DIPLOMA THESIS

Isotropy in geometric integration

Author: *Håvard Berland*

Spring 2002

Norwegian University of Science and Technology
Department of Mathematical Sciences

NTNU
Norwegian University of
Science and Technology

Faculty of Information Technology,
Mathematics and Electrical Engineering

DIPLOMA THESIS

FOR

STUD.TECHN. Håvard Berland

FACULTY OF INFORMATION TECHNOLOGY, MATHEMATICS AND ELECTRICAL ENGINEERING

NTNU

Date due: June 14, 2002

Discipline: Numerics

Title: "Isotropy in geometric integration"

Purpose of the work: Make a numerical study of the role of isotropy in Lie group methods and find ways to improve the quality of numerical solvers by using isotropy.

This diploma thesis is to be carried out at the Department of Mathematical Sciences under guidance by Professor Brynjulf Owren.

Trondheim, January 14, 2002.

Trond Digernes
Instituttleder
Dept. of Mathematical Sciences

Brynjulf Owren
Professor
Dept. of Mathematical Sciences

Preface

The ideas of this diploma thesis emerged from a recent paper on isotropy in geometric integration by Lewis and Olver [15] where they used isotropy to significantly improve the numerical solution of Lie group methods. I was motivated for a subject that included both numerical analysis and numerical implementation of Lie group methods.

The work started by various implementation attempts of the isotropy correction for rigid body, and also some literature studies of Hamiltonian theory and also the generalization to Poisson systems. An early goal was to extend the isotropy correction on the sphere to Stiefel manifolds. For various reasons, interest turned to utilization of isotropy for $SL(2)$ actions, as it to our knowledge never has been used before for an $\mathbf{R}^2$-solver, and as it might be simple enough to facilitate analysis.

The thesis ended by a numerical investigation of the performance of the rigid body correction and the $SL(2)$-correction, together with a successful tweak discovered by luck for the $SL(2)$-action. Deeper analysis explaining the tweak was sought, but it has not been found yet.

I wish to thank my supervisor, Brynjulf Owren, for continuous support and supervision.

Trondheim, June 2002

Håvard Berland

Abstract

Lie group methods are formulated by means of a Lie group action on a manifold. If the dimension of the Lie group is greater than the dimension of the manifold, there is a certain freedom in the formulation of the method. We focus on Runge-Kutta-Munthe-Kaas Lie group methods. Background from differential topology, Lie and matrix groups is provided and from there a presentation of Runge-Kutta-Munthe-Kaas which methods are given.

Lewis and Olver [15] has recently shown how to improve the accuracy for algorithms on the sphere by means of an $SO(3)$-action. We elaborate and extend their approach by using Lie series, and find an equation that can be used to determine a choice of the isotropy freedom leading to better numerical behavior.

The same methodology is then applied to an $SL(2)$-based Lie group method on $\mathbf{R}^2$. We use the Lotka-Volterra system and the Duffing oscillator as examples and obtain excellent long time behaviour comparable to Symplectic Euler by a careful choice of isotropy.

Contents

Chapter 1

Introduction

We consider the solution of differential equations on manifolds,

$$\dot{y}(t) = F(y(t)), \qquad y(t) \in M,\ F(y(t)) \in T_{y(t)}M \ \text{ for all } t \in \mathbf{R} \tag{1.1}$$

to which Lie group methods are applied. If the dimension of the Lie group is greater than the dimension of the solution manifold M, the Lie group possesses an isotropy subgroup. The formulation of the Lie group method (we consider Runge-Kutta-Munthe-Kaas methods) is not uniquely determined by Equation (1.1), and our goal for this thesis is to see what can be done regarding the choice of isotropy.

We do not opt for numerical methods with the highest possible local order. Because of this, emphasis is restricted to the Lie group version of the classical Forward Euler algorithm. Focus is on global properties, and we will use examples from Poisson systems which has invariants we know a priori should be conserved. Our numerical methods should be able to preserve these invariants when integrating over long time intervals. Nevertheless, we use local order theory to construct the isotropy correction for Lie-Euler, which we will has greater impact on global behavior than local.

First necessary background in differential topology and Lie groups is given. The matrix groups that are going to be used in the following chapters, The special orthogonal group $SO(n)$ and The special linear group $SL(n)$ are given special attention regarding relevant results and proofs. Readers already familiar with Lie group methods and the noted matrix groups may skip Chapter 2 and 3.

Doing a straightforward Taylor expansion of the local error by Lie series, we find a second order condition for the isotropy. In general, this condition can not be fulfilled by the isotropy, so we resort to a minimalization of the local order in the numerical implementations.

Lewis and Olver [15] have written a paper on how to use the isotropy for rigid body dynamics. This seems to be the only paper available to the current date discussing isotropy corrections. By local error analysis in a basis especially suited for the $SO(3)$ action on S^2, they develop a corrected version of Lie-Euler with stability properties of the Hamiltonian superior to the uncorrected Lie-Euler. In Chapter 6 we rephrase their analysis, and show that their result is equivalent to our more general proposition on how to use the isotropy.

Chapter 7 contains a new application of the $SL(2)$ action to differential equations in $\mathbf{R}^2$. Using $SL(2)$ seems uninteresting if isotropy is not paid attention to, because of the additional computational overhead inherent in a Lie group method compared to a classical solver. A straightforward construction of a Lie-Euler method performs roughly the same as

the standard (and bad-performing) Euler method on the well know Lotka-Volterra system and on the Duffing oscillator. It becomes interesting when the isotropy corrected Lie-Euler with the same strategy for isotropy correction as for the rigid body problem, performs in the league of Symplectic Euler, which has superb proven stability properties for the Lotka-Volterra system, as it is a Poisson map and is symplectic for the Hamiltonian Duffing oscillator. The downside of the best correction we construct, is a dependency on a constant which has to be determined by trial and error. Still to be done is analysis explaining the role of this constant, and if it is possible to choose it a priori.

Chapter 2

Manifolds and Lie groups

A manifold is a generalization of spaces, to spaces more difficult to grasp in the human mind.

We are in this thesis going to solve ordinary differential equations for which the solution evolves on a manifold. Our main tool for this will be the use of Lie groups. For this, we will present some general theory on manifolds and Lie groups.

The presentation given will emphasize on notation and vital results for our applications. Deep proofs are skipped.

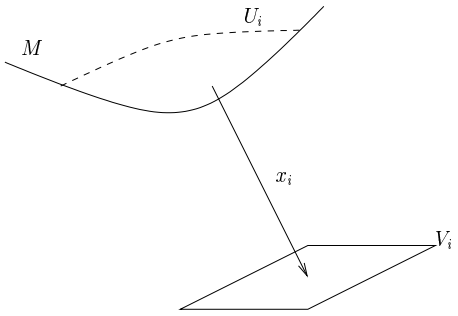
2.1 Manifolds

Definition 2.1 (Manifolds)**.** *An n-dimensional manifold M is a defined as a Hausdorff topological space with a countable basis for its topology, and locally homeomorphic to a subset of* $\mathbf{R}^n$.

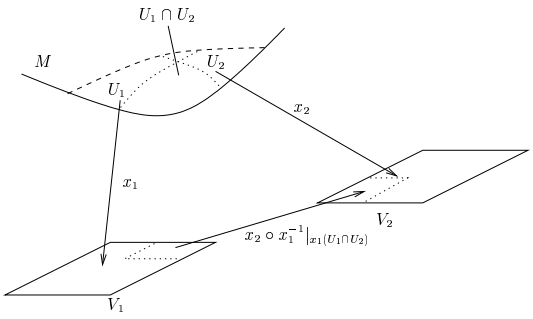
Locally homeomorphic to a subset of $\mathbf{R}^n$ means that there exists an atlas — a collection of charts

$$x_i \colon U_i \to V_i \tag{2.1}$$

where $U_i \subseteq M$ and $x_i(U_i) = V_i \subseteq \mathbf{R}^n$, such that all the U_i's cover M.



In order to do *differential* topology we need to bring in the concept of a differentiable structure on the manifold. This is to ensure that all the properties we discuss will be independent of the choice of charts.



For any two charts

$$x_i \colon U_i \longrightarrow x_i(U_i), \quad i = 1, 2$$

we define the chart transformation

$$x_{12} = x_2 \circ x_1^{-1}|_{x_1(U_1 \cap U_2)} \colon x_1(U_1 \cap U_2) \longrightarrow x_2(U_1 \cap U_2)$$

which is a mapping from an Euclidean space to another Euclidean space, in which differentiation is well defined. We say that that the manifold M has a *smooth* structure if all possible chart transformations in the atlas is smooth, or C^∞.

To each point $p \in M$ we associate a vector space containing all possible tangent vectors at p. Each of these vector spaces (or tangent spaces) is named T_pM and the tangent bundle of the manifold is defined as the union of these

$$\begin{array}{c} TM = \coprod_{p \in M} T_pM \\ {\scriptstyle \pi}\big\downarrow \\ M \end{array} \tag{2.2}$$

where π is the canonical projection sending any vector in T_pM to p. T_pM is also called the fiber above p.

The elements of the tangent space are written as equivalence classes of curves $[\gamma]$, $\gamma\colon \mathbf{R} \to M$ where two curves γ and σ are considered equivalent if their derivative under any chart coincide at zero, that is if $(x\gamma)'(0) = (x\sigma)'(0)$. The choice of the chart x or the value 0 does not matter.

Locally the tangent bundle can always be written as a product bundle $U \times E \to U \subset M$.

Definition 2.2. *Let $f\colon M \longrightarrow N$ be a mapping between two manifolds. The tangent mapping of f over a point $p \in M$ is defined as*

$$\begin{gathered} T_pf\colon T_pM \longrightarrow T_{f(p)}N \\ T_pf([\gamma]) = [f\gamma] \end{gathered}$$

where $\gamma(0) = p$.

A tangent bundle is said to be *trivial* if the space TM can be represented by a product manifold $M \times E$ for a vector space E, that is the following diagram commutes

$$\begin{array}{ccc} TM & \xrightarrow{\quad h \quad} & M \times E \\ & {\scriptstyle \pi}\searrow \quad \swarrow{\scriptstyle pr_M} & \\ & M & \end{array} \tag{2.3}$$

where h is the *bundle chart* as defined in [8, Definition 5.1.1], and pr_M is the projection on the first factor.

For a simple example take the manifold $\mathbf{R}^n$. The tangent space $T_p\mathbf{R}^n$ at each point $p \in \mathbf{R}^n$ can be identified by $\mathbf{R}^n$ itself, and we get the isomorphism $T\mathbf{R}^n \cong \mathbf{R}^n \times \mathbf{R}^n$.

When the tangent bundle of a manifold is *trivial*, the manifold itself is called *parallelizable*.

2.2 Submanifolds

A submanifold is as its name suggests, a subset of a manifold. We require some more for the subset to be a manifold on its own.

Definition 2.3. *A submanifold of dimension k is a subset of a manifold of dimension n, in which there exists homeomorphisms that will map domains in the subspace to $\mathbf{R}^k \times 0^{n-k} \subset \mathbf{R}^n$.*

A subset of all submanifolds can be realized as inverse images of surjective maps.

Theorem 2.4. *[8, Theorem 6.4.3] Let $f\colon M \to N$ where $\dim(M) = n + k$ and $\dim(N) = n$. If $q = f(p)$ is a regular value, that is, the linear mapping $T_p f$ is surjective, then $f^{-1}(q)$ is a k-dimensional submanifold of M.*

All manifolds which we will encounter in this thesis, can be realized in this way. For example, the matrix groups we will cater for in Section 2.7 will all be submanifolds of $n \times n$ matrices.

2.3 Vector fields and differential equations

Definition 2.5. *A vector field on a manifold M is a mapping*

$$F\colon M \longrightarrow TM$$

such that $\pi \circ F = id_M$ (F is a section to the tangent bundle). The collection of all vector fields on a manifold M is denoted by $\mathfrak{X}(M)$.

Vector fields may be added together pointwise, $(F + G)(p) = F(p) + G(p)$, and we also have scalar multiplication $(\alpha F)(p) = \alpha(F(p))$, so $F + G \in \mathfrak{X}(M)$ and $\alpha F \in \mathfrak{X}(M)$.

A vector field may be applied to a function. In this setting, the vector field acts as a derivation on the ring of functions $M \to \mathbf{R}$. Let F be a vector field on M, and given a function $\psi\colon M \to \mathbf{R}$, we define

$$F[\psi] = \pi_2 \circ T\psi \circ F \tag{2.4}$$

where π_2 is the projection to the value in the tangent space above $\psi(p)$, which is also $\mathbf{R}$ here. This is motivated by the fact that $T\mathbf{R}$ has two components through the isomorphism $T\mathbf{R} \cong \mathbf{R} \times \mathbf{R}$. By this definition, we get the Leibniz rule for vector fields, here shown pointwise

$$F[\psi\phi](p) = F[\psi](p) \cdot \phi(p) + \psi(p) \cdot F[\phi](p) \tag{2.5}$$

where multiplication and addition take place in $\mathbf{R}$. This results in a value in $T_{\psi(p)\phi(p)}\mathbf{R}$.

It is possible to attach an *algebra* structure to the collection of all vector fields. We have already defined addition pointwise, so we form a product (a bracket $[\cdot,\cdot]\colon \mathfrak{X}(M) \times \mathfrak{X}(M) \to \mathfrak{X}(M)$ with the following properties

$$\begin{aligned} [F, G] &= -[G, F] \\ [\alpha F, G] &= \alpha[F, G], \quad \alpha \in \mathbf{R} \\ [F + G, H] &= [F, H] + [G, H] \\ 0 &= [F, [G, H]] + [G, [H, F]] + [H, [F, G]] \end{aligned} \tag{2.6}$$

The bracket with these properties is

$$[F, G] = GF - FG \tag{2.7}$$

to be understood as

$$[F, G][\phi] = G[F[\phi]] - F[G[\phi]]$$

when $[F, G]$ is applied to the function ϕ.

Given coordinate charts $x_1, \ldots, x_n \colon M \to \mathbf{R}$, component i of the bracket(through the use of charts) is given as

$$[F, G]_i = \sum_{j=1}^{n} \left(G_j \frac{\partial F_i(y)}{\partial x_j} - F_j \frac{\partial G_i(y)}{\partial x_j} \right)$$

Next we define what a differential equation on a manifold looks like.

Definition 2.6. *Let F be a vector field on M. A (non-autonomous) differential equation on M is an equation of the form*

$$\frac{\mathrm{d}y}{\mathrm{d}t} = F(y(t)), \quad y(0) = y_0 \in M. \tag{2.8}$$

For an autonomous differential equation we replace the right-hand-side by $F(t, y(t))$.

A solution of Equation (2.8) is denoted by a flow operator $\Psi_{t,F} \colon M \to M$. The solution for any time given the initial condition can then be written as

$$y(t) = \Psi_{t,F}(y_0) \tag{2.9}$$

In the following chapters, ϕ_h will also be used instead of $\Psi_{t,F}$ for the exact solution, and Φ_h for a numerical solution. There is no restriction in always specifying the initial condition at $t = 0$. For compact smooth manifolds the solution will be globally defined (for all $t \in \mathbf{R}$) [8, Chapter 9], while for non-compact manifolds we can only hope for a locally defined solution, ie. for $t \in J \subset \mathbf{R}$ where J will depend on the initial value.

Often exp is used as the solution operator, since the exponential may also be defined as an operator mapping a vector field F to the solution of the associated differential equation (2.8).

2.4 Lie groups

Lie groups are manifolds equipped with a group structure. They stem from the works of Sophus Lie in the 19th century, who named them Transformation groups. An extensive source of information for Lie groups and Lie algebras is Varadarajan's book [24].

Definition 2.7 (Lie group)**.** *A Lie group G is a smooth manifold with a smooth group structure. That is for each element $g, h \in G$ there exist a*

i) Group operation, $g \cdot h \in G$ which is smooth in the sense of the manifold.

ii) Inverse, $g^{-1} \in G$ such that $g \cdot g^{-1} = id \in G$.

Often the smoothness of the inverse map $g \mapsto g^{-1}$ is included in the definition of a Lie group. It can be shown that this is strictly not necessary as it follows from the smoothness of the multiplication.

Definition 2.8 (Lie subgroup)**.** *A subgroup H of a group G is a subset of G which is also a group. For a subset H of a Lie group G to be Lie subgroup, we must have that*

i) H is a subgroup of G

ii) H is a submanifold of G

A beautiful theorem significantly helping later results is the following

Theorem 2.9 (E. Cartan)**.** *Closed subgroups of Lie groups are Lie subgroups.*

Proof. The proof may be found in Lemma 2.28, 2.29 and 2.30 in [1] and in Theorem 3.6 of [20] □

This theorem greatly simplifies the proofs of why several matrix groups really are Lie groups.

Proposition 2.10. *All Lie groups are parallelizable.*

Proof. We recall from Section 2.1 that manifolds are parallelizable if their tangent bundle is trivial, Equation (2.3).

Let g be an element in a Lie group G. By the group structure, there exist a $g^{-1} \in G$. Let e be the identity in G and $T_{id}G$ the tangent space at the identity. We need a diffeomorphism

$$TG \xrightarrow{\cong} G \times T_{id}G.$$

Let γ be a curve in G with $\gamma(0) = g$. An isomorphism is

$$[\gamma] \mapsto (\gamma(0), [\gamma(0)^{-1}\gamma]).$$

□

2.5 Lie series

For functions $\psi\colon M \to \mathbf{R}$ we are interested in how ψ varies along a flow of a differential equation on the Lie Group, defined by the vector field F. Given the Lie group differential equation

$$y'(t) = F(y(t)), \quad y(0) = y_0 \in G, \quad F\colon G \to TG$$

the solution defines the exponential of a vector field as

$$y(t) = \exp(tF)y_0).$$

Applying ψ to the solution $y(t)$ and differentiate at $t = 0$

$$\left.\frac{\mathrm{d}}{\mathrm{d}t}\right|_{t=0} \psi(\exp(tF)y_0) = F[\psi](y_0) \tag{2.10}$$

where the equality stems from the definition of $F[\psi]$. $F[\psi]$ is also a function $G \to \mathbf{R}$ so it is possible to continue

$$\left.\frac{\mathrm{d}}{\mathrm{d}t}\right|_{t=0} F[\psi](\exp(tF)y_0) = F[F[\psi]](y_0) = F^2[\psi](y_0)$$

Taylor's theorem for real-valued functions must apply to our ψ as well, assuming F and ψ to be in C^∞, we have the formal series

$$\psi(\exp(tF)y_0) = \sum_{k=0}^{\infty} \frac{t^k}{k!} F^k[\psi](y_0) \tag{2.11}$$

This expansion is called the *Lie series* for the function ψ. Note that this is a formal series, as convergence is not considered. As we have only required C^∞ it might happen that the series diverge. In applications, we will always use truncations.

2.6 Lie algebras

Definition 2.11 (Lie algebra and Lie bracket)**.** *A vector space over a field is called a* Lie algebra *if there is a bilinear map* $\mathfrak{g} \times \mathfrak{g} \to \mathfrak{g}$ *that satisfies*

i) $[a, b] + [b, a] = 0$ *(skew-symmetry)*

ii) $[a, [b, c]] + [b, [c, a]] + [c, [a, b]] = 0$ *(Jacobi identity).*

This map is the Lie bracket.

From Proposition 2.10 we have that the tangent space at identity of a Lie group determines any point in the tangent space TG. This is a remarkable fact for Lie groups, and means that we can focus on the tangent space at identity for all our tangent purposes in the Lie group.

The tangent space has the algebra structure as in Section 2.3, and we may therefore associate the tangent space at identity $T_{id}G$ of a Lie group G with a Lie algebra, and denote it by the symbol $\mathfrak{g}$.

We will return to more concrete examples of Lie algebras for matrix groups in the next section.

2.7 Matrix groups

Matrix groups are subsets of all $n \times n$ real matrices (denoted $M_n(\mathbf{R})$) to which we attach a group structure. Our groups will also turn out to be Lie groups later. The product in these groups will be the standard matrix product, and the inverse is taken from linear algebra. The group identity element is the diagonal identity matrix $I = \mathrm{diag}(1, \ldots, 1)$.

These matrix groups are going to be used in Lie group methods, catered for in Chapter 3. Especially the groups $SO(3)$ and $SL(2)$ are going to be used in Chapter 6 and 7 respectively.

i) *The General Linear group* $GL(n)$.
This is all the matrices in $M_n(\mathbf{R})$ where the determinant is different from zero,

$$GL(n) = \{A \in M_n(\mathbf{R}) \mid \det(A) \neq 0\} \tag{2.12}$$

We immediately see that the product AB of two matrices, $A, B \in GL(n)$ will still be in $GL(n)$, by

$$\det(AB) = \underbrace{\det(A)}_{\neq 0}\underbrace{\det(B)}_{\neq 0} \neq 0.$$

The inverse of A is A^{-1}, and A^{-1} is defined because $\det(A) \neq 0$. This is also of course in $GL(n)$ because

$$1 = \det(I) = \underbrace{\det(A)}_{\neq 0}\det(A^{-1})$$

and therefore $\det(A^{-1}) \neq 0$.

ii) *The Special Linear group* $SL(n)$.
This is a subset of the $GL(n)$-group, defined as

$$SL(n) = \{A \in M_n(\mathbf{R}) \mid \det(A) = 1\} \tag{2.13}$$

If $A, B \in SL(n)$, then

$$\det(AB) = \det(A)\det(B) = 1$$

so that $AB \in SL(n)$ as well. For the inverse,

$$1 = \det(AA^{-1}) = \det(A)\det(A^{-1}) = \det(A^{-1})$$

so $A^{-1} \in SL(n)$.

iii) *The Orthogonal group* $O(n)$.
This is defined as

$$O(n) = \{A \in M_n(\mathbf{R}) \mid AA^T = I\} \tag{2.14}$$

The condition $AA^T = I$ means that A^T is the inverse of A and that all column vectors $v_i, A = [v_1, \ldots, v_n]$ are orthonormal, $\langle v_i, v_j \rangle = \delta_{ij}$. For the value of the determinant we have

$$1 = \det(AA^T) = \det(A)\det(A^T)$$

and since the determinant is invariant under transposition, $\det(A) = \det(A^T)$ we get that

$$\det(A) \in \{-1, 1\}, \quad \text{for all } A \in O(n). \tag{2.15}$$

A set in $\mathbf{R}^n$ is connected if there exist a path from any element to any other element in the set which is continuous. Let $\gamma\colon [0,1] \to O(n)$ be a curve in $O(n)$ where $\gamma(0) = A, \det(A) = -1$ and $\gamma(1) = B$, $\det(B) = 1$. Can γ be continuous? We compose by a function which we know is continuous, the determinant. If γ is continuous, then $\det \circ \gamma$ is also continuous. But $(\det \circ \gamma)[0,1]$ is a discrete set, and there can't be a continuous path in $\{-1, 1\}$ from -1 to 1. So $O(n)$ is disconnected.

iv) *The Special Orthogonal group* $SO(n)$.
As $O(n)$ was not connected, we name the component containing the identity element the Special Orthogonal group,

$$SO(n) = \{A \in O(n) \mid \det(A) = 1\}. \tag{2.16}$$

The group structure is inherited from $O(n)$. The other component in $O(n)$, $O(n)\backslash SO(n)$, is not a group, because if $B, C \in O(n)\backslash SO(n)$, then

$$\det(BC) = \det(B)\det(C) = (-1)(-1) = 1$$

so $BC \in SO(n)$. In addition, $O(n)\backslash SO(n)$ does not contain any identity element.

$SO(n)$ is also called the group of rotations. This is because applied to vectors in $\mathbf{R}^n$, the vectors are rotated around the origin. Their lengths are preserved.

Proposition 2.12. *All orthogonal matrices preserve lengths of vectors under matrix-vector products, for* $A \in O(n)$ *and* $x \in \mathbf{R}^n$

$$\|Ax\| = \|x\|.$$

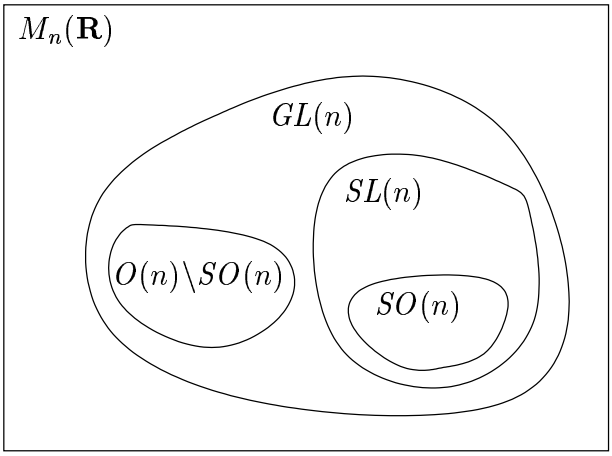


Figure 2.1: The introduced matrix groups and their subset-relationships. Note that $O(n) = SO(n) \cup O(n)\backslash SO(n)$.

Proof. By use of the definition of norm, and the definition of the transpose via the inner product

$$\begin{aligned} \|Ax\| &= \langle Ax, Ax \rangle \\ &= \langle x, A^T Ax \rangle \\ &= \langle x, x \rangle = \|x\|. \end{aligned}$$

□

2.8 The matrix exponential

The exponential of a matrix, to be defined as an infinite series (like the Taylor series of e^x when $x \in \mathbf{R}$), is the most important construct for matrix groups. The exponential map will later be used as a chart for the matrix groups when we prove that they are manifolds and thereby Lie groups.

