UNIVERSITAT POLITÈCNICA DE VALÈNCIA

MAGNUS-BASED GEOMETRIC INTEGRATORS FOR DYNAMICAL SYSTEMS WITH TIME-DEPENDENT POTENTIALS

NIKITA KOPYLOV

SUPERVISORS:
Sergio Blanes
Philipp Bader

PhD Thesis
Universitat Politècnica de València

Valencia
February 2019

ABSTRACT

The present thesis addresses the numerical integration of Hamiltonian systems with explicitly time-dependent potentials. These problems are common in mathematical physics because they come from quantum, classical and celestial mechanics.

The goal of the thesis is to construct integrators for several important non-autonomous problems: the Schrödinger equation, which is the cornerstone of quantum mechanics; the Hill and the wave equations, that describe oscillating systems; the Kepler problem with time-variant mass.

Chapter 1 describes the motivation and the aims of the work in the historical context of numerical integration. In Chapter 2 essential concepts and some fundamental tools used throughout the thesis are introduced.

The design of the proposed integrators is based on the composition and splitting methods and the Magnus expansion. In Chapter 3, the former is described. Their main idea is to recombine some simpler integrators to obtain the solution. The salient concept of order conditions is described in that chapter. Chapter 4 summarises Lie algebras and the Magnus expansion — algebraic tools that help to express the solution of time-dependent differential equations.

The linear Schrödinger equation with time-dependent potential is considered in Chapter 5. Given its particular structure, new, Magnus-based *quasi-commutator-free* integrators are build. Their efficiency is shown in numerical experiments with the Walker–Preston model of a molecule in an electromagnetic field.

In Chapter 6, *Magnus–splitting* methods for the wave and the Hill equations are designed. Their performance is demonstrated in numerical experiments with various oscillatory systems: the Mathieu equation, the matrix Hill eq., the wave and the Klein–Gordon–Fock eq.

Chapter 7 shows how the algebraic approach and the Magnus expansion can be generalised to non-linear problems. The example used is the Kepler problem with decreasing mass.

The thesis is concluded by Chapter 8, in which the results are reviewed and possible directions of future work are outlined.

RESUMEN

Esta tesis trata sobre la integración numérica de sistemas hamiltonianos con potenciales explícitamente dependientes del tiempo. Los problemas de este tipo son comunes en la física matemática, porque provienen de la mecánica cuántica, clásica y celestial.

La meta de la tesis es construir integradores para unos problemas relevantes no autónomos: la ecuación de Schrödinger, que es el fundamento de la mecánica cuántica; las ecuaciones de Hill y de onda, que describen sistemas oscilatorios; el problema de Kepler con la masa variante en el tiempo.

El Capítulo 1 describe la motivación y los objetivos de la obra en el contexto histórico de la integración numérica. En el Capítulo 2 se introducen los conceptos esenciales y unas herramientas fundamentales utilizadas a lo largo de la tesis.

El diseño de los integradores propuestos se basa en los métodos de composición y escisión y en el desarrollo de Magnus. En el Capítulo 3 se describe el primero. Su idea principal consta de una recombinación de unos integradores sencillos para obtener la solución del problema. El concepto importante de las condiciones de orden se describe en ese capítulo. En el Capítulo 4 se hace un resumen de las álgebras de Lie y del desarrollo de Magnus que son las herramientas algebraicas que permiten expresar la solución de ecuaciones diferenciales dependientes del tiempo.

La ecuación lineal de Schrödinger con potencial dependiente del tiempo está examinada en el Capítulo 5. Dado su estructura particular, nuevos métodos *casi sin conmutadores*, basados en el desarrollo de Magnus, son construidos. Su eficiencia es demostrada en unos experimentos numéricos con el modelo de Walker–Preston de una molécula dentro de un campo electromagnético.

En el Capítulo 6, se diseñan los métodos de *Magnus–escisión* para las ecuaciones de onda y de Hill. Su eficiencia está demostrada en los experimentos numéricos con varios sistemas oscilatorios: con la ecuación de Mathieu, la ec. de Hill matricial, las ecuaciones de onda y de Klein–Gordon–Fock.

El Capítulo 7 explica cómo el enfoque algebraico y el desarrollo de Magnus pueden generalizarse a los problemas no lineales. El ejemplo utilizado es el problema de Kepler con masa decreciente.

El Capítulo 8 concluye la tesis, reseña los resultados y traza las posibles direcciones de la investigación futura.

RESUM

Aquesta tesi tracta de la integració numèrica de sistemes hamiltonians amb potencials explícitament dependents del temps. Els problemes d'aquest tipus són comuns en la física matemàtica, perquè provenen de la mecànica quàntica, clàssica i celest.

L'objectiu de la tesi és construir integradors per a uns problemes rellevants no autònoms: l'equació de Schrödinger, que és el fonament de la mecànica quàntica; les equacions de Hill i d'ona, que descriuen sistemes oscil·latoris; el problema de Kepler amb la massa variant en el temps.

El Capítol 1 descriu la motivació i els objectius de l'obra en el context històric de la integració numèrica. En Capítol 2 s'introdueixen els conceptes essencials i unes ferramentes fonamentals utilitzades al llarg de la tesi.

El disseny dels integradors proposats es basa en els mètodes de composició i escissió i en el desenvolupament de Magnus. En el Capítol 3, es descriu el primer. La seua idea principal consta d'una recombinació d'uns integradors senzills per a obtenir la solució del problema. El concepte important de les condicions d'orde es descriu en eixe capítol. El Capítol 4 fa un resum de les àlgebres de Lie i del desenvolupament de Magnus que són les ferramentes algebraiques que permeten expressar la solució d'equacions diferencials dependents del temps.

L'equació lineal de Schrödinger amb potencial dependent del temps està examinada en el Capítol 5. Donat la seua estructura particular, nous mètodes *quasi sense commutadors*, basats en el desenvolupament de Magnus, són construïts. La seua eficiència és demostrada en uns experiments numèrics amb el model de Walker–Preston d'una molècula dins d'un camp electromagnètic.

En el Capítol 6 es dissenyen els mètodes de *Magnus–escissió* per a les equacions d'onda i de Hill. El seu rendiment està demostrat en els experiments numèrics amb diversos sistemes oscil·latoris: amb l'equació de Mathieu, l'ec. de Hill matricial, les equacions d'onda i de Klein–Gordon–Fock.

El Capítol 7 explica com l'enfocament algebraic i el desenvolupament de Magnus poden generalitzar-se als problemes no lineals. L'exemple utilitzat és el problema de Kepler amb massa decreixent.

El Capítol 8 conclou la tesi, ressenya els resultats i traça les possibles direccions de la investigació futura.

CONTENTS

LIST OF TABLES

ACRONYMS

BCH Baker–Campbell–Hausdorff

BDF backward differentiation formula

CF commutator-free

DE differential equation

DFT discrete Fourier transform

FFT fast Fourier transform

FSAL first-same-as-last

GL Gauss–Legendre

HE Hill equation

IFFT inverse fast Fourier transform

ISO International Organization for Standardization

IVP initial value problem

KGFE Klein–Gordon–Fock equation

MATLAB Matrix Laboratory

ME Magnus expansion

MMP matrix–matrix product

MVP matrix–vector product

ODE ordinary differential equation

PDE partial differential equation

QCF quasi-commutator-free

RKN Runge–Kutta–Nyström

RK Runge–Kutta

SE Schrödinger equation

WE wave equation

1

MOTIVATION

The theme of the present thesis is *geometric numerical integrators* for DEs with variable coefficients. In short, the flow of a DE expresses its solution, and some classes of equations possess flows with specific structural properties. Geometric methods, compared to most of the so-called *general-purposed* methods, are specifically tailored to preserve these properties in numerical solutions. Besides an improvement in qualitative behaviour, they also provide better long-time integration results.

The motivation of geometric integration comes from many sciences: quantum physics, molecular dynamics, physical chemistry, classical mechanics, astronomy, molecular dynamics, numerical analysis itself, and other branches of mathematics.

THE GOAL of the thesis is to construct geometric numerical integrators for the linear time-dependent Schrödinger, Hill and wave equations, and the non-linear Kepler problem with time-variant mass by taking into account the specific structure of these problems. The new methods are expected to be more efficient with lower computational cost and higher accuracy, especially in large time scales.

THE STRUCTURE of the work is as following. In order to have a look on a bigger picture and to explain the rationale behind the ideas used, we provide a short overview that outlines relevant events of the history numerical integration.

The rest of the thesis is arranged in a prevalently deductive manner. We begin with basic definitions. The introductory chapters describe mathematical tools that lay foundations for the design of integrators:

- discretization of DEs,
- splitting and composition methods,
- Lie algebras which simplify the analysis of Lie groups,
- the Magnus expansion (ME) that expresses the solution of time-dependent linear DE,
- matrix exponentiation.

With respect to the main part, we describe the design of the following types of methods:

- unitary integrators for the Schrödinger equation (SE) with time-dependent potentials in Chapter 5;
- symplectic integrators for non-autonomous Hamiltonian systems: the Hill equation (HE) and the wave equation (WE) in Chapter 6;
- a generalisation of the above-mentioned to the non-linear case on the example of the Kepler problem with time-dependent mass.

All the methods are constructed as symmetric schemes, therefore they are suitable for reversible systems.

1.1 HISTORICAL OVERVIEW

Although some apparatus of infinitesimal calculus was known and used to find finding areas and quadratures of curves as early as classical antiquity, it did not form any closed theory [118].

But it was not until the late 17^{th} century, when Isaac Newton and Gottfried Leibniz created calculus. The term itself and most of the modern notation are credited to the latter [7, 121]. The two scholars set forth the idea of describing processes via DEs, both ordinary and partial.

The theory and its tools were rapidly advanced by Newton, Leibniz, the brothers Jakob (who proposed the name 'integral' [121]) and Johann Bernoulli [7, 66], and Jacopo Francesco Riccati.

Leonhard Euler extensively studied linear autonomous DEs of higher orders and corresponding systems with constant coefficients. He also researched some equations with variable coefficients [121]. Euler's ideas were developed by Daniel Bernoulli, Jean-Baptiste d'Alambert, Joseph-Louis Lagrange and Adrien-Marie Legendre [121]. By the middle of the 18^{th} century, the theory of differential equations was shaped as a separate branch of mathematics.

The growth of interest in DEs was prevalently driven by the applied problems in general and celestial mechanics, physics, ballistics, etc. The one-dimensional wave equation, which describes a vibrating string, was discovered by d'Alambert, and the three-dimensional — by Euler. Similar problems drew the interest of Bernoulli and Lagrange [38, 107].

Classical numerical integrators

Already the works of the pioneers showed the limited possibility of the resolution of the DE in analytic form. Consequently, new ways for integration were to be found.

Leonhard Euler created [55, 121] the first general approximate method for solving initial value problem (IVP) for a 1^{st}-order DE

$$\frac{\mathrm{d}y(t)}{\mathrm{d}t} = f(t,y), \quad y(t_0) = y_0. \tag{1.1}$$

To apply the method, we divide the time interval $[t_0;\ t_f]$ by n points into $n-1$ *time steps* $t_0, t_1, \ldots, t_{n-1} \coloneqq t_f$ (for simplicity assume them to be constant $\tau \coloneqq t_k - t_{k-1}\ \forall k$). Then we use the simplest finite-differences approximation to $\dot{y}$ to calculate $y_k \coloneqq y(t_k)$:

Henceforth we use the dot notation for derivatives with respect to time: $\dot{\boldsymbol{y}} \coloneqq \frac{\mathrm{d}\boldsymbol{y}}{\mathrm{d}t}$

$$y_k = y_{k-1} + \tau f(t_{k-1}, y_{k-1}). \tag{1.2}$$

The scheme is called the *explicit Euler method*. This approach is generalised straightforwardly to linear systems of differential equations.

More advanced, in a certain manner, modifications of the Euler method are the *implicit (backward) Euler*

$$y_k = y_{k-1} + \tau f(t_k, y_k)$$

and the *implicit midpoint rule*

$$y_k = y_{k-1} + \tau f\left(t_{k-1} + \frac{\tau}{2}, \frac{y_{k-1} + y_k}{2}\right), \tag{1.3}$$

which already has 2^{nd} order of accuracy and is the simplest symmetric method: the formula stays the same if we integrate backwards in time by changing τ to $-\tau$ and y_{k-1} to y_k.

More advanced methods can be obtained with the following logic. we can integrate (1.1) and get

$$y_k = y_{k-1} + \int_{t_{k-1}}^{t_k} f(t, y(t))\mathrm{d}t. \tag{1.4}$$

Approximating the right-hand side with an appropriate quadrature rule, we get a new numerical method, for instance, the implicit trapezoidal rule:

$$y_k = y_{k-1} + \frac{\tau}{2}(f(t_{k-1}, y_{k-1}) + f(t_k, y_k)). \tag{1.5}$$

In the end of the 19^{th} century Carl David Tolmé Runge used this approach with 'prediction' of y_k by means of the Euler method [55]. It became the foundation of a large family of integrators. These ideas were developed by Martin Wilhelm Kutta and Karl Heun. Various optimised versions of Runge–Kutta methods eventually became the default option in many numerical packages [21, 64, 91, 102].

An s-stage Runge–Kutta method is defined through a set of coefficients $a_{i,j}$, b_i and c_i:

$$\begin{aligned} y_k &= y_{k-1} + \tau \sum_{i=1}^{s} b_i k_i, \\ k_i &= f\left(t_{k-1} + c_i\tau,\ y_{k-1} + \tau \sum_{j=1}^{s} a_{ij} k_j\right),\ i = 1, \ldots, s. \end{aligned} \tag{1.6}$$

The coefficients are usually displayed as a *Butcher tableau*

$$\begin{array}{c|c} \boldsymbol{c} & \boldsymbol{A} \\ \hline & \boldsymbol{b}^T \end{array} := \begin{array}{c|ccc} c_1 & a_{1,1} & \dots & a_{1,s} \\ \vdots & \vdots & \ddots & \vdots \\ c_s & a_{s,1} & \dots & a_{s,s} \\ \hline & b_1 & \dots & b_s \end{array}$$

The matrix $\boldsymbol{A}$ of explicit methods is strictly lower triangular (i. e. $a_{i,j} \equiv 0$ for $j \geq i$), and these zeros are omitted for clarity. For example, we provide tableaux for two 4^{th}-order methods: the explicit 3/8-rule (on the left) and the implicit collocation Runge–Kutta (RK) Gauss–Legendre (GL) method:

$$\begin{array}{c|cccc} \frac{1}{3} & \frac{1}{3} & & & \\ \frac{2}{3} & -\frac{1}{3} & 1 & & \\ 1 & 1 & -1 & 1 & \\ \hline & \frac{1}{8} & \frac{3}{8} & \frac{3}{8} & \frac{1}{8} \end{array} \qquad \begin{array}{c|cc} \frac{1}{2}-\frac{\sqrt{3}}{6} & \frac{1}{4} & \frac{1}{4}-\frac{\sqrt{3}}{6} \\ \frac{1}{2}+\frac{\sqrt{3}}{6} & \frac{1}{4}+\frac{\sqrt{3}}{6} & \frac{1}{4} \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}$$

As the theory of differential equations grew and their understanding deepened, it became apparent that some problems possess properties that hinder them from being solved 'well' by general methods.

One example is stiff problems which are defined in various ways as systems: a) where implicit methods are 'tremendously' better than explicit; b) with components in different timescales (i. e. fast and slow); c) with enormous maximal eigenvalue [36, 46, 54, 105]. As a result, these equations compel small time steps (hence high computational cost) to overcome numerical instability when solved with explicit methods [105]. In this case *linear multistep methods*, which use previously computed values to advance the solution, produce robust results. They were introduced by John Couch Adams, Francis Bashforth and advanced by Forest Ray Moulton and William Edmund Milne, hence the names of explicit *Adams–Bashforth* and implicit *Adams–Moulton* methods. Another implicit family named the backward differentiation formula (BDF) was introduced in the middle of 20^{th} century [46]. Good results for stiff problems are also obtained by the implicit Runge–Kutta methods, described later by John Charles Butcher [37, 52].

Geometric integration

According to ISO 5725-1, accuracy is the closeness of agreement between a test result and the accepted reference value

Although the general methods provide convenient practical schemes for solving many types of DEs, they suffer from certain shortcomings. Most importantly, when a problem has some underlying geometry, most of them fail to preserve the structural properties of solutions such as: symplecticity, volume, energy and other first integrals [60] — which has a direct effect on the accuracy of the solution, especially for long-term integration.

Fortunately, it is possible to construct specialised methods that take into account the properties of the problem at hand and prevent the loss of geometric structure. This approach was termed geometric integration by Jesús María Sanz Serna [101]. Geometric solvers were mainly developed for Hamiltonian problems which have different structural invariants, for instance, energy and symplecticity [98]. However, no method can preserve both properties [122]. The preservation of structure is addressed by different approaches [60]: backward error analysis; composition and splitting methods; exponential integrators; variational integrators; Lie-group methods [62].

Methods presented in this work are designed to preserve symplecticity in Hamiltonian problems.

1.2 SHORTCOMING OF THE EXISTING METHODS

As we have seen, some common problems can benefit from the application of geometric integrators. Now, we will enumerate some issues we strive to improve upon in new numerical schemes.

GENERAL LINEAR METHODS Most classical integrators belong to this class. They can solve a wide range of equations but have significant disadvantages:

- explicit RK methods cannot be symplectic [29] (thus, produce not so good results for long-time integration of Hamiltonian systems) and can be unstable for large time steps;
- implicit RK can be symplectic (as are the collocation GL methods) and stable, but have reasonable implementation cost only up to order 6, with more effort required if the system is not linear [61];
- multistep methods can use variable time step and lower computational expenses by changing their order, but are either not symplectic or of low order [54].

GEOMETRIC METHODS In general, one expects structure preservation and, thus, more accurate results from geometric methods. However, we can name several flaws related to the integration of non-autonomous problems:

- most composition and splitting methods can be adapted to this case, but show suboptimal results;
- integrators based on the direct application of the Magnus expansion (ME) are well-suited for time-dependent problems, but the presence of commutators makes them expensive;

- the commutator-free methods try to circumvent the aforementioned obstacle; these methods perform well in both linear [5] and non-linear [41] cases; they also show good results for oscillatory problems; however, they are rather general;
- exponential variational methods are designed for stiff and highly-oscillatory problems with constant linear part [56] and cannot be directly applied to the fully time-dependent case;
- the so-called Runge–Kutta–Nyström (RKN) methods for 2^{nd}-order problems $\ddot{y} = f(y)$ have reasonable cost and good performance, but it is possible to make better methods for non-autonomous problems with some specific structure.

The techniques we will use in this work are expected to produce numerical algorithms that address these issues when the aforementioned problems of interest are solved.

2

NUMERICAL MODELLING

In order to simulate a physical problem on a computer, mathematical and then numerical models are created. The former encode the problem in the formal language of mathematics, while the latter is needed to create a suitable digital representation.

We consider physical problems that are described by differential equations (DEs). If the equations are given in partial derivatives or have order higher than one, then the first step is to convert them to a more convenient 1st-order system of ordinary differential equations (ODEs).

2.1 BASIC DEFINITIONS

*Following ISO 80000-2, capital bold letters (**A**, **B** etc.) denote matrices and lowercase bold (**y**…) – vectors*

To introduce basics concepts we consider the initial value problem (IVP) for a d-dimensional 1st-order system of ODEs in normal form:

$$\dot{\boldsymbol{y}}(t) = \boldsymbol{f}(t, \boldsymbol{y}(t)),\ \boldsymbol{y}(t_0) = \boldsymbol{y}_0, \quad \boldsymbol{y} \in \mathbb{C}^d \tag{2.1}$$

Every state $\boldsymbol{y}$ of the system can be represented in the *phase space* [6]. The right-hand side function $\boldsymbol{f}(t, \boldsymbol{y}(t))$ is a *vector field* that at any point of the phase space shows velocity of the solution $\boldsymbol{y}(t)$ that passes through that point [52].

The relation between an initial value $\boldsymbol{y}_0$ and the solution $\boldsymbol{y}(t)$ is expressed by a one-parametric group of transformations of the state space called the *exact flow* φ_t of the system [3, 29, 52]:

$$\varphi_t(\boldsymbol{y}_0) = \boldsymbol{y}(t) \text{ for } \boldsymbol{y}_0 = \boldsymbol{y}(t_0). \tag{2.2}$$

On the other hand, if $\boldsymbol{y}_0$ is fixed, then $\varphi_t(\boldsymbol{y}_0)$ expresses the solution of the IVP. The map φ_t is time symmetric: the integration from t_0 to t_f and then backwards does not change the trajectory and recovers the initial value.

Every numerical method (say, the explicit Euler (1.2)) is associated with the one-parameter mapping $\Psi_\tau : \boldsymbol{y}_{k-1} \mapsto \boldsymbol{y}_k$ which is called the *numerical flow* of the system [29]. A method is *of order p* if it approximates the exact flow with an error $\mathcal{O}\left(\tau^{p+1}\right) = \Psi_\tau(\boldsymbol{y}_{k-1}) - \varphi_\tau(\boldsymbol{y}_{k-1})$.

The *adjoint method* Ψ^*_τ is the inverse of Ψ_τ with reversed time step [29, 52]: $\Psi^*_\tau := \Psi^{-1}_{-\tau}$. The adjoint method has the same order as the original [52]. If a method and its adjoint are the same: $\Psi^*_\tau := \Psi_\tau$, then it is called *symmetric* (e. g. the implicit midpoint on page 3).

The main scope of the thesis is the time-integration of non-autonomous Hamiltonian systems, which are common in physics. They are de-

scribed equations defined by means of a function with vector arguments $H(t, \boldsymbol{q}, \boldsymbol{p})\colon \mathbb{R} \times \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$, called *Hamiltonian*:

Subscripted gradient $\nabla_{\boldsymbol{x}} H$ stands for elementwise differentiation $\frac{\partial H}{\partial x_i}$

$$\dot{\boldsymbol{p}} = -\nabla_{\boldsymbol{q}} H(t, \boldsymbol{q}, \boldsymbol{p}), \quad \dot{\boldsymbol{q}} = \nabla_{\boldsymbol{p}} H(t, \boldsymbol{q}, \boldsymbol{p}). \tag{2.3}$$

H represents the total energy of the system. When H is autonomous, a numerical method is called *energy-preserving* if $H(\boldsymbol{q}_0, \boldsymbol{p}_0) \equiv H(\boldsymbol{q}_k, \boldsymbol{p}_k) \equiv const$ for any k.

If we put $\boldsymbol{u} \coloneqq (\boldsymbol{q}, \boldsymbol{p})^T$, $\dim \boldsymbol{u} = 2d$, then (2.3) can be written as

$$\dot{\boldsymbol{u}} = \boldsymbol{J}_{2d} \nabla_{\boldsymbol{u}} H(\boldsymbol{u}), \tag{2.4}$$

where $\boldsymbol{J}_{2d}$ is the canonical matrix

$$\boldsymbol{J} = \boldsymbol{J}_{2d} = \begin{bmatrix} \boldsymbol{O} & \boldsymbol{I}_d \\ -\boldsymbol{I}_d & \boldsymbol{O} \end{bmatrix}. \tag{2.5}$$

A system's flow $\varphi_t(\boldsymbol{u})$ is a *symplectic* transformation if its Jacobian matrix φ_t' for $t \geq 0$ verifies

$$\varphi_t'(\boldsymbol{u})^T \boldsymbol{J} \varphi_t'(\boldsymbol{u}) \equiv \boldsymbol{J}.$$

Accordingly, an integrator Ψ is symplectic if

$$\Psi_\tau'(\boldsymbol{u})^T \boldsymbol{J} \Psi_\tau'(\boldsymbol{u}) \equiv \boldsymbol{J}. \tag{2.6}$$

2.2 REDUCTION TO FIRST-ORDER SYSTEMS

Assume a 2^{nd}-order system of ODEs in normal form

$$\ddot{\boldsymbol{y}} = \boldsymbol{f}(t, \boldsymbol{y}).$$

Then, by putting $\boldsymbol{u}_1(t) \coloneqq \boldsymbol{y}$ and $\boldsymbol{u}_2(t) \coloneqq \dot{\boldsymbol{y}}$, we obtain the following 1^{st}-order system:

$$\dot{\boldsymbol{u}} = \begin{bmatrix} \dot{\boldsymbol{y}} \\ \boldsymbol{f}(t, \boldsymbol{y}) \end{bmatrix}.$$

If the equation is linear, as

$$\ddot{\boldsymbol{y}} = \boldsymbol{N}(t) \boldsymbol{y},$$

then the corresponding system is

$$\dot{\boldsymbol{u}} = \begin{bmatrix} \boldsymbol{O} & \boldsymbol{I} \\ \boldsymbol{N} & \boldsymbol{O} \end{bmatrix} \boldsymbol{u}. \tag{2.7}$$

As it has been mentioned on page 3, to propagate solution $\boldsymbol{u}$ in time, the given integration interval $[t_0;\ t_f]$ is divided into equal small time steps of length τ, and stepwise approximations $\boldsymbol{u}_k$ to the solution values $\boldsymbol{u}(t_k)$ are evaluated at the temporal grid points $t_k = t_0 + \tau k$.

We have introduced time steps of equal length. However, general methods, like RK, improve their efficiency if adaptive time-stepping is used [54, 55]. It turns out that geometric integrators generally deteriorate when used with standard step size control [29, 52]. Consequently, to maintain good qualitative behaviour and a slow error growth, they should be used with fixed time step. Nevertheless, some methods were adapted to use variable steps, retaining the benefits [22, 53].

Spatial discretization is necessary when a problem is formalised with a partial differential equation (PDE) with a linear operator $f(t,x)$:

$$\frac{\partial y(t)}{\partial t} = f(t,x)y(t).$$

The usual way to proceed in to discretize it in space with a sufficiently dense d-point mesh [72] which yields a d-dimensional vector $\boldsymbol{u}$. By doing so, we obtain a first-order system of ODEs with respect to time:

$$\dot{\boldsymbol{u}} = \boldsymbol{F}(t)\boldsymbol{u}.$$

Alternatively, finite differences, spectral collocation methods with trigonometric polynomials, the Galerkin method with Hermite basis or other methods can be used [72] for certain types of equations.

Moreover, the so-called *postponed discretization* can also be used [18, 19]. This approach is of service when certain relations, arising from problem, are easier to treat in analytical form. Methods are constructed using this representation and converted to a discrete (matrix) form at the last stage.

2.3 FAST FOURIER TRANSFORM

This is the most important algorithm
of our lifetime.
Gilbert Strang [108]

Partial differential equations considered in this work require calculation of spatial derivatives. Under an assumption of periodic boundary conditions, fast Fourier transform (FFT) algorithms [45] can be used to compute them in an accurate and efficient manner.

The *continuous Fourier transform* $\mathcal{F}$ represents a function $f(x)$ in the basis of harmonic functions of frequency ω:

$$\hat{f}(\omega) \coloneqq (\mathcal{F}f)(\omega) = \int_{-\infty}^{\infty} \mathrm{e}^{-\mathrm{i}\omega x} f(x)\mathrm{d}x.$$

The inverse transform $\mathcal{F}^{-1}\hat{f}$ restores f:

$$f(x) = \int_{-\infty}^{\infty} \mathrm{e}^{\mathrm{i}\omega x} \hat{f}(\omega)\mathrm{d}\omega.$$

Let $f(x)$ be a periodic function with period X and $\{f_k\}$ is a sequence of its values at d points $\{x_k\} := \{kX/d\}$, $k = 0,\dots,d-1$. Then, the *discrete Fourier transform (DFT)* processes $\{f_k\}$ in the following way:

$$\hat{f}_n = \frac{1}{d}\sum_{k=0}^{d-1} \mathrm{e}^{-\frac{2\pi \mathrm{i}}{d}kn} f_k,\ n = 0,\dots,d-1;$$

the inverse being

$$f_k = \sum_{n=0}^{d-1} \mathrm{e}^{\frac{2\pi \mathrm{i}}{d}kn} \hat{f}_n,\ k = 0,\dots,d-1.$$

In other words, the DFT switches representations between a space and its corresponding frequency (Fourier) domain.

The naive implementation, following from the definition, requires $\mathcal{O}(d^2)$ products. Fortunately, FFT algorithms can employ the 'divide and conquer' paradigm to reduce the number of operations to $\mathcal{O}(d \log d)$, which is specifically notable for large sequences.

A result, important for the construction of integrators is that, given samples of a periodic function, an FFT can be used to compute derivatives of this function [63, 72, 112]. To this end, it is convenient to express the DFT in a symmetric form. As before, let $f(x)$ be X-periodic and d be even, and the DFT reads [63, 112]:

$$\hat{f}_n = \frac{1}{d}\sum_{k=0}^{d-1} \mathrm{e}^{-\frac{2\pi \mathrm{i}}{d}kn} f_k,\ n = -\frac{d}{2}+1,\dots,\frac{d}{2}. \tag{2.8}$$

This permutation means moving the zero frequency component to the centre of the spectrum. After putting $\hat{f}_{-d/2} \equiv \hat{f}_{d/2}$ inverse symmetric DFT reads [112]:

$$f_k = \sum_{n=-d/2}^{d/2} \mathrm{e}^{\frac{2\pi \mathrm{i}}{d}kn} \hat{f}_n,\ k = 0,\dots,d-1. \tag{2.9}$$

Then, the following procedure is applied to calculate the j^{th} derivative of $f(x)$ evaluated at $\{x_k\}$:

Algorithm 1: Periodic spectral differentiation

Data: values $\{f_k\}$ of $f(x)$, derivative order j;
1 centre the zero frequency compute $\{\hat{f}_k\}$ with an FFT;
2 compute $\hat{w}_k := \left(k\frac{2\pi \mathrm{i}}{X}\right)^j \hat{f}_k$;
3 **if** *j is odd* **then**
4 $\quad \hat{w}_{d/2} := 0$
5 **end**
6 compute w from $\hat{w}$ with the inverse FFT;
Result: values $\{w_k\}$ of $\frac{\mathrm{d}^j f(t)}{\mathrm{d}x^j}$.

For practical implementation it is important to take into account the order in which the Fourier coefficients are stored [112]. For example, in MATLAB [77] and SciPy [64] they are stored in an array, where the first element is the zero frequency, then go the positive and negative frequencies in ascending order. Consequently, functions like `fftshift`/`ifftshift` should be applied either to the sample or the wavenumber array.

2.4 MATRIX EXPONENTIALS

The core stage of the integration of linear systems is matrix exponentiation. In the numerical case there are many approaches to compute matrix exponentials [84], and they should be chosen according to the matrix size and structure. In this section we review some methods, relevant to the problems we strive to solve.

2.4.1 *Simplest cases*

The matrix exponential of a square matrix $\boldsymbol{Z}$ is defined as the power series

$$\mathrm{e}^{\boldsymbol{Z}} = \sum_{k=0}^{\infty} \frac{\boldsymbol{Z}^k}{k!}. \tag{2.10}$$

As a result, $\mathrm{e}^{\boldsymbol{O}} = \boldsymbol{I}$.

There are two particular cases when the matrix exponential is expressed trivially:

Assume a diagonal matrix $\boldsymbol{D} = diag\{d_1, \ldots, d_n\}$. Its exponential is computed simply by exponentiating the diagonal elements:

$$\mathrm{e}^{\boldsymbol{D}} = \begin{bmatrix} \mathrm{e}^{d_1} & & \\ & \ddots & \\ & & \mathrm{e}^{d_n} \end{bmatrix}.$$

A matrix $\boldsymbol{Z}$ is called *nilpotent* of order k, if $\boldsymbol{Z}^k = \boldsymbol{O}$. In this case, the power series reduces to a polynomial

$$\mathrm{e}^{\boldsymbol{Z}} = \boldsymbol{I} + \ldots + \frac{\boldsymbol{Z}^{k-1}}{(k-1)!}. \tag{2.11}$$

Particularly, if $\boldsymbol{Z}$ is block nilpotent or order 2, for instance,

$$\boldsymbol{Z} = \begin{bmatrix} \boldsymbol{O} & \boldsymbol{B} \\ \boldsymbol{O} & \boldsymbol{O} \end{bmatrix},$$

then its exponential does not require raising the matrix to any power:

$$\mathrm{e}^{\boldsymbol{Z}} = \begin{bmatrix} \boldsymbol{I} & \boldsymbol{B} \\ \boldsymbol{O} & \boldsymbol{I} \end{bmatrix}.$$

2.4.2 *Scaling and squaring*

The most widely used algorithm to compute the exponential of a general matrix is the scaling and squaring with Padé rational approximation [64, 76, 83], although more efficient methods based on Taylor polynomials were proposed [13]. These general algorithms are suitable for matrix dimensions of several hundred [84]. In general, given a matrix $\boldsymbol{Z}$ from a Lie algebra $\mathfrak{g}$, they do not guarantee that $\exp \boldsymbol{Z}$ belongs to the corresponding group G (see Chapter 4). However, for some specific classes of matrices there exist methods with similar computational cost that ensure that the $\exp \boldsymbol{Z} \in G$ [42].

The cornerstone of the approach is the following property:

$$\mathrm{e}^{\boldsymbol{Z}} = \left(\mathrm{e}^{\boldsymbol{Z}/2^m}\right)^{2^m}.$$

The exponent m is usually chosen to be the minimal integer to obtain $\|\boldsymbol{Z}\|/2^m \leq 1$, so that $\mathrm{e}^{\boldsymbol{Z}/2^m}$ can be efficiently and robustly computed by the approximants. Afterwards, the exponential is restored by repeated squaring.

2.4.3 *Approximation in the Krylov subspace*

Often the exponential itself is not important, but the action $\mathrm{e}^{\boldsymbol{Z}}\boldsymbol{u}$ on a non-zero vector $\boldsymbol{u}$ is of interest. Moreover, when $\boldsymbol{Z}$ is large (with dimension being thousands) and sparse, scaling and squaring may be overly expensive. Then, it is reasonable to approximate $\mathrm{e}^{\boldsymbol{Z}}\boldsymbol{u}$ in the Krylov subspace.

To this end, the orthonormal basis $\boldsymbol{V}_m \coloneqq [\boldsymbol{v}_1, \ldots, \boldsymbol{v}_m]$ of the subspace is built by the *Arnoldi process*, which consists of the repeated multiplication of $\boldsymbol{Z}$ and $\boldsymbol{u}$ with subsequent orthogonalisation. The parameter m is typically much smaller that the matrix dimension d. The process requires m matrix–vector products (MVPs) and yields $m+1$ vectors. Next, the exponential is approximated via the exponential of a smaller Hessenberg ('almost triangular') matrix $\boldsymbol{H}_m$, which can be computed by scaling and squaring:

$$\mathrm{e}^{\boldsymbol{Z}}\boldsymbol{u} \approx \|\boldsymbol{u}\|_2 \boldsymbol{V}_m \mathrm{e}^{\boldsymbol{H}_m}\boldsymbol{e}_1, \tag{2.12}$$

where $\boldsymbol{e}_1$ is the first column of the identity matrix $\boldsymbol{I}_{d\times d}$.

The Krylov subspaces of $\boldsymbol{Z}$ and $\tau\boldsymbol{Z}$ are identical [96], therefore

$$\mathrm{e}^{\tau\boldsymbol{Z}}\boldsymbol{u} \approx \|\boldsymbol{u}\|_2 \boldsymbol{V}_m \mathrm{e}^{\tau\boldsymbol{H}_m}\boldsymbol{e}_1, \ \forall\tau.$$

Notice, that the resulting matrices $\boldsymbol{V}_m$ and $\boldsymbol{H}_m$ are square, as they do not include the last obtained elements $\boldsymbol{v}_{m+1}$ and $h_{m+1,m}$.

Let $\boldsymbol{Z}_H$ be an Hermitian matrix as in the discretized SE $\partial\boldsymbol{u}/\partial t = -\mathrm{i}\boldsymbol{Z}_H\boldsymbol{u}$. Then the *Lanczos algorithm* [72] can be used to speed up the basis calculation, and the Hessenberg matrix becomes symmetric and tridiagonal. Taking into account the imaginary unit and unitarity of the state vector ($\|\boldsymbol{u}\|_2 = 1$), the exponential is calculated as

$$\mathrm{e}^{-\mathrm{i}\tau\boldsymbol{Z}_H}\,\boldsymbol{u} \approx \boldsymbol{V}_m\,\mathrm{e}^{-\mathrm{i}\tau\boldsymbol{T}_m}\,\boldsymbol{e}_1. \tag{2.13}$$

It is convenient to estimate the subspace dimension m during iteration. To this end, the following practical estimator can be used at step j:

$$error_j = h_{j+1,j}\left(\frac{2}{3}\left|\boldsymbol{e}_j^T\,\mathrm{e}^{-i\tau\boldsymbol{T}_j/2}\,\boldsymbol{e}_1\right| + \frac{1}{6}\left|\boldsymbol{e}_j^T\,\mathrm{e}^{-i\tau\boldsymbol{T}_j}\,\boldsymbol{e}_1\right|\right),$$

where $\boldsymbol{e}_1$ and $\boldsymbol{e}_j$ are the first and last column of $\boldsymbol{I}_{j\times j}$, respectively. Having estimated the error, the iteration can terminate if the error is smaller that a desired tolerance (i. e. $error \leq tol$), and $m := j$.

Algorithm 2: Arnoldi iteration [96]

Data: matrix $\boldsymbol{Z}$, non-zero vector $\boldsymbol{u}$

normalise $\boldsymbol{v}_1 \coloneqq \boldsymbol{u}/\|\boldsymbol{u}\|_2$;
for $j = 1,\ldots,m$ **do**
  multiply $\boldsymbol{w} \coloneqq \boldsymbol{Z}\boldsymbol{v}_j$;
  for $i = 1,\ldots,j$ **do**
    compute the dot product $h_{i,j} = \boldsymbol{w} \cdot \boldsymbol{v}_i$;
    modify $\boldsymbol{w} = \boldsymbol{w} - h_{i,j}\boldsymbol{v}_i$;
  end
  compute the next elements $h_{j+1,j} \coloneqq \|\boldsymbol{w}\|_2$ and $\boldsymbol{v}_{j+1} \coloneqq \boldsymbol{w}/h_{j+1,j}$.
end

Result: matrix of basis column vectors $\boldsymbol{V}_m \coloneqq [\boldsymbol{v}_1,\ldots,\boldsymbol{v}_m]$ and $m \times m$ upper Hessenberg matrix $\boldsymbol{H}_m$.

Algorithm 3: Lanczos algorithm [72, 96]

Data: Hermitian matrix $\boldsymbol{Z}_H$, unitary vector $\boldsymbol{u}$, $\beta_1 \coloneqq 0$

for $j = 1,\ldots,m$ **do**
  multiply $\boldsymbol{w} \coloneqq \boldsymbol{Z}_H\boldsymbol{v}_j$;
  compute the dot product $\alpha_j = \boldsymbol{w} \cdot \boldsymbol{v}_j$;
  modify $\boldsymbol{w} = \boldsymbol{w} - \alpha_j\boldsymbol{v}_j - \beta_j\boldsymbol{v}_{j-1}$;
  compute the next elements $\beta_{j+1} \coloneqq \|\boldsymbol{w}\|_2$, $\boldsymbol{v}_{j+1} \coloneqq \boldsymbol{w}/\beta_{j+1}$;
end
fill the matrix $\boldsymbol{T}_m$ with $t_{j,j} \coloneqq \alpha_j$ and $t_{j+1,j} \equiv t_{j,j+1} \coloneqq \beta_{j+1}$.

Result: matrix of basis column vectors $\boldsymbol{V}_m \coloneqq [\boldsymbol{v}_1,\ldots,\boldsymbol{v}_m]$ and $m \times m$ symmetric tridiagonal matrix $\boldsymbol{T}_m$.

3 COMPOSITION AND SPLITTING METHODS

This chapter describes two important techniques for designing geometric integrators: composition and splitting [80]. The main idea is to combine simpler schemes with appropriate coefficients to get higher-order methods.

First, we show how the idea is applied to autonomous problems. Then, how it can be generalised to time-dependent ones. Finally, we explain how suitable composition coefficients are found.

3.1 AUTONOMOUS CASE

3.1.1 *Composition of methods*

In the notation of Chapter 2, assume that φ_t is the flow of an autonomous equation $\dot{\boldsymbol{y}} = \boldsymbol{f}(\boldsymbol{y})$. Let ψ_τ be a numeric approximation of order p to φ_τ that preserves some geometric properties of the solution.

It is possible to construct a method of a higher order in the following way: build a composition with appropriate real coefficients $a_1, \dots, a_s$:

$$\Psi_\tau := \psi_{a_s\tau} \circ \psi_{a_{s-1}\tau} \cdots \circ \psi_{a_2\tau} \circ \psi_{a_1\tau}.$$

If it satisfies two conditions [52]

$$\begin{cases} a_1 + \ldots + a_s = 1, \\ a_1^{p+1} + \ldots + a_s^{p+1} = 0, \end{cases}$$

then the new method has at least order $p+1$. Since these equations do not have real solutions for odd p, the procedure is only valid for even-order methods.

However, we can compose the same method with its adjoint ψ_τ^*:

$$\Psi_\tau := \psi_{b_s\tau} \circ \psi^*_{a_s\tau} \cdots \circ \psi_{b_1\tau} \circ \psi^*_{a_1\tau}. \tag{3.1}$$

Consequently, conditions for order $p+1$ become [52]

$$\begin{cases} a_1 + b_1 + \cdots + a_s + b_s = 1, \\ a_1^{p+1} + (-1)^p b_1^{p+1} + \cdots + a_s^{p+1} + (-1)^p b_s^{p+1} = 0, \end{cases}$$

allowing an order increase for odd p. This tactic produces a 2^{nd}-order symmetric method from any consistent 1^{st}-order method (with $p = s = 1$ and $a = b = 1/2$):

$$\Psi_\tau^{[2]} = \psi_{\tau/2} \circ \psi^*_{\tau/2} \quad \text{or} \quad \Psi_\tau^{[2]} = \psi^*_{\tau/2} \circ \psi_{\tau/2}.$$

By combination of these two approaches we can raise the order further. Given an even-order symmetric scheme $\Psi_\tau^{[p]}$, the *Suzuki–Yoshida technique* [109, 120] yields a symmetric scheme of order $p+2$:

$$\begin{aligned} \mathrm{SS}_\tau^{[p+2]} &= \Psi_{a\tau}^{[p]} \circ \Psi_{b\tau}^{[p]} \circ \Psi_{a\tau}^{[p]}, \\ \text{where } a &= \frac{1}{2 - 2^{1/(p+1)}},\ b = 1 - 2a. \end{aligned} \tag{3.2}$$

Compositions have the same structure-preserving properties as the basic method. However, the trade-off is that high-order methods obtained this way lead to numerous function evaluations.

