Karlsruhe Institute of Technology

Efficient Time Integration Methods for Semilinear Damped Wave Equations

Zur Erlangung des akademischen Grades eines

DOKTORS DER NATURWISSENSCHAFTEN

von der KIT-Fakultät für Mathematik des
Karlsruher Instituts für Technologie (KIT)

genehmigte

DISSERTATION

von

Daniel Hans Eckhardt

Tag der mündlichen Prüfung: 22. Juli 2026

1. Referentin: Prof. Dr. Marlis Hochbruck
2. Referentin: Prof. Dr. Barbara Verfürth
3. Referent: Prof. Dr. Tobias Jahnke

KIT - Universität des Landes Baden-Württemberg und
nationales Forschungszentrum in der Helmholtz-Gemeinschaft

www.kit.edu

Acknowledgement

I gratefully acknowledge funding by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – Project-ID 258734477 – SFB 1173.

I continue in German.

Zuallererst möchte ich meiner Betreuerin Marlis Hochbruck danken, die bereits während meines Studiums durch ihre hervorragenden Vorlesungen meine Begeisterung für die Numerik geweckt hat. Sie hat mich ermutigt, eine Promotion zu beginnen, und mich ohne Zögern in ihre Arbeitsgruppe aufgenommen, als meine damalige Betreuerin das KIT verlassen hat. Während meiner gesamten Promotionszeit hat sie mich stets unterstützt, und ich bin ihr sehr dankbar für alles, was ich von ihr lernen konnte.

Barbara Verfürth möchte ich für die sehr gute Betreuung insbesondere in der Anfangszeit meiner Promotion sowie bereits während meiner Masterarbeit danken. Das hat mir den Start in die Promotion sehr erleichtert.

Zusätzlich möchte ich Tobias Jahnke für die (spontane) Bereitschaft danken, diese Arbeit zu begutachten, sowie für die vielen interessanten Gespräche und gemeinsamen Laufeinheiten.

Volker Grimm gilt mein besonderer Dank für die Unterstützung im zweiten Teil dieser Arbeit. Die Zusammenarbeit mit ihm hat mir sehr viel Spaß gemacht.

Ich möchte meiner gesamten Arbeitsgruppe sowie meinen ehemaligen Kommilitonen Lukas, Selina, Stefan und Tim für die gemeinsame Zeit danken, die wir miteinander verbringen durften, sowie insbesondere Benjamin, Julian, Malik, Tim und Carl für das Korrekturlesen dieser Arbeit. Ein besonderer Dank gilt auch unserer Laufgruppe kit@waves sowie meinem Zimmerkollegen Carl für die gemeinsamen Gym-Sessions. Der gemeinsame Sport war stets ein wertvoller Ausgleich, insbesondere in den stressigen Phasen der Promotion.

Zu guter Letzt möchte ich meinen Eltern und Großeltern, meinen Geschwistern und ganz besonders meiner Freundin Delia und ihrer Familie danken. Ihr habt mich immer unterstützt, motiviert und mich aufgebaut, wenn es nötig war.

Abstract

This thesis develops efficient time integration methods for semilinear wave equations with damping and multiscale features. In particular, we study two promising schemes that exploit the underlying structure of the equations: implicit–explicit (IMEX) schemes and exponential integrators.

We analyze an IMEX scheme for semilinear wave equations with strong damping, prove unconditional stability and derive error bounds for the fully discrete problem in a general spatial discretization framework. As an application, we provide convergence results for the heterogeneous multiscale method applied to wave equations with highly oscillatory coefficients.

In addition, we study exponential integrators for semilinear damped wave equations, which require the efficient approximation of matrix functions. We present a Chebyshev-based method for approximating these matrix functions, which relies on a Faber expansion over suitably chosen ellipses and includes both a residual error estimator and a novel, efficient strategy for selecting the ellipse parameters.

Numerical experiments demonstrate the accuracy and efficiency of the proposed methods.

Contents

1 Introduction **1**

2 Discretization of semilinear wave equations **7**

2.1 Prototypical example 7

2.2 Space discretization 8

2.3 Time integration 10

2.3.1 Explicit vs. implicit methods 11

2.3.2 Runge-Kutta methods 13

2.3.3 Rosenbrock-type methods 20

2.3.4 IMEX Runge-Kutta methods 22

2.3.5 Exponential integrators 26

2.3.6 Conclusion 29

2.4 The matrix φ_k-functions 30

2.4.1 Definition of matrix functions 31

2.4.2 Krylov subspace methods 32

2.4.3 Rational Krylov subspace methods 41

2.4.4 Conclusion 43

3 An IMEX scheme for semilinear wave equations **45**

3.1 Introduction 46

3.2 Time Integration 48

3.2.1 Wellposedness of the continuous problem 48

3.2.2 First-order formulation 49

3.2.3 The IMEX scheme 50

3.3 Full discretization . . . 51
3.3.1 Space discretization . . . 51
3.3.2 Error bounds for the full discretization . . . 54
3.4 Application: Semilinear wave equations with highly oscillatory coefficients 61
3.4.1 Notations . . . 61
3.4.2 Model problem and G-convergence . . . 61
3.4.3 Heterogeneous multiscale method . . . 64
3.4.4 Error bounds for the fully discrete FE-HMM . . . 66
3.5 Numerical Examples . . . 69
3.5.1 Implementation . . . 70
3.5.2 Setting . . . 70
3.5.3 Results . . . 71

4 Matrix function approximation using a Chebyshev-based method 73
4.1 Introduction . . . 74
4.2 Error bounds for truncated Faber/Chebyshev series . . . 75
4.3 Residual-based error estimator . . . 80
4.4 Efficient calculation of an optimal ellipse . . . 83
4.4.1 Construction of the best-fitting ellipse from a discrete complex set . 83
4.4.2 Arnoldi method . . . 89
4.4.3 Transformation of the spectrum . . . 89
4.4.4 Find best-fitting ellipse for eigenvalues of **S** . . . 91
4.5 Application for discrete semilinear damped wave equation . . . 94
4.5.1 Model problem . . . 94
4.5.2 Implementation details of Algorithms 2 and 3 . . . 95
4.5.3 Krylov subspace methods . . . 97
4.5.4 Numerical examples . . . 98
4.6 Useful identities . . . 102

Bibliography 104

CHAPTER 1

Introduction

Motivation

Wave equations represent foundational models within the field of physics, employed for the purpose of describing the propagation of diverse wave phenomena.

A particularly important class is given by *semilinear* wave equations, which include nonlinear terms involving lower-order derivatives. Such equations arise in a wide range of applications. In the area of nonlinear optics, semilinear wave equations serve to describe the propagation of light waves through complex materials, such as optical fibers. Similarly, in acoustics, they model sound waves traveling through different media, such as gases or fluids, especially under conditions of high sound pressure. In addition, many physical systems experience energy losses due to damping, which can be caused by mechanisms such as viscosity, resistance, or friction. This leads to the class of *semilinear damped wave equations*. In inhomogeneous materials, these equations may contain coefficients that oscillate strongly, reflecting the microscopic properties of the underlying medium. Such equations are referred to as *multiscale problems*, since they involve both microscopic and macroscopic features.

From a mathematical perspective, these equations are particularly challenging, since analytical solutions are generally not available. Consequently, numerical methods have been developed to compute approximations, enabling accurate simulation of the corresponding physical processes. These methods have in common that the solution is computed only at a finite number of discrete points in space and time, where temporal and spatial components are usually treated separately, in order to reduce computational effort

and memory requirements. This approach also accounts for the fundamental theoretical differences between these components.

This thesis primarily focuses on time integration methods while also providing a space discretization error analysis, as well as applications of the abstract results to multiscale problems for a class of semilinear damped wave equations. Numerical time integration of such equations is particularly difficult due to their stiffness and nonlinearity. An equation is referred to as stiff if explicit methods (such as the explicit Euler method) require extremely small time-step sizes to guarantee an accurate approximation. When using implicit methods (such as the implicit Euler method), on the other hand, nonlinear systems of equations have to be solved, which involves considerable computational effort. In the case of a semilinear wave equation, we can split the equation into a linear, stiff part, and a nonlinear, nonstiff part.

The overall goal of this thesis is to develop and analyze *efficient and tailored numerical methods* that treat each component of the equation according to its specific properties. In particular, we focus on two important approaches: implicit–explicit (IMEX) time integration schemes and exponential integrators.

Outline

The main contributions of this thesis are:

- An efficient IMEX method for semilinear damped wave equations together with a rigorous error analysis for the full discretization within the abstract framework of the unified error analysis, with application to multiscale problems with strongly oscillating coefficients

- A novel and efficient implementation of exponential integrators, focusing on the approximation of the arising matrix functions using Chebyshev polynomials, together with rigorous error bounds for the proposed method

The organization of this thesis is as follows.

First, in Chapter 2, we provide an overview and comparison of (one-step) time integration methods for discretized semilinear wave equations. We highlight that IMEX and exponential integrators are well suited for this type of equation. Since the efficiency of exponential integrators relies on the computation of matrix functions, we discuss advantages and disadvantages of different approaches in relation with the Chebyshev method.

In Chapter 3, we present the IMEX scheme, derive full discretization error bounds, and apply the abstract results to a multiscale problem. In particular, we employ the heterogeneous multiscale method (HMM) to tackle the numerical challenges arising from the multiscale structure and establish convergence rates in both space and time for the fully discrete problem. The content of this chapter is published in [52].

In Chapter 4, we study the Chebyshev method for approximating matrix functions. We derive error bounds for a specific class of matrix functions when the field of values of the matrix is contained in an ellipse. Furthermore, we present an algorithm for approximating this ellipse from suitable Ritz values and propose a residual-based error estimator for selecting the degree of the Chebyshev polynomial. Finally, we demonstrate the efficiency of the method by applying it to a matrix resulting from the spatial discretization of a semilinear damped wave equation. We compare the results with (rational) Krylov subspace methods and present numerical experiments to illustrate the competitiveness of the approach. The chapter contains the content of the preprint [51].

We emphasize that Chapter 3 and 4 aim to find efficient time integration methods for semilinear damped wave equations, but address different aspects of the problem. While Chapter 3 focuses on the analysis of the stability and convergence of the scheme considered there, Chapter 4 deals more with the practical implementation and efficient computation of the matrix functions required by exponential integrators.

Notation

In this thesis, $(\cdot,\cdot)$ denotes an inner product with which the vector space $\mathbb{C}^N$ has been equipped, and $\|\cdot\|$ denotes the corresponding norm. The induced matrix norm is designated by the same symbol. For vectors $\boldsymbol{v},\boldsymbol{w}\in\mathbb{C}^N$, the Euclidean dot product is defined by

$$\boldsymbol{v}\cdot\boldsymbol{w}\coloneqq\boldsymbol{v}^H\boldsymbol{w}=\overline{v_1}w_1+\cdots+\overline{v_N}w_N.$$

The identity matrix in $\mathbb{C}^{N\times N}$ is denoted by $\mathbf{I}=\mathbf{I}_N$. The Kronecker symbol (or Kronecker delta) is defined by $\delta_{ij}=1$ if $i=j$ and $\delta_{ij}=0$ otherwise.

Lebesgue and Sobolev spaces

Let $\Omega\subset\mathbb{R}^d, d\in\mathbb{N}$, be an open and bounded domain. We write $C(\Omega)$ for the space of continuous functions on Ω, and $C^k(\Omega)$, $k\in\mathbb{N}$, for the space of k-times continuously differentiable functions on Ω.

The classical Lebesgue spaces are given by $L^p(\Omega)$, where $1 \leq p \leq \infty$. The scalar product on $L^2(\Omega)$ is defined as $(\cdot,\cdot)_0$.

Let $\alpha = (\alpha_1, \ldots, \alpha_d) \in \mathbb{N}_0^d$ be a multi-index with

$$|\alpha| = \alpha_1 + \cdots + \alpha_d.$$

For a sufficiently smooth function $f : \Omega \to \mathbb{C}$, we define the (weak) partial derivative of order α by $\partial^\alpha f = \partial_1^{\alpha_1} \cdots \partial_d^{\alpha_d} f$. The Sobolev space of order $k \in \mathbb{N}_0$ is defined by

$$H^k(\Omega) = \left\{ f : \Omega \to \mathbb{C} \;\middle|\; \partial^\alpha f \in L^2(\Omega),\ |\alpha| \leq k \right\}. \tag{1.0.1}$$

This space is equipped with the scalar product

$$(f,g)_k \coloneqq (f,g)_{H^k(\Omega)} = \sum_{|\alpha| \leq k} \int_\Omega \overline{\partial^\alpha f(x)}\, \partial^\alpha g(x)\, \mathrm{d}x,$$

and the induced norm

$$\|f\|_k = \|f\|_{H^k(\Omega)} = (f,f)_k^{1/2}.$$

In particular, $H^0(\Omega) = L^2(\Omega)$. We denote by $H_0^k(\Omega)$ the subspace of $H^k(\Omega)$ consisting of functions that are zero on the boundary $\partial\Omega$ in the trace sense.

Operators

The gradient of a smooth function f is defined by

$$\nabla f(x) = \begin{pmatrix} \partial_1 f(x) \\ \vdots \\ \partial_d f(x) \end{pmatrix},$$

and the Laplacian by

$$\Delta f(x) = \nabla \cdot \nabla f(x) = \sum_{i=1}^d \partial_i^2 f(x),$$

for $x = (x_1, \ldots, x_d) \in \Omega$.

Banach space valued functions

For a Banach space X and $T > 0$, we denote by $C([0,T], X)$ the space of continuous functions $u : [0,T] \to X$. Moreover, for $k \in \mathbb{N}$, we denote by $C^k([0,T], X)$ the space of k-times continuously Fréchet differentiable functions from $[0,T]$ into X. For a function $u \in C^1([0,T], X)$ we denote the partial derivative with respect to the variable t by $\partial_t u = u' = u_t$.

For $1 \leq p < \infty$, we denote by $L^p([0,T];X)$ the Bochner space of measurable functions $u : [0,T] \to X$ equipped with the norm

$$\|u\|_{L^p([0,T],X)} = \left(\int_0^T \|u(t)\|_X^p \, \mathrm{d}t \right)^{1/p}.$$

CHAPTER 2

Discretization of semilinear wave equations

In this chapter, we provide a brief overview of methods for discretizing the semilinear damped wave equation and outline the key concepts and steps involved in its numerical discretization. We focus primarily on the time integration aspect, discussing several schemes available in the literature, and highlight the approaches considered in the following chapters, which offer advantageous properties for semilinear wave equations.

2.1 Prototypical example

We begin by considering a prototypical equation, namely a semilinear, strongly damped wave equation in a bounded Lipschitz domain $\Omega \subset \mathbb{R}^d, d = 1, 2, 3$, given by

$$\begin{aligned} \partial_{tt} u(t,x) - \Delta u(t,x) - \Delta \partial_t u(t,x) &= \mathcal{G}(u(t,x)), && x \in \Omega,\ t > 0, \\ u(0,x) = u_0(x), \qquad \partial_t u(0,x) &= v_0(x), && x \in \Omega, \end{aligned} \tag{2.1.1}$$

where $\mathcal{G} \in C^1(\mathbb{R})$ and $u(t,x) = 0$ on the boundary of Ω. Further, let

$$u_0, v_0 \in H_0^1(\Omega) \cap H^2(\Omega).$$

Prominent examples of $\mathcal{G}$ that are frequently discussed in the literature (see, e.g., [14, 62, 65, 107, 141]) for semilinear strongly damped nonlinear wave equations include:

$$\begin{aligned} &\mathcal{G}(s) = |s|^{\gamma-1} s, \quad \text{for } \gamma \in (1,5), \\ &\mathcal{G}(s) = \sin s. \end{aligned} \tag{2.1.2}$$

Such nonlinearities also arise in the context of nonlinear Klein–Gordon equations or sine-Gordon equations [35, 77], which play a significant role in many scientific applications such as nonlinear optics or quantum field theory, see, e.g., [58, 118] and references therein.

Under additional assumptions on $\mathcal{G}$, the existence of a (weak) solution is guaranteed.

Theorem 2.1.1 ([142]). *Let δ be*

$$\delta = \begin{cases} +\infty, & d = 1, 2, \\ 6, & d = 3. \end{cases}$$

If there exist positive constants c_1, c_2, c_3, c_4 and $p_1 \in (1, \delta - 1)$ and $p_2 \in (0, \delta - 2)$ such that

$$|\mathcal{G}(s)| \leq c_1 + c_2|s|^{p_1}, \qquad |\mathcal{G}'(s)| \leq c_3 + c_4|s|^{p_2}, \tag{2.1.3}$$

then there exists a unique (weak) solution

$$u \in C([0,T], H_0^1(\Omega)) \cap C^1([0,T], L^2(\Omega)) \cap C^2([0,T], H^{-1}(\Omega)), \quad u_t \in L^2([0,T]; H_0^1(\Omega)),$$

for (2.1.1) and some $T > 0$. This means that u satisfies the corresponding weak formulation:

$$\begin{gathered} (\partial_{tt} u(t), w)_0 + (\nabla u(t), \nabla w)_0 + (\nabla \partial_t u(t), \nabla w)_0 = (\mathcal{G}(u(t)), w)_0, \\ u(0) = u_0, \qquad \partial_t u(0) = v_0, \end{gathered} \tag{2.1.4}$$

for all $w \in H_0^1(\Omega)$ and a.e. $t \in [0,T]$.

For better readability, we omit the argument x in (2.1.4). It can be easily seen that the growth conditions in (2.1.3) are satisfied by the nonlinearities (2.1.2), and therefore a weak solution exists for these cases.

Remark 2.1.2. *A more general and abstract framework that covers a much wider range of equations is developed in Chapter 3.*

To discretize equation (2.1.1), we follow the method of lines (see, e.g., [75, Chapter I.1]), which first uses spatial discretization to transform the equation into a system of ordinary differential equations (ODEs), to which we later apply time integration schemes.

2.2 Space discretization

We consider a family of finite dimensional spaces $(V_h)_h$ with $\dim(V_h) = N_h < \infty$. For example, if we have a simplicial partition $\mathcal{T}_h$ of Ω, where we denote the elements of $\mathcal{T}_h$ by K, we can choose

$$V_h = \{v_h \in H_0^1(\Omega) \mid v_{h|_K} \in \mathbb{P}_p, \forall K \in \mathcal{T}_h\}.$$

Here $\mathbb{P}_p$ is the space of polynomials of maximal degree p, which is the space of conforming finite elements of order p. The spatial parameter h denotes the mesh size, i.e., the maximum diameter of all elements (see [28, 54] for more details).

We then seek a solution $u_h \in C^2([0,T], V_h)$ such that

$$\begin{aligned}(\partial_{tt} u_h(t), w_h)_0 + (\nabla u_h(t), \nabla w_h)_0 + (\nabla \partial_t u_h(t), \nabla w_h)_0 = (\mathcal{G}_h(u_h(t)), w_h)_0,\\ u_h(0) = u_{h,0}, \qquad \partial_t u(0) = v_{h,0},\end{aligned} \tag{2.2.1}$$

for all $w_h \in V_h$. Here, $u_{h,0}, v_{h,0} \in V_h$ and $\mathcal{G}_h$ are approximations to u_0, v_0 and $\mathcal{G}$. Error bounds for the spatial discretization can be derived using the unified error analysis framework proposed in Chapter 3.

For practical computations, we aim to derive a system of ODEs. For this purpose, we choose a basis $\Phi_1, \dots, \Phi_{N_h}$ of V_h such that

$$u_h(t,x) = \sum_{i=1}^{N_h} \mu_i(t)\Phi_i(x), \qquad \boldsymbol{\mu} = \begin{bmatrix} \mu_1 \\ \vdots \\ \mu_{N_h} \end{bmatrix}.$$

In conforming finite element methods, one usually chooses the basis functions Φ_i as *nodal basis functions*. These are defined with respect to a set of nodal points $\{a_1, \dots, a_{N_h}\}$ by the property

$$\Phi_i(a_j) = \delta_{ij}, \quad i,j = 1, \dots, N_h,$$

where δ_{ij} is the Kronecker delta. In this case, the coefficient $\mu_i(t)$ corresponds to the value of $u_h(t)$ at the nodal point a_i. Inserting this into (2.2.1), and using the fact that it suffices to test with the basis functions Φ_j, leads to

$$\sum_{i=1}^{N_h} \mu_i''(t)(\Phi_i, \Phi_j)_0 + \sum_{i=1}^{N_h} \mu_i(t)(\nabla\Phi_i, \nabla\Phi_j)_0 + \sum_{i=1}^{N_h} \mu_i'(t)(\nabla\Phi_i, \nabla\Phi_j)_0 = (\mathcal{G}_h(u_h(t)), \Phi_j)_0$$

for $j = 1, \dots, N_h$. Thus, we obtain the following matrices $\mathbf{M}, \mathbf{A} \in \mathbb{R}^{N_h \times N_h}$, which are symmetric and positive definite, and a nonlinear vector field $\mathbf{G}$, which is locally Lipschitz continuous, defined by

$$\begin{aligned}
\mathbf{M} &= (m_{ij})_{i,j=1}^{N_h}, & m_{ij} &= (\Phi_i, \Phi_j)_0, & &\text{(mass matrix)},\\
\mathbf{A} &= (a_{ij})_{i,j=1}^{N_h}, & a_{ij} &= (\nabla\Phi_i, \nabla\Phi_j)_0, & &\text{(stiffness matrix)},\\
\mathbf{G}(\boldsymbol{\mu}) &= (g_i(\boldsymbol{\mu}))_{i=1}^{N_h}, & g_i(\boldsymbol{\mu}) &= (\mathcal{G}_h(u_h), \Phi_i)_0, & &\text{(nonlinear load vector)}.
\end{aligned}$$

This leads to the following system of ODEs:

$$\begin{aligned} \mathbf{M}\boldsymbol{\mu}''(t) + \mathbf{A}\boldsymbol{\mu}(t) + \mathbf{A}\boldsymbol{\mu}'(t) &= \mathbf{G}(\boldsymbol{\mu}(t)), \\ \boldsymbol{\mu}(0) = \boldsymbol{\mu}_0, \quad \boldsymbol{\mu}'(0) &= \boldsymbol{\nu}_0, \end{aligned} \tag{2.2.2}$$

where $\boldsymbol{\mu}_0$ and $\boldsymbol{\nu}_0$ denote the coefficient vectors corresponding to the initial data $u_{h,0}$ and $v_{h,0}$, respectively. Since we later aim to apply a time integration scheme designed for first-order differential equations (in time), we reformulate (2.2.2) as follows: We set $\boldsymbol{\mu}' = \boldsymbol{\nu}$, and

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{\mu} \\ \boldsymbol{\nu} \end{bmatrix}, \quad \mathbf{S} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{M}^{-1}\mathbf{A} & -\mathbf{M}^{-1}\mathbf{A} \end{bmatrix}, \quad \mathbf{F}(\boldsymbol{x}) = \begin{bmatrix} \mathbf{0} \\ \mathbf{M}^{-1}\mathbf{G}(\boldsymbol{\mu}) \end{bmatrix}. \tag{2.2.3}$$

We end up with the following first-order formulation of (2.2.2):

$$\begin{aligned} \boldsymbol{x}'(t) - \mathbf{S}\boldsymbol{x}(t) &= \mathbf{F}(\boldsymbol{x}(t)), \quad t > 0, \\ \boldsymbol{x}(0) = \boldsymbol{x}_0 &= \begin{bmatrix} \boldsymbol{\mu}_0 \\ \boldsymbol{\nu}_0 \end{bmatrix}. \end{aligned} \tag{2.2.4}$$

Remark 2.2.1. *When using conforming finite elements, the mass matrix* $\mathbf{M}$ *is generally not a diagonal matrix. This makes the direct calculation of the inverse matrix computationally very intensive. In practice, multiplication of a vector by* $\mathbf{M}^{-1}$ *is performed by solving a system of linear equations. Iterative solvers such as the conjugate gradient method (CG) are often used for this purpose. Because the mass matrix has a condition number that is independent of the mesh size* h *(cf. [54, Proposition 28.6]) preconditioning is generally not required. However, if these matrix-vector multiplications need to be performed very frequently, preconditioning can be beneficial (see the discussion in Chapter 4).*

An alternative approach is to approximate the L^2*-inner product using an appropriate quadrature formula, when working with first-order polynomials, such that mass matrix* $\mathbf{M}$ *becomes diagonal. This approach is called mass lumping (cf. [38]). For higher orders, there are approaches to extend the conforming finite element domain* V_h *using so-called bubble functions (cf. [39, 64, 70]).*

Another option is to use discontinuous finite elements instead of conforming finite elements (see [43]), which results in a block-diagonal mass matrix, where the block size corresponds to the degrees of freedom of one element and is therefore easy to invert. However, this approach is not pursued further in this thesis.

