and Magnetic Charges*, Physical Review, 176 (1968), pp. 1480–1488.