EXAMPLE: SYMMETRIC COMPOSITIONS Consider compositions of a 2nd-order symmetric method $\Psi_\tau^{[2]}$:

$$\mathrm{SS}_\tau^{[p]} = \prod_{i=1}^{s} \Psi_{a_i\tau}^{[2]}. \tag{3.3}$$

If we do not aim to minimize the number of steps s, then we can build methods which show a good performance.

Among 4th-order methods, more precise approximations are obtained with the Suzuki fractal scheme [109]:

$$a_1 = a_2 = a_4 = a_5 = \frac{1}{4 - 4^{1/3}},\ a_3 = \frac{4^{1/3}}{4 - 4^{1/3}}. \tag{3.4}$$

The coefficients of the 6th-order 9-stage method $\mathrm{SS}_9^{[6]}$ that minimises $\max_i |a_i|$ are the following [79]:

$$\begin{aligned} a_1 = a_9 &= \ \ 0.392161444007314 \\ a_2 = a_8 &= \ \ 0.332599136789359 \\ a_3 = a_7 &= -0.706246172557639 \end{aligned} \qquad \begin{aligned} a_4 = a_6 &= 0.082213596293551 \\ a_5 &= 0.798543990934830. \end{aligned} \tag{3.5}$$

Analogously, an optimised 8th-order method $\mathrm{SS}_{17}^{[8]}$ has 17 stages, and its coefficients read [65]:

$$\begin{aligned} a_1 = a_{17} &= \ \ 0.130202483088890 \\ a_2 = a_{16} &= \ \ 0.561162981775108 \\ a_3 = a_{15} &= -0.389474962644847 \\ a_4 = a_{14} &= \ \ 0.158841906555156 \\ a_5 = a_{13} &= -0.395903894133238 \end{aligned} \qquad \begin{aligned} a_6 = a_{12} &= \ \ 0.184539640978316 \\ a_7 = a_{11} &= \ \ 0.258374387686322 \\ a_8 = a_{10} &= \ \ 0.295011723609310 \\ a_9 &= -0.605508533830035. \end{aligned} \tag{3.6}$$

A well-optimised 10th-order method $\mathrm{SS}_{35}^{[10]}$ [106] has the coefficients

$$\begin{aligned}
a_1 = a_{35} &= 0.078795722521686 & a_{10} = a_{26} &= -0.399105630136036\\
a_2 = a_{34} &= 0.313096103415109 & a_{11} = a_{25} &= 0.103087398527471\\
a_3 = a_{33} &= 0.027918383235078 & a_{12} = a_{24} &= 0.411430873955890\\
a_4 = a_{32} &= -0.229592841593907 & a_{13} = a_{23} &= -0.004866360583135\\
a_5 = a_{31} &= 0.130962061077165 & a_{14} = a_{22} &= -0.392033353708640\\
a_6 = a_{30} &= -0.269733405654511 & a_{15} = a_{21} &= 0.051942502962450\\
a_7 = a_{29} &= 0.074973343155891 & a_{16} = a_{20} &= 0.050665090759924\\
a_8 = a_{28} &= 0.111993423999810 & a_{17} = a_{19} &= 0.049674370639730\\
a_9 = a_{27} &= 0.366133449546227 & a_{18} &= 0.049317735759595.
\end{aligned} \tag{3.7}$$

Note: although in these examples the number of stages increases exponentially each time the order grows by 2, it approximately doubles, not triples as in (3.2).

3.1.2 *System splitting*

Let us turn our attention to systems $\dot{\boldsymbol{y}} = \boldsymbol{f}(\boldsymbol{y})$, whose right-hand side can be split into a sum of vector fields:

$$\boldsymbol{f}(\boldsymbol{y}) = \boldsymbol{f}^{\{1\}}(\boldsymbol{y}) + \boldsymbol{f}^{\{2\}}(\boldsymbol{y}) + \ldots \boldsymbol{f}^{\{k\}}(\boldsymbol{y}).$$

Suppose that each part can be solved numerically (or even analytically), with solutions being $\psi_\tau^{\{i\}}$, $i = 1,\ldots,k$. Then the composition of the corresponding solutions

$$\Psi_\tau = \psi_\tau^{\{k\}} \circ \ldots \circ \psi_\tau^{\{2\}} \circ \psi_\tau^{\{1\}} \tag{3.8}$$

provides a 1st-order approximation to the exact solution (hence the name *splitting method*).

A rather common structure of DEs is a system of two linear vector fields

$$f(\boldsymbol{u}) \coloneqq \boldsymbol{A}\boldsymbol{u} + \boldsymbol{B}\boldsymbol{u}. \tag{3.9}$$

For instance, in Hamiltonian systems $\boldsymbol{A}$ may represent the kinetic energy and $\boldsymbol{B}$ — the potential. Consider the autonomous case, when each part is integrated exactly. We can compose the solutions of each part to get the simple Lie–Trotter splitting and its adjoint [29, 52]:

Also called symplectic Euler, Euler–Cromer

$$\begin{aligned}
\mathrm{LT}_\tau &= \psi_\tau^{\{\boldsymbol{A}\}} \circ \psi_\tau^{\{\boldsymbol{B}\}} = \mathrm{e}^{\tau\boldsymbol{A}}\mathrm{e}^{\tau\boldsymbol{B}}\\
\mathrm{LT}_\tau^* &= \psi_\tau^{\{\boldsymbol{B}\}} \circ \psi_\tau^{\{\boldsymbol{A}\}} = \mathrm{e}^{\tau\boldsymbol{B}}\mathrm{e}^{\tau\boldsymbol{A}}
\end{aligned} \tag{3.10}$$

Taylor expansion shows that both of them are 1st-order methods, that is, $\mathrm{LT}_\tau(\boldsymbol{u}_k) = \varphi_\tau(\boldsymbol{u}_k) + \mathcal{O}\left(\tau^2\right)$.

Other names are: Strang splitting, leapfrog, Newton–Størmer–Verlet, Encke

The composition of the Lie–Trotter scheme with its adjoint gives the symmetric 2^{nd}-order Størmer–Verlet methods [29, 52]:

$$\begin{aligned} \mathrm{SV}_\tau &= \mathrm{LT}_{\tau/2} \circ \mathrm{LT}^*_{\tau/2} = \mathrm{e}^{\frac{\tau}{2}\boldsymbol{A}} \mathrm{e}^{\tau \boldsymbol{B}} \mathrm{e}^{\frac{\tau}{2}\boldsymbol{A}}; \\ \mathrm{SV}^*_\tau &= \mathrm{LT}^*_{\tau/2} \circ \mathrm{LT}_{\tau/2} = \mathrm{e}^{\frac{\tau}{2}\boldsymbol{B}} \mathrm{e}^{\tau \boldsymbol{A}} \mathrm{e}^{\frac{\tau}{2}\boldsymbol{B}}. \end{aligned} \tag{3.11}$$

These schemes possess the first-same-as-last (FSAL) property:

$$(\mathrm{SV}_\tau)^k = \underbrace{\mathrm{e}^{\frac{\tau}{2}\boldsymbol{A}} \mathrm{e}^{\tau \boldsymbol{B}} \mathrm{e}^{\frac{\tau}{2}\boldsymbol{A}} \dots \mathrm{e}^{\frac{\tau}{2}\boldsymbol{A}} \mathrm{e}^{\tau \boldsymbol{B}} \mathrm{e}^{\frac{\tau}{2}\boldsymbol{A}}}_{k \text{ times}} = \mathrm{e}^{\frac{\tau}{2}\boldsymbol{A}} \mathrm{e}^{\tau \boldsymbol{B}} \left(\mathrm{e}^{\tau \boldsymbol{A}} \mathrm{e}^{\tau \boldsymbol{B}} \right)^{k-1} \mathrm{e}^{\frac{\tau}{2}\boldsymbol{A}},$$

which means that k steps of the Størmer–Verlet require nearly the same number of operations as k steps of the Lie–Trotter scheme.

3.2 TIME-DEPENDENT CASE

A non-autonomous IVP $\dot{\boldsymbol{y}} = \boldsymbol{f}(t, \boldsymbol{y}), \boldsymbol{y}(t_0) = \boldsymbol{y}_0$ can be written in autonomous form [55] by making t a dependent variable:

$$\frac{\mathrm{d}}{\mathrm{d}t} \begin{bmatrix} \boldsymbol{y} \\ t_d \end{bmatrix} = \begin{bmatrix} \boldsymbol{f} \\ 1 \end{bmatrix}, \quad \begin{bmatrix} \boldsymbol{y}(t_0) \\ t_d(t_0) \end{bmatrix} = \begin{bmatrix} \boldsymbol{y}_0 \\ t_0 \end{bmatrix}.$$

Accordingly, a similar procedure is possible for a split system when applied to its parts. Let us again consider a linear two-part system, but with time-dependence:

$$\dot{\boldsymbol{u}} = \boldsymbol{A}(t)\boldsymbol{u} + \boldsymbol{B}(t)\boldsymbol{u}. \tag{3.12}$$

Two approaches [34] can be used to make the system autonomous:

- 'time averaging' replaces the autonomous flows in (3.8) by the flows, corresponding to time-dependent problems

$$\begin{aligned} &\dot{\boldsymbol{u}} = \boldsymbol{A}(t)\boldsymbol{u}, \ t \in [t_0 + c_k\tau; \ t_0 + (c_k + a_k)\tau], \\ &\dot{\boldsymbol{u}} = \boldsymbol{B}(t)\boldsymbol{u}, \ t \in [t_0 + d_k\tau; \ t_0 + (d_k + b_k)\tau]; \\ &a_0 \coloneqq 0, \ c_k = \sum_{i=0}^{k-1} a_i, \quad b_0 \coloneqq 0, \ d_k = \sum_{i=0}^{k-1} b_i; \end{aligned} \tag{3.13}$$

- 'time freezing' substitutes $\psi^{\{\boldsymbol{A}\}}_{a_k\tau}$ and $\psi^{\{\boldsymbol{B}\}}_{b_k\tau}$ by the flows, associated with the autonomous vector fields of

$$\begin{aligned} \dot{\boldsymbol{u}} &= \boldsymbol{A}(t_0 + d_k\tau, \boldsymbol{u}), \ t \in [t_0 + c_k\tau; \ t_0 + (c_k + a_k)\tau], \\ \dot{\boldsymbol{u}} &= \boldsymbol{B}(t_0 + c_k\tau, \boldsymbol{u}), \ t \in [t_0 + d_k\tau; \ t_0 + (d_k + b_k)\tau]. \end{aligned} \tag{3.14}$$

Note how c_k and d_k are switched in the time-freezing. Consequently, time-averaging is equivalent to

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix}\boldsymbol{u}\\ t_A\\ t_B\end{bmatrix}=\begin{bmatrix}\boldsymbol{A}(t_A)\boldsymbol{u}\\ 1\\ 0\end{bmatrix}+\begin{bmatrix}\boldsymbol{B}(t_B)\boldsymbol{u}\\ 0\\ 1\end{bmatrix}, \tag{3.15}$$

and freezing can be expressed as

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix}\boldsymbol{u}\\ t_A\\ t_B\end{bmatrix}=\begin{bmatrix}\boldsymbol{A}(t_A)\boldsymbol{u}\\ 0\\ 1\end{bmatrix}+\begin{bmatrix}\boldsymbol{B}(t_B)\boldsymbol{u}\\ 1\\ 0\end{bmatrix}. \tag{3.16}$$

3.3 ORDER CONDITIONS

Splitting and composition methods are defined by means of coefficients which define the fractional time steps of each basic scheme. They are obtained as the solutions of so-called *order conditions* that are a set of non-linear algebraic equations. There are two [29] commonly used techniques: based on the Baker–Campbell–Hausdorff (BCH) formula [114] and generalised B-series [86]. We will focus on the first one.

3.3.1 *Baker–Campbell–Hausdorff formula*

The formula can be generalised for operators

When matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ do not commute, it follows from (2.10) that $\mathrm{e}^{\boldsymbol{A}}\mathrm{e}^{\boldsymbol{B}} \neq \mathrm{e}^{\boldsymbol{A}+\boldsymbol{B}}$. The BCH formula expresses

$$\boldsymbol{C} = \ln\left(\mathrm{e}^{\boldsymbol{A}}\mathrm{e}^{\boldsymbol{B}}\right) \tag{3.17}$$

as a formal power series in $\boldsymbol{A}$ and $\boldsymbol{B}$. The BCH theorem states that

$$\boldsymbol{C} = \boldsymbol{A}+\boldsymbol{B}+\frac{1}{2}[\boldsymbol{A},\boldsymbol{B}]+\frac{1}{12}([\boldsymbol{A},[\boldsymbol{A},\boldsymbol{B}]]-[\boldsymbol{B},[\boldsymbol{A},\boldsymbol{B}]])+\ldots \tag{3.18}$$

Here square brackets stand for the default matrix commutator $[\boldsymbol{A},\boldsymbol{B}] := \boldsymbol{A}\boldsymbol{B}-\boldsymbol{B}\boldsymbol{A}$.

We can get its symmetric version, applying the formula twice:

$$\mathrm{e}^{\boldsymbol{S}} = \mathrm{e}^{\boldsymbol{A}/2}\mathrm{e}^{\boldsymbol{B}}\mathrm{e}^{\boldsymbol{A}/2}. \tag{3.19}$$

Due to symmetry, even elements will vanish:

$$\boldsymbol{S} = \boldsymbol{A}+\boldsymbol{B}-\frac{1}{24}[\boldsymbol{A},[\boldsymbol{A},\boldsymbol{B}]]-\frac{1}{12}[\boldsymbol{B},[\boldsymbol{B},\boldsymbol{A}]]+\ldots \tag{3.20}$$

3.3.2 *Order conditions via BCH*

We assume this form without loss of generality: a method could start with $\boldsymbol{B}$

Let us consider a composition for an autonomous split linear system (3.9):

$$\Psi_\tau^{[p]} = \mathrm{e}^{\tau b_s \boldsymbol{B}} \mathrm{e}^{\tau a_s \boldsymbol{B}} \dots \mathrm{e}^{\tau b_1 \boldsymbol{B}} \mathrm{e}^{\tau a_1 \boldsymbol{A}} \tag{3.21}$$

After the sequential application of the BCH formula to the right-hand side, we obtain $\Psi_\tau^{[p]} = \exp P_\tau^{[p]}$. The exponent $P_\tau^{[p]}$ is a power series in τ:

$$P_\tau^{[p]} = \tau(p_A \boldsymbol{A} + p_B \boldsymbol{B}) + \tau^2 p_{AB}[\boldsymbol{A}, \boldsymbol{B}] + \dots \tag{3.22}$$

Its coefficients $p_{IJK\dots}$ are multivariate polynomials in a_i and b_i. We want a method of order p:

$$\mathrm{e}^{P_\tau^{[p]}} = \mathrm{e}^{\tau(\boldsymbol{A}+\boldsymbol{B})} + \mathcal{O}\left(\tau^{p+1}\right). \tag{3.23}$$

Now we see that the *consistency conditions* $p_A = 1$, $p_B = 1$ should be satisfied and for the rest $p_{IJK\dots} = 0$.

There is one order condition for each element generated by $\boldsymbol{A}$ and $\boldsymbol{B}$ (more formally, for each basis element of the free Lie algebra, generated by these operators; see the next chapter). For a general, non-symmetric splitting method their number grows drastically with the desired order of a method. However, if we compose flows symmetrically by putting $b_s := 0$ and

$$\begin{gathered} \Psi_\tau^{[p]} = \mathrm{e}^{\tau a_s \boldsymbol{A}} \mathrm{e}^{\tau b_{s-1} \boldsymbol{B}} \dots \mathrm{e}^{\tau b_1 \boldsymbol{B}} \mathrm{e}^{\tau a_1 \boldsymbol{A}}, \\ a_i = a_{s+1-i},\ b_i = b_{s-i},\ i = 1 \dots s, \end{gathered} \tag{3.24}$$

then even order conditions are automatically satisfied.

In general, it is desirable to have only real positive coefficients to avoid backward integration in time, which can be ill-defined for some problems. Unfortunately, any splitting method of order $p \geq 3$ has negative coefficients [103, 110]; for an elegant proof of the statement we refer to [28]. Backward steps do not pose any limitation for reversible problems (Hamiltonian, in particular) but are still undesirable as they may lower computational efficiency.

Here we use the notation of [25]

Order conditions may be easier to solve if a problem possesses a favourable structure. Consider linear equations $\ddot{\boldsymbol{y}} = \boldsymbol{f}(t, \boldsymbol{y})$. For example, Hamiltonian systems with the kinetic energy quadratic in momenta belong to this class and are defined through $H(\boldsymbol{q}, \boldsymbol{p}) = \boldsymbol{p}^T \boldsymbol{p}/2 + V(\boldsymbol{q}) =: \boldsymbol{A}(\boldsymbol{p}) + \boldsymbol{B}(\boldsymbol{q})$. Simplification occurs thanks to $[\boldsymbol{B}, [\boldsymbol{B}, [\boldsymbol{B}, \boldsymbol{A}]]] \equiv \boldsymbol{O}$. Furthermore, take $\ddot{\boldsymbol{y}} = \boldsymbol{N}\boldsymbol{y}$ equivalently described by $H(\boldsymbol{q}, \boldsymbol{p}) = \boldsymbol{p}^T \boldsymbol{p}/2 + \boldsymbol{q}^T \boldsymbol{N} \boldsymbol{q}/2$. It can be represented as a 1^{st}-order system (2.7):

$$\dot{\boldsymbol{u}} = \left(\underbrace{\begin{bmatrix} \boldsymbol{O} & \boldsymbol{I} \\ \boldsymbol{O} & \boldsymbol{O} \end{bmatrix}}_{=:\boldsymbol{A}} + \underbrace{\begin{bmatrix} \boldsymbol{O} & \boldsymbol{O} \\ \boldsymbol{N} & \boldsymbol{O} \end{bmatrix}}_{=:\boldsymbol{B}} \right) \boldsymbol{u}, \tag{3.25}$$

in which the order conditions for $[\boldsymbol{A},[\boldsymbol{A},[\boldsymbol{A},\boldsymbol{B}]]] \equiv \boldsymbol{O}$ are also satisfied automatically.

EXAMPLE: SYMMETRIC RKN METHODS Consider symmetric compositions (3.24) for Hamiltonian systems written as (3.25). Efficient methods of this form were proposed in [25]. The 4^{th}-order 6-stage method $\text{RKN}_6^{[4]}$ is defined by

$$\begin{aligned} a_1 &= a_6 = 0.245298957184271 \\ a_2 &= a_5 = 0.604872665711080 \\ a_3 &= a_4 = 1/2 - (b_1 + b_2); \end{aligned} \qquad \begin{aligned} b_1 &= b_7 = \ \ 0.082984406417405 \\ b_2 &= b_6 = \ \ 0.396309801498368 \\ b_3 &= b_5 = -0.039056304922349 \\ b_4 &= 1 - 2(a_1 + a_2 + a_3); \end{aligned} \tag{3.26}$$

and the 6^{th}-order 11-stage $\text{RKN}_{11}^{[6]}$ has the following coefficients:

$$\begin{aligned} a_1 = a_{11} &= \ \ 0.123229775946271 & b_1 = b_{12} &= \ \ 0.041464998518262 \\ a_2 = a_{10} &= \ \ 0.290553797799558 & b_2 = b_{11} &= \ \ 0.198128671918067 \\ a_3 = a_9 &= -0.127049212625417 & b_3 = b_{10} &= -0.040006192104153 \\ a_4 = a_8 &= -0.246331761062075 & b_4 = b_9 &= \ \ 0.075253984301581 \\ a_5 = a_7 &= \ \ 0.357208872795928 & b_5 = b_8 &= -0.011511387420688 \\ a_6 &= 1 - 2\textstyle\sum_{i=1}^{5} b_i; & b_6 = b_7 &= 1/2 - \textstyle\sum_{i=1}^{5} a_i. \end{aligned} \tag{3.27}$$

4 MAGNUS EXPANSION-BASED INTEGRATORS

This chapter is dedicated to the main tools used for the design of the integrators in the present thesis: Lie groups and the Magnus expansion (ME). The ME is one of the options to express the solution of a time-dependent problem as an infinite series. It is chosen for the following important property: if the solution belongs to some Lie group, any ME truncation is in this group too. Consequently, the geometric structure can be preserved.

4.1 LIE GROUPS AND LIE ALGEBRAS

4.1.1 *Lie groups*

A *group* G is a set with a binary operation $\times$ (usually called multiplication), obeying the following rules:

- $a \times b \in G$ if $a, b \in G$ (closure);
- $a \times (b \times c) = (a \times b) \times c \ \forall a, b, c \in G$ (associativity);
- $\exists e \in G\colon\ e \times a \equiv a \times e \equiv a \ \forall a \in G$ (identity element);
- for any $a \in G\ \exists a^{-1}\colon\ a \times a^{-1} \equiv a^{-1} \times a \equiv e$ (inverse elements).

A *Lie group* G is a group with the structure of a differentiable manifold, such that the mapping $\mu\colon\ G \times G \to G$ defined as $\mu(a, b) \to a\ b^{-1}$ is differentiable [114].

In general, Lie groups are complicated objects, but their study can be simplified by considering a connected entity called *Lie algebras*. Omitting most formal definitions, the correspondence is expressed by the exponential mapping $\exp\colon\ \mathfrak{g} \to G$ which associates with an element $\alpha \in \mathfrak{g}$ the element $x(1)$ of its integral curve $\dot{x} = \alpha x(t)$, $x(0) = e$ [90].

4.1.2 *Lie algebras*

A *Lie algebra* $\mathfrak{g}$ is a vector space, endowed with a bilinear mapping $\mathfrak{g} \times \mathfrak{g} \to \mathfrak{g}$, satisfying the following properties for $\forall \alpha, \beta, \gamma \in \mathfrak{g}$:

- $[\alpha, \beta] = -[\beta, \alpha]$ (skew symmetry; anticommutativity);
- $[\alpha, \alpha] \equiv 0$ (alternation);
- $[\alpha, [\beta, \gamma]] + [\beta, [\gamma, \alpha]] + [\gamma, [\alpha, \beta]] \equiv 0$ (Jacobi identity).

For any element $\chi \in \mathfrak{g}$ the *adjoint action* [114] can be defined by means of commutators:

$$\begin{aligned} &\mathrm{ad}_\chi\, \alpha := [\chi, \alpha], \\ &\mathrm{ad}_\chi^0\, \alpha := \alpha,\ \mathrm{ad}_\chi^k\, \alpha := [\chi, \mathrm{ad}_\chi^{k-1}\, \alpha]. \end{aligned} \tag{4.1}$$

It has the following properties:

- $\mathrm{ad}_{[\alpha,\beta]} \equiv [\mathrm{ad}_\alpha, \mathrm{ad}_\beta]$,
- $\mathrm{ad}_\chi\, [\alpha, \beta] \equiv [\alpha, \mathrm{ad}_\chi\, \beta] + [\mathrm{ad}_\chi\, \alpha, \beta]$.

We use lowercase fraktur for Lie algebras

A Lie algebra $\mathfrak{f}$ is called *free* over the index set I [94] if

- to any index $i \in I$ corresponds an element $\alpha_i \in \mathfrak{f}$;
- for any Lie algebra $\mathfrak{g}$ and mapping $i \to \beta_i \in \mathfrak{g}$ there exists a unique Lie algebra homomorphism $\phi : \mathfrak{f} \to \mathfrak{g}$, such that $\phi(\alpha_i) = \beta_i\ \forall i \in I$.

The definition means that a free Lie algebra only possesses properties, common for any Lie algebra, and there are no additional assumptions. Consequently, $\mathfrak{f}$ may be considered a universal object, and computations in $\mathfrak{f}$ can be transferred to $\mathfrak{g}$ by the homomorphism ϕ.

$[\mathfrak{g}_i, \mathfrak{g}_j]$ means all possible commutators $[\alpha_i, \beta_j]$ of $\alpha_i \in \mathfrak{g}_i$ and $\beta_i \in \mathfrak{g}_j$

A Lie algebra $\mathfrak{g}$ is called *graded* by an Abelian (commutative) group a [116] if it can be represented as a direct sum of subspaces $\mathfrak{g}_i$, $i \in A$, in such a way that $[\mathfrak{g}_i, \mathfrak{g}_j] \subseteq \mathfrak{g}_{i+j}$. As an example, consider scaled elements $\tau^i \alpha_i$ of a free Lie algebra $\mathfrak{f}$. Since $[\tau^i \alpha_i, \tau^j \alpha_j] = \tau^{i+j} [\alpha_i, \alpha_j] \in \mathfrak{g}$, they generate a graded free Lie algebra.

4.1.2.1 *Matrix Lie groups and their algebras*

For our goals the following *matrix Lie groups* are of major interest:

THE GENERAL LINEAR GROUP $\mathrm{GL}(d)$ consist of $d \times d$ invertible square matrices. The elements of its corresponding Lie algebra $\mathfrak{gl}(d)$ matrices with zero trace.

THE SYMPLECTIC GROUP $\mathrm{Sp}(2d)$ arises from both classical and quantum mechanics. It consists of $2d \times 2d$ matrices $\boldsymbol{M}$, such that $\boldsymbol{M}^T \boldsymbol{J} \boldsymbol{M} = \boldsymbol{J}$, where $\boldsymbol{J}$ is the canonical symplectic matrix (2.5). Hamiltonian matrices $\boldsymbol{H}$: $\boldsymbol{J}\boldsymbol{H} + \boldsymbol{H}^T \boldsymbol{J} \equiv \boldsymbol{O}$ form the corresponding algebra $\mathfrak{sp}(2d)$.

THE SPECIAL UNITARY GROUP $\mathrm{SU}(d)$ is the group of unitary $d \times d$ matrices with determinant equal to 1. Its algebra $\mathfrak{su}(d)$ is formed by traceless skew-Hermitian matrices. $\mathrm{SU}(2)$ and $\mathrm{SU}(3)$ are widely used in quantum physics.

4.1.2.2 *Lie algebra bases and BCH formula*

By definition, Lie algebras consists of *generators* and all their possible commutators. Working with a free algebra may be cumbersome due to the redundant elements generated by the Jacobi identity

$$[\alpha,[\beta,\gamma]]+[\beta,[\gamma,\alpha]]+[\gamma,[\alpha,\beta]]\equiv 0.$$

Nevertheless, Lie algebra manipulation can be facilitated by the introduction of an appropriate basis.

Different approaches are possible but the most used are: Lyndon–Shirshov words [94, 104], right-nested (right-normed) commutator basis [8, 43] and Hall sets [94], which have well-defined generating procedures.

Assume $\boldsymbol{A}$ and $\boldsymbol{B}$ be matrices of some Lie algebra. Since $\boldsymbol{C}$ in the BCH formula $\boldsymbol{C}=\ln\left(\mathrm{e}^{\boldsymbol{A}}\mathrm{e}^{\boldsymbol{B}}\right)$ belongs to the same algebra, representation in the Hall basis can simplify the calculation of order conditions [40].

4.2 MAGNUS EXPANSION

Consider an IVP for a linear differential equation on a Lie group G (albeit valid in general):

$$\dot{\boldsymbol{u}}=\boldsymbol{Z}(t)\boldsymbol{u}(t),\ \boldsymbol{u}(t_0)=\boldsymbol{u}_0\in G,\ \boldsymbol{Z}\in\mathfrak{g}, \tag{4.2}$$

to which different equations can be reduced as shown in Chapter 2. We are interested in cases when $\boldsymbol{u}(t)$ is a real- or complex-valued vector function. For convenience, in this section we will consider the equivalent fundamental matrix solution $\boldsymbol{\Phi}(t)$, such that $\boldsymbol{u}(t)=\boldsymbol{\Phi}(t)\boldsymbol{u}(t_0)$ and $\boldsymbol{\Phi}(t_0)=\boldsymbol{I}$.

The solution of (4.2) cannot be expressed in a closed form, except for isolated cases when the system matrix commutes with its integral (called *the Lappo-Danilevskii condition* [68]):

$$\left[\boldsymbol{Z}(t),\int_{t_0}^{t}\boldsymbol{Z}(s)\,\mathrm{d}s\right]\equiv\boldsymbol{O},\ t\geq t_0 \tag{4.3}$$

and the solution is

$$\boldsymbol{\Phi}(t)=\mathrm{e}^{\int_{t_0}^{t}\boldsymbol{Z}(s)\mathrm{d}s}\boldsymbol{\Phi}(t_0). \tag{4.4}$$

Nevertheless, for a sufficiently small time interval, the solution of (4.2) can be expressed by means of the Magnus expansion (ME) $\Omega(t;t_0)$ that is the exponent of the solution $\boldsymbol{\Phi}(t)=\exp\Omega(t;t_0)$. We will omit the second argument t_0, which represents the initial time, unless it causes confusion.

4.2.1 *Existence and properties*

THE EXISTENCE and uniqueness of the solution is guaranteed by the Peano theorem and the regularity of the exponential mapping from $\mathfrak{g}$ to G. Its form $\boldsymbol{Z}(t) = \exp\Omega(t)$ is due to $\det \boldsymbol{Z}(t) \neq 0$ [23]. The following result explains how to obtain this solution:

Theorem (Magnus [74]). *Let $\boldsymbol{Z}(t)$ be a known function of t, and let $\boldsymbol{\Phi}(t)$ be an unknown function such that $\boldsymbol{\Phi}(t_0) = \boldsymbol{I}$. Then, if certain unspecified conditions of convergence are satisfied, $\boldsymbol{\Phi}(t)$ can be written as*

$$\boldsymbol{\Phi}(t) = \mathrm{e}^{\Omega(t)} \tag{4.5}$$

where

$$\frac{\mathrm{d}\Omega(t)}{\mathrm{d}t} = \sum_{k=0}^{\infty} \frac{B_k}{k!} \operatorname{ad}_{\Omega}^{k} \boldsymbol{Z}(t), \tag{4.6}$$

$\operatorname{ad}_{\chi}\alpha := [\chi, \alpha]$

B_k is the k^{th} Bernoulli number [87], and $\operatorname{ad}_{\Omega}$ is the adjoint action. Integration of (4.6) leads to an infinite series for $\Omega(t)$:

$$\Omega = \sum_{k=0}^{\infty} \Omega_k. \tag{4.7}$$

The theorem can be reformulated and proved in terms of the $\mathrm{d}\exp$-inv operator as following:

Theorem ([23]). *the solution of eq. (4.2) with initial condition $\boldsymbol{\Phi}(t_0) = \boldsymbol{I}$ can be written as $\boldsymbol{\Phi}(t) = \mathrm{e}^{\Omega(t)}$ with $\Omega(t)$ defined by*

$$\frac{\mathrm{d}\Omega(t)}{\mathrm{d}t} = \mathrm{d}\exp_{\Omega}^{-1} \boldsymbol{Z}(t),\ \Omega(t_0) = \boldsymbol{I}, \tag{4.8}$$

where

$$\mathrm{d}\exp_{\Omega}^{-1} \boldsymbol{Z}(t) = \sum_{k=0}^{\infty} \frac{B_k}{k!} \operatorname{ad}_{\Omega}^{k} \boldsymbol{Z}(t).$$

The first three terms of eq. (4.8) are

$$\frac{\mathrm{d}\Omega(t)}{\mathrm{d}t} = \boldsymbol{Z} - \frac{1}{2}[\Omega(t), \boldsymbol{Z}(t)] + \frac{1}{12}[\Omega(t), [\Omega(t), \boldsymbol{Z}(t)]] + \ldots$$

Putting $\Omega_{(0)} := \boldsymbol{O}$, we apply the Picard iteration $\Omega_{(k+1)} = \int_{t_0}^{t} \Omega'_{(k)}(s)\mathrm{d}s$ and get

$$\Omega_{(k)} = \int_{t_0}^{t} \Big(\boldsymbol{Z}(s) - \frac{1}{2}\big[\Omega_{(k-1)}(s), \boldsymbol{Z}(s)\big] + \frac{1}{12}\Big[\Omega_{(k-1)}(s), \big[\Omega_{(k-1)}(s), \boldsymbol{Z}(s)\big]\Big] + \ldots \Big) \mathrm{d}s$$

which converges to $\Omega(t)$ as $n \to \infty$ in an appropriately small neighbourhood. Note that the Lappo-Danilevskii condition corresponds to the case when $\Omega_{(k)} \equiv \boldsymbol{O}$, $k \geq 2$.

On the other hand, according to the Magnus's theorem, the expansion can be represented as a series

$$\Omega(t) = \sum_{k=0}^{\infty} \Omega_n(t).$$

Its terms are expressed by means of the recursive generator $S_n^{(k)}$ [67]:

$$\begin{aligned} S_n^{(1)} &= [\Omega_{n-1}, \boldsymbol{Z}], \quad S_n^{(n-1)} = \mathrm{ad}_{\Omega_1}^{n-1} \boldsymbol{Z}; \\ S_n^{(k)} &= \sum_{m=1}^{n-j} \left[\Omega_m, S_{n-m}^{(k-1)} \right], \ 2 \leq k \leq n-1. \end{aligned} \tag{4.9}$$

Then

$$\Omega_1 = \int_{t_0}^{t} \boldsymbol{Z}(s)\mathrm{d}s, \quad \Omega_n = \sum_{k=1}^{n-1} \frac{B_k}{k!} \int_{t_0}^{t} S_n^{(k)}(s)\mathrm{d}s, \ k \geq 2.$$

In particular, the first three terms are:

$$\begin{aligned} &\Omega_1 = \int_{t_0}^{t} \boldsymbol{Z}(s)\mathrm{d}s; \qquad \Omega_2 = \frac{1}{2} \int_{t_0}^{t} \int_{t_0}^{s_1} [\boldsymbol{Z}(s_1), \boldsymbol{Z}(s_2)]\mathrm{d}s_2\mathrm{d}s_1; \\ &\Omega_3 = \frac{1}{6} \int_{t_0}^{t} \int_{t_0}^{s_1} \int_{t_0}^{s_2} [\boldsymbol{Z}(s_1), [\boldsymbol{Z}(s_2), \boldsymbol{Z}(s_3)]] + [[\boldsymbol{Z}(s_1), \boldsymbol{Z}(s_2)], \boldsymbol{Z}(s_3)]\mathrm{d}s_3\mathrm{d}s_2\mathrm{d}s_1. \end{aligned} \tag{4.10}$$

THE CONVERGENCE of the ME was thoroughly researched. The following theorem provides sufficient conditions for a generic bounded operator.

Theorem ([23, 39, 82]). *Let us consider eq.* (4.2) *with* $\boldsymbol{\Phi}(t_0) = \boldsymbol{I}$, *defined in a Hilbert space, and let* $\boldsymbol{Z}(t)$ *be a bounded operator in this space. Then the* ME *converges for* $t \in [t_0;\ t)$ *such that*

$$\int_{t_0}^{t} \left\| \boldsymbol{Z}(s) \right\| \mathrm{d}s \leq \pi, \tag{4.11}$$

and $\mathrm{e}^{\Omega(t)} = \boldsymbol{\Phi}(t)$.

It is important to note that for some operators the convergence time interval can be larger.

In the context of numerical integration, the local convergence of the ME makes it necessary to split the integration interval $[t_0;\ t_f]$ into subintervals $[t_{k-1};\ t_k]$ of length τ (for simplicity assume it to be constant as in Section 1.1), such that the expansion converges in each of them. Then, the solution is propagated from one time point to another.

LIE ALGEBRA OF Ω If $\boldsymbol{Z}$ belongs to some Lie algebra $\mathfrak{g}$, then $\Omega(t)$ and any truncation, containing elements Ω_k, belongs to $\mathfrak{g}$. For this reason, the ME is a good starting point for the construction of geometric integrators: it will preserve the group properties of the solution.

TIME SYMMETRY is another favourable property of the ME. Recall that the exact flow φ_t (page 7), which defines the solution $\varphi_t\colon G \to G$, is time symmetric. In terms of the ME, it is expressed as

$$\Omega(t;t_0) = -\Omega(t_0;t), \tag{4.12}$$

and implies that $\Omega(t)$ can be expanded in even powers of t, which simplifies the construction of time-symmetric integrators.

4.3 APPLICATION TO THE CONSTRUCTION OF INTEGRATORS

The solution of eq. (4.2) over a succession of steps can be expressed as

$$\boldsymbol{u}(t) = \prod_{k=1}^{f} \mathrm{e}^{\Omega(t_k;t_{k-1})}\boldsymbol{u}(t_0). \tag{4.13}$$

The general procedure to design a numerical method for finding $\boldsymbol{u}_f$ is as follows:

1. take a feasible truncation $\Omega^{[p]} := \sum_{i=1}^{p} \Omega_i$ of the infinite ME series;
2. approximate the multivariate integrals of commutators in $\Omega^{[p]}$;
3. select a suitable exponentiation procedure.

The truncation depends on the desired order of a method, and it shares geometric properties with the expansion itself, as it has been shown in the previous section. An algorithm for the exponential (recall Section 2.4) is chosen based on the information we want to get from the solution and the problem's properties. Now, we will discuss how to treat the integrals in the Magnus truncation appropriately.

4.3.1 *Approximation with moment integrals*

The main difficulty in applying the ME directly to get the solution in form of (4.5) is to calculate multiple integrals of nested commutators,

as it is seen from explicit expressions (4.10) for Ω_i. However, thanks to the underlying structure of the problem, this task can be carried out by calculating a set of univariate integrals of $\boldsymbol{Z}(t)$ [59].

To this end, define the k^{th} central moments on one subinterval of length $\tau := t_n - t_{n-1}$ as:

$$\boldsymbol{Z}^{(k)} = \int_{t_{n-1}}^{t_n} \left(\frac{t - (t_n + \tau/2)}{\tau} \right)^k \boldsymbol{Z}(t) \mathrm{d}t. \tag{4.14}$$

Now, for convenience, we will change the variable t to work in the 'handbook' symmetric interval $[-1;\ 1]$ instead of $[t_{n-1};\ t_n]$:

$$\hat{t} = 2\frac{t - (t_{n-1} + \tau)}{\tau} \leftrightarrow t = \frac{\tau\hat{t} + (2t_{n-1} + \tau)}{2}.$$

Consequently, (4.14) becomes

$$\boldsymbol{Z}^{(k)} = \int_{-1}^{1} \frac{\tau}{2} \left(\frac{\hat{t}}{2} \right)^k \boldsymbol{Z}(\hat{t}) \mathrm{d}\hat{t}.$$

By approximating the integral in the k^{th} moment with a q-point GL quadrature rule of order $p = 2q$ we get:

$$\boldsymbol{Z}^{(k)[p]} = \sum_{j=1}^{q} \frac{\tau}{2} \left(\frac{\hat{c}_j}{2} \right)^k \hat{w}_j \boldsymbol{Z}(\hat{c}_j), \tag{4.15}$$

where $\hat{c}_j$ are the nodes and $\hat{w}_j$ are the weight on $[-1;\ 1]$.

To get a feasible numerical method, let us consider the $(q-1)$-grade Lagrange interpolating polynomial of $\boldsymbol{Z}(\hat{t})$ at the nodes $\hat{c}_j$:

$$\begin{gathered} \widetilde{\boldsymbol{Z}}(\hat{t}) := \sum_{i=1}^{q} \boldsymbol{L}_i(\hat{t}) \boldsymbol{Z}(\hat{c}_i), \\ \text{where } \boldsymbol{L}_i(\hat{t}) = \prod_{\substack{j=1 \\ j \neq i}}^{q} \frac{\hat{t} - \hat{c}_j}{\hat{c}_i - \hat{c}_j},\ \hat{t} \in [-1;\ 1]. \end{gathered} \tag{4.16}$$

The Alekseev–Gröbner lemma [27, 59, 81] guarantees that the solution $\boldsymbol{u}(t)$ of the initial problem and the solution $\tilde{\boldsymbol{u}}(t)$ of the modified one with $\widetilde{\boldsymbol{Z}}(t)$ satisfy $\boldsymbol{u}(t) - \tilde{\boldsymbol{u}}(t) = \mathcal{O}\left(\tau^{p+1}\right)$ [17].

Afterwards, we take the Maclaurin series of $\widetilde{\boldsymbol{Z}}(t)$:

$$\widetilde{\boldsymbol{Z}}\left(\frac{\tau\hat{t} + (2t_{n-1} + \tau)}{2} \right) = \sum_{l=0}^{q-1} \frac{1}{l!} \left(\frac{\tau}{2} \right)^l \frac{\mathrm{d}^l \widetilde{\boldsymbol{Z}}(\hat{t})}{\mathrm{d}\hat{t}^l} \hat{t}^l \tag{4.17}$$

and define the following elements [27]:

$$\zeta_l \coloneqq \tau^l \frac{1}{(l-1)!} \left. \frac{\mathrm{d}^{l-1} \widetilde{\boldsymbol{Z}}(\hat{t})}{\mathrm{d}\hat{t}^{l-1}} \right|_{\hat{t}=0} = \mathcal{O}\left(\tau^l\right),\ l = 1,\dots,q, \tag{4.18}$$

which are the generators of a graded Lie algebra (as on page 24).