Definition 2.13 (The exponential of a matrix)**.** *Let $A \in M_n(\mathbf{R})$ and define*

$$\exp(A) = \sum_{i=0}^{\infty} \frac{A^i}{i!}.$$

This sequence is said to converge if all the elements (scalar) of the matrix $\exp(A)$ converge in $\mathbf{R}$. It does in fact converge for all $A \in M_n(\mathbf{R})$ [7, Chapter 4, Proposition 1].

Proposition 2.14. *If the matrices A and B in $M_n(\mathbf{R})$ commute ($AB = BA$) then*

$$\exp(A + B) = \exp(A) \exp(B)$$

Proof.

$$\begin{aligned}
\exp(A+B) &= \sum_{i=0}^{\infty} \frac{(A+B)^i}{i!} \\
&= \sum_{i=0}^{\infty} \frac{1}{i!} \sum_{k=0}^{i} \binom{i}{k} A^{i-k} B^k \quad \text{(by commutativity)} \\
&= \sum_{i=0}^{\infty} \sum_{k=0}^{i} \frac{A^{i-k}}{(i-k)!} \frac{B^k}{k!} = \sum_{k=0}^{\infty} \frac{B^k}{k!} \sum_{i=k}^{\infty} \frac{A^{i-k}}{(i-k)!} \\
&= \left(\sum_{j=0}^{\infty} \frac{A^j}{j!} \right) \left(\sum_{k=0}^{\infty} \frac{B^k}{k!} \right) \quad \text{by } j = i-k \text{ and commutativity} \\
&= \exp(A) \exp(B)
\end{aligned}$$

□

From this proposition, the next one follows

Proposition 2.15. *The exponential of any matrix in $M_n(\mathbf{R})$ is invertible.*

Proof. The matrices A and $-A$ commute, so

$$I = \exp(0) = \exp(A - A) = \exp(A) \exp(-A)$$

and

$$1 = \det(\exp(A) \exp(-A)) = \det(\exp(A)) \det(\exp(-A))$$

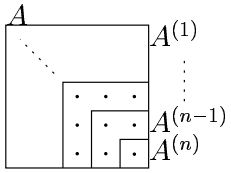
so $\det(\exp(A)) \neq 0$ and the matrix $\exp(A)$ is invertible. □

Proposition 2.16. *The derivative of the determinant at the identity I is*

$$T_I \det(A) = \operatorname{tr}(A)$$

where $\operatorname{tr}(A)$ is the trace of the matrix A.

We need some notation to get ready for the proof. Let here a parenthesized superscript $^{(k)}$ of a matrix A denote the lower right $(n-k+1 \times n-k+1)$-submatrix of A, $A^{(1)} = A$, $A^{(n)} = a_{nn}$, and two subscripts $_{ij}$ of a matrix denotes the ij'th cofactor of the matrix (row i and column j removed). The ij'th element of a matrix A is denoted by lower case letter and subscripts, a_{ij}. The superscript has precedence over the subscript in this proof, that is we are making cofactors of the submatrices, although the cofactor-indices are relative to the whole matrix.



Proof. Let $I + tA$ be a path in $M_n(\mathbf{R})$, then

$$T_I \det(A) = \left. \frac{\mathrm{d}}{\mathrm{d}t} \right|_{t=0} \det(I + tA)$$

The proof will be on induction, our induction hypothesis is

$$\left.\frac{\mathrm{d}}{\mathrm{d}t}\right|_{t=0} \det(I+tA)^{(k)} = a_{kk} + \left.\frac{\mathrm{d}}{\mathrm{d}t}\right|_{t=0} \det(I+tA)^{(k+1)} \tag{2.17}$$

For the smallest lower right submatrix, we get

$$\left.\frac{\mathrm{d}}{\mathrm{d}t}\right|_{t=0} \det(I+tA)^{(n)} = \left.\frac{\mathrm{d}}{\mathrm{d}t}\right|_{t=0} (1+ta_{nn}) = a_{nn}.$$

Let us prove the hypothesis:

$$\begin{aligned}\left.\frac{\mathrm{d}}{\mathrm{d}t}\right|_{t=0} \det(I+tA)^{(k)} = \left.\frac{\mathrm{d}}{\mathrm{d}t}\right|_{t=0} \Big(&(1+ta_{kk}) \det(I+tA)^{(k)}_{kk} \\ &-(ta_{k,k+1}) \det(I+tA)^{(k)}_{k,k+1} + (ta_{k,k+2}) \det(I+tA)^{(k)}_{k,k+2} - \cdots \Big)\end{aligned}$$

where we have expanded the determinant along the first row using cofactors

$$\begin{aligned}&= a_{kk} \det(I^{(k)}) + 1 \cdot \left.\frac{\mathrm{d}}{\mathrm{d}t}\right|_{t=0} \det(I+tA)^{(k+1)} \\ &\quad - a_{k,k+1} \underbrace{\det(I^{(k)}_{k,k+1})}_{=0} - (0 \cdot a_{k,k+1}) \left.\frac{\mathrm{d}}{\mathrm{d}t}\right|_{t=0} \det(I+tA)^{(k)}_{k,k+1} + \cdots\end{aligned}$$

all cofactors except the diagonal cancel

$$= a_{kk} + \left.\frac{\mathrm{d}}{\mathrm{d}t}\right|_{t=0} \det(I+tA)^{(k+1)}$$

Since this was true for $k = n$, we must have that

$$\left.\frac{\mathrm{d}}{\mathrm{d}t}\right|_{t=0} \det(I+tA)^{(1)} = \sum_{k=1}^{n} a_{kk} = \operatorname{tr}(A) \tag{2.18}$$

□

Proposition 2.17. *For any $A \in M_n(\mathbf{R})$ we have*

$$\det(\exp(A)) = \mathrm{e}^{\operatorname{tr}(A)} \tag{2.19}$$

Before the proof, we note the result

Corollary 2.18. exp *maps matrices with zero trace to $SL(n)$.*

Proof of Proposition 2.17. We make no assumptions on A. A possesses a Jordan decomposition

$$A = WJW^{-1} \tag{2.20}$$

where $\det(W) \neq 0$ and J is block diagonal, with s block elements

$$J_k = \begin{pmatrix} \lambda_k & 1 & \\ & \ddots & 1 \\ & & \lambda_k \end{pmatrix}$$

each with dimension $n_k \times n_k$. We need the exponential of these blocks, only shown here for 2×2-matrices. By using the series expansion for exp and that

$$\begin{pmatrix} \lambda_k & 1 \\ 0 & \lambda_k \end{pmatrix}^i = \begin{pmatrix} \lambda_k^i & (i+1)\lambda_k^i \\ 0 & \lambda_k^i \end{pmatrix}$$

we get

$$\exp(J_k) = \begin{pmatrix} \mathrm{e}^{\lambda_k} & \sum_{j=0}^{\infty} \frac{(j+1)\lambda_k^j}{j!} \\ 0 & \mathrm{e}^{\lambda_k} \end{pmatrix}.$$

Then

$$\det(\exp(J_k)) = \mathrm{e}^{\lambda_k}\mathrm{e}^{\lambda_k} - 0 \cdot \sum_{j=0}^{\infty} \frac{(j+1)\lambda_k^j}{j!} = \mathrm{e}^{2\lambda_k}.$$

By a tedious extension to arbitrarily sized blocks we have an intermediate result:

$$\begin{aligned} \det(\exp(A)) &= \det(\exp(WJW^{-1})) \\ &= \det(W \exp(J) W^{-1}) \qquad \text{(exp is invariant under conjugation)} \\ &= \det(\exp(J)) \\ &= \prod_{k=1}^{s} \det(\exp(J_k)) \\ &= \mathrm{e}^{\sum_{k=1}^{s} n_k \lambda_k} = \mathrm{e}^{\mathrm{tr}(J)} \end{aligned}$$

Starting at the other end, we show that the trace of matrix is equal to the sum of the eigenvalues (repeated eigenvalues are summed according to their multiplicity). This is based on $\mathrm{tr}(AB) = \mathrm{tr}(BA)$ which is verified by simply writing out the elementwise expression of the matrix product. Because of this,

$$\mathrm{tr}(A) = \mathrm{tr}(W(JW^{-1})) = \mathrm{tr}((JW^{-1})W) = \mathrm{tr}(J)$$

and we have proved Proposition 2.17. □

Proposition 2.19. exp *maps skew-symmetric matrices to the Special Orthogonal group,*

$$\exp\colon \mathfrak{so}(n) \longrightarrow SO(n). \tag{2.21}$$

Proof. First we prove that exp maps to the Orthogonal group. A skew-symmetric A is a matrix $A \in M_n(\mathbf{R})$ such that $A + A^T = 0$. A and A^T commute when A is skew-symmetric because $AA^T - A^TA = -AA + AA = 0$. From the previous propositions we get

$$I = \exp(0) = \exp(A + A^T) = \exp(A)\exp(A^T) = \exp(A)\exp(A)^T$$

so $\exp(A)$ must be orthogonal.

To prove that it only maps to the identity component $SO(n)$ of $O(n)$ we need a connectedness argument from topology. Both the exponential and the determinant (and their composition) are continuous functions, which map connected sets to connected sets. Let $\gamma(t) = \exp(tB)$ be a path in $O(n)$, where $B \in \mathfrak{so}(n)$. We have

$$\det \circ \exp\colon \mathfrak{so}(n) \longrightarrow \{-1, 1\}$$

For $t \in [0, 1]$ (a connected set in $\mathbf{R}$), we have $\det(\gamma(0)) = \det(I) = 1$, and the only possible value for $\det(\gamma(1))$ is 1 because we need a connected set in $\{-1, 1\}$. So exp must map only to the Special Orthogonal group. □

We note that we have not proved any surjectivity, which it is not true either. But we are eventually going to prove local injectivity, for this we need a potential inverse.

Definition 2.20 (The logarithm of a matrix)**.** *The logarithm of a matrix $A \in M_n(\mathbf{R})$ is defiend as*

$$\log(A) = \sum_{i=1}^{\infty} (-1)^{i+1} \frac{(A-I)^i}{i}. \tag{2.22}$$

For A sufficiently near the identity I this series will converge [7, Chapter 4, Proposition 5] (each component of $A - I$ must be less than $1/n$).

Proposition 2.21. *Let $\mathcal{N}_I$ be a sufficiently small neighborhood of I in $M_n(\mathbf{R})$ and $\mathcal{N}_0$ a neighborhood of 0 in $M_n(\mathbf{R})$ such that $\exp(\mathcal{N}_0) \subseteq \mathcal{N}_I$. The map*

$$\exp\colon \mathcal{N}_0 \longrightarrow \mathcal{N}_I \tag{2.23}$$

will be injective with log *as its inverse. That is, there exists*

i) a $B \in \mathcal{N}_0$ such that $\log(\exp(B)) = B$

ii) an $A \in \mathcal{N}_I$ such that $\exp(\log(A)) = A$.

Proof. We use the series expansions of exp and log and rearrange the terms so that they cancel

i)

$$\begin{aligned}
\log(\exp(B)) &= \left(B + \frac{B^2}{2!} + \cdots\right) - \frac{1}{2}\left(B + \frac{B^2}{2!} + \cdots\right)^2 + \frac{1}{3}\left(B + \frac{B^2}{2!} + \cdots\right)^3 \\
&= B + \left(\frac{B^2}{2!} - \frac{B^2}{2}\right) + \left(\frac{B^3}{6} - \frac{B^3}{2} + \frac{B^3}{3}\right) + \cdots \\
&= B.
\end{aligned}$$

ii)

$$\begin{aligned}
\exp(\log(A)) &= I + \left((A-I) - \frac{(A-I)^2}{2} + \cdots\right) + \frac{1}{2!}\left((A-I) - \frac{(A-I)^2}{2} + \cdots\right) + \cdots \\
&= A - \left(\frac{(A-I)^2}{2} + \frac{(A-I)^2}{2}\right) + \left(\frac{(A-I)^3}{3} - \frac{(A-I)^3}{2} + \frac{(A-I)^3}{6}\right) + \cdots \\
&= A.
\end{aligned}$$

□

Proposition 2.22. *If A, B and AB are in $\mathcal{N}_I$ (from Proposition 2.21), and if $\log(A)$ and $\log(B)$ commute, then*

$$\log(AB) = \log(A) + \log(B).$$

Moreover, if A is orthogonal, $\log(A)$ is skew-symmetric.

Proof. A, B and AB are in the domain where exp is injective,

$$\begin{aligned}\exp(\log(AB)) = AB &= \exp(\log(A))\exp(\log(B)) \\ &= \exp(\log(A) + \log(B))\end{aligned}$$

where the last equality follows from Proposition 2.14. Taking log of this equation yields the result. For the second part, let now $A \in \mathcal{N}_I \cap SO(n)$. A and A^T commute as mentioned before, that causes $\log(A)$ and $\log(A^T)$ to commute as well by looking at the series expansion. Since $\log(I) = 0$ and the transpose operation commutes with the logarithm operation (again by looking at the series expansion) we get

$$0 = \log(I) = \log(AA^T) = \log(A) + \log(A^T) = \log(A) + (\log(A))^T$$

which means that $\log(A)$ is skew-symmetric. □

The last part of this proposition was an extension to Proposition 2.19 saying that no other than skew-symmetric matrices map to the special orthogonal group. The results obtained so far on orthogonal matrices may be visualized as in Figure 2.2.

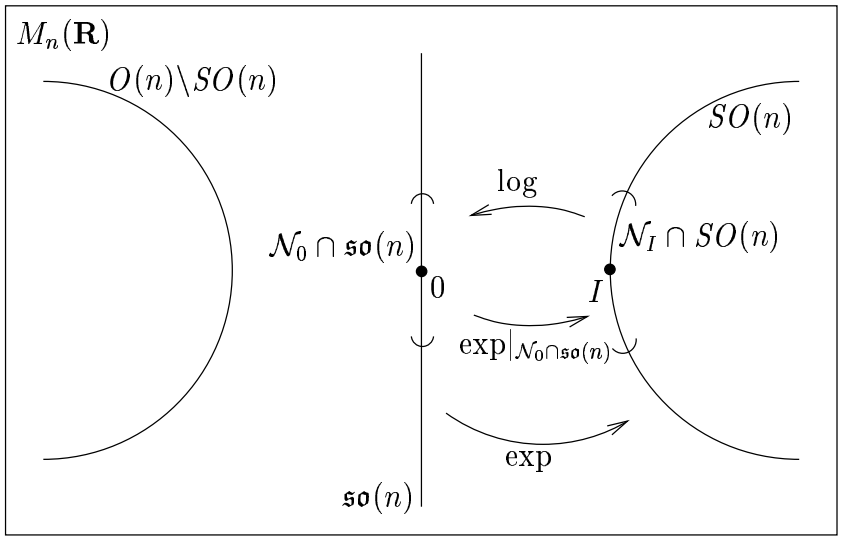


Figure 2.2: Summary of results on the exponential mapping for skew-symmetric matrices and the Special Orthogonal group, we have $\exp\colon \mathfrak{so}(n) \to SO(n)$, $\log\colon \mathcal{N}_I \to \mathfrak{so}(n)$, and that $\exp|_{\mathcal{N}_0}\colon \mathcal{N}_0 \to \mathcal{N}_I$ is injective. This figure also illustrates the linearity of $\mathfrak{so}(n)$ and the non-linearity of $SO(n)$.

2.9 Matrix groups are Lie groups

In this section we would like to prove that all our matrix groups covered mentioned in Section 2.7 really are Lie groups. Since we know they are groups, we need to prove that they are manifolds.

All our groups may be given chart in the following way. Let $\mathcal{N}_g$ be a neighborhood of the element $g \in G$. Let $\mathcal{N}_g$ be a chart domain, and use the chart

$$\log \circ L_{g^{-1}} : \mathcal{N}_g \longrightarrow \mathcal{N}_0 \tag{2.24}$$

where $L_{g^{-1}}$ is left-translation by the inverse of g in G. The domains $\mathcal{N}_g$ and $\mathcal{N}_0$ are chosen such that $\log \circ L_{g^{-1}}$ becomes a homeomorphism. We need $\mathcal{N}_0$ to be a subset of $\mathbf{R}^n$, this is so because of the properties of the logarithmic map developed in the previous section.

Cartan's Theorem, here Theorem 2.9, also provides proof with the result that our matrix groups are manifolds, with examples below.

The last source of proof is Theorem 2.4 which says that a subset of all submanifolds can be realized as inverse images of regular values.

i) *The General Linear group GL(n)*
This is an open subset of $M_n(\mathbf{R})$, as the determinant

$$\det : M_n(\mathbf{R}) \longrightarrow \mathbf{R}$$

is a continuous function, and so the inverse image of $\mathbf{R}\backslash 0$ must also be open.

ii) *The Special Linear group SL(n)*
This group is the inverse image of $\{1\}$ for the following determinant map

$$\det : M_n(\mathbf{R}) \longrightarrow \mathbf{R}$$

because $T_A \det$ is surjective for some A with $\det(A) = 1$. The fact that $T_A \det$ is surjective follows from the proof of Proposition 2.16.

In addition, $\{1\}$ is a closed set, and the determinant is continuous, so $SL(n)$ is closed and thereby a manifold by Cartan's Theorem.

iii) *The Orthogonal group O(n)*
Here we look at the mapping

$$\begin{aligned} f : GL(n) &\longrightarrow Sym(n) \\ A &\mapsto A^T A \end{aligned}$$

where $Sym(n)$ is the space of $n \times n$-symmetric matrices. If A is orthogonal, then $f(A) = I$. By Example 6.4.11 in [8], $T_A f$ is surjective, and $f^{-1}(I) = O(n)$ is a manifold.

Cartan's Theorem also applies here, since every element in every column vector of $O(n)$ must be less than 1 in absolute value for the column vectors to be orthonormal. Thus $O(n)$ is a closed subset of $[-1, 1]^{n^2}$ and therefore a manifold.

iv) *The Special Orthogonal group SO(n)*
This is the component of $O(n)$ where the determinant is equal to 1. For the continuous mapping

$$\det : O(n) \longrightarrow \{-1, 1\}$$

$SO(n)$ is the inverse image of $\{1\}$, which is closed, and therefore $SO(n)$ is a submanifold of $O(n)$ by Cartan's Theorem.

Note that the other component of $O(n)$, $O(n)\backslash SO(n)$ is also closed by the same argument. Since $O(n)$ is closed, this means that the complement of $O(n)\backslash SO(n)$ is open, that is $SO(n)$ (and its complement) is both open and closed.

2.10 The tangent spaces of the Lie groups

By the trivialization of Lie group tangent bundles (Proposition 2.10), the tangent space at the identity of all Lie group characterize the whole tangent bundle of the group.

Essential for the tangent spaces is the following definition:

Definition 2.23. *A one-parameter subgroup of G is a homomorphism of Lie groups*

$$\gamma\colon \mathbf{R} \longrightarrow G$$

ie, $\gamma(s+t) = \gamma(s)\gamma(t)$.

Adams [1, Theorem 2.6] proves the one-to-one correspondence between vectors in $T_{id}G$ and one-parameter subgroups. These one-parameter subgroups are solutions to differential equations determined by the value in $T_{id}G$, and the solution is defined to be the exponential map. So for a vector $A \in T_{id}G$, the corresponding one-parameter subgroup is $\gamma(tA) = \exp(tA)$. The crucial point now is to see that all the results on the exponential map for various Lie groups we have obtained, characterize the Lie group's tangent spaces.

Let us summarize this for the matrix groups already discussed:

i) *The General Linear group $GL(n)$*
Proposition 2.15 gives that for $A \in M_n\mathbf{R}$, $\exp(A)$ will be in $GL(n)$. Thereby, $T_{id}\,GL(n) = M_n\mathbf{R}$ which is the Lie algebra $\mathfrak{gl}(n)$. The dimension of $GL(n)$ and $\mathfrak{gl}(n)$ is n^2.

ii) *The Special Linear group $SL(n)$*
Corollary 2.18 gives that all trace-free matrices map to $SL(n)$ under exp, and $\mathfrak{sl}(n)$ is thereby $T_{id}SL(n)$. The zero trace requirement removes one degree of freedom from $M_n\mathbf{R}$, so $\dim \mathfrak{sl}(n) = n^2 - 1$.

iii) *The Orthogonal and Special Orthogonal group $O(n)$and $SO(n)$*
Proposition 2.19 says that skew-symmetric matrices map to orthogonal matrices. An $n \times n$-matrix has $n(n-1)/2$ elements above its diagonal, and this is the dimension of $n \times n$-skew-symmetric matrices, so $\mathfrak{o}(n) = \mathfrak{so}(n)$ is the tangent space at identity of $O(n)$and $SO(n)$. Another common way of seeing this is by differentiating the curve $\rho(t)\rho(t)^T$ which equals to id for all $t \in \mathbf{R}$ when $\rho(t) \in O(n)$, and $\rho(0) = id$.

$$0 = \frac{\mathrm{d}}{\mathrm{d}t}\bigg|_{t=0} (\rho(t))\rho(0)^T + \rho(0)\,\frac{\mathrm{d}}{\mathrm{d}t}\bigg|_{t=0} (\rho(t)^T) = \rho(0) + \rho(0)^T$$

which says that $\rho(0)$ is skew-symmetric.

Chapter 3

Lie group Methods

Solvers for the standard problem in numerical analysis of differential equations for the last 100 or so years has been designed for the problem

$$y' = F(y), \quad y(0) = y_0, \quad y(t) \in \mathbf{R}^n. \tag{3.1}$$

Numerical solutions to this problem has been developed with few assumptions on the right hand side f, through careful discretization of the equations and solvers and ensuring the local truncation error is minimized. This together with step-size control, has led to robust and general black-box algorithms for solving (3.1), covered in great detail by the books by Hairer, Nørsett and Wanner [11, 12].

3.1 Introduction to Geometric Integration

Geometric integration represents a new philosophy for the solution of (3.1). Notice the solution space in (3.1), $y(t) \in \mathbf{R}^n$. We can imagine more general structures (enter manifolds) on which the solution is known to evolve (which we also call the configuration space), but by Whitney's embedding theorem [8, Theorem 7.5.1] we know that any manifold of dimension n may be embedded in a Euclidean space of at most dimension $2n$ — and thereby the general solvers for equation (3.1) do still apply. However, there is room for improvement. The classical solvers of Equation (3.1) do not always take advantage of any special structures that the manifold or equation in question may possess.

Geometric integration is about using a priori knowledge about the solution from the given differential system, whether the solution is to evolve in a general manifold or in a Euclidean space, and then obtaining a solver which produces a numerical approximation preserving the qualitative attributes of the system. Classical solvers typically preserve the attributes less accurately or not at all. For differential equations on manifolds, we are interested in having a solver which at all times will produce approximations in the manifold, in addition to any other properties of the equation.

The downside of Geometric Integration is the loss of generality, we are no longer developing robust routines which are able to tackle any differential equation, but rather narrowing in the scope of our solvers to only a subset of all differential equations.

For a comprehensive introduction to Geometric Integration there is the new book by Hairer, Lubich and Wanner to be recommended [10], and also the overview article by Budd and Piggott [3].

3.2 Lie group methods on Homogeneous Manifolds

This chapter will cover a subset of Geometric Integration, the Lie group methods. The usability of Lie groups stem from their ability to *"act"* on manifolds and thereby provide means of motion on the manifold. The Lie group methods we are using here, were originally developed in the papers of Munthe-Kaas [17, 18, 19]. The methods due to Crouch and Grossman will not be considered. Other reference material is the overview article by Iserles, Munthe-Kaas, Nørsett and Zanna [14]. Iserles has also written a brief introduction to Lie group methods [13].

The configuration space of Equation (3.1) is $\mathbf{R}^n$ which is a linear space and the standard classical methods (Runge-Kutta and multistep) use linear translations for motion in the configuration space, $y_{n+1} = y_n + \delta, \delta \in \mathbf{R}^n$. For a manifold M (not linear in general) embedded in $\mathbf{R}^n$ for some n, such linear motions can not guarantee that $y_n \in M$ for all n, which will be the case for the exact solution. We would like to have another way of motion which can make that guarantee, specifically designed for the manifold M.

Lie group methods provide a non-linear way of motion through *Lie group actions* for several types of configuration spaces.

Definition 3.1 (Lie group action)**.** *Let M be a smooth manifold and let G be a Lie group. An action of the Lie group G on the manifold M is a smooth mapping $\Lambda : G \times M \to M$ such that*

i) $\Lambda(id, m) = m, \quad$ *for all* $m \in M$.

ii) $\Lambda(g, \Lambda(h, m)) = \Lambda(g \cdot h, m), \quad$ *for all* $g, h \in G, m \in M$.

For fixed $g \in G$ we obtain a diffeomorphism $\Lambda_g \colon M \to M$, and the map $g \mapsto \Lambda_g$ becomes a Lie group homomorphism as

$$\Lambda_{id_G} = id_M, \quad \Lambda_{g \cdot h} = \Lambda_g \circ \Lambda_h.$$

Now we have defined the tool to be used to move around in the manifold M. The idea is have a way to find an element $g \in G$ such that our next step in the manifold is

$$y_{n+1} = \Lambda(g, y_n), \quad \text{for } g \in G, y_n \in M.$$

This is our current framework for a Lie group solver.

If for any points $m, m^* \in M$ there exist a $g \in G$ such that $\Lambda(g, m) = m^*$ then the action is *transitive*.

If we for a solution space have available a transitive Lie group action we are well equipped for developing a Lie group solver.