2.3 Time integration

We now discuss methods for approximating solutions to the equation (2.2.4). For simplicity, we consider a general system of ODEs, where we seek a function $\boldsymbol{x} : [0, T] \to \mathbb{R}^N$

for some $T > 0$, such that

$$\begin{aligned} \boldsymbol{x}'(t) &= \boldsymbol{f}(t, \boldsymbol{x}(t)) \\ \boldsymbol{x}(0) &= \boldsymbol{x}_0, \end{aligned} \tag{2.3.1}$$

where $\boldsymbol{f} : \mathbb{R} \times \mathbb{R}^N \to \mathbb{R}^N, (t, x) \to \boldsymbol{f}(t, x)$ is nonlinear and locally Lipschitz continuous, and $\boldsymbol{x_0} \in \mathbb{R}^N$. We aim to approximate $\boldsymbol{x}$ at discrete time points $t_n = n\tau$ for a time-step size $\tau > 0$. This results in a set of approximations $\boldsymbol{x}_n \approx \boldsymbol{x}(t_n)$.

First, we will explain the difference between implicit and explicit methods. Furthermore, in Section 2.3.2.2, we discuss fully implicit Runge–Kutta methods, which allow large time-step sizes but are also among the most computationally intensive methods. More efficient are diagonal-implicit and related methods, which are discussed in Sections 2.3.2.3 and Section 2.3.3. Methods that utilize the structure of the discrete semilinear wave equation (2.2.4), in particular IMEX schemes and exponential integrators, are considered in Sections 2.3.4–2.3.5.

This section is based primarily on [75, 76] for Section 2.3.3-2.3.4 [12, 25, 26] for Section 2.3.4 and [86, 87] for Section 2.3.5.

2.3.1 Explicit vs. implicit methods

A time integration scheme is called *explicit*, if $\boldsymbol{x}_{n+1}$ can be calculated directly from known quantities as

$$\boldsymbol{x}_{n+1} = \Psi(t_n, \tau, \boldsymbol{x}_n, \cdots, \boldsymbol{x}_{n-k}),$$

for $n \geq k \geq 0$ and some given function Ψ. For $k = 0$, the method is called a one-step method, where for $k > 0$, the scheme is called a multi-step method. A well-known example is the explicit Euler scheme

$$\boldsymbol{x}_{n+1} = \boldsymbol{x}_n + \tau \boldsymbol{f}(t_n, \boldsymbol{x}_n).$$

The scheme is very simple, therefore often used, but suffers from stability issues. To explain these phenomena for the problem (2.2.4), we choose $\mathbf{S}$ as in (2.2.3) and consider, for the sake of simplicity, the case $\mathbf{F} = 0$, such that

$$\boldsymbol{f}(\boldsymbol{y}) = \mathbf{S}\boldsymbol{y}, \quad \boldsymbol{y} \in \mathbb{R}^N.$$

The exact solution of (2.3.1) is given by

$$\boldsymbol{x}(t) = e^{\mathbf{S}t}\boldsymbol{x}_0. \tag{2.3.2}$$

We further define $\tilde{\mathbf{A}} := \mathbf{M}^{-1}\mathbf{A}$. Then $\tilde{\mathbf{A}}$ is symmetric, positive definite matrix w.r.t. the scalar product

$$(\boldsymbol{v}, \boldsymbol{w})_{\mathbf{M}} = \boldsymbol{v}^T \mathbf{M} \boldsymbol{w}, \quad \boldsymbol{v}, \boldsymbol{w} \in \mathbb{R}^N,$$

and therefore diagonalizable. There exists an invertible matrix $\mathbf{T} \in \mathbb{R}^{N\times N}$ such that

$$\tilde{\mathbf{A}} = \mathbf{T}^{-1}\mathbf{D}\mathbf{T}, \quad \mathbf{D} = \begin{pmatrix} \mu_1 & 0 & \cdots & 0 \\ 0 & \mu_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mu_N \end{pmatrix},$$

where $\mu_i > 0$ are the eigenvalues of $\tilde{\mathbf{A}}$. Furthermore, $\tilde{\mathbf{A}}$ is similar to the symmetric positive definite matrix w.r.t the Euclidean inner product

$$\mathbf{B} = \mathbf{M}^{-1/2}\mathbf{A}\mathbf{M}^{-1/2}.$$

Consequently, since the $\mathbf{M}$-induced norm of $\tilde{\mathbf{A}}$ coincides with the spectral norm of $\mathbf{B}$, we get

$$\|\tilde{\mathbf{A}}\|_{\mathbf{M}} = \|\mathbf{B}\|_2 = \max_{i\in\{1,\dots,N\}} \mu_i. \tag{2.3.3}$$

Let λ be an eigenvalue of $\mathbf{S}$. By [130, Theorem 3], we obtain

$$\begin{aligned} 0 = \det \begin{pmatrix} -\lambda \mathrm{I}_N & \mathrm{I}_N \\ -\tilde{\mathbf{A}} & -\lambda \mathrm{I}_N - \tilde{\mathbf{A}} \end{pmatrix} &= \det\left(\lambda^2 \mathbf{I}_N + \tilde{\mathbf{A}}\lambda + \tilde{\mathbf{A}}\right) \\ &= \det\left(\lambda^2 \mathbf{I}_N + \mathbf{D}\lambda + \mathbf{D}\right) \\ &= \prod_{i=1}^{N} \left(\lambda^2 + \mu_i\lambda + \mu_i\right). \end{aligned}$$

The eigenvalues of the matrix $\mathbf{S}$ are then given by

$$\lambda_{1,2}^{(i)} = \frac{-\mu_i \pm \sqrt{\mu_i^2 - 4\mu_i}}{2}, \quad i = 1, \dots, N. \tag{2.3.4}$$

Now consider the case where $\mathbf{A}$ and $\mathbf{M}$ are the stiffness and mass matrices, respectively, for linear conforming finite elements on the interval $[0, 1]$ with a uniform mesh of width h. By (2.3.3) it holds that $\|\tilde{\mathbf{A}}\|_{\mathbf{M}} \sim h^{-2}$, see, e.g., [54, Example 28.2]. This implies that there exists an eigenvalue of $\tilde{\mathbf{A}}$ with $\mu_i \sim h^{-2}$, which yields that, for sufficiently small mesh sizes h, there exists an eigenvalue λ_i of $\mathbf{S}$ such that

$$\mathrm{Re}(\lambda_i) \ll 0. \tag{2.3.5}$$

If we assume that $\mathbf{S}$ is diagonalizable with a regular matrix $\mathbf{P} \in \mathbb{C}^{2N\times 2N}$, then the

system of equations (2.2.4) with $\mathbf{F} = 0$ decouples

$$\boldsymbol{u}'(t) = \begin{pmatrix} u_1'(t) \\ u_2'(t) \\ \vdots \\ u_N'(t) \end{pmatrix} = \begin{pmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_N \end{pmatrix} \begin{pmatrix} u_1(t) \\ u_2(t) \\ \vdots \\ u_N(t) \end{pmatrix},$$

where $\mathbf{P}\boldsymbol{x} = \boldsymbol{u}$. The resulting equation for the eigenvalue $\lambda = \lambda_i$ is given by

$$u'(t) = \lambda u(t). \tag{2.3.6}$$

The solution to the differential equation decays exponentially since $\mathrm{Re}\lambda < 0$. A good numerical method should reproduce this behavior as well or should at least stay bounded. However, this is only the case if we choose small time-step sizes when using explicit methods, such as the explicit Euler method (see [76, Chapter IV.2]). Differential equations for which this situation occurs are called *stiff*.

We will focus in the following on *implicit methods*, that is, methods where $\boldsymbol{x}_{n+1}$ is defined implicitly by an equation of the form

$$0 = \Psi(t_n, \tau, \boldsymbol{x}_{n+1}, \ldots, \boldsymbol{x}_{n-k}),$$

for $n \geq k \geq 0$ and some given function Ψ. Many implicit methods have significantly better stability properties and are therefore particularly well-suited for stiff differential equations. For this reason, our focus will be on these methods.

In particular, we will focus on *one-step methods*. Although multi-step methods have the advantage of requiring fewer function evaluations, they often require more memory. Furthermore, the maximum *order* of multistep methods for which the numerical solution remains bounded when applied to problems of the form (2.3.6), with

$$\mathrm{Re}(\lambda) \leq 0,$$

and arbitrary constant step size τ, is at most two. For more details we refer to [76, Chapter V.1] and [42].

In the following, we will study the class of *Runge-Kutta methods*.

2.3.2 Runge-Kutta methods

Runge–Kutta methods form a classical and widely used class of time integration schemes. A historical overview can be found in [31]. They are based on the idea of writing the

solution of the differential equation (2.3.1) in integral form by the fundamental theorem of calculus

$$\boldsymbol{x}(t_n+\tau) = \boldsymbol{x}(t_n) + \int_0^\tau \boldsymbol{f}(t_n+s, \boldsymbol{x}(t_n+s))\,\mathrm{d}s, \qquad t_n = n\tau, \quad n = 0,1,2,\dots .$$

Assume we have approximations

$$\boldsymbol{x}_n \approx \boldsymbol{x}(t_n), \qquad \boldsymbol{Y}_{n,i} \approx \boldsymbol{x}(t_n + c_i\tau), \quad i = 1,\dots,s.$$

Then the integral can be approximated using a suitable quadrature formula with weights $b_1,\dots,b_s$ and nodes $c_1,\dots,c_s$

$$\boldsymbol{x}_{n+1} = \boldsymbol{x}_n + \tau \sum_{i=1}^{s} b_i\, \boldsymbol{f}(t_n + c_i\tau, \boldsymbol{Y}_{n,i}).$$

The intermediate values $\boldsymbol{Y}_{n,i}$ are calculated by another quadrature formula with weights a_{ij} and the same nodes c_i:

$$\boldsymbol{x}(t_n + c_i\tau) \approx \boldsymbol{Y}_{n,i} = \boldsymbol{x}_n + \tau \sum_{j=1}^{s} a_{ij}\, \boldsymbol{f}(t_n + c_j\tau, \boldsymbol{Y}_{n,j}), \qquad i = 1,\dots,s.$$

In general, an s-stage Runge–Kutta method is defined as follows [75, Chapter II.]:

$$\begin{aligned} \boldsymbol{Y}_{n,i} &= \boldsymbol{x}_n + \tau \sum_{j=1}^{s} a_{ij}\, \boldsymbol{Y}'_{n,j}, \\ \boldsymbol{Y}'_{n,i} &= \boldsymbol{f}(t_n + c_j\tau, \boldsymbol{Y}_{n,i}), \\ \boldsymbol{x}_{n+1} &= \boldsymbol{x}_n + \tau \sum_{i=1}^{s} b_i\, \boldsymbol{Y}'_{n,i}, \end{aligned} \tag{2.3.7}$$

for $i = 1,\dots,s$. Equation (2.3.7) is compactly represented by the *Butcher tableau*:

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

or

$$\begin{array}{c|c} \boldsymbol{c} & \mathfrak{A} \\ \hline & \boldsymbol{b}^T \end{array} \qquad \text{with } \mathfrak{A} = (a_{ij})_{i,j=1}^{s}, \quad \boldsymbol{b} = (b_i)_{i=1}^{s}, \quad \boldsymbol{c} = (c_i)_{i=1}^{s}. \tag{2.3.8}$$

If $a_{ij} = 0$ for $j \geq i$, then the corresponding method is explicit, otherwise implicit. The explicit Euler method presented in the previous section is an explicit Runge-Kutta method. Its Butcher tableau is

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

2.3.2.1 Stability and convergence

The most important property for getting an accurate solution is the *order* of a Runge-Kutta method (cf. [75, Definition 2.1]).

Definition 2.3.1. A one-step time integration scheme is of (classical) *order* p if, for sufficiently smooth problems of the form (2.3.1), there exists a constant $C > 0$, independent of τ, such that

$$\|\boldsymbol{x}(t_0 + \tau) - \boldsymbol{x}_1\| \leq C\tau^{p+1},$$

where $\boldsymbol{x}_1$ is the approximation after one step with the method of size τ.

Order conditions, i.e., algebraic conditions on the coefficients a_{ij}, b_i, c_i that determine whether a scheme has order p, can be systematically derived using *labelled trees* (cf. [75, Chapter II.2]). For an explicit Runge-Kutta method, which has at least order $p \geq 1$, we obtain the following convergence result [75, Chapter II.3].

Theorem 2.3.2. *Let $\boldsymbol{f} \in C^p$. If an explicit Runge-Kutta method is of order p, then the global error satisfies*

$$\|\boldsymbol{x}(t_n) - \boldsymbol{x}_n\| \leq \Lambda\, \tau^p,$$

where the constant $\Lambda = \frac{C}{M}\left(\mathrm{e}^{MT} - 1\right)$ is independent of n and τ, and $t_n \in [0, T]$. Here, C and M are additional constants, where M depends on the Lipschitz constant of $\boldsymbol{f}$.

The constant Λ in Theorem 2.3.2 can grow rapidly when the Lipschitz constant of $\boldsymbol{f}$ becomes large. To investigate this further, we consider again the test problem (2.3.6). As explained in Section 2.3.1, the numerical solution (2.2.2) of this problem (for λ with $\mathrm{Re}(\lambda) \leq 0$) should remain bounded. We are particularly interested in methods for which this property holds without any restriction on the time-step size τ. Such methods are referred to as *A-stable*. Therefore, we apply a Runge-Kutta method to the test problem (2.3.6), which yields the following

$$\boldsymbol{x}_{n+1} = \boldsymbol{x}_n + z\boldsymbol{b}^T\boldsymbol{Y}_n, \tag{2.3.9a}$$

$$\boldsymbol{Y}_n = \boldsymbol{x}_n\mathbb{1} + z\,\mathfrak{A}\boldsymbol{Y}_n, \tag{2.3.9b}$$

with

$$\boldsymbol{Y}_n = (\boldsymbol{Y}_{n,i})_{i=1}^s, \quad z = \tau\lambda, \quad \mathbb{1} = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}.$$

Assuming that $\mathbf{I} - z\mathfrak{A}$ is invertible, it can be shown ([76, Proposition 3.2]) that (2.3.9a) can be rewritten as

$$\boldsymbol{x}_{n+1} = R(z)\boldsymbol{x}_n \quad \text{with} \quad R(z) = \frac{\det(\mathbf{I} - z\mathfrak{A} + z\mathbb{1}\boldsymbol{b}^T)}{\det(\mathbf{I} - z\mathfrak{A})}.$$

The function R is called the *stability function*. It is clear that the numerical solution stays bounded if $|R(z)| \leq 1$. Therefore, we define the stability region as

$$\mathcal{S} = \{z \in \mathbb{C} \mid (\mathbf{I} - z\mathfrak{A}) \text{ invertible and } |R(z)| \leq 1\}.$$

Accordingly, a numerical method is *A-stable* if

$$\mathbb{C}^- := \{z \in \mathbb{C} \mid \mathrm{Re}(z) \leq 0\} \subseteq S.$$

For an explicit method, it holds that $\det(\mathrm{I} - z\mathfrak{A}) = 1$. As a consequence, the stability regions of such methods are bounded, implying that explicit methods cannot be A-stable.

A-stable methods which ensure that very stiff components decay rapidly to zero are called *L-stable*.

Definition 2.3.3 (L-stability)**.** A numerical method is called *L-stable* if it is *A-stable* and, in addition, its stability function $R(z)$ satisfies

$$\lim_{z\to\infty} R(z) = 0.$$

Remark 2.3.4. *Another important concept is B-stability, particularly in the context of nonlinear right-hand sides. However, since we will assume that the nonlinear part of the right-hand side is not stiff (see Section 2.3.4), we only refer to [76, Chapter IV.12] for further details.*

2.3.2.2 Gauss and Radau methods

We aim to construct high-order A- and L-stable Runge–Kutta methods. One systematic approach is based on *collocation polynomials* [75, Chapter II.7]. A collocation polynomial p_n approximates the solution of an ODE by satisfying the *collocation conditions* at nodes $0 \leq c_1 < \cdots < c_s \leq 1$:

$$\begin{aligned} \boldsymbol{p}_n(t_n) &= \boldsymbol{x}_n, \\ \boldsymbol{p}_n'\big(t_n + c_i\tau\big) &= \boldsymbol{f}\big(t_n + c_i\tau,\, p_n(t_n + c_i\tau)\big), \quad i = 1, \ldots, s, \end{aligned}$$

where $\boldsymbol{x}_n$ is the approximation at time t_n. We compute the new approximation by

$$\boldsymbol{x}_{n+1} = \boldsymbol{p}_n(t_n + \tau).$$

The collocation polynomial is not computed explicitly. Instead, we use the fact that the collocation approach is equivalent to an implicit Runge-Kutta method with coefficients

$$a_{ij} = \int_0^{c_i} \ell_j(x)\,dx, \qquad b_j = \int_0^1 \ell_j(x)\,dx, \quad i,j = 1,\dots,s, \tag{2.3.10}$$

where

$$\ell_j(x) = \prod_{\substack{k=1\\k\neq j}}^{s} \frac{x - c_k}{c_j - c_k} \tag{2.3.11}$$

are the Lagrange interpolation polynomials at the collocation points $c_1,\dots,c_s$ (cf. [75, Chapter II.7 Theorem 7.7]). The order of the resulting Runge–Kutta method is the same as the order of the corresponding quadrature rule $(b_i, c_i)_{i=1,\dots s}$, see, e.g., in the proof of [75, Chapter II.7 Theorem 7.9]. Two important examples are presented below.

Example 2.3.5 (Gauss Methods)**.** Gauss methods are collocation methods derived from the nodes of Gaussian quadrature. The order of these methods is $2s$. Moreover, they are A-stable, but not L-stable. For $s = 1$, the Gauss method is the *implicit midpoint rule*:

$$\boldsymbol{x}_{n+1} = \boldsymbol{x}_n + \tau\, \boldsymbol{f}\Big(t_n + \frac{\tau}{2}, \frac{\boldsymbol{x}_n + \boldsymbol{x}_{n+1}}{2}\Big), \tag{2.3.12}$$

which can equivalently be expressed in a two-stage form:

$$\begin{aligned} \boldsymbol{x}_{n+\frac{1}{2}} &= \boldsymbol{x}_n + \frac{\tau}{2}\, \boldsymbol{f}\Big(t_n + \frac{\tau}{2}, \boldsymbol{x}_{n+\frac{1}{2}}\Big),\\ \boldsymbol{x}_{n+1} &= \boldsymbol{x}_n + \tau\, \boldsymbol{f}\Big(t_n + \frac{\tau}{2}, \boldsymbol{x}_{n+\frac{1}{2}}\Big). \end{aligned} \tag{2.3.13}$$

For $s = 2$, the Gauss method is defined by the following Butcher tableau:

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

Further details and examples can be found in [76, Chapters IV.3 and IV.5].

◇

Example 2.3.6 (Radau Methods)**.** Radau methods (or Radau IIA) are collocation methods of order $2s - 1$ with the last node at the right endpoint ($c_s = 1$). They are both

A-stable and L-stable. For $s = 1$, the Radau method coincides with the *implicit Euler scheme*. For $s = 3$, the fifth-order Radau method has the Butcher tableau:

$$\begin{array}{c|ccc} \frac{4-\sqrt{6}}{10} & \frac{88-7\sqrt{6}}{360} & \frac{296-169\sqrt{6}}{1800} & \frac{-2+3\sqrt{6}}{225} \\ \frac{4+\sqrt{6}}{10} & \frac{296+169\sqrt{6}}{1800} & \frac{88+7\sqrt{6}}{360} & \frac{-2-3\sqrt{6}}{225} \\ 1 & \frac{16-\sqrt{6}}{36} & \frac{3616+\sqrt{6}}{1296} & \frac{16+\sqrt{6}}{36} \\ \hline & \frac{16-\sqrt{6}}{36} & \frac{3616+\sqrt{6}}{1296} & \frac{16+\sqrt{6}}{36} \end{array} . \tag{2.3.14}$$

Again, details and more examples can be found in [76, Chapters IV.3 and IV.5].

◇

Remark 2.3.7. *For stiff problems, in particular those arising from the discretization of partial differential equations, Runge–Kutta methods often suffer from the phenomenon of order reduction, even when the methods are A-stable or L-stable (see, e.g., [76, Chapter IV.15]). Under additional assumptions, such as algebraic stability (see Definiton 12.5 in [76, Chapter IV.12]), one can nevertheless derive rigorous bounds for the global error in terms of the stage order $q \geq 0$, which is defined by the conditions*

$$\sum_{i=1}^{s} b_i c_i^{k-1} = \frac{1}{k}, \quad \sum_{j=1}^{s} a_{ij} c_j^{k-1} = \frac{c_i^k}{k}, \qquad k = 1, \dots, q. \tag{2.3.15}$$

In contrast to Theorem 2.3.2, the constants appearing in these error bounds are, for stiff problems, significantly smaller than the Lipschitz constant of $\boldsymbol{f}$. Gauss and Radau methods are algebraically stable and have stage order s. We refer to again to [76, Chapter IV.13-15] for the details.

We will now turn to the details of the implementation of (fully) implicit methods, as described in [76, Chapter IV.8]. A major drawback is their computational complexity, particularly when dealing with nonlinear right-hand sides. In each time-step, one has to solve a coupled nonlinear system of equations:

$$\boldsymbol{Z}_{n,i} = \tau \sum_{j=1}^{s} a_{ij}\, \boldsymbol{f}(t_n + c_j\tau, \boldsymbol{x}_n + \boldsymbol{Z}_{n,j}),$$

where we introduced the transformation

$$\boldsymbol{Z}_{n,i} = \boldsymbol{Y}_{n,i} - \boldsymbol{x}_n, \quad \boldsymbol{Z}_n = (\boldsymbol{Z}_{n,1}, \dots, \boldsymbol{Z}_{n,s}),$$

in order to reduce the influence of round-off errors. To solve this system, Newton's method is typically applied. In each Newton iteration, one has to solve a linear system with a block matrix of the form

$$\Big(\mathbf{I}_N - \tau\, a_{ij} \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{y}}(t_n + c_j\tau, \boldsymbol{x}_n + \boldsymbol{Z}_{n,j})\Big)_{i,j=1,\dots N}. \tag{2.3.16}$$

The computational cost can be significantly reduced by using the *simplified Newton method*. In this approach, the Jacobian matrices are all fixed at $(t_n, \boldsymbol{x}_n)$ and approximated by

$$\mathbf{J} \approx \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{y}}(t_n, \boldsymbol{x}_n).$$

If we apply this approximation to (2.3.16), we can write it as follows using the Kronecker product:

$$\mathbf{I}_{sN} - \tau\, \mathfrak{A} \otimes \mathbf{J}. \tag{2.3.17}$$

This particular structure can be exploited in practice to decouple the linear system. In particular, if $\mathfrak{A}^{-1}$ is diagonalizable, there exists an invertible matrix $\mathbf{T} \in \mathbb{R}^{s\times s}$ such that

$$\mathbf{T}^{-1}\mathfrak{A}^{-1}\mathbf{T} = \mathbf{\Lambda} = \operatorname{diag}(\lambda_1, \ldots, \lambda_s), \tag{2.3.18}$$

where λ_i are the eigenvalues of $\mathfrak{A}^{-1}$. Multiplying (2.3.17) from the left by $\tau^{-1}(\mathfrak{A}^{-1} \otimes \mathbf{I}_N)$, and transforming the resulting linear system using (2.3.18), we obtain an equivalent system whose coefficient matrix has a block-diagonal structure given by

$$\tau^{-1}\mathbf{\Lambda} \otimes \mathbf{I}_N - \mathbf{I}_s \otimes \mathbf{J} = \begin{pmatrix} \tau^{-1}\lambda_1\mathbf{I}_N - \mathbf{J} & 0 & \cdots & 0 \\ 0 & \tau^{-1}\lambda_2\mathbf{I}_N - \mathbf{J} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \tau^{-1}\lambda_s\mathbf{I}_N - \mathbf{J} \end{pmatrix}. \tag{2.3.19}$$

Thus, the original system decomposes into s independent linear systems, which can be solved separately. The transformation (2.3.18) only needs to be computed once, since $\mathfrak{A}^{-1}$ depends only on the chosen method.