Finally, we can combine (4.15), (4.17) and (4.18), which yields an expression for the moments $\boldsymbol{Z}^{(k)}$ in terms of the generators ζ_i:

$$\begin{aligned} \boldsymbol{Z}^{(k)[p]} &= \sum_{j=1}^{q} \frac{\tau}{2} \left(\frac{\hat{c}_j}{2}\right)^k \hat{w}_j \sum_{l=0}^{q-1} \left(\frac{\tau}{2}\right)^l \frac{1}{l!} \frac{\mathrm{d}^l \widetilde{\boldsymbol{Z}}(\hat{t})}{\mathrm{d}\hat{t}^l} \hat{c}_j^l \\ &= \sum_{j=1}^{q} \left(\frac{\hat{c}_j}{2}\right)^k \hat{w}_j \sum_{l=0}^{q-1} \left(\frac{\tau}{2}\right)^{l+1} \frac{1}{l!} \frac{\mathrm{d}^l \widetilde{\boldsymbol{Z}}(\hat{t})}{\mathrm{d}\hat{t}^l} \hat{c}_j^l \\ &= \sum_{j=1}^{q} \left(\frac{\hat{c}_j}{2}\right)^k \hat{w}_j \sum_{l=1}^{q} \left(\frac{1}{2}\right)^l \hat{c}_j^{l-1} \zeta_l. \end{aligned} \tag{4.19}$$

Henceforth, we will consider the original interval $[t_{n-1};\ t_{n-1}+\tau]$ and it is convenient to rescale the handbook quadrature nodes and weights to $[0;\ 1]$. For compactness, we introduce a shorthand for function values at the nodes:

$$\boldsymbol{Z}_j \coloneqq \boldsymbol{Z}\left(t_{n-1} + \frac{\tau}{2}\left(c_j + 1\right)\right) \text{ in } t \leftrightarrow \boldsymbol{Z}_j \coloneqq \boldsymbol{Z}\left(\hat{c}_j\right) \text{ in } \hat{t}. \tag{4.20}$$

With this notation, we can write the inverse of (4.19) as the formal product of a matrix and a vector of matrices:

$$\zeta_i = \tau \sum_{j=1}^{q} s_{i,j} \boldsymbol{Z}_i,\ i = 1,\dots,q, \tag{4.21}$$

where $s_{i,j}$ are the elements of the transition matrix $\boldsymbol{S} \coloneqq \boldsymbol{T}^{-1}\boldsymbol{Q}$. Matrices $\boldsymbol{T}$ and $\boldsymbol{Q}$ are defined elementwise as

$$(\boldsymbol{T})_{i,j} = \frac{1}{2} \int_{-1}^{1} \left(\frac{\hat{t}}{2}\right)^{i+j-2} \mathrm{d}\hat{t}, \quad (\boldsymbol{Q})_{i,j} = \hat{w}_j \left(\frac{\hat{c}_j}{2}\right)^i, \quad \begin{matrix} i = 1,\dots,q; \\ j = 1,\dots,q. \end{matrix} \tag{4.22}$$

For 2- and 3-point GL rules the matrices are:

$$\boldsymbol{S}_{2\times 2} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ -\sqrt{3} & \sqrt{3} \end{bmatrix}, \quad \boldsymbol{S}_{3\times 3} = \begin{bmatrix} 0 & 1 & 0 \\ -\frac{\sqrt{15}}{3} & 0 & \frac{\sqrt{15}}{3} \\ \frac{10}{3} & -\frac{20}{3} & \frac{10}{3} \end{bmatrix}. \tag{4.23}$$

It is easy to see that the elements of every row, except for the first, sum up to zero.

Alternatively, ζ_i can be expressed by means of, for example, shifted Legendre polynomials [5]. The transition between the representations (4.21) and the one from [5] can be done by expressing the latter through central moments.

TIME–SPACE SEPARABLE SYSTEMS are rather common in applications, when a system's matrix has an additively–multiplicatively separable structure with respect to time t and space x:

Space coordinate x is one-dimensionl for simplicity

$$\boldsymbol{Z}(t,x) = \boldsymbol{A}(t) + \boldsymbol{B}(x) + f(t)\boldsymbol{D}(x), \tag{4.24}$$

with $f(t)$ being a scalar function that represents, say, an external control. If quadrature rule nodes are symmetric with respect to the centre of the time interval, then from (4.21) and (4.22) it follows that

$$\zeta_i = \tau \sum_{j=1}^{q} s_{i,j}\Big(\boldsymbol{A}_j + f_j \boldsymbol{D}(x)\Big),\ i \geq 2. \tag{4.25}$$

Furthermore, when $\boldsymbol{A}$ is absent, the calculation becomes simpler: it reduces to scalar linear combinations of $f(t)$ with subsequent multiplication by $\boldsymbol{D}$:

$$\zeta_i = \tau \left(\sum_{j=1}^{q} s_{i,j} f_j \right) \boldsymbol{D}(x),\ i \geq 2. \tag{4.26}$$

These relations mean that when $\boldsymbol{Z}$ contains an autonomous additive part $\boldsymbol{B}$, e. g., $\boldsymbol{Z}(t,x) = \boldsymbol{B}(x) + f(t)\boldsymbol{D}(x)$, it only persists in the first element, ζ_1. This fact plays a major part in Hamiltonian problems with no time dependence in the kinetic energy.

As a short illustration, we write out the elements with the 4^{th}-order and the 6^{th}-order GL quadrature rules. From (4.23) it follows immediately that

$$\begin{aligned} \zeta_1 &= \tau \frac{1}{2}(\boldsymbol{Z}_1 + \boldsymbol{Z}_2) = \mathcal{O}(\tau), \\ \zeta_2 &= \tau\sqrt{3}(\boldsymbol{Z}_2 - \boldsymbol{Z}_1) = \mathcal{O}\big(\tau^2\big); \end{aligned} \qquad \begin{aligned} \zeta_1 &= \tau \boldsymbol{Z}_2 = \mathcal{O}(\tau), \\ \zeta_2 &= \tau \frac{\sqrt{15}}{3}(\boldsymbol{Z}_3 - \boldsymbol{Z}_1) = \mathcal{O}\big(\tau^2\big), \\ \zeta_3 &= \tau \frac{10}{3}(\boldsymbol{Z}_3 - 2\boldsymbol{Z}_2\boldsymbol{Z}_1) = \mathcal{O}\big(\tau^3\big) \end{aligned} \tag{4.27}$$

4.3.2 *Magnus expansion in terms of generators*

Ultimately, we can express the ME in terms of free Lie algebra generators ζ_i. In this work we present methods up to order 8, for which it is sufficient to take the 4-point GL rule and 4 generators [23, 27, 85]. In the representation below, we group elements and commutators by order and the corresponding number of generators.

The ME truncation then reads [17]:

$$\begin{aligned}\Omega^{[8]} = &\overbrace{\underbrace{\zeta_1 - \frac{1}{12}[12] + \frac{1}{12}\zeta_3 + \frac{1}{240}[23] + \frac{1}{360}[113] - \frac{1}{240}[212] + \frac{1}{720}[1112]}_{\text{order 6 with 3 gen.}}}^{\text{order 4 with 2 gen.}} \\ &- \frac{1}{80}[14] - \frac{1}{1344}[34] - \frac{1}{2240}[124] + \frac{1}{6720}[223] + \frac{1}{6048}[313] - \frac{1}{840}[412] \\ &+ \frac{1}{6720}[1114] - \frac{1}{7560}[1123] + \frac{1}{4032}[1312] + \frac{11}{60480}[2113] - \frac{1}{6720}[2212] \\ &- \frac{1}{15120}[11113] - \frac{1}{30240}[11212] + \frac{1}{7560}[21112] - \frac{1}{30240}[111112],\end{aligned} \tag{4.28}$$

where $[i_1 \dots i_{k-1} i_k]$ is a compact form to write the right-nested commutator $[\zeta_{i_1}, [\dots, [\zeta_{i_{k-1}}, \zeta_{i_k}]] \dots]$.

EXAMPLE: MAGUNS-BASED METHODS Let us examine three simple 4^{th}-order methods which use elements (4.27).

The simplest symmetric method of this family is the exponential midpoint

$$\mathrm{MP}^{[2]} = \exp(\zeta_1) = \exp\Big(\tau \boldsymbol{Z}\Big(t_{k-1} + \frac{\tau}{2}\Big)\Big). \tag{4.29}$$

The second is the Magnus method, obtained by the corresponding truncation of the ME:

$$\mathrm{MM}^{[4]} = \exp\Big(\zeta_1 - \frac{1}{12}[\zeta_1, \zeta_2]\Big) = \exp\Big(\frac{\tau}{2}(\boldsymbol{Z}_1 + \boldsymbol{Z}_2) + \frac{\tau^2\sqrt{3}}{12}[\boldsymbol{Z}_1, \boldsymbol{Z}_2]\Big).$$

Provided that $\boldsymbol{Z}_i$ are already evaluated, this method requires 2 matrix–matrix products (MMPs) and 1 exponentiation to propagate the fundamental matrix solution. However, if a state vector propagation is considered, then 5 matrix–vector products (MVPs) are necessary at each iteration of the Krylov subspace approximation process.

Alternatively, we can obtain a 4^{th}-order method without commutators:

$$\mathrm{CF}_2^{[4]} = \exp\Big(\frac{1}{2}\zeta_1 - \frac{1}{6}\zeta_2\Big)\exp\Big(\frac{1}{2}\zeta_1 + \frac{1}{6}\zeta_2\Big) = \mathrm{MM}^{[4]} + \mathcal{O}\big(\tau^5\big). \tag{4.30}$$

Compared to the straightforward Magnus method, this one requires 2 exponentials and 1 product of the exponentials. On the other hand, its matrix–vector cost is 4 per iteration.

This scheme is belongs to the efficient commutator-free (CF) family mentioned in the thesis introduction. In [5] it is defined as

$$\mathrm{CF}_2^{[4]} = \exp\big(f_{1,1}\check{\boldsymbol{Z}}_1 + f_{1,2}\check{\boldsymbol{Z}}_2\big)\exp\big(f_{1,1}\check{\boldsymbol{Z}}_1 - f_{1,2}\check{\boldsymbol{Z}}_2\big)$$

in terms of the matrix-valued coefficients $\check{\boldsymbol{Z}}_i$ which define the expansion of $\boldsymbol{Z}(t)$ into the series of shifted Legendre polynomials. We can make

the transition to our formalism of the central moments in the following manner:

$$\begin{bmatrix} a_{1,1} & a_{1,2} \\ a_{2,1} & a_{2,2} \end{bmatrix} = \underbrace{\begin{bmatrix} f_{1,1} & f_{1,2} \\ f_{2,1} & f_{2,2} \end{bmatrix}}_{:=\boldsymbol{F}} \underbrace{\begin{bmatrix} 1 & 0 \\ 0 & 6 \end{bmatrix}}_{:=\boldsymbol{G}_{2\times 2}} \underbrace{\begin{bmatrix} q_{1,1} & q_{1,2} \\ q_{2,1} & q_{2,2} \end{bmatrix}}_{:=\boldsymbol{Q}_{2\times 2}}.$$

Here $\boldsymbol{F}$ is the initial matrix of coefficients; $\boldsymbol{G}$ provides transition from the Legendre polynomials to the central moments (4.14); $\boldsymbol{Q}$ is defined by (4.22). The result is

$$\begin{aligned} \mathrm{CF}_2^{[4]} = \exp\left(\tau\left(\frac{1}{4} - \frac{\sqrt{3}}{6}\right)\boldsymbol{Z}_1 + \tau\left(\frac{1}{4} + \frac{\sqrt{3}}{6}\right)\boldsymbol{Z}_2\right) \\ \times \exp\left(\tau\left(\frac{1}{4} + \frac{\sqrt{3}}{6}\right)\boldsymbol{Z}_1 + \tau\left(\frac{1}{4} - \frac{\sqrt{3}}{6}\right)\boldsymbol{Z}_2\right). \end{aligned}$$

5

SCHRÖDINGER EQUATION

The exposition in this chapter is based on the article [16]

In this chapter we construct unitarity-preserving geometric integrators for the non-autonomous Schrödinger equation (SE) with the Hamiltonian given as the sum of the kinetic energy and a time-dependent potential. The new methods belong to the class of *quasi-commutator-free (QCF) quasi-Magnus* [33] exponential integrators.

Regarding the possible structure of methods, our considerations are the following. First of all, the use of the Magnus expansion (ME) yields a numerical method which is 'better' in a sense: if a problem is autonomous, then the ME provides the exact analytical solution.

Secondly, we are aiming to improve upon efficient, but general CF methods [5, 10, 26, 111] by optimizing our schemes, using the structure of the problem.

Finally, we do not concentrate on the splitting approach from Chapter 3 because it may cause numerical instabilities for large time steps if both the kinetic and potential energies have large values but cancel each other when summed.

THE SCHRÖDINGER EQUATION is the cornerstone of quantum mechanics. Its most general form with variable coefficients, written in units, such that the reduced Planck constant $\hbar = 1$, reads

Upright roman i *stands for the imaginary unit:* $(\pm\mathrm{i})^2 = -1$

$$\mathrm{i}\frac{\partial\psi(t,\boldsymbol{x})}{\partial t} = H(t,\boldsymbol{x})\psi(t,\boldsymbol{x}),$$
$$\psi(t_0,\boldsymbol{x}) = \psi_0(\boldsymbol{x}), \quad t \in \mathbb{R},\ \boldsymbol{x} \in \mathbb{R}^\delta,$$

where $H(t,\boldsymbol{x})$ is a Hermitian operator. We assume that the wave function $\psi(t,\boldsymbol{x})$ and its derivatives vanish at the boundaries of the region of interest, thus periodic boundary conditions can be considered.

According to the Copenhagen interpretation of quantum mechanics the state of a system is fully described by a wave function $\psi(t,\boldsymbol{x})$ which represents the probability of finding the system at a given point of the state space. Therefore, the square integral of the wave function $\psi(t,\boldsymbol{x})$ over all the space should result the probability equal to one:

$$\int_{\boldsymbol{x}\in\mathbb{R}^\delta} \left\|\psi(t,\boldsymbol{x})\right\|^2 \mathrm{d}\boldsymbol{x} = 1. \tag{5.1}$$

For that reason, it is desirable to have unitarity-preserving methods.

We are interested in quantum mechanical systems that have time-dependent potentials, that is, the Hamiltonian has the following two-part structure:

$$H(t,\boldsymbol{x}) = T(\boldsymbol{x}) + V(t,\boldsymbol{x}) \coloneqq -\frac{\nabla^2}{2\mu} + V(t,\boldsymbol{x}), \tag{5.2}$$

where μ is the reduced mass, ∇^2 is the Laplacian operator, and $V(t,\boldsymbol{x})$ is a potential with explicit time dependence. In particular, we will focus on the case when the potential takes a time–space separable form

Introduced on page 31

$$V(t,\boldsymbol{x}) = V_a(\boldsymbol{x}) + f(t)D(\boldsymbol{x}), \tag{5.3}$$

where $f(t)$ represents some external force, for example, a controlling laser. It will be shown later in this chapter how this particular structure allows for simpler calculation, although the methods are valid for the general time-dependent potentials.

THE STRATEGY in this chapter is as follows.

1. To simplify the exposition, we consider the SE in one spatial dimension.

2. As described in Chapter 2, we discretize the SE (5.2) in space, using a mesh of d points $x_0,\dots,x_{d-1}$ (the endpoint x_f is not included to the periodicity assumption), and get a 1st-order system of ODEs:

$$\begin{aligned} &\dot{\boldsymbol{u}} = -\mathrm{i}\boldsymbol{Z}(t)\boldsymbol{u},\ \boldsymbol{u}(t_0) = \boldsymbol{u}_0, \\ &\text{where } \boldsymbol{Z}(t) \coloneqq \boldsymbol{T} + \boldsymbol{V}(t) \\ &\text{and } \boldsymbol{u}(t) \coloneqq \Big[\psi(t,x_0),\cdots,\psi(t,x_{d-1})\Big]^T \in \mathbb{C}^d \end{aligned} \tag{5.4}$$

   We assume a potential to have the form (5.3) with V_a and D such that $\boldsymbol{V}$ is a diagonal matrix. Due to (5.1), $\boldsymbol{u}$ should be a unit vector: $\|\boldsymbol{u}\| = 1$. Analogously, H is represented by a Hermitian matrix, so the fundamental matrix solution is a unitary operator.

3. For the discretized equation, our design will be based on the ME (4.28), expressed through the generators of a Lie algebra. They, in turn, are expressed by means of linear combinations of the Hamiltonian values $\boldsymbol{Z}_j$, computed at the nodes c_j of a quadrature rule.

4. To circumvent the overheads related to the commutators in the ME, we will search for methods that approximate $\exp\Omega^{[p]}$ by a composition (i. e. product) of exponentials that contain $\boldsymbol{Z}_j$ and cheap commutators. Unitary transformations are reversible, so we build symmetric schemes to lower the number of order conditions to solve.

5. Despite aiming for CF methods, the cost of the exponentiation still can be high, and we will diminish it by using an approximation in an m-dimensional ($m \ll d$) Krylov subspace, described in Section 2.4. To calculate the action of the discrete Laplacian $\boldsymbol{T}$, we will use the FFT-based differentiation from Section 2.3. Consequently, the FFT–IFFT pair is the most expensive operation in this context.

5.1 LIE ALGEBRA

In order to express the ME, we associate a Lie algebra with the problem, following Section 4.2. For methods up to order 8, we will need no more than four generators ζ_i, $i = 1, \dots, 4$.

Let us examine their commutators. Firstly, as it has been mentioned on page 31, for any symmetric quadrature rule only ζ_1 contains the kinetic part $\boldsymbol{T}$. Accordingly, under the assumption that $\boldsymbol{V}$ is diagonal, all other ζ_i commute with each other: $[\zeta_i, \zeta_j] = 0$, $\forall i, j \geq 2$. Also, note that $\boldsymbol{T}$ is the most computationally expensive part.

Secondly, we look how the commutator $[\zeta_1, \zeta_j] \coloneqq [1j]$, $j \geq 2$, acts on the wave function $\psi(t, x)$. Leaving out scalar factors, its core is

$$\left[-\frac{\partial^2}{\partial x^2}, V_j\right]\psi = -\frac{\partial^2 V_j}{\partial x^2}\psi + V_j\frac{\partial^2\psi}{\partial x^2} = -\frac{\partial^2 V_j}{\partial x^2} - 2\frac{\partial V_j}{\partial x}\frac{\partial\psi}{\partial x}.$$

Consequently, the further nested commutators $[i1j]$ yield

$$\left[V_i, -\frac{\partial^2}{\partial x^2}, V_j\right]\psi = \left(\frac{\partial V_i}{\partial x}\right)\left(\frac{\partial V_j}{\partial x}\right)\psi,$$

which means they are local (depend only on coordinates) and commute with any other local operator.

In sum, commutators [23], [34], [124], [223], [1123], [2213] disappear, and the 8^{th}-order truncation (4.28) of the ME for this problem becomes

$$\begin{aligned}\Omega^{[8]} = \overbrace{\underbrace{\zeta_1 - \frac{1}{12}[12] + \frac{1}{12}\zeta_3 + \frac{1}{360}[113] - \frac{1}{240}[212] + \frac{1}{720}[1112]}_{\text{order 6 with 3 gen.}}}^{\text{order 4 with 2 gen.}} \\ - \frac{1}{80}[14] + \frac{1}{6048}[313] - \frac{1}{840}[412] + \frac{1}{6720}[1114] \\ + \frac{1}{4032}[1312] + \frac{11}{60480}[2113] - \frac{1}{15120}[11113] \\ - \frac{1}{30240}[11212] + \frac{1}{7560}[21112] - \frac{1}{30240}[111112].\end{aligned} \tag{5.5}$$

Consider now the discretized form of the commutators $[i1j]$. Given a q-point GL quadrature rule and the transition matrix $\boldsymbol{S}$ from (4.21), which expresses ζ_i in terms of $\boldsymbol{Z}_i$, we can compute the $[i1j]$ matrix:

$$[i1j] = \tau^3 \sum_{k=1}^{q} s_{i,k} \boldsymbol{V}_k \sum_{k=1}^{q} s_{j,k} \boldsymbol{V}_k, \tag{5.6}$$

Recall (4.20): $\boldsymbol{V}_j := \boldsymbol{V}(t + c_j\tau)$

which is an $(i+1+j)$-grade element.

Furthermore, when the potential has the form (5.3), the commutators become

$$[i1j] = \tau^3 \left(\sum_{k=1}^{q} s_{i,k} f_k \sum_{k=1}^{q} s_{j,k} f_k \right) \left(\frac{\mathrm{d}\boldsymbol{D}(x)}{\mathrm{d}x} \right)^2 \tag{5.7}$$

thanks to the quadrature rule symmetry around the origin.

The above analysis leads us to the following design of integrators: compose exponentials in such a way, that a) the stages contains as few as possible ζ_1 to lower computational cost; b) diagonal elements ζ_i and commutators $[i1j]$ with $i, j \geq 2$ are included as 'cheap' linear combinations if $\partial V/\partial x$ is available. The general form is

$$\Upsilon_c^{[p]} = \prod_{s=1}^{m} \exp\left(\sum_{i=1}^{q} x_{s,i} \zeta_i + \sum_{i,j=1}^{q} v_{s,i,j} [i1j] \right),$$

where the coefficients obey time symmetry: $x_{m-k+1,i} = (-1)^{i+1} x_{k,i}$ and $v_{m-k+1,i,j} = (-1)^{i+j} v_{k,i,j}$ for $k = 1, \ldots, s$.

Although we would like to obtain purely CF methods, whose stages do not perform backwards fractional step integration ($\sum_{i=1}^{q} x_{m,i} > 0 \ \forall m$), it has been shown [57] that there exist no such methods with order higher than four.

Exponentials with ζ_1 are approximated by the Lanczos algorithm, described in Section 2.4, while the rest are diagonal matrices. Therefore, we count the cost (denoted by the subscript c) of each method in the units of FFT–IFFT pairs, which is the most computationally expensive operation in this case. It other words, it is the number of stages that contain ζ_1 with the discrete Laplacian $\boldsymbol{T}$.

5.2 FOURTH-ORDER METHODS

The minimal requirement for a 4^{th}-order method is the 2-point GL quadrature with the nodes:

The interval is $[0; 1]$

$$c_1 = \frac{1}{2} - \frac{\sqrt{3}}{6}, \ c_2 = \frac{1}{2} + \frac{\sqrt{3}}{6}, \tag{5.8}$$

thus

$$\zeta_1 = \boldsymbol{T} + \frac{1}{2}(\boldsymbol{V}_1 + \boldsymbol{V}_2), \ \zeta_2 = \sqrt{3}(\boldsymbol{V}_2 - \boldsymbol{V}_1). \tag{5.9}$$

To illustrate how the tools from previous chapters are used, we take the following straightforward symmetric scheme:

$$\Upsilon_2^{[4]} \equiv \mathrm{CF}_2^{[4]} = \exp(x_{2,1}\zeta_1 + x_{2,2}\zeta_2)\exp(x_{1,1}\zeta_1 + x_{1,2}\zeta_2). \tag{5.10}$$

Immediately, $x_{2,1} = x_{1,1}$ and $x_{2,2} = -x_{1,2}$ due to symmetry, thus

$$\Upsilon_2^{[4]} = \exp(x_{1,1}\zeta_1 - x_{1,2}\zeta_2)\exp(x_{1,1}\zeta_1 + x_{1,2}\zeta_2).$$

We apply the BCH formula (3.18) to obtain:

$$\Upsilon_2^{[4]} = \exp\Big(2x_{1,1}\zeta_1 + x_{1,1}x_{1,2}[12] + \mathcal{O}\big(\tau^5\big)\Big).$$

Notice that even elements are zeroed out. In the ME (5.5), we have only two elements that have grade lower than 4 and depend only on ζ_1 and ζ_2, therefore, we have two order conditions:

$$\begin{cases} 2x_{1,1} = 1, & \text{(consistency)} \\ x_{1,1}x_{1,2} = -\frac{1}{12}, \end{cases}$$

and their solution is

$$x_{1,1} = \frac{1}{2},\ x_{1,2} = -\frac{1}{6}. \tag{5.11}$$

The same procedure is carried out for all the rest methods in this thesis, but, obviously, it becomes more involved as the number of stages grows.

It is possible to optimise methods by solving additional order conditions if we use ζ_i of higher grades:

$$\begin{aligned} \zeta_1 &= -\mathrm{i}\tau(\boldsymbol{T} + \boldsymbol{V}_2), \\ \zeta_2 &= -\mathrm{i}\tau\frac{\sqrt{15}}{3}(\boldsymbol{V}_3 - \boldsymbol{V}_1), \\ \zeta_3 &= -\mathrm{i}\tau\frac{10}{3}(\boldsymbol{V}_3 - 2\boldsymbol{V}_2 + \boldsymbol{V}_1). \end{aligned} \tag{5.12}$$

From (5.5) we see that an optimised m-stage 4^{th}-order CF method should satisfy only 3 order conditions:

$$\Upsilon^{[4opt]} = \prod_{s=1}^{m} \mathrm{e}^{x_{i,1}\zeta_1 + x_{i,2}\zeta_2 + x_{i,3}\zeta_3} = \exp\Big(\zeta_1 + \frac{1}{12}\zeta_3 - \frac{1}{12}[12] + \mathcal{O}\big(\tau^5\big)\Big). \tag{5.13}$$

When we take the minimal $m = 2$, we have exactly 3 variables in the order conditions:

$$\begin{aligned} \Upsilon_2^{[4opt]} = &\exp(x_{1,1}\zeta_1 - x_{1,2}\zeta_2 + x_{1,3}\zeta_3) \\ &\times\exp(x_{1,1}\zeta_1 + x_{1,2}\zeta_2 + x_{1,3}\zeta_3). \end{aligned} \tag{5.14}$$

As a result, $x_{1,1}$ cannot be zero, hence the cost of 2 FFT–IFFT units. Although viable, we discard this method because cheaper optimised compositions are possible.

Now, if we consider a symmetric composition with $m = 3$, then the even element in the centre should disappear due to its skew-symmetry: $x_{2,2} = 0$. A simple method with one ζ_1 and three variables to solve the order conditions is either

$$\begin{aligned}\Upsilon_{1a}^{[4opt]} = &\exp(-x_{1,2}\zeta_2 + x_{1,3}\zeta_3)\\ &\times\exp(x_{2,1}\zeta_1)\\ &\times\exp(x_{1,2}\zeta_2 + x_{1,3}\zeta_3),\end{aligned} \tag{5.15}$$

where $x_{1,2} = -1/12$, $x_{1,3} = 1/24$, $x_{2,1} = 1$, or

$$\begin{aligned}\Upsilon_{1b}^{[4opt]} = &\exp(-x_{1,2}\zeta_2)\\ &\times\exp(x_{2,1}\zeta_1 + x_{2,3}\zeta_3)\\ &\times\exp(x_{1,2}\zeta_2),\end{aligned} \tag{5.16}$$

with coefficients $x_{1,2} = -1/12$, $x_{2,1} = 1$, $x_{2,3} = 1/12$. Without commutator injection, a 3-stage method can have 5 parameters at most. To lower the computational cost we can exclude $x_{1,1}$, which leaves only one expensive exponential with the Laplacian:

$$\begin{aligned}\Upsilon_{1c}^{[4opt]} = &\exp(-x_{1,2}\zeta_2 + x_{1,3}\zeta_3)\\ &\times\exp(x_{2,1}\zeta_1 + x_{2,3}\zeta_3)\\ &\times\exp(x_{1,2}\zeta_2 + x_{1,3}\zeta_3),\end{aligned} \tag{5.17}$$

The resulting composition has 4 parameters to solve 3 order equations, leaving one parameter for further optimization. The scheme that satisfies an additional condition (for [113]) has the solution [17]

$$x_{1,2} = -\frac{1}{12},\ x_{1,3} = \frac{1}{60},\ x_{2,1} = 1,\ x_{2,3} = \frac{1}{20}. \tag{5.18}$$

In these methods the first and the last exponentials are diagonal, so their computational cost in units of FFT–IFFT is the same as the 2^{nd}-order midpoint exponential method (4.29).

By adding one more stage with the Laplacian and raising the cost to 2FFT–IFFT units, we obtain the following 4^{th}-order scheme with 2 free parameters:

$$\begin{aligned}\Upsilon_{2}^{[4opt]} = &\exp(-x_{1,2}\zeta_2 + x_{1,3}\zeta_3)\\ &\times\exp(x_{2,1}\zeta_1 - x_{2,2}\zeta_2 + x_{2,3}\zeta_3)\\ &\times\exp(x_{2,1}\zeta_1 + x_{2,2}\zeta_2 + x_{2,3}\zeta_3)\\ &\times\exp(x_{1,2}\zeta_2 + x_{1,3}\zeta_3).\end{aligned} \tag{5.19}$$

The solution of its order equations is

$$x_{1,2} = -x_{1,3} = -\frac{1}{60},\ x_{2,1} = \frac{1}{2},\ x_{2,2} = -\frac{2}{15},\ x_{2,3} = \frac{1}{40},$$

which satisfies two of three 6^{th}-order conditions: $[113]$ and $[1112]$. The condition for $[212]$ is the only one left, which means that

$$\Upsilon_2^{[4opt]} = \exp\Big(\Omega^{[6]} - 2v_{1,212}[212] + \mathcal{O}\big(\tau^7\big)\Big).$$

5.3 SIXTH-ORDER METHODS

For 6^{th}-order methods we mostly focus on QCF methods when the derivatives $\partial V_j \big/ \partial x$ are easily computed, so we can introduce cheap diagonal commutators to the scheme. Otherwise, the absence of $[i1j]$ requires additional stages to satisfy all the conditions of order 6.

Continuing from the last 4^{th}-order method (5.19), it is immediate to see that the following scheme can fulfil all the order 6 conditions:

$$\begin{aligned}\Upsilon_2^{[6]} = &\exp(-x_{1,2}\zeta_2 + x_{1,3}\zeta_3 + v_{1,212}[212])\\ &\times\exp(x_{2,1}\zeta_1 - x_{2,2}\zeta_2 + x_{2,3}\zeta_3)\\ &\times\exp(x_{2,1}\zeta_1 + x_{2,2}\zeta_2 + x_{2,3}\zeta_3)\\ &\times\exp(x_{1,2}\zeta_2 + x_{1,3}\zeta_3 + v_{1,212}[212]).\end{aligned} \tag{5.20}$$

The solution $x_{i,j}$ is the same as in (5.19) with additional $v_{1,212} = 1/43200$. Its cost is 2 FFT–IFFT units.

To obtain a purely CF method, we start with the same structure that (5.20) has: the expensive part ζ_1 in the outermost exponentials is zeroed out by $x_{1,1} = 0$. Then, we add an inner stage, which automatically leads to $x_{3,2} = 0$ and an increase in the cost up to 3 FFT–IFFT units. There is one free parameter among the coefficients of ζ_3: $x_{1,3}$, $x_{2,3}$ and $x_{3,3}$. These variables only appear linearly in two of the order conditions. We put $x_{1,3} = 0$, and the scheme becomes:

$$\begin{aligned}\Upsilon_3^{[6]} = &\exp(-x_{1,2}\zeta_2)\\ &\times\exp(x_{2,1}\zeta_1 - x_{2,2}\zeta_2 + x_{2,3}\zeta_3)\\ &\times\exp(x_{3,1}\zeta_1 + x_{3,3}\zeta_3)\\ &\times\exp(x_{2,1}\zeta_1 + x_{2,2}\zeta_2 + x_{2,3}\zeta_3)\\ &\times\exp(x_{1,2}\zeta_2)\end{aligned} \tag{5.21}$$

Using the free parameter, we minimize the quality functional $\sum|x_{i,j}|$ to get short fractional time steps during the integration (which usually leads to smaller errors):

$$\begin{aligned} x_{1,2} &= -0.015446203250884 & x_{2,3} &= \ 0.085748160282456\\ x_{2,1} &= \ 0.567040718865477 & x_{3,1} &= -0.134081437730955\\ x_{2,2} &= -0.156797955467218 & x_{3,3} &= -0.088162987231579. \end{aligned} \tag{5.22}$$

Optimised sixth-order methods

Similarly to the 4^{th}-order methods, we can take a higher-order quadrature rule to satisfy additional conditions for 6^{th}-order methods in (5.5). To this end we make compositions, similar to (5.20), with diagonal commutators in the outermost stages:

$$\begin{aligned}\Upsilon_3^{[6opt]} = &\exp(-x_{1,2}\zeta_2 + x_{1,3}\zeta_3 - x_{1,4}\zeta_4 + v_{1,212}[212] + v_{1,313}[313] - v_{1,213}[213])\\ &\times\exp(x_{2,1}\zeta_1 - x_{2,2}\zeta_2 + x_{2,3}\zeta_3 - x_{2,4}\zeta_4)\\ &\times\exp(-x_{3,2}\zeta_2 + x_{3,3}\zeta_3 - x_{3,4}\zeta_4)\\ &\times\exp(x_{4,1}\zeta_1 + x_{4,3}\zeta_3)\\ &\times\exp(x_{3,2}\zeta_2 + x_{3,3}\zeta_3 + x_{3,4}\zeta_4)\\ &\times\exp(x_{2,1}\zeta_1 + x_{2,2}\zeta_2 + x_{2,3}\zeta_3 + x_{2,4}\zeta_4)\\ &\times\exp(x_{1,2}\zeta_2 + x_{1,3}\zeta_3 + x_{1,4}\zeta_4 + v_{1,212}[212] + v_{1,313}[313] + v_{1,213}[213])\end{aligned} \tag{5.23}$$

This scheme provides two alternatives: either discard the equation corresponding to [111112] or [21112]. The solutions, respectively, are

$$\begin{aligned}\text{for } \Upsilon_{3a}^{[6opt]}\text{:}\qquad & \qquad \text{for } \Upsilon_{3b}^{[6opt]}\text{:}\\ x_{1,2} &= -0.013381037301689 & x_{1,2} &= -0.008752729116750\\ x_{1,3} &= 0.006456698486049 & x_{1,3} &= 0.005323928662358\\ x_{1,4} &= -0.005062980277979 & x_{1,4} &= -0.004450414289558\\ x_{2,1} &= 0.692033178955027 & x_{2,1} &= 0.768023282768151\\ x_{2,2} &= -0.200354649256364 & x_{2,2} &= -0.239740381573067\\ x_{2,3} &= 0.078762858864116 & x_{2,3} &= 0.096007548854092\\ x_{2,4} &= -0.021090755452322 & x_{2,4} &= -0.026193474535960\\ x_{3,2} &= 0.021479912126313 & x_{3,2} &= 0.035382033447741\\ x_{3,3} &= 0.006520012008462 & x_{3,3} &= 0.007033760004535\\ x_{3,4} &= 0.002452094000258 & x_{3,4} &= 0.003681227717073\\ x_{4,1} &= -0.384066357910053 & x_{4,1} &= -0.536046565536302\\ x_{4,3} &= -0.100145805383921 & x_{4,3} &= -0.133397141708636\\ v_{1,212} &= -0.000012785453959 & v_{1,212} &= 0.000022652861510\\ v_{1,213} &= 0.000013916301486 & v_{1,213} &= 0.000050348766403\\ v_{1,313} &= 0.000042648468634, & v_{1,313} &= 0.000086455336413.\end{aligned} \tag{5.24}$$

5.4 EIGHTH-ORDER METHODS

In this case possible scheme structures branch off faster. From other options, we consider methods with outermost commutators once again. The first one has 7 stages with the cost of 5 FFT–IFFT units:

$$\begin{aligned}\Upsilon^{[8]}_{5,1} =& \exp(-x_{1,2}\zeta_2 + x_{1,3}\zeta_3 - x_{1,4}\zeta_4 + v_{1,212}[212] + v_{1,313}[313] - v_{1,213}[213])\\ &\times\exp(x_{2,1}\zeta_1 - x_{2,2}\zeta_2 + x_{2,3}\zeta_3 - x_{2,4}\zeta_4)\\ &\times\exp(x_{3,1}\zeta_1 - x_{3,2}\zeta_2 + x_{3,3}\zeta_3 - x_{3,4}\zeta_4)\\ &\times\exp(x_{4,1}\zeta_1 + x_{4,3}\zeta_3)\\ &\times\exp(x_{3,1}\zeta_1 + x_{3,2}\zeta_2 + x_{3,3}\zeta_3 + x_{3,4}\zeta_4)\\ &\times\exp(x_{2,1}\zeta_1 + x_{2,2}\zeta_2 + x_{2,3}\zeta_3 + x_{2,4}\zeta_4)\\ &\times\exp(x_{1,2}\zeta_2 + x_{1,3}\zeta_3 + x_{1,4}\zeta_4 + v_{1,212}[212] + v_{1,313}[313] + v_{1,213}[213]).\end{aligned} \tag{5.25}$$

The order conditions consist of 16 non-linear equations with 16 variables. Among four real-valued solutions the most promising in terms of minimising the sum of coefficients are

for $\Upsilon^{[8]}_{5,1a}$:	for $\Upsilon^{[8]}_{5,1b}$:
$x_{1,2} = 0.008696241374160$	$x_{1,2} = -0.005555689802628$
$x_{1,3} = -0.008696241374160$	$x_{1,3} = 0.005555689802628$
$x_{1,4} = -1/240$	$x_{1,4} = -1/240$
$x_{2,1} = 0.258478395344154$	$x_{2,1} = 0.689505417442239$
$x_{2,2} = -0.120726493065581$	$x_{2,2} = -0.250263632191044$
$x_{2,3} = 0.051611313197706$	$x_{2,3} = 0.085548634263565$
$x_{2,4} = -0.011893828397299$	$x_{2,4} = -0.026814053285156$
$x_{3,1} = 1.040067278127429$	$x_{3,1} = -0.379540734471510$
$x_{3,2} = 0.004502986469442$	$x_{3,2} = -0.136141876544218$
$x_{3,3} = -0.029153958678289$	$x_{3,3} = -0.150901769396850$
$x_{3,4} = -0.000872848073453$	$x_{3,4} = -0.014559460067438$
$x_{4,1} = -1.597091346943166$	$x_{4,1} = 0.380070634058541$
$x_{4,3} = 0.055811107042819$	$x_{4,3} = 0.202928223994648$
$v_{1,212} = 0.000070200570963$	$v_{1,212} = 0.000001103123116$
$v_{1,213} = -0.000140401141926$	$v_{1,213} = -0.000002206246233$
$v_{1,313} = 0.000070200570963,$	$v_{1,313} = 0.000001103123116.$

(5.26)

Note the large absolute values of $x_{3,1}$ and $x_{4,1}$ in $\Upsilon^{[8]}_{5,1a}$; we expect this method to be less accurate.

Alternatively, we can move the commutators into the inner stages, obtaining schemes similar to the one proposed in [17]:

$$\begin{aligned}\Upsilon_5^{[8]} =& \exp(x_{1,2}\zeta_1 - x_{1,2}\zeta_2 + x_{3,3}\zeta_3 - x_{1,4}\zeta_4 + v_{1,212}[212])\\ &\times\exp(-x_{2,2}\zeta_2 + x_{2,3}\zeta_3 - x_{2,4}\zeta_4 + v_{2,212}[212] + v_{2,313}[313])\\ &\times\exp(x_{3,1}\zeta_1 - x_{3,2}\zeta_2 + x_{3,3}\zeta_3 - x_{3,4}\zeta_4)\\ &\times\exp(x_{4,1}\zeta_1 + x_{4,3}\zeta_3)\\ &\times\exp(x_{3,1}\zeta_1 + x_{3,2}\zeta_2 + x_{3,3}\zeta_3 + x_{3,4}\zeta_4)\\ &\times\exp(x_{2,2}\zeta_2 + x_{2,3}\zeta_3 + x_{2,4}\zeta_4 + v_{2,212}[212] + v_{2,313}[313])\\ &\times\exp(x_{1,2}\zeta_1 + x_{1,2}\zeta_2 + x_{3,3}\zeta_3 + x_{1,4}\zeta_4 + v_{1,212}[212]).\end{aligned}$$

We make a composition of 9 exponentials with the commutators in the second and the penultimate stages:

$$\begin{aligned}\Upsilon_{5,2}^{[8]} =& \exp(x_{1,1}\zeta_1 - x_{1,2}\zeta_2 + x_{1,3}\zeta_3 - x_{1,4}\zeta_4)\\ &\times\exp(-x_{2,2}\zeta_2 + x_{2,3}\zeta_3 - x_{2,4}\zeta_4 + v_{2,212}[212] - v_{2,313}[213] + v_{2,313}[313])\\ &\times\exp(x_{3,1}\zeta_1 - x_{3,2}\zeta_2 + x_{3,3}\zeta_3 - v_{3,4}\zeta_4)\\ &\times\exp(-x_{4,2}\zeta_2 + x_{4,3}\zeta_3 - x_{4,4}\zeta_4)\\ &\times\exp(x_{5,1}\zeta_1 + x_{5,3}\zeta_3)\\ &\times\exp(x_{4,2}\zeta_2 + x_{4,3}\zeta_3 + x_{4,4}\zeta_4)\\ &\times\exp(x_{3,1}\zeta_1 + x_{3,2}\zeta_2 + x_{3,3}\zeta_3 + v_{3,4}\zeta_4)\\ &\times\exp(x_{2,2}\zeta_2 + x_{2,3}\zeta_3 + x_{2,4}\zeta_4 + v_{2,212}[212] + v_{2,213}[213] + v_{2,313}[313])\\ &\times\exp(x_{1,1}\zeta_1 + x_{1,2}\zeta_2 + x_{1,3}\zeta_3 + x_{1,4}\zeta_4).\end{aligned} \tag{5.27}$$

The advantage of this design is that we have 3 additional variables that can be used as free parameters for tuning the scheme, maintaining the cost at only 5 FFT units.