Definition 3.2 (Homogeneous space)**.** *A manifold M with a transitive Lie group action Λ is called a homogeneous space, denoted by the triple (M, G, Λ).*

Although we refer to the methods we are going to develop as Lie *group* methods (note again that Crouch-Grossman methods are not considered), they might just as well be referred to as Lie *algebra* methods. What we aim to do, is to transfer our equation into an equation on the Lie algebra of the Lie group of our homogeneous space. The Lie algebra $\mathfrak{g}$ is a linear space, and if we are able to reformulate our equation in $\mathfrak{g}$, we can apply our classical methods

for solving ordinary differential equations, and then step back to our manifold through the chart (the exponential map) of the Lie group and the Lie group action.

Let $\tilde{\Phi}_h$ be a time stepping-procedure on $\mathfrak{g}$ which takes a point $u_n \in \mathfrak{g}$ to a new point $u_{n+1} \in \mathfrak{g}$ for some differential equation. When u_{n+1} has been found, we may step back to our manifold. We summarize the Lie group approach in the following figure, where we note that not all maps are defined yet.

$$\begin{array}{ccccc} M \ni y_n & \longmapsto & & & u_n \in \mathfrak{g} \\ & & & & \Big\downarrow \tilde{\Phi}_h(y_n) \\ M \ni y_{n+1} & \xleftarrow{\Lambda_{y_n}} & \exp(u_{n+1}) \in G & \xleftarrow{\exp} & u_{n+1} \in \mathfrak{g} \end{array} \tag{3.2}$$

As for the Lie group action, we define a corresponding Lie algebra action using the chart for the Lie group and the Lie group action.

$$\begin{aligned} \lambda : \mathfrak{g} \times M &\to M \\ \lambda(v, m) &= \Lambda(\exp(v), m) \end{aligned} \tag{3.3}$$

This way of solving the differential equation on M guarantees that our approximated solution will stay exactly on the manifold M, this is ensured by the construction of the Lie group action.

Differential equations for Lie (matrix) group methods are often represented in a slightly different way than equation (3.1), as

$$y' = f(y)y, \quad y(0) = y_0$$

where $f\colon M \to \mathfrak{g}$ and $f(y)y$ really is $\lambda_*(f(y))(y)$ (λ_* to be defined in Equation (3.11)). We will soon get back to why and how this can be done, we here just note it to get ready for a full-fleshed example. Using this as a starting point, we are already ready to grasp a first example of a Lie group method (justification will appear shortly), a Lie-version of the simple Euler method, Lie-Euler. Referring to Diagram (3.2), we set $u_n = 0$, and use a forward Euler step on $\mathfrak{g}$ with time step h, then let u_{n+1} act on our manifold M again, and the result becomes

$$y_{n+1} = \Lambda(\exp(hf(y_n)), y_n)$$

Later we will clarify how and why this will work. Note that for most applications, G will be a matrix Lie group acting on a manifold M embedded in a Euclidean space (each point represented by a vector), and the Lie group action will be manifested as just ordinary matrix-vector multiplication. We therefore allow for a shorter notation for Lie-Euler (and other Lie group methods)

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

Example 3.1 (Euler and Lie-Euler)**.**
We elaborate here on a simple and familiar example of the flow of a differential equation on the sphere, S^2. The homogeneous space consists of the manifold S^2, the Lie group $SO(3)$ of rotations in $\mathbf{R}^3$ and the Lie group action Λ which is matrix-vector multiplications. Points in S^2 are represented by a vector $y \in \mathbf{R}^3$, $||y|| = 1$. The Lie algebra of $SO(3)$ is $\mathfrak{so}(3)$, the space of three-by-three skew-symmetric matrices.

From [22, Page 233] we have an example vector field, there given as a vector field in $\mathbf{R}^3$.

$$\dot{y} = \begin{pmatrix} \dot{y_1} \\ \dot{y_2} \\ \dot{y_3} \end{pmatrix} = \begin{pmatrix} -y_2 + y_1 y_3^2 \\ y_1 + y_2 y_3^2 \\ -y_3(y_1^2 + y_2^2) \end{pmatrix}. \tag{3.4}$$

We can easily check by insertion that $\frac{\mathrm{d}}{\mathrm{d}t}(y_1^2 + y_2^2 + y_3^2) = 0$ and thereby $\|y\|$ will be a constant for this flow. Given an initial value on S^2, the solution will evolve on S^2. The flow for initial values in $\mathbf{R}^3 \backslash S^2$ will be topologically equivalent to the flow on S^2.

To develop a Lie-Euler version we need the corresponding mapping from S^2 to the Lie algebra $\mathfrak{so}(3)$. We make use of the hat map $\mathbf{R}^3 \to \mathfrak{so}(3)$ to be defined in Definition 4.3 for further simplification. A vector field on S^2 may be represented as a cross-product $\omega \times y$ where both ω and y are taken as $\mathbf{R}^3$-vectors. The corresponding element in $\mathfrak{so}(3)$ will then be $\widehat{\omega}$.

To find ω we set up the system of equations

$$\omega \times y = \begin{pmatrix} \omega_2 y_3 - \omega_3 y_2 \\ \omega_3 y_1 - \omega_1 y_3 \\ \omega_1 y_2 - \omega_2 y_1 \end{pmatrix} = \begin{pmatrix} -y_2 + y_1 y_3^2 \\ y_1 + y_2 y_3^2 \\ -y_3(y_1^2 + y_2^2) \end{pmatrix}. \tag{3.5}$$

At each point y two of these equation will be linearly dependent ($T_y S^2$ is two-dimensional, so this is expected). To uniquely determine ω we enforce the additional constraint $\langle \omega, y \rangle = 0$ which gives the unique solution

$$\begin{aligned} \omega_1 &= -y_3(y_1 + y_2) \\ \omega_2 &= y_3(y_1 - y_2) \\ \omega_3 &= y_1^2 + y_2^2 \end{aligned} \tag{3.6}$$

Referring back to Equation (3.2) we have now provided the upper mapping from the manifold to the algebra, through the ω and its hat version $\widehat{\omega}$. The time step on the algebra is a straightforward Euler-step, and we step back to the manifold just as in Equation (3.2).

Algorithm 1 Lie-Euler on the sphere

1: Given y_0.
2: **for** $n = 1$ to $n = N$ **do**
3: $\quad y_n = \exp(\widehat{\omega}(y_{n-1}) y_{n-1}$
4: **end for**

Applied to our vector field from Equations (3.4) and (3.6) to a starting point near $(0, 0, 1)^T$ (a repulsive equilibria) we get the numerical solutions depicted in Figure 3.1.

From the figure and the proposition below we see that the Lie-Euler flow stays (in fact by machine accuracy) on the sphere, while the Euler spirals outwards. In time usage, the Lie-Euler is in fact roughly 10% faster than Euler when calculated in Matlab 6 for the same number of time steps.

Proposition 3.3. *The solution obtained by the Lie-Euler algorithm on the sphere will stay exactly on the manifold M.*

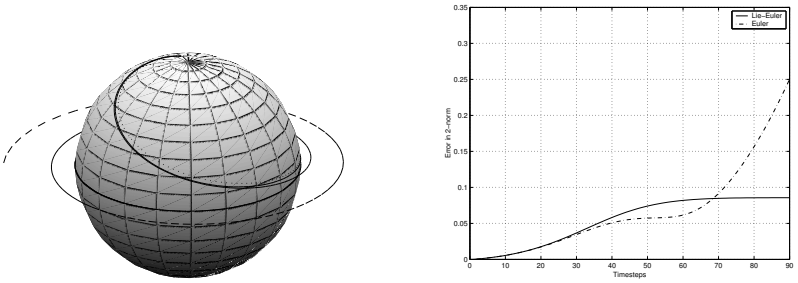


Figure 3.1: Comparison of Euler and Lie-Euler. Lie-Euler plotted in solid, Euler dash-dot and exact solution dotted.

Proof. Let $y_n \in S^2$, then $\|y_n\| = 1$. y_{n+1} is obtained by multiplying from left by a matrix $A \in SO(3)$. We should have $\|y_{n+1}\| = \|Ay_n\| = 1$. This is so because all orthogonal matrices preserve lengths of vectors by Proposition 2.12.

□

Lie-Euler was so simple that we did not need to pay attention to any more details than we did, although we did skip some fundamental parts. For developing higher-order Runge-Kutta methods we need to do more analysis to determine how the time stepping on the Lie algebra can be found. We will follow the lead of Munthe-Kaas [19], and also inspired by [9, 14, 13].

3.3 The differential equation on the Lie algebra

Our starting point is the differential equation on the manifold M which we want to solve.

$$y' = F(y), \quad y(0) = y_0, \quad y(t) \in \mathbf{R}^n.$$

The aim for the Lie group methods of Munthe-Kaas is to find a differential equation

$$\dot{u}(t) = \tilde{f}(u(t))$$

which has a flow on $\mathfrak{g}$ which again reproduces the flow of our original equation on M by use of the Lie algebra action λ_p.

For this, we construct the diagram

$$\begin{array}{ccccc} T\mathfrak{g} & \xrightarrow{T\lambda_p} & TM & & \\ \uparrow{\scriptstyle \tilde{f}} & & \uparrow{\scriptstyle F} & & \\ \mathfrak{g} & \xrightarrow{\tilde{\phi}_h} & \mathfrak{g} & \xrightarrow{\lambda_p} & M \\ & \searrow^{\lambda_p} & & \nearrow_{\phi_h} & \\ & & M & & \end{array} \tag{3.7}$$

We require the flow $\tilde{\phi}_h$ to reproduce the original flow ϕ_h, that is we require the horizontal part of Diagram (3.7) to be commutative,

$$\lambda_p \circ \tilde{\phi}_h = \phi_h \circ \lambda_p \tag{3.8}$$

Differentiating this requirement with respect to time we are left with

$$T\lambda_p \circ \tilde{f} = F \circ \lambda_p \tag{3.9}$$

which becomes the requirement for the $\tilde{f}$, and which is the commutativity of the vertical part of Diagram (3.7). Equation (3.9) is also called λ_p-relatedness of the vector fields $\tilde{f}$ and F, and is denoted $\tilde{f} \sim_{\lambda_p} F$.

Before we can use Equation (3.9) to find $\tilde{f}$ we need to sort out some details on the Lie algebra action

$$\lambda_p = \Lambda_p \circ \exp\colon \mathfrak{g} \longrightarrow M.$$

The tangent version appearing in Diagram (3.7) becomes by the chain rule

$$\begin{aligned} T\lambda_p &= T\Lambda_p \circ T\exp\colon T\mathfrak{g} \longrightarrow TM. \\ T_u\lambda_p &= T_{\exp(u)}\Lambda_p \circ T_u\exp\colon T_u\mathfrak{g} \longrightarrow T_{\lambda_p(u)}M \quad \text{pointwise} \end{aligned}$$

For reasons to be clear later we like to split $T\exp$ into two factors as in the diagram

$$\begin{array}{ccc} & T_{id}G \cong \mathfrak{g} & \\ {\scriptstyle d\exp_u}\nearrow & & \searrow{\scriptstyle T_{id}R_{\exp(u)}} \\ T_u\mathfrak{g} & \xrightarrow[T_u\exp]{} & T_{\exp(u)}G \end{array} \tag{3.10}$$

where $d\exp$ is called a right trivialization of $T\exp$. The motivation for this splitting is the parallelizable property of all Lie groups, making their tangent spaces trivial. In the following sections explicit expressions for $d\exp$ and its inverse $d\exp^{-1}$ will be developed.

The construction of an $\tilde{f}$ in Equation (3.9) relies on an assumption on F being representable by an $f\colon M \to \mathfrak{g}$ through a Lie algebra homomorphism

$$\lambda_*\colon \mathfrak{g} \times M \longrightarrow TM$$

defined as

$$\lambda_*(u)(p) = \left.\frac{\mathrm{d}}{\mathrm{d}t}\right|_{t=0} \Lambda(\exp(tu), p) \in T_pM. \tag{3.11}$$

We are now ready for the theorem characterizing the differential equation on $\mathfrak{g}$.

Theorem 3.4. *The differential equation on $\mathfrak{g}$ that will have a flow equivariant with the flow on M is*

$$\frac{\mathrm{d}u}{\mathrm{d}t} = d\exp_u^{-1}(f \circ \lambda_p(u(t))) \tag{3.12}$$

and where $f\colon M \to \mathfrak{g}$ is such that $\lambda_*(f(p))(p) = F(p), p \in M$ and $\lambda_p\colon \mathfrak{g} \to M$ is the Lie algebra action.

Proof. We need to prove that this choice of $\tilde{f}$ satisfies λ_p-relatedness.

To prove Equation (3.9) we start from the left and in the point $u \in \mathfrak{g}$

$$\begin{aligned} T_{\tilde{f}(u)}\lambda_p \circ \tilde{f}(u) &= T_{\exp(u)}\Lambda_p \circ T_u\exp(u) \circ \tilde{f}(u) \\ &= T_{\exp(u)}\Lambda_p \circ T_{id}R_{\exp(u)} \circ d\exp_u \circ \tilde{f}(u) \\ &= T_{\exp(u)}\Lambda_p \circ T_{id}R_{\exp(u)} \circ d\exp_u \circ d\exp_u^{-1} \circ f \circ \lambda_p. \end{aligned} \tag{3.13}$$

where we inserted the hypothesis of the theorem.

Now for the right side we use the assumption on F being representable by a Lie algebra homomorphism, λ_*, defined in Equation (3.11)

$$\begin{aligned} F \circ \lambda_p(u) &= \lambda_* \left(f(\lambda_p(u))\right)(\lambda_p(u)) \\ &= \frac{\mathrm{d}}{\mathrm{d}t}\bigg|_{t=0} \Lambda\left(\exp(tf(\lambda_p(u))), \Lambda(\exp(u), p)\right) \\ &= \frac{\mathrm{d}}{\mathrm{d}t}\bigg|_{t=0} \Lambda\left(\exp(tf(\lambda_p(u)))\exp(u), p\right) \\ &= T_{\exp(u)}\Lambda_p \circ T_{id}R_{\exp(u)} \circ f \circ \lambda_p(u). \end{aligned}$$

which is equal to the last line of Equation (3.13). □

For non-autonomous equations on M we should replace the time-dependent variable in Equation (3.12) by another letter, say s (because it will not be equal to the current time in the flow on M, s will be restarted for each step). We would then get

$$\tilde{f}(t, u(s)) = d\exp_u^{-1}(f(t, \lambda_{p(t)}(u)))$$

3.4 The *d*exp map and its inverse

In order to use Theorem 3.4 we need to have an expression for *d*exp and its inverse. This requires a significant amount of theory which we will delve into now. We will mainly follow the lecture notes of Brynjulf Owren [21] and some theory from [14].

We recall that the exponential map maps from a Lie group's tangent space at identity to the Lie group (locally):

$$\exp\colon T_{id}G \longrightarrow G$$

where $T_{id}G = \mathfrak{g}$. The tangent lift of this mapping then becomes

$$\begin{aligned} T\exp\colon T\mathfrak{g} &\longrightarrow TG \\ T_u\exp\colon T_u\mathfrak{g} &\longrightarrow T_{\exp(u)}G \quad \text{pointwise.} \end{aligned}$$

The notation *d*exp scarcely introduced in Lemma 3.4 is a (right) trivialization of $T\exp$. $T_u\exp$ splits as in Diagram (3.10).

Definition 3.5 (The Adjoint representation)**.** *Let $g \in G$ and $\nu(t)$ a curve in G such that $\nu(0) = id$ and $\nu'(0) = v \in \mathfrak{g}$, then the Adjoint representation is defined as*

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

Sometimes the shorthand (abuse of) notation $\mathrm{Ad}_g(v) = gvg^{-1}$ will be used. The derivative of Ad with respect to the group element (the lowered index) is denoted by ad, we set

$$\mathrm{ad}_u(v) = \frac{\mathrm{d}}{\mathrm{d}s}\bigg|_{s=0} \mathrm{Ad}_{\mu(s)}(v)$$

where $\mu(0) = id$ and $\mu'(0) = u$.

Writing and calculating the expression for ad we get

$$\begin{aligned}\mathrm{ad}_u(v) &= \left.\frac{\partial^2}{\partial s \partial t}\right|_{s=t=0} \mu(s)\nu(t)\mu(-s) \\ &= \left.\frac{\partial}{\partial s}\right|_{s=0} \mu(s)\nu'(0)\mu(s) \\ &= \mu'(0)\nu(0)'\mu(0) - \mu(0)\nu'(0)\mu'(0) \\ &= uv - vu = [u,v]\end{aligned}$$

the standard commutator in the Lie algebra from Definition 2.11.

Powers of ad may be recursively defined as

$$\mathrm{ad}_u^k(v) = \mathrm{ad}_u(\mathrm{ad}_u^{k-1}(v))$$

so that for example $\mathrm{ad}_u^2(v) = [u,[u,v]]$. Inspired by the Taylor series for scalar exponentiation we set

$$\exp(\mathrm{ad}_u)(v) = \sum_{k=0}^{\infty} \frac{1}{k!}\mathrm{ad}_u^k(v) \tag{3.14}$$

Now we can relate Ad and exp.

Lemma 3.6.

$$\mathrm{Ad}_{\exp(u)}(v) = \exp(\mathrm{ad}_u)(v)$$

Proof. Extend both sides of the equation to curves in t, like

$$\begin{aligned}y_L(t) &= \mathrm{Ad}_{\exp(tu)}(v) \\ y_R(t) &= \exp(t\mathrm{ad}_u)(v)\end{aligned}$$

Differentiating with respect to t,

$$\begin{aligned}\frac{\mathrm{d}y_L}{\mathrm{d}t} &= \exp(tu)uv\exp(-tu) - \exp(tu)vu\exp(-tu) \\ &= \mathrm{ad}_u(y_L) = [u, y_L]. \\ \frac{\mathrm{d}y_R}{\mathrm{d}t} &= \mathrm{ad}_u(y_R).\end{aligned}$$

As $y_L(0) = v = y_R(0)$ both curves must be equal (for t in some $J \subset \mathbf{R}$) by uniqueness of solutions of first order differential equations. In particular $y_L(1) = y_R(1)$ which is what we were looking for. □

Lemma 3.7 (The derivative of the exponential mapping)**.** *The tangent mapping of* $\exp\colon \mathfrak{g} \to G$ *is* $T\exp$*. Applied to* $v \in T_u\mathfrak{g}$ *we have*

$$T_u\exp(v) = \left.\frac{\mathrm{d}}{\mathrm{d}s}\right|_{s=0} \exp(u+sv) = d\exp_u(v)\exp(u)$$

where

$$d\exp_u(v) = \int_0^1 \exp(r\,\mathrm{ad}_u)(v)\,\mathrm{d}r \tag{3.15}$$

Proof. Let $\mu(s)$ be a curve in $\mathfrak{g}$, where $\mu(0) = u$, $\mu'(0) = v$. Because $T_u\mathfrak{g} \cong \mathfrak{g}$ we may write $\mu(s) = u + sv$ as an example of such a curve. $\exp(\mu(s))$ becomes a curve in G, and the tangent mapping must then be

$$T_u \exp(v) = \left.\frac{\mathrm{d}}{\mathrm{d}s}\right|_{s=0} \exp(u + sv).$$

We now make a surface in $\mathfrak{g}$ by $(s,t) \mapsto t\mu(s) = t(u + sv)$, and define $g(s,t) = \exp(t(u + sv))$. Our expression for $T_u \exp(v)$ now becomes $T_u \exp(v) = \left.\frac{\mathrm{d}}{\mathrm{d}s}\right|_{s=0} g(s,1)$.

For s sufficiently close to 0 we have by Lie series that $g(s,t) = \exp(tu) + \mathcal{O}(s)$. Keeping s fixed we write $g_s(t) = g(s,t)$, and differentiating by time we get

$$\frac{\mathrm{d}}{\mathrm{d}t} g_s(t) = (u + sv)\exp(t(u + sv)) = (u + sv)g_s(t).$$

From here we obtain

$$\begin{aligned} \dot{g}_s - ug_s = svg_s &= sv\exp(tu) + \mathcal{O}(s^2) \\ \frac{\mathrm{d}}{\mathrm{d}t}(\exp(-tu)g_s) &= s\exp(-tu)v\exp(tu) + \mathcal{O}(s^2) \end{aligned}$$

which is recognized as a differential equation where $\exp(-tu)$ is an integrating factor, integrating both sides from 0 to t we get

$$\begin{aligned} \exp(-tu)g_s - id &= s\int_0^t \exp(-ru)v\exp(ru)\,\mathrm{d}r + \mathcal{O}(s^2) \\ g_s(t) &= \exp(tu) + s\int_0^t \exp(tu)\exp(-ru)v\exp(ru)\,\mathrm{d}r + \mathcal{O}(s^2) \\ &= \exp(tu) + s\int_0^t \exp(ru)v\exp(-ru)\exp(tu)\,\mathrm{d}r + \mathcal{O}(s^2) \end{aligned}$$

by the change of variables $t - r \mapsto r$. Introducing s as a variable again

$$g(s,1) = \exp(u) + s\int_0^1 \exp(ru)v\exp(-ru)\,\mathrm{d}r\exp(u) + \mathcal{O}(s^2)$$

and our definition for $T_u \exp(v)$ was

$$\left.\frac{\mathrm{d}}{\mathrm{d}s}\right|_{s=0} g(s,1) = \int_0^1 \exp(r\,\mathrm{ad}_u)(v)\,\mathrm{d}r\exp(u)$$

□

Using the analytic expansion of exp in powers of ad, Equation (3.14), we may integrate the expression from the above lemma

$$\begin{aligned} d\exp_u &= \int_0^1 \exp(r\,\mathrm{ad}_u)\,\mathrm{d}r \\ &= \int_0^1 \sum_{k=0}^{\infty} \frac{r^k}{k!}\mathrm{ad}_u^k\,\mathrm{d}r \\ &= \sum_{k=0}^{\infty} \frac{1}{(k+1)!}\mathrm{ad}_u^k. \end{aligned}$$

This is recognized as the Taylor series of the entire function

$$g(z) = \frac{\mathrm{e}^z - 1}{z} = \sum_{k=0}^{\infty} \frac{1}{(k+1)!} z^k.$$

Using Equation (3.14), we can write just as for $g(z)$

$$d\exp_u = \frac{\exp(\mathrm{ad}_u) - id}{\mathrm{ad}_u}$$

This means that $d\exp$ is analytic in ad, and we may invert $d\exp$ by just inverting the analytic expansion

$$\begin{aligned} h(z) &= \frac{1}{g(z)} = \frac{z}{\mathrm{e}^z - 1} \\ &= \frac{1}{\sum_{k=0}^{\infty} \frac{1}{(k+1)!} z^k} = \frac{1}{1 + \frac{z}{2!} + \frac{z^2}{3!} + \frac{z^3}{4!} + \cdots} \\ &= 1 - \frac{z}{2} + \frac{z^2}{12} - \frac{z^4}{720} + \frac{z^6}{30240} - \frac{z^8}{1209600} + \cdots \\ &= 1 - \frac{z}{2} + \sum_{k=1}^{\infty} \frac{\mathrm{B}_{2k}}{(2k)!} z^{2k} \end{aligned}$$

where B_{2k} are the Bernoulli numbers (odd Bernoulli numbers except B_1 are zero). Replacing z by ad again, we obtain the usable expression for $d\exp^{-1} \colon \mathfrak{g} \to \mathfrak{g}$

$$d\exp_u^{-1}(v) = 1 - \frac{\mathrm{ad}_u(v)}{2} + \sum_{k=1}^{\infty} \frac{\mathrm{B}_{2k}}{(2k)!} \mathrm{ad}_u^{2k}(v) \tag{3.16}$$

3.5 Runge-Kutta-Munthe-Kaas methods

We are now able to cater for examples for the time stepping procedure Φ_h in Diagram (3.2). We set the upper mapping $M \to \mathfrak{g}$ to map to $0 \in \mathfrak{g}$ for all y_n, and then perform a classical Runge-Kutta step in $\mathfrak{g}$of the differential equation in Theorem 3.4. The Runge-Kutta method is characterized by the Butcher-tableau

$$\begin{array}{c|ccc} c_1 & a_{11} & \cdots & a_{s1} \\ \vdots & \vdots & & \vdots \\ c_s & a_{1s} & \cdots & a_{ss} \\ \hline & b_1 & \cdots & b_s \end{array}$$

with conditions on the coefficients up to order p.

Definition 3.8. *The order of a time stepping procedure Φ_h on a manifold is p if for all functions $\psi \colon M \to \mathbf{R}$ we have*

$$\psi(\Phi_h(y_n)) - \psi(y(t_n + h)) = \mathcal{O}(h^{p+1}) \tag{3.17}$$

In order to use $d\exp^{-1}$ numerically, we need a truncated version of Equation (3.16), and give a slightly modified name for the truncated version

$$d\text{expinv}(u, v, p) = v - \frac{\text{ad}_u(v)}{2} + \sum_{i=2}^{p-2} \frac{\text{B}_i}{i!} \text{ad}_u^i(v) \tag{3.18}$$

Compared to Equation (3.16) we see that we are not in this formula using that all odd Bernoulli numbers are zero.