One technical difficulty is that complex eigenvalues frequently occur (e.g., in the three-stage Radau method (2.3.14)), requiring complex arithmetic. If $\mathfrak{A}^{-1}$ is not diagonalizable, only partial decoupling can be achieved.

2.3.2.3 Diagonally implicit Runge–Kutta methods

We now consider methods that are less complex than fully implicit schemes, namely *Diagonally Implicit Runge–Kutta* (DIRK) methods. These methods are a subclass of

implicit Runge-Kutta methods where the coefficient matrix $\mathfrak{A}$ is a lower triangular matrix

$$\begin{array}{c|cccc} c_1 & a_{11} & 0 & \cdots & 0 \\ c_2 & a_{21} & a_{22} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ c_s & a_{s1} & a_{s2} & \cdots & a_{ss} \\ \hline & b_1 & b_2 & \cdots & b_s \end{array} .$$

The resulting nonlinear systems can be solved sequentially, starting from the first stage and proceeding to the last. On parallel architectures, the same techniques as in the previous section can be employed to decouple these systems, with the advantage that no complex eigenvalues of $\mathfrak{A}^{-1}$ arise. In the special case of *singly diagonally implicit Runge–Kutta* (SDIRK) methods, where $a_{ii} = \gamma$ for all $i = 1, \dots, s$ with $\gamma > 0$, the identical diagonal entries allow the reuse of a previously computed LU factorization of $\tau^{-1}\gamma^{-1}\mathbf{I}_N - \mathbf{J}$ in each stage, which further reduces the computational effort.

Moreover, SDIRK methods can be constructed to be A-stable and even L-stable, which makes them well suited for stiff problems. However, compared to fully implicit Runge–Kutta methods with the same number of stages, they generally achieve a lower order of accuracy. For A-stable SDIRK methods, the maximal order is $p \leq s + 1$, with $p = s + 1$ possible only for $s \leq 5$ and $s \neq 4$. For L-stable SDIRK methods, the maximal order is further restricted to $p \leq s$. The choice of γ plays a crucial role in the development of these methods.

When it comes to stiff problems, DIRK and SDIRK methods are not well suited, since they suffer heavily from order reduction due to their low stage order—two in the DIRK case and one in the SDIRK case ([76, Chapter IV.15]). In addition, algebraically stable DIRK methods have a highest possible order of four [73]. However, these methods form the basis for the development of efficient schemes, whose low order is offset by the fact that no nonlinear systems need to be solved.

We refer to [76, Chapter IV.6] for detailed discussions and concrete examples of DIRK methods.

2.3.3 Rosenbrock-type methods

To avoid the computational cost of solving nonlinear systems, one can apply a linearization technique to the implicit Runge-Kutta methods, which leads to the class of *Rosenbrock*

methods (cf. [76, Chapter IV.8]). For simplification, we look at an autonomous system

$$\boldsymbol{x}'(t) = \boldsymbol{f}(\boldsymbol{x}(t)), \quad 0 < t \leq T, \qquad \boldsymbol{x}(0) = \boldsymbol{x}_0. \tag{2.3.20}$$

Note that every non-autonomous system can be transformed into an autonomous system by adding the equation $t' = 1$ (see [76, p.103]). When applying Runge-Kutta methods to the autonomous systems, we assume the following simplifying assumption

$$c_i = \sum_{j=1}^{s} a_{ij}, \quad i = 1, \ldots, s.$$

The idea is to linearize the nonlinear system arising from a DIRK method, which results in a system of the form

$$\begin{aligned}
\boldsymbol{W}_{n,i} &= \boldsymbol{x}_n + \tau \sum_{j=1}^{i-1} a_{ij}\, \boldsymbol{Y}'_{n,j}, \\
\boldsymbol{Y}'_{n,i} &= \boldsymbol{f}(\boldsymbol{x}_n + \tau \sum_{j=1}^{i} a_{ij}\, \boldsymbol{Y}'_{n,j}) = \boldsymbol{f}(\boldsymbol{W}_{n,i} + \tau a_{ii}\, \boldsymbol{Y}'_{n,i}) \\
&\approx \boldsymbol{f}(\boldsymbol{W}_{n,i}) + \tau a_{ii} \boldsymbol{f}'(\boldsymbol{W}_{n,i}) \boldsymbol{Y}'_{n,i}.
\end{aligned}$$

As in the previous section, we would replace $\boldsymbol{f}'(\boldsymbol{W}_{n,i})$ by $\mathbf{J} = \boldsymbol{f}'(\boldsymbol{x}_n)$ to reduce the computational cost. A general s-stage Rosenbrock (or *Rosenbrock-Wanner*) (ROW) method is defined as

$$\begin{aligned}
\boldsymbol{Y}'_{n,i} &= \boldsymbol{f}(\boldsymbol{x}_n + \tau \sum_{j=1}^{i-1} a_{ij}\, \boldsymbol{Y}'_{n,j}) + \mathbf{J} \sum_{j=1}^{i} \gamma_{ij} \boldsymbol{Y}'_{n,j}, \quad i = 1, \ldots, s \\
\boldsymbol{x}_{n+1} &= \boldsymbol{x}_n + \tau \sum_{i=1}^{s} b_i \boldsymbol{Y}'_{n,i}.
\end{aligned} \tag{2.3.21}$$

These methods can be interpreted as DIRK methods, in which a single Newton iteration is applied to the resulting nonlinear system. As with DIRK methods, a common simplification is to set $\gamma_{ii} = \gamma$ for all $i = 1, \ldots, s$.

The method can then be described using the extended Butcher tableau

$$\begin{array}{c|c|c} \mathbf{c} & \mathfrak{A} & \mathbf{\Gamma} \\ \hline & \mathbf{b}^T & \end{array},$$

where $\mathfrak{A} = (a_{ij})_{i,j}$ and $\mathbf{\Gamma} = (\gamma_{ij})_{i,j}$ are lower-triangular matrices. If we apply (2.3.21) to the test problem (2.3.6), the method is equivalent to applying the s-stage SDIRK method with the Butcher tableu

$$\begin{array}{c|c} \mathbf{c} & \mathfrak{A} + \mathbf{\Gamma} \\ \hline & \mathbf{b}^T \end{array}$$

to the problem. Consequently, A- and L-stable Rosenbrock methods can also be derived. In this case, the order p satisfies $p \leq s+1$ and $p \leq s$ for A- and L-stable methods, respectively. Methods of order $p = 4$ are explicitly given in [76, Chapter IV.7]. In general, the stability for $p \geq s$ is uniquely determined by γ, just as for the SDIRK methods.

A disadvantage of ROW methods is that they require computing the Jacobian at each step. This motivated the idea of approximating the Jacobian instead of computing it exactly, which leads to the so-called W-methods.

2.3.3.1 W-methods

W-methods are defined as in (2.3.21), with the Jacobian $\mathbf{J}$ replaced by an approximation matrix $\widetilde{\mathbf{J}}$. This allows for the development of more efficient methods, for example by keeping the Jacobian constant over several time-steps. However, in contrast to ROW methods, the number of order conditions grows rapidly with the number of stages, making it more difficult to construct schemes of higher order (see, e.g., [76, Chapter IV.11]).

Remark 2.3.8. *In general, ROW and W-methods may partially lose nonlinear stability properties (such as B-stability) by the linearization. We do not pursue this issue further in this thesis and refer instead to [76, Chapter IV.15] and [133, Chapter 8.7] for more details. In addition, similar to DIRK methods, the order of convergence for stiff problems can only be two.*

2.3.4 IMEX Runge-Kutta methods

For the discrete semilinear damped wave equation (2.2.2), the corresponding first-order operator $\mathbf{S}$ (2.2.3) is stiff (see Section 2.3.1). This stiffness arises in particular from the fact that $\mathbf{S}$ involves the discretization of differential operators. On the other hand, $\mathbf{G}(\boldsymbol{\mu})$ is the discretization of a locally Lipschitz-continuous nonlinearity that does not depend on (spatial) derivatives of the solution. This motivates treating the nonlinear term $\mathbf{G}(\boldsymbol{\mu})$, and consequently also $\mathbf{F}(\boldsymbol{x})$, as *non-stiff*.

In the following, we consider numerical methods that exploit a decomposition of the differential equation into a stiff and a non-stiff part. In particular, we consider the autonomous system (2.3.20) with decomposition $\boldsymbol{f}(\boldsymbol{z}) = \mathbf{S}\boldsymbol{z} + \mathbf{F}(\boldsymbol{z})$,

$$\boldsymbol{x}'(t) = \mathbf{S}\boldsymbol{x}(t) + \mathbf{F}(\boldsymbol{x}(t)), \quad 0 < t \leq T, \qquad \boldsymbol{x}(0) = \boldsymbol{x}_0, \tag{2.3.22}$$

where $\mathbf{S} \in \mathbb{R}^{N\times N}$ and $\mathbf{F} : \mathbb{R}^N \to \mathbb{R}^N$ is a nonlinear, locally Lipschitz continuous function. The stiff component is treated implicitly, whereas the non-stiff component is treated explicitly. Methods of this type are known as implicit–explicit (IMEX) schemes [12, 26].

For both construction and analysis, it is advantageous if one finds implicit and explicit schemes that are closely related. Therefore, we revisit DIRK methods, which can be regarded as implicit schemes that are as close as possible to explicit ones. A general s-stage implicit–explicit Runge–Kutta (IMEX–RK) method applied to the autonomous system (2.3.22) is defined as follows:

$$\begin{aligned} \boldsymbol{Y}_{n,i} &= \boldsymbol{x}_n + \tau \sum_{j=1}^{i} a_{ij}\, \mathbf{S}\boldsymbol{Y}_{n,j} + \tau \sum_{j=1}^{i-1} \widetilde{a}_{ij}\, \mathbf{F}(\boldsymbol{Y}_{n,j}), \quad i = 1, \dots, s, \\ \boldsymbol{x}_{n+1} &= \boldsymbol{x}_n + \tau \sum_{i=1}^{s} b_i\, \mathbf{S}\boldsymbol{Y}_{n,i} + \tau \sum_{i=1}^{s} \widetilde{b}_i\, \mathbf{F}(\boldsymbol{Y}_{n,i}). \end{aligned} \tag{2.3.23}$$

For intermediate nodes $\boldsymbol{c} = (c_1, \dots, c_s)^T$ and $\widetilde{\boldsymbol{c}} = (\widetilde{c}_1, \dots, \widetilde{c}_s)^T$, we assume

$$c_i = \sum_{j=1}^{i} a_{ij}, \qquad \widetilde{c}_i = \sum_{j=1}^{i-1} \widetilde{a}_{ij}, \quad i = 1, \dots, s.$$

As for standard Runge–Kutta methods, the scheme can be written in compact form using the same notation as in (2.3.8):

$$\begin{array}{c|c} \boldsymbol{c} & \mathfrak{A} \\ \hline & \boldsymbol{b}^T \end{array}, \qquad \begin{array}{c|c} \widetilde{\boldsymbol{c}} & \widetilde{\mathfrak{A}} \\ \hline & \widetilde{\boldsymbol{b}}^T \end{array},$$

where $(\mathfrak{A}, \boldsymbol{c}, \boldsymbol{b})$ is a DIRK method and $(\widetilde{\mathfrak{A}}, \widetilde{\boldsymbol{c}}, \widetilde{\boldsymbol{b}})$ an explicit RK method.

2.3.4.1 Order conditions

The IMEX-RK schemes can be interpreted as W-methods (see Section 2.3.3.1) with $\widetilde{\mathbf{J}} = \mathbf{S}$. Therefore, the order conditions derived for W-methods can also be applied here. For IMEX-RK schemes, this leads to the standard order conditions for each scheme, along with additional coupling conditions. For first and second order, these are (cf. [26, Chapter 2.2]):

- **First order:**

$$\sum_i \widetilde{b}_i = 1, \qquad \sum_i b_i = 1. \tag{2.3.24}$$

- **Second order:**

$$\sum_i \widetilde{b}_i \widetilde{c}_i = \frac{1}{2}, \qquad \sum_i b_i c_i = \frac{1}{2}, \tag{2.3.25a}$$

$$\sum_i b_i \widetilde{c}_i = \frac{1}{2}, \qquad \sum_i \widetilde{b}_i c_i = \frac{1}{2}. \tag{2.3.25b}$$

For higher orders, the number of order conditions increases rapidly, which makes the use of simplifying assumptions necessary. We refer again to [26, Chapter 2.2] for details.

The simplest example of an IMEX scheme is the *IMEX Euler method*, with separate Butcher tableaus:

$$\begin{array}{c|c} 1 & 1 \\ \hline & 1 \end{array} \qquad \begin{array}{c|c} 0 & 0 \\ \hline & 1 \end{array}, \qquad \text{or equivalently,} \qquad \begin{array}{c|cc} 0 & 0 & 0 \\ 1 & 0 & 1 \\ \hline & 0 & 1 \end{array} \qquad \begin{array}{c|cc} 0 & 0 & 0 \\ 1 & 1 & 0 \\ \hline & 1 & 0 \end{array}.$$

This is an IMEX scheme of first order. It applies the implicit and explicit Euler methods to the corresponding terms.

A second example is the *IMEX midpoint rule*, which is of second order:

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

In particular, applying the implicit midpoint rule to autonomous system (2.3.20) gives

$$\boldsymbol{x}_{n+1} = \boldsymbol{x}_n + \tau \mathbf{f}\Big(\frac{\boldsymbol{x}_n + \boldsymbol{x}_{n+1}}{2}\Big).$$

This can be reformulated in a half-step formulation:

$$\begin{aligned} \boldsymbol{x}_{n+\frac{1}{2}} &= \boldsymbol{x}_n + \frac{\tau}{2}\mathbf{f}(\boldsymbol{x}_{n+\frac{1}{2}}), \\ \boldsymbol{x}_{n+1} &= \boldsymbol{x}_n + \tau \mathbf{f}(\boldsymbol{x}_{n+\frac{1}{2}}). \end{aligned}$$

Accordingly, the IMEX midpoint rule for (2.3.22) can be formulated as

$$\begin{aligned} \boldsymbol{x}_{n+\frac{1}{2}} &= \boldsymbol{x}_n + \frac{\tau}{2}\mathbf{S}\boldsymbol{x}_{n+\frac{1}{2}} + \frac{\tau}{2}\mathbf{F}(\boldsymbol{x}_n), \\ \boldsymbol{x}_{n+1} &= \boldsymbol{x}_n + \tau \mathbf{S}\boldsymbol{x}_{n+\frac{1}{2}} + \tau \mathbf{F}(\boldsymbol{x}_{n+\frac{1}{2}}). \end{aligned} \tag{2.3.26}$$

2.3.4.2 An IMEX scheme for second-order problems

In [83], an IMEX scheme for second-order problems is considered, based on applying the Crank–Nicolson method to (2.3.22):

$$\boldsymbol{x}_{n+1} = \boldsymbol{x}_n + \frac{\tau}{2}\Big(\mathbf{S}(\boldsymbol{x}_n + \boldsymbol{x}_{n+1}) + \mathbf{F}(\boldsymbol{x}_n) + \mathbf{F}(\boldsymbol{x}_{n+1})\Big).$$

We consider the following second-order problem:

$$\begin{aligned} \boldsymbol{\mu}''(t) + \mathbf{A}\boldsymbol{\mu}(t) + \mathbf{B}\boldsymbol{\mu}'(t) &= \mathbf{G}(\boldsymbol{\mu}(t)), \\ \boldsymbol{\mu}(0) = \boldsymbol{\mu}_0, \quad \boldsymbol{\mu}'(0) &= \boldsymbol{\nu}_0, \end{aligned} \tag{2.3.27}$$

where $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{N\times N}$, and $\mathbf{G} : \mathbb{R}^N \to \mathbb{R}^N$ is a nonlinear, locally Lipschitz continuous function. As discussed in Section 2.2, (2.3.27) can be reformulated as a first-order system of the form (2.3.22).

The Crank–Nicolson scheme can equivalently be written in a full-half-step formulation (see [83, Lemma 2.5]):

$$\begin{aligned}
\boldsymbol{\nu}^{n+\frac{1}{2}} &= \boldsymbol{\nu}^n - \frac{\tau}{2}\mathbf{A}\boldsymbol{\mu}^n - \frac{\tau^2}{4}\mathbf{A}\boldsymbol{\nu}^{n+\frac{1}{2}} - \frac{\tau}{2}\mathbf{B}\boldsymbol{\nu}^{n+\frac{1}{2}} + \frac{\tau}{4}\Big(\mathbf{G}(\boldsymbol{\mu}^n) + \mathbf{G}(\boldsymbol{\mu}^{n+1})\Big),\\
\boldsymbol{\mu}^{n+1} &= \boldsymbol{\mu}^n + \tau\boldsymbol{\nu}^{n+\frac{1}{2}},\\
\boldsymbol{\nu}^{n+1} &= \boldsymbol{\nu}^{n+\frac{1}{2}} - \frac{\tau}{2}\mathbf{A}\boldsymbol{\mu}^{n+1} + \frac{\tau^2}{4}\mathbf{A}\boldsymbol{\nu}^{n+\frac{1}{2}} - \frac{\tau}{2}\mathbf{B}\boldsymbol{\nu}^{n+\frac{1}{2}} + \frac{\tau}{4}\Big(\mathbf{G}(\boldsymbol{\mu}^n) + \mathbf{G}(\boldsymbol{\mu}^{n+1})\Big).
\end{aligned} \tag{2.3.28}$$

By replacing the trapezoidal rule for the nonlinear term with the left- and right-rectangular rule, the authors of [83] obtained the following IMEX scheme

$$\begin{aligned}
\boldsymbol{\nu}^{n+\frac{1}{2}} &= \boldsymbol{\nu}^n - \frac{\tau}{2}\mathbf{A}\boldsymbol{\mu}^n - \frac{\tau^2}{4}\mathbf{A}\boldsymbol{\nu}^{n+\frac{1}{2}} - \frac{\tau}{2}\mathbf{B}\boldsymbol{\nu}^{n+\frac{1}{2}} + \frac{\tau}{2}\mathbf{G}(\boldsymbol{\mu}^n),\\
\boldsymbol{\mu}^{n+1} &= \boldsymbol{\mu}^n + \tau\boldsymbol{\nu}^{n+\frac{1}{2}},\\
\boldsymbol{\nu}^{n+1} &= \boldsymbol{\nu}^{n+\frac{1}{2}} - \frac{\tau}{2}\mathbf{A}\boldsymbol{\mu}^{n+1} + \frac{\tau^2}{4}\mathbf{A}\boldsymbol{\nu}^{n+\frac{1}{2}} - \frac{\tau}{2}\mathbf{B}\boldsymbol{\nu}^{n+\frac{1}{2}} + \frac{\tau}{2}\mathbf{G}(\boldsymbol{\mu}^{n+1}).
\end{aligned} \tag{2.3.29}$$

This scheme can be interpreted as a combination of the Crank–Nicolson method for the linear part and the leap-frog method (see, e.g., [74]) for the nonlinear part. By exploiting the second-order structure, the scheme can be highly efficient (see discussion in [83]).

However, this scheme is tailored specifically the equation (2.3.27). If, for instance, $\mathbf{G}$ depends not only on $\boldsymbol{\mu}$ but also on $\boldsymbol{\mu}'$, the efficiency is lost. In that case, the scheme becomes essentially as expensive as a fully implicit Crank–Nicolson method, since the nonlinear system must be solved at each time-step.

2.3.4.3 Error analysis

As we will discuss the error analysis in more detail later, we only provide some references here. In Chapter 3, we consider the IMEX midpoint rule for the semilinear damped wave equation, prove second-order convergence, and derive error bounds for the fully discrete scheme, including the spatial discretization, within an abstract framework. Similar results can be found in [83] for the IMEX scheme (2.3.29).

For ODEs, regarding error analysis, we refer to [26, Theorems 2.2.10 and 2.2.13], which establish convergence with respect to the order p for a large class of IMEX schemes, where p denotes the order of the explicit part.

For stiff problems, the IMEX-RK methods suffer from similar order reduction effects as the DIRK and Rosenbrock methods if the classical order is greater than two [25].

2.3.5 Exponential integrators

We revisit the autonomous system (2.3.22). The main idea of exponential integrators is to treat the stiff part of the equation exactly. Consequently, the resulting scheme reproduces the stability properties of the exact solution associated with this part. Only the remaining, non-stiff part is then approximated.

For the construction, we use the *variation-of-constants formula*

$$\boldsymbol{x}(t) = \mathrm{e}^{t\mathbf{S}}\,\boldsymbol{x}_0 + \int_0^t \mathrm{e}^{(t-s)\mathbf{S}}\,\mathbf{F}(\boldsymbol{x}(s))\,\mathrm{d}s, \tag{2.3.30}$$

where

$$\mathrm{e}^{t\mathbf{S}} = \sum_{k=0}^{\infty} \frac{(t\mathbf{S})^k}{k!}$$

denotes the matrix exponential. In the following, we construct a particular class of exponential integrators based on (2.3.30), namely exponential Runge–Kutta methods, which are closely related to classical Runge–Kutta methods of Section 2.3.2.

As presented there, we consider $0 \le c_1 < \cdots < c_s \le 1$ to be nodes at which we approximate the solution.

We approximate the integral in (2.3.30) by a quadrature formula in which only $\mathbf{F}$ is approximated, while the matrix function is treated exactly.

$$\begin{aligned} \boldsymbol{Y}_{n,i} &= \mathrm{e}^{c_i\tau\mathbf{S}}\,\boldsymbol{x}_n + \tau \sum_{j=1}^{s} a_{ij}(\tau\mathbf{S})\,\mathbf{F}\big(\boldsymbol{Y}_{n,j}\big), \\ \boldsymbol{x}_{n+1} &= \mathrm{e}^{\tau\mathbf{S}}\,\boldsymbol{x}_n + \tau \sum_{i=1}^{s} b_i(\tau\mathbf{S})\,\mathbf{F}\big(\boldsymbol{Y}_{n,i}\big). \end{aligned} \tag{2.3.31}$$

The Runge–Kutta method with coefficients $a_{ij}(0)$, $b_i(0)$, and c_i is called the *underlying Runge–Kutta method*. An exponential Runge–Kutta method can be represented by the tableau

$$\begin{array}{c|c} \boldsymbol{c} & \mathfrak{A}(z) \\ \hline & \boldsymbol{b}(z) \end{array}$$

with

$$\mathfrak{A}(z) = (a_{ij}(z))_{i,j=1}^{s}, \quad \boldsymbol{b}(z) = (b_i(z))_{i=1}^{s}, \quad \boldsymbol{c} = (c_i)_{i=1}^{s},$$

where the coefficients are analytic functions evaluated at the linear operator $\tau\mathbf{S}$. Since we assume that the nonlinear part is non-stiff (see Section 2.3.4), we restrict ourselves to explicit methods, i.e., $a_{ij}(z) = 0$ for $i \le j$.

Further, the following simplifying assumptions should hold

$$\sum_{j=1}^{s} b_j(z) = \varphi_1(z), \qquad \sum_{j=1}^{s} a_{ij}(z) = c_i\,\varphi_1(c_i z), \quad i = 1, \dots, s,$$

where the function φ_1 is defined by

$$\varphi_1(z) = \int_0^1 \mathrm{e}^{(1-\sigma)z}\,\mathrm{d}\sigma = \frac{\mathrm{e}^z - 1}{z}.$$

For $z = 0$, the simplifying assumptions reduce to

$$\sum_{i=1}^{s} b_i(0) = 1, \qquad \sum_{j=1}^{s} a_{ij}(0) = c_i.$$

Hence, the underlying Runge–Kutta scheme automatically has order $p \geq 1$.