First, we explore the case when two inner diagonal exponential are excluded by $x_{4,2} = x_{4,3} = x_{4,4} = 0$. There are 4 real solutions, the best with respect to quality functional gives the method $\Upsilon_{5,2a}^{[8]}$ that has the coefficients

$$\begin{aligned}x_{1,1} &= \ \ 0.670219114423756 & x_{3,2} &= \ \ 0.138260115373570\\ x_{1,2} &= -0.304894500128406 & x_{3,3} &= -0.117677987842383\\ x_{1,3} &= \ \ 0.137339721522467 & x_{3,4} &= \ \ 0.051942668557384\\ x_{1,4} &= -0.061889862325139 & x_{5,1} &= \ \ 0.681384887167753\\ x_{2,2} &= \ \ 0.018665991927430 & x_{5,3} &= \ \ 0.031300631510253\\ x_{2,3} &= \ \ 0.006354617231456 & v_{2,212} &= -0.000416674497669\\ x_{2,4} &= \ \ 0.002775087956074 & v_{2,213} &= -0.000283703855984\\ x_{3,1} &= -0.510911558007632 & v_{2,313} &= -0.000048291819124.\end{aligned} \tag{5.28}$$

To get another method, we fix $x_{4,3} = x_{4,4} = 0$ and explore the set of solutions when $x_{4,2}$ has a small value in the interval $[-0.1;\ 0.1]$. For two

distinct values of $x_{4,2}$, we get relatively small coefficients for ζ_1, while the remaining coefficients remain not very large. We fix these $x_{4,2}$ and vary both $x_{4,3}$ and $x_{4,4}$, which results in slightly different schemes with rather close coefficients for ζ_1:

for $\Upsilon^{[8]}_{5,2b}$:	for $\Upsilon^{[8]}_{5,2c}$:
$x_{1,1} = 0.312720638582279$	$x_{1,1} = 0.322618899122869$
$x_{1,2} = -0.136763329733817$	$x_{1,2} = -0.140802452623684$
$x_{1,3} = 0.058158591955603$	$x_{1,3} = 0.059750258819098$
$x_{1,4} = -0.024519487903109$	$x_{1,4} = -0.025031572376557$
$x_{2,2} = 0.118007839703159$	$x_{2,2} = 0.144003840876807$
$x_{2,3} = -0.026642462443984$	$x_{2,3} = -0.030812811435236$
$x_{2,4} = 0.015257525315299$	$x_{2,4} = 0.016059935948481$
$x_{3,1} = 0.286322787565297$	$x_{3,1} = 0.272663507732993$
$x_{3,2} = -0.290313815590736$	$x_{3,2} = -0.359509883630990$
$x_{3,3} = -0.034594343029162$	$x_{3,3} = -0.039409484358590$
$x_{3,4} = -0.015448368096472$	$x_{3,4} = -0.015122581410081$
$x_{4,2} = 1/25$	$x_{4,2} = 1/20$
$x_{5,1} = -0.198086852295151$	$x_{5,1} = -0.190564813711725$
$x_{5,3} = 0.089489760368421$	$x_{5,3} = 0.104277407282790$
$v_{2,212} = 0.000158096387628$	$v_{2,212} = 0.000230735287877$
$v_{2,213} = -0.000112991943243$	$v_{2,213} = -0.000157445707004$
$v_{2,313} = 0.000095925940695,$	$v_{2,313} = 0.000126306010355.$

(5.29)

5.5 NUMERICAL EXAMPLE

In this section we examine the performance of the new methods. As it has been mentioned in the introduction to this chapter, we consider one-dimensional problems with the wave function $\psi(t,x)$, $x \in \mathbb{R}$ on a sufficiently large spatial domain $[x_0;\, x_f)$ to ensure that its value and its derivatives vanish. Therefore, we can impose periodic boundary conditions $\psi(t,x_0) \equiv \psi(t,x_f)$ and use the FFT to approximate the Laplacian operator.

We divide the space interval into d bins of length $\Delta x = (x_f - x_0)/d$, hence $x_k = x_0 + k\Delta x$, $k = 0,\dots,d-1$. The discrete representation of $\psi(t,x)$ is a d-dimensional vector $\boldsymbol{u}$ with components $u_k = \sqrt{\Delta x}\,\psi(t,x_k)$, and its norm $\|\boldsymbol{u}\|_2$ does not depend on t.

To check the accuracy of a method, we calculate a reference solution $\boldsymbol{u}_{ref}$ at the final time t_f with sufficiently low tolerance. Then, we take a set of distinct (but constant) time steps $\tau_1,\dots,\tau_r$, which is equivalent

to dividing the time interval $[t_0;\ t_f]$ into $n_1,\dots,n_r$ subintervals, respectively, and simultaneous increase in final integration cost. With each of these time steps, we obtain r solutions $\tilde{\boldsymbol{u}}_i$, $i = 1,\dots,r$ using a method of interest and compute the norm of the error as $\|err\|_2 := \left\|\boldsymbol{u}_{ref} - \tilde{\boldsymbol{u}}_i\right\|_2$. Finally, we plot error norm versus the corresponding final cost (i. e., number FFT–IFFT pair calls, which is equal to the number of steps per the number of stages with $\boldsymbol{T}$) in double logarithmic scale.

We compare the following families of integrators:

- quasi-commutator-free (QCF) methods $\Upsilon_c^{[p]}$ presented in this chapter;
- 2^{nd}-order exponential midpoint (4.29) is given as a reference;
- general-purpose Magnus-based commutator-free (CF) methods from [5];
- classical explicit Runge–Kutta (RK) methods;
- implicit symplectic RK Gauss–Legendre (GL) methods.

In CF and QCF methods we approximate the exponentials by Lanczos algorithm from Section 2.4 with the tolerance set to the machine $\varepsilon_M \approx 2.2\times10^{-16}$ and the subspace dimension limited to $m_{max} := 15$.

The implicit RK method require solving a linear system of equations at each step to get the coefficients. It can be done by an iterative process, and to count the cost we assume that 4–6 iterations are sufficient, although we obtain the solution using solvers with low tolerance.

THE WALKER–PRESTON MODEL Our example is a simple model that represents adequately multiple typical applications and may serve as an indicative benchmark. It is a one-dimensional Schrödinger equation with time–space separable potential:

$$\mathrm{i}\frac{\partial\psi(t,x)}{\partial t} = \left(-\frac{1}{2\mu}\frac{\partial^2}{\partial x^2} + V_a(x) + f(t)x\right)\psi(t,x),\ \psi(t_0,x) = \psi_0(x), \tag{5.30}$$

where the time-dependent part $f(t) = A\cos(\omega t)$ corresponds to an external laser field. The autonomous part is the Morse potential $V_a(x) = D(1-\exp(-\alpha x))^2$. The initial condition is the ground state of the potential:

$$\psi_0 = \sigma\exp\left(-\left(\gamma - \frac{1}{2}\right)\alpha x\right)\exp(-\gamma\,\mathrm{e}^{-\alpha x}).$$

The parameters describe the standard example of diatomic HF molecule in a strong laser field [117]: reduced mass $\mu = 1745$ a. u.; $D = 0.2251$ a. u.; $\alpha = 1.1741$ a. u.; amplitude $A = A_0 := 0.011025$ a. u.; frequency $\omega = \omega_0 := 0.01787$ a. u.; $w_0 = \alpha\sqrt{2D/\mu}$; $\gamma = 2D/w_0$; σ is a normalizing constant. The intervals are chosen as follows: $[0;\ 10\pi/\omega]$ for time and $[-1.3;\ 3.2)$ for space. To illustrate the sensibility of the

methods to the parameters of the system, the experiments are repeated with two meshes ($d = 64$ and $d = 128$) and halved parameters $A = A_0/2$ and $\omega = \omega_0/2$, leaving time steps the same.

To facilitate the comparison between methods of different order we use the same axis limits in plots for each value d. Figures 5.1 to 5.3 show the results for $d = 64$, and Figures 5.4 to 5.6 for $d = 128$. It is evident that the classical RK methods have inferior performance, and the explicit RK requires small time steps for stability (for this reason the higher-order RK methods are removed from the graphs). The first thing we can highlight is the better performance of the new, tailored QCF integrators when compared to the more general CF methods. The second is that the relative ranking of the methods does not change with the parameters.

In the case of the 4^{th}-order methods, we see that optimisation with the 6^{th}-order quadrature rule brings significant improvement both for CF and QCF methods. Although in Figures 5.1a and 5.4a $\Upsilon_{1a}^{[4opt]}$ (whose cost is almost the same as the midpoint's) surpasses the highly-optimised $\Upsilon_{2}^{[4opt]}$, the latter is generally to be a generally better method.

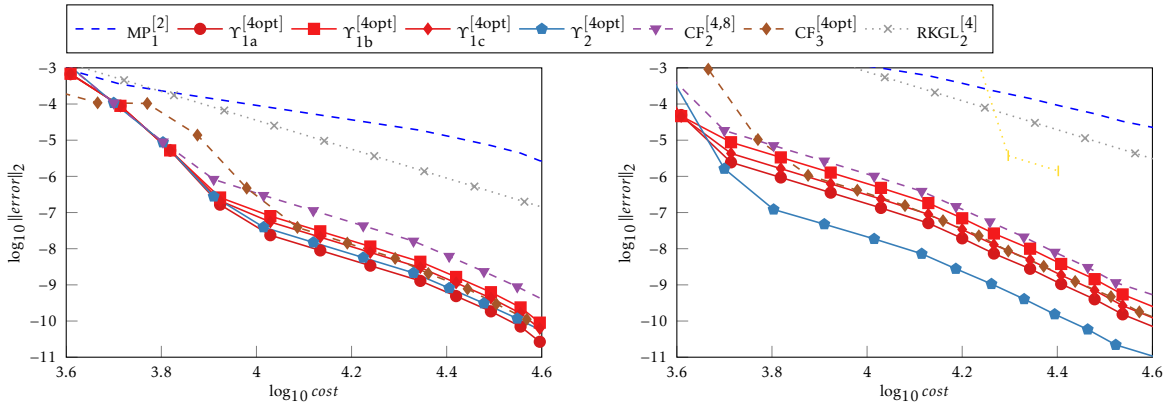
On the contrary, optimisation of 6^{th}-order methods does not seem to improve efficiency in this benchmark. Thanks to its cost of 2 FFT–IFFT units, $\Upsilon_{2}^{[6]}$ is the most computationally efficient method, when derivatives of the potential are available. Otherwise, $\Upsilon_{3}^{[6]}$ can be used instead of CF methods.

Among the 8^{th}-order methods, the best results are obtained with the $\Upsilon_{5,2}^{[8]}$ schemes. It is worth noting that the microoptimisation of coefficients within a family does not make a noticeable difference but $\Upsilon_{5,2}^{[8]}$ is, as we have expected, the worst method due to the large negative coefficient. In this example the 8^{th}-order methods do not outperform the 6^{th}-order methods due to a twice higher computational cost. Notwithstanding, the optimised 6^{th}-order and 8^{th}-order methods are expected to yield more accuracy at higher (say, quadruple) precision.

5.6 CONCLUSIONS

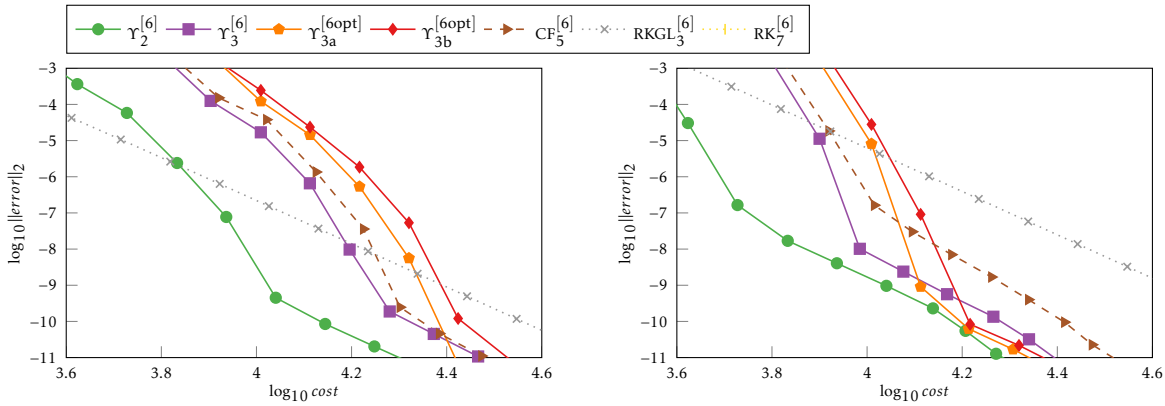
In this chapter, we have constructed quasi-commutator-free (QCF) Magnus expansion-based methods for the Schrödinger equation (SE) with time-dependent Hamiltonian $H(t,\boldsymbol{x})$ that can be presented as sum of the kinetic energy $T(\boldsymbol{x})$ and an explicitly time-dependent potential $V(t,\boldsymbol{x})$. The new methods gain their efficiency thanks to the adaptation to the specific properties of the problem, hence their reduced computational cost.

We have shown appropriateness of the methods on a widely used benchmark example. They turn out to be better than the stable mid-



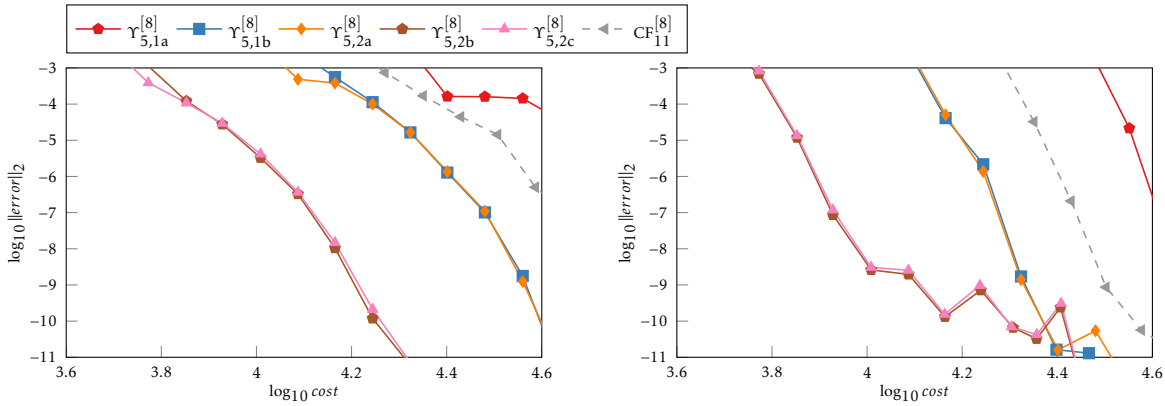
(a) $A = A_0/2$, $\omega = \omega_0/2$ (b) $A = A_0$, $\omega = \omega_0$

Figure 5.1: Efficiency graphs of the 4$^{\text{th}}$-order methods for the Walker–Preston model with $d = 64$ and distinct laser parameters.



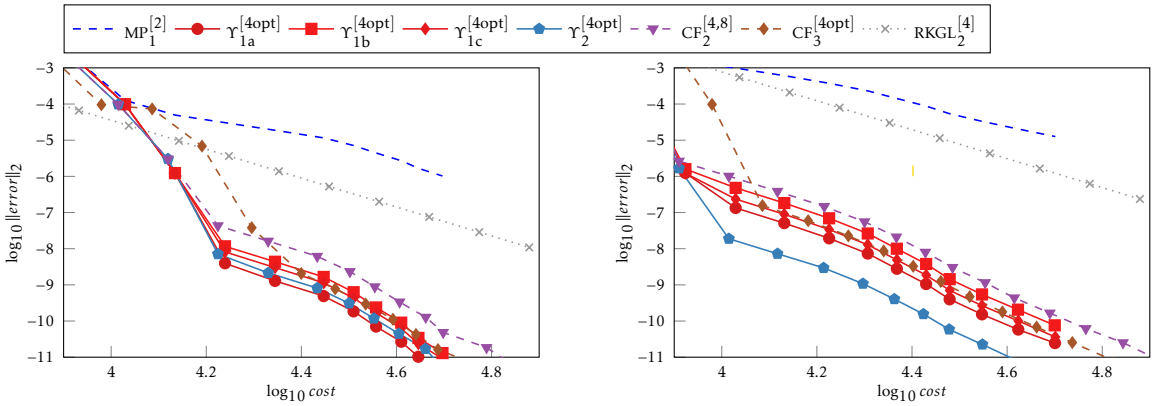
(a) $A = A_0/2$, $\omega = \omega_0/2$ (b) $A = A_0$, $\omega = \omega_0$

Figure 5.2: WPM: efficiency graphs of the 6$^{\text{th}}$-order methods, $d = 64$.



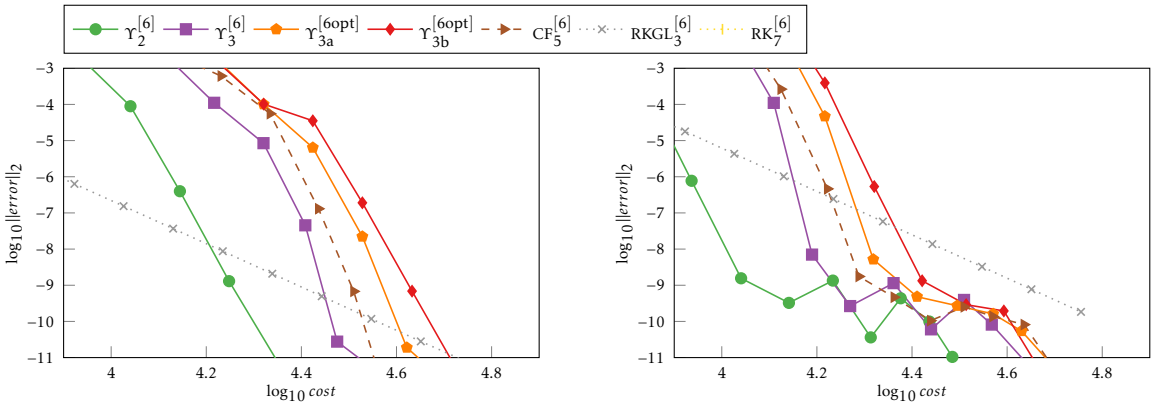
(a) $A = A_0/2$, $\omega = \omega_0/2$ (b) $A = A_0$, $\omega = \omega_0$

Figure 5.3: WPM: efficiency graphs of the 8$^{\text{th}}$-order methods, $d = 64$.



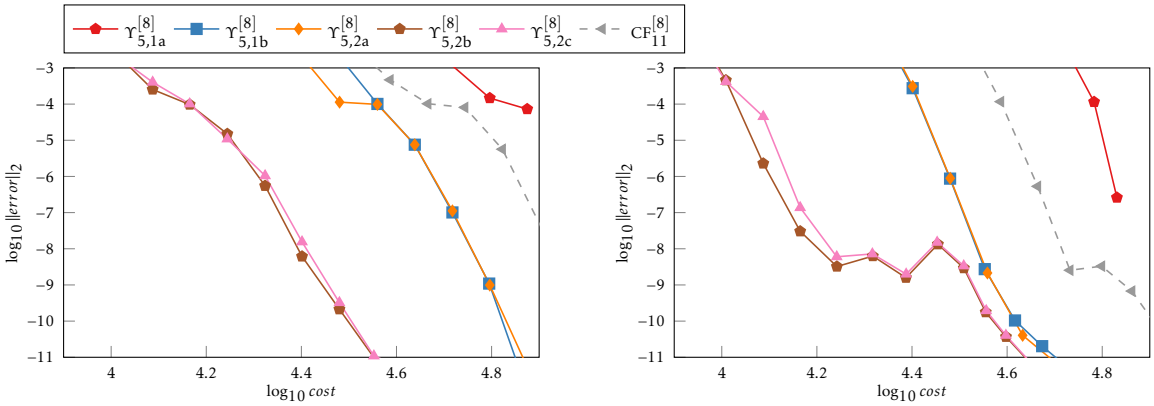
(a) $A = A_0/2$, $\omega = \omega_0/2$ (b) $A = A_0$, $\omega = \omega_0$

Figure 5.4: Efficiency graphs of the 4^{th}-order methods for the Walker–Preston model with $d = 128$ and distinct laser parameters.



(a) $A = A_0/2$, $\omega = \omega_0/2$ (b) $A = A_0$, $\omega = \omega_0$

Figure 5.5: WPM: efficiency graphs of the 6^{th}-order methods, $d = 128$.

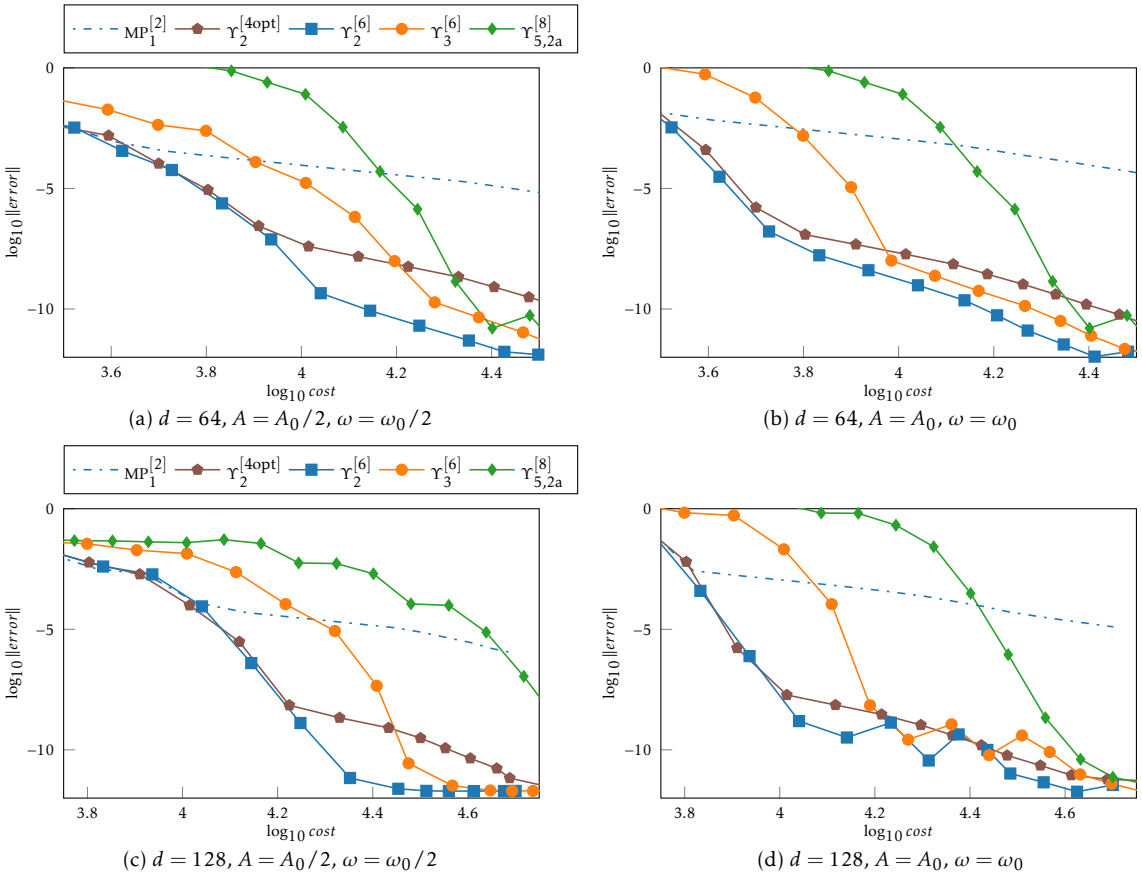


(a) $A = A_0/2$, $\omega = \omega_0/2$ (b) $A = A_0$, $\omega = \omega_0$

Figure 5.6: WPM: efficiency graphs of the 8^{th}-order methods, $d = 128$.

point scheme for all accuracies of practical interest and have similar implementation complexity, so any user of the exponential midpoint method might consider one of the new methods for solving this class of problems. The numerical experiments also have shown the superior performance of the new QCF schemes compared to the non-customised commutator-free (CF) methods.

To sum up, the optimized 4^{th}-order $\Upsilon_2^{[4opt]}$ is appropriate for use with larger time steps when high accuracy is not required. The 6^{th}-order $\Upsilon_2^{[6]}$ is the method of choice when spatial derivatives of the potential $V(t,\boldsymbol{x})$ are available. Otherwise, $\Upsilon_3^{[6]}$ should be preferred. Although in this example the 8^{th}-order methods have not surpassed the lower-order ones due to a notably higher computational cost, they may be more accurate in higher precision arithmetic.



(a) $d = 64$, $A = A_0/2$, $\omega = \omega_0/2$

(b) $d = 64$, $A = A_0$, $\omega = \omega_0$

(c) $d = 128$, $A = A_0/2$, $\omega = \omega_0/2$

(d) $d = 128$, $A = A_0$, $\omega = \omega_0$

Figure 5.7: WPM: efficiency graphs of the best methods.

6 HILL AND WAVE EQUATION

The exposition in this chapter is based on the articles [14, 15]

In this chapter, we address two types of problems that — after certain transformations — are represented by systems of ODEs with similar structure. Although the general approach is the same as for the Schrödinger equation (SE), numerical methods will differ according to the goal of integration and the properties of the initial problems.

HILL DIFFERENTIAL EQUATION is the first problem we consider:

$$\ddot{\boldsymbol{y}} = \boldsymbol{N}(t)\boldsymbol{y},\ \boldsymbol{y}(t_0) = \boldsymbol{y}_0,\ \boldsymbol{y} \in \mathbb{R}^\delta. \tag{6.1}$$

Here, $\boldsymbol{N}(t)$ is a periodic matrix-valued function. Two special scalar cases are: the Mathieu equation

$$\ddot{y} + (\omega^2 + \varepsilon\cos 2t)y = 0 \tag{6.2}$$

and the Meissner equation with a square wave

$$\ddot{y} + (\omega^2 + \varepsilon\,\mathrm{sign}(\cos 2t))y = 0. \tag{6.3}$$

These equations arise from different practical applications with periodically variable systems, for example, spatially linear electric fields, waves in periodic media, electrons in crystal lattices, in Bose–Einstein condensates [17, 73, 75, 78], quadrupole devices [49], microelectromechanical systems [113], etc. The importance of applications poses the question of the stability analysis because parametric resonances can occur. Typically, the analytical representation of stability regions in the parametric space is inaccessible, and to find them one should repeatedly calculate the solution for one period with various sets of parameters.

Recall from Section 2.2, that we can express Hill equation (HE) (6.1) as a 1st-order system:

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix}\boldsymbol{y}\\ \dot{\boldsymbol{y}}\end{bmatrix} = \begin{bmatrix}\boldsymbol{O} & \boldsymbol{I}\\ \boldsymbol{N}(t) & \boldsymbol{O}\end{bmatrix}\begin{bmatrix}\boldsymbol{y}\\ \dot{\boldsymbol{y}}\end{bmatrix}, \quad \begin{aligned}\boldsymbol{y}(t_0) &= \boldsymbol{y}_0,\\ \dot{\boldsymbol{y}}(t_0) &= \dot{\boldsymbol{y}}_0.\end{aligned} \tag{6.4}$$

When $\boldsymbol{N}(t)$ is relatively small (say, $d \propto 100$), finding the numerical solution of the system as the fundamental matrix $\boldsymbol{\Phi}(t)$ is computationally feasible. If $\boldsymbol{N}(t)$ is real and symmetric, then $\boldsymbol{\Phi}$ is symplectic, i. e. $\boldsymbol{\Phi}^T\boldsymbol{J}\boldsymbol{\Phi} \equiv \boldsymbol{J}$, where $\boldsymbol{J}$ is the canonical matrix (2.5). The eigenvalues of $\boldsymbol{\Phi}$ form reciprocal pairs: $\{\lambda, 1/\lambda; \lambda^*, 1/\lambda^*\}$. Consequently, the system is stable if and only if its eigenvalues lie on the unit circle [48]. In general, this salient property is not preserved by general-purpose integrators, so they require small time steps to avoid numerical instability.

λ^* is the complex conjugate of λ.

The aforementioned properties urge for the creation of geometric methods for the HE. Any symplecticity-preserving method will show a better behaviour on long-time simulations compared to classical methods [29, 52, 71, 99]. Fortunately, (6.4) has a favourable structure that allows for scheme tailoring.

THE MODIFIED INHOMOGENEOUS WAVE EQUATION WITH VARIABLE COEFFICIENTS is the second problem of interest in this chapter:

Also called non-homogeneous non-autonomous WE

$$\begin{aligned} \frac{\partial^2 \psi(t,\boldsymbol{x})}{\partial t^2} &= \nabla^2 \psi(t,\boldsymbol{x}) + V(t,\boldsymbol{x})\psi(t,\boldsymbol{x}), \\ \psi(t_0,\boldsymbol{x}) &= \psi_0(\boldsymbol{x}), \quad t \in [t_0;\, t_f],\, \boldsymbol{x} \in \mathbb{R}^\delta \end{aligned} \tag{6.5}$$

Similarly to the previous chapter, for the sake of simplicity we will use its one-dimensional variant and assume periodic boundary conditions:

$$\begin{aligned} \frac{\partial^2 \psi(t,x)}{\partial t^2} &= \frac{\partial^2 \psi(t,x)}{\partial x^2} + V(t,x)\psi(t,x), \\ \psi(t_0,x) = \psi_0(x),\ \psi(t,x_0) &= \psi(t,x_f), \\ t \in [t_0;\, t_f],\, x \in [x_0;\, x_f&). \end{aligned} \tag{6.6}$$

In this setup, schemes do not suffer order reductions, and accurate results can be obtained with high-order methods [4, 50, 58, 100].

To reduce this PDE to a more suitable and simpler form, we follow a procedure similar to the one for the SE. First, we discretize in space using a sufficiently dense grid, which results in a 2^{nd}-order DE

$$\ddot{\boldsymbol{y}} = \boldsymbol{T}\boldsymbol{y} + \boldsymbol{V}(t)\boldsymbol{y},\ \boldsymbol{y}(t_0) = \boldsymbol{y}_0, \tag{6.7}$$

where $\boldsymbol{T}$ is the discrete Laplacian, $\boldsymbol{V}$ is the potential, and $\dim \boldsymbol{T} = \dim \boldsymbol{V} = d \times d$. This equation can be expressed as a 1^{st}-order system:

$$\ddot{\boldsymbol{u}} = \begin{bmatrix} \boldsymbol{O} & \boldsymbol{I} \\ \boldsymbol{T} + \boldsymbol{V}(t) & \boldsymbol{O} \end{bmatrix} \boldsymbol{u},\ \boldsymbol{u}(t_0) = \boldsymbol{u}_0, \tag{6.8}$$

whose dimension is $2d \times 2d$.

It is easy to see that IVPs originating from the Hill and the wave equation share the same basic structure. Nevertheless, there are several differences that should be addressed.

First of all, systems (6.4) and (6.8) differ in size. The matrix $\boldsymbol{N}(t)$ in the HE has a moderate dimension ($d \propto 100$), so matrix operations are carried out in reasonable time and exponentiating can be performed by the scaling and squaring method described in Section 2.4. Using an explicit exponentiation algorithm will also provide the fundamental matrix solution, which is necessary for estimating the eigenvalues for stability tests. On the other hand, when solving PDEs, fidelity imposes a requirement to discretize with a sufficiently dense mesh, and the result

is a significantly larger ($d \propto 1000$) system matrix. In this case straightforward matrix exponentiation may become unreasonable. Thus, an iterative procedure, like approximation in the Krylov subspace used for the SE, can be employed instead.

The second point follows immediately. If we calculate the solution of the HE as $\boldsymbol{\Phi}(t)$, then numerical methods involve multiplication of block matrices, containing polynomials of $\boldsymbol{N}(t)$. Therefore, the cost of these schemes should be counted in terms of matrix–matrix products (MMPs) units. Alternatively, for large systems we propagate the state vector itself, so the cost, in general, is different and should be counted as matrix–vector products (MVPs).

Finally, eq. (6.8) has inner separable structure $\boldsymbol{T} + \boldsymbol{V}(t)$ that can be used for additional optimisation.

6.1 LIE ALGEBRA

6.1.1 *Time-dependent case*

We start with a generalised version of eqs. (6.4) and (6.8), in which both antidiagonal blocks are time-dependent. We examine this type of systems and later deduce problem-specific properties from these computations.

Assume a splittable system of ODEs

$$\begin{gathered}\dot{\boldsymbol{u}} = \underbrace{\begin{bmatrix} \boldsymbol{O} & \boldsymbol{O} \\ \boldsymbol{N}(t) & \boldsymbol{O} \end{bmatrix}}_{\coloneqq \boldsymbol{A}(t)} \boldsymbol{u} + \underbrace{\begin{bmatrix} \boldsymbol{O} & \boldsymbol{M}(t) \\ \boldsymbol{O} & \boldsymbol{O} \end{bmatrix}}_{\coloneqq \boldsymbol{B}(t)} \boldsymbol{u}, \\ \dim \boldsymbol{N} = \dim \boldsymbol{M} = d \times d, \ \boldsymbol{u} \in \mathbb{R}^{2d}\end{gathered} \tag{6.9}$$

These splitting parts are block-nilpotent of index 2.

Following Section 4.2, we associate a Lie algebra with these splitting parts to express the solution in terms of the ME (4.5). Given a time step τ, the generators of the algebra read

$$\alpha_i \coloneqq \tau \begin{bmatrix} \boldsymbol{O} & \boldsymbol{O} \\ \bar{\boldsymbol{N}}_i & \boldsymbol{O} \end{bmatrix}, \quad \beta_i \coloneqq \tau \begin{bmatrix} \boldsymbol{O} & \bar{\boldsymbol{M}}_i \\ \boldsymbol{O} & \boldsymbol{O} \end{bmatrix}, \tag{6.10}$$

where $\bar{\boldsymbol{N}}_i$ and $\bar{\boldsymbol{M}}_i$ are linear combinations of $\boldsymbol{N}_j$ and $\boldsymbol{M}_j$ respectively. The exact form of these combinations depends on the choice of a quadrature rule but is similar to (4.27).

Recall that $\boldsymbol{N}_j \coloneqq \boldsymbol{N}\big(t_n + c_j \tau\big)$.

Now, let us examine elements spanned by α_i and β_i. Since generators are block nilpotent matrices, their commutators are quite simple. Par-

ticularly, some of these elements are same-shaped nilpotent matrices, and other are trivially zeroes:

$$\begin{aligned}
[\alpha_i,\alpha_j] &\equiv [\beta_i,\beta_j] \equiv \boldsymbol{O}\ \forall i,j;\\
[\alpha_i,\beta_j] &= \tau^2\begin{bmatrix} -\bar{\boldsymbol{M}}_j\bar{\boldsymbol{N}}_i & \boldsymbol{O} \\ \boldsymbol{O} & \bar{\boldsymbol{N}}_i\bar{\boldsymbol{M}}_j \end{bmatrix};\\
[\alpha_k,[\beta_j,\alpha_i]] &= \tau^3\begin{bmatrix} \boldsymbol{O} & \boldsymbol{O} \\ \bar{\boldsymbol{N}}_i\bar{\boldsymbol{M}}_j\bar{\boldsymbol{N}}_k + \bar{\boldsymbol{N}}_k\bar{\boldsymbol{M}}_j\bar{\boldsymbol{N}}_i & \boldsymbol{O} \end{bmatrix};\\
[\beta_k,[\beta_j,\alpha_i]] &= -\tau^3\begin{bmatrix} \boldsymbol{O} & \bar{\boldsymbol{M}}_j\bar{\boldsymbol{N}}_i\bar{\boldsymbol{M}}_k + \bar{\boldsymbol{M}}_k\bar{\boldsymbol{N}}_i\bar{\boldsymbol{M}}_j \\ \boldsymbol{O} & \boldsymbol{O} \end{bmatrix};\\
[\beta_i,[\beta_j,[\beta_k,\alpha_l]]] &\equiv [\alpha_i,[\alpha_j,[\alpha_k,\beta_l]]] \equiv \boldsymbol{O}\ \forall i,j,k,l.
\end{aligned} \tag{6.11}$$

Most importantly, these block nilpotent matrices are easily exponentiated (as shown in Section 2.4):

$$\mathrm{e}^{\alpha_i} = \begin{bmatrix} \boldsymbol{I} & \boldsymbol{O} \\ \tau\bar{\boldsymbol{N}}_i & \boldsymbol{I} \end{bmatrix}, \quad \mathrm{e}^{[\beta_k,[\alpha_i,\beta_j]]} = \begin{bmatrix} \boldsymbol{I} & \tau^3\left(\bar{\boldsymbol{M}}_j\bar{\boldsymbol{N}}_i\bar{\boldsymbol{M}}_k + \bar{\boldsymbol{M}}_k\bar{\boldsymbol{N}}_i\bar{\boldsymbol{M}}_j\right) \\ \boldsymbol{O} & \boldsymbol{I} \end{bmatrix}. \tag{6.12}$$

For that reason, it is logical to build a splitting scheme which is symmetric and includes easily computed commutators. As we have seen, this splitting leads to the cheap exponentials of nilpotent matrices, while symmetry and commutators facilitate finding good coefficients: some equations with the commutators are easier to solve and there appear free parameters for optimisation. A general representation for such a family may be the following:

$$\begin{aligned}
\Psi_s^{[p]} = \prod_{m=1}^{s} &\exp\left(\sum_{i=1}^{q} y_{m,i}\beta_i + \sum_{i,j,k} w_{m,i,j,k}[\beta_k,[\alpha_i,\beta_j]]\right)\\
&\times\exp\left(\sum_{i=1}^{q} x_{m,i}\alpha_i + \sum_{i,j,k} v_{m,i,j,k}[\alpha_k,[\beta_j,\alpha_i]]\right),
\end{aligned} \tag{6.13}$$

where coefficients $x_{m,i}$, $y_{m,i}$, $v_{m,i,j,k}$ and $w_{m,i,j,k}$ satisfy appropriate symmetry relations.

In principle, we strive to build methods whose computational cost is as low as possible. However, certain systems other commutators like $[\alpha_i,\beta_i]$ can be added if they are easy to calculate.

6.1.2 *Half-autonomous case*

After general considerations, we can proceed to the Lie algebra of the Hill eq. (6.4) and the wave eq. (6.8). Recall the split representation of their corresponding 1^{st}-order systems:

$$\boldsymbol{Z}(t) = \begin{bmatrix} \boldsymbol{O} & \boldsymbol{O} \\ \boldsymbol{N}(t) & \boldsymbol{O} \end{bmatrix} + \begin{bmatrix} \boldsymbol{O} & \boldsymbol{I} \\ \boldsymbol{O} & \boldsymbol{O} \end{bmatrix} =: \boldsymbol{A}(t) + \boldsymbol{B}. \tag{6.14}$$

The major difference is that the upper block of $\boldsymbol{B}$ is now the identity matrix $\boldsymbol{I}$, and this property leads to further simplifications in the Lie algebra, whose generators become

$$\alpha_i := \tau \begin{bmatrix} \boldsymbol{O} & \boldsymbol{O} \\ \bar{\boldsymbol{N}}_i & \boldsymbol{O} \end{bmatrix}, \quad \beta_1 \equiv \iota := \tau \begin{bmatrix} \boldsymbol{O} & \boldsymbol{I} \\ \boldsymbol{O} & \boldsymbol{O} \end{bmatrix}, \tag{6.15}$$

where $\bar{\boldsymbol{N}}_i$ are the derivatives of the Lagrange interpolating polynomial (4.16) of $\boldsymbol{N}(t)$, evaluated at the centre of time subinterval.

Thanks to the identity block $\boldsymbol{I}$ in ι, commutators (6.11) are significantly simplified.

$$\begin{aligned} [\alpha_i, \iota] &= \tau^2 \begin{bmatrix} -\bar{\boldsymbol{N}}_i & \boldsymbol{O} \\ \boldsymbol{O} & \bar{\boldsymbol{N}}_i \end{bmatrix}, \quad [\iota, [\iota, \alpha_i]] = -\tau^3 \begin{bmatrix} \boldsymbol{O} & 2\bar{\boldsymbol{N}}_i \\ \boldsymbol{O} & \boldsymbol{O} \end{bmatrix}, \\ [\alpha_i, [\alpha_k + \iota, \alpha_j]] &\equiv [\alpha_i, [\iota, \alpha_j]] = \tau^3 \begin{bmatrix} \boldsymbol{O} & \boldsymbol{O} \\ \bar{\boldsymbol{N}}_i \bar{\boldsymbol{N}}_k + \bar{\boldsymbol{N}}_k \bar{\boldsymbol{N}}_i & \boldsymbol{O} \end{bmatrix}. \end{aligned} \tag{6.16}$$

Compare the second and the third commutators above: the former requires only one multiplication of a matrix by a scalar, while in the latter linear combinations of matrices are multiplied.

Let us look at the explicit representation of the generators. As before, we take the 3-point 6^{th}-order GL quadrature rule, which is the minimum requirement for 6^{th}-order integrators and is also suitable for optimized 4^{th}-order methods. Obviously, ι does not change with quadrature rules, so the generators are:

$$\begin{aligned} \alpha_1 &= \tau \boldsymbol{A}_2 = \tau \begin{bmatrix} \boldsymbol{O} & \boldsymbol{O} \\ \boldsymbol{N}_2 & \boldsymbol{O} \end{bmatrix}, \quad \iota = \tau \boldsymbol{B} = \tau \begin{bmatrix} \boldsymbol{O} & \boldsymbol{I} \\ \boldsymbol{O} & \boldsymbol{O} \end{bmatrix}, \\ \alpha_2 &= \tau \frac{\sqrt{15}}{3} (\boldsymbol{A}_3 - \boldsymbol{A}_1) = \tau \frac{\sqrt{15}}{3} \begin{bmatrix} \boldsymbol{O} & \boldsymbol{O} \\ \boldsymbol{N}_3 - \boldsymbol{N}_2 & \boldsymbol{O} \end{bmatrix}, \\ \alpha_3 &= \tau \frac{10}{3} (\boldsymbol{A}_3 - 2\boldsymbol{A}_2 + \boldsymbol{A}_1) = \tau \frac{10}{3} \begin{bmatrix} \boldsymbol{O} & \boldsymbol{O} \\ \boldsymbol{N}_3 - 2\boldsymbol{N}_2 + \boldsymbol{N}_1 & \boldsymbol{O} \end{bmatrix}. \end{aligned} \tag{6.17}$$

From this point, depending on the problem at hand, slightly different strategies are possible for building new methods.

6.2 HILL EQUATION

Now, we turn our attention to the Hill equation (HE) in its (more compact) 1st-order system representation (6.4):

$$\frac{\mathrm{d}\boldsymbol{u}(t)}{\mathrm{d}t} = \boldsymbol{Z}(t)\boldsymbol{u}(t) \coloneqq \begin{bmatrix} \boldsymbol{O} & \boldsymbol{I} \\ \boldsymbol{N}(t) & \boldsymbol{O} \end{bmatrix} \boldsymbol{u}(t), \quad \boldsymbol{u}(t_0) = \boldsymbol{u}_0.$$

Recall that we assume $\boldsymbol{N}$ to be periodic and have a moderate dimension, so that we compute the solution as the fundamental matrix $\boldsymbol{\Phi}(t)$: $\boldsymbol{u}(t) = \boldsymbol{\Phi}(t)\boldsymbol{u}_0$, and we count computation cost in the units of d-dimensional matrix–matrix multiplications.