Applying the Runge-Kutta method given by the tableau above to the differential equation $u(t) = \tilde{f}(u(t))$ and using Theorem 3.4 we obtain the *Runge-Kutta-Munthe-Kaas* algorithm [19]

Algorithm 2 s-stage Runge-Kutta-Munthe-Kaas of order p

1: Given y_n, t_n, set $u_0 = 0 \in \mathfrak{g}$.
2: **for** $i = 1$ to s **do**
3: $\quad g_i = h \sum_{j=1}^{s} a_{ij} k_j$
4: $\quad k_i^* = f(t_n + c_i h, \lambda_{y_n}(g_i)) \in \mathfrak{g}$
5: $\quad k_i = \tilde{f}(t_n + c_i h, g_i) = d\text{expinv}(g_i, k_i^*, p) \in T_{g_i}\mathfrak{g}$
6: **end for**
7: $y_{n+1} = \lambda_{y_n} \left(h \sum_{j=1}^{s} b_j k_j \right) \in M$

Proposition 3.9. *A classical Runge-Kutta method of order p applied to the differential equation on the Lie algebra given by Theorem 3.4, yields an order p Runge-Kutta-Munthe-Kaas method.*

Proof. This follows from the fact that our mappings between $\mathfrak{g}$ and M are smooth. □

Note that we must ensure high enough order of our truncated $d\exp^{-1}$. We need order $p - 1$ on $d\exp^{-1}$ because its output will be multiplied once by h in line 7 of Algorithm 2.

Example 3.2 (First order RKMK)**.**
For a first order RKMK method we use a first order truncation of $d$expinv and a first order RK-method on $\mathfrak{g}$. That is Euler on $\mathfrak{g}$ and $d\text{expinv}(u, v, 0) = v$, Butcher tableau on the left and corresponding Lie-Euler algorithm on the right

$$\begin{array}{c|c} 0 & 0 \\ \hline & 1 \end{array}$$

1: Given y_n, t_n
2: $g_1 = 0$
3: $k_1^* = f(t_n, y_n)$
4: $k_1 = k_1^*$
5: $y_{n+1} = \lambda_{y_n}(hk_1) = \lambda_{y_n}(hf(t_n, y_n))$

Example 3.3 (Second order RKMK)**.**
A classic example of a second order Runge-Kutta method. We need one higher order of the $d$expinv approximation, but the expression is the same, $d\text{expinv}(u, v, 1) = v$,

$$\begin{array}{c|cc} 0 & & \\ 1 & 1 & \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}$$

1: Given y_n, t_n
2: $g_1 = 0$
3: $k_1^* = f(t_n, y_n)$
4: $k_1 = k_1^*$
5: $g_2 = hk_1$
6: $k_2^* = f(t_n + h, \lambda_{y_n}(g_2))$
7: $k_2 = k_2^*$
8: $y_{n+1} = \lambda_{y_n}\left(h\left(\frac{1}{2}k_1 + \frac{1}{2}k_2\right)\right)$

Example 3.4 (Third order RKMK (Heun))**.**
This is the first method in which we need commutators to correct $d\exp^{-1}$, $d\text{expinv}(u, v, 2) = v - \frac{\text{ad}_u(v)}{2}$.

$$\begin{array}{c|ccc} 0 & & & \\ \frac{1}{3} & \frac{1}{3} & & \\ \frac{2}{3} & 0 & \frac{2}{3} & \\ \hline & \frac{1}{4} & 0 & \frac{3}{4} \end{array}$$

1: Given y_n, t_n
2: $g_1 = 0$
3: $k_1^* = f(t_n, y_n)$
4: $k_1 = k_1^*$
5: $g_2 = \frac{h}{3}k_1$
6: $k_2^* = f(t_n + \frac{1}{3}h, \lambda_{y_n}(g_2))$
7: $k_2 = k_2^* - \frac{1}{2}\text{ad}_{g_2}(k_2^*)$
8: $g_3 = \frac{2}{3}k_2$
9: $k_3^* = f(t_n + \frac{2}{3}h, \lambda_{y_n}(g_3))$
10: $k_3 = k_3 - \frac{1}{2}\text{ad}_{g_3}(k_3^*)$
11: $y_{n+1} = \lambda_{y_n}\left(h\left(\frac{1}{4}k_1 + \frac{3}{4}k_3\right)\right)$

The cost of the above algorithm is dominated by the number of times λ is used. For these methods these numbers is 1, 2 and 3 respectively, thereby providing acceptable cost/accuracy relationships.

Chapter 4

Using isotropy to improve the solution

Chapter 3 defined the Lie group action. Some Lie group actions have the property of an *isotropy subgroup*. The aim of this chapter is to explain what this subgroup is, which problem it presents, and what use we can have from it in the construction of Lie group methods through a Proposition which we will use in the following chapters.

The matrix group $SO(3)$ is the example we use for explaining isotropy, as it has a nice geometric interpretation. This group is used in Chapter 6 for the rotation of a rigid body. In Chapter 7 we use a different matrix group, $SL(2)$, for the Lie group action, which does not have the same geometric interpretation, but to which still the theory presented here will apply.

4.1 Isotropy by example

Example 4.1.
Given the Lie group action

$$\Lambda\colon SO(3) \times S^2 \longrightarrow S^2 \tag{4.1}$$

manifested as a matrix-vector product.

The following is easily verified:

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \quad \text{for all } \theta \in \mathbf{R}$$

which means that for all matrices of this form, the action will not move the point $p = (1, 0, 0)^T$ in the manifold S^2, for arbitrary θ. The lower right 2×2-corner of the matrix is recognized as the group of rotations in $\mathbf{R}^2$, namely $SO(2)$. This $SO(2)$ is the isotropy subgroup at $p = (1, 0, 0)^T$ for this Lie group action.

Definition 4.1 (Isotropy subgroup)**.** *The isotropy (or stabilizer) subgroup of a Lie group action $\Lambda\colon G \times M \longrightarrow M$ is defined pointwise on the manifold as*

$$G_p = \{g \in G \mid \Lambda(g, p) = p\}$$

From the definition we easily verify that G_p has the necessary properties for being a group.

Definition 4.2 (Isotropy subalgebra)**.** *For each isotropy subgroup, there is a isotropy subalgebra associated to the Lie algebra action (Equation* (3.3)*) defined pointwise on the manifold as*

$$\mathfrak{g}_p = \{u \in \mathfrak{g} \mid \lambda(u, p) = 0\}$$

The isotropy subgroup from Example 4.1 has a corresponding subalgebra in $\mathfrak{so}(3)$, which is $\mathfrak{so}(2)$.

For the action (4.1) there is an aforementioned nice geometric interpretation of the isotropy. $SO(3)$ is the group of rotation in 3 dimensions, with three degrees of freedom. For rigid bodies, its orientation in space is fully described by a $SO(3)$ matrix together with three coordinates. When working with vectors in S^2 instead of rigid bodies, the vectors are invariant with respect to rotations around the vector itself. This type of rotation is the isotropy subgroup for this action, visualized in Figure 4.1.

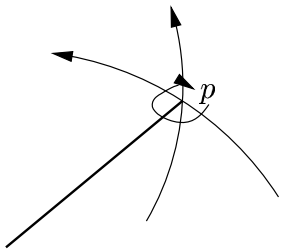


Figure 4.1: The effect of the three elements of $\mathfrak{so}(3)$ on S^2 through λ_p. The $\mathfrak{so}(3)$-element which yields rotation around p is the isotropy element $\widehat{p}$.

We are going to use $SO(3)$ and $\mathfrak{so}(3)$ a lot, so we make some details explicit. The basis for $\mathfrak{so}(3)$ is three skew-symmetric matrices, we set

$$e_1 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{pmatrix}, \quad e_2 = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{pmatrix} \quad e_3 = \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

Definition 4.3 (The hat map)**.** *The Lie algebra* $\mathfrak{so}(3)$ *of* $SO(3)$ *is isomorphic to* $\mathbf{R}^3$ *by the hat map:*

$$\widehat{\ }: \mathbf{R}^3 \longrightarrow \mathfrak{so}(3)$$
$$\begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix} \mapsto \begin{pmatrix} 0 & -v_3 & v_2 \\ v_3 & 0 & -v_1 \\ -v_2 & v_1 & 0 \end{pmatrix}$$

Note that the hat map is defined such that the isotropy algebra at a point $p \in S^2 \subset \mathbf{R}^3$ is simply $\widehat{p}$. It is natural to identify the vector itself with the isotropy subalgebra because of the geometric interpretation above.

4.2 Isotropy in rkmk algorithms

The last chapter introduced the Runge-Kutta-Munthe-Kaas algorithms. Theorem 3.4 assumes the existence of a map $f\colon M \to \mathfrak{g}$ such that $\lambda_*(f(p))(p) = F(p)$. But in the presence of isotropy, f is not uniquely determined by F, because

$$\lambda_*(f(p) + \zeta(p))(p) = \lambda_*(f(p))(p) + \underbrace{\lambda_*(\zeta(p))(p)}_{=0} = F(p) \tag{4.2}$$

if $\zeta(p) \in \mathfrak{g}_p$ for all $p \in M$. This is so because

$$\lambda_*(\zeta(p))(p) = \left.\frac{\mathrm{d}}{\mathrm{d}t}\right|_{t=0} \Lambda(\exp(t\zeta(p))p) = 0$$

by the definition of $\mathfrak{g}_m$. But

$$\Lambda(\exp(h(f(p) + \zeta(p))), p) \neq \Lambda(\exp(hf(p)), p)$$

so our *numerical* flow *will* be influenced by the choice of f or ζ.

By the way of Lie series, Section 2.4, it is possible to see the local role of isotropy for the Lie-Euler solver (Example 3.1 and 3.2). We assume here that we have a matrix acting on a manifold by matrix-products, and that our isotropy subalgebra is one-dimensional. Let $\zeta(p)$ be a basis element for the isotropy subalgebra at the point $p \in M$, and let $\sigma\colon M \to \mathbf{R}$ be a scalar function meant to be multiplied by ζ. Lie-Euler with isotropy-correction then becomes

$$y_1 = \exp(h(f(y_0) + \sigma(y_0)\zeta(y_0)))y_0$$

Order analysis for RK(MK) methods are found by comparing the Lie series expansion for this numerical solution to the exact solution for any function $\psi\colon M \to \mathbf{R}$. To avoid cluttering the notation, we employ the shorthand $f_* := \lambda_*(f(y_0))$ and similarly for ζ. The Lie series for the numerical solution is then

$$\begin{aligned}
\psi(y_1) &= \psi\left(\exp(h(f(y_0) + \sigma(y_0)\zeta(y_0)))y_0\right) \\
&= \psi(y_0) + h(f_* + \sigma\zeta_*)[\psi](y_0) + \frac{h^2}{2}(f_* + \sigma\zeta_*)^2[\psi](y_0) + \mathcal{O}(h^3) \\
&= \psi(y_0) + h(f_* + \sigma\zeta_*)[\psi](y_0) + \frac{h^2}{2}(f_*^2 + f_*\sigma\zeta_* + \sigma\zeta_* f_* + \sigma^2\zeta_*^2)[\psi](y_0) + \mathcal{O}(h^3) \\
&= \psi(y_0) + h(f_* + \sigma\zeta_*)[\psi](y_0) + \frac{h^2}{2}(f_*^2 + \sigma\zeta_* f_*)[\psi](y_0) + \mathcal{O}(h^3)
\end{aligned} \tag{4.3}$$

because $\zeta_*[\psi](y_0)$ is zero everywhere.

The Lie series for the exact flow $y(t)$, where $y(t_0) = y_0$, becomes

$$\psi(y(t_0 + h)) = \psi(y_0) + hF[\psi](y_0) + \frac{h^2}{2}F^2[\psi](y_0) + \mathcal{O}(h^3) \tag{4.4}$$

where

$$\begin{aligned} F^2[\psi](y_0) &= \left.\frac{\mathrm{d}}{\mathrm{d}h_2}\right|_{h_2=0} \left.\frac{\mathrm{d}}{\mathrm{d}h_1}\right|_{h_1=0} \psi(y(t_0+h_1+h_2)) \\ &= \left.\frac{\mathrm{d}}{\mathrm{d}h_2}\right|_{h_2=0} T\psi \circ F(y(t_0+h_2)) \\ &= \left.\frac{\mathrm{d}}{\mathrm{d}h_2}\right|_{h_2=0} T\psi \circ f(y(t_0+h_2))(y(t_0+h_2)) \\ &= TT\psi \circ \left(\frac{\mathrm{d}f(y(t))}{\mathrm{d}t} y(t) + f(y(t))^2 y(t)\right) \end{aligned} \tag{4.5}$$

This results in the main result on how we may use isotropy to improve the solution for Lie group methods with a one-dimensional isotropy subgroup. This result is what we are going to implement numerically the following applications

Proposition 4.4. *Lie-Euler may be raised to second order if there is a* $\sigma\colon M \to \mathbf{R}$ *such that*

$$\sigma(p)\zeta(p)f(p)p = \frac{\mathrm{d}f(p)}{\mathrm{d}t}p \tag{4.6}$$

for all $p \in M$.

Proof. Follows from the h^2-coefficient functions of Equation (4.3) and (4.5). □

Note that we have not said anything about the possibility of satisfaction of Equation (4.6). One strategy of choosing a σ will be based on minimizing the difference between the right and left side of the equation in a suitable norm. Chapter 6 will make additional notes about to which degree this equality is fulfilled, and what type of error is left.

Proposition 4.5. *The isotropy subalgebra at a point* $p = \Lambda(Q, I) = QI \in M$ *where* I *is any other point in* M*, is*

$$\mathrm{Ad}_Q(\zeta_I) = Q\zeta_I Q^{-1}$$

where ζ_I *is the isotropy subalgebra at the origin* I *of* M

Proof. We know that $\exp(\zeta_I)I = I$ by definition of ζ_I. We are working with matrices and use the matrix exponential:

$$\begin{aligned} \exp(Q\zeta_I Q^{-1})QI &= Q\exp(\zeta_I)Q^{-1}QI \\ &= Q\exp(\zeta_I)I \\ &= QI = p \end{aligned}$$

which proves the proposition. □

It is customary to define an origin in the manifold, denote it I and then find ζ_I. This proposition is not being used directly in the implementation of the $SO(3)$ and $SL(2)$ solvers to come, as we will use simpler ways to find $\zeta(p)$ for them.

4.3 Isotropy for actions on Stiefel manifolds

We started of by an example concerning the $SO(3) \times S^2 \to S^2$ action. As the generalization is simple, we generalize the S^2 manifold to a Stiefel manifold, and increase the dimension of the matrix group to $SO(n)$ accordingly.

Definition 4.6 (Stiefel manifold)**.** *A (n,k)-Stiefel manifold V_n^k is the set of all $n \times k$-matrices in which all columns are orthonormal and $k \leq n$, that is if $p \in V_n^k$ then*

$$p^T p = I_k$$

where I_k is the $k \times k$-identity matrix.

The S^2 manifold is the same as the Stiefel manifold V_3^1.
We use $SO(n)$ to act on V_n^k manifolds with arbitrary k, and define the origin in V_n^k as

$$I_{n,k} = \begin{pmatrix} I_k \\ 0 \end{pmatrix} \tag{4.7}$$

where 0 is a $(n-k) \times k$ null matrix. The isotropy subalgebra at this origin is

$$\zeta_{I_{n,k}} = \begin{pmatrix} 0 & 0 \\ 0 & C \end{pmatrix} \tag{4.8}$$

where $C \in \mathfrak{so}(n-k)$, which is easily verified. We use Proposition 4.5 to determine the isotropy subalgebra for any other point $p \in V_n^k$ as long as we know of a matrix $Q \in SO(n)$ such that $QI_{n,k} = p$.

M	$\dim V_n^k$	$\dim \mathfrak{so}(n)$	$\dim \mathfrak{g}_p$	M	$\dim V_n^k$	$\dim \mathfrak{so}(n)$	$\dim \mathfrak{g}_p$
V_1^1	0	0	0	V_5^1	4	10	6
V_2^1	1	1	0	V_5^2	7	10	3
V_2^2	1	1	0	V_5^3	9	10	1
V_3^1	2	3	1	V_5^4	10	10	0
V_3^2	3	3	0	V_5^5	10	10	0
V_3^3	3	3	0				
V_4^1	3	6	3				
V_4^2	5	6	1				
V_4^3	6	6	0				
V_4^4	6	6	0				

Table 4.1: Dimension of the isotropy subalgebra for various Stiefel manifolds

We will here make no other uses of the Stiefel manifold other than V_3^1 which is the sphere S^2. For a higher-dimensional isotropy, the analysis in Section 4.2 must be redone to cater for the additional basis elements of the isotropy subalgebra, and we will possibly be able to obtain condition for an order increase of more than one. Lewis and Olver has found conditions and constructed a method which is an improved second order method for the rigid body equations in [15]. We have chosen not to pursue this any further here.

Typical uses of the Stiefel manifold involves matrices with $n \gg k$, where $\dim \mathfrak{g}_m$ will be large. In these situations, the time of evaluation of the exponential map is dependent on n and not on the small k. Rather than trying to utilize this redundancy by adjusting the isotropy, a different approach has been taken in [6]. The authors have there created retractions to avoid use of the exponential map, and there is no isotropy left to play with. It remains to be seen whether anything can be done to improve algorithms on Stiefel manifolds using the full exponential map and taking advantage of the large isotropy group, and how this compares the retraction approach.

Chapter 5

Hamiltonian and Poisson systems

5.1 Hamiltonian systems

Hamiltonian systems are a large class of dynamical systems which often is used to describe mechanical problems in space. The first results appeared in 1834 by Hamilton, inspired by previous research in optics, and further developed by Jacobi, which connected Hamiltonian systems to partial differential equations. Hamiltonian theory has its feet in three domains, ordinary differential equations, which is where we will focus, but also in the theory of variational principles (Lagrange) and first order partial differential equations (Jacobi).

The presentation given here is mostly based on Chapter VI of [10] and the classical book by Arnold [2].

5.1.1 Lagrangian formulation

Joseph-Louis Lagrange introduced the variables $q = (q_1, \ldots, q_n)^T$ for any mechanical system, describing the positions and thereby the configuration manifold. These are called generalized coordinates and may be dependent on each other. In addition, he assumed an expression representing the *kinetic* energy of the system

$$T = T(q, \dot{q})$$

where $\dot{q}$ is the time-derivative of the coordinates (generalized velocities). Secondly, there is an expression for the *potential* energy for the system

$$U = U(q).$$

Lagrange then set

$$L(q, \dot{q}) = T(q, \dot{q}) - U(q) \tag{5.1}$$

to be the *Lagrangian* of the system. Denote the work functional of the Lagrangian to be the integral of L along a curve γ described by the q coordinates:

$$W(\gamma) = \int_{t_0}^{t_1} L(q(t), \dot{q}(t))\,\mathrm{d}t \tag{5.2}$$

Hamiltons principle of least action says that a solution of the mechanical system T and U describes, is a curve γ that minimizes $W(\gamma)$. That is we want to find an extremal of W. Finding this extremal is finding a curve such that the functional's differential is zero.

Theorem 5.1 (The Euler-Lagrange Equation)**.** *The curve $\gamma(t)$ is an extremal of the work functional $W(q)$ on the space of curves passing through $q(t_0)$ and $q(t_1)$ if and only if*

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial L}{\partial \dot{q}}\right) = \frac{\partial L}{\partial q} \tag{5.3}$$

Proof. The proof may be found on page 57 in [2]. □

5.1.2 Hamiltonian formulation

Hamilton introduced another variable in order to simplify the equations for mechanical systems, namely Poisson's conjugate momenta

$$p = \frac{\partial L}{\partial \dot{q}}(q, \dot{q}) \tag{5.4}$$

He then found that the system of ordinary differential equations

$$\begin{aligned} \dot{p} &= -\frac{\partial H}{\partial q} \\ \dot{q} &= \frac{\partial H}{\partial p} \end{aligned} \tag{5.5}$$

where $H(p, q)$ is the Hamiltonian

$$H(p, q) = p^T \dot{q} - L(q, \dot{q}). \tag{5.6}$$

is equivalent to the Euler-Lagrange equations (5.3). This is easily shown by differentiating H given by Equation (5.6) by p and q and using the definition of p.

Writing $y = (p, q)^T$ we may write Equation (5.5) as

$$\dot{y} = J^{-1}\nabla H(y), \quad \text{where } J = \begin{pmatrix} 0 & I \\ -I & 0 \end{pmatrix} \tag{5.7}$$

5.1.3 First integrals

First integrals of systems are quantities, functions $I\colon M \to \mathbf{R}$, which have their values conserved for all points along a solution path of a system. These may also be called *invariants* or *constants of motion*. Physical conservation laws are often expressed as first integral, such as energy preservation or conservation of angular momentum. We have already encountered a first integral in Example 3.1, where $\|y\|$ was a constant.

Definition 5.2. *Consider the differential equations $\dot{y} = f(y)$. A non-constant function $I(y)$ is called a first integral of the system if*

$$I'(y)f(y) = 0 \quad \text{for all } y \in M. \tag{5.8}$$

If a first integral of a system is known, this is effectively a constraint in the solution space in which the flow of the system must be at all times. Level curves of certain first integrals will sometimes be equivalent to the solution curves, as for the Lotka-Volterra system and the Duffing oscillator in Chapter 7.

For Hamiltonian systems, the Hamiltonian H is always a first integral. This follows from $H'(p, q) = (\partial H/\partial p, \partial H/\partial q)$ and

$$H'(p, q)J^{-1}\nabla H(p, q) = \frac{\partial H}{\partial p}\left(-\frac{\partial H}{\partial q}\right)^T + \frac{\partial H}{\partial q}\left(\frac{\partial H}{\partial p}\right)^T = 0.$$

5.1.4 Symplecticness

Symplecticness is an important property for linear mappings. It is a sort of area preservation which is proven to be important for Hamiltonian systems.

Definition 5.3. *A linear mapping $A\colon \mathbf{R}^{2n} \to \mathbf{R}^{2n}$ is called symplectic if*

$$A^T J A = J \tag{5.9}$$

where J is defined in Equation (5.7).

The flow of a system and the numerical integrators are differentiable mappings in h, for which we define symplecticness as follows

Definition 5.4. *A differentiable mapping $g\colon U \to \mathbf{R}^{2n}$ where U is open in $\mathbf{R}^{2n}$, is called symplectic if its Jacobian is everywhere symplectic.*

The crucial point regarding symplecticness is Poincaré's theorem from 1899:

Theorem 5.5. *The solution flow ϕ_t for a Hamiltonian system where H is at least twice continuously differentiable, is symplectic.*

Proof. See Hairer, Wanner and Lubich [10, Theorem VI.2.4] □

It now seems plausible that if our numerical integrator share this symplectic property with the corresponding exact flow, then the numerical solver will perform better. Indeed it does, but it requires the use of backward error analysis together with results from complex analysis to prove rigorously. In Section IX.8 of [10] it is proved that a symplectic integrator will constrain the *global* error of the Hamiltonian over exponentially long time intervals.

Examples of symplectic integrators are most notably Symplectic Euler, which we will use in Chapter 7, and its composition with its adjoint which results in the Störmer-Verlet scheme of order two.

5.2 Poisson systems

Rigid body dynamics and the Lotka-Volterra system are not Hamiltonian systems, but Poisson systems. Poisson systems are a generalization of Hamiltonian systems, where the crucial point is to let the matrix J be dependent on the current position y.

5.2.1 The structure of Poisson systems

Definition 5.6. *A differential system*

$$\dot{y} = B(y)\nabla H(y) \tag{5.10}$$

where H is a functional (still named the Hamiltonian) and the coefficient of the matrix $B(y)$ satisfy the equations

$$b_{ij}(y) = -b_{ji}(y) \qquad \text{for all } i, j \tag{5.11}$$

$$\sum_{l=1}^{n} \left(\frac{\partial b_{ij}(y)}{\partial y_l} b_{lk}(y) + \frac{\partial b_{jk}(y)}{\partial y_l} b_{li}(y) + \frac{\partial b_{ki}(y)}{\partial y_l} b_{lj}(y) \right) = 0 \quad \text{for all } i, j, k, \tag{5.12}$$

is a Poisson *system.*

The matrix $B(y)$ defines a structure of a *Poisson* bracket on the space of functions of the phase space of the system in question.

Definition 5.7 (Poisson bracket)**.** *The Poisson bracket corresponding to the matrix $B(y)$ with elements $b_{ij}(y)$ is defined as the operation sending the two smooth functions $F(p,q)$ and $G(p,q)$ to another function $\{F,G\}(p,q)$ as in*

$$\{F,G\}(y) = \sum_{i,j=1}^{n} \frac{\partial F(y)}{\partial y_i} b_{ij}(y) \frac{\partial G(y)}{\partial y_j}. \tag{5.13}$$

The Poisson bracket has a close connection to first integrals. Taking the Lie derivative of a function along the flow of a system, is equivalent to the Poisson bracket of the function and the Hamiltonian, that is, I is a first integral if and only if

$$\{I,H\} = 0.$$

The Poisson bracket also satisfies bilinearity, skew-symmetry and the Jacobi-identity. We are going to keep out of rephrasing too much details regarding Poisson systems, and rather focus on results relevant to our applications.