Example 2.3.9. We rewrite the variation-of-constants formula and approximate the integral as follows:

$$\begin{aligned} \boldsymbol{x}(t_{n+1}) &= \mathrm{e}^{\tau \mathbf{S}}\,\boldsymbol{x}(t_n) + \int_0^\tau \mathrm{e}^{(\tau-\sigma)\mathbf{S}}\,\mathbf{F}\big(\boldsymbol{x}(t_n+\sigma)\big)\,\mathrm{d}\sigma \\ &\approx \mathrm{e}^{\tau \mathbf{S}}\,\boldsymbol{x}(t_n) + \tau\,\varphi_1(\tau\mathbf{S})\,\mathbf{F}\big(\boldsymbol{x}(t_n)\big), \end{aligned}$$

where we freeze $\mathbf{F}$ at $\boldsymbol{x}(t_n)$. This yields the exponential Euler method

$$\begin{aligned} \boldsymbol{x}_{n+1} &= \mathrm{e}^{\tau \mathbf{S}}\,\boldsymbol{x}_n + \tau\,\varphi_1(\tau\mathbf{S})\,\mathbf{F}\big(\boldsymbol{x}_n\big) \\ &= \boldsymbol{x}_n + \tau\,\varphi_1(\tau\mathbf{S})\,\big(\mathbf{S}\boldsymbol{x}_n + \mathbf{F}\big(\boldsymbol{x}_n\big)\big), \end{aligned} \tag{2.3.32}$$

which is a first-order exponential Runge–Kutta method with $s = 1$ and $b_1(z) = \varphi_1(z)$. ◇

2.3.5.1 Order and convergence

We present the order conditions and convergence results for exponential Runge–Kutta methods. Further details can be found in [44, 86, 87], where the analysis is presented in a more general framework for evolution equations. In addition, [44] also provides error bounds for the fully discrete scheme, including the spatial discretization, within an abstract framework. Here, we restrict ourselves to the case of a system of ordinary differential equations (2.3.22).

We begin with the order conditions for exponential Runge–Kutta methods. These are summarized in Table 2.1 and involve the functions

$$\psi_j(\tau\mathbf{S}) = \varphi_j(\tau\mathbf{S}) - \sum_{i=1}^{s} b_i(\tau\mathbf{S})\,\frac{c_i^{j-1}}{(j-1)!}, \tag{2.3.33}$$

No.	Order	Condition
1	1	$\psi_1(\tau\mathbf{S}) = 0$
2	2	$\psi_2(\tau\mathbf{S}) = 0$
3	2	$\psi_{1,i}(\tau\mathbf{S}) = 0$
4	3	$\psi_3(\tau\mathbf{S}) = 0$
5	3	$\sum_{i=1}^{s} b_i(\tau\mathbf{S})\,\mathbf{J}\psi_{2,i}(\tau\mathbf{S}) = 0$
6	4	$\psi_4(\tau\mathbf{S}) = 0$
7	4	$\sum_{i=1}^{s} b_i(\tau\mathbf{S})\,\mathbf{J}\psi_{3,i}(\tau\mathbf{S}) = 0$
8	4	$\sum_{i=1}^{s} b_i(\tau\mathbf{S})\,\mathbf{J}\sum_{j=2}^{i-1} a_{ij}(\tau\mathbf{S})\,\mathbf{J}\psi_{2,j}(\tau\mathbf{S}) = 0$
9	4	$\sum_{i=1}^{s} b_i(\tau\mathbf{S})\,c_i\,\mathbf{K}\psi_{2,i}(\tau\mathbf{S}) = 0$

Table 2.1: Conditions for explicit exponential Runge–Kutta methods (cf. [87, Table 2.2]). Here $\mathbf{J}, \mathbf{K} \in \mathbb{R}^{N\times N}$ denote arbitrary matrices. The functions ψ_i and $\psi_{k,\ell}$ are defined in (2.3.33) and (2.3.34), respectively.

and

$$\psi_{j,i}(\tau\mathbf{S}) = \varphi_j(c_i\tau\mathbf{S})\,c_i^j - \sum_{k=1}^{i-1} a_{ik}(\tau\mathbf{S})\,\frac{c_k^{j-1}}{(j-1)!}. \tag{2.3.34}$$

The φ_k-functions are defined for $k \geq 0$ by

$$\varphi_0(z) = e^z, \qquad \varphi_k(z) = \int_0^1 e^{(1-\theta)z}\frac{\theta^{k-1}}{(k-1)!}\,d\theta, \quad k \geq 1, \tag{2.3.35}$$

or, equivalently, by the recursion formula

$$\varphi_{k+1}(z) = \frac{\varphi_k(z) - \frac{1}{k!}}{z}, \quad k \geq 0. \tag{2.3.36}$$

The order conditions arise by inserting the exact solution into (2.3.31) and analyzing the so-called defect terms.

Under suitable regularity assumptions on $\boldsymbol{x}$ and $\mathbf{F}$, the following convergence result can be established.

Theorem 2.3.10 ([44, 86])**.** *We assume that* $\mathbf{F}$ *is differentiable in a strip along the exact solution. Then there exist* $\tau_0 > 0$ *and* $C > 0$ *such that for all* $\tau \leq \tau_0$ *the numerical solution of the exponential Euler method* (2.3.32) *satisfies the error bound*

$$\|\boldsymbol{x}_n - \boldsymbol{x}(t_n)\| \leq C\,\tau.$$

For higher order methods with $2 \leq p \leq 4$, the following convergence result holds.

Theorem 2.3.11 ([44, 86])**.** *We assume that the solution $\boldsymbol{x}$ of (2.3.22) is sufficiently smooth and that $\mathbf{F}$ is sufficiently often differentiable in a strip along the exact solution. For $2 \leq p \leq 4$, assume that the order conditions of Table 2.1 hold up to order p. Then there exist $\tau_0 > 0$ and $C > 0$ such that for all $\tau \leq \tau_0$ the numerical solution $\boldsymbol{x}_n$ obtained from (2.3.31) satisfies the error bound*

$$\|\boldsymbol{x}_n - \boldsymbol{x}(t_n)\| \leq C\,\tau^p.$$

Remark 2.3.12. *In [44], an abstract spatial discretization framework is employed. In particular, the authors consider an evolution equation of the form (2.3.22) in a Hilbert space $\mathbf{X}$ together with a discrete version of this equation in a family of finite-dimensional spaces $\mathbf{X}_h$, and establish error bounds for the fully discrete scheme, including the spatial discretization error. In our setting, however, we have $\mathbf{X} = \mathbf{X}_h = \mathbb{R}^N$, so that the error terms associated with the spatial discretization vanish.*

Remark 2.3.13. *For parabolic problems, smoothness can be used to slightly relax the order conditions. For details, see [86, 87] and the references therein.*

The implementation of exponential integrators requires the calculation of products of matrix functions with a vector. This can be computationally expensive, especially for large-scale problems, and efficient algorithms are therefore essential. We will discuss this issue in more details in Section 2.4.

2.3.6 Conclusion

The time integration methods presented in the last sections involve different trade-offs between accuracy, stability, and computational cost.

Fully implicit methods, such as Gauss and Radau methods, combine high order with excellent stability properties, making them well suited for stiff problems. However, they are difficult to implement, as they require multiple (simplified) Newton iterations per time-step and the solution of (possibly coupled) systems of nonlinear equations.

DIRK and SDIRK methods are computationally cheaper than fully implicit methods. However, they still require a nonlinear solve at each stage and achieve lower classical order compared to Gauss and Radau methods, and suffer significantly from order reduction when applied to stiff problems.

Rosenbrock methods require only the solution of linear systems. They are easier to implement than fully implicit methods and are particularly efficient when the Jacobian of the nonlinearity can be computed or approximated efficiently. However, they are generally less stable for nonlinear problems than DIRK methods, and constructing high-order schemes is more challenging.

IMEX-RK methods are especially useful for problems that can be split into stiff and non-stiff components. They retain good stability properties by treating the stiff part implicitly while remaining efficient through the explicit treatment of the non-stiff (often nonlinear) terms. However, if the splitting is poorly chosen, the method may fail.

For stiff problems, IMEX-RK and Rosenbrock methods suffer from order reduction effects when the classical order is greater than two.

Exponential integrators, as well as IMEX schemes, are well suited for problems that can be decomposed into linear stiff and nonlinear non-stiff parts. They preserve stability properties of the exact solution while avoiding nonlinear solves. However, the efficient computation and implementation of products of matrix functions with vectors is challenging.

For semilinear damped wave equations, IMEX methods and exponential integrators are particularly attractive because they exploit the underlying problem structure. This leads to computational efficiency while maintaining very good stability properties.

Bibliographical notes for IMEX and exponential integrator schemes for semilinear wave equations. IMEX–RK schemes for semilinear wave equations are studied in [11, 52]. Crank–Nicolson/leapfrog (CNLF) schemes are analyzed in [30, 83, 98], while θ-scheme/leapfrog combinations are considered in [106].

Exponential integrators are well understood in the context of semilinear wave equations. In [61], spatially discretized semilinear wave equations with periodic boundary conditions are investigated, while [78] studies exponential splitting methods for abstract semilinear evolution equations. Full discretization error bounds for explicit exponential Runge–Kutta methods combined with an abstract spatial discretization are derived in [44]. A class of second-order exponential integrators for low-regularity solutions is investigated in [29]. Semilinear wave problems with highly oscillatory solutions in time are studied in [16].

2.4 The matrix φ_k-functions

As discussed in Section 2.3.5, exponential integrators require the efficient evaluation of matrix φ_k-functions applied to a vector $\mathbf{v} \in \mathbb{C}^N$, that is,

$$\varphi_k(\mathbf{S})\mathbf{v}, \qquad \mathbf{S} \in \mathbb{C}^{N\times N}, \quad k \geq 0,$$

where the φ_k-function are defined in (2.3.35). For large-scale problems, the direct computation of these quantities is generally infeasible. Therefore, efficient approximation

techniques are required. In the following, we present several methods for approximating matrix-function–vector products of this form. The section is based on [66, 71, 79, 84, 123].

2.4.1 Definition of matrix functions

Let $\phi : \mathbb{C} \to \mathbb{C}$ be a function that is analytic on an open set $\Lambda \subset \mathbb{C}$, which contains the spectrum of $\mathbf{S}$. Then there are three equivalent ways to define matrix functions $\phi(\mathbf{S})$. The first approach relies on the Jordan canonical form of $\mathbf{S}$:

$$\mathbf{T}^{-1}\mathbf{S}\mathbf{T} = \mathbf{J} = \operatorname{diag}(\mathbf{J}_1, \ldots, \mathbf{J}_p),$$

with the

$$\mathbf{J}_s = \mathbf{J}_s(\lambda_s) = \begin{pmatrix} \lambda_s & 1 & & \\ & \lambda_s & \ddots & \\ & & \ddots & 1 \\ & & & \lambda_s \end{pmatrix} \in \mathbb{C}^{n_s \times n_s},$$

where $\mathbf{T}$ is nonsingular and $n_1 + \cdots + n_p = N$.

Definition 2.4.1. The matrix function $\phi(\mathbf{S})$ is defined as

$$\phi(\mathbf{S}) := \mathbf{T}\phi(\mathbf{J})\mathbf{T}^{-1} = \mathbf{T}\, \operatorname{diag}(\phi(\mathbf{J}_1), \ldots, \phi(\mathbf{J}_p))\, \mathbf{T}^{-1},$$

where

$$\phi(\mathbf{J}_s) = \phi(\mathbf{J}_s(\lambda_s)) = \begin{pmatrix} \phi(\lambda_s) & \phi'(\lambda_s) & \cdots & \dfrac{\phi^{(n_s-1)}(\lambda_s)}{(n_s-1)!} \\ & \phi(\lambda_s) & \ddots & \vdots \\ & & \ddots & \phi'(\lambda_s) \\ & & & \phi(\lambda_s) \end{pmatrix} \in \mathbb{C}^{n_s \times n_s}.$$

Let N_k denote the size of the largest Jordan block corresponding to the eigenvalue λ_k, for $k = 1, \ldots, n$, where n is the number of distinct eigenvalues of $\mathbf{S}$. Then we can alternatively define the matrix function using interpolation.

Definition 2.4.2. The matrix function $\phi(\mathbf{S})$ is defined as

$$\phi(\mathbf{S}) := p(\mathbf{S}),$$

where p denotes the Hermite interpolation polynomial of degree at most $N_1 + \cdots + N_n$, which satisfies

$$p^{(j)}(\lambda_s) = \phi^{(j)}(\lambda_s), \quad j = 0, \ldots, N_s - 1, \quad s = 1, \ldots, n.$$

The third definition is obtained from Cauchy's integral formula (see, e.g. [6, Chapter 4]). Let Γ be a closed, positively oriented, rectifiable curve in Λ and $\lambda \in \mathbb{C}$ be enclosed by Γ with winding number one. The function ϕ, which is analytic in Λ, can be represented by

$$\phi(\lambda) = \frac{1}{2\pi i} \int_\Gamma \phi(z)(z-\lambda)^{-1}\, d\lambda.$$

This motivates the following definition.

Definition 2.4.3. The matrix function $\varphi(\mathbf{S})$ is defined as

$$\phi(\mathbf{S}) := \frac{1}{2\pi i} \int_\Gamma \phi(\lambda)(\lambda \mathbf{I} - \mathbf{S})^{-1}\ \mathrm{d}\lambda,$$

where Γ is a closed, positively oriented, rectifiable curve in Λ which has winding number one about all eigenvalues of $\mathbf{S}$.

An important result is that all three definitions are equivalent. A proof of this can be found in [79, Theorem 1.12].

2.4.2 Krylov subspace methods

As explained in the introduction, the focus lies on computing the action of $\varphi_k(\mathbf{S})$ on a vector $\mathbf{v}$, rather than forming $\varphi_k(\mathbf{S})$ explicitly. In particular, constructing $\varphi_k(\mathbf{S})$ can be very memory-intensive, since it is generally dense even when $\mathbf{S}$ itself is sparse.

Since matrix functions can be represented in terms of matrix polynomials, the (standard or polynomial) Krylov subspace is considered. This subspace consists of all vectors generated by applying successive powers of $\mathbf{S}$ to $\mathbf{v}$ up to a certain degree.

Definition 2.4.4. For $m \geq 1$, the m-th Krylov subspace of the matrix $\mathbf{S}$ and the vector $\mathbf{v} \in \mathbb{C}^N$ is defined by

$$\mathcal{K}_m(\mathbf{S}, \mathbf{v}) := \operatorname{span}\{\mathbf{v}, \mathbf{S}\mathbf{v}, \ldots, \mathbf{S}^{m-1}\mathbf{v}\} = \{p(\mathbf{S})\mathbf{v} : p \in \mathcal{P}_{m-1}\}.$$

As seen in Definition 2.4.2, matrix functions can be represented by a polynomial p of sufficiently large degree such that $\varphi_k(\mathbf{S}) = p(\mathbf{S})$. This motivates seeking an approximation of the form

$$\varphi_k(\mathbf{S})\mathbf{v} \approx p_m(\mathbf{S})\mathbf{v} \in \mathcal{K}_m(\mathbf{S}, \mathbf{v}), \qquad p_m \in \mathcal{P}_{m-1}. \tag{2.4.1}$$

There are two common approaches to construct such approximations:

One approach is to construct an orthonormal basis of $\mathcal{K}_m$ and use it to compute the matrix function of a suitable projection of $\mathbf{S}$ onto the Krylov subspace, which will be presented in Section 2.4.2.1.

Alternatively, by [40], we have

$$\|\varphi_k(\mathbf{S})\| \leq \mathcal{C} \sup_{z \in W(\mathbf{S})} |\varphi_k(z)|, \tag{2.4.2}$$

for $\mathcal{C} = (1 + \sqrt{2})$, where

$$W(\mathbf{S}) = \{(\mathbf{S}v, v) \mid v \in \mathbb{C}^N,\ \|v\| = 1\} \tag{2.4.3}$$

denotes the *field-of-values* of $\mathbf{S}$ with respect to a given inner product $(\cdot, \cdot)$, which induces the norm $\|\cdot\|$. Therefore, as a second approach to approximating the matrix function, we seek a suitable compact set K that contains $W(\mathbf{S})$, as well as a sequence of polynomials such that the scalar φ_k-function is well approximated on K. This is discussed in Section 2.4.2.3.

2.4.2.1 Arnoldi method

We construct an orthonormal basis

$$\mathbf{V}_m = [\mathbf{v}_1\ \mathbf{v}_2\ \cdots\ \mathbf{v}_m] \in \mathbb{C}^{N \times m}$$

of the Krylov subspace $\mathcal{K}_m(\mathbf{S}, \mathbf{v}_1)$ using a stabilized Gram–Schmidt procedure, the so-called Arnoldi process. In this process, a new basis vector is obtained via

$$\widetilde{\mathbf{v}}_{m+1} = \mathbf{S}\mathbf{v}_m - \sum_{j=1}^{m} h_{j,m}\mathbf{v}_j, \qquad \mathbf{v}_{m+1} = \frac{1}{h_{m+1,m}}\widetilde{\mathbf{v}}_{m+1}, \qquad h_{m+1,m} = \|\widetilde{\mathbf{v}}_{m+1}\|.$$

By applying the orthogonality conditions $(\mathbf{v}_j, \mathbf{v}_{m+1}) = 0$ for $j = 1, \ldots, m$, we obtain

$$h_{j,m} = (\mathbf{S}\mathbf{v}_m, \mathbf{v}_j), \qquad j = 1, \ldots, m.$$

This leads to the Arnoldi recurrence

$$\mathbf{S}\mathbf{V}_m = \mathbf{V}_m\mathbf{H}_m + h_{m+1,m}\,\mathbf{v}_{m+1}\mathbf{e}_m^T, \tag{2.4.4}$$

where $\mathbf{H}_m = (h_{ij})_{i,j=1}^m \in \mathbb{C}^{m \times m}$ is an upper Hessenberg matrix, i.e., $h_{ij} = 0$ for $i > j + 1$, and $\mathbf{e}_m$ denotes the mth unit vector in $\mathbb{C}^m$.

In the following, we assume that $\mathbf{V}_m$ is orthonormal with respect to the standard Euclidean inner product for the sake of presentation. However, the inner product used within the Arnoldi method should be chosen according to the application. Adapting the algorithm and the results stated here to other inner products is straightforward.

Using the orthonormality condition $\mathbf{V}_m^H \mathbf{V}_m = \mathbf{I}$, we multiply the Arnoldi relation from the left by $\mathbf{V}_m^H$ and obtain

$$\mathbf{V}_m^H \mathbf{S} \mathbf{V}_m = \mathbf{H}_m.$$

A key property is that the field-of-values of $\mathbf{H}_m$ is contained in the field-of-values of $\mathbf{S}$. For an arbitrary $\boldsymbol{y} \in \mathbb{C}^m$ with $\|\boldsymbol{y}\| = 1$, it holds that:

$$\boldsymbol{y} \cdot \mathbf{H}_m \boldsymbol{y} = \boldsymbol{y} \cdot \mathbf{V}_m^H \mathbf{S} \mathbf{V}_m \boldsymbol{y} = \boldsymbol{x} \cdot \mathbf{S} \boldsymbol{x} \in W(\mathbf{S}),$$

by setting $\boldsymbol{x} := \mathbf{V}_m \boldsymbol{y} \in \mathbb{C}^N$, where $\|\boldsymbol{x}\| = \|\mathbf{V}_m \boldsymbol{y}\| = \|\boldsymbol{y}\| = 1$. This implies

$$W(\mathbf{H}_m) \subseteq W(\mathbf{S}). \tag{2.4.5}$$

It is easy to see that for each $p \in \mathcal{P}_{m-1}$

$$p(\mathbf{S})\mathbf{v} = \mathbf{V}_m p(\mathbf{V}_m^H \mathbf{S} \mathbf{V}_m) \mathbf{V}_m^H \mathbf{v} = \|\mathbf{v}\| \, \mathbf{V}_m p(\mathbf{H}_m) \mathbf{e}_1.$$

Motivated by this consideration and (2.4.1), the Arnoldi approximation of order m is defined by

$$\varphi_k(\mathbf{S})\mathbf{v} \approx \|\mathbf{v}\| \, \mathbf{V}_m \varphi_k(\mathbf{H}_m) \mathbf{e}_1. \tag{2.4.6}$$

Since in practice $m \ll N$, the matrix function $\varphi_k(\mathbf{H}_m)$ can be computed by diagonalizing $\mathbf{H}_m$ or approximated using the Padé approximation. An alternative approach, leading to the same result, is presented in [84], where the deep connection between computing matrix functions and solving linear systems of the form

$$(\lambda \mathbf{I} - \mathbf{S}) \boldsymbol{x}(\lambda) = \mathbf{v}, \tag{2.4.7}$$

is demonstrated. In the following, we will use the identity

$$\mathcal{K}_m(\mathbf{S}, \mathbf{v}) = \mathcal{K}_m(\lambda \mathbf{I} - \mathbf{S}, \mathbf{v})$$

for all $\lambda \in \mathbb{C}$, which implies

$$(\lambda \mathbf{I} - \mathbf{S}) \mathbf{V}_m = \mathbf{V}_m (\lambda \mathbf{I} - \mathbf{H}_m) - h_{m+1,m} \mathbf{v}_{m+1} \mathbf{e}_m^T.$$

The Galerkin approximation to the solution $\boldsymbol{x}(\lambda)$ of (2.4.7) is defined by

$$\boldsymbol{x}_m(\lambda) = \mathbf{V}_m \boldsymbol{y}_m(\lambda) \in \mathcal{K}_m(\mathbf{S}, \mathbf{v}), \tag{2.4.8}$$

for some $\boldsymbol{y}_m(\lambda) \in \mathbb{C}^m$ such that the residual

$$\mathbf{r}_m(\lambda) = \mathbf{v} - (\lambda \mathbf{I} - \mathbf{S}) \boldsymbol{x}_m(\lambda)$$

is orthogonal to the Krylov subspace $\mathcal{K}_m(\mathbf{S}, \mathbf{v})$, i.e.,

$$\mathbf{0} = \mathbf{V}_m^H \mathbf{r}_m(\lambda) = \mathbf{V}_m^H \mathbf{v} - \mathbf{V}_m^H(\lambda \mathbf{I} - \mathbf{S})\boldsymbol{x}_m(\lambda) = \|\mathbf{v}\|\mathbf{e}_1 - (\lambda \mathbf{I} - \mathbf{H}_m)\boldsymbol{y}_m(\lambda). \tag{2.4.9}$$

If $\lambda \notin W(\mathbf{S})$, then the matrices $\lambda \mathbf{I} - \mathbf{S}$ and, by (2.4.5), $\lambda \mathbf{I} - \mathbf{H}_m$ are invertible, and (2.4.9) yields

$$(\lambda \mathbf{I} - \mathbf{S})^{-1}\mathbf{v} = \boldsymbol{x} \approx \boldsymbol{x}_m = \mathbf{V}_m(\lambda \mathbf{I} - \mathbf{H}_m)^{-1}\|\mathbf{v}\|\mathbf{e}_1,$$

which is the approximation obtained by the Full Orthogonalization Method (FOM) (cf., e.g., [121]) applied to (2.4.7). Using this relation in Definition 2.4.3 leads to the Arnoldi approximation (2.4.6)

$$\begin{aligned}
\varphi_k(\mathbf{S})\mathbf{v} &= \frac{1}{2\pi i}\int_\Gamma \varphi_k(\lambda)(\lambda \mathbf{I} - \mathbf{S})^{-1}\mathbf{v}\ \mathrm{d}\lambda \\
&\approx \|\mathbf{v}\|\mathbf{V}_m \left(\frac{1}{2\pi i}\int_\Gamma \varphi_k(\lambda)(\lambda \mathbf{I} - \mathbf{H}_m)^{-1}\mathbf{e}_1\ \mathrm{d}\lambda \right) \\
&= \|\mathbf{v}\|\mathbf{V}_m \varphi_k(\mathbf{H}_m)\mathbf{e}_1,
\end{aligned}$$

where Γ be a closed contour in $\mathbb{C}$ surrounding the field-of-values $W(\mathbf{S}) \supset W(\mathbf{H}_m)$.