Assume the period of $\boldsymbol{N}(t)$ is T, thus $\boldsymbol{Z}(t) = \boldsymbol{Z}(t+T)$. Floquet theory states [17, 78, 95] that in this case $\boldsymbol{\Phi}(t+T) = \boldsymbol{\Phi}(t)\boldsymbol{\Phi}(T)$. Consequently, $\boldsymbol{u}(kT) = \boldsymbol{\Phi}^k(T)\,\boldsymbol{u}_0$ for an integer k.

If $\boldsymbol{N}(t)$ is real and symmetric, then $\boldsymbol{Z}(t)$ belongs to the symplectic algebra $Sp(2d,\mathbb{C})$ and $\boldsymbol{\Phi}(t)$ is symplectic. These properties formalise the motivation: system (6.4) is stable if all the $2d$ eigenvalues λ_i of $\boldsymbol{\Phi}(t)$ lie on the unit disc. As ME-based schemes preserve symplecticity, they do not push the eigenvalues out from the unit disc. Nevertheless, proposed methods will be valid for non-symmetric and non-periodic $\boldsymbol{N}(t)$, but their performance will depend on the problem structure.

Since the discretized HE has a relatively small dimension, we can, in a sense, simplify methods by taking a step back. To this end, we recombine the two 1st-grade elements of the Lie algebra: $\zeta_1 \coloneqq \alpha_1 + \iota$. As a result, we get formally the same Lie algebra as in Chapter 5. Therefore, we can reuse some of those results, albeit with some modifications.

Let us compare the explicit forms of generators of the HE and SE. From (6.17) it is easy to get the matrix form of ζ_i for the current problem in terms of linear $\bar{\boldsymbol{N}}_i$:

$$\zeta_1 = \tau \begin{bmatrix} \boldsymbol{O} & \boldsymbol{I} \\ \boldsymbol{N}_2 & \boldsymbol{O} \end{bmatrix}, \quad \zeta_i = \tau \begin{bmatrix} \boldsymbol{O} & \boldsymbol{O} \\ \bar{\boldsymbol{N}}_i & \boldsymbol{O} \end{bmatrix}, \; i \geq 2. \tag{6.18}$$

Although ζ_1 now does not contain the discrete Laplacian, it is still the most expensive element to exponentiate. Namely, $\exp(\tau\zeta_i)$, $i \geq 2$ does not require matrix multiplication at all, while $\exp(\tau\zeta_1)$ can be approximated to order $2m$ with symplecticity preservation at the cost of $m + 4/3$ MMP [17].

For illustration, we take $\Upsilon_2^{[6]}$ because it is a typical and short scheme of our QCF family. Its matrix form for the HE is the following:

$$\begin{aligned} \Upsilon_2^{[6]} &= \exp(-x_{1,2}\zeta_2 + x_{1,3}\zeta_3 + v[212]) \exp(x_{2,1}\zeta_1 - x_{2,2}\zeta_2 + x_{2,3}\zeta_3) \\ &\quad \times \exp(x_{2,1}\zeta_1 + x_{2,2}\zeta_2 + x_{2,3}\zeta_3) \exp(x_{1,2}\zeta_2 + x_{1,3}\zeta_3 + v[212]) \\ &= \begin{bmatrix} \boldsymbol{I} & \boldsymbol{O} \\ \tau\boldsymbol{C}_2^{[6]} & \boldsymbol{I} \end{bmatrix} \exp\left(\frac{\tau}{2} \begin{bmatrix} \boldsymbol{O} & \boldsymbol{I} \\ \boldsymbol{D}_2^{[6]} & \boldsymbol{O} \end{bmatrix} \right) \exp\left(\frac{\tau}{2} \begin{bmatrix} \boldsymbol{O} & \boldsymbol{I} \\ \boldsymbol{D}_1^{[6]} & \boldsymbol{O} \end{bmatrix} \right) \begin{bmatrix} \boldsymbol{I} & \boldsymbol{O} \\ \tau\boldsymbol{C}_1^{[6]} & \boldsymbol{I} \end{bmatrix}, \end{aligned}$$

where $\boldsymbol{D}_i^{[p]}$ are linear combinations of $\boldsymbol{N}_j$ and $\boldsymbol{C}_i^{[p]}$ are polynomials of $\boldsymbol{N}(t)$, arising from the commutator [212]. Assume the fundamental solution $\boldsymbol{\Phi}_{k-1}$ at step $k-1$ is divided into four $d \times d$ blocks. Then, considering the cost of symplectic approximation and the FSAL property, it will cost $33\,2/3$ $d \times d$-dimensional MMPs to calculate $\boldsymbol{\Phi}_k$.

For these methods, FSAL means that the outermost parts have the same structure and can be concatenated

It is possible to lower the method's cost by decomposing the exponentials. Indeed, since combinations $\boldsymbol{D}_i^{[p]}$ are constant and real-valued in each time subinterval, the following result can be applied [20]:

$$\exp\left(\tau\begin{bmatrix}\boldsymbol{O} & \boldsymbol{I}\\ \boldsymbol{D} & \boldsymbol{O}\end{bmatrix}\right) = \begin{bmatrix}\cosh\left(\tau\sqrt{\boldsymbol{D}}\right) & \frac{1}{\sqrt{\boldsymbol{D}}}\sinh\left(\tau\sqrt{\boldsymbol{D}}\right)\\ \sqrt{\boldsymbol{D}}\sinh\left(\tau\sqrt{\boldsymbol{D}}\right) & \cosh\left(\tau\sqrt{\boldsymbol{D}}\right)\end{bmatrix}.$$

If the spectral radius ρ of $\sqrt{\boldsymbol{D}}$ satisfies the condition $\rho(\sqrt{\boldsymbol{D}}) < \pi/\tau$, then it can be decomposed into the following product [12]:

$$\exp\left(\tau\begin{bmatrix}\boldsymbol{O} & \boldsymbol{I}\\ \boldsymbol{D} & \boldsymbol{O}\end{bmatrix}\right) \approx \begin{bmatrix}\boldsymbol{I} & \boldsymbol{O}\\ \boldsymbol{L}^{[s]} & \boldsymbol{I}\end{bmatrix}\begin{bmatrix}\boldsymbol{I} & \boldsymbol{U}^{[s+2]}\\ \boldsymbol{O} & \boldsymbol{I}\end{bmatrix}\begin{bmatrix}\boldsymbol{I} & \boldsymbol{O}\\ \boldsymbol{L}^{[s]} & \boldsymbol{I}\end{bmatrix}, \tag{6.19}$$

where the blocks $\boldsymbol{L}^{[s]}$ and $\boldsymbol{U}^{[s+2]}$ are Taylor series of the hyperbolic functions up to a sufficiently high order in τ. For $s = 13$ they read:

$$\begin{aligned}\boldsymbol{L}^{[s]} =& \sqrt{\boldsymbol{D}}\tanh\left(\tau\frac{\sqrt{\boldsymbol{D}}}{2}\right) = \frac{\tau\boldsymbol{D}}{2} - \frac{\tau^3\boldsymbol{D}^2}{24} + \frac{\tau^5\boldsymbol{D}^3}{240}\\ &- \frac{17\tau^7\boldsymbol{D}^4}{40320} + \frac{31\tau^9\boldsymbol{D}^5}{725760} - \frac{691\tau^{11}\boldsymbol{D}^6}{159667200} + \mathcal{O}\left(\tau^{13}\right);\\ \boldsymbol{U}^{[s+2]} =& \frac{1}{\sqrt{\boldsymbol{D}}}\sinh\left(\tau\sqrt{\boldsymbol{D}}\right) = \tau\boldsymbol{I} + \frac{\tau^3\boldsymbol{D}}{6} + \frac{\tau^5\boldsymbol{D}^2}{120} + \frac{\tau^7\boldsymbol{D}^3}{5040}\\ &+ \frac{\tau^9\boldsymbol{D}^4}{362880} + \frac{\tau^{11}\boldsymbol{D}^5}{39916800} + \frac{\tau^{13}\boldsymbol{D}^6}{6227020800} + \mathcal{O}\left(\tau^{15}\right).\end{aligned} \tag{6.20}$$

Note that the truncated series expansions of $\boldsymbol{L}^{[s]}$ and $\boldsymbol{U}^{[s+2]}$ can be computed simultaneously with only $\lfloor (s-1)/2 \rfloor$ MMPs. More precisely, if we want an overall 6^{th}-order approximation, then we need only 2 products to compute $\tau^5\boldsymbol{D}^3$, and with the same number of products $\boldsymbol{U}$ will have order $s+2$. By construction $\boldsymbol{L}^{[s]}$ and $\boldsymbol{U}^{[s+2]}$ are symmetric, and the s^{th}-order approximation of (6.19) is a symplectic matrix for any s.

$\lfloor x \rfloor$ is the floor function

Now, we can substitute each inner exponential of $\Upsilon_2^{[6]}$ by (6.19), taking into account that matrices with the $\boldsymbol{L}^{[s]}$ and $\boldsymbol{C}_i^{[6]}$ blocks commute:

$$\begin{aligned}\Upsilon_2^{[6,s]} = &\begin{bmatrix}\boldsymbol{I} & \boldsymbol{O}\\ \tau\boldsymbol{C}_2^{[6]} + \boldsymbol{L}_2^{[6,s]} & \boldsymbol{I}\end{bmatrix}\begin{bmatrix}\boldsymbol{I} & \boldsymbol{U}_2^{[6,s]}\\ \boldsymbol{O} & \boldsymbol{I}\end{bmatrix}\begin{bmatrix}\boldsymbol{I} & \boldsymbol{O}\\ \boldsymbol{L}_2^{[6,s]} + \boldsymbol{L}_1^{[6,s]} & \boldsymbol{I}\end{bmatrix}\\ &\times\begin{bmatrix}\boldsymbol{I} & \boldsymbol{U}_1^{[6,s]}\\ \boldsymbol{O} & \boldsymbol{I}\end{bmatrix}\begin{bmatrix}\boldsymbol{I} & \boldsymbol{O}\\ \tau\boldsymbol{C}_1^{[6]} + \boldsymbol{L}_1^{[6,s]} & \boldsymbol{I}\end{bmatrix}.\end{aligned} \tag{6.21}$$

The scheme retains the FSAL property, and the total cost per step is

$$\underbrace{1}_{[212]}+\underbrace{2\times\left\lfloor\frac{s-1}{2}\right\rfloor}_{\text{(6.19) twice}}+\underbrace{4\times 2}_{\text{4 stages}}=8+s,$$

meaning that even with $s=8$ the cost is twice lower compared to the original $\Upsilon_2^{[6]}$ method.

Algorithm 4: One step of the $\Upsilon_2^{[p,s]}$ $p=4,6$ methods

Data: initial time t_{k-1}, time step τ, initial value $\boldsymbol{\Phi}_{k-1}$

compute $\boldsymbol{N}\big(t_{k-1}+c_j\tau\big)$ with $c_1=\frac{1}{2}-\frac{\sqrt{15}}{10}$, $c_2=\frac{1}{2}$, $c_3=\frac{1}{2}+\frac{\sqrt{15}}{10}$;

compute linear combinations of $\boldsymbol{N}_j:=\boldsymbol{N}(t_{k-1}+c_j\tau)$

$$\begin{aligned}
\boldsymbol{C}_1^{[4]}&:=\frac{\big(10-\sqrt{15}\big)}{180}\boldsymbol{N}_1-\frac{1}{9}\boldsymbol{N}_2+\frac{\big(\sqrt{15}+10\big)}{180}\boldsymbol{N}_3,\\
\boldsymbol{D}_1^{[p]}&:=\frac{\big(15-8\sqrt{15}\big)}{180}\boldsymbol{N}_1+\frac{1}{3}\boldsymbol{N}_2+\frac{\big(15+8\sqrt{15}\big)}{180}\boldsymbol{N}_3,\\
\boldsymbol{D}_2^{[p]}&:=\frac{\big(15+8\sqrt{15}\big)}{180}\boldsymbol{N}_1+\frac{1}{3}\boldsymbol{N}_2+\frac{\big(15-8\sqrt{15}\big)}{180}\boldsymbol{N}_3,\\
\boldsymbol{C}_2^{[4]}&:=\frac{\big(\sqrt{15}+10\big)}{180}\boldsymbol{N}_1-\frac{1}{9}\boldsymbol{N}_2+\frac{\big(10-\sqrt{15}\big)}{180}\boldsymbol{N}_3;
\end{aligned}\tag{6.22}$$

if $p=6$ **then**

compute matrix polynomials

$$[212]=\frac{\tau^2(\boldsymbol{N}_1-\boldsymbol{N}_3)^2}{12960},\quad \boldsymbol{C}_i^{[6]}=\boldsymbol{C}_i^{[4]}+[212],\ i=1,2;\tag{6.23}$$

end

decompose $\boldsymbol{D}_m^{[p]}$, $m=1,2$ according to (6.19);

calculate approximations (6.20);

propagate $\boldsymbol{\Phi}_{k-1}$ with the corresponding method;

Result: value $\boldsymbol{\Phi}_k$.

6.3 TIME-DEPENDENT WAVE EQUATION

In this section we turn our attention to the one-dimensional example wave equation

$$\frac{\partial^2\psi(t,x)}{\partial t^2}=\frac{\partial^2\psi(t,x)}{\partial x^2}+V(t,x)\psi(t,x),$$

which after discretization and transformations (6.7) and (6.8) has the same block antidiagonal form as the HE:

$$\frac{\mathrm{d}\boldsymbol{u}(t)}{\mathrm{d}t} = \begin{bmatrix} \boldsymbol{O} & \boldsymbol{I} \\ \boldsymbol{N}(t) & \boldsymbol{O} \end{bmatrix} \boldsymbol{u}(t),\ \boldsymbol{u}(t_0) = \boldsymbol{u}_0,\ \boldsymbol{u}(t) \in \mathbb{R}^{2d} \tag{6.24}$$

where $\boldsymbol{N}(t) \coloneqq \boldsymbol{T} + \boldsymbol{V}(t)$ and its properties should be addressed when designing integrators.

Firstly, in contrast with the rather moderately sized HE where we need the fundamental matrix solution to estimate stability, we now have a large-dimensional system where we can suffice with the state vector $\boldsymbol{u}$. For this reason, it is convenient to propagate the solution by MVPs.

Secondly, $\boldsymbol{N}(t)$ is a negative definite and symmetric matrix. As a consequence, the fundamental solution is an oscillatory symplectic transformation, and it is desirable to construct symplectic integrators that could be effective in a wide frequency band.

Finally, in many cases the discretized potential $\boldsymbol{V}(t)$ is diagonal, and the Laplacian $\boldsymbol{T}$ is the most expensive part, so we will count the cost of methods as the number of matrix–vector product (MVP) with $\boldsymbol{T}$.

With these considerations for the WE, our initial step will be to split the system according to (6.14) and apply the results of Section 6.1 directly.

We will limit the order of new schemes to 6. Then, the corresponding Lie algebra has three generators: the lower-triangular α_i, $i = 1, 2, 3$, and ι with the identity matrix as the upper right block. The 6^{th}-order Magnus expansion (ME) truncation is

$$\begin{aligned} \Omega^{[6]} = \underbrace{\iota + \alpha_1 + \frac{1}{12}\alpha_3 + \frac{1}{12}[\alpha_2, \iota]}_{\text{up to order } 4} + \frac{1}{360}\big([\alpha_1, [\iota, \alpha_3]] - [\iota, [\alpha_3, \iota]]\big) \\ - \frac{1}{240}[\alpha_2, [\iota, \alpha_2]] + \frac{1}{720}\big([\iota, [\alpha_1, [\iota, \alpha_2]]] - [\alpha_1, [\iota, [\alpha_2, \iota]]]\big), \end{aligned} \tag{6.25}$$

hence 4 order conditions to reach order 4, and 9 to reach order 6.

6.3.1 *A general sixth-order method without commutators*

The first method we derive is based on slitting methods [30, 31] for the general time-dependent problem (6.9). It does not explicitly use the separable structure of $\boldsymbol{N} = \boldsymbol{T} + \boldsymbol{V}$.

We consider commutator-free s-stage 6^{th}-order that without loss of generality can be written as

$$\Psi_s^{[6]} = \prod_{m=1}^{s} \exp(y_m \iota) \exp\left(\sum_{i=1}^{3} x_{m,i} \alpha_i \right). \tag{6.26}$$

As before, we build symmetric compositions to lower the number of order conditions. For coefficients, it means that $x_{1,i} = 0$, $\forall i$ or $y_s = 0$; the rest should satisfy $x_{i,j} = (-1)^{j+1} x_{s-i,j}$, $i = 1,\dots,s-1$ and $y_i = y_{s+1-i}$, $i = 1,\dots,s$.

As a starting point, we take coefficients $x_{m,i}$ and y_m of an 11-stage 6^{th}-order scheme (6.26) from [31], optimised for constant $\boldsymbol{N}$ and $\boldsymbol{M}$. This tactic serves two purposes: it allows solving fewer order conditions for the time-dependent case and simultaneously results in a method efficient for autonomous problems. In that case commutators which involve only ι and α_1 vanish up to order 6, and he contribution of higher orders is minimised by adding more stages than strictly necessary. Despite the increase in cost, the scheme gains an improved accuracy and stability.

With the initial set of 'autonomous' coefficients, we have a scheme with 5 coefficients $x_{m,2}$ and 6 coefficients $x_{m,3}$ to solve 7 order conditions. The structure of these equation makes one of the $x_{m,2}$ and three of the $x_{m,3}$ as free parameters which are chosen to minimise the objective function $\sum_m (|x_{m,2}|^2 + |x_{m,3}|^2)$ to get smaller error. The solution is such that the scheme starts and ends with exponentials of ι:

$$
\begin{aligned}
x_{1,1} &= 0.184330483502666 & x_{4,1} &= 0.203764547132355\\
x_{1,2} &= -0.077156381298956 & x_{4,2} &= -0.030210447662554\\
x_{1,3} &= 0.015810389108601 & x_{4,3} &= 0.008289779992453\\
x_{2,1} &= -0.041056903297711 & x_{5,1} &= -0.011760166914960\\
x_{2,2} &= 0.014540760463978 & x_{5,2} &= -0.000581981140461\\
x_{2,3} &= 0.007553493094496 & x_{5,3} &= -0.006294997717777\\
x_{3,1} &= 0.133755679666750 & x_{6,1} &= 0.061932719821802\\
x_{3,2} &= -0.035178021085693 & x_{6,2} &= 0\\
x_{3,3} &= 0.012409269716451 & x_{6,3} &= 0.007797464944885
\end{aligned}
$$

$$
\begin{aligned}
y_1 = y_{12} &= 0.046487454790863\\
y_2 = y_{11} &= -0.060691671165643\\
y_3 = y_{10} &= 0.218466526463407\\
y_4 = y_9 &= 0.168053579483093\\
y_5 = y_8 &= 0.314392364170353\\
y_6 = y_7 &= -0.186708253742073.
\end{aligned}
$$

After obtaining the solution for the coefficients $x_{i,j}$, we repeat the familiar procedure: we use the 6^{th}-order GL quadrature to replace the generators α_i by corresponding linear combinations (6.17) of $\boldsymbol{N}_j$. Since

the generators in (6.26) are combined to retain nilpotent structure, the matrix form of the method is:

$$\Psi_{11}^{[6]} = \begin{bmatrix} \boldsymbol{I} & \tau b_{12}\boldsymbol{I} \\ \boldsymbol{O} & \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{I} & \boldsymbol{O} \\ \tau\bar{\boldsymbol{N}}_{11} & \boldsymbol{I} \end{bmatrix} \cdots \begin{bmatrix} \boldsymbol{I} & \boldsymbol{O} \\ \tau\bar{\boldsymbol{N}}_{1} & \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{I} & \tau b_{1}\boldsymbol{I} \\ \boldsymbol{O} & \boldsymbol{I} \end{bmatrix}, \tag{6.27}$$

where $\bar{\boldsymbol{N}}_m = \sum_{j=1}^{3} a_{m,j}\boldsymbol{N}_j$, $i = 1,\dots,11$. The coefficients $a_{m,j}$ are obtained through $\boldsymbol{A} = \boldsymbol{X}\boldsymbol{S}_{3\times 3}$, while y_i are unchanged: $b_m = y_m$.

In this method, the most expensive part is the action of the linear combination $\bar{\boldsymbol{N}}_m$ on the upper d-dimensional half of the vector $\boldsymbol{u}(t)$, and there are 11 multiplications of this type. On the other hand, the ι-blocks require multiplying a vector by a scalar, so their cost can be neglected. Consequently, the total cost is 11 d-dimensional MVPs.

6.3.2 *Methods with modified potentials*

In different contexts, for example, relativistic quantum mechanics, the WE takes the form of the Klein–Gordon–Fock equation (KGFE), in which the operator consists of two pasts: as the kinetic energy T and the variable mass $\mu(t)$ or a confining potential $V(t,x)$.

In contrast with the previous section, we will construct schemes which take into account the split structure of $\boldsymbol{N} = \boldsymbol{T} + \boldsymbol{V}$ and include (similarly to the SE) some commutators that correspond to the time-derivatives of $\boldsymbol{V}(t)$. This technique yields methods that predominantly use positive fractional time steps, which is not possible for classical splitting methods of order $p > 2$ [103], hence a smaller error at a noticeably reduced number of stages with a lower computational cost.

We begin with the autonomous case and develop the ideas from [15, 30, 32]. Consider both $\boldsymbol{A}$ and $\boldsymbol{B}$ in (6.9) to be independent of time. Then, the simplest form of the composition (6.13) is

$$\begin{aligned} \Psi_{s}^{[p]} &= \mathrm{e}^{b_s\iota}\mathrm{e}^{a_s\alpha_s}\cdots\mathrm{e}^{b_1\iota}\mathrm{e}^{a_1\alpha_1} \\ &= \begin{bmatrix} \boldsymbol{I} & \tau b_s\boldsymbol{B} \\ \boldsymbol{O} & \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{I} & \boldsymbol{O} \\ \tau a_s\boldsymbol{A} & \boldsymbol{I} \end{bmatrix} \cdots \begin{bmatrix} \boldsymbol{I} & \tau b_1\boldsymbol{B} \\ \boldsymbol{O} & \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{I} & \boldsymbol{O} \\ \tau a_1\boldsymbol{A} & \boldsymbol{I} \end{bmatrix}. \end{aligned} \tag{6.28}$$

The coefficients a_m and b_m are obtained through the solution of $\Psi_s^{[p]} = \mathrm{e}^{\tau(\boldsymbol{A}+\boldsymbol{B})+\mathcal{O}\left(\tau^{p+1}\right)}$. According to (6.16), the commutators for the autonomous case have the following familiar block structure:

$$[\boldsymbol{B},[\boldsymbol{B},\boldsymbol{A}]] = \begin{bmatrix} \boldsymbol{O} & -2\boldsymbol{N} \\ \boldsymbol{O} & \boldsymbol{O} \end{bmatrix}, \quad [\boldsymbol{A},[\boldsymbol{B},\boldsymbol{A}]] = \begin{bmatrix} \boldsymbol{O} & \boldsymbol{O} \\ 2\boldsymbol{N}^2 & \boldsymbol{O} \end{bmatrix}. \tag{6.29}$$

Therefore, it is natural to introduce them into the scheme without losing the nilpotent properties beneficial for matrix exponentiation.

The enhanced composition has more coefficients to solve arising order equations:

[ABA] is a shorthand for the nested commutator [A,[B,A]]

$$\Psi_s^{[p]} = \mathrm{e}^{\tau b_s \boldsymbol{B}+\tau^3 d_s [\boldsymbol{BBA}]}\mathrm{e}^{\tau a_s \boldsymbol{A}+\tau^3 c_s [\boldsymbol{ABA}]}\cdots \mathrm{e}^{\tau b_1 \boldsymbol{B}+\tau^3 d_1 [\boldsymbol{BBA}]}\mathrm{e}^{\tau a_1 \boldsymbol{A}+\tau^3 c_1 [\boldsymbol{ABA}]}, \tag{6.30}$$

so higher orders can be achieved with fewer stages with positive coefficients. A similar modification applied to the SE is called *modified potentials* [44, 88].

Let us now assess the cost of the enhancement. The action of $\mathrm{e}^{\tau a_m \boldsymbol{A}}$ on a vector $\boldsymbol{u} = [\boldsymbol{y}, \dot{\boldsymbol{y}}]^T$ requires one costly d-dimensional matrix–vector product $\boldsymbol{N}\boldsymbol{y}$. When $[\boldsymbol{A},[\boldsymbol{B},\boldsymbol{A}]]$ containing $\boldsymbol{N}^2$ is added, only one more product is needed because $\boldsymbol{N}\boldsymbol{y}$ can be reused. On the other hand, $\exp(\tau b_m \boldsymbol{B})\boldsymbol{u}$ costs virtually nothing and $[\boldsymbol{B},[\boldsymbol{B},\boldsymbol{A}]]$ is linear in $\boldsymbol{N}$, so the cost also increases by one multiplication. Therefore, the most expensive s-stage scheme with all the commutators included will require $3s$ d-dimensional MMPs. If a method of this type is symmetric, then it has the FSAL property, leading to a lower final cost.

For the time-dependent case the inner separable structure $\boldsymbol{N}(t) = \boldsymbol{T} + \boldsymbol{V}(t)$ becomes important. Following the strategy proposed in [30, 31, 32], new methods will use appropriate time-averaging combinations of $\boldsymbol{N}(t)$ as described in Chapter 3. We will use the 3-point GL quadrature for both (optimised) 4^{th}- and 6^{th}-order methods. Using this quadrature rule we can write out the explicit matrix of the generators α_i, ι:

$$\begin{aligned} \alpha_1 &= \tau \begin{bmatrix} \boldsymbol{O} & \boldsymbol{O} \\ \boldsymbol{T}+\boldsymbol{V}_2 & \boldsymbol{O} \end{bmatrix}, \quad \iota = \tau \begin{bmatrix} \boldsymbol{O} & \boldsymbol{I} \\ \boldsymbol{O} & \boldsymbol{O} \end{bmatrix}, \\ \alpha_2 &= \tau \frac{\sqrt{15}}{3} \begin{bmatrix} \boldsymbol{O} & \boldsymbol{O} \\ \bar{\boldsymbol{V}}_2 & \boldsymbol{O} \end{bmatrix}, \quad \alpha_3 = \tau \frac{10}{3} \begin{bmatrix} \boldsymbol{O} & \boldsymbol{O} \\ \bar{\boldsymbol{V}}_3 & \boldsymbol{O} \end{bmatrix}, \end{aligned} \tag{6.31}$$

while the commutators (6.16) become

$$\begin{aligned} [\alpha_k,[\beta_j,\alpha_i]] = [\alpha_k,[\iota,\alpha_i]] &= \tau^3 \begin{bmatrix} \boldsymbol{O} & \boldsymbol{O} \\ \bar{\boldsymbol{V}}_i\bar{\boldsymbol{V}}_k + \bar{\boldsymbol{V}}_k\bar{\boldsymbol{V}}_i & \boldsymbol{O} \end{bmatrix}, \\ [\beta_k,[\beta_j,\alpha_i]] = [\iota,[\iota,\alpha_i]] &= -\tau^3 \begin{bmatrix} \boldsymbol{O} & 2\bar{\boldsymbol{V}}_i \\ \boldsymbol{O} & \boldsymbol{O} \end{bmatrix}, \ i,k \geq 2. \end{aligned} \tag{6.32}$$

where $\bar{\boldsymbol{V}}_2 = \boldsymbol{V}_3 - \boldsymbol{V}_1$ and $\bar{\boldsymbol{V}}_3 = \boldsymbol{V}_3 - 2\boldsymbol{V}_2 - \boldsymbol{V}_1$. When they act on a vector $\boldsymbol{u}$, $\exp(\tau[\alpha_k,[\iota,\alpha_i]])\boldsymbol{u}$ requires 4 MVPs, while $\exp(\tau[\iota,[\iota,\alpha_i]])$ requires only one (as in the autonomous case). Consequently, it is advantageous

to consider compositions (6.13) that only involve the cheap commutators $[\imath,[\alpha_j,\imath]]$ but not $[\alpha_i,[\imath,\alpha_j]]$:

$$\begin{aligned}\Psi_s^{[p]} &= \prod_{m=1}^{s} \exp\left(y_m \imath + \sum_{i=3}^{3} w_{m,i}[\imath,[\imath,\alpha_i]]\right)\exp\left(\sum_{i=1}^{3} x_{m,i}\alpha_i\right)\\ &= \begin{bmatrix} \boldsymbol{I} & \tau b_m \boldsymbol{I} - \tau^3 d_{m,i}\bar{\boldsymbol{N}}_i \\ \boldsymbol{O} & \boldsymbol{I}\end{bmatrix}\begin{bmatrix} \boldsymbol{I} & \boldsymbol{O} \\ \tau\sum_{i=j}^{3} a_{m,j}\boldsymbol{N}_j & \boldsymbol{I}\end{bmatrix}.\end{aligned} \tag{6.33}$$

We build symmetric methods, so the coefficients should satisfy the following relations:

$$\begin{aligned} x_{m+1-k,i} &= (-1)^{i+1}x_{k,i};\\ y_{m+1-k} &= y_k,\ w_{m+1-k,i} = (-1)^{i+1}w_{k,i};\end{aligned} \qquad k = 1\dots s. \tag{6.34}$$

After sequential application of the BCH formula to (6.33), we get a particular version of the order conditions polynomial (3.22) that contains only odd powers of τ:

$$\begin{aligned} P_\tau^{[6]} &= p_{1,1}\imath + p_{1,2}\alpha_1 + p_{3,1}[\imath,[\alpha_1,\imath]] + p_{3,2}[\alpha_1,[\imath,\alpha_1]] + p_{3,3}\alpha_3 + p_{3,4}[\alpha_2,\imath]\\ &+ p_{5,1}[\imath,[\imath,[\alpha_1,[\imath,\alpha_1]]]] + p_{5,2}[\alpha_1,[\imath,[\alpha_1,[\imath,\alpha_1]]]] + p_{5,3}[\imath,[\alpha_3,\imath]]\\ &+ p_{5,4}[\alpha_1,[\imath,\alpha_3]] + p_{5,5}[\alpha_2,[\imath,\alpha_2]] + p_{5,6}[\imath,[\alpha_1,[\imath,\alpha_2]]]\\ &+ p_{5,7}[\alpha_1,[\imath,[\alpha_2,\imath]]] + \mathcal{O}\left(\tau^7\right).\end{aligned}$$

Compared to the ME, there are 4 additional commutators (e. g. $[\alpha_1,[\imath,\alpha_1]]$) of $\imath$ and α_1 that must be cancelled out to reach a desired order, totalling 6 conditions for order 4 and 13 — for order 6.

Let us now estimate the cost of the scheme eq. (6.33). A linear combination of α_i requires one MVP, in which the most expensive part is the discrete Laplacian $\boldsymbol{T}$. In the other exponentials, there is also one MVP, but if it turns out that in some stage m the commutator is not necessary (i. e. $d_{m,j} = 0$), then the cost virtually drops to zero, as it reduces to the multiplication of the vector by a scalar.

6.3.2.1 *Fourth-order methods*

In the previous parts of the work we have seen that optimising 4^{th}-order methods with a 6^{th}-order quadrature leads to notably lower errors. Therefore, we will use the same strategy here to solve some conditions at order 5, expecting a better accuracy.

For instance, the following 3-stage composition

$$\begin{aligned}\Psi_3^{[4]} = &\exp\left(\frac{1}{2}\alpha_1 + \frac{1}{48}[\alpha_1,[\imath,\alpha_1]]\right)\\ &\times\exp\left(\imath - \frac{1}{12}[\imath,[\imath,\alpha_1]]\right)\\ &\times\exp\left(\frac{1}{2}\alpha_1 + \frac{1}{48}[\alpha_1,[\imath,\alpha_1]]\right)\end{aligned} \tag{6.35}$$

satisfies all the autonomous conditions up to order 4. If we introduce the cheap commutators in the following way:

$$\begin{aligned}\Psi_3^{[4]} =& \exp(x_{1,1}\alpha_1 - x_{1,2}\alpha_2 + x_{1,3}\alpha_3)\\ &\times\exp(y_2\iota + w_{2,1}[\iota,[\iota,\alpha_1]] + w_{2,3}[\iota,[\iota,\alpha_3]])\\ &\times\exp(x_{1,1}\alpha_1 + x_{1,2}\alpha_2 + x_{1,3}\alpha_3)\end{aligned} \tag{6.36}$$

or

$$\begin{aligned}\Psi_3^{[4]} =& \exp(y_1\iota + w_{1,1}[\iota,[\iota,\alpha_1]] - w_{1,2}[\iota,[\iota,\alpha_2]] + w_{1,3}[\iota,[\iota,\alpha_3]])\\ &\times\exp(x_{2,1}\alpha_1 + x_{2,3}\alpha_3)\\ &\times\exp(y_1\iota + w_{1,1}[\iota,[\iota,\alpha_1]] + w_{1,2}[\iota,[\iota,\alpha_2]] + w_{1,3}[\iota,[\iota,\alpha_3]]),\end{aligned} \tag{6.37}$$

then there will be only 6 variables to solve 6 equations and no free parameters for any optimisation.

However, more intricate but cheap combinations have enough parameters to solve one 5^{th}-order equation $p_{5,3}$:

$$\begin{cases} p_{1,1} = p_{1,2} = 1; \quad \text{(consistency)}\\ p_{3,1} = p_{3,2} = 0,\ p_{3,3} = p_{3,4} = \frac{1}{12};\\ p_{5,3} = -\frac{1}{360}. \end{cases} \tag{6.38}$$

The schemes have 5 stages each but cost 3 MVPs (considering the FSAL property):

$$\begin{aligned}\Psi_{3,1}^{[4]} =& \exp\Big(\frac{1}{6}\iota - \frac{1}{144}[\iota,[\iota,\alpha_1]] + \frac{7}{8640}[\iota,[\iota,\alpha_3]]\Big)\\ &\times\exp\Big(\frac{1}{2}\alpha_1 + \frac{1}{8}\alpha_2 + \frac{1}{24}\alpha_3\Big)\exp\Big(\frac{2}{3}\iota\Big)\\ &\times\exp\Big(\frac{1}{2}\alpha_1 - \frac{1}{8}\alpha_2 + \frac{1}{24}\alpha_3\Big)\\ &\times\exp\Big(\frac{1}{6}\iota - \frac{1}{144}[\iota,[\iota,\alpha_1]] + \frac{7}{8640}[\iota,[\iota,\alpha_3]]\Big)\end{aligned} \tag{6.39}$$

and

$$\begin{aligned}\Psi_{3,2}^{[4]} =& \exp\Big(\frac{1}{6}\iota\Big)\exp\Big(\frac{1}{2}\alpha_1 + \frac{1}{8}\alpha_2 + \frac{1}{24}\alpha_3\Big)\\ &\times\exp\Big(\frac{2}{3}\iota - \frac{1}{72}[\iota,[\iota,\alpha_1]] + \frac{7}{4320}[\iota,[\iota,\alpha_3]]\Big)\\ &\times\exp\Big(\frac{1}{2}\alpha_1 - \frac{1}{8}\alpha_2 + \frac{1}{24}\alpha_3\Big)\exp\Big(\frac{1}{6}\iota\Big).\end{aligned} \tag{6.40}$$

By taking stages with more elements, it is possible to satisfy additional 5^{th}-order conditions besides (6.38) at the cost of one additional MVP. For example, $p_{5,6} = -p_{5,7} = 1/720$ are fulfilled by

$$\begin{aligned}\Psi_{4,1}^{[4]} =& \exp\Big(\frac{1}{6}\imath + \frac{1}{240}[\imath,[\imath,\alpha_1]] + \frac{1}{720}[\imath,[\imath,\alpha_2]] + w_{1,3}[\imath,[\imath,\alpha_3]]\Big)\\ &\times\exp\Big(\frac{1}{2}\alpha_1 + \frac{1}{8}\alpha_2 + \frac{1}{24}\alpha_3\Big)\\ &\times\exp\Big(\frac{2}{3}\imath - \frac{1}{45}[\imath,[\imath,\alpha_1]] + \frac{7-8640w_{1,3}}{4320}[\imath,[\imath,\alpha_3]]\Big)\\ &\times\exp\Big(\frac{1}{2}\alpha_1 - \frac{1}{8}\alpha_2 + \frac{1}{24}\alpha_3\Big)\\ &\times\exp\Big(\frac{1}{6}\imath + \frac{1}{240}[\imath,[\imath,\alpha_1]] - \frac{1}{720}[\imath,[\imath,\alpha_2]] + w_{1,3}[\imath,[\imath,\alpha_3]]\Big),\end{aligned} \tag{6.41}$$

where $w_{1,3}$ is a free parameter which we put to $w_{1,3} = 0$.

Alternatively, $p_{5,3} = -p_{5,4} = -1/360$ and (6.38) are satisfied by means of a composition that starts with α_i:

$$\begin{aligned}\Psi_{4,2}^{[4]} =& \exp\Big(\frac{3-\sqrt{3}}{6}\alpha_1 + \frac{1}{12}\alpha_2 + \frac{5-\sqrt{3}}{120}\alpha_3\Big)\\ &\times\exp\Big(\frac{1}{2}\imath + \frac{-2+\sqrt{3}}{48}[\imath,[\imath,\alpha_1]] + \frac{-2+\sqrt{3}}{960}[\imath,[\imath,\alpha_3]]\Big)\\ &\times\exp\Big(\frac{\sqrt{3}}{3}\alpha_1 + \frac{\sqrt{3}}{60}\alpha_3\Big)\\ &\times\exp\Big(\frac{1}{2}\imath + \frac{-2+\sqrt{3}}{48}[\imath,[\imath,\alpha_1]] + \frac{-2+\sqrt{3}}{960}[\imath,[\imath,\alpha_3]]\Big)\\ &\times\exp\Big(\frac{3-\sqrt{3}}{6}\alpha_1 - \frac{1}{12}\alpha_2 + \frac{5-\sqrt{3}}{120}\alpha_3\Big).\end{aligned} \tag{6.42}$$

6.3.2.2 *Sixth-order methods*

For methods of order 6, it is compulsory to solve the full set of 13 order conditions arising from the polynomial $P_\tau^{[6]}$.

As we strive to achieve minimal computational cost, we consider the shortest possible compositions. Methods with 4 or 5 (symmetric) stages either do not have enough parameters or do not yield real solutions.

We examine 6-stage symmetric composition. The first one is

$$\begin{aligned}\Psi^{[6]} =& \exp(x_{1,1}\alpha_1 - x_{1,2}\alpha_2 + x_{1,3}\alpha_3)\exp(y_1\imath)\\ &\times\exp(x_{2,1}\alpha_1 - x_{2,2}\alpha_2 + x_{2,3}\alpha_3)\exp(y_2\imath)\\ &\times\exp(x_{3,1}\alpha_1 - x_{3,2}\alpha_2 + x_{3,3}\alpha_3)\\ &\times\exp(y_3\imath + w_{3,1}[\imath,[\imath,\alpha_1]] + w_{3,3}[\imath,[\imath,\alpha_3]])\\ &\times\exp(x_{3,1}\alpha_1 x_{3,2}\alpha_2 + x_{3,3}\alpha_3)\\ &\times\exp(y_2\imath)\exp(x_{2,1},x_{2,2},x_{2,3})\\ &\times\exp(y_1\imath)\exp(x_{1,1}\alpha_1 x_{1,2}\alpha_2 + x_{1,3}\alpha_3)\end{aligned} \tag{6.43}$$

and has only one free parameter. Unfortunately, an exhaustive analysis has not found real solutions. However, it is possible to construct another composition with cost 6:

$$\begin{aligned}\Psi_6^{[6]} =& \exp(y_1\imath + w_{1,1}[\imath,[\imath,\alpha_1]] - w_{1,2}[\imath,[\imath,\alpha_2]] + w_{1,3}[\imath,[\imath,\alpha_3]])\\ &\times\exp(x_{1,1}\alpha_1 - x_{1,2}\alpha_2 + x_{1,3}\alpha_3)\\ &\times\exp(y_2\imath)\exp(x_{2,1}\alpha_1 - x_{2,2}\alpha_2 + x_{2,3}\alpha_3)\\ &\times\exp(y_3\imath)\exp(x_{3,1}\alpha_1 + x_{3,3}\alpha_3)\exp(y_3)\\ &\times\exp(x_{2,1}\alpha_1 + x_{2,2}\alpha_2 + x_{2,3}\alpha_3)\exp(y_2)\\ &\times\exp(x_{1,1}\alpha_1 + x_{1,2}\alpha_2 + x_{1,3}\alpha_3)\\ &\times\exp(y_1\imath + w_{1,1}[\imath,[\imath,\alpha_1]] + w_{1,2}[\imath,[\imath,\alpha_2]] + w_{1,3}[\imath,[\imath,\alpha_3]])\end{aligned} \tag{6.44}$$

Since the order conditions are non-linear equations, there are two families of solutions that depend on the same parameter $w_{1,3}$:

for $\Psi_{6a}^{[6]}$:	for $\Psi_{6b}^{[6]}$:
$x_{1,1} = 0.564248616311064$	$x_{1,1} = 0.911084237567662$
$x_{1,2} = -0.176690750968964$	$x_{1,2} = -0.089268394480535$
$x_{1,3} = -0.313516035197005$	$x_{1,3} = -0.028447058456873$
$x_{2,1} = -0.239362702177329$	$x_{2,1} = 0.174005954233266$
$x_{2,2} = -0.057187275163375$	$x_{2,2} = 0.075685506226831$
$x_{2,3} = 0.613049370231532$	$x_{2,3} = 0.034479151199232$
$x_{3,1} = 0.350228171732531$	$x_{3,1} = -1.170180383601855$
$x_{3,2} = 0$	$x_{3,2} = 0$
$x_{3,3} = -0.515733336735721$	$x_{3,3} = 0.071269147848614$
$y_1 = 0.186856563115511$	$y_1 = 0.402019603896500$
$y_2 = 0.552058166051478$	$y_2 = 0.532939685630854$
$y_3 = -0.238914729166989$	$y_3 = -0.434959289527354$
$w_{1,1} = -5.414454638226355\times10^{-26}$	$w_{1,1} = 5.647154890872904\times10^{-26}$
$w_{1,2} = -0.002294120141002$	$w_{1,2} = 0.000920165692567$
$w_{1,3} = 1/1000;$	$w_{1,3} = 0.$

The disadvantage is that not all the coefficients for the autonomous part are positive.