Any Poisson system may be transformed to a *canonical form* by a differentiable variable transformation, $z = \chi(y) = (P_i(y), Q_i(y), C_k(y))$. The Poisson system $\dot{y} = B(y)\nabla H(y)$ transforms to

$$\dot{z} = B_0 \nabla K(z) \qquad \text{with } B_0 = \begin{pmatrix} 0 & -I & 0 \\ I & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \tag{5.14}$$

where $K(z) = H(y)$. The functions $C_k(y)$ are here the Casimirs, which are a special kind of first integrals only dependent on the Poisson structure ($B(y)$). Writing $z = (p,q,c)$ the system above becomes

$$\begin{aligned} \dot{p} &= -K_q(p,q,c) \\ \dot{q} &= K_p(p,q,c) \\ \dot{c} &= 0 \end{aligned} \tag{5.15}$$

which is quite similar to how we defined Hamiltonian systems above. The Casimirs are here seen to be constant throughout time. The proof of the existence of this transformation is deep and covered in Section VII2.4 of [10]. It is mainly based on the Darboux-Lie Theorem.

5.2.2 Poisson maps

Many properties of Hamiltonian systems may be transformed into equivalent properties valid for Poisson systems. The symplectic property of flows Hamiltonian systems was seen to be vitally important for global behavior of the solution when using numerical integrators. The integrators had to be symplectic maps, and the corresponding property we would like our integrators to have for Poisson systems is Poisson maps.

Definition 5.8 (Poisson map)**.** *A transformation $\rho\colon U \to \mathbf{R}^n$, $U \subseteq \mathbf{R}^n$ is a* Poisson map *for a system determined by the structure matrix $B(y)$ if its Jacobian matrix satisfies*

$$\rho'(y) B(y) \rho'(y)^T = B(\rho(y)) \tag{5.16}$$

For Hamiltonian systems, the structure matrix is the constant matrix $B(y) = J^{-1}$, and the definition of Poisson map and Symplectic map is then equivalent.

We also have the analogue to Poincaré's theorem (Theorem 5.5) for Poisson systems:

Theorem 5.9. *The flow of a Poisson system is a Poisson map.*

Proof. This may be found on page 237 of [10] □

Because of this, numerical integrators for Poisson systems which are Poisson maps will also perform well just as symplectic integrators perform well on Hamiltonian systems. Note that the definition of a Poisson map is now dependent on the system in question, so there is little hope to develop integrators that are Poisson maps for all possible Poisson systems.

In Appendix B we will establish that Symplectic Euler is a Poisson integrator for the Lotka-Volterra system.

Chapter 6

Isotropy corrections for rigid body dynamics

Rigid body dynamics is the mechanical description of how rigid bodies (rigid means that there is a fixed distance between any two points in a body) move and rotate in space. We will focus here on a body to which there are no applied forces or torques. An example is a satellite in orbital motion around the earth. The satellite is under gravitational interaction, but is in free fall, thus we may model it as an object with no present forces or torques. After the satellite leaves the space shuttle which has brought it to space, it will probably have some initial rotation, and the satellite will afterwards rotate accordingly.

Rigid body dynamics is about rotations in three-dimensional space, for which we choose to apply the rotation group $SO(3)$ for a Lie group method. But the manifold S^2 which is the phase space of the rotation, is only two-dimensional, whereas $SO(3)$ is three-dimensional. The extra dimension is what we call the isotropy, and is what we are going to utilize for rigid body dynamics in this chapter.

Lewis and Olver have in [15] successfully developed an isotropy correction for the rigid body problem. We are in this chapter going to reproduce their result, and see that the outcome is identical to the result we found earlier on how to improve a solution by isotropy, Proposition 4.4. The contribution here is the replacement of analytical derivations of the vector fields by numerical differentiation, which will have the same behavior. Also it is noted that multiplying the isotropy by a constant further improves the numerical solution.

6.1 The Euler equations

Arnold describes in [2, Chapter 6] all details of the theory up to the equations of motion for rigid bodies which we will present here. Any object has moments of inertia. Our three-dimensional bodies will have three axes with three corresponding moments of inertia. For simplicity we may assume a transformed coordinate system so that the matrix of moments of inertia is diagonal, and to each axis named 1, 2 and 3, we assign the moments of inertia I_1, I_2 and I_3. Let Ω_i be the angular velocity around axis i. The Euler equations for angular

velocity are

$$\begin{aligned} I_1 \frac{\mathrm{d}\Omega_1}{\mathrm{d}t} &= (I_2 - I_3)\Omega_2\Omega_3 \\ I_2 \frac{\mathrm{d}\Omega_2}{\mathrm{d}t} &= (I_3 - I_1)\Omega_3\Omega_1 \\ I_3 \frac{\mathrm{d}\Omega_3}{\mathrm{d}t} &= (I_1 - I_2)\Omega_1\Omega_2 \end{aligned} \tag{6.1}$$

We may rephrase the Euler equations for angular velocity into the Euler equations for the angular momentum vector $m = (m_1, m_2, m_3)^T$, where $m_i = I_i\Omega_i$, and we get

$$\begin{pmatrix} \dot{m}_1 \\ \dot{m}_2 \\ \dot{m}_3 \end{pmatrix} = \begin{pmatrix} \frac{I_2 - I_3}{I_2 I_3} m_2 m_3 \\ \frac{I_3 - I_1}{I_3 I_1} m_3 m_1 \\ \frac{I_1 - I_2}{I_1 I_2} m_1 m_2 \end{pmatrix} = \begin{pmatrix} 0 & \frac{m_3}{I_3} & -\frac{m_2}{I_2} \\ -\frac{m_3}{I_3} & 0 & \frac{m_1}{I_1} \\ \frac{m_2}{I_2} & -\frac{m_1}{I_1} & 0 \end{pmatrix} \begin{pmatrix} m_1 \\ m_2 \\ m_3 \end{pmatrix} \tag{6.2}$$

Defining a Hamiltonian

$$H(m_1, m_2, m_3) = \frac{1}{2}\left(\frac{m_1^2}{I_1} + \frac{m_2^2}{I_2} + \frac{m_3^2}{I_3}\right) \tag{6.3}$$

this becomes a Poisson system, Section 5.2

$$\dot{m} = \begin{pmatrix} 0 & m_3 & -m_2 \\ -m_3 & 0 & m_1 \\ m_2 & -m_1 & 0 \end{pmatrix} \nabla H(m) \tag{6.4}$$

Using the hat map from Definition 4.3 we may write the Poisson structure matrix $B(m)$ as $B(m) = \widehat{m}$.

6.2 Invariants

The Casimirs of a Poisson system are one form of first integrals, which are constant throughout time. The Casimir of our rigid body equations is the conservation of angular momentum,

$$L(m) = m_1^2 + m_2^2 + m_3^2 \tag{6.5}$$

By use of Lie group methods ($SO(3)$), this invariant is automatically exactly conserved. The update operation in our integrators will be of the form $y_{n+1} = Ay_n$ where A is an orthogonal matrix, which ensures conservation of (6.5). This was also seen in Example 3.1.

The second invariant will be the Hamiltonian (6.3) itself. This is the same as the total energy of the system, which should be conserved as no forces were applied. Our goal is to see how this invariant may be conserved.

If the I_1, I_2 and I_3 are distinct we have a triaxial body, and the level curves of the Hamiltonian on the unit sphere uniquely determines the periodic paths of motion.

6.3 Known solvers

There are numerous papers with proposals of solvers for the rigid body problem. By use of the Darboux-Lie theorem the Poisson system may be transformed into a canonical form (a Hamiltonian system) to which a symplectic integrator may be applied. McLachlan and Scovel [16] and Reich [23] have done this independently. Hairer, Lubich and Wanner [10, Section VII.2] formulates this using the method RATTLE to a method of of order two which conserves both the above mentioned invariants to machine accuracy. The computational complexity involves the solution of a Riccati type equation and a linear problem in each step.

Buss [4] develops some new algorithms based on geometric understanding of the problem, and compares them to other well known rigid body solvers. These methods are heavily specialized to the rigid body equations for accuracy and efficiency.

The Lie group solvers is the approach we are going to follow in this text on utilization of isotropy. A reference on Lie group methods applied to rigid body dynamics (where isotropy is not considered) is [5].

6.4 Order conditions by Lie series expansion

We will here outline the corrected Lie-Euler algorithm for the rigid body as done by Lewis and Olver in [15, Section 3 and 4]. The notation used here is to a little extent different from Lewis and Olver.

Given a differential equation on the sphere S^2

$$\dot{m} = F(m) \qquad m \in S^2 \tag{6.6}$$

we would like to develop a Lie group solver. This needs the corresponding $f\colon M \to \mathfrak{g}$ which we here denote as

$$\omega\colon S^2 \longrightarrow \mathbf{R}^3 \quad \text{or as} \quad \widehat{\omega}\colon S^2 \longrightarrow \mathfrak{so}(3)$$

by the use of the hat map in Definition 4.3. ω and $\widehat{\omega}$ is related through

$$\omega(m) \times m = \widehat{\omega}(m)m$$

To simplify the notation, we will when appropriate suppress the argument m of ω and $\widehat{\omega}$. Just note that $\widehat{\omega}(m) \neq \widehat{\omega}m$.

$\widehat{\omega}$ is not uniquely given, as explained in Section 4.2. We therefore enforce the constraint $\langle \omega(m), m\rangle = 0$.

The exact flow ϕ_h will have a Lie series expansion as in Equation (4.4)

$$\psi(\phi_h(m)) = \psi(m) + h\widehat{\omega}m[\psi] + \frac{h^2}{2}\left(\widehat{\omega}^2 m + \dot{\widehat{\omega}}m\right)[\psi] + \mathcal{O}(h^3) \tag{6.7}$$

Lewis and Olver then introduce the orthonormal basis in $\mathbf{R}^3 \cong \mathfrak{so}(3)$

$$\left\{m, \frac{\omega}{v}, \frac{\omega \times m}{v}\right\} \tag{6.8}$$

where $v = \|\omega(m)\|$ is the normalizing factor. The objective now is to expand both the exact solution and the numerical solution in terms of this basis. This is nothing else than

a specialized way of dealing with the order conditions by standard Taylor expansions, but this special choice of basis makes it clear what role isotropy plays. Note that the isotropy subalgebra at a point m is m (or $\widehat{m}$), and is thus our first basiselement.

The ω-map and its derivatives with respect to time are written in the basis as

$$\omega^{(j)} = \frac{\mathrm{d}^j}{\mathrm{d}t^j}\omega = a_j m + b_j \frac{\omega}{v} + c_j \frac{\omega \times m}{v}, \qquad j = 0, 1, 2, \ldots \tag{6.9}$$

For doing cross-product calculations in the basis, this table of cross products of the basiselements is helpful

$\times$	e_1	e_2	e_3
e_1	0	e_3	$-e_2$
e_2	$-e_3$	0	e_1
e_3	e_2	$-e_1$	0

With respect to the basis, the exact flow (6.7) has the expansion (now suppressing the function $\psi\colon S^2 \to \mathbf{R}$ which plays no role here):

$$\phi_h(m) = \begin{pmatrix}1\\0\\0\end{pmatrix} + h\begin{pmatrix}0\\0\\-v\end{pmatrix} + \frac{h^2}{2}\begin{pmatrix}-v^2\\c_1\\-b_1\end{pmatrix} + \mathcal{O}(h^3) \tag{6.10}$$

We write any RKMK method characterized by a ξ as

$$\Phi_h^\xi = \exp(\widehat{\xi}(m,t))m, \qquad \text{where } \xi(m,t) = \sum_{j=1}^{\infty} \frac{h^j}{j!}\xi_j(m) \tag{6.11}$$

and $\xi_j = (\alpha_j, \beta_j, \gamma_j)^T$ in the basis (6.8).

The expansion of any such a RKMK method becomes

$$\begin{aligned}
\Phi_h^\xi(m) &= \left(id + \sum_{j=1}^{\infty} \frac{h^j}{j!}\widehat{\xi}_j(m) + \left(\sum_{j=1}^{\infty} \frac{h^j}{j!}\widehat{\xi}_j(m)\right)^2 + \cdots \right) m \\
&= \left(id + h\widehat{\xi}_1 + \frac{h^2}{2}\left(\widehat{\xi}_2 + \widehat{\xi}_1^2\right) + \cdots \right) m \\
&= \begin{pmatrix}1\\0\\0\end{pmatrix} + h\begin{pmatrix}0\\\gamma_1\\-\beta_1\end{pmatrix} + \frac{h^2}{2}\left[\begin{pmatrix}\alpha_2\\\beta_2\\\gamma_2\end{pmatrix} \times \begin{pmatrix}1\\0\\0\end{pmatrix} + \begin{pmatrix}\alpha_1\\\beta_1\\\gamma_1\end{pmatrix} \times \left(\begin{pmatrix}\alpha_1\\\beta_1\\\gamma_1\end{pmatrix} \times \begin{pmatrix}1\\0\\0\end{pmatrix}\right)\right] + \cdots \\
&= \begin{pmatrix}1\\0\\0\end{pmatrix} + h\begin{pmatrix}0\\\gamma_1\\-\beta_1\end{pmatrix} + \frac{h^2}{2}\begin{pmatrix}-\beta_1^2 - \gamma_1^2\\\alpha_1\beta_1 + \gamma_2\\\alpha_1\gamma_1 - \beta_2\end{pmatrix} + \mathcal{O}(h^3)
\end{aligned}$$

Comparing this result to the expansion of the exact flow (6.10) we have the order conditions up to order 2 in Table 6.1. Further order conditions may be found in the same manner. Conditions for order 3 are explicitly given in [15].

Order 1	$\beta_1 = v$
	$\gamma_1 = 0$
Order 2	$\beta_2 = b_1$
	$\gamma_2 = c_1 - \alpha_1 v$

Table 6.1: Order conditions for RKMK methods in the basis (6.8).

6.5 Orbit capture

A Lie-Euler solver is based on letting $\xi_1(m) = \omega(m)$ and $\xi_j = 0$ for $j \geq 2$ (when ω obeys $\langle \omega(m), m \rangle = 0$). If we intend to correct Lie-Euler by an isotropy correction of magnitude σ, this corresponds to choosing

$$\xi(m) = h\xi_1(m) = h(\omega(m) + \sigma m) = h \begin{pmatrix} \sigma \\ 1 \\ 0 \end{pmatrix} \tag{6.12}$$

in the basis (6.8). As $\gamma_2 = 0$ here, the second order condition for order 2 becomes

$$\sigma = \alpha_1 = \frac{c_1}{v} \tag{6.13}$$

This σ will change at every point m during integration, so we must expect to calculate the correction at every time step. The error in integration for the isotropy-corrected Lie-Euler with $\sigma = c_1/v$ is

$$\phi_h(m) - \Phi_h(m) = \frac{h^2}{2} \begin{pmatrix} 0 \\ 0 \\ -b_1 \end{pmatrix} + \mathcal{O}(h^3) \tag{6.14}$$

So we have that the error done by this (still first order) method is in the direction of the vector field $(\omega \times m)$. This means that the only difference from our first order corrected method and a true second order method is the speed of movement along the trajectory. By a reparametrization of time of the exact flow, we adjust the reparametrization such that the error $-b_1$ becomes zero, and thus have a true second order method up to this reparametrization. Lewis and Olver call this *orbit capture* and do some calculations for proving its validity. Lie-Euler with isotropy correction (6.13) thus have a *second order orbit capture*.

Recall from Proposition 4.4 which says how to correct Lie-Euler. Equation (4.6) translates to

$$\sigma \widehat{m}\widehat{\omega}m = \widehat{\dot{\omega}}m \tag{6.15}$$

written componentwise in the basis (6.8)

$$\sigma \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \times \left(\begin{pmatrix} 0 \\ v \\ 0 \end{pmatrix} \times \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \right) = \begin{pmatrix} a_1 \\ b_1 \\ c_1 \end{pmatrix} \times \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \tag{6.16}$$

$$\Leftrightarrow$$

$$\sigma \begin{pmatrix} 0 \\ v \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ c_1 \\ -b_1 \end{pmatrix} \tag{6.17}$$

for which we set

$$\sigma = \frac{c_1}{v} \tag{6.18}$$

the same as we just found above, Equation (6.13). The $-b_1$ reappears here, there is nothing we can do about it, as predicted by Equation (6.14).

6.6 Choosing σ for the rigid body problem

6.6.1 Exact differentiation

Lewis and Olver find an exact formula for c_1, which is what we need at each point. We have that

$$\sigma = \frac{c_1}{v} = \frac{\langle \dot{\omega}, \omega \times m \rangle}{v^2} = \frac{\langle \ddot{m}, m \times \dot{m} \rangle}{\|\dot{m}\|^2} \tag{6.19}$$

Inserting the differential equation

$$\dot{m} = F(m) = m \times \mathbb{I}^{-1} m \tag{6.20}$$

which is equivalent to Equation (6.2) when $\mathbb{I} = \mathrm{diag}(I_1, I_2, I_3)$, we get

$$\begin{aligned} \ddot{m} &= \dot{m} \times \mathbb{I}^{-1} m + m \times \mathbb{I}^{-1} \dot{m} \\ &= F(m) \times \mathbb{I}^{-1} m + m \times \mathbb{I}^{-1} F(m) \end{aligned} \tag{6.21}$$

Using this, we find

$$\begin{aligned} \langle \ddot{m}, m \times \dot{m} \rangle &= \langle F(m) \times \mathbb{I}^{-1} m + m \times \mathbb{I}^{-1} F(m), m \times F(m) \rangle \\ &= \langle F(m), \mathbb{I}^{-1} F(m) \rangle - \langle m, \mathbb{I}^{-1} m \rangle \|F(m)\|^2 \end{aligned}$$

This yields the easily computable σ

$$\sigma(m) = \frac{\langle F(m), \mathbb{I}^{-1} F(m) \rangle}{\|F(m)\|^2} - \langle m, \mathbb{I}^{-1} m \rangle. \tag{6.22}$$

6.6.2 Numerical differentiation

The calculation done above may not always be available in all situations, so it is interesting to see what can be done without doing such explicit calculations of the derivative $\dot{\omega}$ and $\ddot{m}$. The remedy must be numerical differentiation. We propose a first order forward difference. Say we want to evaluate $\dot{\omega}$ at m,

$$\dot{\omega}(m) \approx \frac{\omega(\Phi_{\tilde{h}}(m)) - \omega(m)}{\tilde{h}} \tag{6.23}$$

where $\Phi_{\tilde{h}}$ steps forward using Lie-Euler with no isotropy correction with a possibly very small time step, typically $\tilde{h} \ll h$. $\tilde{h}$ may be chosen as small possible as long as no roundoff errors due to fixed machine precision occur. Experiments have shown that $\tilde{h} = \frac{h}{100}$ is adequate for the rigid body problem.

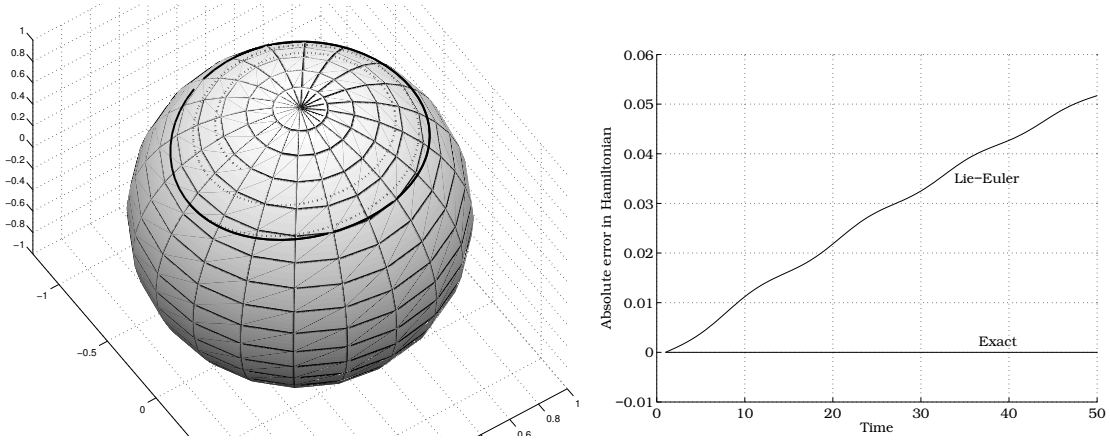
The drawback compared to the analytic derivation in Section 6.6.1 is increased computational time, as two Lie-Euler steps are required for every effective step. Note that for the Lotka-Volterra system and the Duffing oscillator in Chapter 7, it turned out to be sufficient to set $\tilde{h} = h$, thus removing the need for an extra step for each effective step.

6.7 Numerical results

For all the numerical results, we have used the inertia matrix $\mathbb{I} = \text{diag}(7, 5, 2)$, the starting point $y_0 = (0, -\sin(50°), \sin(50°))$ and time step $h = 0.5$.

6.7.1 Uncorrected Lie-Euler

Figure 6.1 shows how a standard implementation of Lie-Euler performs on the rigid body equations. The dotted line is heading steadily towards the equilibrium point on the top pole of the sphere. This linear drift away from the solution is typical for both an Euler solver and a Lie-Euler solver. The Hamiltonian in Figure 6.1(b) will eventually stabilize when the equilibrium is reached, soon after time 50.



(a) Plot of numerical solution on S^2.

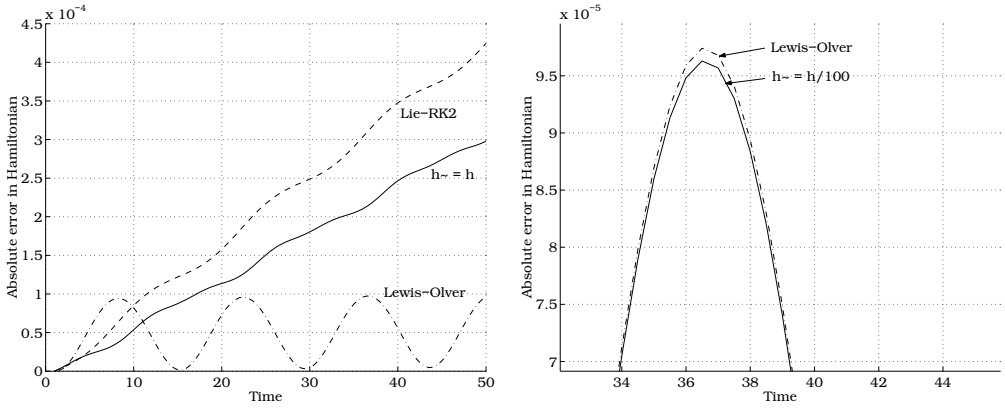
(b) Preservation of Hamiltonian

Figure 6.1: Failure of a standard application of Lie-Euler (dotted) with no isotropy correction compared to an exact solution (solid).

6.7.2 Isotropy corrected Lie-Euler

We now employ the isotropy correction from Section 6.6 to the Lie-Euler solver. This results in a considerably better conservation of the Hamiltonian, as seen in Figure 6.2(a) and 6.2(b). The left figure shows the result when the numerical differentiation is performed using the values already known, such that the extra step is unnecessary (there will be a problem with the very first step though). This plot shows that the approach using numerical differentiation is not quite good enough to match Lewis and Olver's estimate, but still outperforms the uncorrected Lie-Euler by two magnitudes and is slightly better than the second order RKMK method of Example 3.3.

By choosing $\tilde{h} = h/100$ as in Figure 6.2(b), the numerical differentiation is almost equivalent to the Lewis and Olver estimate. There seems to be little gain in having an even smaller $\tilde{h}$ and thereby a more correct $\dot{\omega}$ in terms of preservation of the Hamiltonian.



(a) Using $\tilde{h} = h$ in Equation 6.23.

(b) Using $\tilde{h} = h/100$. Zoomed plot.

Figure 6.2: Preservation of Hamiltonian for isotropy-corrected Lie-Euler, compared to a standard second order Lie group method.

6.7.3 Long time behavior

Our algorithms are not supposed to preserve the Hamiltonian exactly. This is seen in the Figure 6.3, where the time interval has been increased by an order of magnitude. The mean value of the Hamiltonian increases linearly by time (thick lines), and the method by numerical differentiation performs slightly better, although this difference is negligible (and is just a matter of luck because $\tilde{h}$ was not small enough to reproduce the derivative accurately enough). Looking at the magnitude of the Hamiltonian, we again see that the isotropy correction performs very well, taking into account the fatal introductory plot of the underlying Lie-Euler in Figure 6.1.

It is possible to stabilize this linear drift even further, by replacing σ by a scaled $\tilde{\sigma}$

$$\tilde{\sigma} = \alpha\, \sigma \tag{6.24}$$

where α is a constant independent of the point in the manifold. Smart values of α must be found by trial and error, or by "shooting". This has been done in Figure 6.4 below for an even longer period of time. We have currently not been able to find any mathematical rationale for doing this other than the numerical results in the figure.

6.7.4 Order plots

We have used the local Taylor expansion for the correcting isotropy term, and from Equation (6.14) we know that our method is still of order one. In terms of local error, our correction may only provide a better error coefficients. This is supported by the two plots in Figure 6.5 showing the local error of the basic Lie-Euler and the isotropy corrected Lie-Euler, compared to a RKMK implementation of a standard second order solver, Example 3.3. The isotropy correction yields a local error constant which is slightly better than the basic Lie-Euler, but this does in no way explain the large difference in global stability found in Figure 6.2.