2.4.2.2 Quasi-optimality of Krylov subspace approximations

We recall that a direct consequence of the estimate (2.4.2) is that for any bounded set $K \supset W(\mathbf{S})$, we get the estimate

$$\|\varphi_k(\mathbf{S})\| \leq \mathcal{C} \sup_{z \in K} |\varphi_k(z)| \leq \mathcal{C} \sup_{z \in \partial K} |\varphi_k(z)|, \tag{2.4.10}$$

where the last inequality follows from the maximum principle for holomorphic functions (cf. e.g, [6, Chapter 3.4]). This enables us to bound the error arising from the Arnoldi approximation in the following way.

Theorem 2.4.5 (Theorem 4.16 in [66])**.** *Let $\mathbf{H}_m = \mathbf{V}_m^H \mathbf{S} \mathbf{V}_m$ be the compression of $\mathbf{S}$ onto the Krylov subspace $\mathcal{K}_m(\mathbf{S}, \mathbf{v})$. Then, for any set $K \supseteq W(\mathbf{S})$, we have*

$$\|\varphi_k(\mathbf{S})\mathbf{v} - \mathbf{V}_m \varphi_k(\mathbf{H}_m)\mathbf{V}_m^H \mathbf{v}\| \leq 2\mathcal{C}\|\mathbf{v}\| \min_{p \in \mathcal{P}_{m-1}} \sup_{z \in K} |\varphi_k(z) - p(z)|.$$

Remark 2.4.6. *Error bounds can be derived w.r.t. to m, for instance, using Chebyshev or Faber polynomials [17, 19, 84, 132], as well as truncated Taylor series [123]. These bounds depend on the field of values $W(\mathbf{S})$, in particular on its extent in the complex plane. If, e.g., $W(\mathbf{S})$ contains points with large magnitude, this typically leads to worse approximation properties and therefore slower convergence.*

For normal matrices, this estimate holds with $\mathcal{C} = 1$. Moreover, in this case it suffices that K contains the convex hull of the eigenvalues of $\mathbf{S}$ (cf. [66, p.51]).

This result demonstrates the near-optimality of Krylov subspace methods, namely that the approximation error is reduced to that of the best possible polynomial approximation of degree m for the complex φ_k-function, up to a constant factor. A major advantage of the Arnoldi method is that this quasi-optimality can be achieved without explicit knowledge of the field-of-values. However, this requires long recurrences to compute new basis vectors, see (2.4.2.1). In addition, the entire basis matrix $\mathbf{V}_m$ must be stored in order to approximate the matrix function in $\mathcal{K}_m$.

This drawback can be partially avoided by using the Lanczos algorithm, proposed in 1950 by [96]. In this approach, an auxiliary basis $\mathbf{W}_m = [\mathbf{w}_1\ \mathbf{w}_2\ \cdots\ \mathbf{w}_m]$ is constructed, spanning the Krylov subspace associated with $\mathbf{S}^H$.

The Lanczos vectors $\mathbf{v}_j$ and $\mathbf{w}_j$ are constructed to satisfy a biorthogonality condition and can be computed via short recursions. For non-symmetric matrices, however, these recursions may break down or become numerically unstable due to occurrence of (near) zero denominators. Such instabilities can be avoided by employing look-ahead strategies, which skip over problematic steps (see, e.g., [57]).

In the context of linear systems, the Lanczos process is used to construct the biconjugate gradient (BiCG) method [126, Chapter 7]. In this method it is possible to efficiently update the approximation of the solution in $\mathcal{K}_m$ from the previous iteration in $\mathcal{K}_{m-1}$. To the best of our knowledge, no equally efficient and generally applicable update strategy is known for matrix functions. Consequently, the entire basis matrix $\mathbf{V}_m$ must also be stored for the Lanczos method. We refer to [84, Chapter 4] for more details.

In the following, we therefore study a method that, under the assumption that a set containing the field-of-values is known, avoids the storage of this basis and achieves superlinear convergence.

2.4.2.3 Chebychev approximations

We consider the Chebyshev polynomials T_m of degree m, which for $z \in \mathbb{C}$ can be efficiently computed using the three-term recurrence relation

$$T_{m+1}(z) = 2z\,T_m(z) - T_{m-1}(z), \qquad T_0(z) = 1, \quad T_1(z) = z. \tag{2.4.11}$$

These polynomials play a central role in approximation theory, in particular due to their characterization as solutions of a min–max problem on $[-1, 1]$, see, e.g., [104, 120] for further details.

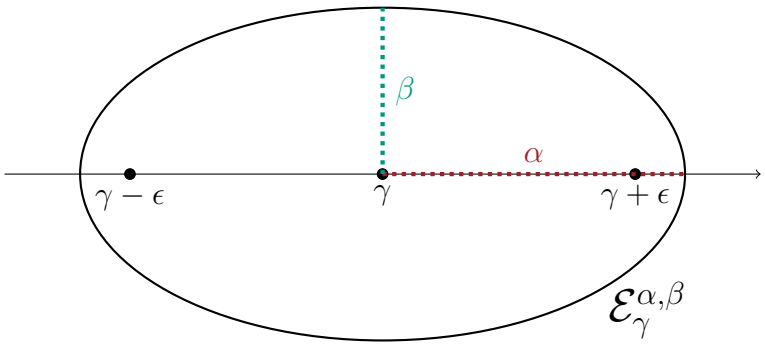


Figure 2.1: Ellipse $\mathcal{E}_\gamma^{\alpha,\beta}$ with linear eccentricity ϵ

For (skew-)Hermitian matrices, the field of values coincides with the convex hull of the spectrum, and all eigenvalues are contained in a bounded interval on either the real or imaginary axis. After an appropriate affine transformation mapping this interval to $[-1, 1]$, Chebyshev polynomial approximations can be applied effectively. In this setting, they have been used to derive practical methods for approximating the matrix φ_k-functions as motivated by (2.4.10), see, for example, [100, Chapter III.2.1].

In general, $\mathbf{S}$ may have complex eigenvalues, so its field-of-values is not contained in the real or the imaginary axis. We therefore seek a simple and small geometric shape that enclose the field-of-values. In this thesis, we consider ellipses of the form

$$\mathcal{E}_\gamma^{\alpha,\beta} = \{z = \alpha\cos\varphi + i\beta\sin\varphi + \gamma \mid \varphi \in [-\pi, \pi]\} \subset \mathbb{C}, \qquad \alpha, \beta \geq 0,\ \gamma \in \mathbb{R}, \tag{2.4.12}$$

with linear eccentricity $\epsilon = \sqrt{\alpha^2 - \beta^2}$. The center is given by γ, and the foci are located at $\gamma \pm \epsilon$ (see Figure 2.1). Note that these ellipses are symmetric with respect to the real axis, since in application we consider only matrices $\mathbf{S} \in \mathbb{R}^{N\times N}$ whose field-of-values is symmetric with respect to the real axis, since $\lambda \in W(\mathbf{S})$ if and only if $\overline{\lambda} \in W(\overline{\mathbf{S}}) = W(\mathbf{S})$.

A key reason for considering ellipses is that Chebyshev polynomials provide nearly optimal polynomial approximations in these regions (see [63]), analogous to the interval case. Therefore, we aim to enclose the field-of-values by a suitable ellipse of the form (2.4.12) and derive a Chebyshev approximation of the scalar φ_k-functions on this ellipse. This will be done in Chapter 4 using a general approach based on Faber polynomials, which coincide with shifted and scaled Chebyshev polynomials in the case of ellipses. In the following, however, we present a more elementary derivation, illustrating how the results for the interval case naturally extend to ellipses.

Before constructing this approximation, we first present two technical lemmas. We begin by showing how ellipses can be parametrized in terms of their linear eccentricity,

center and a (complex) parameter ω.

Lemma 2.4.7. *Let $\alpha, \beta \geq 0$ and $\gamma \in \mathbb{R}$ and*

$$z = \alpha \cos\psi + \mathrm{i}\,\beta \sin\psi + \gamma \in \mathcal{E}_\gamma^{\alpha,\beta}, \quad \psi \in (-\pi, \pi].$$

Then it holds that if $\alpha \neq \beta$

$$\frac{z-\gamma}{\epsilon} = \cos\theta_\psi^\omega = \cosh\omega \cos\psi + \mathrm{i}\sinh\omega \sin\psi,$$

where $\theta_\psi^\omega = \psi - \mathrm{i}\log(\omega)$ with

$$\epsilon = \sqrt{\alpha^2 - \beta^2}\,, \quad \omega = \sqrt{\frac{\alpha+\beta}{\alpha-\beta}}. \tag{2.4.13}$$

Remark 2.4.8. *We work with the branches of the complex square root and the logarithm, so that $\sqrt{-1} = \mathrm{i}$ and $\log(\mathrm{i}) = \mathrm{i}\frac{\pi}{2}$.*

Proof of Lemma 2.4.7. Set $\delta = \log(\omega)$. It follows by the complex representation of the cosine function that

$$\begin{aligned}\cos(\theta_\psi^\omega) &= \frac{1}{2}(\mathrm{e}^{\mathrm{i}\,\psi+\delta} + \mathrm{e}^{-\mathrm{i}\,\psi-\delta}) \\ &= \frac{1}{2}((\cos\psi + \mathrm{i}\sin\psi)\,\mathrm{e}^{\delta} + (\cos\psi - \mathrm{i}\sin\psi)\,\mathrm{e}^{-\delta}) \\ &= \cosh\delta\cos\psi + \mathrm{i}\sinh\delta\sin\psi.\end{aligned}$$

We get

$$\begin{aligned}\cosh\delta\cos\psi + \mathrm{i}\sinh\delta\sin\psi &= \frac{1}{2}\Big((e^{\delta} + e^{-\delta})\cos\psi + \mathrm{i}(e^{\delta} - e^{-\delta})\sin\psi\Big) \\ &= \frac{\alpha\cos\psi + \mathrm{i}\,\beta\sin\psi}{\sqrt{\alpha^2-\beta^2}} = \frac{z-\gamma}{\epsilon}.\end{aligned}$$

□

If a function g is even and 2π-periodic, then multiplying it by the sine function produces a function that is odd and 2π-periodic. As a result, the integral of this product over one full period vanishes (see, e.g. [129, Chapter 1]). We show that in our case, this extends in an analogous way to the complex setting.

Lemma 2.4.9. *Using the notation and assumptions of Lemma 2.4.7, we obtain for all $j \in \mathbb{N}$ that*

$$\int_{-\pi}^{\pi} \sin(j\theta_\psi^\omega)\varphi_k(\epsilon\cos\theta_\psi^\omega + \gamma)\,\mathrm{d}\psi = 0.$$

Proof. First, we assume $\alpha > \beta$. We set $\delta = \log(\omega) \in \mathbb{R}$, and $g(\psi) = \varphi_k(\epsilon \cos\psi + \gamma)$. Let Γ be the closed curve that is the boundary of the complex square $[-\pi, \pi] + \mathrm{i}[-\delta, 0]$. Then, by the Cauchy integral theorem, we have

$$\begin{aligned} 0 &= \int_{\Gamma} \sin(j\psi)\varphi_k(\epsilon(\cos\psi) + \gamma)\ \mathrm{d}\psi \\ &= \int_{[-\pi - \mathrm{i}\,\delta, \pi - \mathrm{i}\,\delta]} \sin(j\psi)g(\psi)\ \mathrm{d}\psi + \int_{[\pi, -\pi]} \sin(j\psi)g(\psi)\ \mathrm{d}\psi \\ &\quad + \int_{[-\pi, -\pi - \mathrm{i}\,\delta]} \sin(j\psi)g(\psi)\ \mathrm{d}\psi + \int_{[\pi - \mathrm{i}\,\delta, \pi]} \sin(j\psi)g(\psi)\ \mathrm{d}\psi. \end{aligned}$$

Since g is an even and 2π-periodic function, it holds that

$$\int_{-\pi}^{\pi} \sin(j\psi)g(\psi)\ \mathrm{d}\psi = 0.$$

Further, we obtain

$$\begin{aligned} &\int_{-\delta}^{0} \sin\big(j(\pi + \mathrm{i}\,y)\big)g(\pi + \mathrm{i}\,y)\ \mathrm{d}y \\ &\qquad = -\int_{0}^{-\delta} \sin\big(j(-\pi + \mathrm{i}\,y)\big)g(-\pi + \mathrm{i}\,y)\ \mathrm{d}y. \end{aligned}$$

Therefore, we conclude that

$$0 = \int_{[-\pi - \mathrm{i}\,\delta, \pi - \mathrm{i}\,\delta]} \sin(j\psi)g(\psi)\ \mathrm{d}\psi.$$

Substitution gives the result. If $\alpha < \beta$, then $\log(\omega) = \log|\omega| + \mathrm{i}\,\frac{\pi}{2}$ and therefore $\theta_{\psi}^{\omega} = \theta_{\psi + \frac{\pi}{2}}^{|\omega|} = \theta_{\eta}^{|\omega|}$ for $\eta \in (-\frac{\pi}{2}, \frac{3\pi}{2}]$. Hence, by the same arguments as in the previous case, the result follows. □

We are now able to construct Chebyshev polynomial approximations of the φ_k-functions on ellipses. We do this by generalizing the approach used in the simpler case of approximations on an interval (see [100, Chapter III.2.1]).

Theorem 2.4.10. *Let $\mathcal{E}_{\gamma}^{\alpha,\beta}$ be an ellipse with $\alpha \neq \beta$ and $z \in \mathcal{E}_{\gamma}^{\alpha,\beta}$. We obtain*

$$\begin{aligned} \varphi_k(z) &= c_0^{(k)} + 2\sum_{j=1}^{\infty} c_j^{(k)} \left(\frac{\alpha - \beta}{\alpha + \beta}\right)^{\frac{j}{2}} T_j(\widehat{z}), \\ c_j^{(k)} &= \frac{1}{2\pi} \int_{-\pi}^{\pi} \mathrm{e}^{-\mathrm{i}\,j\psi}\, \varphi_k(\alpha\cos\psi + \mathrm{i}\,\beta\sin\psi + \gamma)\ \mathrm{d}\psi, \end{aligned} \tag{2.4.14}$$

where $\widehat{z} = \dfrac{z - \gamma}{\epsilon}$ with $\epsilon = \sqrt{\alpha^2 - \beta^2}$.

Proof. The Fourier series of the smooth and periodic function

$$\begin{aligned} g(\psi) &= \varphi_k(\alpha \cos\psi + \mathrm{i}\,\beta \sin\psi + \gamma) \\ &= \varphi_k(\epsilon \cos\theta_\psi^\omega + \gamma) \end{aligned}$$

with ω and θ_ψ^ω as in Lemma 2.4.7, is given by

$$\sum_{j=-\infty}^{\infty} c_j^{(k)}\, \mathrm{e}^{\mathrm{i}\, j\psi}, \quad c_j^{(k)} = \frac{1}{2\pi} \int_{-\pi}^{\pi} \mathrm{e}^{-\mathrm{i}\, j\psi}\, g(\psi)\, \mathrm{d}\psi,$$

and converges uniformly (see, e.g. [129, Corollary 10.4.]) towards g. Therefore, we obtain

$$\begin{aligned} g(\psi) &= \sum_{j=-\infty}^{\infty} c_j^{(k)}\, \mathrm{e}^{\mathrm{i}\, j\psi - \log(\omega) j + \log(\omega) j} \\ &= \sum_{j=-\infty}^{\infty} c_j^{(k)} \omega^{-j}\, \mathrm{e}^{\mathrm{i}\, j\theta_\psi^\omega}. \end{aligned}$$

By Lemma 2.4.9 it holds

$$\begin{aligned} c_j^{(k)}\, \mathrm{e}^{-\omega j} &= \frac{1}{2\pi} \int_{-\pi}^{\pi} \mathrm{e}^{-\mathrm{i}\, j\theta_\psi^\omega}\, g(\psi)\, \mathrm{d}\psi \\ &= \frac{1}{2\pi} \int_{-\pi}^{\pi} (\cos(j\theta_\psi^\omega) - \mathrm{i} \sin(j\theta_\psi^\omega)) g(\psi)\, \mathrm{d}\psi \\ &= \frac{1}{2\pi} \int_{-\pi}^{\pi} \cos(j\theta_\psi^\omega) g(\psi)\, \mathrm{d}\psi. \end{aligned}$$

Therefore, we get

$$g(\psi) = c_0^{(k)} + 2 \sum_{j=1}^{\infty} c_j^{(k)}\, \mathrm{e}^{-\log(\omega) j} \cos(j\theta_\psi^\omega).$$

Substituting $z = \epsilon \cos\theta_\psi^\omega + \gamma$ and using $T_j(\cos(\theta_\psi^\omega)) = \cos(j\theta_\psi^\omega)$, which is true also for the complex θ_ψ^ω argument (cf. [111, p.433]) proves the claim. □

The theorem motivates approximating the product $\varphi_k(\mathbf{S})\mathbf{v}$ by truncation

$$\varphi_k(\mathbf{S})\mathbf{v} \approx c_0 \mathbf{v} + 2 \sum_{j=1}^{m} c_j^{(k)} \left(\frac{\alpha - \beta}{\alpha + \beta} \right)^{\frac{j}{2}} T_j\left(\frac{\mathbf{S} - \gamma \mathbf{I}}{\epsilon} \right) \mathbf{v}. \tag{2.4.15}$$

This representation can be computed efficiently using the three terms recursion of the Chebychev polynomials (2.4.11), whereby only the last three vectors need to be stored to compute the polynomial of the next order. The coefficients can be computed using a fast Fourier transform (FFT). In Chapter 4, we prove that (2.4.15) converges uniformly for $m \to \infty$ and derive error bounds in an arbitrary matrix norm induced by a vector norm.

Moreover, we present an algorithm for computing a *small* ellipse enclosing the convex hull of the spectrum for $\mathbf{S} \in \mathbb{R}^{N\times N}$. This construction can be used when $\mathbf{S}$ is normal or a small perturbation of a normal matrix, since in this case the field-of-values coincides with the convex hull of the spectrum or differs from it only slightly. In addition, we introduce a stopping criterion for the method that is based on an estimator for the residual error and can be efficiently computed using short-term recursion.

2.4.3 Rational Krylov subspace methods

We recall that in standard Krylov subspace methods, we seek a polynomial approximation for the φ_k-function of the matrix of the form

$$\varphi_k(\mathbf{S})\mathbf{v} \approx p(\mathbf{S})\mathbf{v} \in \mathcal{K}_m(\mathbf{S}, \mathbf{v}), \qquad p \in \mathcal{P}_{m-1},$$

where p approximates φ_k on a superset of the field-of-value of $\mathbf{S}$. If this is large, high-degree polynomials are required to obtain a sufficient approximation. Rational functions, however, can often achieve comparable accuracy much more efficiently. In particular, they are able to capture the asymptotic behavior of the φ_k-functions significantly better than polynomial approximations.

Therefore, we consider *rational* Krylov subspaces based on rational functions with $m-1$ poles that are not contained in the spectrum $\sigma(\mathbf{A})$. Since these methods are not explored further in this thesis, we restrict ourselves to a brief overview and refer to [66, 71] for more details.

Definition 2.4.11. Assume that the polynomial $q_{m-1} \in \mathcal{P}_{m-1}$ has no roots in $\sigma(\mathbf{S})$. Then, for $m \geq 1$, the rational Krylov subspace is defined by

$$\mathcal{Q}_m(\mathbf{S}, \mathbf{v}) := q_{m-1}(\mathbf{S})^{-1}\mathcal{K}_m(\mathbf{S}, \mathbf{v}) = \left\{ r(\mathbf{S})\mathbf{v} \ : \ r \in \frac{\mathcal{P}_{m-1}}{q_{m-1}} \right\},$$

where we denote by

$$\frac{\mathcal{P}_{m-1}}{q_{m-1}} := \left\{ \frac{p_{m-1}(z)}{q_{m-1}(z)} \ : \ p_{m-1} \in \mathcal{P}_{m-1} \right\}.$$

When approximating the matrix φ_k function using rational Krylov subspaces, we seek an approximation of the form

$$\varphi_k(\mathbf{S})\mathbf{v} \approx r(\mathbf{S})\mathbf{v} \in \mathcal{Q}_m(\mathbf{S}, \mathbf{v}), \qquad r \in \frac{\mathcal{P}_{m-1}}{q_{m-1}}.$$

Similar to standard Krylov subspace methods, rational Krylov methods exhibit a near-optimal approximation property.

Theorem 2.4.12 (Corollary 3.4 in [72]). *Let $\mathbf{V}_m$ be an orthonormal basis of the rational Krylov subspace $\mathcal{Q}_m(\mathbf{S},\mathbf{v})$ and $\widehat{\mathbf{H}}_m = \mathbf{V}_m^H \mathbf{S} \mathbf{V}_m$ be the compression of $\mathbf{S}$ onto $\mathcal{Q}_m(\mathbf{S},\mathbf{v})$. Then, for any set $K \supseteq W(\mathbf{S})$, we have*

$$\|\varphi_k(\mathbf{S})\mathbf{v} - \mathbf{V}_m \varphi_k(\widehat{\mathbf{H}}_m)\mathbf{V}_m^H \mathbf{v}\| \leq 2\mathcal{C}\|\mathbf{v}\| \min_{r \in \frac{\mathcal{P}_{m-1}}{q_{m-1}}} \sup_{z \in K} |\varphi_k(z) - r(z)|,$$

where $\mathcal{C}$ is the Crouzeix constant in (2.4.2).

Since there are many possible choices of q_{m-1}, there is, in contrast to standard Krylov subspace methods, no unique rational Krylov method. In the following, we focus on the *shift-and-invert method* [138], also known as the *restricted-denominator rational Arnoldi method* [108, 110, 115] or the *resolvent Krylov subspace method* [69], where $q_m(z) = (\mu - z)^m$ with $\mu \in \mathbb{C}$ and $\mu \notin \sigma(\mathbf{S})$. Note that, for example, if $W(\mathbf{S})$ is contained in the left half-plane, then choosing μ such that $\mathrm{Re}(\mu) > 0$ ensures that $q_m(\mathbf{S})$ is not singular.

It is well known that for the shift-and-invert method, one can apply the standard Arnoldi algorithm to the matrix $(\mu\mathbf{I} - \mathbf{S})^{-1}$ in order to obtain an orthonormal basis $\mathbf{V}_m$ (see, e.g., [66, p. 45 ff.]). The corresponding Krylov relation then reads

$$(\mu\mathbf{I} - \mathbf{S})^{-1}\mathbf{V}_m = \mathbf{V}_m \mathbf{H}_m + h_{m+1,m}\mathbf{v}_{m+1}e_m^T. \tag{2.4.16}$$

An approximation of the matrix function is then calculated using (2.4.6) and replacing $\mathbf{H}_m$ by $\widehat{\mathbf{H}}_m$, which is defined by

$$\widehat{\mathbf{H}}_m := \mathbf{V}_m^H \mathbf{S} \mathbf{V}_m = \mu\mathbf{I} - \mathbf{H}_m^{-1} - h_{m+1,m}\, \mathbf{V}_m^H \mathbf{S}\mathbf{v}_{m+1}\, \mathbf{e}_m^T \mathbf{H}_m^{-1}, \tag{2.4.17}$$

cf. [69, eq.(5.6)]. A different approach, leading to a similar matrix $\widetilde{\mathbf{H}}_m$, was proposed in [138], where $\mathbf{S}$ is a symmetric negative semi-definite matrix and the function $\varphi_0(z) = e^z$ is considered. Motivated by the observation that $e^A v$ is primarily determined by eigenvalues of $\mathbf{S}$ with the smallest absolute values, we define

$$f_\mu(t) = e^{\mu - t^{-1}},$$

such that

$$f_\mu\big((\mu\mathbf{I} - \mathbf{S})^{-1}\big) = e^{\mathbf{S}}.$$

This leads to the approximation

$$e^{\mathbf{S}}\mathbf{v} \approx \|v\|\, \mathbf{V}_m\, f_\mu(\mathbf{H}_m)\mathbf{e}_1 = \|v\|\, \mathbf{V}_m\, e^{\widetilde{\mathbf{H}}_m}\mathbf{e}_1,$$

where

$$\widetilde{\mathbf{H}}_m = \mu\mathbf{I} - \mathbf{H}_m^{-1},$$

and $\mathbf{H}_m$ is the matrix from the relation (2.4.16). Comparing $\widetilde{\mathbf{H}}_m$ to (2.4.17), one can see that it is only a rank-one modification of $\widehat{\mathbf{H}}_m$ and therefore yields a similar approximation.