However, with one additional product and total cost 7, we obtain solution with positive coefficients for the autonomous part when $0.3 < x_{3,1} < 0.4$:

$$\begin{aligned}\Psi_{7,1}^{[6]} =& \exp(x_{1,1}\alpha_1 - x_{1,2}\alpha_2 + x_{1,3}\alpha_3)\exp(y_1\iota)\\ &\times\exp(x_{2,1}\alpha_1 - x_{2,2}\alpha_2 + x_{2,3}\alpha_3)\\ &\times\exp(y_2\iota + w_{2,1}[\iota,[\iota,\alpha_1]] - w_{2,2}[\iota,[\iota,\alpha_2]])\\ &\times\exp(x_{3,1}\alpha_1 - x_{3,2}\alpha_2 + x_{3,3}\alpha_3)\\ &\times\exp(y_3\iota)\\ &\times\exp(x_{3,1}\alpha_1 + x_{3,2}\alpha_2 + x_{3,3}\alpha_3)\\ &\times\exp(y_2\iota + w_{2,1}[\iota,[\iota,\alpha_1]] + w_{2,2}[\iota,[\iota,\alpha_2]])\\ &\times\exp(x_{2,1}\alpha_1 + x_{2,2}\alpha_2 + x_{2,3}\alpha_3)\\ &\times\exp(y_1\iota)\exp(x_{1,1}\alpha_1 + x_{1,2}\alpha_2 + x_{1,3}\alpha_3)\end{aligned} \tag{6.45}$$

Fixing $x_{3,1} = 1/3$, we get

$$\begin{aligned} x_{1,1} &= \ 0.051458286600180 \\ x_{1,2} &= -0.007607519969111 \\ x_{1,3} &= -0.007310852657106 \\ x_{2,1} &= \ 0.115208380066487 \\ x_{2,2} &= -0.064270386358967 \\ x_{2,3} &= \ 0.040001852838863 \\ x_{3,1} &= \ 1/3 \\ x_{3,2} &= \ 0.054976313297277 \\ x_{3,3} &= \ 0.008975666484911 \end{aligned} \qquad \begin{aligned} y_1 &= \ 0.058085523104699 \\ y_2 &= \ 0.269824135673790 \\ y_3 &= \ 0.344180682443021 \\ w_{2,1} &= -0.001784230468531 \\ w_{2,2} &= \ 0.000368903662768. \end{aligned} \tag{6.46}$$

At the cost of 8 products there are many possibilities. For instance, a family of methods $\Psi_{8,1}^{[6]}$ with one additional parameter y_0, added to the outermost stages of $\Psi_{7,1}^{[6]}$ (6.45):

$$\begin{aligned}\Psi_{8,1}^{[6]} =& \exp(y_0\iota)\exp(x_{1,1}\alpha_1 - x_{1,2}\alpha_2 + x_{1,3}\alpha_3)\\ &\times\exp(y_1\iota)\exp(x_{2,1}\alpha_1 - x_{2,2}\alpha_2 + x_{2,3}\alpha_3)\\ &\times\exp(y_2\iota + w_{2,1}[\iota,[\iota,\alpha_1]] - w_{2,2}[\iota,[\iota,\alpha_2]])\\ &\times\exp(x_{3,1}\alpha_1 - x_{3,2}\alpha_2 + x_{3,3}\alpha_3)\\ &\times\exp(y_3\iota)\\ &\times\exp(x_{3,1}\alpha_1 + x_{3,2}\alpha_2 + x_{3,3}\alpha_3)\\ &\times\exp(y_2\iota + w_{2,1}[\iota,[\iota,\alpha_1]] + w_{2,2}[\iota,[\iota,\alpha_2]])\\ &\times\exp(x_{2,1}\alpha_1 + x_{2,2}\alpha_2 + x_{2,3}\alpha_3)\exp(y_1\iota)\\ &\times\exp(x_{1,1}\alpha_1 + x_{1,2}\alpha_2 + x_{1,3}\alpha_3)\exp(y_0\iota).\end{aligned} \tag{6.47}$$

In this case we start from the coefficients (6.46) with $y_0 = 0$ and then increase y_0 to minimize $\sum_i \left(\left|y_i\right|^2 + \left|x_{i,1}\right|^2\right)$ which yields:

$$\begin{aligned}
x_{1,1} &= \ 0.153132961756204 \\
x_{1,2} &= -0.092876346639766 \\
x_{1,3} &= \ 0.022461739927107 \\
x_{2,1} &= \ 0.013533704910463 \\
x_{2,2} &= \ 0.050116838506557 \\
x_{2,3} &= \ 0.009170580973135 \\
x_{3,1} &= \ 1/3, \\
x_{3,2} &= -0.079279474173539 \\
x_{3,3} &= \ 0.010034345766425
\end{aligned}
\qquad
\begin{aligned}
y_0 &= \ 53/2000, \\
y_1 &= \ 0.156724739535823 \\
y_2 &= \ 0.145665509223912 \\
y_3 &= \ 0.342219502480528 \\
w_{2,1} &= -0.001891832186460 \\
w_{2,2} &= \ 0.001497252599416.
\end{aligned}$$

Alternatively, we can formally add 4 more parameters:

$$\begin{aligned}
\Psi^{[6]}_{8,2} = &\exp(x_{1,1}\alpha_1 - x_{1,2}\alpha_2 + x_{1,3}\alpha_3) \\
&\times \exp(y_1\iota)\exp(x_{2,1}\alpha_1 - x_{2,2}\alpha_2 + x_{2,3}\alpha_3) \\
&\times \exp(y_2\iota + w_{2,1}[\iota,[\iota,\alpha_1]] - w_{2,2}[\iota,[\iota,\alpha_2]]) \\
&\times \exp(x_{3,1}\alpha_1 - x_{3,2}\alpha_2 + x_{3,3}\alpha_3) \\
&\times \exp(y_3\iota)\exp(x_{4,1}\alpha_1 + x_{4,3}\alpha_3)\exp(y_3\iota) \\
&\times \exp(x_{3,1}\alpha_1 + x_{3,2}\alpha_2 + x_{3,3}\alpha_3) \\
&\times \exp(y_2\iota + w_{2,1}[\iota,[\iota,\alpha_1]] + w_{2,2}[\iota,[\iota,\alpha_2]]) \\
&\times \exp(x_{2,1}\alpha_1 + x_{2,2}\alpha_2 + x_{2,3}\alpha_3)\exp(y_1\iota) \\
&\times \exp(x_{1,1}\alpha_1 + x_{1,2}\alpha_2 + x_{1,3}\alpha_3),
\end{aligned} \tag{6.48}$$

but y_3 turns out to be fixed; $x_{4,2}$ should be zero due symmetry; $x_{4,3}$ does not influence the solution. Thus, only $x_{4,1}$ is left. The minimisation of the functional produces the following coefficients:

$$\begin{aligned}
x_{1,1} &= \ 0.050464730342452 \\
x_{1,2} &= -0.005130147412848 \\
x_{1,3} &= -0.011087056214938 \\
x_{2,1} &= \ 0.111201936324214 \\
x_{2,2} &= -0.065004872268242 \\
x_{2,3} &= \ 0.043422732113659 \\
x_{3,1} &= \ 1/3 \\
x_{3,2} &= -0.056691613668185 \\
x_{3,3} &= \ 0.009330990767946
\end{aligned}
\qquad
\begin{aligned}
x_{4,1} &= \ 1/100 \\
x_{4,2} &= \ 0 \\
x_{4,3} &= \ 0 \\
y_1 &= \ 0.051501998994974 \\
y_2 &= \ 0.273039285884142 \\
y_3 &= \ 0.175458715120884 \\
w_{2,1} &= -0.001789878228315 \\
w_{2,2} &= \ 0.000383001386749.
\end{aligned}$$

Method for separable diagonal potentials

In various noteworthy applications $\boldsymbol{V}(t)$ has a separable structure (4.24), that is $\boldsymbol{V}(t) = f(t)\boldsymbol{D}$, where $f(t)$ is a scalar function, representing some external force, and $\boldsymbol{D}$ is a diagonal matrix.

In this case, further improvement is possible thanks to as simple form of the commutators (6.32):

$$\begin{aligned} [\imath, \alpha_i] &= \tau^2 \begin{bmatrix} \bar{\boldsymbol{D}}_1 & \boldsymbol{O} \\ \boldsymbol{O} & -\bar{\boldsymbol{D}}_1 \end{bmatrix}, \ i \geq 2; \\ [\imath, [\imath, \alpha_i]] &= -\tau^3 \begin{bmatrix} \boldsymbol{O} & \bar{\boldsymbol{D}}_3 \\ \boldsymbol{O} & \boldsymbol{O} \end{bmatrix}; \quad [\alpha_i, [\imath, \alpha_j]] = \tau^3 \begin{bmatrix} \boldsymbol{O} & \boldsymbol{O} \\ \bar{\boldsymbol{D}}_4 & \boldsymbol{O} \end{bmatrix}, \end{aligned} \tag{6.49}$$

where $\bar{\boldsymbol{D}}_i$ are functions of $\boldsymbol{V}$ and, thus, are also diagonal matrices.

This simple structure allows introducing more parameters to a scheme. For instance, the following composition with some coefficients x_i does not contain α_1 (the kinetic part $\boldsymbol{T}$):

$$\begin{aligned} \exp\left(\begin{bmatrix} \bar{\boldsymbol{D}}_4 & \bar{\boldsymbol{D}}_5 \\ \bar{\boldsymbol{D}}_6 & -\bar{\boldsymbol{D}}_4 \end{bmatrix}\right) = \exp\big(&x_1\imath + x_2\alpha_2 + x_3\alpha_3 + x_4[\imath, \alpha_2] + x_5[\imath, \alpha_3] \\ &+ x_6[\alpha_2, [\imath, \alpha_2]] + x_7[\imath, [\imath, \alpha_2]] + x_8[\imath, [\imath, \alpha_3]]\big). \end{aligned} \tag{6.50}$$

It is a 2×2 block matrix, consisting of diagonal matrices $\bar{\boldsymbol{D}}_i$, and it has a low exponentiaton cost and contributes up to 8 independent parameters.

As for the autonomous part, we can use compositions with α_1:

$$\begin{aligned} x_1\imath + x_2[\imath, [\imath, \alpha_1]] &= \begin{bmatrix} \boldsymbol{O} & 2x_2\boldsymbol{T} + \bar{\boldsymbol{D}}_7 \\ \boldsymbol{O} & \boldsymbol{O} \end{bmatrix}; \\ x_3\alpha_1 + x_4[\alpha_1, [\imath, \alpha_1]] &= \begin{bmatrix} \boldsymbol{O} & \boldsymbol{O} \\ (y_1\boldsymbol{T} + \bar{\boldsymbol{D}}_8)(y_2\boldsymbol{T} + \bar{\boldsymbol{D}}_9) + \bar{\boldsymbol{D}}_{10} & \boldsymbol{O} \end{bmatrix}. \end{aligned} \tag{6.51}$$

Then, we add commutators with the same structure:

$$\begin{aligned} x_1\imath + x_2[\imath, [\imath, \alpha_1]] + x_3[\imath, [\imath, \alpha_2]] + x_4[\imath, [\imath, \alpha_3]] \\ = \begin{bmatrix} \boldsymbol{O} & 2x_2\boldsymbol{T} + \bar{\boldsymbol{D}}_{11} \\ \boldsymbol{O} & \boldsymbol{O} \end{bmatrix} \end{aligned} \tag{6.52}$$

and

$$\begin{aligned} x_1\alpha_1 + x_2\alpha_2 + x_3\alpha_3 + x_4[\alpha_1, [\imath, \alpha_1]] + x_5[\alpha_2, [\imath, \alpha_2]] \\ = \begin{bmatrix} \boldsymbol{O} & \boldsymbol{O} \\ (y_1 T + \bar{\boldsymbol{D}}_{12})(y_2 T + \bar{\boldsymbol{D}}_{13}) + \bar{\boldsymbol{D}}_{14} & \boldsymbol{O} \end{bmatrix}. \end{aligned} \tag{6.53}$$

The last two matrices are nilpotent; their exponentials will require 1 and 2 calculations of the action of $\boldsymbol{T}$ on a d-dimensional vectors, respectively.

Consequently, new methods can use additional parameters without significant increase in cost. Their general form would be similar to (6.33) but with additional commutator, contributing to the block diagonal stages:

$x^{dg}_{m,i}$ stand for the coefficients of α_i included to the block diagonal matrices (6.50)

$$\begin{aligned}\Psi^{[p]}_m = \prod_{m=1}^{s} \exp\Big(& y_m \iota + x^{dg}_{m,2}\alpha_2 + x^{dg}_{m,3}\alpha_3 + z_{m,2}[\iota,\alpha_2] + z_{m,3}[\iota,\alpha_3] \\ & + w_{m,1}[\iota,[\iota,\alpha_1]] + w_{m,3}[\iota,[\iota,\alpha_3]]\Big) \\ & \times \exp\Big(x_{m,1}\alpha_1 + x^{dg}_{m,2}\alpha_2 + x^{dg}_{m,3}\alpha_3 + v_{m,1,1}[\alpha_1,[\iota,\alpha_1]]\Big).\end{aligned} \tag{6.54}$$

The application of the BCH formula gives rise to 13 order conditions.

After an extensive analysis of this type of schemes, we propose

$$\begin{aligned}\Psi^{[6]}_5 = & \exp(y_1\iota + z_{1,2}[\iota,\alpha_2]) \\ & \times \exp(x_{2,1}\alpha_1 - x_{2,2}\alpha_2 + x_{2,3}\alpha_3) \\ & \times \exp(y_3\iota + z_{3,2}[\iota,\alpha_2]) \\ & \times \exp(x_{4,1}\alpha_1 - x_{4,2}\alpha_2 + x_{4,3}\alpha_3) \\ & \times \exp(y_5\iota + w_{5,1}[\iota,[\iota,\alpha_1]] + w_{5,3}[\iota,[\iota,\alpha_3]]) \\ & \times \exp(x_{4,1}\alpha_1 + x_{4,2}\alpha_2 + x_{4,3}\alpha_3) \\ & \times \exp(y_3\iota + z_{3,2}[\iota,\alpha_2]) \\ & \times \exp(x_{2,1}\alpha_1 + x_{2,2}\alpha_2 + x_{2,3}\alpha_3) \\ & \times \exp(y_1\iota + z_{1,2}[\iota,\alpha_2]).\end{aligned} \tag{6.55}$$

The solution

$$\begin{aligned} x_{2,1} &= 0.240042507426491 & y_1 &= 0.089100765990115 \\ x_{2,2} &= -0.066189698716673 & y_3 &= 0.286949960842075 \\ x_{2,3} &= 0.038622655574735 & y_5 &= 0.247898546335620 \\ x_{4,1} &= 0.259957492573509 & w_{5,1} &= -0.002855510275609 \\ x_{4,2} &= -0.068421380317335 & w_{5,3} &= -0.000317745321648 \\ x_{4,3} &= 0.003044011091932 & z_{1,2} &= -0.000976189642908 \\ & & z_{3,2} &= -0.005012400162261 \end{aligned} \tag{6.56}$$

leads to particularly small error terms at higher orders.

6.4 NUMERICAL EXAMPLES

In this section we study and demonstrate the performance of the new methods valid for solving systems of the form (6.4). Specifically, for the Hill equation we choose the following methods:

Table 6.1: The orders of decomposition of the two best performing methods

ω	1/125	1/25	1/5	1	5	25	125
				$\varepsilon = 1$			
s of $\Upsilon_{1c}^{[4,s]}$	6, 10	6, 10	6, 10	6, 10	12, 10	8, 12	8, 12
s of $\Upsilon_{2}^{[6,s]}$	10, 8	10, 8	10, 8	10, 8	12, 8	12, 8	12, 8
				$\varepsilon = 1/10$			
s of $\Upsilon_{1}^{[4,s]}$	10, 6	10, 6	10, 6	10, 12	12, 8	8, 12	8, 12
s of $\Upsilon_{2}^{[6,s]}$	8, 10	8, 10	8, 10	8, 10	12, 8	12, 8	12, 8

- decomposed QCF methods $\Upsilon_2^{[4opt,s]}$, $\Upsilon_2^{[6,s]}$, $\Upsilon_{5,1a}^{[8,s]}$, $\Upsilon_{5,2b}^{[8,s]}$ that have shown their efficiency in the SE numerical example;
- splitting schemes for the WE obtained in this chapter;
- decomposed CF methods: $\mathrm{CF}_3^{[4opt,s]}$, $\mathrm{CF}_5^{[6,s]}$, $\mathrm{CF}_{11}^{[8,s]}$;
- symplectic RKN methods (3.26) and (3.27);
- explicit RKs of orders 4, 6, and 8;
- implicit symplectic RKGL methods of orders 4 and 6.

For the numerical experiments with the WE we do not use the QCF methods, but include the scheme (6.55).

6.4.1 *Mathieu equation*

The first performance test is executed employing the Mathieu equation

$$\ddot{y} + (\omega^2 + \varepsilon \cos 2t) y = 0, \tag{6.57}$$

written as a first-order system (6.4).

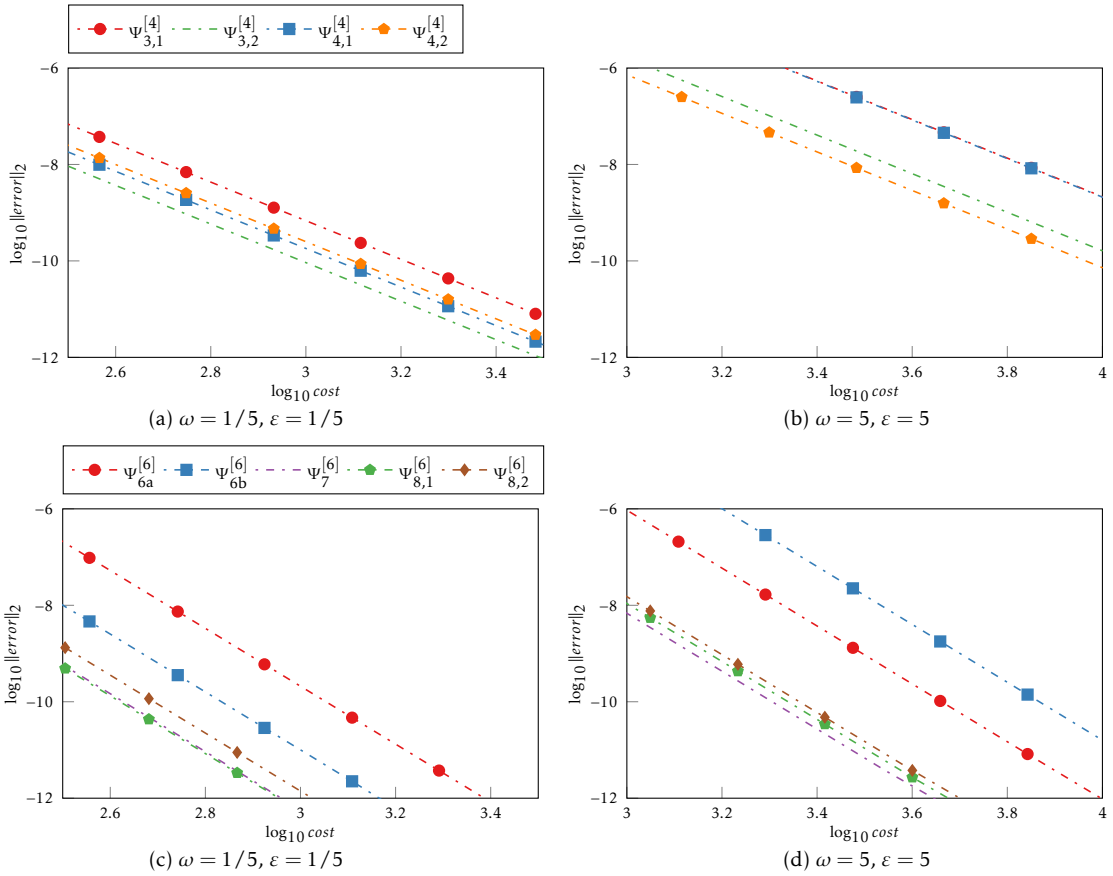
We will intergate on $[0;\ \pi]$ *throughout this subsection*

First, we want to know a reasonable order of decomposition for the $\Upsilon^{[p,s]}$ and $\mathrm{CF}^{[p,s]}$ methods. To this end, we take a 4^{th}-order and a 6^{th}-order method and integrate for $t \in [0;\ \pi]$ with the identity matrix $\boldsymbol{I}_{2\times 2}$ as the initial condition and then measure the norm of the error of the fundamental matrix $\boldsymbol{\Phi}$ at the final time. This procedure is repeated for different time steps and different choices of ε, ω, and s. Table 6.1 summarises the values of s that provided the best performances for $\omega = 5^k$, $k = -3, \ldots, 3$ (the better one comes first). We observe that good results are obtained when the order of decomposition (6.19) is $s = p + 4$, which we will use in our experiments.

To compare the performance of the schemes, we consider two cases: weaker time dependence with $\omega = 1/5$, $\varepsilon = 1/5$ and a stronger one with

$\omega = 5$, $\varepsilon = 5$. We integrate for one period and then plot the error in the fundamental matrix solution at the final time versus the computation cost in units of MMPs in double logarithmic scale. In each figure the axis range of subfigures is the same to simplify the comparison of different orders.

First, we make in-family comparison of the splitting methods to chose only one of each order because we expect them to perform similarly. Figure 6.1 shows that the best ones are $\Psi^{[4]}_{3,2}$ and $\Psi^{[6]}_{7}$.



(a) $\omega = 1/5$, $\varepsilon = 1/5$ (b) $\omega = 5$, $\varepsilon = 5$

(c) $\omega = 1/5$, $\varepsilon = 1/5$ (d) $\omega = 5$, $\varepsilon = 5$

Figure 6.1: Efficiency graphs of the splitting methods with commutators for the Mathieu equation.

Now we can compare different families of methods. From Figures 6.2 to 6.4 we observe that the standard explicit and implicit RK methods in all cases (except one low-oscillatory) perform worse than the more specialised methods. We also see that the ME-based decomposed integrators $\Upsilon^{[p,s]}$ perform noticeably better when there is a strong time dependence with high frequency ω. Nevertheless, they and the new Magnus–splitting schemes produce good results in the less oscillatory

scenario, in which the highly optimised RKN are expected to be the most efficient. It is possible thanks to a comparable cost, lowered by the decomposition (6.19).

As we have seen, the performance of the methods changes with parameters. Now, we examine how the accuracy depends on the frequency ω. We fix $\varepsilon = 1$, put the time step $\tau := \pi/20$ and measure the norm of the error in $\boldsymbol{\Phi}$ for $\omega \in [0, 10]$, $\Delta\omega = 5/1000$. The results are shown in Figure 6.5. We observe that the new QCF methods show a slower error growth as frequency of the problem increases, albeit the CF methods are more accurate in some cases. In this context, it is important that the latter require 2–3 times more MMPs.

To illustrate how the methods are applied to the problem of finding stability regions, we take $\tau := \pi/5$ fix $\varepsilon := 5$ and consider $\omega \in [0;\ 5]$. We compute the distance $|\lambda_i - 1 + \varepsilon_M|$ between the eigenvalues and the unit circle. From Figure 6.6 we can conclude that the explicit RK methods with few time steps cannot adequately find the regions of stability. On the other hand, every symplectic methods shows good results, and with larger time steps the 4^{th}-order methods show better accuracy than the higher-order methods (which is a quite common case). Consequently, the overall strategy may be the following: first, do a fast search with a low-order method; then, rectify the boundaries of the regions with high-order methods with small time step.

We add the machine ε_M to avoid singularities in $\log\|error\|_2$ when plotting

6.4.2 *Hill equation*

The second benchmark we use is the matrix Hill equation:

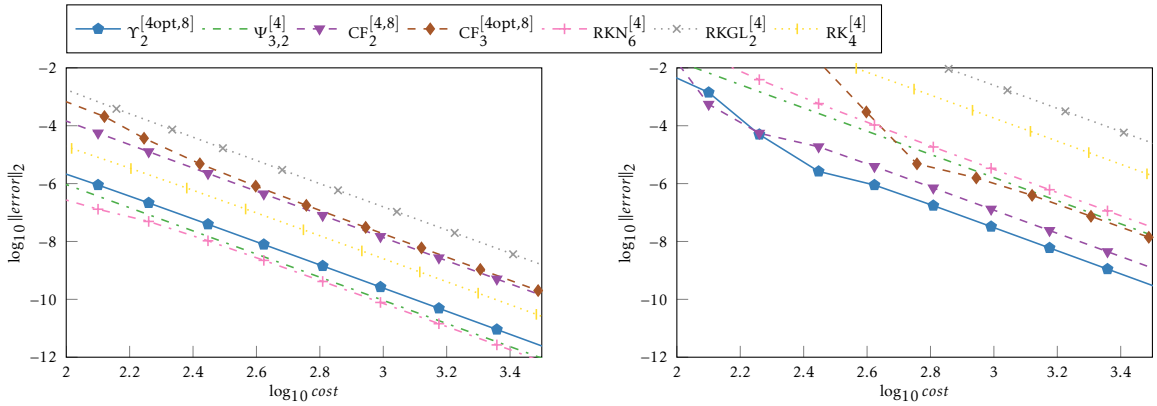
$$\ddot{\boldsymbol{Y}} + (\boldsymbol{B}_0 + \boldsymbol{B}_1 \cos 2t + \boldsymbol{B}_2 \cos 4t)\boldsymbol{Y} = 0 \tag{6.58}$$

where $\boldsymbol{B}_0$, $\boldsymbol{B}_1$, $\boldsymbol{B}_2$ have dimension $d \times d$. We put $\boldsymbol{B}_0 := d^2\boldsymbol{I} + \boldsymbol{P}$, where $\boldsymbol{P}$ is a Pascal matrix, that is,

$$p_{1,i} = p_{i,1} = 1,\ p_{i,j} = p_{i-1,j} + p_{i,j-1}, \quad 1 < i, j \le d,$$

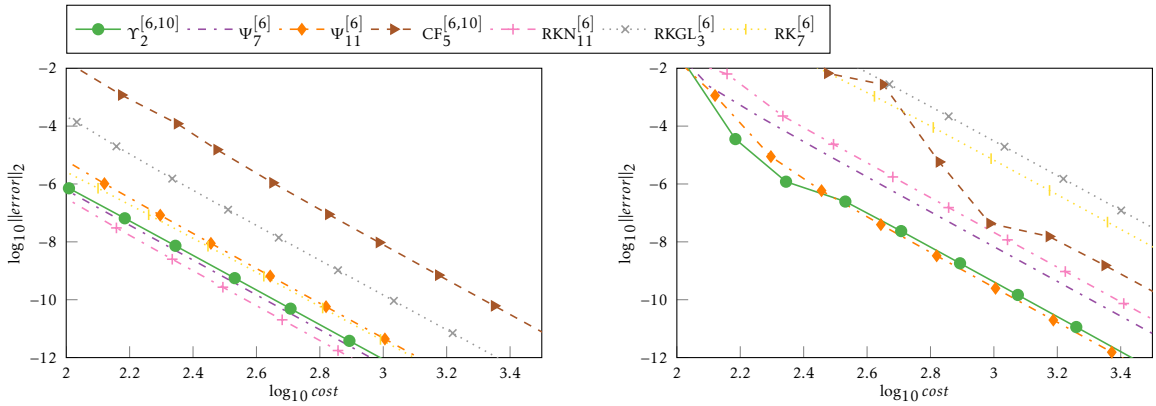
and set $\boldsymbol{B}_1 := \varepsilon\boldsymbol{I}$, $\boldsymbol{B}_2 := \varepsilon/10\boldsymbol{I}$ We consider $d \in \{5,\ 7\}$, $\varepsilon := d$ and $\varepsilon := d/10$. As in the Mathieu equation example, we integrate for $t \in [0;\ \pi]$ and measure norm of the error in the fundamental matrix solution $\boldsymbol{\Phi}$ at the final time.

As for in the Mathieu example, the best splitting methods have turned out to be $\Psi^{[4]}_{3,2}$ and $\Psi^{[6]}_{7}$, so we omit their comparison graphs. Figures 6.7 to 6.12 show the performance of the schemes for various parameters. Thanks to their lower computational cost, the decomposed $\Upsilon^{[p,s]}$, as long as the Magnus–splitting $\Psi^{[6]}_{11}$, methods consistently produce better results in both oscillatory (larger d) and nearly autonomous (small ε) cases.



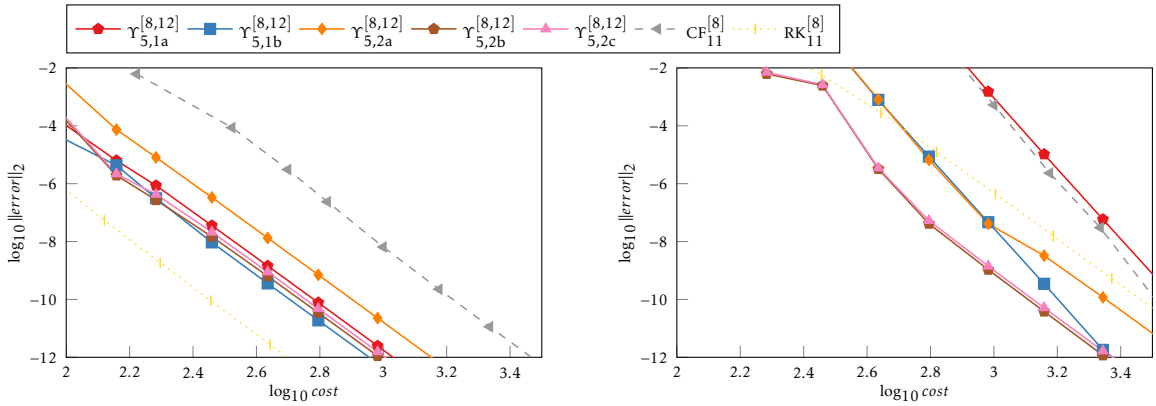
(a) $\omega = 1/5$, $\varepsilon = 1/5$ (b) $\omega = 5$, $\varepsilon = 5$

Figure 6.2: Efficiency graphs of the 4^{th}-order families for the Mathieu equation with slow and fast time dependence.



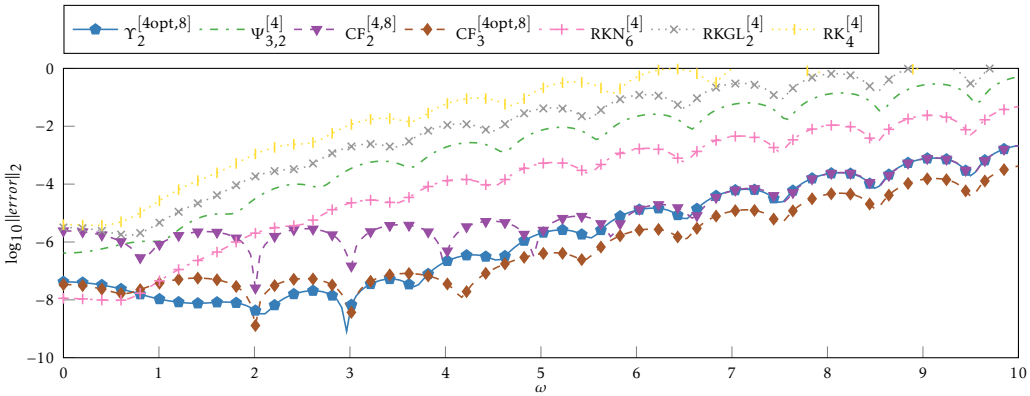
(a) $\omega = 1/5$, $\varepsilon = 1/5$ (b) $\omega = 5$, $\varepsilon = 5$

Figure 6.3: Mathieu: efficiency graphs of the 6^{th}-order methods.

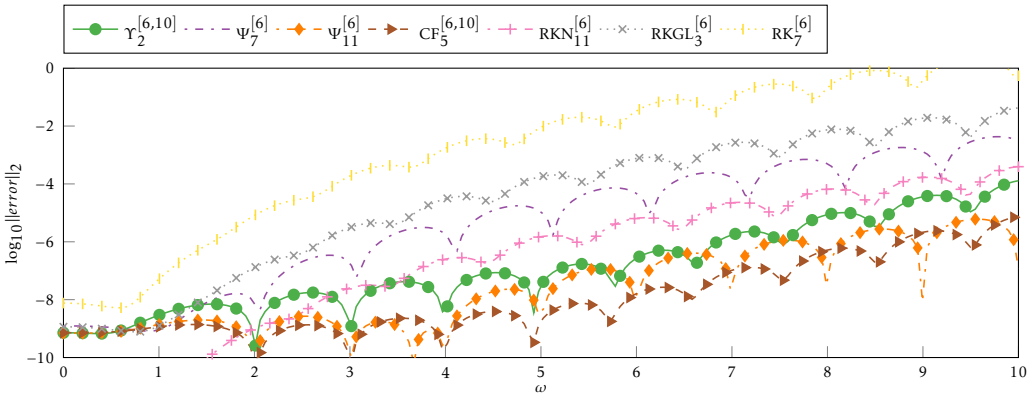


(a) $\omega = 1/5$, $\varepsilon = 1/5$ (b) $\omega = 5$, $\varepsilon = 5$

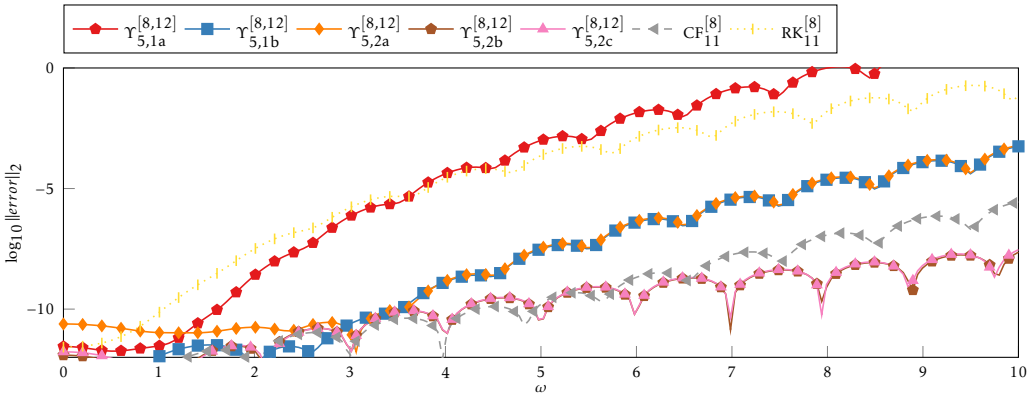
Figure 6.4: Mathieu: efficiency graphs of the 8^{th}-order methods.



(a) 4^{th}-order methods

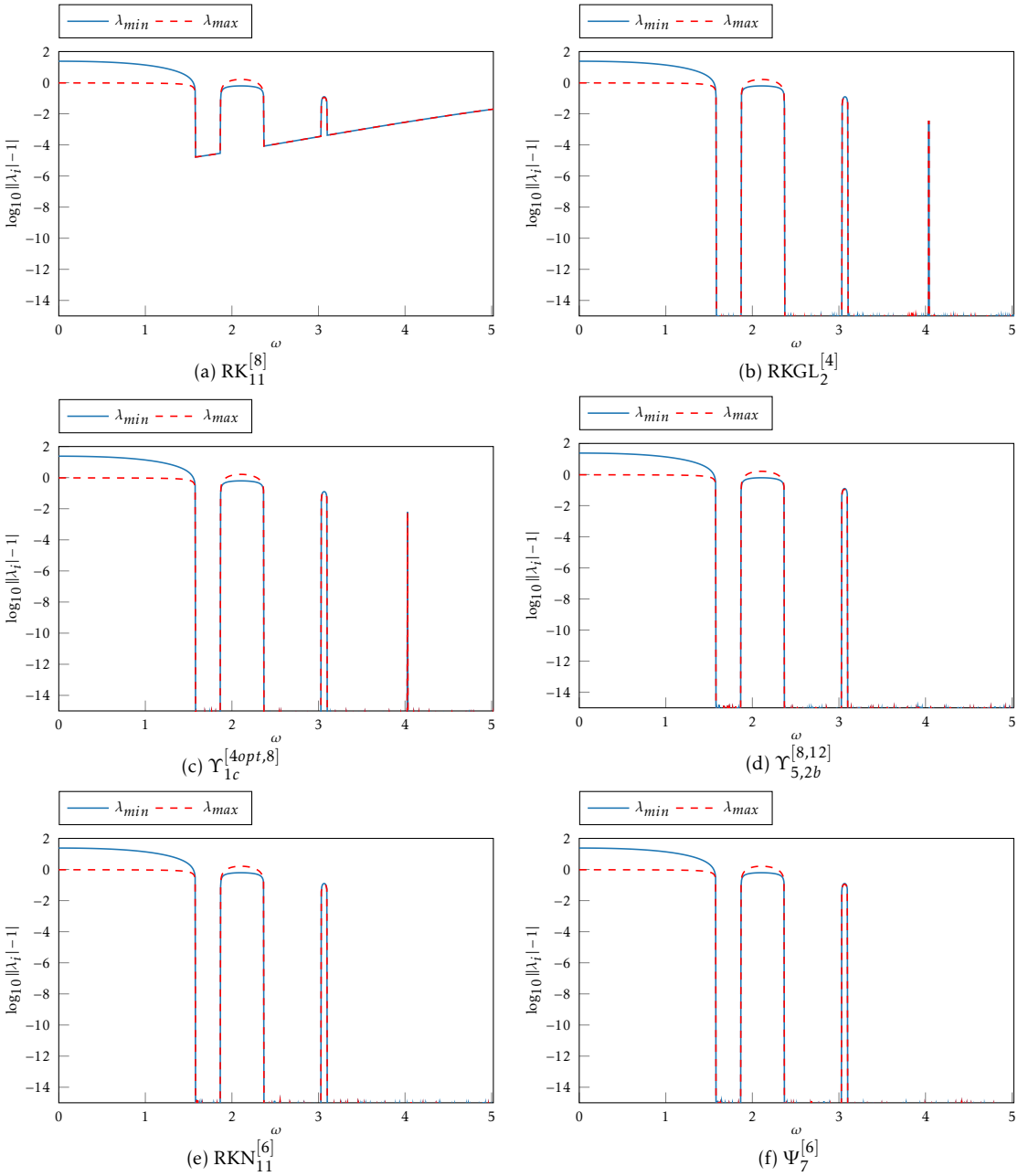


(b) 6^{th}-order methods



(c) 8^{th}-order methods

Figure 6.5: Error growth vs. the frequency ω of the Mathieu equation.

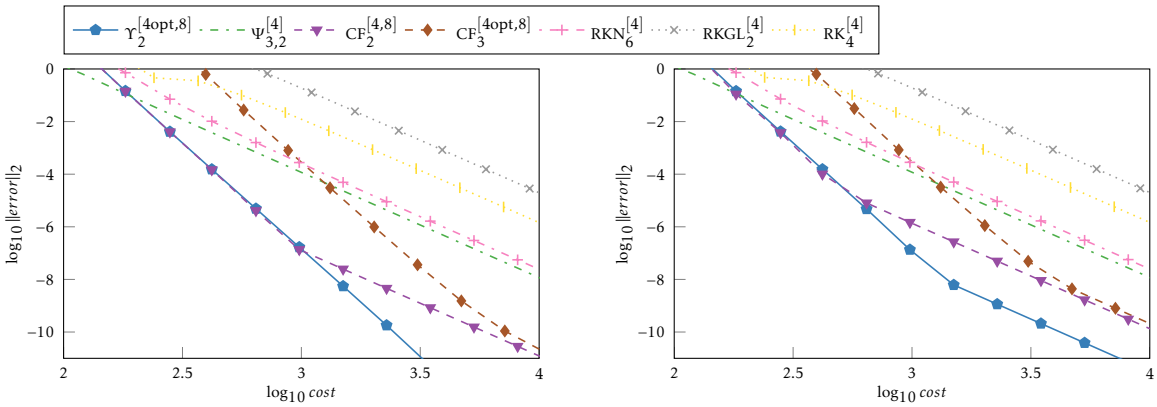


(a) $\mathrm{RK}_{11}^{[8]}$ (b) $\mathrm{RKGL}_{2}^{[4]}$

(c) $\Upsilon_{1c}^{[4opt,8]}$ (d) $\Upsilon_{5,2b}^{[8,12]}$

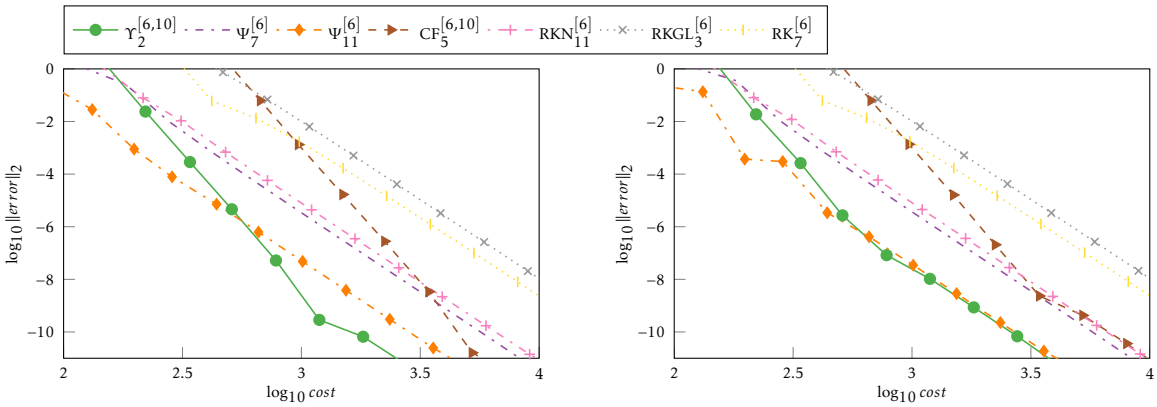
(e) $\mathrm{RKN}_{11}^{[6]}$ (f) $\Psi_{7}^{[6]}$

Figure 6.6: Stability regions of the Mathieu equation with $\varepsilon = 5$ obtained by different methods.