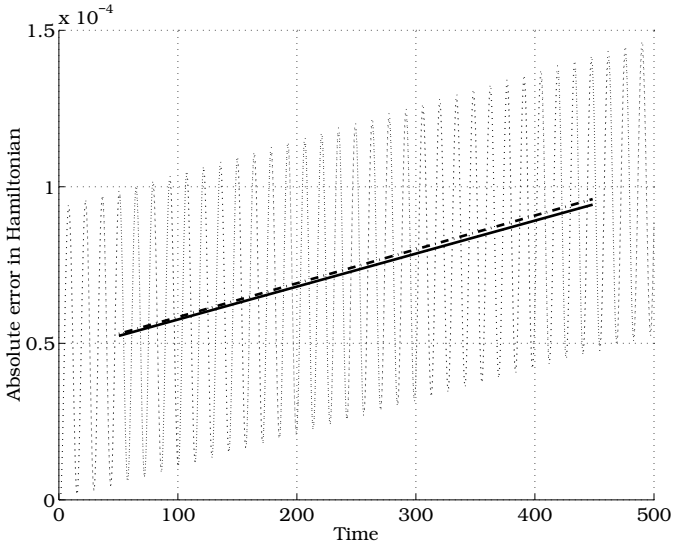


Figure 6.3: Long time behavior of the isotropy-corrected Lie-Euler. The thick line is the average of the rapidly oscillating Hamiltonian. Dash-dot is the correction estimate of Lewis and Olver, solid line is the correction estimate using Equation (6.23) with $\tilde{h} = h/100$. Time step 0.5.

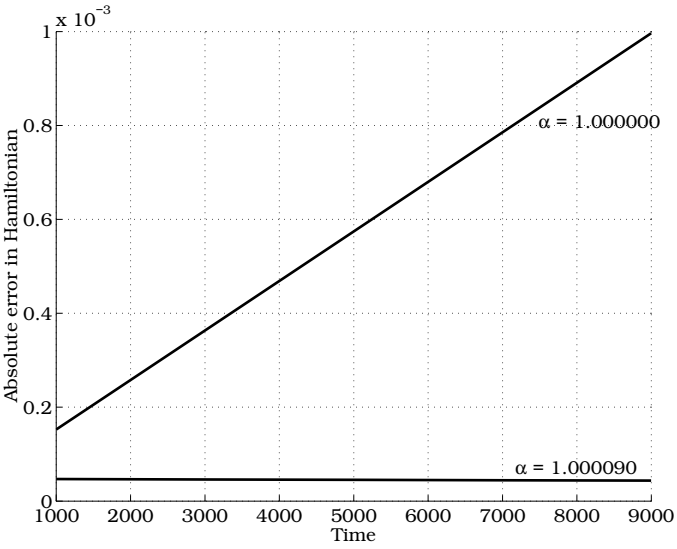
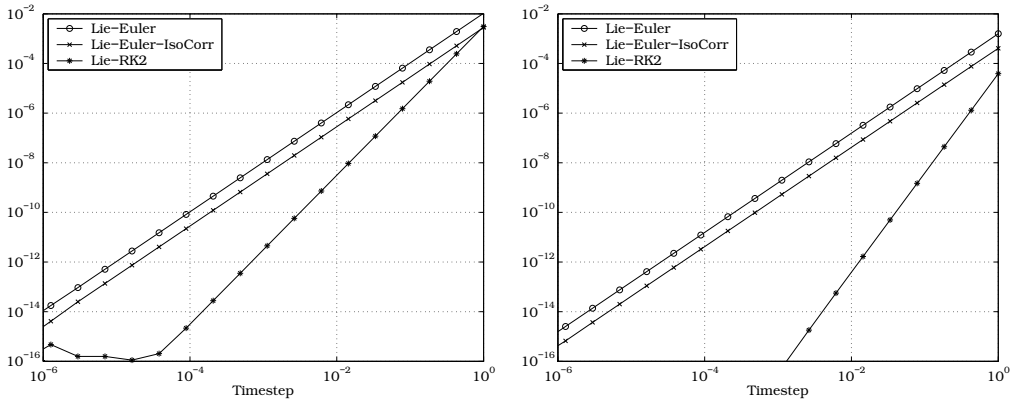


Figure 6.4: Further stabilization of the isotropy correction by scaling, $\tilde{\sigma} = \alpha\,\sigma$. Time step 0.5.



(a) The absolute error.

(b) Absolute error in Hamiltonian

Figure 6.5: Order plots for the uncorrected Lie-Euler and the corrected Lie-Euler.

6.8 Concluding remarks

We have seen that the isotropy corrected version of Lie Euler performs very well on the rigid body problem. The general approach using Lie series (Proposition 4.4) is equivalent to the specialized results by Lewis and Olver [15] in the basis (6.8). Although we used a condition for second order local behavior, we obtained global accuracy way better than our arbitrary second order method.

Even further stabilization of the corrected version was possible through a slight scaling of the isotropy correction, but this scaling can not currently be known a priori.

Chapter 7

Lie group methods for $\mathbf{R}^2$ based on $SL(2)$

Differential equations in $\mathbf{R}^2$ has never been the primary aim of Lie group methods. This configuration manifold is in the "domain" of the classical solvers. Using an $SL(2)$ action on $\mathbf{R}^2$ adds complexity. Comparing Euler and the Lie-version of it, Lie-Euler, we see that Lie-Euler is more demanding computationally, as it involves the exponentiation of an $\mathfrak{sl}(2)$-matrix. In addition, the formulation of Lie-Euler is not uniquely given, as the Lie group $SL(2)$ is of dimension three, whereas $\mathbf{R}^2$ is of dimension two. This extra dimension in $SL(2)$ is the isotropy, which we will use in the Lie-Euler solver for an improved numerical solution. The isotropy is the only reason we have for using $SL(2)$, as we do not gain any other qualitative attributes for free as we did in the rigid body problem (where the $SO(3)$-action ensured that our approximation stayed on the sphere).

The methods developed are applied to the Lotka-Volterra system, and a simplified Duffing oscillator. Lotka-Volterra has been the area of primary focus, and some analysis and experiments are not repeated for the Duffing oscillator. We will only be considering isotropy correction to first order solvers, that is Lie-Euler.

Throughout the chapter, we will use u and v for the coordinates in $\mathbf{R}^2$ and y will always be the vector $(u, v)^T$.

7.1 Using an $SL(2)$ action on $\mathbf{R}^2$

Instead of using affine transformations to move around in $\mathbf{R}^2$ which in this case works perfectly as opposed to the last chapter's S^2, we are going to consider a new approach, using the $SL(2)$ matrix group to act on the plane. The action

$$\Lambda\colon SL(2) \times \mathbf{R}^2 \longrightarrow \mathbf{R}^2 \tag{7.1}$$

manifested as matrix-vector multiplication, is transitive on the punctured plane $\mathbf{R}\backslash\{0\}$ because it is impossible to move to the point $(0, 0)$ by any matrix with determinant equal to 1.

Say we want to move from the point $p_0 = (1, 1)^T$ to $p_1 = (u, v)^T$, that is we want a matrix A such that $Ap_0 = p_1$. Solve

$$\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} 1 \\ 1 \end{pmatrix} = \begin{pmatrix} u \\ v \end{pmatrix}, \quad a_{11}a_{22} - a_{12}a_{21} = 1 \quad (A \in SL(2)) \tag{7.2}$$

which gives

$$A = \begin{pmatrix} \frac{1+uv-\gamma u}{v} & \frac{\gamma u-1}{v} \\ -\gamma + v & \gamma \end{pmatrix}, \quad \gamma \in \mathbf{R} \text{ arbitrary.} \tag{7.3}$$

which says that this works fine as long as $v \neq 0$. We can solve similarly if $v = 0$ but $u \neq 0$ and get a slightly different result. The difference between the results will always lie in the isotropy part, that is γ. Both u and v may not be zero, as that would imply that $(1, 1)^T$ is in the nullspace of A, but the nullspace of A is empty because it has determinant different from zero. We are not going to use Equation (7.3), it is only used to illustrate the concept. Note that γ plays the role of isotropy here. Our solution is a one-parameter subgroup of $SL(2)$, in which all group elements yields our desired transformation in $\mathbf{R}^2$.

7.1.1 The matrix exponential for $\mathfrak{sl}(2)$

The matrix exponential has a special form for the mapping $\mathfrak{sl}(2) \to SL(2)$. We develop simple formulas in Appendix A, and the result for a matrix $hS \in \mathfrak{sl}(2)$ (h is a scalar which we have included here for use in later results) is

$$\exp hS = \begin{cases} \cos h\sqrt{\det S}\, I + \frac{\sin h\sqrt{\det S}}{h\sqrt{\det S}} hS & \det S > 0 \\ \cosh h\sqrt{-\det S}\, I + \frac{\sinh h\sqrt{-\det S}}{h\sqrt{-\det S}} hS & \det S < 0 \\ I + hS & \det S = 0 \end{cases} \tag{7.4}$$

This splitting into three cases is because we have chosen to work with real numbers. For further analysis, we work with the series expansions of the above trigonometric and hyperbolic functions, which has identical expansions (Appendix A). The series expansion of the exponential becomes

$$\exp hS = I + hS - \frac{1}{2}\det(S)h^2 - \frac{1}{6}\det(S)Sh^3 + \mathcal{O}(h^3) \tag{7.5}$$

7.1.2 The isotropy subgroup

The isotropy subgroup may be found by solving similar to the above the equation

$$\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} u \\ v \end{pmatrix}, \qquad a_{11}a_{22} - a_{12}a_{21} = 1 \tag{7.6}$$

for which the solution is trivially found as a one-parameter family of solutions with the free parameter denoted by γ

$$SL(2)_y = \begin{pmatrix} 1 + \gamma uv & -\gamma u^2 \\ \gamma v^2 & 1 - \gamma uv \end{pmatrix}, \quad \gamma \in \mathbf{R}$$

Equation (7.6) is an inverse eigenvalue problem, we would like to find matrices in $SL(2)$ with eigenvalue 1 as a function of the eigenvector y.

7.1.3 The isotropy subalgebra

Similarly as the isotropy subgroup, we may find the isotropy subalgebra by solving

$$\begin{pmatrix} s_{11} & s_{12} \\ s_{21} & s_{22} \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \qquad s_{11} + s_{22} = 0 \tag{7.7}$$

with the solution

$$\sigma \begin{pmatrix} uv & -u^2 \\ v^2 & -uv \end{pmatrix} =: \sigma\zeta(y) \tag{7.8}$$

which is also an inverse eigenvalue problem. We searched for matrices with eigenvalue 0 as a function of the eigenvector. As 0 is an eigenvalue, the matrix is singular and the determinant is zero (which is easily verified).

From Equation (7.5) we see that the determinant is crucial for the exponentiation. It is easily seen that $\det(\zeta(y)) = 0$, and thus the exponential becomes as easy as

$$\exp(\sigma\zeta(y)) = I + \sigma\zeta(y) = \begin{pmatrix} 1 + uv\sigma & -\sigma u^2 \\ \sigma v^2 & 1 - uv\sigma \end{pmatrix}$$

equivalent to the isotropy subgroup we found above.

7.1.4 Constructing f for rkmk-methods

The construction of Runge-Kutta-Munthe-Kaas methods relies on a map $f\colon \mathbf{R}^2 \to \mathfrak{sl}(2)$ such that $\lambda_*(f(y))(y) = F(y)$ where $y = (u, v)^T$ and F comes from Equation (7.14). Since we are working with matrices, $\lambda_*(f(y))(y)$ is just the matrix-vector product $f(y)y$.

We should find functions such that

$$\begin{pmatrix} s_{11}(u,v) & s_{12}(u,v) \\ s_{21}(u,v) & s_{22}(u,v) \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} F_1(u,v) \\ F_2(u,v) \end{pmatrix} = F(u,v), \quad s_{11}(u,v) + s_{22}(u,v) = 0 \tag{7.9}$$

This equation has four unknowns and three constraints, so there is one degree of freedom (the isotropy). To develop general expressions for $f(y)$, we choose in succession $s_{21}(u, v) = 0$, $s_{12}(u, v) = 0$, and at last $s_{11}(u, v) = -s_{22}(u, v) = 0$. Straightforward algebraic manipulation of Equation (7.9) results in the three versions of f:

$$f_1 \begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} -\frac{F_2(u,v)}{v} & \frac{F_1(u,v)+F_2(u,v)\frac{u}{v}}{v} \\ 0 & \frac{F_2(u,v)}{v} \end{pmatrix} \tag{7.10}$$

if $s_{21}(u, v) = 0$. If $s_{12}(u, v) = 0$ we get

$$f_2 \begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} \frac{F_1(u,v)}{u} & 0 \\ \frac{F_2(u,v)+F_1(u,v)\frac{v}{u}}{u} & -\frac{F_1(u,v)}{u} \end{pmatrix} \tag{7.11}$$

and for the last choice, $s_{11}(u, v) = -s_{22}(u, v) = 0$, we get

$$f_3 \begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} 0 & \frac{F_1(u,v)}{v} \\ \frac{F_2(u,v)}{u} & 0 \end{pmatrix} \tag{7.12}$$

There is nothing fundamental about these three versions. They are all corresponding to different choices of the isotropy. For example, we have

$$f_1(u,v) + \underbrace{\frac{F_2(u,v)}{v}}_{\sigma} \zeta(u,v) = f_3(u,v).$$

Whichever of these versions we should apply for our RKMK-solver, is dependent on the functions F_1 and F_2 for the problem in question.

From Equation (7.5) on the exponential of $\mathfrak{sl}(2)$-matrices, we see that $\det(f(y))$ is crucial. We compute this for our first general result, Equation (7.10):

$$\begin{aligned}\det(f_1(y)+\sigma\zeta(y)) &= \begin{vmatrix} -\frac{F_2}{v}+\sigma & \frac{F_1+F_2\frac{u}{v}}{v}-\sigma\frac{u}{v} \\ \sigma\frac{v}{u} & \frac{F_2}{v}-\sigma \end{vmatrix} \\ &= -\left(\frac{F_2}{v}+\sigma\right)^2-\sigma\frac{v}{u}\left(\frac{F_1+F_2\frac{u}{v}}{u}-\sigma\frac{u}{v}\right) \\ &= -\frac{F_2^2}{v^2}+\sigma\left(\frac{F_2}{v}-\frac{F_1}{u}\right) \end{aligned} \tag{7.13}$$

It is now apparent that if $F_2/v - F_1/u = 0$, any isotropy correction will not play *any* role for the numerical result of a RKMK-solver. We will see that this is an obstacle for some of the following methods.

Calculating the determinant using f_2 or f_3 we obtain the same condition for where the isotropy has no effect.

7.2 The Lotka-Volterra model

The Lotka-Volterra equations is a model from mathematical biology describing the growth and decay of animal species. The Lotka-Volterra equations models two species, with the population $u(t)$ and $v(t)$ respectively. The rate of change for each of the species is assumed to be proportional to its population, and one specie eats the other one. Choosing constants as in [10, Section I.1.1] we arrive at the equations

$$\begin{aligned}\dot{u} &= u(v-2) \\ \dot{v} &= v(1-u)\end{aligned} \tag{7.14}$$

Dividing the first with the second yields

$$\begin{aligned}\frac{\dot{u}}{\dot{v}} &= \frac{u(v-1)}{v(1-u)} \\ \frac{1-u}{u}\dot{u} &= \frac{v-2}{v}\dot{v}\end{aligned} \tag{7.15}$$

which may now be integrated (separation of variables) to

$$\ln u - u = v - 2\ln v + C$$

where C is the constant from the integration. Along a solution path of (7.14) C must be constant. Pick a C, and this determines a solution curve, which may be implicitly plotted, as in Figure 7.1 below. Writing $I(u,v) = \ln u - u + 2\ln v - v$ we denote this the *invariant* for this system, in close resemblance to the invariants for Hamiltonian systems.

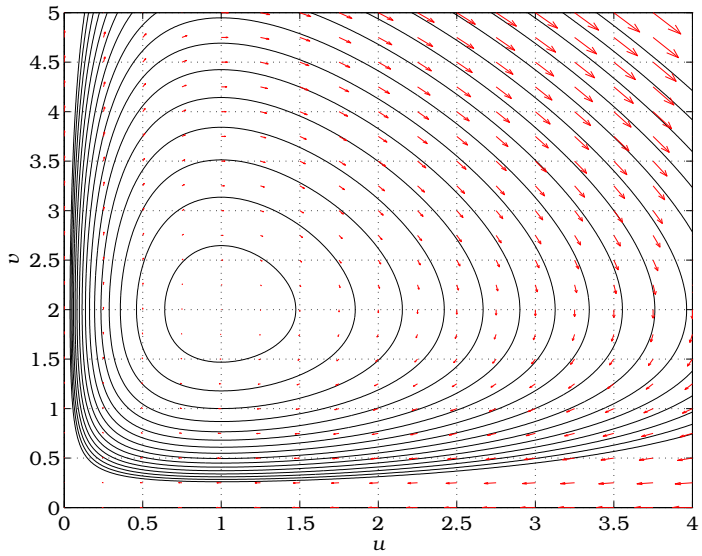


Figure 7.1: The periodic solutions of the Lotka-Volterra model (Equation (7.14)) and the vector field.

7.2.1 The Poisson structure of Lotka-Volterra

Referring back to Section 5.2, the Lotka-Volterra system may be set in the framework of Poisson systems as follows:

$$\dot{y} = \begin{pmatrix} \dot{u} \\ \dot{v} \end{pmatrix} = B(y)\nabla H(y) = \begin{pmatrix} 0 & uv \\ -uv & 0 \end{pmatrix} \begin{pmatrix} \frac{1}{u} - 1 \\ \frac{2}{v} - 1 \end{pmatrix} \tag{7.16}$$

where the Hamiltonian $H(y)$ for this Poisson system is our invariant $I(u, v)$. We note that the structure matrix $B(y)$ has the necessary skew-symmetric property and also what Equation (5.12) requires for the Jacobi identity to hold.

7.3 The Duffing oscillator

The Duffing oscillator is a model of the flexing of a beam of steel, acted upon by an electromagnet. The Duffing oscillator is described by the following differential equation

$$\ddot{x} + \alpha\dot{x} - x + x^3 = \beta\cos(\omega t) \tag{7.17}$$

We will only consider the case when $\alpha = 0$ and $\beta = 0$,

$$\ddot{x} - x + x^3 = 0 \tag{7.18}$$

which transformed to a system of first order differential equations in u and v becomes

$$\begin{aligned} \dot{u} &= v \\ \dot{v} &= u - u^3 \end{aligned}$$

This system may be integrated as we did for Lotka-Volterra, which results in an invariant, the Hamiltonian, for this Duffing oscillator,

$$H(u,v) = \frac{1}{2}(v^2 - u^2) + \frac{1}{4}u^4 \tag{7.19}$$

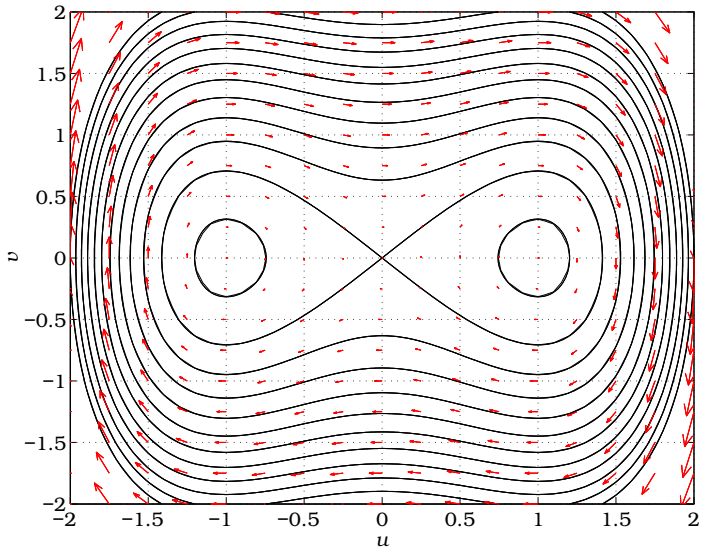


Figure 7.2: The periodic solutions of the Duffing oscillator (Equation (7.18)) and the associated vector field. The ∞-shaped separatrix is associated to a value of 0 for the Hamiltonian.

7.4 Basic methods

We implement two standard classical methods for the Lotka-Volterra system, and then our Lie group version of the Forward Euler method. The plain forward Euler method usually performs bad, accumulating error linearly over time. This is expected here as well. The performance of Lie-Euler (uncorrected) is unknown, but will perhaps perform roughly equivalent to Euler, which we will also see is true.

Symplectic Euler is on the other hand known to preserve the Hamiltonian (and thereby the trajectory of the solution) extremely well over long time periods.

7.4.1 Forward Euler

Forward Euler for the Lotka-Volterra system becomes explicitly

$$\begin{pmatrix} u_{n+1} \\ v_{n+1} \end{pmatrix} = \begin{pmatrix} u_n \\ v_n \end{pmatrix} + h \begin{pmatrix} u_n(v_n - 2) \\ v_n(1 - u_n) \end{pmatrix} \tag{7.20}$$

and for the Duffing oscillator

$$\begin{pmatrix} u_{n+1} \\ v_{n+1} \end{pmatrix} = \begin{pmatrix} u_n \\ v_n \end{pmatrix} + h \begin{pmatrix} v_n \\ u_n - u_n^3 \end{pmatrix} \tag{7.21}$$

7.4.2 Symplectic Euler

Symplectic Euler is a partitioned Euler method that treats the u-variable by implicit Euler and the v-variable by explicit Euler. Its name tells us this integrator is symplectic, and it will thereby perform well for the Hamiltonian Duffing oscillator, but in addition it is also a Poisson map (see Section 5.2.2 and Appendix B) which is what we need for the Lotka-Volterra system.

For systems with separable Hamiltonians like the Lotka-Volterra and the Duffing problem, the method has an explicit form. Lotka-Volterra is

$$\begin{aligned}\begin{pmatrix} u_{n+1} \\ v_{n+1} \end{pmatrix} &= \begin{pmatrix} u_n \\ v_n \end{pmatrix} + h \begin{pmatrix} u_{n+1}(v_n - 2) \\ v_n(1 - u_{n+1}) \end{pmatrix} \\ &= \begin{pmatrix} \frac{u_n}{1-h(v_n-2)} \\ v_n + hv_n(1 - u_{n+1}) \end{pmatrix}\end{aligned} \tag{7.22}$$

and for Duffing it is

$$\begin{pmatrix} u_{n+1} \\ v_{n+1} \end{pmatrix} = \begin{pmatrix} u_n \\ v_n \end{pmatrix} + h \begin{pmatrix} v_n \\ u_{n+1} - u_{n+1}^3 \end{pmatrix} \tag{7.23}$$

which both are explicit as long as u_{n+1} is calculated before v_{n+1}.

7.4.3 Lie-Euler

For the Lie-Euler solvers, we use the results from Section 7.1.4. The solution path of the Lotka-Volterra systems stays in the first quadrant (as long as the initial value is there), so we can choose whichever of the f-variants we want. We have chosen f_1 from Equation (7.10) which is undefined on the u-axis ($v = 0$) and arrive at

$$f_{1,LV}(y) = \begin{pmatrix} u - 1 & -\frac{u(u-v+1)}{v} \\ 0 & 1 - u \end{pmatrix} + \sigma\zeta(y) \tag{7.24}$$

where the linear isotropy part $\sigma\zeta(u, v)$ has been separated out.

For the Duffing oscillator, the simplest choice is f_3 from Equation (7.12), which becomes

$$f_{3,Duff}(y) = \begin{pmatrix} 0 & 1 \\ 1 - u^2 & 0 \end{pmatrix} \tag{7.25}$$

Using f_1 or f_2 leads to singularites on either the u- or v-axis. As the difference between f_1, f_2 and f_3 is only a matter of isotropy, these singularities should not really pose a problem. It is possible to use f_1 and f_2 for the Duffing oscillator if one employs switching between the two functions according to Figure 7.3. Numerically, either choice performs equivalent.

We define the "standard" Lie-Euler to be the method that uses these f's uncritically with regard to isotropy, that is using $\sigma = 0$, and the numerical Lie-Euler methods becomes

$$y_{n+1} = \exp[hf(y_n)]y_n \tag{7.26}$$

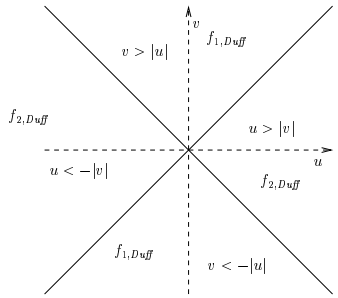


Figure 7.3: Strategy for choosing the correct version of $f\colon \mathbf{R}^2 \to \mathfrak{sl}(2)$ for the Duffing oscillator if only f_1 and f_2 are to be used, because of singularities. A remedy is to only use f_3, for which there is no problems of singularites. Both approaches have been tested numerically with equivalent results.

7.4.4 Lie-Euler with isotropy correction

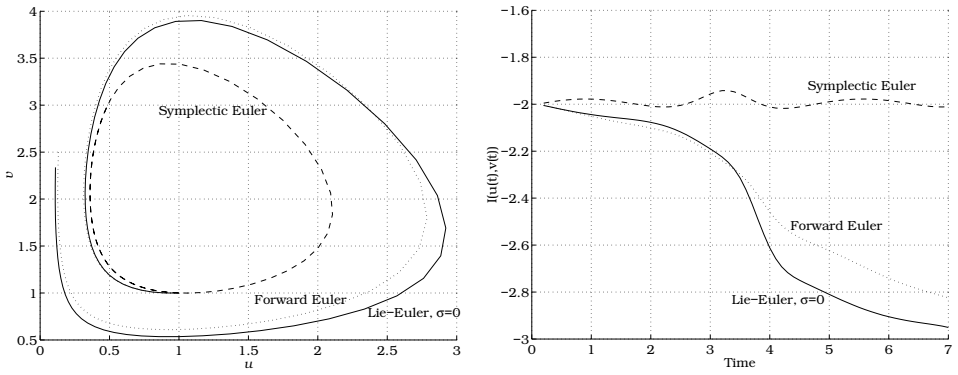
The isotropy correction is the usage of the value σ in Equations (7.24) and (7.25). The σ-value should be dependent on the position in the phase plane, so we have a function $\sigma\colon \mathbf{R}^2 \to \mathbf{R}$, and the isotropy corrected Lie-Euler becomes

$$y_{n+1} = \exp\left[h(f(y_n) + \sigma(y_n)\zeta(y_n))\right] y_n. \tag{7.27}$$

for the respective f's.