Rational Krylov methods require the solution of a linear system in each iteration, which makes each step significantly more expensive than in polynomial Krylov subspace methods. However, since rational functions may provide better approximation properties than polynomials, the number of iterations required in rational Krylov methods is often significantly lower than in polynomial methods, assuming that the poles have been chosen appropriately. In particular, the convergence is, in contrast to polynomial approaches (see Remark 2.4.6), independent of the size of the field-of-values, as can be seen from the following error result.

Theorem 2.4.13 (Corollary 5.1 in [69]). *Let* $\mathbf{S}$ *be a matrix satisfying*

$$W(\mathbf{S}) \subset \{z \in \mathbb{C} \mid \mathrm{Re}(z) \leq 0\}.$$

Then there exists a constant C*, depending only on* k *and* $\mu > 0$*, such that the approximation of* $\varphi_k(\mathbf{S})\mathbf{v}$ *obtained by the shift-and-invert method satisfies*

$$\left\| \varphi_k(\mathbf{S})\mathbf{v} - \mathbf{V}_m \varphi_k(\widehat{\mathbf{H}}_m)\, \mathbf{e}_1 \right\| \leq \frac{C}{m^{\frac{k}{2}}} \|\mathbf{v}\|,$$

where $\widehat{\mathbf{H}}_m$ *is defined in* (2.4.17).

A drawback of shift-and-invert methods is that their implementation is significantly more complex and their performance depends heavily on the choice of the shift μ, as well as on whether the resulting linear systems can be solved efficiently. In [66, Section 5.4], a discussion and numerical experiments regarding the choice of shift in connection with the approximation of the φ_k-functions are presented.

For more details and further rational Krylov methods, we refer to [20, 22, 67, 68, 71, 72].

2.4.4 Conclusion

Krylov subspace methods provide a powerful and computationally efficient framework for approximating the product of matrix function with a vector.

Polynomial Krylov subspace methods are characterized by their simplicity, which makes them highly scalable for large sparse systems. The Arnoldi and Lanczos algorithms provide efficient and numerically stable procedures for constructing an orthonormal basis of the Krylov subspace, while also ensuring quasi-optimal convergence properties. To reduce memory requirements, Chebyshev-based methods can be very effective when a priori

spectral information is available. These approaches are based on three-term recursions, which significantly reduce storage demands and exhibit superlinear convergence.

Rational Krylov subspace methods extend this approach by incorporating rational functions, resulting in significantly richer approximation spaces. These methods can substantially improve convergence for matrix functions, particularly when appropriate pole-selection strategies are applied and efficient solvers for systems of linear equations are available.

Bibliographical notes on Krylov approximations for matrix functions. Foundational convergence analyses for polynomial Krylov methods were carried out, for instance, by [46–48, 84, 124]. The approximation of matrix functions by rational Krylov subspace methods has also been extensively investigated, see, for example, [21, 67–69, 108, 110, 115, 138]. Beckermann and Reichel [17] proved, using Faber transformations, the near-optimal approximation of standard and rational Arnoldi processes, as well as of Chebyshev and Faber series of matrix functions. Stopping criteria and adaptive error estimation for Krylov subspace approximations of the matrix φ_k-functions have been investigated, by [27, 85, 90, 91]. Applications to systems of ordinary differential equations and partial differential equations, as well as practical implementation details and software aspects, can be found in [33, 114, 119, 136]. For further surveys on Krylov subspace approximations for matrix functions, we refer to [66, 71, 72, 87].

CHAPTER 3

An Implicit-Explicit Time Discretization Scheme for Semilinear Wave Equations

The content of the following chapter was published in the IMA Journal of Numerical Analysis, cf. [52]. The authors of this paper are Daniel Eckhardt, Marlis Hochbruck and Barbara Verfürth [1].

In this chapter we present an implicit-explicit (IMEX) scheme for semilinear wave equations with strong damping. By treating the nonlinear, nonstiff term explicitly and the linear, stiff part implicitly, we obtain a method which is not only unconditionally stable but also highly efficient. Our main results are error bounds of the full discretization in space and time for the IMEX scheme combined with a general abstract space discretization. As an application, we consider the heterogeneous multiscale method for wave equations with highly oscillating coefficients in space for which we show spatial and temporal convergence rates by using the abstract result.

[1]BV acknowledges funding by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under project number 496556642 and under Germany's Excellence Strategy – EXC-2047/1 – 390685813.

3.1 Introduction

For numerically solving partial differential equations (PDEs) in time, a variety of different methods have been investigated in the literature, broadly categorized as implicit or explicit schemes. Explicit schemes are popular since they are simple to implement, computationally efficient, but they suffer from step-size restrictions, so-called CFL conditions. On the other hand, implicit schemes involve solving a system of nonlinear equations at each time step, resulting in improved stability and allowing for larger time steps compared to explicit methods. However, they require a higher computational effort. To combine the advantages of both strategies, we construct and numerically analyze an implicit-explicit (IMEX) method in which the CFL condition only depends on non-stiff terms. Besides many applications, e.g., air pollution models [139], hydrodynamics and atmospheric dynamics [60, 92], fluid-structure interaction [117], there is a well-developed theory of IMEX-Runge Kutta (IMEX-RK) [12, 25, 95, 97, 99] and IMEX multistep methods [7, 13, 88, 89] for ordinary differential equations and parabolic problems. For the wave equation IMEX-RK [11], Crank-Nicolson/Leapfrog (CNLF) schemes [83, 98] (note that these are not equivalent) and a θ-scheme/Leapfrog combination [106] are considered. As in [83], we consider numerical solutions of semilinear wave equations in a general Hilbert space $\mathbf{H}$ of the following form

$$u'' + \mathcal{A}u + \mathcal{B}u' = \mathcal{G}(u, u') + f. \tag{3.1.1}$$

Here, $\mathcal{A}$ and $\mathcal{B}$ are linear, possibly unbounded operators, representing the wave propagation and (strong) linear damping of the system, respectively. The nonlinear operator $\mathcal{G}$ and the function f denote additional (nonlinear) source and damping terms. In [83] a rigorous error analysis for the CNLF applied to (3.1.1) was presented if $\mathcal{G}(u, u') = \mathcal{G}(u)$. However, if $\mathcal{G}$ depends on u', this scheme is not very attractive, as it leads to a system of nonlinear equations. In addition, [83] requires boundedness conditions on $\mathcal{B}$ which are not fulfilled in our application. We therefore propose a different method based on the implicit and explicit midpoint rule applied to the first-order system of (3.1.1). Our scheme leads to a system of *linear* equations also in the general case, where $\mathcal{G}$ depends on both u and u' and can be formulated for a larger class of operators $\mathcal{B}$. Even with these greater application possibilities, the computational effort of the implicit/explicit midpoint rule is about the same as for the CNLF in the case $\mathcal{G}(u, u') = \mathcal{G}(u)$ (see the discussion in Section 3.5). Although the scheme has already been considered in the literature [13], to the best of our knowledge there is no error analysis for (nonlinear) wave equations. We provide a

proof of second-order convergence, which requires significantly different arguments than in [83]. A central aspect of the proof is to consider the first-order formulation and to interpret the scheme as a perturbation of the implicit midpoint rule. This view allows us to bound the error terms of the IMEX method by terms of the implicit midpoint rule. Building on the unified error analysis from [80], our main result is a full discretization bound for the IMEX scheme combined with an abstract space discretization method.

As an application we consider the semilinear wave equation with damping

$$\partial_{tt} u^\varepsilon - \nabla \cdot (a^\varepsilon(x) \nabla u^\varepsilon) - \nabla \cdot (\beta(x) \nabla \partial_t u^\varepsilon) = \mathcal{G}(x, \partial_t u^\varepsilon) + f \tag{3.1.2}$$

in a bounded Lipschitz domain Ω and with Dirichlet boundary conditions and initial values. The parameter ε describes small scale effects, e.g., fast oscillations, which appear in the coefficient a^ε and therefore also in the solution u^ε. This in connection with Finite Elements (FE) can lead to a high computational effort, since a grid width in the order of magnitude of ε must be chosen. On the other hand, in applications we are often only interested in the macroscopic behavior of u^ε. We therefore propose a multiscale method based on the Heterogeneous Multiscale Method (HMM) introduced in [2].

The idea of the HMM is to derive a model related to (3.1.2), which is independent of ε. Missing microscopic data is collected by solving microscopic problems on small subdomains and only where the data is needed.

Here we make use of a convergence result for $\varepsilon \to 0$ (see [113]) which leads to a homogenized version of (3.1.2) with an ε-independent coefficient a^{hom}. This coefficient can be calculated explicitly if scale separation and ε-periodicity of a^ε is assumed, by solving so-called cell problems. For the space discretization we consider FE for the homogenized problem of (3.1.2), where we can now choose a much coarser mesh than if we would apply FE directly for (3.1.2), since the homogenized problem does not contain any microscopic features. We show that the HMM fits into the abstract framework and thus the error estimates can be applied, which extends the known results for the HMM discretization from linear [3, 4] to semilinear wave equations.

This chapter is organized as follows. In Section 3.2 we introduce the abstract setting, present a wellposedness result, the time integration scheme, and the error result for the semidiscrete equation. The full discretization error is investigated in Section 3.3 where we combine the IMEX scheme with an abstract space discretization. Finally, we derive the fully discrete finite element heterogeneous multiscale methode for semilinear wave equations with damping in Section 3.4 and prove convergence rates using the abstract results.

These theoretical findings are demonstrated by numerical results presented in Section 3.5.

3.2 Time Integration

In this section, we consider time-integration methods for the first-order model equation (3.1.1). We start with the general framework and the wellposedness result. Secondly, we construct an IMEX scheme motivated by the observation that the operator $\mathcal{G}$ is often non-stiff in practice.

3.2.1 Wellposedness of the continuous problem

Let $\mathbf{H}$ be a Hilbert space with inner product $(\cdot,\cdot)_{\mathbf{H}}$ and $\mathbf{V} \subset \mathbf{H}$ be a densely embedded subspace. The corresponding norm is denoted by $\|\cdot\|_{\mathbf{H}}$. We consider the variational problem

$$\begin{aligned} \big(u'', w\big)_{\mathbf{H}} + d(u,w) + b(u',w) &= (\mathcal{G}(u,u'),w)_{\mathbf{H}} + \big(f(t),w\big)_{\mathbf{H}} \\ u(0) = u^0, \qquad u'(0) &= v^0 \end{aligned} \tag{3.2.1}$$

for all $w \in \mathbf{V}$, $t \geq 0$, and impose the following conditions on d, b, $\mathcal{G}$ and f :

Assumption 3.2.1 (Forms, nonlinearity, source term)**.**

(i) *The bilinear form $d : \mathbf{V} \times \mathbf{V} \to \mathbb{R}$ is symmetric and continuous. In addition, there exist constants c_d and $C_{\mathbf{H},\mathbf{V}}$ such that*

$$(\cdot,\cdot)_{\mathbf{V}} = d(\cdot,\cdot) + c_d\,(\cdot,\cdot)_{\mathbf{H}}$$

is an inner product on $\mathbf{V}$ and the embedding $\mathbf{V} \hookrightarrow \mathbf{H}$ is continuous, i.e.,

$$\|v\|_{\mathbf{H}} \leq C_{\mathbf{H},\mathbf{V}}\|v\|_{\mathbf{V}} \qquad \text{for all } v \in \mathbf{V}.$$

(ii) *The bilinear form $b : \mathbf{V} \times \mathbf{V} \to \mathbb{R}$ is continuous and quasi-monotone, i.e., there exists a constant $c_{qm} \geq 0$ such that:*

$$b(v,v) \geq -c_{qm}\|v\|_{\mathbf{H}}^2.$$

(iii) *The nonlinearity $\mathcal{G} : \mathbf{V} \times \mathbf{H} \to \mathbf{H}$ is locally Lipschitz continuous , i.e, for all $\rho > 0$ there exists a Lipschitz constant L_ρ such that for all $v_1, v_2 \in \mathbf{V}$ and $w_1, w_2 \in \mathbf{H}$ with $\max\{\|[v_1,w_1]\|_{\mathbf{V}\times\mathbf{H}}, \|[v_2,w_2]\|_{\mathbf{V}\times\mathbf{H}}\} \leq \rho$ it holds*

$$\|\mathcal{G}(v_1,w_1) - \mathcal{G}(v_2,w_2)\|_{\mathbf{H}} \leq L_\rho(\|v_1 - v_2\|_{\mathbf{V}} + \|w_1 - w_2\|_{\mathbf{H}}),$$

(iv) *The right-hand side $f : [0,\infty) \to \mathbf{H}$ satisfies*

$$f \in W^{1,1}_{loc}([0,\infty);\mathbf{H}).$$ ○

By (i), $(\cdot,\cdot)_{\mathbf{V}}$ induces a norm $\|\cdot\|_{\mathbf{V}}$. The product space $\mathbf{V}\times\mathbf{H}$ will be equipped with the norm $\|\cdot\|_{\mathbf{V}\times\mathbf{H}} := \|\cdot\|_{\mathbf{V}} + \|\cdot\|_{\mathbf{H}}$. Moreover, we identify $\mathbf{H}$ with its dual space $\mathbf{H}^\star$, resulting in the following Gelfand triple:

$$\mathbf{V} \hookrightarrow \mathbf{H} = \mathbf{H}^\star \hookrightarrow \mathbf{V}^\star.$$

We define the operators $\mathcal{A}, \mathcal{B} \in \mathcal{L}(\mathbf{V}, \mathbf{V}^\star)$ associated to d, b :

$$\begin{aligned}\langle \mathcal{A}v, w\rangle_{\mathbf{V}^\star\times\mathbf{V}} &:= d(v,w),\\ \langle \mathcal{B}v, w\rangle_{\mathbf{V}^\star\times\mathbf{V}} &:= b(v,w).\end{aligned}$$

Problem (3.2.1) can thus be written as the following evolution equation

$$\begin{aligned} u'' + \mathcal{A}u + \mathcal{B}u' &= \mathcal{G}(u,u') + f \\ u(0) = u^0 \quad &, u'(0) = v^0. \end{aligned} \tag{3.2.2}$$

3.2.2 First-order formulation

In order to derive a first-order formulation of (3.2.2), we set: $u' = v$,

$$x = \begin{bmatrix} u \\ v \end{bmatrix}, \quad \mathcal{S}x = \begin{bmatrix} v \\ -\mathcal{A}u - \mathcal{B}v \end{bmatrix}, \quad F(x) = \begin{bmatrix} 0 \\ \mathcal{G}(u,v) + f \end{bmatrix}, \quad x^0 = \begin{bmatrix} u^0 \\ v^0 \end{bmatrix}.$$

Then (3.2.2) reads as follows:
Seek $x \in \mathbf{Y} := \mathbf{V}\times\mathbf{V}$, such that

$$\begin{aligned} x' - \mathcal{S}x &= F(x) \\ x(0) &= x^0. \end{aligned} \tag{3.2.3}$$

To obtain an evolution equation in $\mathbf{X} := \mathbf{V}\times\mathbf{H}$ we consider the following restriction of the operator S:

$$\begin{aligned} S : D(S) \to \mathbf{X}, \qquad y \to Sy = \mathcal{S}y \qquad \text{on } D(S) &= \{y \in \mathbf{Y} \mid \mathcal{S}y \in \mathbf{X}\} \\ &= \{(u,v) \in \mathbf{V}\times\mathbf{V} \mid \mathcal{A}u + \mathcal{B}v \in \mathbf{H}\}. \end{aligned}$$

Lemma 3.2.2. *The operator $S : D(S) \to \mathbf{X}$ is the generator of C_0-semigroup, which satisfies*

$$\| \mathrm{e}^{tS} \|_{\mathbf{X}\leftarrow\mathbf{X}} \le \mathrm{e}^{c_{qm}^S t}$$

with $c_{qm}^S := \frac{1}{2} c_d C_{\mathbf{H},\mathbf{V}} + c_{qm}$.

Proof. This follows by a combination of Lemma 4.2 (with $\alpha = 1$), Lemma 2.3 and Theorem 2.4 in [80] and [116, Section 1.3. and 4.2]. □

The well-posedness result can be found in [82, Lemma 3.2].

Corollary 3.2.3. *Let Assumption 3.2.1 be satisfied and u^0, $v^0 \in \mathbf{V}$ with $\mathcal{A}u^0 + \mathcal{B}v^0 \in \mathbf{H}$. Then there exist a maximal existence time $t^\star(u^0, v^0) \in (0, \infty]$ such that for all $T < t^\star(u^0, v^0)$, (3.2.2) has a unique solution $u \in W^{2,\infty}([0,T];\mathbf{H}) \cap C^1([0,T],\mathbf{V})$ which satisfies $\mathcal{A}u + \mathcal{B}u' \in C([0,T];\mathbf{H})$.*

3.2.3 The IMEX scheme

The implicit midpoint scheme applied to (3.2.3) can be written in a half step formulation as

$$x^{n+\frac{1}{2}} = x^n + \frac{\tau}{2} S x^{n+\frac{1}{2}} + \frac{\tau}{2} F(x^{n+\frac{1}{2}}), \tag{3.2.4a}$$

$$x^{n+1} = x^n + \tau S x^{n+\frac{1}{2}} + \tau F(x^{n+\frac{1}{2}}), \tag{3.2.4b}$$

while explicit midpoint rule uses

$$x^{n+\frac{1}{2}} = x^n + \frac{\tau}{2} S x^n + \frac{\tau}{2} F(x^n) \tag{3.2.5}$$

instead of (3.2.4a). Combining (3.2.4a) and (3.2.5) leads to the following linearly implicit scheme:

$$x^{n+\frac{1}{2}} = x^n + \frac{\tau}{2} S x^{n+\frac{1}{2}} + \frac{\tau}{2} F(x^n), \tag{3.2.6a}$$

$$x^{n+1} = x^n + \tau S x^{n+\frac{1}{2}} + \tau F(x^{n+\frac{1}{2}}). \tag{3.2.6b}$$

Hence the S operator is treated implicitly and the nonlinear operator F explicitly. This results in second-order error bounds for the IMEX scheme.

Theorem 3.2.4. *Let $u \in C^4([0,T];\mathbf{H}) \cap C^3([0,T];\mathbf{V})$ be the solution of (3.2.2) with $\mathcal{A}u + \mathcal{B}u' \in C^2([0,T];\mathbf{H})$ and $x^n = [u^n, v^n]^\intercal$, $t_n = \tau n \in [0,T]$, be the approximations obtained by the IMEX scheme (3.2.6). For*

$$\rho = 2(\|u\|_{L^\infty([0,T];\mathbf{V})} + \|u'\|_{L^\infty([0,T];\mathbf{H})}),$$

there exists $\tau^\star > 0$ such that for all $\tau < \tau^\star$ and all $t_n \in [0,T]$

$$\max_{t_n \le T} \|u^n\|_{\mathbf{V}} \le \rho \quad \text{and} \quad \max_{t_n \le T} \|v^n\|_{\mathbf{H}} \le \rho.$$

Furthermore, the error bound

$$\|u^n - u(t_n)\|_{\mathbf{V}} + \|v^n - u'(t_n)\|_{\mathbf{H}} \le C t_n \, \mathrm{e}^{\left(c_{qm}^S + L_\rho(2+\frac{\tau}{2}L_\rho)\right)t_n} \, \tau^2$$

holds true with a constant C which depends on derivatives of u and L_ρ, T but is independent of n and τ.

Since the proof works similarly as the more complicated proof of Theorem 3.3.6 of the full discretization, we omit it here, cf. Remark 3.3.13 for more details.

3.3 Full discretization

In this section we analyze the full discretization of (3.2.2) using the IMEX scheme (3.2.6) as time stepping method and an abstract space discretization. We first recall some results from [82].

3.3.1 Space discretization

Let $(\mathbf{H}_H)_H \subset \mathbf{V}$ be a family of finite-dimensional spaces, e.g., a finite element space. For these subspaces we can formulate a discrete version of (3.2.2) as

$$\begin{aligned}(u_H'', w_H)_{\mathbf{H}_H} + d_H(u_H, w_H) + b_H(u_H', w_H) \\ + (\mathcal{G}_H(u_H, u_H'), w_H)_{\mathbf{H}_H} = (f_H(t), w_H)_{\mathbf{H}_H}\end{aligned} \tag{3.3.1}$$

for $t \geq 0$, $w_H \in \mathbf{H}_H$ with $u_H(0) = u_H^0, v_H(0) = v_H^0$. Here, $(\cdot,\cdot)_{\mathbf{H}_H}, d_H, b_H, \mathcal{G}_H, u_H^0, v_H^0$ are approximations to $(\cdot,\cdot)_{\mathbf{H}}, d, b, \mathcal{G}, u^0, v^0$. Similar to the continuous case in Assumption 3.2.1, we assume the following conditions:

Assumption 3.3.1 (Discrete setting)**.**

(i) *The bilinear form $d_H : \mathbf{H}_H \times \mathbf{H}_H \to \mathbb{R}$ is symmetric and continuous, and*

$$(\cdot,\cdot)_{\mathbf{V}_H} = d_H(\cdot,\cdot) + c_d(\cdot,\cdot)_{\mathbf{H}_H}$$

is a scalar product on $\mathbf{H}_H$. The approximation space equipped with the scalar product $(\cdot,\cdot)_{\mathbf{V}_H}$ will be denoted by $\mathbf{V}_H$ with induced norm $\|\cdot\|_{\mathbf{V}_H}$. In addition, we assume that there is a constant $C_{\mathbf{H}_H,\mathbf{V}_H} > 0$ such that

$$\|v_H\|_{\mathbf{H}_H} \leq C_{\mathbf{H}_H,\mathbf{V}_H}\|v_H\|_{\mathbf{V}_H} \quad \text{for all } v_H \in \mathbf{V}_H.$$

(ii) *The bilinear form $b_H : \mathbf{V}_H \times \mathbf{V}_H \to \mathbb{R}$ is continuous and quasi-monotone w.r.t. the $\|\cdot\|_{\mathbf{H}_H}$-norm with constant $\widehat{c}_{qm}$.*

(iii) *The nonlinearity $\mathcal{G}_H : \mathbf{V}_H \times \mathbf{H}_H \to \mathbf{H}_H$ is locally Lipschitz continuous with Lipschitz constant L_{ρ_H} and w.r.t. the $\|\cdot\|_{\mathbf{V}_H \times \mathbf{H}_H}$-norm.*

(iv) *The right-hand side $f_H : [0,\infty) \to \mathbf{H}_H$ satisfies*

$$f_H \in W^{1,1}_{loc}([0,\infty); \mathbf{H}_H).$$ ○

In (i), we take the Garding constant c_d from the Assumptions 3.2.1. Note that a different constant could also be chosen. Then c_d must be set as the maximum of the constants from the Assumptions 3.2.1 and the Assumptions 3.3.1 so that the same constants are used for $(\cdot,\cdot)_{\mathbf{V}}$ and $(\cdot,\cdot)_{\mathbf{V}_H}$.

The discrete associated operators $\mathcal{A}_H, \mathcal{B}_H \in \mathcal{L}(\mathbf{V}_H, \mathbf{V}_H)$ are defined via

$$(\mathcal{A}_H v, w)_{\mathbf{H}_H} := d_H(v, w),$$
$$(\mathcal{B}_H v, w)_{\mathbf{H}_H} := b_H(v, w).$$

This yields the following evolution equation, which is the discrete counterpart of (3.2.2):

$$\begin{aligned} u_H'' + \mathcal{A}_H u_H + \mathcal{B}_H u_H' &= \mathcal{G}_H(u_H, u_H') + f_H \qquad \text{in } \mathbf{V}_H, \\ u_H(0) = u_H^0 \quad &, u_H'(0) = v_H^0. \end{aligned} \tag{3.3.2}$$

To derive error bounds for the abstract spatial discretization, we need interpolation and projection operators for which the following assumptions hold.