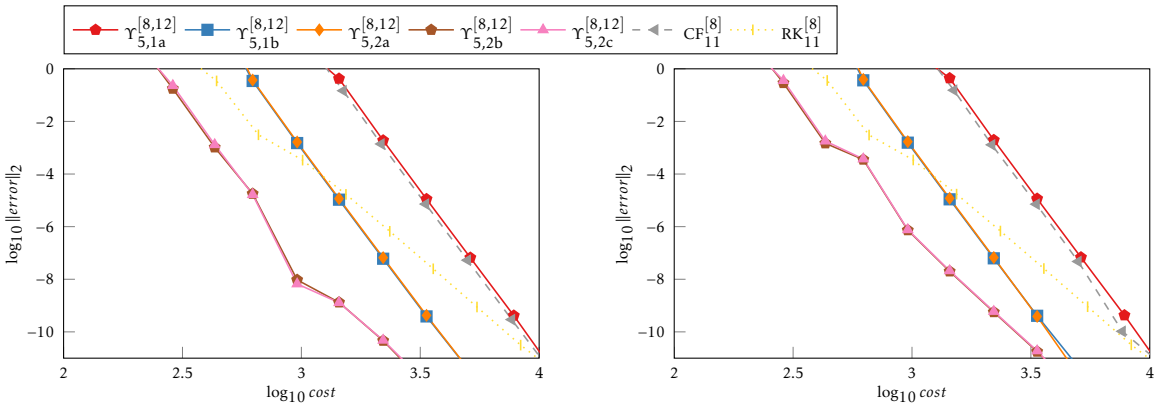
(a) $d = 5$, $\varepsilon = 5/10$ (b) $d = 5$, $\varepsilon = 5$

Figure 6.7: Efficiency graphs of the 4^{th}-order families for the Hill equation with $d = 5$.



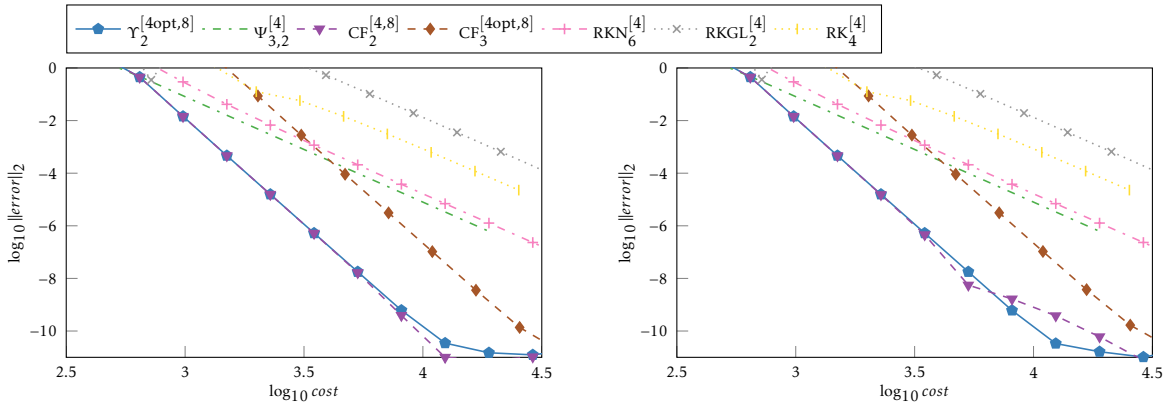
(a) $d = 5$, $\varepsilon = 5/10$ (b) $d = 5$, $\varepsilon = 5$

Figure 6.8: HE: efficiency graphs of the 6^{th}-order methods, $d = 5$.



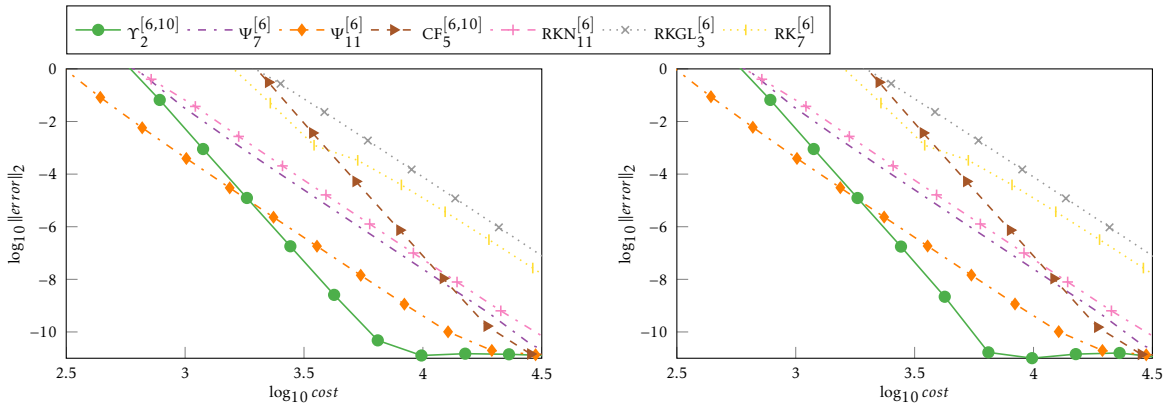
(a) $d = 5$, $\varepsilon = 5/10$ (b) $d = 5$, $\varepsilon = 5$

Figure 6.9: HE: efficiency graphs of the 8^{th}-order methods, $d = 5$.



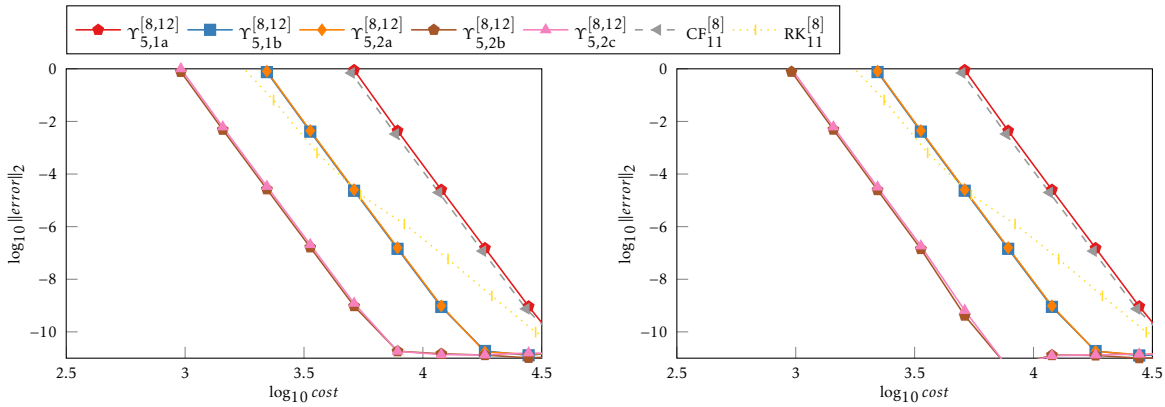
(a) $d = 7$, $\varepsilon = 7/10$ (b) $d = 7$, $\varepsilon = 7$

Figure 6.10: Efficiency graphs of the 4^{th}-order families for the Hill equation with $d = 7$.



(a) $d = 7$, $\varepsilon = 7/10$ (b) $d = 7$, $\varepsilon = 7$

Figure 6.11: HE: efficiency graphs of the 6^{th}-order methods, $d = 7$.



(a) $d = 7$, $\varepsilon = 7/10$ (b) $d = 7$, $\varepsilon = 7$

Figure 6.12: HE: efficiency graphs of the 8^{th}-order methods, $d = 7$.

6.4.3 *Wave equation*

First, we consider the wave equation with time-dependent potential with amplitude σ and frequency ω:

$$\begin{aligned} \partial_t^2 \psi(t,x) &= \partial_x^2 \psi(t,x) - \sigma\Big(1 + \frac{1}{5}\cos\omega t\Big) x^2 \psi(t,x); \\ \psi(t_0,x) &= \mathrm{e}^{-\frac{1}{2}(x-3)^2} + \mathrm{e}^{-\frac{1}{2}(x+2)^2},\ \dot\psi(t_0,x) = 0; \\ \psi(t,-10) &\equiv \psi(t,10). \end{aligned} \tag{6.59}$$

We choose the spatial interval to be $[-10;\ 10]$ and integrate for $t \in [0;\ 10\pi]$. For the numerical experiments we take a spatial mesh with $d = 128$ points and two (low and high) values of the frequency ω and the coefficient σ. As before, we compute the reference solution with a sufficiently small time step. Then, for each method we plot the error norm versus the number of MVPs. The results are shown in the Figure 6.13.

When σ and ω are small, the oscillations are low, and the equation becomes 'almost' harmonic oscillator. The best methods to solve this type of problem is the new Magnus–splitting method $\Psi^{[6]}_{11}$, which is designed to solve the autonomous harmonic oscillator with high precision by exploiting several free parameters for optimisation. Unfortunately, for higher-oscillatory example the new methods do not show performance improvement. On the other hand, their cost is lower, for example, 14 MVPs of $\Psi^{[6]}_{7}$ versus 22 for the $\mathrm{RKN}^{[6]}_{11}$, which leaves room for further optimisation of this family of schemes. Nevertheless, the most of the new schemes are better than the RK methods.

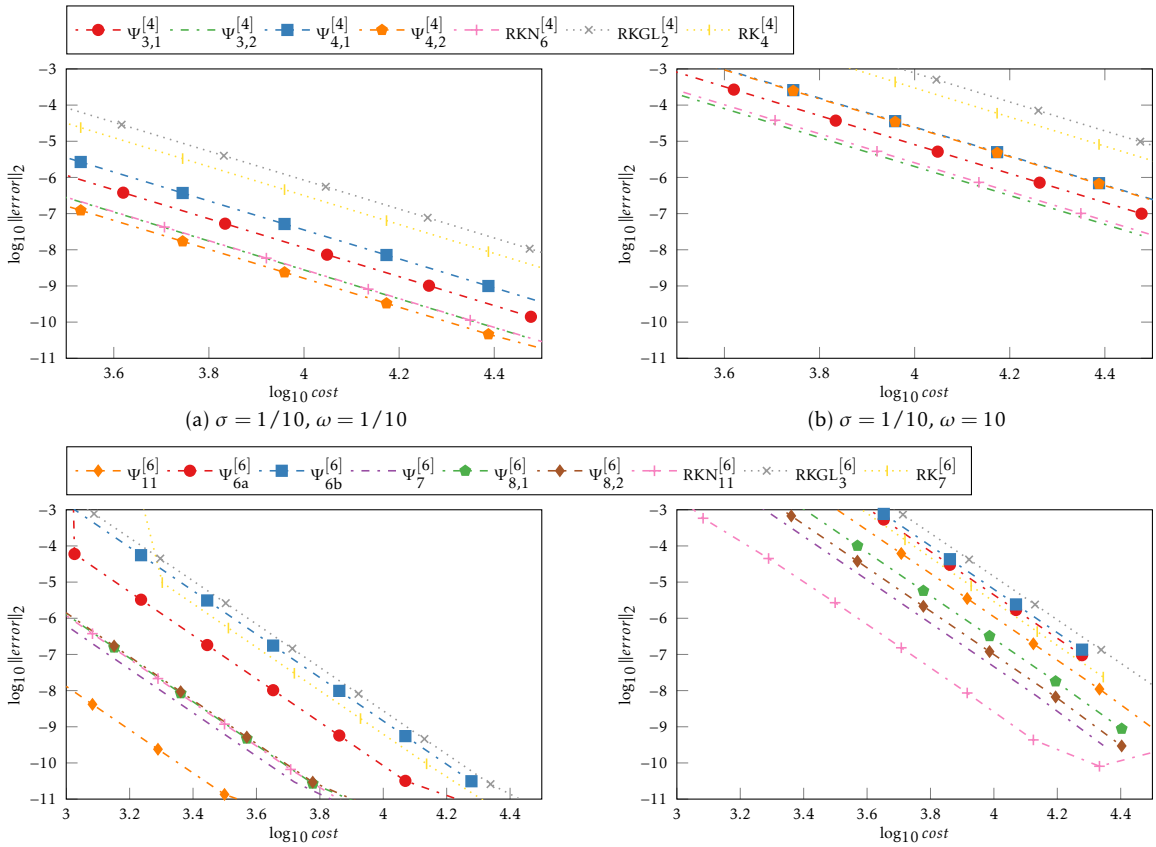
Regarding order 4, we observe that the introduction of cheap commutators to the schemes results in methods with a performance comparable to the $\mathrm{RKN}^{[4]}_{6}$ method, but with a 1.5–2 times lower computational cost.

6.4.4 *Klein–Gordon–Fock equation*

The second example is the Klein–Gordon equation with time-dependent mass [35] with the same initial and boundary condition as in the previous example:

$$\begin{aligned} \partial_t^2 \psi(t,x) &= \partial_x^2 \psi(t,x) - \frac{\mu^2}{(1+t)^2}\psi(t,x), \\ \psi(t_0,x) &= \mathrm{e}^{-\frac{1}{2}(x-3)^2} + \mathrm{e}^{-\frac{1}{2}(x+2)^2},\ \dot\psi(t_0,x) = 0; \\ \psi(t,-10) &\equiv \psi(t,10), \end{aligned} \tag{6.60}$$

which we integrate over time interval $[0, 10\pi]$ with $\mu = 1/5$ and $\mu = 5$.



(a) $\sigma = 1/10$, $\omega = 1/10$ (b) $\sigma = 1/10$, $\omega = 10$

(c) $\sigma = 1$, $\omega = 1/10$ (d) $\sigma = 1$, $\omega = 10$

Figure 6.13: Efficiency of the methods for the wave equation.

Overall, the results shown in Figure 6.14 are similar to the previous examples: RKN methods, $\Psi^{[6]}_{11}$ and $\Psi^{[6]}_{7}$ show comparable high efficiently. The important difference it that the specifically tailored $\Psi^{[6]}_{5}$ shows notable improvement, both when initial mass μ is small, and the time dependence of the problem contributes less, and when μ is relatively large.

6.5 CONCLUSIONS

In this chapter we have presented three classes of numerical integrators. The algebraic structure of the considered problems has allowed us to build families of symplectic schemes with low computational cost.

The first family, based on the Magnus expansion, addresses problems of moderate dimension, like the Mathieu and the matrix Hill equation. In this case matrix–matrix product (MMP) are feasible and the solution

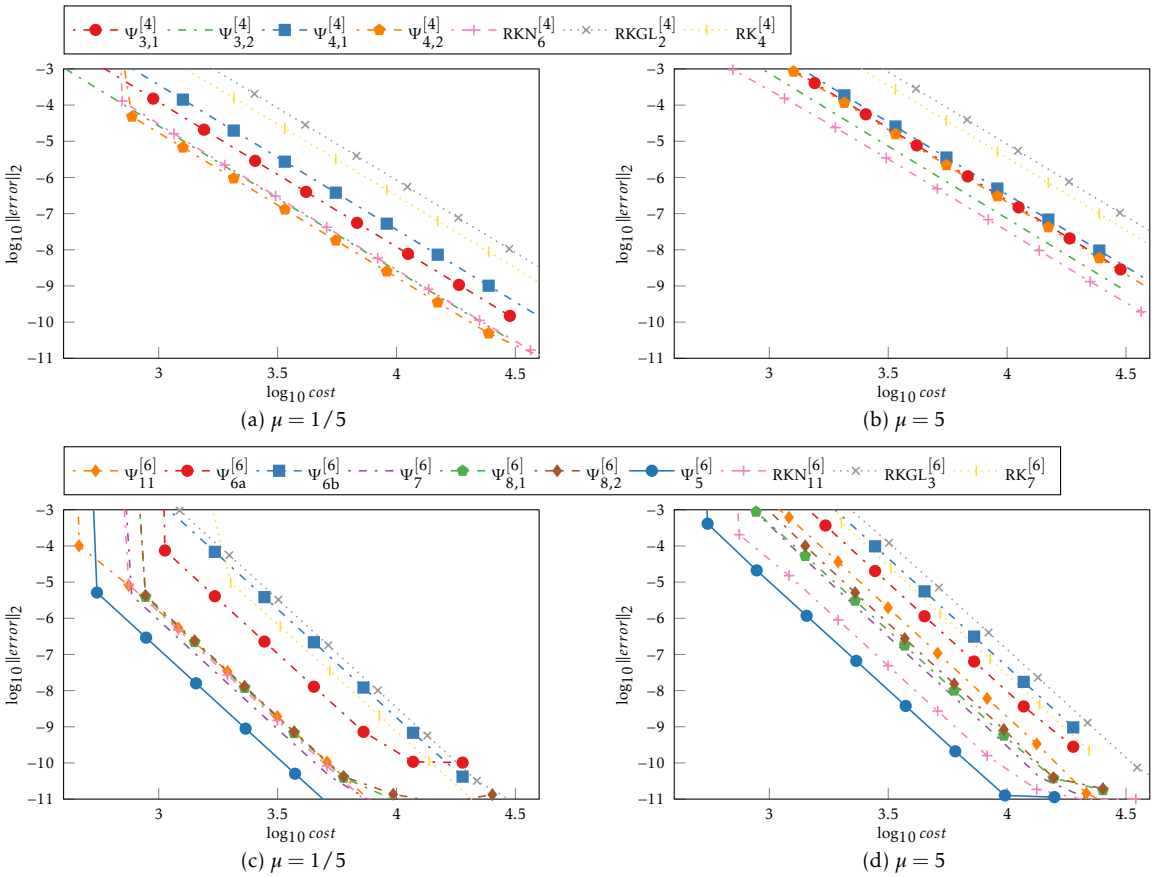


Figure 6.14: Efficiency of the methods for the Klein–Gordon–Fock equation.

is obtained in terms of the fundamental matrix, which allows to examine the system stability via its eigenvalues.

The second and the third family employ the splitting approach and approximate the solution given by the Magnus expansion (ME). The former does not contain commutators and is highly optimised for the harmonic oscillator, while the latter incorporates cheap commutators which leads to a reduced number of order conditions to be satisfied by the coefficients of the schemes. Commutators allow overcoming the order barrier on the negative coefficients, which allows for better accuracy at a low computational cost. These methods are advantageous when the system dimension is large, which appears, for example, from the discretization of PDEs like the wave equation. It is worth noting that such a setup leaves possibility to build methods, using more optimising coefficients and keeping their cost lower that of the methods from the literature. Moreover, since the HE and the WE have similar structure, the splitting schemes can also be applied to the HE.

The numerical experiments have showed all new methods perform efficiently on the scalar Mathieu equation and the matrix Hill equation, but the Magnus-based family shows better results for highly oscillatory problems.

The 4^{th}- and 6^{th}-order splitting schemes, in comparison with the highly optimised schemes from the literature, have shown their superiority for moderately oscillatory problems. Moreover, a notably lower computational cost leaves room for further optimisation by adding more stages.

7

NON-LINEAR CASE: KEPLER PROBLEM

In the earlier chapters we have built Magnus expansion-based methods for linear problems: the Schrödinger equation, the Hill equation, and the WE. Fortunately, a similar approach can be used to design integrators for non-linear systems. In this chapter, we illustrate the idea by the example of the time-dependent Kepler problem.

THE KEPLER PROBLEM, in which two bodies interact by a central force proportional to the inverse square of the distance between them, is one of the most studied dynamical systems in classical mechanics. It is a typical example of an integrable system [9] often used as benchmark for numerical integration methods for differential equations, ranging from Runge–Kutta (RK) schemes with adaptive step size to multistep methods [55]. More importantly, its geometric structure is the reason of great attention in the context of structure-preserving numerical methods, such as energy-preserving and symplectic integrators [52, 99].

In many-body problems, which describe the motion of planetary systems, the Kepler problem plays a fundamental role, because in an appropriate set of coordinates (e. g., heliocentric or Jacobi [69]), can be derived from a Hamiltonian of the form

$$H(\boldsymbol{q},\boldsymbol{p}) = H_K(\boldsymbol{q},\boldsymbol{p}) + \varepsilon H_I(\boldsymbol{q},\boldsymbol{p}), \tag{7.1}$$

where H_K corresponds to the Keplerian motion of each planet, εH_I is a perturbation with a small parameter ε introduced by the interplanetary interaction. In one of the early works [119] a symmetric 2^{nd}-order symplectic scheme was successfully used to study the chaotic behaviour of the solar system, and since then many efficient symplectic integrators have been designed with this *near-integrable* structure in mind [11, 24, 51, 70, 79].

Each of the aforementioned schemes require solving transcendental equations to determine with great accuracy the phase state of each body subjected to the Hamiltonian H_K. In practice, it is carried out by numerical integration as follows. We write the Hamiltonian function for the Kepler problem as

$$H(\boldsymbol{q},\boldsymbol{p}) = \frac{1}{2}\boldsymbol{p}^T\boldsymbol{p} - \mu\frac{1}{r}, \tag{7.2}$$

where $\mu = GM$, G is the gravitational constant, $M := m_1 + m_2$ is the total mass, and $\boldsymbol{p}, \boldsymbol{q} \in \mathbb{R}^3$, $r = \|\boldsymbol{q}\| = \sqrt{\boldsymbol{q}^T \boldsymbol{q}}$. The mapping advancing the solution in time from t_0 to t can be expressed as

$$(\boldsymbol{q}(t), \boldsymbol{p}(t)) \equiv \Phi_K(t, \boldsymbol{q}_0, \boldsymbol{p}_0, \mu) = (f_q \boldsymbol{q}_0 + g_q \boldsymbol{p}_0, f_p \boldsymbol{q}_0 + g_p \boldsymbol{p}_0) \tag{7.3}$$

in terms of functions f_q, f_p, g_q, g_p that are determined through the mapping $\Phi_K(t, \boldsymbol{q}, \boldsymbol{p}, \mu)$ [29, 47]:

$$\begin{gathered} r := \|\boldsymbol{q}\|,\ u = \boldsymbol{q}^T \boldsymbol{p},\ E = \frac{1}{2}\boldsymbol{p}^T \boldsymbol{p} - \mu \frac{1}{r},\ a\ = -\frac{\mu}{2E}, \\ w = \sqrt{\frac{\mu}{a^3}},\ \sigma = 1 - \frac{r}{a},\ \psi = \frac{u}{wa^2},\ wt = x - \sigma \sin x + \psi(1 - \cos x); \\ f_q = 1 + \frac{a(\cos x - 1)}{r}, \quad g_q = t + \frac{\sin x - x}{w}, \\ f_p = -\frac{aw \sin x}{r(1 - \sigma \cos x + \psi \sin x)}, \quad g_p = 1 + \frac{\cos x - 1}{1 - \sigma \cos x + \psi \sin x}, \end{gathered} \tag{7.4}$$

where x is evaluated by numerical iteration. Depending on the time step and the values of $\boldsymbol{q}$ and $\boldsymbol{p}$, it usually suffices to make 2–5 iterations to reach the round-off accuracy.

Various astronomical problems are modelled by (7.1) with explicitly time-dependent mass. Some examples of system with the loss of mass are: evolving planetary systems [2], the revolution of exoplanets around binary star systems [92] with stellar loss mass [115], and the two-body problem with varying mass [93, 97]. In those situations one has to solve in an accurate and efficient way the dynamics of the Hamiltonian

$$H(t, \boldsymbol{q}, \boldsymbol{p}) \equiv T + V(t) = \frac{1}{2}\boldsymbol{p}^T \boldsymbol{p} - \mu(t) \frac{1}{r}, \tag{7.5}$$

where now $\mu(t)$ is a time-variant mass. A natural question arises: can the design approach, valid for the linear problems, be applied to time-dependent non-linear equations? The answer is positive, and in this chapter we show how to adapt the QCF quasi-Magnus integrators from the previous chapter to the Kepler problem. The strategy used is generally the same, and we will highlight the main differences, appearing from non-linearity.

7.1 LIE DERIVATIVES AND POISSON BRACKETS

In contrast to the linear case, the derivation process of the non-linear schemes is done by introducing the formalism of *Lie derivatives* and *Lie transformations* [1, 9, 29]. The autonomous equations of motion are (7.2)

$$\dot{\boldsymbol{q}} = \boldsymbol{p}, \quad \dot{\boldsymbol{p}} = -\mu \frac{\boldsymbol{q}}{r^3}, \tag{7.6}$$

and the corresponding Hamiltonian vector is $\boldsymbol{x}_H = (\boldsymbol{\nabla}_p H, -\boldsymbol{\nabla}_q H)^T$. We introduce the Lie derivative $L_{\boldsymbol{x}_H}$ (the generalisation of commutator), whose action on a differentiable function $G(\boldsymbol{q},\boldsymbol{p})$ is

$$L_{\boldsymbol{x}_H} G = (\boldsymbol{J}\boldsymbol{\nabla}_{\boldsymbol{u}} H)^T \boldsymbol{\nabla}_{\boldsymbol{u}} G = -(\boldsymbol{\nabla}_{\boldsymbol{u}} H)^T \boldsymbol{J}\boldsymbol{\nabla}_{\boldsymbol{u}} G = -\{H, G\} = \{G, H\}. \tag{7.7}$$

Here $\boldsymbol{u} := (\boldsymbol{q},\boldsymbol{p})^T$, and $\boldsymbol{J}$ is the basic canonical matrix (2.5). The Poisson bracket $\{H, G\}$ of scalar functions $H(\boldsymbol{q},\boldsymbol{p})$ and $G(\boldsymbol{q},\boldsymbol{p})$ is defined as

$$\{G, H\} = \sum_i \left(\frac{\partial G}{\partial q_i}\frac{\partial H}{\partial p_i} - \frac{\partial G}{\partial p_i}\frac{\partial H}{\partial q_i} \right).$$

Consequently, in terms of the Poisson bracket (7.6) can be written in a form similar to (2.4)

$$\dot{\boldsymbol{u}}_i = \{\boldsymbol{u}_i, H\}. \tag{7.8}$$

If G is a vector function, then $L_{\boldsymbol{x}_H}$ in (7.7) acts on each of its components.

Let φ_t be the exact flow corresponding to (7.6). For any infinitely differentiable map G, the Taylor series of $G(\varphi_t(\boldsymbol{u}_0))$ at $t = t_0$ is given by

$$G(\varphi_t(\boldsymbol{u}_0)) = \sum_{k\geq 0} \frac{(t-t_0)^k}{k!} (L^k_{\boldsymbol{x}_H} G)(\boldsymbol{u}_0) = \exp\Big((t-t_0) L_{\boldsymbol{x}_H}\Big)[G](\boldsymbol{u}_0), \tag{7.9}$$

where $\exp\Big(tL_{\boldsymbol{x}_H}\Big)$ is the *Lie transformation*. If we introduce the operator $\Theta_t[G](\boldsymbol{u}) := G(\varphi_t(\boldsymbol{u}))$ = acting on differentiable functions, then we can write

$$G(\varphi_t(\boldsymbol{u})) \equiv \Theta_t[G](\boldsymbol{u}) = \exp\Big((t-t_0) L_{\boldsymbol{x}_H}\Big)[G](\boldsymbol{u}),$$

so that the solution of (7.6) is obtained by replacing $G(\boldsymbol{u})$ in (7.9) by the identity map $\mathrm{Id}(\boldsymbol{u}) \equiv \boldsymbol{u}$:

$$\varphi_t(\boldsymbol{u}_0) = \sum_{k\geq 0} \frac{(t-t_0)^k}{k!} (L^k_{\boldsymbol{x}_H} \mathrm{Id})(\boldsymbol{u}_0) = \exp\Big((t-t_0) L_{\boldsymbol{x}_H}\Big)[\mathrm{Id}](\boldsymbol{u}_0).$$

An important distinction between linear and non-linear cases is called *Vertauschungssatz* [52]. Given the flows $\varphi^{\{1\}}_{t_1}$ and $\varphi^{\{2\}}_{t_2}$ corresponding to the differential equations $\dot{\boldsymbol{u}} = f^{\{1\}}(\boldsymbol{u})$ and $\dot{\boldsymbol{u}} = f^{\{2\}}(\boldsymbol{u})$, respectively, they are applied as

$$\Big(\varphi^{\{2\}}_{t_2} \circ \varphi^{\{1\}}_{t_1}\Big)(\boldsymbol{u}_{k-1}) = \exp\Big(t_1 L_{f^{\{1\}}}\Big) \exp\Big(t_2 L_{f^{\{2\}}}\Big)[\mathrm{Id}](\boldsymbol{u}_{k-1}).$$

Note how the indices 1 and 2 are rearranged depending on whether one is dealing with maps or with exponentials of operators. This relation is generalised by induction to any number of flows [52].

In the non-autonomous problem one can still formally write the operator Θ_t associated with the exact flow as the Lie transformation corresponding to a function $\Omega(t;t_0)$,

$$\varphi_t(\boldsymbol{u}_0) = \Theta_t[\mathrm{Id}](\boldsymbol{u}_0) = \exp\Big(L_{\boldsymbol{x}_{\Omega(t;t_0)}}\Big)[\mathrm{Id}](\boldsymbol{u}_0), \tag{7.10}$$

which can be approximated for sufficiently small time intervals $[t_{k-1};\, t_k]$ by the corresponding Magnus expansion (ME) or its appropriate truncation [27, 89]. Similarly to the linear case, the ME is an infinite series

$$\Omega(t;t_0) = \sum_{m=1}^{\infty} \Omega_m(t;t_0), \tag{7.11}$$

but now it involves multiple integrals of nested Poisson brackets of scalar Hamiltonians:

$$\Omega_1(t;t_0) = \int_{t_0}^{t} H(s,\boldsymbol{q}_0,\boldsymbol{p}_0)\mathrm{d}s$$

$$\Omega_2(t;t_0) = -\frac{1}{2}\int_{t_0}^{t}\int_{t_0}^{s_1} \{H(s_1,\boldsymbol{q}_0,\boldsymbol{p}_0), H(s_2,\boldsymbol{q}_0,\boldsymbol{p}_0)\}\mathrm{d}s_2\mathrm{d}s_1.$$

Further terms are obtained by familiar generating procedures (4.9) [23].

7.2 MAGNUS-BASED METHODS FOR NON-LINEAR PROBLEMS

To obtain numerical methods, we need to approximate the operator Θ_t up to order p. In the general non-autonomous case (7.5), it is expressed as the composition of numerical flows:

$$\boldsymbol{u}_k = \Big(\psi_\tau^{\{s\}} \circ \psi_\tau^{\{s-1\}} \circ \cdots \circ \psi_\tau^{\{2\}} \circ \psi_\tau^{\{1\}}\Big)(\boldsymbol{u}_{k-1}),$$

where $\psi_\tau^{[j]}$ corresponds to the Lie transformation $\exp\big(\hat{B}_m\big) \coloneqq \exp\Big(L_{\boldsymbol{x}_{B_m}}\Big)$ that can be expressed as a function of $H_j \coloneqq H(t_{k-1} + c_j, \boldsymbol{q}_{k-1}, \boldsymbol{p}_{k-1})$ evaluated at quadrature nodes c_j. Therefore, taking into account the Vertauschungssatz, we can express one step of the method as $\boldsymbol{u}_k = \Psi_h^{[p]}[\mathrm{Id}]\boldsymbol{u}_{k-1}$, where

$$\Psi_h^{[p]} = \exp\big(x_1\hat{B}_1\big)\exp\big(x_2\hat{B}_2\big)\cdots\exp\big(x_{s-1}\hat{B}_{s-1}\big)\exp\big(x_s\hat{B}_s\big). \tag{7.12}$$

In order to approximate the non-linear ME, we adhere to the same strategy as in Chapter 4. We take the Lagrange interpolant $\widetilde{H}(t)$ of the time-dependent Hamiltonian H. As in the linear case, it approximates the solution at the end of the subinterval up to order p, that is,

$\boldsymbol{u}(t) - \tilde{\boldsymbol{u}}(t) = \mathcal{O}\left(\tau^{p+1}\right)$ [27]. In our specific case with time-variant mass (7.5), it is only necessary to interpolate $\mu(t)$:

$$\widetilde{H} = \frac{1}{2}\boldsymbol{p}^T\boldsymbol{p} - \tilde{\mu}(t)\frac{1}{r}.$$

As before, we define graded generators ζ_i through time derivatives of $\widetilde{H}$ at the centre of $[t_{k-1};\ t_k]$:

$$\zeta_l := \tau^l \frac{1}{(l-1)!} \left.\frac{\partial^{l-1}\widetilde{H}(t)}{\partial t^{l-1}}\right|_{t=t_{k-1}+\frac{\tau}{2}} = \mathcal{O}\left(\tau^l\right). \tag{7.13}$$

Four elements ζ_i, $i = 1,\dots,4$ will suffice to build schemes up to order 8, and the resulting algebra is formally the same that we used in Chapter 5 about the SE. With these elements, the truncation of the ME reads

$$\begin{aligned}\Omega^{[8]} = \zeta_1 &+ \frac{1}{12}\zeta_3 + \frac{1}{12}\{12\} - \frac{1}{240}\{23\} + \frac{1}{360}\{113\} - \frac{1}{240}\{212\} - \frac{1}{720}\{1112\}\\ &+ \frac{1}{80}\{14\} + \frac{1}{1344}\{34\} - \frac{1}{2240}\{124\} + \frac{1}{6720}\{223\} + \frac{1}{6048}\{313\} - \frac{1}{840}\{412\}\\ &- \frac{1}{6720}\{1114\} + \frac{1}{7560}\{1123\} - \frac{1}{4032}\{1312\} - \frac{11}{60480}\{2113\} + \frac{1}{6720}\{2212\}\\ &- \frac{1}{15120}\{11113\} - \frac{1}{30240}\{11212\} + \frac{1}{7560}\{21112\} + \frac{1}{30240}\{111112\}\end{aligned} \tag{7.14}$$

where $\{ij\dots k\ell\}$ represents the nested Poisson bracket $\{\zeta_i,\{\dots,\{\zeta_k,\zeta_\ell\}\dots\}\}$. Note how in comparison with (4.28) the signs in the expansion change.

The requirement that the scheme (7.12) provides an approximation to the exact solution up to a given order p implies that the coefficients x_i have to satisfy corresponding order conditions, which are obtained by reproducing the exact solution provided by (7.10) with (7.12) up to order p.

For example, if we take $\Omega^{[2]} = \zeta_1 = \tau H(t_{k-1} + \tau/2)$, then the method

$$\Psi^{[2]} = \exp\left(\hat{B}\right), \text{ with } B = \tau H\left(t_{k-1} + \frac{\tau}{2}\right),$$

agrees with the operator Θ_τ up to order $\mathcal{O}\left(\tau^2\right)$. On the other hand, the scheme in terms of the mapping $\Phi_K(\tau, \boldsymbol{u}, \mu)$ reads

$$x_k = \varphi_h^{[2]}(\boldsymbol{u}_{k-1}) = \Phi_K\left(\tau, \boldsymbol{u}_k, \mu\left(t_{k-1} + \frac{\tau}{2}\right)\right) \tag{7.15}$$

and is nothing but the 2nd-order midpoint rule.

To illustrate how the Magnus-based methods work in the non-linear case, we take the same 6th-order method $\Upsilon_2^{[6]}$ with brackets (commutators). With 3-point GL quadrature rule ζ_i have the familiar form

$$\begin{aligned}
\zeta_1 &= \tau H_2 = \tau(T+V_2) = \tau(T-\mu_2\frac{1}{r}),\\
\zeta_2 &= \frac{\sqrt{15}h}{3}(H_3-H_1) = -\tau\frac{\sqrt{15}}{3}\underbrace{(\mu_3-\mu_1)}_{:=\bar{\mu}_2}\frac{1}{r},\\
\zeta_3 &= \frac{10h}{3}(H_3-2H_2+H_1) = -\tau\frac{10}{3}\underbrace{(\mu_3-2\mu_2+\mu_1)}_{:=\bar{\mu}_3}\frac{1}{r}.
\end{aligned}$$

As in the SE, only ζ_1 depends on momenta through T, so $\{\zeta_2,\zeta_3\}=0$. Consequently, it is also possible to lower the number of order conditions needed to approximate $\exp\left(L_{\boldsymbol{x}_{\Omega^{[6]}}}\right)$. Triple Poisson brackets $\{i1j\}$ are easily expressed by means of averaged masses:

$$\{\zeta_i,\{\zeta_1,\zeta_j\}\} = -\bar{\mu}_i\bar{\mu}_j\frac{1}{r^4},\ i,j\geq 2. \tag{7.16}$$

Linear combinations of ζ_i give rise to a Hamiltonian function which corresponds to a scaled autonomous Kepler problem with a modified but constant mass $\bar{\mu}$, that is,

$$F = \sum_i x_i\zeta_i = x_1\tau\left(\frac{1}{2}\boldsymbol{p}^T\boldsymbol{p}-\bar{\mu}\frac{1}{r}\right),\ \text{with } x_1\neq 0.$$

Its flow can be determined with the mapping Φ_K given in (7.3). If the simple nested brackets (7.16) are introduced, then the flow of

$$G = \sum_{i\geq 2} x_i\zeta_i + v\{i1j\} = -\tau\bar{\mu}\frac{1}{r} - \tau^3 v\bar{\mu}_i\bar{\mu}_j\frac{1}{r^4}$$

is also easy to obtain, since it only depends on coordinates. In consequence, the composition

$$\begin{aligned}
\Upsilon_2^{[6]} = &\exp\left(-x_{1,2}\hat{\zeta}_2 + x_{1,3}\hat{\zeta}_3 + v_{1,212}[\hat{\zeta}_2,[\hat{\zeta}_1,\hat{\zeta}_2]]\right)\\
&\times\exp\left(x_{2,1}\hat{\zeta}_1 - x_{2,2}\hat{\zeta}_2 + x_{2,3}\hat{\zeta}_3\right)\\
&\times\exp\left(x_{2,1}\hat{\zeta}_1 + x_{2,2}\hat{\zeta}_2 + x_{2,3}\hat{\zeta}_3\right)\\
&\times\exp\left(x_{1,2}\hat{\zeta}_2 + x_{1,3}\hat{\zeta}_3 + v_{1,212}[\hat{\zeta}_2,[\hat{\zeta}_1,\hat{\zeta}_2]]\right)
\end{aligned} \tag{7.17}$$

approximates $\exp\left(L_{\boldsymbol{x}_{\Omega^{[6]}}}\right)$ up to order 6. Here $[\hat{\zeta}_2,[\hat{\zeta}_1,\hat{\zeta}_2]]$ is the Lie bracket corresponding to the Hamiltonian vector field of the function $\{\zeta_2,\zeta_1,\zeta_2\}$. The solution for the coefficients is the same as for the SE:

$$x_{1,2} = -\frac{1}{60},\ x_{1,3} = \frac{1}{60},\ x_{2,1} = \frac{1}{2},\ x_{2,2} = -\frac{2}{15},\ x_{2,3} = \frac{1}{40}.\ v_{1,212} = \frac{1}{43200}.$$

Alternatively, the scheme can be expressed compactly in terms of linear combinations:

$$\begin{aligned}\Upsilon_2^{[6]} =& \exp\Big(\tau(\bar{V}_1 + \tau^2 \hat{C}_{212})\Big)\exp\Big(a_2\tau(\hat{T} + \frac{1}{a_2}\bar{V}_2)\Big)\\ &\times\exp\Big(a_3\tau(\hat{T} + \frac{1}{a_3}\bar{V}_3)\Big)\exp\Big(\tau(\bar{V}_4 + \tau^2 \hat{C}_{212})\Big),\end{aligned} \tag{7.18}$$

with

$$\bar{V}_i = \sum_{j=1}^{3} a_{i,j}\hat{V}_j, \quad \hat{C}_{212} = -v_{1,212}\frac{5}{3}(\mu_3 - \mu_1)^2\frac{1}{r^4},$$

where the coefficients

$$\boldsymbol{A} = \begin{bmatrix} \frac{10+\sqrt{15}}{180} & -\frac{1}{9} & \frac{10-\sqrt{15}}{180} \\ \frac{15+8\sqrt{15}}{190} & \frac{1}{3} & \frac{15-8\sqrt{15}}{180} \\ a_{2,3} & a_{2,2} & a_{2,1} \\ a_{1,3} & a_{1,2} & a_{1,1} \end{bmatrix}; \qquad \begin{aligned} a_2 &= \sum_{j=1}^{3} a_{2,j} = \frac{1}{2}, \\ a_3 &= \sum_{j=1}^{3} a_{3,j} = \frac{1}{2}, \end{aligned} \tag{7.19}$$

are obtained with (4.21) and (4.22) as $\boldsymbol{A} = \boldsymbol{X}\boldsymbol{S}$.

In this method the most computationally expensive part in the mapping Φ_K, which is computed by the iterative procedure (7.19). It also possesses the FSAL property: $r := \|\boldsymbol{q}\|$ from the fourth stage can be directly used in the first and the second stages during the following time step. Consequently, the relative cost of the outermost flows can be neglected, and it is reasonable to measure the overall cost of this family of methods as the number of Φ_K calls.

7.3 NUMERICAL EXAMPLES

In this section we compare the new time-symmetric symplectic integrators with other well-established methods. However, we do not aim to test all possible scenarios, but just to illustrate how different schemes perform in one particular example.

The first competing family will be symmetric composition methods. If we take the 2nd-order midpoint rule (7.15) as a basic integrator, then we can obtain higher-order methods as

$$\mathrm{SS}_m^{[p]} \equiv \psi_{\zeta_m h}^{[2]} \circ \cdots \circ \psi_{\zeta_2 h}^{[2]} \circ \psi_{\zeta_1 h}^{[2]}, \tag{7.20}$$

whose coefficients for various orders are given on page 16; these are the methods $\mathrm{SS}_3^{[4]}$, $\mathrm{SS}_9^{[6]}$, and $\mathrm{SS}_{17}^{[8]}$. It is important to note that composition methods of order higher than two necessarily involve some negative coefficients, and therefore a backward integration in time at some inner

stages. Given a decreasing function $\mu(t)$, backward integration represents a non-physical effect of an increase in the mass.

Commutator-free exponential integrators $\mathrm{CF}_3^{[4opt]}$, $\mathrm{CF}_5^{[6]}$, and $\mathrm{CF}_{11}^{[8]}$ [5], originally proposed for the numerical integration of linear problems can also be adapted to the non-linear setting, eventually resulting in schemes of the form (7.12).

Finally, for the sake of completeness, we also include popular ODE solvers provided by MATLAB:

- the embedded variable-step Dormand–Prince $\mathrm{RK}^{[4(5)]}$ 4^{th}-order method (ode45);
- the variable-step and variable-order Adams–Bashforth–Moulton solver $\mathrm{AB}^{[13]}$, designed for problems with stringent error tolerances (ode113).