7.4.5 Introductory results

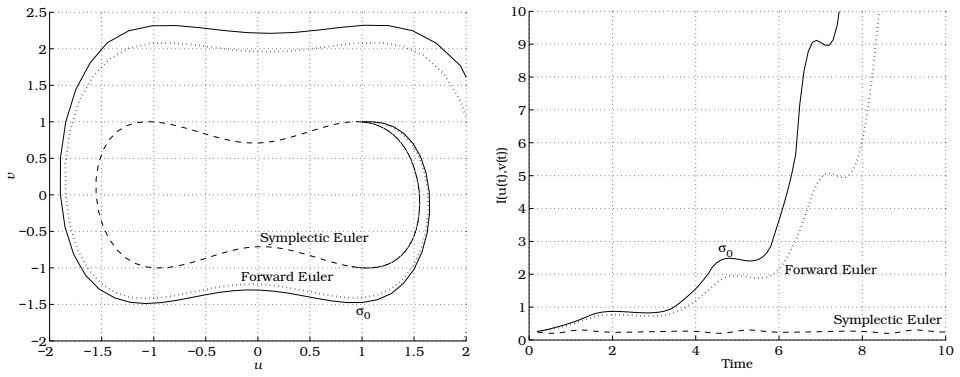
We first give a preliminary result to have an idea of what performance to expect. We see in Figure 7.4 for the Lotka-Volterra system is that letting $\sigma = 0$ yields a Lie-Euler comparable to Forward Euler, while both are totally outperformed by the Symplectic Euler method. Our goal will be to tweak σ such that Lie-Euler becomes comparable to Symplectic Euler. Quite similar behavior is observed for the Duffing oscillator in Figure 7.5.



(a) Plot of numerical trajectories.

(b) Invariant preservation.

Figure 7.4: Introductory results for Lotka-Volterra showing how Forward Euler and the Lie-Euler with $\sigma = 0$ are outperformed by Symplectic Euler. Time step 0.1.



(a) Plot of numerical trajectories.

(b) Invariant preservation.

Figure 7.5: Introductory results for the Duffing oscillator. As for Lotka-Volterra, Forward Euler and Lie-Euler with no isotropy correction is outperformed by Symplectic Euler.

7.5 Analysis of the isotropy corrected Lie-Euler method

7.5.1 Local expansion

As the exponential of $\mathfrak{sl}(2)$ matrices adopt for a very simple exact expression, we are able to perform some analysis and expansion to see the real effect of the isotropy-correction. We use the abbreviation $f_\sigma := f(y_n) + \sigma(y_n)\zeta(y_n)$. Using Equation (7.5) we easily find that Lie-Euler with isotropy correction, Equation (7.27), becomes

$$y_{n+1} = \left(I + hf(y_n) - \frac{1}{2}h^2 \det f_\sigma I - \frac{1}{6}h^3 \det f_\sigma \, f(y_n) + \mathcal{O}(h^4) \right) y_n \tag{7.28}$$

Note $f_\sigma y_n = f(y_n)y_n$ as used in the last line.

We use Equation (7.13) to find the expression for the determinant, for the Lotka-Volterra system

$$\det(f_{LV}(y) + \sigma\zeta(y)) = -(1-u)^2 + \sigma(3-u-v) \tag{7.29}$$

and for the Duffing oscillator

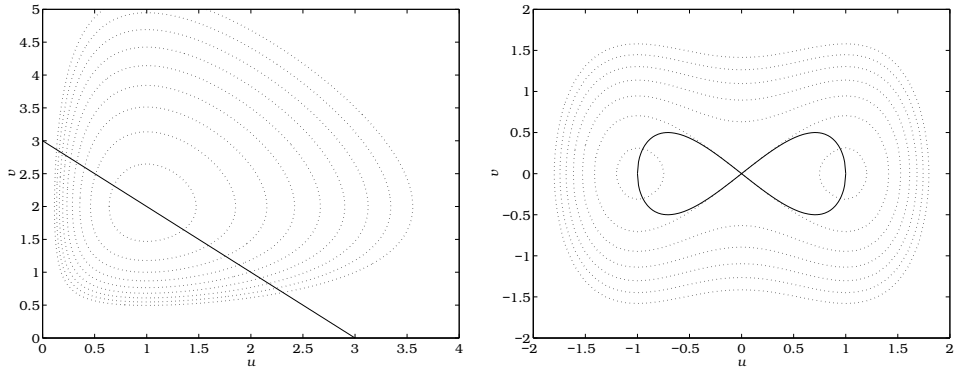
$$\det(f_{Duff}(y) + \sigma\zeta(y)) = -\frac{(u-u^3)^2}{v^2} + \sigma\left(\frac{u-u^3}{v} - \frac{v}{u}\right) \tag{7.30}$$

These expressions are important, as they tell us that there are points in which any choice of the isotropy has *no* effect. This happens if the coefficients of σ in Equation (7.29) or (7.30) become zero. For Lotka-Volterra this happens on the line $u + v = 3$, and for the Duffing oscillator on the implicitly given line $v^2 = u^2 - u^4$, see Figure 7.6 for plots.. These lines will always go through equilibrium points, easily seen from Equation (7.13).

For the Lotka-Volterra system, we are able to assign $\det f_\sigma$ any value we want (excluding the line $u + v = 3$), say D, because setting

$$\sigma = -\frac{u^2 - 2u + 1 + D}{u + v - 3} \quad \text{gives} \quad \det f_\sigma = D \tag{7.31}$$

Note that if we choose σ such that $\det f_\sigma = 0$ we are left with the Forward Euler algorithm, which has also been verified numerically. The same thing can also be done for the Duffing oscillator.



(a) Lotka-Volterra (b) Duffing oscillator

Figure 7.6: The points in the phase-space where the isotropy has no effect at all on the numerical solution. Actual solution trajectories dotted.

7.5.2 Backward error analysis

Up to now we have only considered local behavior. For our solvers we are searching for good *global* behavior. We have done local analysis and are hoping it will lead to stability of the solution and the invariant. One tool for analyzing global behavior is backward error analysis. We search for a modified vector field for which our numerical solution is an exact solution. This modified vector field is assumed to be of the form

$$\dot{\tilde{y}} = F_h(\tilde{y}) = F(\tilde{y}) + h\bar{F}_2(\tilde{y}) + h^2\bar{F}_3(\tilde{y}) + \cdots \tag{7.32}$$

where $\dot{y} = F(y)$ is the original equation and $\bar{F}_i$ are functions to be determined. Our (consistent) numerical method has the expansion

$$\Phi_h(y) = y + hF(y) + h^2 d_2(y) + h^3 d_3(y) + \cdots \tag{7.33}$$

which for us is the expansion we found in Equation (7.28). The trick is to set $\tilde{y}(t+h) = \Phi_h(y)$ and the results for $\bar{F}_i$ appear as

$$\begin{aligned}
\bar{F}_2(y) &= d_2(y) - \frac{1}{2!}F'F(y) \\
\bar{F}_3(y) &= d_3(y) - \frac{1}{3!}\left(F''(F,F)(y) + F'F'F(y)\right) - \frac{1}{2!}\left(F'\bar{F}_2(y) + \bar{F}_2'F(y)\right) \\
&\vdots
\end{aligned}$$

We refer to [10, Chapter IX] for details. Some work done by Maple results in the expression for $\bar{F}_2$ for the Lotka-Volterra system:

$$\bar{F}_2(u,v) = \begin{pmatrix} \frac{1}{2}u(v+u-3)\sigma + \frac{1}{2}u^3 - u^2 - \frac{1}{2}v^2u - \frac{3}{2}u + \frac{1}{2}vu^2 + \frac{3}{2}vu \\ \frac{1}{2}v(v+u-3)\sigma - vu + \frac{1}{2}v^2u \end{pmatrix} \tag{7.34}$$

where it is again clear that the role of isotropy disappears at the line $u + v = 3$. It is also clear that there cannot be any σ that makes $\bar{F}_2$ zero, if there were, we would have a second order method. We have not been able to conclude any further on how to use this result, but it is possible that it may be used as a tool to explain why scaling the isotropy by a constant makes for remarkable stability, as we are going to see later in Section 7.8.2.

7.6 Conservation of the Lotka-Volterra invariant

The first integral of the Lotka-Volterra model is the function

$$I(u,v) = \ln u - u + 2\ln v - v. \tag{7.35}$$

Every solution curve of the system follows a path where this value is preserved. Numerical solvers should also preserve this invariant to some extent.

We may see through Taylor expansions how our methods conserve this invariant. The complexity of the calculations is of such a degree that manual calculation is not recommendable, so here results from using the Taylor-function in Maple has been merely inserted.

Forward Euler has the expansion

$$\begin{aligned} I(u_{\mathrm{FE}}, v_{\mathrm{FE}}) - I(u,v) = &\left(-3 + 2v - \frac{1}{2}v^2 - u^2 + 2u\right) h^2 \\ &+ \left(-2 + 4v - 2v^2 + \frac{1}{3}v^3 - \frac{2}{3}u^3 + 2u^2 - 2u\right) h^3 + \mathcal{O}(h^4) \end{aligned} \tag{7.36}$$

Symplectic Euler has the expansion

$$\begin{aligned} I(u_{\mathrm{SE}}, v_{\mathrm{SE}}) - I(u,v) = &\left(1 - 2v + \frac{1}{2}v^2 - u^2 + 2u\right) h^2 \\ &+ \left(-2 + 2vu - \frac{2}{3}u^3 + \frac{1}{3}v^3 - 2v^2 - 2vu^2 + 6u^2 - 6u + 4v\right) h^3 + \mathcal{O}(h^4) \end{aligned} \tag{7.37}$$

Lie-Euler with isotropy correction σ. We set $\det f_\sigma = \beta$ for notational simplicity:

$$\begin{aligned} I(u_{\mathrm{LE}}, v_{\mathrm{LE}}) - I(u,v) = &\left(-\frac{3}{2}\beta - u^2 + 2u - 3 + 2v - \frac{1}{2}v^2 + \frac{1}{2}\beta u + \frac{1}{2}\beta v\right) h^2 \\ &+ \left(\frac{1}{2}\beta v - \beta u - 2 + 4v - 2v^2 + \frac{1}{2}v^3 - 2u + 2u^2 - \frac{2}{3}u^3\right) h^3 + \mathcal{O}(h^4) \end{aligned} \tag{7.38}$$

These expansions predict that all the three solvers are "equal" in their invariant preservation performance, as they all preserve the invariant to first order. The secret in the success of Symplectic Euler lies in the long-time behavior. This is shown in Table 7.1 where the coefficients of Forward Euler grows drastically while the coefficients of Symplectic Euler averages to around zero for integration over many periods. The time-value 26.1 is chosen because it is the last time-value before Forward Euler collapses, sending the point corresponding to 26.2 outside our quadrant and the coefficients soon go to infinity. The negativity of the Forward Euler coefficients tells us that the invariant is decreasing, meaning an outwards spiraling.

7.7 Strategies for choosing σ

7.7.1 Minimizing Lie-series error expansion by numerical differentiation

Chapter 4 presents a general way of improving the accuracy of Lie-Euler, through the Lie-series expansion of the error. The requirement for raising the order of Lie-Euler to two is given in Proposition 4.4. We cannot fulfill this requirement, so we will not be able to get

Integration length	1	10	26.1	100	1000	10000
Forward Euler h^2-coeff.	-0.558	-2.66	-13.8			
Forward Euler h^3-coeff.	-0.184	-0.557	70.3			
Symplectic Euler h^2-coeff.	0.228	0.0205	0.00863	-0.001251	0.000119	0.000101
Symplectic Euler h^3-coeff.	-0.515	0.0575	0.105	0.0922	0.0970	0.0971

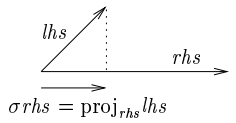
Table 7.1: Arithmetic mean values for the invariant expansion coefficients for Forward and Symplectic Euler. Time step 0.1. One period in the system has time-length of approximately 5.

an order two method, but it could possibly perform better than Lie-Euler with no isotropy correction.

Proposition 4.4 requires the derivative of the mapping $f\colon \mathbf{R}^2 \to \mathfrak{sl}(2)$. We have two options for this, either explicitly calculating the derivative of our f, or doing numerical differentiation. We opt for numerical differentiation, as experiments show it is sufficient, and use a first order backwards difference.

$$\sigma\zeta(y_n)f(y_n)y_n = \frac{f(y_n) - f(y_{n-1})}{h}y_n \tag{7.39}$$

This is an equation in $\mathbf{R}^2$, and we would like to adjust σ so that the two vectors become as close as possible. We do this by projection. There are two alternatives, either project the right hand side vector from the left hand side vector or the other way around. Define the vectors $lhs(y_n) = \zeta(y_n)f(y_n)y_n$ and $rhs(y_n) = \frac{f(y_n)-f(y_{n-1})}{h}y_n$. We choose to project the left hand side vector down to the right hand side vector and then equating to find a suitable σ, as illustrated in the figure to the right. Equivalent behavior occurs if we choose the other way around.



Proposition 4.4 mentions that "Lie-Euler may be raised to second order". This will happen if the two vectors *lhs* and *rhs* happen to be linearly dependent, which we cannot expect them to be in general. Because of this, we resort to a projection, assuming that the best use of isotropy is the one that mimics second-order behavior as close as possible.

$$\boxed{\sigma_{Diff}(y) = \frac{lhs(y)^T rhs(y)}{lhs(y)^T lhs(y)}} \tag{7.40}$$

Choosing this isotropy correction leads to better results than no isotropy correction at all, but it is not stable for long time integration on the Lotka-Volterra system. This is the correction used for the rigid body problem in Chapter 6 where it also performed well, and it will also perform reasonably well for the Duffing oscillator.

An important remark is that we are going to see that a constant scaling of this σ results in a significantly better long time behavior.

Note that we in Chapter 6 used the same method (Proposition 4.4), but used an extra step with a smaller step size to make sure the numerical differentiation for *rhs* was accurate enough, Equation (6.23). Experiments have shown that this was not necessary for the solution of Lotka-Volterra or the Duffing oscillator, and thus no extra step is necessary, we just use the value of f evaluated at the previous point.

The remaining suggestions for σ in this section has only been calculated for the Lotka-Volterra system.

7.7.2 Minimizing h^2-coefficient in the invariant expansion

Equation (7.38) gives the error expansion for Lotka-Volterra in the invariant which we are to conserve. We may solve the h^2-coefficient in terms of β, and then using Equation (7.31) to find an expression for σ.

For β we get

$$\det(f_\sigma(y)) = \beta(y) = \frac{2u^2 + v^2 - 4u - 4v + 6}{u + v - 3} \tag{7.41}$$

and which gives the $\sigma_{InvCoeff} : \mathbf{R}^2 \to \mathbf{R}$

$$\boxed{\sigma_{InvCoeff}(y) = -\frac{u^3 - 3u^2 + u^2v + 3u - 2vu - 3v + 3 + v^2}{(u + v - 3)^2}} \tag{7.42}$$

Numerical results are in Section 7.8.

7.7.3 Making a Poisson integrator

The Lotka-Volterra system is a Poisson system with the structure matrix

$$B(y) = \begin{pmatrix} 0 & uv \\ -uv & 0 \end{pmatrix}.$$

The reason for Symplectic Euler to perform as well as it does for the Lotka-Volterra is not because of its symplectic property, but because of the fact that it is a Poisson integrator. We may see if it is possible to adjust σ such that our Lie-Euler also becomes a Poisson integrator.

From [10, Section VII.2.5] we have the requirement for a mapping $\Phi : \mathbf{R}^2 \to \mathbf{R}^2$ to be a Poisson map:

$$\Phi'(y)B(y)\Phi'(y)^T = B(\Phi(y));$$

The elements of Jacobian $\Phi'_{LE}(y)$ (found by differentiating Equation (7.28) and inserted for $\det f_\sigma$) become after truncation of their Taylor-series

$$\begin{aligned}
\Phi'_{LE}(y,\sigma)_{11} &= 1 + (v-2)h + \left(\frac{1}{2}(2u - 2 + \sigma)u + \frac{1}{2}u^2 - u + \frac{1}{2}\sigma u + \frac{1}{2} - \frac{3}{2}\sigma + \frac{1}{2}\sigma v\right)h^2 \\
\Phi'_{LE}(y,\sigma)_{12} &= uh + \frac{1}{2}h^2\sigma u \\
\Phi'_{LE}(y,\sigma)_{21} &= -vh + \frac{1}{2}(2u - 2 + \sigma)vh^2 \\
\Phi'_{LE}(y,\sigma)_{22} &= 1 + (1-u)h + \left(\sigma v + \frac{1}{2}u^2 - u + \frac{1}{2}\sigma u + \frac{1}{2} - \frac{3}{2}\sigma\right)h^2
\end{aligned} \tag{7.43}$$

Now the computation

$$\Phi'_{LE}(y,\sigma)B(y)\Phi'_{LE}(y,\sigma)^T - B(\Phi'_{LE}(y,\sigma)) = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} C(y,\sigma)h^2 + \mathcal{O}(h^3)$$

is done in Maple, and we solve for the coefficient-function $C(y,\sigma)$ to be zero. This results in the following simple expression for $\sigma : \mathbf{R}^2 \to R$

$$\boxed{\sigma_{Poiss}(y) = -2\frac{u(u + v - 1)}{u + v}} \tag{7.44}$$

7.7.4 Projecting away isotropy

This strategy is inspired by the "orthogonal" rigid-body solver from [15], where the isotropy-part of the $\mathfrak{so}(3)$-matrix is projected away. We may do so here too by requiring

$$\langle f + \sigma\zeta, \zeta\rangle = 0 \quad \text{which gives} \quad \sigma = -\frac{\langle f, \zeta\rangle}{\langle \zeta, \zeta\rangle}$$

Using a Froebenius norm on $\mathfrak{sl}(2)$ this results in the function $\sigma_{Orth}\colon \mathbf{R}^2 \to \mathbf{R}$

$$\boxed{\sigma_{Orth}(y) = -\frac{\mathrm{Trace}(f(y)^T\zeta(y))}{\mathrm{Trace}(\zeta(y)^T\zeta(y))}} \tag{7.45}$$

This method is expected to perform in the league of the uncorrected Lie-Euler and has only been numerically tested on the Lotka-Volterra system. This method will be equivalent for all the three choices of f as their mutual differences lie in the isotropy, and we could have used it to define the "standard" Lie-Euler method. This would not do much else than clutter our notation, so we have rather defined the "standard" Lie-Euler as one of the versions of f together with $\sigma = 0$.

7.8 Numerical results

7.8.1 Lie-Euler with isotropy correction

Here we test the methods, first for Lotka-Volterra, over a fairly small integration period of time, up to $T = 7$, using the above choices of σ. The results are in Figure 7.7. All non-trivial choices of σ seem to be decent improvements over σ_0 which collapses soon after the here shown time-interval.

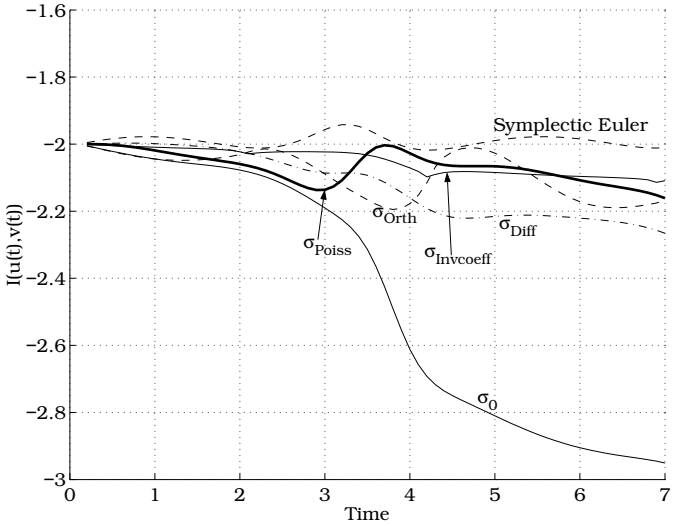


Figure 7.7: The invariant of all choices of σ for Lotka-Volterra plotted against Symplectic Euler. The choice $\sigma = 0$ collapses soon after $T = 7$.

From now on we ignore the trivial choice $\sigma = 0$. First we show that σ_{Orth} and $\sigma_{InvCoeff}$ collapse after long enough time, Figure 7.8 (the invariants jump over several magnitudes right after the shown time intervals).

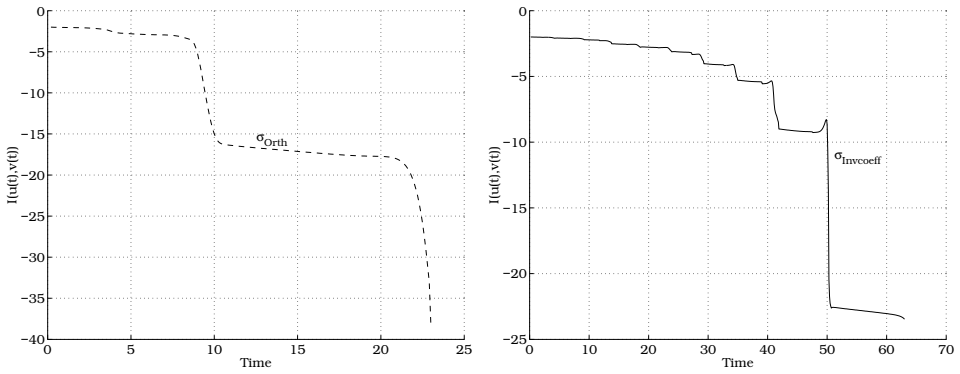


Figure 7.8: Failure of σ_{Orth} and $\sigma_{InvCoeff}$ for the Lotka-Volterra problem.

Now neglecting the methods which until now have performed badly in Figure 7.7 and Figure 7.8, we turn our attention to a somewhat bigger interval of time. Figure 7.9 shows σ_{Poiss}, σ_{Diff} and Symplectic Euler up to $T = 30$. σ_{Diff} does net seem to be any better than σ_{Orth} or $\sigma_{InvCoeff}$, but it is possible to stabilize it as we will see shortly.

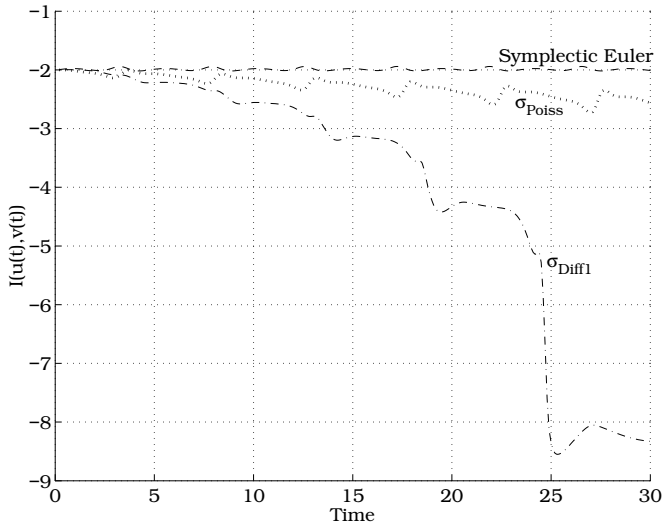
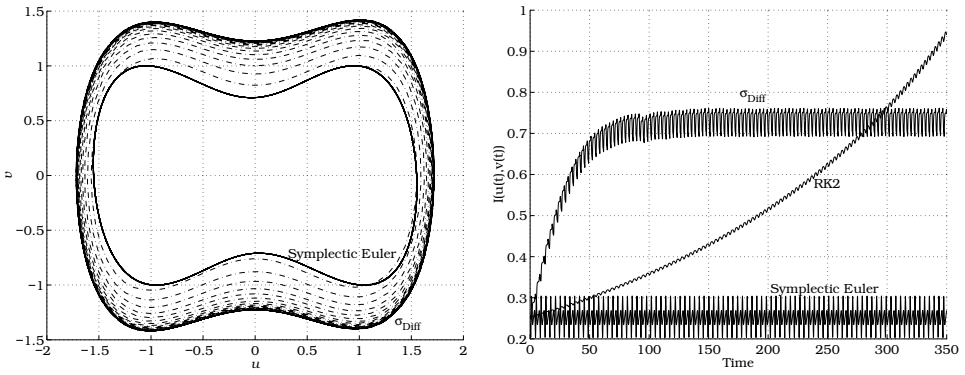


Figure 7.9: The invariant of the better-performing choices of σ up to $T = 30$ for the Lotka-Volterra problem. Time step 0.1.

For the Duffing oscillator, we have available the corresponding σ_{Diff} and Symplectic Euler. We compare them both to a second order Runge-Kutta method (not a Lie group version), as the local analysis which has determined σ_{Diff}, predicts that Lie-Euler with σ_{Diff} should perform worse than a standard second order method. The results are plotted in Figure 7.10. Lie-Euler with σ_{Diff} spirals outwards, but stabilizes at a trajectory slightly outside the exact

trajectory (very well approximated by Symplectic Euler). The second order method on the other hand, spirals outwards (the trajectory is not plotted) and shows no sign of stabilization.