Assumption 3.3.2 (Operators)**.**

(i) Projection operator: There exist linear operators $\mathcal{P}_H^{\mathbf{V}} : \mathbf{V} \to \mathbf{V}_H$ and $\mathcal{P}_H^{\mathbf{H}} : \mathbf{H} \to \mathbf{H}_H$ such that:

$$(\mathcal{P}_H^{\mathbf{H}} v, w_H)_{\mathbf{H}_H} := (v, w_H)_{\mathbf{H}} \quad \text{for all } v \in \mathbf{H}, w_H \in \mathbf{H}_H$$
$$(\mathcal{P}_H^{\mathbf{V}} v, w_H)_{\mathbf{V}_H} := (v, w_H)_{\mathbf{V}} \quad \text{for all } v \in \mathbf{V}, w_H \in \mathbf{V}_H$$

(ii) Interpolation operator: There exists an interpolation operator $\mathcal{I}_H \in \mathcal{L}(\mathbf{Z}^{\mathbf{V}}, \mathbf{V}_H)$ defined on a dense subspace $(\mathbf{Z}^{\mathbf{V}}, \|\cdot\|_{\mathbf{Z}^{\mathbf{V}}})$ of $\mathbf{V}$ with

$$\|\mathcal{I}_H\|_{\mathbf{V}_H \leftarrow \mathbf{Z}^{\mathbf{V}}} \leq C_{\mathbf{Z}^{\mathbf{V}}}.$$

(iii) Embedding: There exist constants $C_{\mathbf{H}}, C_{\mathbf{V}}$ such that

$$\|v_H\|_{\mathbf{H}} \leq C_{\mathbf{H}} \|v_H\|_{\mathbf{H}_H}, \qquad \|v_H\|_{\mathbf{V}} \leq C_{\mathbf{V}} \|v_H\|_{\mathbf{V}_H}$$

for all $v_H \in \mathbf{V}_H$. ○

Remark 3.3.3. *In [82] lift operators are introduced that are not needed here, as we assume that $(\mathbf{H}_H)_H \subset \mathbf{V}$.*

In addition, we have to measure the error between the scalar products $(\cdot,\cdot)_{\mathbf{V}}$, $(\cdot,\cdot)_{\mathbf{H}}$ and the corresponding discrete counterparts.

Definition 3.3.4. For $v_H, w_H \in \mathbf{V}_H$ we define the following errors by the differences of the scalar product:

$$(v_H, w_H)_{\Delta\mathbf{H}} := (v_H, w_H)_{\mathbf{H}} - (v_H, w_H)_{\mathbf{H}_H}$$
$$(v_H, w_H)_{\Delta\mathbf{V}} := (v_H, w_H)_{\mathbf{V}} - (v_H, w_H)_{\mathbf{V}_H}.$$

Analogously to the continuous case, we now want to write the discrete equation (3.3.2) in first-order formulation. This allows us to transfer the error result from [82].
For the semidiscrete equation we set: $u_H' = v_H$

$$x_H = \begin{bmatrix} u_H \\ v_H \end{bmatrix}, \quad S_H x_H = \begin{bmatrix} v_H \\ -\mathcal{A}_H u_H - \mathcal{B}_H v_H \end{bmatrix}, \quad F_H(x_H) = \begin{bmatrix} 0 \\ \mathcal{G}_H(u_H, v_H) + f_H \end{bmatrix},$$

and

$$x_H^0 = \begin{bmatrix} u_H^0 \\ v_H^0 \end{bmatrix}.$$

We can write the discrete equation as follows: Seek $x_H \in \mathbf{X}_H := \mathbf{V}_H \times \mathbf{H}_H$, such that

$$\begin{aligned} x_H' - S_H x_H &= F_H(x_H) \\ x_H(0) &= x_H^0. \end{aligned} \tag{3.3.3}$$

Local well-posedness for the first-order formulation (3.3.3) and therefore local well-posedness for (3.3.2) follows from the Picard-Lindelöf theorem with maximal existence time of the solution $t_H^\star(x_H^0)$.
Since the assumptions in the discrete case are similar to those in the continuous case, a discrete version of Lemma 3.2.2 holds true with constant

$$c_{qm}^{S_H} := \frac{1}{2} c_d C_{\mathbf{H}_H, \mathbf{V}_H} + \widehat{c}_{qm}. \tag{3.3.4}$$

Then, using the operators from Assumption 3.3.2, the operators for the first-order formulation (3.3.3) are given by the following definition.

Definition 3.3.5. We set $\mathbf{Z} = \mathbf{V} \times \mathbf{Z}^{\mathbf{V}}$ and define the first-order reference operator $\mathcal{J}_H : \mathbf{Z} \to \mathbf{X}_H$ and first-order projection operator $\mathcal{P}_H : \mathbf{X} \to \mathbf{X}_H$ by

$$\mathcal{J}_H \begin{bmatrix} v \\ w \end{bmatrix} := \begin{bmatrix} \mathcal{P}_H^{\mathbf{V}} v \\ \mathcal{I}_H w \end{bmatrix}, \qquad \mathcal{P}_H \begin{bmatrix} v \\ w \end{bmatrix} := \begin{bmatrix} \mathcal{P}_H^{\mathbf{V}} v \\ \mathcal{P}_H^{\mathbf{H}} w \end{bmatrix}.$$

By Assumption 3.3.2 we have

$$\|\mathcal{J}_H\|_{\mathbf{X}_H \leftarrow \mathbf{Z}} \leq C_{\mathcal{J}_H} := C_{\mathbf{V}} + C_{\mathbf{Z}^{\mathbf{V}}}, \qquad \|\mathcal{P}_H\|_{\mathbf{X}_H \leftarrow \mathbf{X}} \leq C_{\mathbf{X}} := C_{\mathbf{V}} + C_{\mathbf{H}}.$$

The reference operator $\mathcal{J}_H$ is needed to obtain optimal error bounds w.r.t H.

3.3.2 Error bounds for the full discretization

For the fully discrete scheme, we apply the IMEX scheme to (3.3.3). This yields

$$x_H^{n+\frac{1}{2}} = x_H^n + \frac{\tau}{2} S_H x_H^{n+\frac{1}{2}} + \frac{\tau}{2} F_H(x_H^n), \tag{3.3.5a}$$

$$x_H^{n+1} = x_H^n + \tau S_H x_H^{n+\frac{1}{2}} + \tau F_H(x_H^{n+\frac{1}{2}}). \tag{3.3.5b}$$

We now present the main result, i.e., error bounds for the fully discrete IMEX scheme.

Theorem 3.3.6. *Let $u \in C^4([0,T];\mathbf{H}) \cap C^3([0,T];\mathbf{V})$ be the solution of (3.2.2) with $u, u', u'' \in L^\infty([0,T];\mathbf{Z}^\mathbf{V})$ and $\mathcal{A}u + \mathcal{B}u' \in C^2([0,T];\mathbf{H})$. Further, let $x_H^n = [u_H^n\, v_H^n]^\intercal$ be the approximations obtained by the fully discrete IMEX scheme (3.2.6) at time $t_n \in [0,T]$. We set*

$$\rho_H := \max\Big\{(C_\mathbf{V} + C_{\mathbf{Z}^\mathbf{V}})\|[u,u']\|_{L^\infty([0,T];\mathbf{V}\times\mathbf{Z}^\mathbf{V})}, \max_{t_n \le T} \|x_H^n\|_{\mathbf{V}_H \times \mathbf{H}_H}\Big\}$$

and assume that τ satisfies the step size restriction

$$\tau < \frac{2}{c_{qm}^{S_H}}. \tag{3.3.6}$$

Then, the error bound

$$\|u_H^n - u(t_n)\|_\mathbf{V} + \|v_H^n - u'(t_n)\|_\mathbf{H} \le C t_n \, \mathrm{e}^{M t_n}(\tau^2 + E_H) \tag{3.3.7}$$

holds true with $M := c_{qm}^{S_H} + L_{\rho_H}(2 + \frac{\tau}{2} L_{\rho_H})$ and a constant C which depends on derivatives of u, the constants of section 3.2 , 3.3 and L_{ρ_H} but is independent of n and τ. Here,

$$E_H = E_H(u,f) = E_{H,1} + E_{H,2} + E_{H,3} + E_{H,4} \tag{3.3.8a}$$

contains the abstract space discretization error terms $E_{H,i} = E_{H,i}(u,f)$ given by

$$\begin{aligned}
E_{H,1} &= \|u_H^0 - \mathcal{P}_H^\mathbf{V} u^0\|_{\mathbf{V}_H} + \|v_H^0 - \mathcal{I}_H v^0\|_{\mathbf{H}_H} + \|\mathcal{P}_H^\mathbf{H} f - f_H\|_{L^\infty([0,T];\mathbf{H}_H)} \\
E_{H,2} &= \max_{\|\phi_H\|_{\mathbf{V}_H}=1} \|(\mathcal{I}_H u, \phi_H)_{\Delta\mathbf{V}}\|_{L^\infty[0,T]} + \max_{\|\psi_H\|_{\mathbf{H}_H}=1} \|(\mathcal{I}_H u, \psi_H)_{\Delta\mathbf{H}}\|_{L^\infty[0,T]} \\
&\quad + \max_{\|\phi_H\|_{\mathbf{V}_H}=1} \|(\mathcal{I}_H u', \phi_H)_{\Delta\mathbf{V}}\|_{L^\infty[0,T]} + \max_{\|\psi_H\|_{\mathbf{H}_H}=1} \|(\mathcal{I}_H u'', \psi_H)_{\Delta\mathbf{H}}\|_{L^\infty[0,T]} \\
E_{H,3} &= \|(\mathrm{I} - \mathcal{I}_H)u\|_{L^\infty([0,T];\mathbf{V})} + \|(\mathrm{I} - \mathcal{I}_H)u'\|_{L^\infty([0,T];\mathbf{V})} + \|(\mathrm{I} - \mathcal{I}_H)u''\|_{L^\infty([0,T];\mathbf{H})} \\
E_{H,4} &= \|\mathcal{P}_H^\mathbf{H}\mathcal{G}(u,u') - \mathcal{G}_H(\mathcal{P}_H^\mathbf{V} u, \mathcal{I}_H u')\|_{L^\infty([0,T];\mathbf{H}_H)} \\
&\quad + \|\mathcal{P}_H^\mathbf{H}\mathcal{B}u' - \mathcal{B}_H\mathcal{I}_H u'\|_{L^\infty([0,T];\mathbf{H}_H)}.
\end{aligned} \tag{3.3.8b}$$

Remark 3.3.7. *The constant ρ_H depends on the discrete solution. Under additional assumptions, however, this can be replaced by a constant independent of x_H^n. See Theorem 3.3.12 for more details.*

In order to prove the theorem, we first define the remainder term, which is an essential part of the error.

Definition 3.3.8. In the following let $z = [u, v]^T \in \mathbf{X}$.

(i) The remainder of the linear monotone operator is given by

$$R_H : \mathbf{Z} \cap D(S) \to \mathbf{X}_H,$$
$$R_H z := \mathcal{P}_H S z - S_H \mathcal{J}_H z = \begin{bmatrix} \mathcal{P}_H^{\mathbf{V}} v - \mathcal{I}_H v \\ \mathcal{P}_H^{\mathbf{H}}(\mathcal{A}u + \mathcal{B}v) - (\mathcal{A}_H \mathcal{P}_H^{\mathbf{V}} u + \mathcal{B}_H \mathcal{I}_H v) \end{bmatrix}.$$

(ii) The remainder of the Lipschitz continuous nonlinearity is given by

$$r_H(z) : \mathbf{Z} \to \mathbf{X}_H : \qquad r_H(z) := \mathcal{P}_H F(z) - F_H(\mathcal{J}_H(z))$$
$$= \begin{bmatrix} 0 \\ \mathcal{P}_H^{\mathbf{H}} \mathcal{G}(u, v) - \mathcal{G}_H(\mathcal{P}_H^{\mathbf{V}} u, \mathcal{I}_H v) + \mathcal{P}_H^{\mathbf{H}} f - f_H \end{bmatrix}.$$

The next lemma provides an error bound for the remainder term. The proof can be found in [80, Lemma 4.7]. Note that the reference uses additional lift operators Q_h^V, Q_h^H, which are here the identity.

Lemma 3.3.9. *For $z(t) = [u(t), v(t)]^T \in (\mathbf{Z}^{\mathbf{V}} \times \mathbf{Z}^{\mathbf{V}}) \cap D(S)$ we have*

$$\begin{aligned} \|R_H z(t)\|_{\mathbf{X}_H} \le C_R \Big(& \max_{\|\phi_H\|_{\mathbf{V}_H}=1} \|(\mathcal{I}_H u, \phi_H)_{\Delta\mathbf{V}}\|_{L^\infty[0,T]} + \max_{\|\psi_H\|_{\mathbf{H}_H}=1} \|(\mathcal{I}_H u, \psi_H)_{\Delta\mathbf{H}}\|_{L^\infty[0,T]} \\ & + \max_{\|\phi_H\|_{\mathbf{V}_H}=1} \|(\mathcal{I}_H v, \phi_H)_{\Delta\mathbf{V}}\|_{L^\infty[0,T]} + \|\mathcal{P}_H^{\mathbf{H}} \mathcal{B} v - \mathcal{B}_H \mathcal{I}_H v\|_{L^\infty([0,T];\mathbf{H}_H)} \\ & + \|(\mathrm{I} - \mathcal{I}_H) u\|_{L^\infty([0,T];\mathbf{V})} + \|(\mathrm{I} - \mathcal{I}_H) v\|_{L^\infty([0,T];\mathbf{V})} \Big), \end{aligned}$$

for all $t \in [0, T]$, where C_R only depends on constants, which appear in Section 3.2.1 and Assumptions 3.3.1, 3.3.2.

In order to provide error bounds, we start by defining defects $D_{\mathrm{IMEX},H}^{n+\frac{1}{2}}$ and $\delta_{\mathrm{IMEX},H}^{n+1}$ of the IMEX scheme (3.3.5) via

$$\tilde{x}_H^{n+\frac{1}{2}} = \tilde{x}_H^n + \frac{\tau}{2} S_H \tilde{x}_H^{n+\frac{1}{2}} + \frac{\tau}{2} \widetilde{F}_H^n + \frac{\tau}{2} D_{\mathrm{IMEX},H}^{n+\frac{1}{2}}, \tag{3.3.9a}$$
$$\tilde{x}_H^{n+1} = \tilde{x}_H^n + \tau S_H \tilde{x}_H^{n+\frac{1}{2}} + \tau \widetilde{F}_H^{n+\frac{1}{2}} + \tau \delta_{\mathrm{IMEX},H}^{n+1}. \tag{3.3.9b}$$

Here, we used the short notation

$$\tilde{x}^n = x(t_n), \qquad \tilde{x}_H^n = \mathcal{J}_H \tilde{x}^n, \qquad \widetilde{F}_H^n = F_H(\tilde{x}_H^n). \tag{3.3.10}$$

Bounds on the defects are given in the following Lemma in terms of the constants

$$M_j = \|x^{(j)}\|_{L^\infty([0,T];\mathbf{X})}, \qquad M_S = \|\mathcal{S} x''\|_{L^\infty([0,T];\mathbf{X})}, \tag{3.3.11}$$

and E_H defined in (3.3.8a).

Lemma 3.3.10. *Let Assumptions of Theorem 3.3.6 hold true. Then the defects defined in (3.3.9) satisfy*

$$\frac{\tau}{2}\|D_{IMEX,H}^{n+\frac{1}{2}}\|_{\mathbf{X}_H} \le C_{\mathbf{X}}\big(M_2 + L_{\rho_H} M_1\big)\frac{\tau^2}{4} + \frac{\tau}{2} C_H E_H, \tag{3.3.12a}$$

$$\tau\|\delta_{IMEX,H}^{n+1}\|_{\mathbf{X}_H} \le C_{\mathbf{X}} M_3 \frac{\tau^3}{24} + \tau C_H E_H, \tag{3.3.12b}$$

$$\frac{\tau}{2}\|\Delta D_{IMEX,H}^{n+\frac{1}{2}}\|_{\mathbf{X}_H} \le C_{\mathbf{X}}(M_3 + M_S)\frac{\tau^3}{2} + \tau C_H E_H, \tag{3.3.12c}$$

where $\Delta D_{IMEX,H}^{n+\frac{1}{2}} = D_{IMEX,H}^{n+\frac{1}{2}} - D_{IMEX,H}^{n-\frac{1}{2}}$ *and* $C_H = \max\{1, C_{\mathbf{H}}, C_R\}$.

A key idea of the proof is to interpret the IMEX scheme (3.3.5) as a perturbation of the implicit midpoint rule.

Proof. (a) Set $\widetilde{F}^n = F\big(x(t_n)\big)$ and we get

$$\begin{aligned}
\frac{\tau}{2} D_{\mathrm{IMEX},H}^{n+\frac{1}{2}} &= \tilde{x}_H^{\,n+\frac{1}{2}} - \tilde{x}_H^n - \frac{\tau}{2}\big(S_H \tilde{x}_H^{\,n+\frac{1}{2}} + \widetilde{F}_H^n\big) \\
&= \mathcal{P}_H\big(\tilde{x}^{\,n+\frac{1}{2}} - \tilde{x}^n - \frac{\tau}{2}\big(S\tilde{x}^{\,n+\frac{1}{2}} + \widetilde{F}^n\big)\big) + (\mathcal{J}_H - \mathcal{P}_H)(\tilde{x}^{n+\frac{1}{2}} - \tilde{x}^n) \\
&\quad + \frac{\tau}{2}(\mathcal{P}_H S\tilde{x}^{\,n+\frac{1}{2}} - S_H\tilde{x}^{\,n+\frac{1}{2}}) + \frac{\tau}{2}(\mathcal{P}_H\widetilde{F}^n - \widetilde{F}_H^n) \\
&= \frac{\tau}{2}\mathcal{P}_H\big(D_{\mathrm{MP}}^{n+\frac{1}{2}} + \widetilde{F}^{n+\frac{1}{2}} - \widetilde{F}^n\big) + \frac{\tau}{2} d_H^{n+\frac{1}{2}},
\end{aligned} \tag{3.3.13}$$

where

$$\frac{\tau}{2} D_{\mathrm{MP}}^{n+\frac{1}{2}} = \tilde{x}^{\,n+\frac{1}{2}} - \tilde{x}^n - \frac{\tau}{2}(S\tilde{x}^{\,n+\frac{1}{2}} + \widetilde{F}^n) = \int_{t_n}^{t_{n+\frac{1}{2}}} x'(s)\,\mathrm{d}s - \frac{\tau}{2}x'(t_{n+\frac{1}{2}})$$

is the projected defect arising in the first stage of the implicit midpoint rule (3.2.4a) applied to (3.2.3) and

$$\begin{aligned}
\frac{\tau}{2} d_H^{n+\frac{1}{2}} &= (\mathcal{J}_H - \mathcal{P}_H)(\tilde{x}^{n+\frac{1}{2}} - \tilde{x}^n) + \frac{\tau}{2}\big(R_H\tilde{x}^{n+\frac{1}{2}} + r_H(\tilde{x}^n)\big) \\
&= \frac{\tau}{2}\Big((\mathcal{J}_H - \mathcal{P}_H)\int_0^1 x'(t_n + \frac{\tau}{2}s)\,\mathrm{d}s + R_H\tilde{x}^{n+\frac{1}{2}} + r_H(\tilde{x}^n)\Big).
\end{aligned} \tag{3.3.14}$$

Since $\frac{\tau}{2} D_{\mathrm{MP}}^{n+\frac{1}{2}}$ is the error of the rectangular rule applied to x' on $[t_n, t_{n+\frac{1}{2}}]$, we can bound it by

$$\frac{\tau}{2}\|D_{\mathrm{MP}}^{n+\frac{1}{2}}\|_{\mathbf{X}} \le \frac{\tau^2}{8}\|x''\|_{L^\infty([0,T];\mathbf{X})}. \tag{3.3.15}$$

Thus, (3.3.13) and the Lipschitz continuity (Assumption 3.2.1) yield

$$\frac{\tau}{2}\|D_{\mathrm{IMEX},H}^{n+\frac{1}{2}}\|_{\mathbf{X}_H} \le C_{\mathbf{X}}\frac{\tau^2}{4}\big(\|x''\|_{L^\infty([0,T];\mathbf{X})} + L_{\rho_H}\|x'\|_{L^\infty([0,T];\mathbf{X})}\big) + \frac{\tau}{2}\|d_H^{n+\frac{1}{2}}\|_{\mathbf{X}_H},$$

Moreover, by Lemma 3.3.9 and [80, Lemma 2.11] we get

$$\begin{aligned}\|d_H^{n+\frac{1}{2}}\|_{\mathbf{X}_H} &\le \frac{\tau}{2}\Big\| \int_0^1 (\mathcal{J}_H - \mathcal{P}_H)x'(t_n + \frac{\tau}{2}s)\ \mathrm{d}s + R_H(\tilde{x}^{n+\frac{1}{2}}) + r_H(\tilde{x}^n)\Big\|_{\mathbf{X}_H} \\ &\le \frac{\tau}{2}\Big(C_{\mathbf{H}}\|(\mathbf{I}-\mathcal{I}_H)u''\|_{L^\infty([0,T];\mathbf{H})} + \max_{\|\psi_H\|_{\mathbf{H}_H}=1} \|(\mathcal{I}_H u'', \psi_H)_{\mathbf{\Delta H}}\|_{L^\infty[0,T]} \\ &\qquad + \|R_H(\tilde{x}^{n+\frac{1}{2}})\|_{\mathbf{X}_H} + \|\mathcal{P}_H^{\mathbf{H}}\mathcal{G}(u,u') - \mathcal{G}_H(\mathcal{P}_H^{\mathbf{V}}u, \mathcal{I}_H u')\|_{L^\infty([0,T];\mathbf{H}_H)}\Big) \\ &\le \frac{\tau}{2}\max\{1, C_{\mathbf{H}}, C_R\}E_H.\end{aligned} \tag{3.3.16}$$

(b) Next we bound the defect $\delta^i_{\mathrm{IMEX},H}$ defined in (3.3.9b). We obtain

$$\begin{aligned}\tau\delta^{n+1}_{\mathrm{IMEX},H} &= \tilde{x}_H^{n+1} - \tilde{x}_H^n - \frac{\tau}{2}\big(S_H\tilde{x}_H^{n+\frac{1}{2}} + \widetilde{F}_H^{n+\frac{1}{2}}\big) \\ &= \tau\mathcal{P}_H\delta^{n+1}_{\mathrm{MP}} + \tau\delta_H^{n+1},\end{aligned} \tag{3.3.17}$$

where

$$\tau\delta^{n+1}_{\mathrm{MP}} = \tilde{x}^{n+1} - \tilde{x}^n - \tau(\mathcal{S}\tilde{x}^{n+\frac{1}{2}} + \widetilde{F}^{n+\frac{1}{2}}) = \int_{t_n}^{t_{n+1}} x'(s)\ \mathrm{d}s - \tau x'(t_{n+\frac{1}{2}})$$

is the projected defect arising in the second stage of the implicit midpoint rule (3.2.4b) applied to (3.2.3) and

$$\tau\delta_H^{n+1} = (\mathcal{J}_H - \mathcal{P}_H)(\tilde{x}^{n+1} - \tilde{x}^n) + \tau\big(R_H(\tilde{x}^{n+\frac{1}{2}}) + r_H(\tilde{x}^{n+\frac{1}{2}})\big).$$

Analogously to (3.3.16) we conclude

$$\tau\|\delta_H^{n+1}\|_{\mathbf{X}_H} \le \tau\max\{1, C_{\mathbf{H}}, C_R\}E_H. \tag{3.3.18}$$