We count the computational cost of the Magnus-based methods as the number of Φ_K calls. For the adaptive methods, it is less obvious how to estimate the computational cost in order to compare them with the exponential algorithms. Obviously, we can count the number of the vector field (i. e. right-hand side) evaluations. However, fully assess the cost, we should know the overhead for estimating the local and global errors and to change either the time step or the order of the method, as well as the cost of evaluating the mass $\mu(t)$ (for some problems this could be the most costly part, depending on the model used). For these reasons, we will consider two evaluations of the vector field (and the cost to change the order/time step) be computationally equal to one call of Φ_K. Different choices for the relative cost would result in a small shift of the corresponding curves, although the overall conclusions should remain the same.

We illustrate the performance of the new schemes on the classical Eddington–Jeans law for the secular evolution of mass in binary systems [115],

$$\dot{\mu} = -\gamma \mu^{\delta}, \quad \mu(0) = \mu_0$$

or, equivalently,

$$\mu(t) = (\mu_0^{1-\delta} + \gamma(\delta - 1)t)^{\frac{1}{1-\delta}}. \tag{7.21}$$

Note that for non-integer values of δ the computational cost of evaluating $\mu(t)$ cannot be neglected. We consider the 2-dimensional case with $\delta = 1.4$, $\mu_0 = 1$, $\gamma = 10^{-2}$, initial conditions

$$\boldsymbol{q}_0 = (1-e, 0),\ \boldsymbol{p}_0 = \left(0, \sqrt{(1+e)/(1-e)}\right),$$

with $e = 0.2$ and $e = 0.8$ and final time $t_f = 20$. We numerically compute the reference solution $(\boldsymbol{q}(t_f), \boldsymbol{p}(t_f))$ to high accuracy and plot

the two-norm error of $(\boldsymbol{q},\boldsymbol{p})$ at the final time versus the computational cost (the number of Φ_K calls) for different time steps.

Figures below show the results obtained for methods of each order: 4^{th}-order in the top, 6^{th}-order in the second, 8^{th}-order in the third row. The graphs for the smaller value of the eccentricity are given in the left column. The best method of each order, along with the classic reference methods, ode45 and ode113 of Matlab, are summarized in Figure 7.4.

From the graphs we conclude that the symplectic methods have better general performance than the classical adaptive methods. Moreover, the adaptation of methods by the inclusion of cheap Poisson brackets leads to a notable performance gain compared to the general-purpose CF methods.

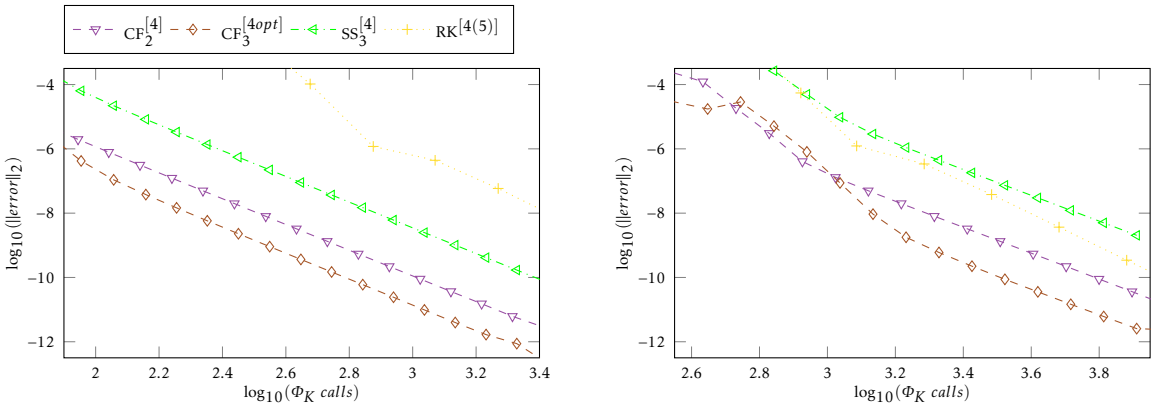
For this type of problems, among the new methods, $\Upsilon_2^{[6]}$ is a method of choice for low and medium accuracies, while $\Upsilon_{5b}^{[8]}$ should be chosen for high accuracies.

7.4 CONCLUSIONS

In this chapter we have considered the numerical integration of the two-body gravitational problem with a time-varying mass. The exact flow corresponds to a symplectic transformation, and different symplectic integrators from the literature can be adapted to solve this non-autonomous systems. However, none of these symplectic methods are designed to solve Hamiltonian systems with this particular structure. This is a relevant problem and new specifically designed symplectic integrators have been built. These new schemes can be seen as a generalization of the commutator-free quasi-Magnus exponential integrators and are based on compositions of symplectic flows corresponding to linear combinations of the Hamiltonian function and certain Poisson brackets. The implementation makes use of the mapping that solves the autonomous problem for averaged masses at each intermediate stage. In the autonomous case the schemes provide the exact solution, so they also show a high performance in the adiabatic limit.

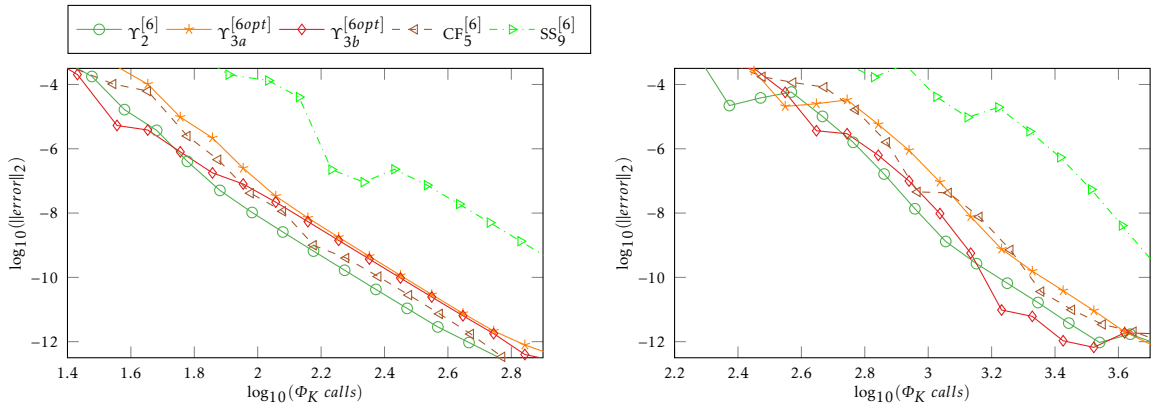
We have built time-symmetric methods of order four, six and eight that can be used with any quadrature rule of the order of the method or higher. Some proposed methods are optimized using a quadrature rule of higher order than the order of the method as well as by adding free parameters into the scheme in order to satisfy certain order conditions at higher orders.

The new methods have shown to be more efficient that other symplectic schemes to all desired accuracies on several numerical examples.



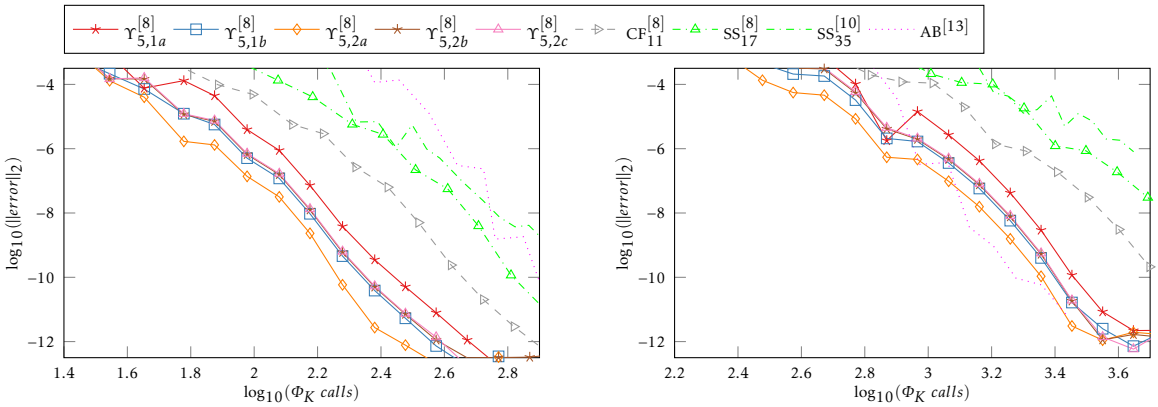
(a) $e = 1/5$ (b) $e = 4/5$

Figure 7.1: Efficiency graphs of the 4^{th}-order methods for the Kepler problem with mass defined by DE (7.21).



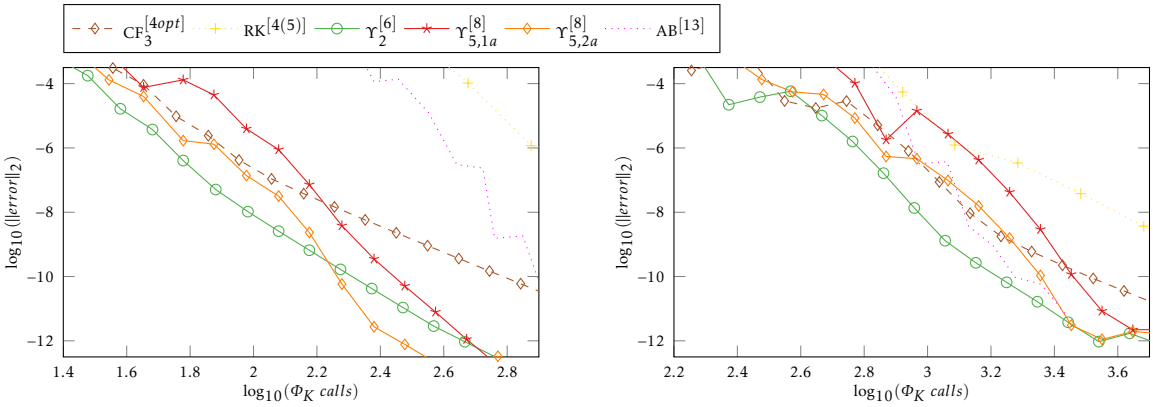
(a) $e = 1/5$ (b) $e = 4/5$

Figure 7.2: KPDE: efficiency graphs of the 6^{th}-order methods.



(a) $e = 1/5$ (b) $e = 4/5$

Figure 7.3: KPDE: efficiency graphs of the 8^{th}-order methods.



(a) $e = 1/5$ (b) $e = 4/5$

Figure 7.4: KPDE: comparison of the best methods of each order.

Algorithm 5: One step of the $\Upsilon_2^{[6]}$ methods for the Kepler problem with a time-variant mass

---

Data: initial time t_{k-1}, time step τ, initial value $(\boldsymbol{q}_{k-1}, \boldsymbol{p}_{k-1})^T$;

1 compute $\mu\big(t_{k-1} + c_j\tau\big)$ with $c_1 = \frac{1}{2} - \frac{\sqrt{15}}{10}$, $c_2 = \frac{1}{2}$, $c_3 = \frac{1}{2} + \frac{\sqrt{15}}{10}$;

2 compute linear combinations of $\bar{\mu}_j := \mu(t_{k-1} + c_j\tau)$ with coefficients (7.19): $\bar{\mu}_m = \sum_{j=1}^{3} a_{m,j}\mu_j$, $i = 1, \ldots, 4$;

3 assign temporary variables $\boldsymbol{q} := \boldsymbol{q}_{k-1}$, $\boldsymbol{p} := \boldsymbol{p}_{k-1}$, and $r := \|\boldsymbol{q}\|$;

4 compute the first stage:

$$\boldsymbol{p} = \boldsymbol{p} - \tau\bar{\mu}_1\frac{\boldsymbol{q}}{r^3} - \tau^3\frac{1}{6480}(\mu_3 - \mu_1)^2\frac{\boldsymbol{q}}{r^6};$$

5 propagate the solution in the second and the third stages with (7.4) :

$$(\boldsymbol{q}, \boldsymbol{p}) = \Phi_K(a_2\tau, \boldsymbol{q}, \boldsymbol{p}, \bar{\mu}_2/a_2)$$
$$(\boldsymbol{q}, \boldsymbol{p}) = \Phi_K(a_3\tau, \boldsymbol{q}, \boldsymbol{p}, \bar{\mu}_3/a_3)$$

6 compute $r = \|\boldsymbol{q}\|$ and the final stage

$$\boldsymbol{p} = \boldsymbol{p} - \tau\bar{\mu}_4\frac{\boldsymbol{q}}{r^3} - \tau^3\frac{1}{6480}(\mu_3 - \mu_1)^2\frac{\boldsymbol{q}}{r^6};$$

7 assign: $\boldsymbol{q}_k = \boldsymbol{q}$, $\boldsymbol{p}_k = \boldsymbol{p}$.

Result: value $(\boldsymbol{q}_k, \boldsymbol{p}_k)^T$.

---

8 CONCLUSION

In this chapter we summarise the obtained results and outline some possibilities for future research. We have build efficient numerical integrators for the following problems with explicitly time-dependent potentials: the Schrödinger equation (SE), the HEs, the wave equation (WE) and the KGFE, the non-linear Kepler problem with time-variant mass.

SCHRÖDINGER EQUATION

For the linear SE with time-dependent potentials, we have designed exponential quasi-commutator-free (QCF) methods based on the Magnus expansion (ME). Their efficiency comes from two sources. Firstly, if compared to high-performing, but more general commutator-free (CF) methods, the new ones exploit the structure of the problem. It allows using fewer stages thanks to the introduction of commutators which, in turn, are cheaply computed. Secondly, we approximate the matrix exponentials in the Krylov subspace. This approach significantly lowers the total cost of the methods in the case of large-dimensional problems.

As regards future work in this area, it seems possible to apply the same technique to tailoring methods for other problems with special structure which may allow additional optimisation, for instance, when the potential have a simpler structure or when the evolution is adiabatic. Additionally, the adaptation of geometric adaptive time-stepping procedures seems to be worth investigating.

WAVE EQUATION

For the wave equation (WE) we have build efficient Magnus–splitting schemes with low computational cost. These methods benefit from the incorporating of cheap commutators to some stages, which is also called the modified potentials approach.

Concerning the future work, it seems interesting to adapt the integrators to the problems, for example, from acoustics, in which the common, autonomous Laplacian is replaced with the derivatives of space- and time-dependent functions.

HILL EQUATION

For the Hill equation (HE), we have build both Magnus-based and splitting methods. For the numerical integrations of problems with highly oscillatory solutions, exponential integrators usually show a high performance. It remains important to analyse when one should turn to exponential integrators and this could depend either on the initial conditions as well as on the size of the spatial mesh or the external interactions.

KEPLER PROBLEM

In the last chapter we have show how the Magnus methods work in a non-linear setting, and as an illustration we have used the Kepler problem with decreasing mass.

We consider two direction of future work. Firstly, exploring the possibility to use the new methods for many-body problems, for instance, planetary systems with stars that lose mass. Secondly, building methods for other non-linear problems.

BIBLIOGRAPHY

1. ABRAHAM, R.; MARSDEN, J. E.; RATIU, T.; CUSHMAN, R. *Foundations of mechanics*. 2nd ed. Addison-Wesley Publishing Company, 1980. ISBN 080530102X.
2. ADAMS, F. C.; ANDERSON, K. R.; BLOCH, A. M. Evolution of planetary systems with time-dependent stellar mass-loss. *Monthly Notices of the Royal Astronomical Society*. 2013, vol. 432, no. 1, pp. 438–454. Available from DOI: 10.1093/mnras/stt479.
3. AGRACHEV, A. A.; SACHKOV, Y. L. *Control theory from the geometric viewpoint*. Springer Berlin Heidelberg, 2004. Available from DOI: 10.1007/978-3-662-06404-7.
4. ALONSO-MALLO, I.; CANO, B.; REGUERA, N. Avoiding order reduction when integrating linear initial boundary value problems with Lawson methods. *IMA Journal of Numerical Analysis*. 2016, pp. drw052. Available from DOI: 10.1093/imanum/drw052.
5. ALVERMANN, A.; FEHSKE, H. High-order commutator-free exponential time-propagation of driven quantum systems. *Journal of Computational Physics*. 2011, vol. 230, no. 15, pp. 5930–5956. Available from DOI: 10.1016/j.jcp.2011.04.006.
6. ANOSOV, D. V. Phase space. In: *Encyclopedia of Mathematics*. 2018.
7. ARCHIBALD, T.; FRASER, C.; GRATTAN-GUINNESS, I. The history of differential equations, 1670–1950. *Oberwolfach Reports*. 2004, pp. 2729–2794. Available from DOI: 10.4171/owr/2004/51.
8. ARNAL, A.; CASAS, F.; CHIRALT, C. A general formula for the Magnus expansion in terms of iterated integrals of right-nested commutators. *Journal of Physics Communications*. 2018, vol. 2, no. 3, pp. 035024. Available from DOI: 10.1088/2399-6528/aab291.
9. ARNOLD, V. I. *Mathematical methods of classical mechanics*. Springer New York, 1989. Available from DOI: 10.1007/978-1-4757-2063-1.
10. AUER, J.; KROTSCHECK, E.; CHIN, S. A. A fourth-order real-space algorithm for solving local Schrödinger equations. *The Journal of Chemical Physics*. 2001, vol. 115, no. 15, pp. 6841–6846. Available from DOI: 10.1063/1.1404142.
11. BADER, P. *Geometric integrators for Schrödinger equations*. 2014. PhD thesis. Universitat Politècnica de València.

12. BADER, P.; BLANES, S. Solving the pertubed quantum harmonic oscillator in imaginary time using splitting methods with complex coefficients. In: *Advances in Differential Equations and Applications*. Springer International Publishing, 2014, pp. 217–227. Available from DOI: 10.1007/978-3-319-06953-1_21.

13. BADER, P.; BLANES, S.; CASAS, F. *An improved algorithm to compute the exponential of a matrix*. 2017. Available from arXiv: http://arxiv.org/abs/1710.10989v1 [math.NA].

14. BADER, P.; BLANES, S.; CASAS, F.; KOPYLOV, N. Novel symplectic integrators for the Klein–Gordon equation with space- and time-dependent mass. *Journal of Computational and Applied Mathematics*. 2019, vol. 350, pp. 130–138. Available from DOI: 10.1016/j.cam.2018.10.011.

15. BADER, P.; BLANES, S.; CASAS, F.; KOPYLOV, N.; PONSODA, E. Symplectic integrators for second-order linear non-autonomous equations. *Journal of Computational and Applied Mathematics*. 2018, vol. 330, pp. 909–919. Available from DOI: 10.1016/j.cam.2017.03.028.

16. BADER, P.; BLANES, S.; KOPYLOV, N. Exponential propagators for the Schrödinger equation with a time-dependent potential. *The Journal of Chemical Physics*. 2018, vol. 148, no. 24, pp. 244109. Available from DOI: 10.1063/1.5036838.

17. BADER, P.; BLANES, S.; PONSODA, E.; SEYDAOĞLU, M. Symplectic integrators for the matrix Hill equation. *Journal of Computational and Applied Mathematics*. 2017, vol. 316, pp. 47–59. Available from DOI: 10.1016/j.cam.2016.09.041.

18. BADER, P.; ISERLES, A.; KROPIELNICKA, K.; SINGH, P. Effective approximation for the semiclassical Schrödinger equation. *Foundations of Computational Mathematics*. 2014, vol. 14, no. 4, pp. 689–720. Available from DOI: 10.1007/s10208-013-9182-8.

19. BADER, P.; ISERLES, A.; KROPIELNICKA, K.; SINGH, P. Efficient methods for linear Schrödinger equation in the semiclassical regime with time-dependent potential. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Science*. 2016, vol. 472, no. 2193, pp. 20150733. Available from DOI: 10.1098/rspa.2015.0733.

20. BERNSTEIN, D. S. *Matrix mathematics. theory, facts, and formulas*. Princeton University Press, 2009.

21. BEZANSON, J.; EDELMAN, A.; KARPINSKI, S.; SHAH, V. B. Julia: a fresh approach to numerical computing. *SIAM Review*. 2017, vol. 59, no. 1, pp. 65–98. Available from DOI: 10.1137/141000671.

22. BLANES, S.; BUDD, C. J. Adaptive geometric integrators for Hamiltonian problems with approximate scale invariance. *SIAM Journal on Scientific Computing*. 2005, vol. 26, no. 4, pp. 1089–1113. Available from DOI: 10.1137/s1064827502416630.

23. BLANES, S.; CASAS, F.; OTEO, J. A.; ROS, J. The Magnus expansion and some of its applications. *Physics Reports*. 2009, vol. 470, no. 5-6, pp. 151–238. Available from DOI: 10.1016/j.physrep.2008.11.001.

24. BLANES, S.; CASAS, F.; ROS, J. Processing symplectic methods for near-integrable Hamiltonian systems. *Celestial Mechanics and Dynamical Astronomy*. 2000, vol. 77, no. 1, pp. 17–35. Available from DOI: 10.1023/A:1008311025472.

25. BLANES, S.; MOAN, P. C. Practical symplectic partitioned Runge–Kutta and Runge–Kutta–Nyström methods. *Journal of Computational and Applied Mathematics*. 2002, vol. 142, no. 2, pp. 313–330. Available from DOI: 10.1016/s0377-0427(01)00492-7.

26. BLANES, S.; MOAN, P. C. Fourth- and sixth-order commutator-free Magnus integrators for linear and non-linear dynamical systems. *Applied Numerical Mathematics*. 2006, vol. 56, no. 12, pp. 1519–1537. Available from DOI: 10.1016/j.apnum.2005.11.004.

27. BLANES, S. Time-average on the numerical integration of non-autonomous differential equations. *SIAM Journal on Numerical Analysis*. 2018, vol. 56, no. 4, pp. 2513–2536. Available from DOI: 10.1137/17m1156150.

28. BLANES, S.; CASAS, F. On the necessity of negative coefficients for operator splitting schemes of order higher than two. *Applied Numerical Mathematics*. 2005, vol. 54, no. 1, pp. 23–37. Available from DOI: 10.1016/j.apnum.2004.10.005.

29. BLANES, S.; CASAS, F. *A concise introduction to geometric numerical integration*. Chapman and Hall/CRC, 2016.

30. BLANES, S.; CASAS, F.; MURUA, A. Splitting methods for non-autonomous linear systems. *International Journal of Computer Mathematics*. 2007, vol. 84, no. 6, pp. 713–727. Available from DOI: 10.1080/00207160701458567.

31. BLANES, S.; CASAS, F.; MURUA, A. Splitting methods in the numerical integration of non-autonomous dynamical systems. *Revista de la Real Academia de Ciencias Exactas, Fisicas y Naturales. Serie A. Matematicas*. 2011, vol. 106, no. 1, pp. 49–66. Available from DOI: 10.1007/s13398-011-0024-8.

32. BLANES, S.; CASAS, F.; MURUA, A. Symplectic time-average propagators for the Schrödinger equation with a time-dependent Hamiltonian. *The Journal of Chemical Physics*. 2017, vol. 146, no. 11, pp. 114109. Available from DOI: 10.1063/1.4978410.

33. BLANES, S.; CASAS, F.; THALHAMMER, M. High-order commutator-free quasi-Magnus exponential integrators for non-autonomous linear evolution equations. *Computer Physics Communications*. 2017, vol. 220, pp. 243–262. Available from DOI: 10.1016/j.cpc.2017.07.016.

34. BLANES, S.; DIELE, F.; MARANGI, C.; RAGNI, S. Splitting and composition methods for explicit time dependence in separable dynamical systems. *Journal of Computational and Applied Mathematics*. 2010, vol. 235, no. 3, pp. 646–659. Available from DOI: 10.1016/j.cam.2010.06.018.

35. BÖHME, C.; REISSIG, M. A scale-invariant Klein–Gordon model with time-dependent potential. *Annali dell'universita di Ferrara*. 2012, vol. 58, no. 2, pp. 229–250. Available from DOI: 10.1007/s11565-012-0153-9.

36. BRUGNANO, L.; MAZZIA, F.; TRIGIANTE, D. Fifty years of stiffness. In: *Recent Advances in Computational and Applied Mathematics*. Springer Netherlands, 2011, pp. 1–21. Available from DOI: 10.1007/978-90-481-9981-5_1.

37. BUTCHER, J. C. Coefficients for the study of Runge–Kutta integration processes. *Journal of the Australian Mathematical Society*. 1963, vol. 3, no. 02, pp. 185. Available from DOI: 10.1017/s1446788700027932.

38. CANNON, J. T.; DOSTROVSKY, S. *The evolution of dynamics: vibration theory from 1687 to 1742*. Springer New York, 2011. ISBN 1461394635.

39. CASAS, F. Sufficient conditions for the convergence of the Magnus expansion. *Journal of Physics A: Mathematical and Theoretical*. 2007, vol. 40, no. 50, pp. 15001–15017. Available from DOI: 10.1088/1751-8113/40/50/006.

40. CASAS, F.; MURUA, A. An efficient algorithm for computing the Baker–Campbell–Hausdorff series and some of its applications. *Journal of Mathematical Physics*. 2009, vol. 50, no. 3, pp. 033513. Available from DOI: 10.1063/1.3078418.

41. CASTRO, A.; MARQUES, M. A. L.; RUBIO, A. Propagators for the time-dependent Kohn–Sham equations. *The Journal of Chemical Physics*. 2004, vol. 121, no. 8, pp. 3425–3433. Available from DOI: 10.1063/1.1774980.

42. CELLEDONI, E.; ISERLES, A. Approximating the exponential from a Lie algebra to a Lie group. *Mathematics of Computation*. 2000, vol. 69, no. 232, pp. 1457–1480.

43. CHIBRIKOV, E. S. A right normed basis for free Lie algebras and Lyndon–Shirshov words. *Journal of Algebra*. 2006, vol. 302, no. 2, pp. 593–612. Available from DOI: 10.1016/j.jalgebra.2006.03.036.

44. CHIN, S. A. Symplectic integrators from composite operator factorizations. *Physics Letters A*. 1997, vol. 226, no. 6, pp. 344–348. Available from DOI: 10.1016/s0375-9601(97)00003-0.

45. COOLEY, J. W.; TUKEY, J. W. An algorithm for the machine calculation of complex Fourier series. *Mathematics of Computation*. 1965, vol. 19, no. 90, pp. 297–297. Available from DOI: 10.1090/s0025-5718-1965-0178586-1.

46. CURTISS, C. F.; HIRSCHFELDER, J. O. Integration of stiff equations. *Proceedings of the National Academy of Sciences*. 1952, vol. 38, no. 3, pp. 235–243. Available from DOI: 10.1073/pnas.38.3.235.

47. DANBY, J. M. A. *Fundamentals of Celestial Mechanics: 2nd Revised and Enlarged Edition*. 2nd ed. Willmann-Bell, Inc., 1988. ISBN 978-0943396200.

48. DRAGT, A. J. *Lie methods for nonlinear dynamics with applications to accelerator physics*. 2018.

49. DREWSEN, M.; BRØNER, A. Harmonic linear Paul trap: Stability diagram and effective potentials. *Physical Review A*. 2000, vol. 62, no. 4. Available from DOI: 10.1103/physreva.62.045401.

50. EINKEMMER, L.; OSTERMANN, A. Overcoming order reduction in diffusion-reaction splitting. part 1: Dirichlet boundary conditions. *SIAM Journal on Scientific Computing*. 2015, vol. 37, no. 3, pp. A1577–A1592. ISSN 1064-8275. Available from DOI: 10.1137/140994204.

51. FARRÉS, A.; LASKAR, J.; BLANES, S.; CASAS, F.; MAKAZAGA, J.; MURUA, A. High precision symplectic integrators for the Solar system. *Celestial Mechanics and Dynamical Astronomy*. 2013, vol. 116, no. 2, pp. 141–174. Available from DOI: 10.1007/s10569-013-9479-6.

52. HAIRER, E.; LUBICH, C.; WANNER, G. *Geometric numerical integration: structure-preserving algorithms for ordinary differential equations*. Springer Verlag, 2006.

53. HAIRER, E.; SÖDERLIND, G. Explicit, time reversible, adaptive step size control. *SIAM Journal on Scientific Computing*. 2005, vol. 26, no. 6, pp. 1838–1851. Available from DOI: 10.1137/040606995.

54. HAIRER, E.; WANNER, G. *Solving ordinary differential equations II: Stiff and differential-algebraic problems*. Springer Berlin Heidelberg, 1996. Available from DOI: 10.1007/978-3-642-05221-7.

55. HAIRER, E.; WANNER, G.; NØRSETT, S. P. *Solving ordinary differential equations I: Nonstiff problems*. Springer Berlin Heidelberg, 1993. Available from DOI: 10.1007/978-3-540-78862-1.

56. HOCHBRUCK, M.; OSTERMANN, A. Exponential integrators. *Acta Numerica*. 2010, vol. 19, pp. 209–286. Available from DOI: 10.1017/s0962492910000048.

57. HOFSTÄTTER, H.; KOCH, O. *Non-satisfiability of a positivity condition for commutator-free exponential integrators of order higher than four*. 2017. Available from arXiv: http://arxiv.org/abs/1709.08473v1 [math.NA].

58. HUNDSDORFER, W.; VERWER, J. *Numerical solution of time-dependent advection-diffusion-reaction equations*. Springer Berlin Heidelberg, 2003. Springer Series in Computational Mathematics. ISBN 3-540-03440-4. Available from DOI: 10.1007/978-3-662-09017-6.

59. ISERLES, A.; NØRSETT, S. P. On the solution of linear differential equations in Lie groups. *Phil. Trans. R. Soc. A*. 1999, vol. 357, no. 1754, pp. 983–1019. Available from DOI: 10.1098/rsta.1999.0362.

60. ISERLES, A.; QUISPEL, G. R. W. Why geometric integration? 2016. Available from arXiv: http://arxiv.org/abs/1602.07755v1 [math.NA].

61. ISERLES, A. *A first course in the numerical analysis of differential equations*. Cambridge University Press, 2008. Available from DOI: 10.1017/CBO9780511995569.

62. ISERLES, A.; MUNTHE-KAAS, H. Z.; NØRSETT, S. P.; ZANNA, A. Lie-group methods. *Acta Numerica*. 2000, vol. 9. Available from DOI: 10.1017/s0962492900002154.

63. JOHNSON, S. G. *Notes on FFT-based differentiation*. 2011.

64. JONES, E.; OLIPHANT, T.; PETERSON, P. et al. *SciPy: open source scientific tools for Python*. 2018.

65. KAHAN, W.; LI, R.-C. Composition constants for raising the orders of unconventional schemes for ordinary differential equations. *Mathematics of Computation*. 1997, vol. 66, no. 219, pp. 1089–1100. Available from DOI: 10.1090/s0025-5718-97-00873-9.

66. KATZ, V. J. *A history of mathematics*. 3rd ed. Pearson/Addison-Wesley, 2008. ISBN 0-321-38700-7.

67. KLARSFELD, S.; OTEO, J. A. Recursive generation of higher-order terms in the Magnus expansion. *Physical Review A*. 1989, vol. 39, no. 7, pp. 3270–3273. Available from DOI: 10.1103/physreva.39.3270.

68. KOMLENKO, Y. V. Fundamental matrix. In: *Encyclopedia of Mathematics*. 2018.

69. LASKAR, J. Analytical framework in Poincare variables for the motion of the solar system. In: *Predictability, Stability, and Chaos in N-Body Dynamical Systems*. Springer US, 1991, pp. 93–114. Available from DOI: 10.1007/978-1-4684-5997-5_7.

70. LASKAR, J.; ROBUTEL, P. High order symplectic integrators for perturbed Hamiltonian systems. *Celestial Mechanics and Dynamical Astronomy*. 2001, vol. 80, no. 1, pp. 39–62. Available from DOI: 10.1023/a:1012098603882.

71. LEIMKUHLER, B.; REICH, S. *Simulating Hamiltonian dynamics*. Cambridge University Press, 2005. ISBN 0521772907. Available from DOI: 10.1017/cbo9780511614118.

72. LUBICH, C. *From quantum to classical molecular dynamics: reduced models and numerical analysis*. European Mathematical Society, 2008. ISBN 978-3-03719-067-8.

73. MAGNUS, W.; WINKLER, S. *Hill's equation*. John Wiley and Sons, 1966.

74. MAGNUS, W. On the exponential solution of differential equations for a linear operator. *Communications on Pure and Applied Mathematics*. 1954, vol. 7, no. 4, pp. 649–673. Available from DOI: 10.1002/cpa.3160070404.

75. MAJOR, F. G.; WERTH, G.; GHEORGHE, V. N. *Charged particle traps*. Springer-Verlag, 2005. ISBN 978-3-540-26576-4. Available from DOI: 10.1007/b137836.

76. MATHWORKS. *Matlab manual. Expm*. 2018.

77. MATHWORKS. *Matlab manual. Fast Fourier transform*. 2018.

78. MCLACHLAN, N. W. *Theory and application of mathieu functions*. Dover, New York, 1964.

79. MCLACHLAN, R. I. On the numerical integration of ordinary differential equations by symmetric composition methods. *SIAM Journal on Scientific Computing*. 1995, vol. 16, no. 1, pp. 151–168. Available from DOI: 10.1137/0916010.

80. MCLACHLAN, R. I.; QUISPEL, G. R. W. Splitting methods. *Acta Numerica*. 2002, vol. 11. Available from DOI: 10.1017/s0962492902000053.

81. MOAN, P. C. *Efficient approximation of Sturm-Liouville problems using Lie-group methods*. 1998. Technical report.

82. MOAN, P. C.; NIESEN, J. Convergence of the Magnus series. *Foundations of Computational Mathematics*. 2007, vol. 8, no. 3, pp. 291–301. Available from DOI: 10.1007/s10208-007-9010-0.

83. AL-MOHY, A. H.; HIGHAM, N. J.; RELTON, S. D. New algorithms for computing the matrix sine and cosine separately or simultaneously. *SIAM Journal on Scientific Computing*. 2015, vol. 37, no. 1, pp. A456–A487. Available from DOI: 10.1137/140973979.

84. MOLER, C.; LOAN, C. V. Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later. *SIAM Review*. 2003, vol. 45, no. 1, pp. 3–49. Available from DOI: 10.1137/s00361445024180.

85. MUNTHE-KAAS, H.; OWREN, B. Computations in a free Lie algebra. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*. 1999, vol. 357, no. 1754, pp. 957–981. Available from DOI: 10.1098/rsta.1999.0361.

86. MURUA, A.; SANZ-SERNA, J. M. Order conditions for numerical integrators obtained by composing simpler integrators. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*. 1999, vol. 357, no. 1754, pp. 1079–1100. Available from DOI: 10.1098/rsta.1999.0365.

87. *NIST Handbook of Mathematical Functions Hardback and CD-ROM*. Cambridge University Press, 2010. ISBN 9780521192255.

88. OMELYAN, I. P.; MRYGLOD, I. M.; FOLK, R. Construction of high-order force-gradient algorithms for integration of motion in classical and quantum systems. *Physical Review E*. 2002, vol. 66, no. 2. Available from DOI: 10.1103/physreve.66.026701.

89. OTEO, J. A.; ROS, J. The Magnus expansion for classical Hamiltonian systems. *Journal of Physics A: Mathematical and General*. 1991, vol. 24, no. 24, pp. 5751–5762. Available from DOI: 10.1088/0305-4470/24/24/011.

90. PLATONOV, V. P. Lie group. In: *Encyclopedia of Mathematics*. 2019. ISBN 1402006098.

91. RACKAUCKAS, C.; NIE, Q. Differentialequations.jl – a performant and feature-rich ecosystem for solving differential equations in julia. *Journal of Open Research Software*. 2017, vol. 5. Available from DOI: 10.5334/jors.151.

92. RAHOMA, W. A. Investigating exoplanet orbital evolution around binary star systems with mass loss. *Journal of Astronomy and Space Sciences*. 2016, vol. 33, no. 4, pp. 257–264. Available from DOI: 10.5140/JASS.2016.33.4.257.

93. RAHOMA, W. A.; EL-SALAM, F. A. A.; AHMED, M. K. Analytical treatment of the two-body problem with slowly varying mass. *Journal of Astrophysics and Astronomy*. 2009, vol. 30, no. 3-4, pp. 187–205. Available from DOI: 10.1007/s12036-009-0012-y.

94. REUTENAUER, C. *Free Lie algebras*. Oxford University Press, 1993.

95. RICHARDS, J. A. *Analysis of periodically time-varying systems*. Springer Berlin Heidelberg, 1983. Available from DOI: 10.1007/978-3-642-81873-8.

96. SAAD, Y. Analysis of some Krylov subspace approximations to the matrix exponential operator. *SIAM Journal on Numerical Analysis*. 1992, vol. 29, no. 1, pp. 209–228. Available from DOI: 10.1137/0729014.

97. EL-SAFTAWY, M. I.; EL-SALAM, F. A. A. Second-order theory for the two-body problem with varying mass including periastron effect. *p*. 2017, vol. 88, no. 3, pp. 1723–1732. Available from DOI: 10.1007/s11071-017-3341-4.

98. SANZ-SERNA, J. M. Symplectic integrators for Hamiltonian problems: an overview. *Acta Numerica*. 1992, vol. 1, pp. 243. Available from DOI: 10.1017/s0962492900002282.

99. SANZ-SERNA, J. M.; CALVO, M. P. *Numerical Hamiltonian problems*. Springer, 1994. ISBN 0412542900.

100. SANZ-SERNA, J. M.; VERWER, J. G.; HUNDSDORFER, W. H. Convergence and order reduction of Runge–Kutta schemes applied to evolutionary problems in partial differential equations. *Numerische Mathematik*. 1986, vol. 50, no. 4, pp. 405–418. ISSN 0029-599X. Available from DOI: 10.1007/BF01396661.

101. SANZ-SERNA, J. M. Geometric integration. In: *The state of the art in numerical analysis*. OXFORD UNIV PR, 1997. ISBN 0198500149.

102. SHAMPINE, L. F.; REICHELT, M. W. The Matlab ODE suite. *SIAM Journal on Scientific Computing*. 1997, vol. 18, no. 1, pp. 1–22. Available from DOI: 10.1137/s1064827594276424.

103. SHENG, Q. Solving linear partial differential equations by exponential splitting. *IMA Journal of Numerical Analysis*. 1989, vol. 9, no. 2, pp. 199–212. Available from DOI: 10.1093/imanum/9.2.199.

104. SHIRSHOV, A. I. Subalgebras of free Lie algebras. *Mat. sbornik*. 1953, vol. 33, no. 75, pp. 441–452.

105. SÖDERLIND, G.; JAY, L.; CALVO, M. Stiffness 1952–2012: sixty years in search of a definition. *BIT Numerical Mathematics*. 2014, vol. 55, no. 2, pp. 531–558. Available from DOI: 10.1007/s10543-014-0503-3.

106. SOFRONIOU, M.; SPALETTA, G. Derivation of symmetric composition constants for symmetric integrators. *Optimization Methods and Software*. 2005, vol. 20, no. 4-5, pp. 597–613. Available from DOI: 10.1080/10556780500140664.

107. SPEISER, D. *Discovering the principles of mechanics 1600-1800: Essays by David Speiser*. Discovering the principles of mechanics 1600-1800. Ed. by WILLIAMS, K.; CAPARRINI, S. Birkhäuser, 2008. ISBN 978-3-7643-8564-4.

108. STRANG, G. Wavelets. *American Scientist*. 1994, vol. 82. Available from DOI: 10.2307/29775194.

109. SUZUKI, M. Fractal decomposition of exponential operators with applications to many-body theories and Monte Carlo simulations. *Physics Letters A*. 1990, vol. 146, no. 6, pp. 319–323. ISSN 0375-9601. Available from DOI: 10.1016/0375-9601(90)90962-N.

110. SUZUKI, M. General theory of fractal path integrals with applications to many-body theories and statistical physics. *Journal of Mathematical Physics*. 1991, vol. 32, no. 2, pp. 400–407. Available from DOI: 10.1063/1.529425.

111. THALHAMMER, M. A fourth-order commutator-free exponential integrator for nonautonomous differential equations. *SIAM Journal on Numerical Analysis*. 2006, vol. 44, no. 2, pp. 851–864. Available from DOI: 10.1137/05063042.

112. TREFETHEN, L. N. *Spectral methods in Matlab*. SIAM, 2000.

113. TURNER, K. L.; MILLER, S. A.; HARTWELL, P. G.; MACDONALD, N. C.; STROGATZ, S. H.; ADAMS, S. G. Five parametric resonances in a microelectromechanical system. *Nature*. 1998, vol. 396, no. 6707, pp. 149–152. Available from DOI: 10.1038/24122.

114. VARADARAJAN, V. S. *Lie groups, Lie algebras, and their representations*. Springer Verlag, 1984.

115. VERAS, D.; HADJIDEMETRIOU, J. D.; TOUT, C. A. An exoplanets response to anisotropic stellar mass loss during birth and death. *Monthly Notices of the Royal Astronomical Society*. 2013, vol. 435, no. 3, pp. 2416–2430. Available from DOI: 10.1093/mnras/stt1451.

116. VINBERG, E. B. Lie algebra, graded. In: *Encyclopedia of Mathematics*. 2018.

117. WALKER, R. B.; PRESTON, R. K. Quantum versus classical dynamics in the treatment of multiple photon excitation of the anharmonic oscillator. *The Journal of Chemical Physics*. 1977, vol. 67, no. 5, pp. 2017. Available from DOI: 10.1063/1.435085.

118. WIELEITNER, H. *History of mathematics from Decartes to the middle of the XIX century*. 2nd ed. Ed. by YUSHKEVICH, A. Nauka, 1966.

119. WISDOM, J.; HOLMAN, M. Symplectic maps for the n-body problem. *The Astronomical Journal*. 1991, vol. 102, pp. 1528. Available from DOI: 10.1086/115978.

120. YOSHIDA, H. Construction of higher order symplectic integrators. *Physics Letters A*. 1990, vol. 150, no. 5-7, pp. 262–268. Available from DOI: 10.1016/0375-9601(90)90092-3.

121. YUSHKEVICH, A. (ed.). *History of mathematics from the ancient times to the beginning of the XIX century*. Nauka, 1972.

122. ZHONG, G.; MARSDEN, J. E. Lie-Poisson Hamilton-Jacobi theory and Lie-Poisson integrators. *Physics Letters A*. 1988, vol. 133, no. 3, pp. 134–139. Available from DOI: 10.1016/0375-9601(88)90773-6.