(a) Trajectories in the phase plane (b) Preservation of the hamiltonian

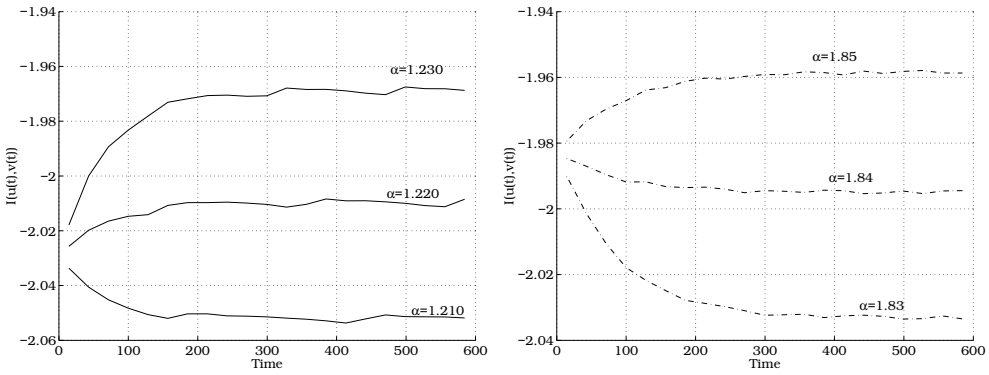
Figure 7.10: Results for the Duffing oscillator. Time step 0.1.

7.8.2 Tweaking σ by shooting

For the Lotka-Volterra system, the method σ_{Diff} seems to collapse in Figure 7.9 just as σ_{Orth} and $\sigma_{InvCoeff}$ did in Figure 7.8. σ_{Poiss} also has a decreasing invariant, but much slower than σ_{Diff}. For the Duffing oscillator, σ_{Diff} seems stable enough, but stabilizes on a wrong orbit. Anyhow, numerical experiments have shown that, both σ_{Diff} and σ_{Poiss} may be significantly improved by scaling them with appropriate constants. Let

$$\tilde{\sigma}(y) = \alpha\,\sigma(y) \tag{7.46}$$

and let α be a constant throughout the integration. The values for α has been found by trial and error for each system and for each choice of σ, plotting the invariant for large time intervals. Sadly enough, these constants seems in addition to be dependent on the time step h.



(a) $\tilde{\sigma}_{Poiss}$ (b) $\tilde{\sigma}_{Diff}$

Figure 7.11: Shooting method to find the α's for Lotka-Volterra. The invariant has been averaged over sequential intervals of time, only 20 data points are used for the plotted invariants. Otherwise the plots would have shown nothing but wild oscillations. Time step 0.1.

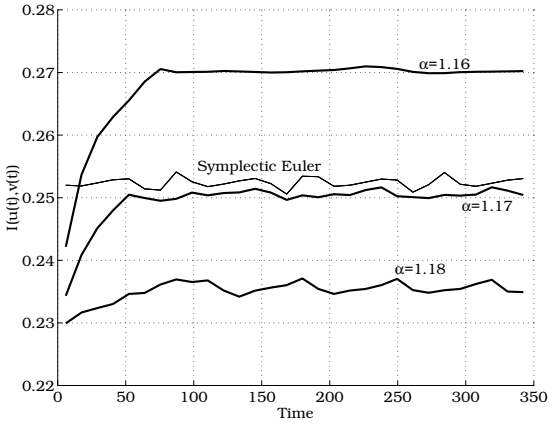


Figure 7.12: Shooting method to find optimal α for the Duffing oscillator. The plot is an average over 30 buckets of invariant values. The optimal α is 1.17 for the time step $h = 0.1$.

What is remarkable by these α's is that the methods now remain stable and more correct for much larger time intervals than shown in Figure 7.11. They have been tested to be stable for time intervals up to 10^5 (one million time steps).

These methods may now be compared to Symplectic Euler as in Figure 7.13. Compared to the exact solution, calculated by MATLAB's `ode23`-function, the $\tilde{\sigma}_{Diff}$ may be claimed to be the best solver, as Symplectic Euler misses more of the exact path in the upper right part of the solution curves. Note that both σ-methods may be further tweaked by using more decimals for the α's.

Numerical tests for stabilization was unsuccessful for the σ_{Orth} and $\sigma_{InvCoeff}$ of Figure 7.8.

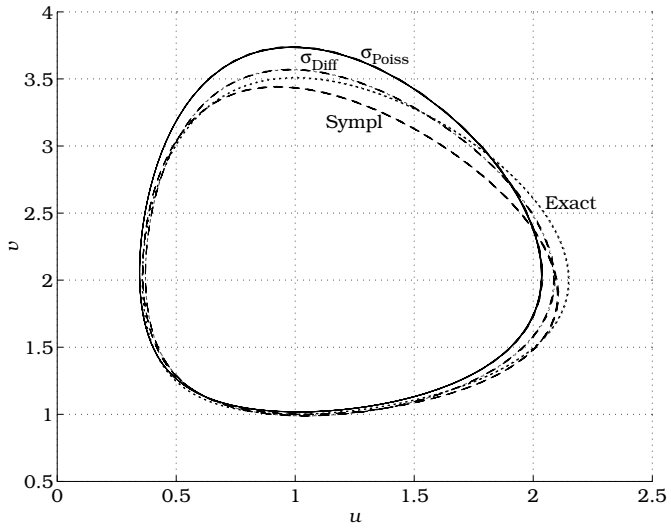


Figure 7.13: Comparing the best $\tilde{\sigma}_{Diff}$ and $\tilde{\sigma}_{Poiss}$ for Lotka-Volterra with Symplectic Euler and the exact solution. Only the trajectory at times in $[1800, 2000]$ are plotted to avoid cluttering the figure. Time step 0.1

7.8.3 Using Newton iteration to find an optimal correction

At the end we implement a Newton solver to find the σ which yield a zero change in the invariant for the Lotka-Volterra system. This is not viable as a solver, as it is way too slow, but could perhaps contribute to the analysis.

Numerical results are shown in Figure 7.14. We knew that along the line $u + v = 3$ the isotropy has no effect on the solution. The Newton algorithm is therefore expected to have convergence problems near the line, and if it converges, the proposed σ value could be very large. The experiments confirm this.

The Newton algorithm for choosing σ performs flawlessly until it is about to cross the line $u + v = 3$ for the first time, around $t = 2.1$. Around this line, values for σ is not be found, so $\sigma = 0$ is chosen. This results in a significant degradation of the invariant preservation, which affects the further integration when we are far enough from $u+v = 3$ again. The lower plot of Figure 7.14 shows that σ_{Diff} and σ_{Poiss} does *not* choose "optimal" σ-values, but the selections lead *globally* to a better solution.

7.8.4 Timing issues

There is of course the issue of computational time. Symplectic Euler, Equation (7.22), is extremely simple and fast to execute compared to what has to be done for the other methods. This issue is not considered important for the current analysis of methods exploiting isotropy, and nothing has been done regarding optimization of the new methods. Our primary goal is to discover new possibilities raised by isotropy, rather than to compete with Symplectic Euler for solving Lotka-Volterra or the Duffing oscillator in real-world applications.

The methods have the following execution time for a time interval of 100 on the Lotka-Volterra system.

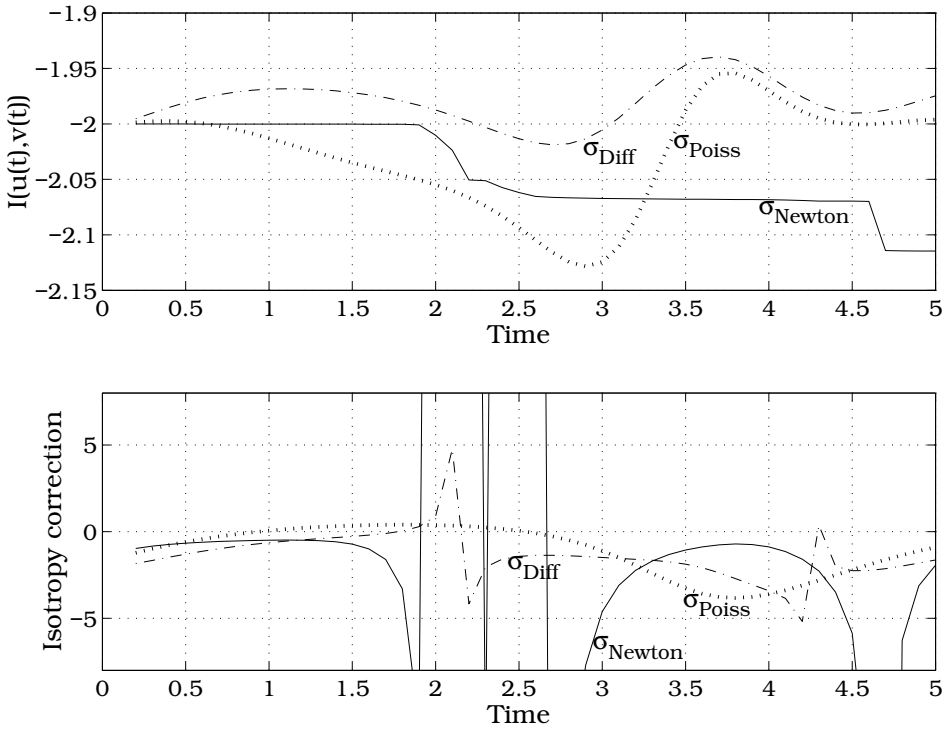


Figure 7.14: Conservation of the invariant for the three methods, compared to the actually chosen σ-values. The Newton algorithm produces very large σ-values near the line $u + v = 3$. This line is crossed around $t = 2.1$ and $t = 4.6$. The constant α is used here, but has negligible effect for such a short time interval. Approximately one period shown with time step 0.1.

Method	Time, $T = 100$
Euler	1.2 s
Lie-Euler, $\sigma = 0$	2.3 s
Symplectic Euler	1.3 s
σ_{Diff}	5.8 s
σ_{Poiss}	2.6 s
σ_{Orth}	3.7 s
$\sigma_{InvCoeff}$	2.5 s

Table 7.2: Execution time.

7.9 Concluding remarks

We have shown that using a Lie group method on $\mathbf{R}^2$ might be a viable alternative, as long as isotropy *is* considered, and if more theory explaining the behavior of the α becomes available.

Symplectic Euler is the method that still outperforms all other attempts. It is fast and it is a Symplectic and Poisson integrator which yields extreme stability properties for these problems.

Another important property of Symplectic Euler is the possibility to find its adjoint method, and then composing Symplectic Euler with its adjoint, which results in the Störmer-Verlet scheme. Störmer-Verlet is also a Poisson-integrator. The other methods are *not* at all this flexible in gaining a higher order of accuracy. Important to notice is that Symplectic Euler is not a Poisson integrator for all Poisson systems.

The σ_{Diff} is perhaps the most interesting method, as it does not use *any* information of the differential equation other than the tweaking of the α (though which are quite critical for its performance on Lotka-Volterra). The price for this is paid in execution time, as it is also the most demanding method. It is the application of Proposition 4.4 which also performed well on the rigid body. σ_{Poiss} also performed equivalently, but was dependent on complicated expansions in Maple to be determined and algebraic solutions therefrom.

A remarkable thing to notify is the comments in Section 7.8.3. The isotropy correction chosen by the successful algorithms, σ_{Diff} and σ_{Poiss}, are not optimal at each point (if they were, they should be equal to the Newton iterates in Figure 7.14). The formulas for σ_{Diff} and σ_{Poiss} are based on a *local* result (excluding the role of the α's), but excels in global behavior. This indicates an intricate relationship between these isotropy corrections and global behavior of the solver, which is an interesting candidate for future research.

Chapter 8

Conclusions

We have shown as in Lewis and Olver [15] that it is possible to take advantage of isotropy subgroups in the formulation of Lie group methods. The added isotropy term does not affect the original differential equation, but has a significant effect on the solution produced by the numerical solver.

Lewis and Olver used analytic derivatives of the differential equation to determine the isotropy correction. We have shown that it is possible to obtain the same effect by just using numerical differentiation.

The main contribution of the isotropy correction is not the decreased error constant of the second order error term, as predicted by the analysis done, but the global behavior. Lie-Euler, which performs badly on problems of this type, has been improved to the league of the superior Symplectic Euler.

As it stands, our isotropy corrected solvers are not ready to replace other numerical solvers, as the numerical cost is still high, in addition to the mysteriousity of the α-constants. The α for the rigid body problem was significantly closer to 1 than for Lotka-Volterra and the Duffing oscillator. Was that a property of the Lie group action, $SO(3)$ contra $SL(2)$, or was it pure luck? Nevertheless, isotropy corrections are an interesting new direction in geometric integration, and future research will hopefully reveal more theory that are able to explain the phenomenas observed.

Bibliography

[1] F. Adams. *Lectures on Lie Groups*. 1965.

[2] V. I. Arnold. *Mathematical Methods of Classical Mechanics*. Springer-Verlag, GTM 60, Second edition, 1989.

[3] C. J. Budd and M. D. Piggott. Geometric integration and its applications, 2001. To appear in Foundations of Computational Mathematics, a volume of the Handbook of Numerical Analysis, ed. Ph. G. Chiarlet and F. Cucker, published by Elsevier Science.

[4] S. R. Buss. Accurate and Efficient Simulation of Rigid Body Rotations. *Journal of Computational Physics*, (164):374–406, 2000.

[5] E. Celledoni and B. Owren. Lie group methods for rigid body dynamics and time integration on manifolds. Technical report, The Norwegian University of Science and Technology, Trondheim, Norway, 1999.

[6] E. Celledoni and B. Owren. On the implementation of Lie group methods on the Stiefel manifold. Technical Report Numerics No. 9/2001, The Norwegian University of Science and Technology, Trondheim, Norway, 2001.

[7] N. Curtis. *Matrix Groups*. Springer-Verlag, 1988.

[8] B. I. Dundas. Differential topology. 2002.

[9] K. Engø. On the construction of geometric integrators in the RKMK class. *BIT*, 40(1):41–61, 2000.

[10] E. Hairer, Ch. Lubich, and G. Wanner. *Geometric Numerical Integration*. Springer-Verlag, 2002.

[11] E. Hairer, S. P. Nørsett, and G. Wanner. *Solving Ordinary Differential Equations I, Nonstiff Problems*. Springer-Verlag, Second revised edition, 1993.

[12] E. Hairer and G. Wanner. *Solving Ordinary Differential Equations II, Stiff and Differential-Algebraic Problems*. Springer, Berlin, 1991.

[13] A. Iserles. Brief introduction to Lie-group methods. To appear in proceedings of the Fort Collins workshop on preservation of stability under discretization (Don Estep & Simon Tavener, eds.), to be published by SIAM, 2001.

[14] A. Iserles, H. Z. Munthe-Kaas, S. P. Nørsett, and A. Zanna. Lie-group methods. *Acta Numerica*, 9:215–365, 2000.

[15] D. Lewis and P. Olver. Geometric Integration Algorithms on Homogeneous Manifolds. 2001.

[16] R. McLachlan and C. Scovel. Equivariant constrained symplectic integration. Technical Report LA-UR-93-3225, Los Alamos National Lab., New Mexico, USA, 1993.

[17] H. Munthe-Kaas. Lie–Butcher theory for Runge–Kutta methods. *BIT*, 35(4):572–587, 1995.

[18] H. Munthe-Kaas. Runge–Kutta methods on Lie groups. *BIT*, 38(1):92–111, 1998.

[19] H. Munthe-Kaas. High order Runge–Kutta methods on manifolds. *Appl. Numer. Math.*, 29:115–127, 1999.

[20] A. L. Onishchik and E. B. Vinberg. *Foundations of Lie Theory*. Springer-Verlag, 1997.

[21] B. Owren. Lecture notes, Geometric Integration. 2002.

[22] L. Perko. *Differential Equations and Dynamical Systems*. Number 7 in Texts in Applied Mathematics. Springer-Verlag, 1996.

[23] S. Reich. Momentum conserving symplectic integrators. *Physica D*, 76:375–383, 1994.

[24] V. S. Varadarajan. *Lie Groups, Lie Algebras, and Their Representations*. GTM 102. Springer-Verlag, 1984.

Appendix A

Matrix exponential for $\mathfrak{sl}(2)$-matrices

Recall that $\mathfrak{sl}(2)$ is all matrices with zero trace, generally in the form

$$A = \begin{pmatrix} a & b \\ c & -a \end{pmatrix} \tag{A.1}$$

Squaring this matrix we get

$$A^2 = \begin{pmatrix} a^2 + bc & 0 \\ 0 & a^2 + bc \end{pmatrix} = -\det(A)I = -\beta I$$

where I is the two by two identity matrix.

We use this property in the following lemma:

Lemma A.1. *The matrix exponential for 2×2 skew-symmetric matrices A is*

$$\exp A = \begin{cases} \cos\sqrt{\det A}\, I + \frac{\sin\sqrt{\det A}}{\sqrt{\det A}} A & \det A > 0 \\ \cosh\sqrt{-\det A}\, I + \frac{\sinh\sqrt{-\det A}}{\sqrt{-\det A}} A & \det A < 0 \\ I + A & \det A = 0 \end{cases} \tag{A.2}$$

Proof. We use $\beta = \det A$ as above for simplicity. Use the infinite sum expression for the matrix exponential

$$\begin{aligned} \sum_{k=0}^{\infty} \frac{A^k}{k!} &= \sum_{m=0}^{\infty} \frac{A^{2m}}{(2m)!} + \sum_{m=0}^{\infty} \frac{A^{2m+1}}{(2m+1)!} \\ &= \sum_{m=0}^{\infty} \frac{\beta^m}{(2m)!} I + A \sum_{m=0}^{\infty} \frac{\beta^m}{(2m+1)} \end{aligned} \tag{A.3}$$

We split up the proof depending on the sign of β:

$\beta > 0$) Set $\gamma = \sqrt{\beta}$ and continue Equation (A.3):

$$\begin{aligned} &= \sum_{m=0}^{\infty} \frac{(-\gamma^2)^m}{(2m)!} + A \sum_{m=0}^{\infty} \frac{(-\gamma^2)^m}{(2m+1)!} \\ &= \sum_{m=0}^{\infty} \frac{(-1)^m \gamma^{2m}}{(2m)!} + \frac{A}{\gamma} \sum_{m=0}^{\infty} \frac{(-1)^m \gamma^{2m+1}}{(2m+1)!} \\ &= \underline{\cos\gamma I + \frac{\sinh\gamma}{\gamma} A.} \end{aligned} \tag{A.4}$$

$\beta < 0$) Set $\gamma = \sqrt{-\beta}$ and continue Equation (A.3):

$$\begin{aligned} &= \sum_{m=0}^{\infty} \frac{\gamma^{2m}}{(2m)!} I + \frac{A}{\gamma} \sum_{m=0}^{\infty} \frac{\gamma^{2m+1}}{(2m+1)!} \\ &= \underline{\cosh\gamma I + \frac{\sinh\gamma}{\gamma} A.} \end{aligned} \tag{A.5}$$

$\beta = 0$) This means that A is nilpotent, and the sum for $\exp(A)$ is merely truncated

$$\sum_{k=0}^{\infty} \frac{A^k}{k!} = \underline{I + A.} \tag{A.6}$$

□

Note the effect of multiplying the matrix by a constant $h > 0$, as $\det(hA) = h^2 \det A$

$$\exp hA = \begin{cases} \cos h\sqrt{\det A}\, I + \frac{\sin h\sqrt{\det A}}{h\sqrt{\det A}} hA & \det A > 0 \\ \cosh h\sqrt{-\det A}\, I + \frac{\sinh h\sqrt{-\det A}}{h\sqrt{-\det A}} hA & \det A < 0 \\ I + hA & \det A = 0 \end{cases} \tag{A.7}$$

The splitting of the result in Lemma A.1 is only because we have restricted ourselves to using real numbers in the argument to the trigonometric functions. Their Taylor-expansions are equal:

$$\begin{aligned} \cos h\sqrt{\beta} &= 1 - \frac{1}{2}\beta h^2 + \frac{1}{24}\beta^2 h^4 + \mathcal{O}(h^6) \\ \cosh h\sqrt{-\beta} &= 1 - \frac{1}{2}\beta h^2 + \frac{1}{24}\beta^2 h^4 + \mathcal{O}(h^6) \\ \frac{\sin h\sqrt{\beta}}{h\sqrt{\beta}} &= 1 - \frac{1}{6}\beta h^2 + \frac{1}{120}\beta^2 h^4 + \mathcal{O}(h^6) \\ \frac{\sinh h\sqrt{-\beta}}{h\sqrt{-\beta}} &= 1 - \frac{1}{6}\beta h^2 + \frac{1}{120}\beta^2 h^4 + \mathcal{O}(h^6) \end{aligned}$$

Appendix B

Symplectic Euler for Lotka-Volterra

We are here going to prove that the Symplectic Euler method is a Poisson integrator for any Poisson systems with a separable Hamiltonian. This includes in particular the Lotka-Volterra system.

The Symplectic Euler method reads

$$\Phi_h(u_n, v_n) = \begin{pmatrix} u_{n+1} \\ v_{n+1} \end{pmatrix} = \begin{pmatrix} u_n + hu_{n+1}v_n H_v(u_{n+1}, v_n) \\ v_n - hu_{n+1}v_n H_u(u_{n+1}, v_n) \end{pmatrix} \tag{B.1}$$

where $H(u, v)$ is the Hamiltonian for the system in question. For Lotka-Volterra it is

$$H(u, v) = I(u, v) = \ln u - u + 2\ln v - v.$$

The criterion for a Poisson integrator is the one from Definition 5.8, namely

$$\Phi'(y)B(y)\Phi'(y)^T = B(\Phi(y)) \tag{B.2}$$

where $B(y) = B(u, v)$ is the structure matrix for the Lotka-Volterra system,

$$B(y) = \begin{pmatrix} 0 & uv \\ -uv & 0 \end{pmatrix}.$$

Equation (B.2) requires the Jacobian of the mapping $(u_n, v_n) \mapsto (u_{n+1}, v_{n+1})$. We will calculate this for any Hamiltonian H for the Symplectic Euler method. Implicit differentiation of Equation (B.1) gives the four equations

$$\begin{aligned}
\frac{\partial u_{n+1}}{\partial u_n} &= 1 + \frac{\partial u_{n+1}}{\partial u_n} h v_n H_v \\
\frac{\partial u_{n+1}}{\partial v_n} &= h\frac{\partial u_{n+1}}{\partial v_n} v_n H_v + h u_{n+1} H_v + h u_{n+1} v_n H_{vv} \\
\frac{\partial v_{n+1}}{\partial u_n} &= -h\frac{\partial u_{n+1}}{\partial u_n} v_n H_u - h u_{n+1} v_n H_{uu} \frac{\partial u_{n+1}}{\partial u_n} \\
\frac{\partial v_{n+1}}{\partial v_n} &= 1 - h\frac{\partial u_{n+1}}{\partial v_n} v_n H_u - h u_{n+1} H_u - h u_{n+1} v_n \frac{\partial u_{n+1}}{\partial v_n}
\end{aligned} \tag{B.3}$$

where the partial derivatives are evaluated at $(u_{n+1}, v_n)^T$. We can sort these equations into the more tractable form

$$\begin{pmatrix} 1 - hv_n H_v & 0 \\ hv_n(H_u + u_{n+1}H_{uu}) & 1 \end{pmatrix} \begin{pmatrix} \frac{\partial u_{n+1}}{\partial u_n} & \frac{\partial u_{n+1}}{\partial v_n} \\ \frac{\partial v_{n+1}}{\partial u_n} & \frac{\partial v_{n+1}}{\partial v_n} \end{pmatrix} = \begin{pmatrix} 1 & hu_{n+1}(H_v + v_n H_{vv}) \\ 0 & 1 - hu_{n+1}H_u \end{pmatrix} \tag{B.4}$$

written compactly as $A \cdot D\Phi_h = C$. The requirement (B.2) now reads

$$\begin{aligned} A^{-1}CB(y)(A^{-1}C)^T &= B(\Phi_h(y)) \\ &\Leftrightarrow \\ CB(y)C^T &= AB(\Phi_h(y))A^T \end{aligned}$$

Multiplying out this matrix equation results in

$$\begin{pmatrix} 0 & u_n v_n(1 - hu_{n+1}H_u) \\ -u_n v_n(1 - hu_{n+1}H_u) & \end{pmatrix} = \begin{pmatrix} 0 & u_{n+1}v_{n+1}(1 - hv_n H_v) \\ -u_n v_n(1 - hv_n H_v) & 0 \end{pmatrix} \tag{B.5}$$

equivalent to the single equation

$$u_n v_n(1 - hu_{n+1}H_u) = u_{n+1}v_{n+1}(1 - hv_n H_v). \tag{B.6}$$

We are able to prove the validity of Equation (B.6) for separable Hamiltonians, $H(u, v) = K(u) + L(v)$, because we then get an explicit expression for the method itself, as in Equation (7.22):

$$\begin{aligned} u_{n+1} &= \frac{u_n}{1 - hL_v(v_n)} \\ v_{n+1} &= v_n(1 - hu_{n+1}K_u(u_{n+1})). \end{aligned}$$

Inserting these expressions for u_{n+1} and v_{n+1} into Equation (B.6) immediately yields our desired result.