The quadrature error of the midpoint rule satisfies

$$\tau\|\delta^{n+1}_{\mathrm{MP}}\|_{\mathbf{X}} \le \frac{\tau^3}{24}\|x'''\|_{L^\infty([0,T];\mathbf{X})}.$$

This, (3.3.17), and (3.3.18) yield

$$\tau\|\delta^{n+\frac{1}{2}}_{\mathrm{IMEX},H}\|_{\mathbf{X}_H} \le C_{\mathbf{X}}\frac{\tau^3}{24}\|x'''\|_{L^\infty([0,T];\mathbf{X})} + \tau\max\{1, C_{\mathbf{H}}, C_R\}E_H, \tag{3.3.19}$$

(c) Finally, we consider $\Delta D^{n+\frac{1}{2}}_{\mathrm{IMEX},H}$. From (3.3.13) we get

$$\begin{aligned}\frac{\tau}{2}\Delta D^{n+\frac{1}{2}}_{\mathrm{IMEX},H} = \frac{\tau}{2}\mathcal{P}_H(\Delta D^{n+\frac{1}{2}}_{\mathrm{MP}} + (\widetilde{F}^{n+\frac{1}{2}} - \widetilde{F}^n) - (\widetilde{F}^{n-\frac{1}{2}} - \widetilde{F}^{n-1})) \\ + (d_H^{n+\frac{1}{2}} - d_H^{n-\frac{1}{2}}).\end{aligned} \tag{3.3.20}$$

Analogously to (3.3.15), the defect $\Delta D^{n+\frac{1}{2}}_{\mathrm{MP}}$ is bounded by

$$\frac{\tau}{2}\|\Delta D^{n+\frac{1}{2}}_{\mathrm{MP}}\|_{\mathbf{X}} \le \frac{\tau^3}{8}\|x'''\|_{L^\infty([0,T];\mathbf{X})}. \tag{3.3.21}$$

Using (3.2.3) and Taylor expansion, one can easily verify that

$$\frac{\tau}{2}\|(\widetilde{F}^{n+\frac{1}{2}}-\widetilde{F}^{n})-(\widetilde{F}^{n-\frac{1}{2}}-\widetilde{F}^{n-1})\|_{\mathbf{X}} \leq \left(\frac{\tau}{2}\right)^2 \tau\Big(\|x'''\|_{L^\infty([0,T];\mathbf{X})}+\|\mathcal{S}x''\|_{L^\infty([0,T];\mathbf{X})}\Big).$$

From this, the space error estimate (3.3.16) and the relation (3.3.20), it follows that:

$$\begin{aligned}\frac{\tau}{2}\|\Delta D^{i+\frac{1}{2}}_{\mathrm{IMEX},H}\|_{\mathbf{X}_H} &\leq C_{\mathbf{X}}\frac{\tau^3}{8}\big(3\|x'''\|_{L^\infty([0,T];\mathbf{X})}+2\|\mathcal{S}x''\|_{L^\infty([0,T];\mathbf{X})}\big)+\frac{\tau}{2}\big(\|d_H^{n+\frac{1}{2}}\|_{\mathbf{X}_H}\\ &\qquad +\|d_H^{n-\frac{1}{2}}\|_{\mathbf{X}_H}\big)\\ &\leq C_{\mathbf{X}}\frac{\tau^3}{2}(\|x'''\|_{L^\infty([0,T];\mathbf{X})}+\|\mathcal{S}x''\|_{L^\infty([0,T];\mathbf{X})})+\tau\max\{1,C_{\mathbf{H}},C_R\}E_H\end{aligned}$$

This completes the proof. □

Next we introduce the operators

$$\widehat{\mathcal{R}}_{\pm} := \mathbf{I} \pm \frac{\tau}{2}S_H : \mathbf{X}_H \to \mathbf{X}_H. \tag{3.3.22}$$

The following properties of $\widehat{\mathcal{R}}_{\pm}$ can be shown similarly to the continuous case [83, Lemma 2.4]:

Lemma 3.3.11. *Let $c^{S_H}_{qm}$ be as defined in (3.3.4). Then, if $\tau c^{S_H}_{qm} < 2$, the following assertions hold true:*

a) $\widehat{\mathcal{R}}_-$ is invertible with $\|\widehat{\mathcal{R}}_-^{-1}\|_{\mathbf{X}_H\leftarrow\mathbf{X}_H} \leq 1$

b) $\widehat{\mathcal{R}} := \widehat{\mathcal{R}}_+\widehat{\mathcal{R}}_-^{-1}$ satisfying $\|\widehat{\mathcal{R}}\|_{\mathbf{X}_H\leftarrow\mathbf{X}_H} \leq \mathrm{e}^{\tau c^{S_H}_{qm}}$.

The main theorem can now be proved as follows.

Proof of Theorem 3.3.6. We start by splitting the error into two parts

$$x_H^n - x(t_n) = e_H^n + e^n_{\mathcal{J}_H}, \tag{3.3.23}$$

where

$$\begin{aligned}e_H^n &= x_H^n - \mathcal{J}_H x(t_n),\\ e^n_{\mathcal{J}_H} &= \mathcal{J}_H x(t_n) - x(t_n).\end{aligned}$$

As in [83, equation (27)] we have

$$\|e^n_{\mathcal{J}_H}\|_{\mathbf{X}} \leq C(E_{H,2}+E_{H,3}). \tag{3.3.24}$$

Hence, it remains to bound e_H^n.

(a) *Error recursion.* By subtracting (3.3.9) from (3.3.5) we obtain the error recursion

$$e_H^{n+\frac{1}{2}} = e_H^n + \frac{\tau}{2}S_H e_H^{n+\frac{1}{2}} + \frac{\tau}{2}\Delta F_H^n - \frac{\tau}{2}D^{n+\frac{1}{2}}_{\mathrm{IMEX},H}, \tag{3.3.25a}$$

$$e_H^{n+1} = e_H^n + \tau S_H e_H^{n+\frac{1}{2}} + \tau\Delta F_H^{n+\frac{1}{2}} - \tau\delta^{n+1}_{\mathrm{IMEX},H}, \tag{3.3.25b}$$

where $\Delta F_H^n = \widetilde{F}_H^n - \widetilde{F}_H^{n-1}$. We rewrite (3.3.25a) using the operators defined in (3.3.22) as

$$e_H^{n+\frac{1}{2}} = \widehat{\mathcal{R}}_-^{-1}(e_H^n + \frac{\tau}{2}(\Delta F_H^n - D_{\mathrm{IMEX},H}^{n+\frac{1}{2}})). \tag{3.3.26}$$

Plugging (3.3.26) into (3.3.25b) and using $\tau S_H = \widehat{\mathcal{R}}_+ - \widehat{\mathcal{R}}_-$ yields

$$e_H^{n+1} = \widehat{\mathcal{R}} e_H^n + \frac{\tau}{2}(\widehat{\mathcal{R}} - \mathrm{I})(\Delta F_H^n - D_{\mathrm{IMEX},H}^{n+\frac{1}{2}}) + \tau \Delta F_H^{n+\frac{1}{2}} - \tau \delta_{\mathrm{IMEX},H}^{n+1}. \tag{3.3.27}$$

(b) *Stability.* Solving the error recursion (3.3.27) and using Lemma 3.3.11 yields

$$\begin{aligned}
\|e_H^n\|_{\mathbf{x}_H} = \; & \|\widehat{\mathcal{R}}^n e_H^0\|_{\mathbf{x}_H} + \frac{\tau}{2}\sum_{i=0}^{n-1} \mathrm{e}^{\tau c_{qm}^{S_H}(n-1-i)} \|\mathrm{I} - \widehat{\mathcal{R}}\|_{\mathbf{x}_H \leftarrow \mathbf{x}_H} \|\Delta F_H^i\|_{\mathbf{x}_H} \\
& + \frac{\tau}{2}\|\sum_{i=0}^{n-1}(\widehat{\mathcal{R}}^{n-i} - \widehat{\mathcal{R}}^{n-i-1}) D_{\mathrm{IMEX},H}^{i+\frac{1}{2}}\|_{\mathbf{x}_H} \\
& + \tau \sum_{i=0}^{n-1} \mathrm{e}^{\tau c_{qm}^{S_H}(n-1-i)} \|\Delta F_H^{i+\frac{1}{2}}\|_{\mathbf{x}_H} + \tau \sum_{i=0}^{n-1} \mathrm{e}^{\tau c_{qm}^{S_H}(n-1-i)} \|\delta_{\mathrm{IMEX},H}^{i+1}\|_{\mathbf{x}_H}.
\end{aligned}$$

Multiplying by $\mathrm{e}^{-\tau c_{qm}^{S_H} n}$ on both sides, using the Lipschitz continuity and summation by parts for $D_{\mathrm{IMEX},H}^n$ terms results in

$$\begin{aligned}
\mathrm{e}^{-\tau c_{qm}^{S_H} n} \|e_H^n\|_{\mathbf{x}_H} \leq \; & \|e_H^0\|_{\mathbf{x}_H} + L_{\rho_H} \frac{\tau}{2}(\mathrm{e}^{-\tau c_{qm}^{S_H}} + 1) \sum_{i=0}^{n-1} \mathrm{e}^{-\tau c_{qm}^{S_H} i} \|e_H^i\|_{\mathbf{x}_H} \\
& + \frac{\tau}{2}\big(\|D_{\mathrm{IMEX},H}^{1/2}\|_{\mathbf{x}_H} + \|D_{\mathrm{IMEX},H}^{n-1/2}\|_{\mathbf{x}_H} + \sum_{i=1}^{n-1} \|\Delta D_{\mathrm{IMEX},H}^{i+\frac{1}{2}}\|_{\mathbf{x}_H}\big) \\
& + \tau \sum_{i=0}^{n-1} \|\delta_{\mathrm{IMEX},H}^{i+1}\|_{\mathbf{x}_H} + L_{\rho_H} \tau \sum_{i=0}^{n-1} \mathrm{e}^{-\tau c_{qm}^{S_H} i} \|e_H^{i+\frac{1}{2}}\|_{\mathbf{x}_H}.
\end{aligned} \tag{3.3.28}$$

Next, we need to estimate the error of the half step in terms of the full step. By (3.3.26), the Lipschitz continuity and Lemma 3.3.11, we derive

$$\|e_H^{n+\frac{1}{2}}\|_{\mathbf{x}_H} \leq (1 + \frac{\tau}{2} L_{\rho_H}) \|e_H^n\|_{\mathbf{x}_H} + \frac{\tau}{2} \|D_{\mathrm{IMEX},H}^{n+\frac{1}{2}}\|_{\mathbf{x}_H}.$$

Using this estimate in (3.3.28) and $\mathrm{e}^{-\tau c_{qm}^{S_H} i} \leq 1$ yields

$$\begin{aligned}
\mathrm{e}^{-\tau c_{qm}^{S_H} n} \|e_H^n\|_{\mathbf{x}_H} \leq \; & L_{\rho_H} \tau (2 + \frac{\tau}{2} L_{\rho_H}) \sum_{i=0}^{n-1} \mathrm{e}^{-\tau c_{qm}^{S_H} i} \|e_H^i\|_{\mathbf{x}_H} \\
& + \|e_H^0\|_{\mathbf{x}_H} + E_{H,\tau,1} + E_{H,\tau,2} + E_{H,\tau,3},
\end{aligned}$$

where

$$
\begin{aligned}
E_{H,\tau,1} &= L_{\rho_H}\tau\sum_{i=0}^{n-1}\frac{\tau}{2}\|D_{\mathrm{IMEX},H}^{i+\frac{1}{2}}\|_{\mathbf{X}_H}\\
E_{H,\tau,2} &= \frac{\tau}{2}\Big(\|D_{\mathrm{IMEX},H}^{1/2}\|_{\mathbf{X}_H}+\|D_{\mathrm{IMEX},H}^{n-1/2}\|_{\mathbf{X}_H}+\sum_{i=1}^{n-1}\|\Delta D_{\mathrm{IMEX},H}^{i+\frac{1}{2}}\|_{\mathbf{X}_H}\Big)\\
E_{H,\tau,3} &= \tau\sum_{i=0}^{n-1}\|\delta_{\mathrm{IMEX},H}^{i+1}\|_{\mathbf{X}_H}.
\end{aligned}
\tag{3.3.29}
$$

By the Grönwall's inequality and $\|e_H^0\|_{\mathbf{X}_H}\le E_{H,1}$, we get

$$
\|e_H^n\|_{\mathbf{X}_H}\le C(E_{H,1}+E_{H,\tau,1}+E_{H,\tau,2}+E_{H,\tau,3})\,\mathrm{e}^{\left(L_{\rho_H}(2+\frac{\tau}{2}L_{\rho_H})t_n+c_{qm}^{S_H}t_n\right)}. \tag{3.3.30}
$$

Overall, we get by Lemma 3.3.10, (3.3.24), (3.3.30), and

$$
\|u_H^n-u(t_n)\|_{\mathbf{V}}+\|v_H^n-u'(t_n)\|_{\mathbf{H}}\le\|x_H^n-x(t_n)\|_{\mathbf{X}},
$$

the error bound (3.3.7). □

Under additional assumptions, the existence of u_H^n and v_H^n on $[0,T]$ can be proved analogously to [83, Corollary 3.5] by the following Theorem.

Theorem 3.3.12. *Let Assumption of Theorem 3.3.6 be fulfilled. Further we assume that the spatial error E_H defined in (3.3.8) satisfies $E_H\to 0$ for $H\to 0$ and set*

$$
\rho := 3(C_{\mathbf{V}}\|u\|_{L^\infty([0,T];\mathbf{V})}+C_{\mathbf{H}}\|u'\|_{L^\infty([0,T];\mathbf{H})})+C_{\mathbf{Z}^{\mathbf{V}}}\|u'\|_{L^\infty([0,T];\mathbf{Z}^{\mathbf{V}})}.
$$

Then, there exist $\tau^\star, H^\star>0$ such that for all $h<H^\star$, $\tau<\tau^\star$ the approximation $[u_H^n\,v_H^n]$ obtained by the fully discrete IMEX scheme satisfies

$$
\max_{t_n\le T}\|u_H^n\|_{\mathbf{V}_H}\le\rho\quad\text{and}\quad\max_{t_n\le T}\|v_H^n\|_{\mathbf{H}_H}\le\rho
$$

and therefore the error bound (3.3.7) is valid with $\rho_H=\rho$.

Remark 3.3.13.

(i) Theorem 3.2.4 can be proved in a similar way as above by replacing the discrete expressions with the continuous ones, since the assumptions are almost identical in both cases. The error terms in space vanish in the semidiscrete case.

(ii) Analogously, one can obtain that Theorems 3.3.6 and 3.3.12 are also valid for the implicit midpoint rule if $\tau<\min\{\frac{1}{c_{qm}^{S_H}},\frac{2}{L_{\rho_H}}\}$ with $M=c_{qm}^{S_H}+\frac{L_{\rho_H}}{1-\frac{\tau}{2}L_{\rho_H}}$.

3.4 Application: Semilinear wave equations with highly oscillatory coefficients

In this section, we illustrate the theory of the previous sections using a specific nonlinear model problem. In particular, we are interested in wave problems with nonlinear damping terms and a coefficient which is oscillating fast in space. Using G-convergence results we derive a homogenized version, for which we apply the heterogeneous multiscale method.

3.4.1 Notations

Let $\Omega \subset \mathbb{R}^d, d \leq 3$, be a bounded domain with a Lipschitz boundary $\partial\Omega$. With $W^{s,p}(\Omega)$ we denote the standard Sobolev space and with $W_0^{s,p}(\Omega)$ the space consisting of functions of $W^{s,p}(\Omega)$ which are zero on $\partial\Omega$ in the trace sense. If $p = 2$ we use the notation $H^s(\Omega)$ and $H_0^s(\Omega)$. In addition, let

$$H^s_{\#}(Y) := \Big\{ w \in H^s_{per} \mid \int_Y w = 0 \Big\},$$

where $H^s_{per}(Y)$ is the closure of $C^\infty_{per}(Y)$ in the H^s-norm on a hypercube $Y \subset \Omega$. In the following, we denote Y_σ as the hypercube with side length σ and Y as the unit hypercube. Further, there exists a constant $c_\Omega > 0$, such that the Poincare inequality is fulfilled for all $w \in H_0^1(\Omega)$.

$$\|w\|_{L^2(\Omega)} \leq c_\Omega \|\nabla w\|_{L^2(\Omega)}.$$

For a Banach space B we denote by $L^p([0,T];B)$ the Bochner space of functions $u : [0,T] \to B$ equipped with the norm

$$\|u\|_{L^p([0,T];B)} := \Big(\int_0^T \|u(t)\|_B^p \, \mathrm{d}t \Big)^{\frac{1}{p}}.$$

In the following, we set

$$\mathbf{V} = H_0^1(\Omega) \quad \text{and} \quad \mathbf{H} = L^2(\Omega).$$

3.4.2 Model problem and G-convergence

We consider the semilinear wave equation with damping:

$$\begin{cases} \partial_{tt} u^\varepsilon - \nabla \cdot (a^\varepsilon(x) \nabla u^\varepsilon) - \nabla \cdot (\beta(x) \nabla \partial_t u^\varepsilon) = \mathcal{G}(x, \partial_t u^\varepsilon) + f & \text{in } (0,T) \times \Omega, \\ u^\epsilon = 0 & \text{on } (0,T) \times \partial\Omega, \\ u^\varepsilon(0) = u^0 \quad , \quad \partial_t u^\varepsilon(0) = v^0 & \text{in } \Omega. \end{cases} \tag{3.4.1}$$

$\partial_t u^\varepsilon$ and $\partial_{tt} u^\varepsilon$ are time derivatives of u, which we denoted in Section 3.2 and 3.3 as u' and u'' , respectively. However, since the index ε occurs here, we have changed the notation for better readability. For the coefficient a^ε, β and the nonlinear operator $\mathcal{G}$ we assume the following

Assumption 3.4.1. *Let $\lambda, \Lambda, C_\beta, c_\beta$ be positive constants.*

(i) $a^\varepsilon \in (W^{1,\infty}(\Omega))^{d\times d}_{sym}$ *and* $\Lambda|\xi|^2 \geq \xi \cdot a^\varepsilon(x)\xi > \lambda\,|\xi|^2$, *a.e. in* Ω *for all* $\xi \in \mathbb{R}^n$, $\xi \neq 0$.

(ii) $\beta \in C(\overline{\Omega})$ *and* $c_\beta < \beta(x) \leq C_\beta$.

(iii) $\mathcal{G}(\cdot, \eta)$ *is Lebesgue measurable for all* $\eta \in \mathbb{R}$ *and*

$$\mathcal{G}(x, \cdot) : L^2([0,T]; \mathbf{H}) \to L^2([0,T]; \mathbf{H})$$

maps bounded sets to bounded sets for almost all $x \in \Omega$.

(iv) $\|\mathcal{G}(\cdot, v) - \mathcal{G}(\cdot, w)\|_{\mathbf{H}} \leq L_\rho \|v - w\|_{\mathbf{H}}$ *for all* $v, w \in \mathbf{H}$

(v) For all sequences $\{w^\varepsilon\}$ *with* $w^\varepsilon \to w$ *in* $L^2([0,T]; \mathbf{H})$ *for* $\varepsilon \to 0$ *there exists a subsequence which we still denote by* $\{w^\varepsilon\}$ *with*

$$\mathcal{G}(\cdot, w^\varepsilon) \rightharpoonup \mathcal{G}(\cdot, w)$$

in $L^2([0,T]; \mathbf{H})$.

(vi)

$$f \in W^{1,1}_{loc}([0,\infty); H^2(\Omega)) \quad \text{and} \quad u^0, v^0 \in H^2(\Omega) \cap \mathbf{V} \qquad \circ$$

The weak formulation of (3.4.1) reads as follows

$$\big(\partial_{tt} u^\varepsilon, \Phi\big)_{\mathbf{H}} + d^\varepsilon(u^\varepsilon, \Phi) + b(\nabla \partial_t u^\varepsilon, \nabla \Phi) = (\mathcal{G}(\cdot, \partial_t u^\varepsilon), \Phi)_{\mathbf{H}} + \big(f, \Phi\big)_{\mathbf{H}}, \tag{3.4.2}$$

where

$$\begin{aligned} d^\varepsilon(u^\varepsilon, \Phi) &:= \big(a^\varepsilon \nabla u^\varepsilon, \nabla \Phi\big)_{\mathbf{H}}, \\ b(u^\varepsilon, \Phi) &:= \big(\beta \nabla u^\varepsilon, \nabla \Phi\big)_{\mathbf{H}}. \end{aligned} \tag{3.4.3}$$

for all $\Phi \in \mathbf{V}$. Next we check whether (3.4.2) fits into the abstract setting.

Lemma 3.4.2. *Let Assumption 3.4.1 be satisfied. Then the weak formulation (3.4.2) is of the form (3.2.1). Further Assumption 3.2.1 is satisfied with $c_d = 1$ and $c_{qm} = 0$.*

Proof. This is a direct consequence of Assumption 3.4.1 and (3.4.3). □

By Lemma 3.4.2 and Corollary 3.2.3 (3.4.1) is locally wellposed.

Next we give an example inspired by [8] for which the Assumptions 3.4.1(iii),(iv) are satisfied.

Example 3.4.3. We set

$$\mathcal{G}(x,\eta) = \theta(x)\big((|\eta|+\sigma)^\gamma - \sigma^\gamma\big)\,\mathrm{sgn}(\eta), \tag{3.4.4}$$

$0 < \gamma \le 1$ and $\sigma \ge 0$ and $\theta \in L^\infty(\Omega)$. Let w^ε with $w^\varepsilon \to w$ in $L^2([0,T];\mathbf{H})$. We have to show that, if $\varepsilon \to 0$,

$$\Big|\int_{(0,T)\times\Omega} (\mathcal{G}(x,w^\varepsilon) - \mathcal{G}(x,w))\varphi\Big| \to 0 \tag{3.4.5}$$

for all $\varphi \in L^2([0,T];\mathbf{H})$. We know that there is a subsequence, which we also denote by $\{w^\varepsilon\}$, such that for almost all t

$$w^\varepsilon(t) \to w(t) \quad \text{in } \mathbf{H}.$$

According to the Theorem of Yegorov there exist a sequence $\{\mathcal{Q}_n\} \subset (0,T)\times\Omega$ with

$$\big|\big((0,T)\times\Omega\big) \setminus \mathcal{Q}_n\big| < \frac{1}{n} \quad \text{and} \quad w^\varepsilon \to w \quad \text{uniform in } \mathcal{Q}_n.$$

Further if ε is small enough

$$\begin{aligned}|\mathcal{G}(x,w^\varepsilon)\varphi| &= |\theta|((|w^\varepsilon|+\sigma)^\gamma - \sigma^\gamma)|\varphi| \\ &\le |\theta|((|w|+1+\sigma)^\gamma - \sigma^\gamma)|\varphi| \in L^1(\mathcal{Q}_n).\end{aligned}$$

By dominated convergence we obtain

$$\int_{\mathcal{Q}_n} \mathcal{G}(x,w^\varepsilon)\varphi \to \int_{\mathcal{Q}_n} \mathcal{G}(x,w)\varphi.$$

If we now let $n \to \infty$ it follows that (3.4.5) converges to zero. If $\sigma > 0$, then $\mathcal{G}(x,\cdot) \in C^1(\mathbb{R})$ for almost all $x \in \Omega$. Then $\mathcal{G}$ is Lipschitz continuous with respect to the $\mathbf{H}$-norm, since

$$\begin{aligned}\|\mathcal{G}(v) - \mathcal{G}(w)\|_{\mathbf{H}} &\le \|\theta \int_0^1 \partial_\xi \mathcal{G}(x, v+\xi(v-w))(v-w)\|_{\mathbf{H}} \\ &\le \|\theta\|_{L^\infty} \|\frac{\gamma}{(|\eta|+\sigma)^{1-\gamma}}\|_{L^\infty(\mathbb{R})} \|(v-w)\|_{\mathbf{H}} \le \frac{\gamma\|\theta\|_{L^\infty}}{\sigma^{1-\gamma}}\ \|v-w\|_{\mathbf{H}}.\end{aligned}$$

◇

By the assumptions we obtain the following convergence result.

Theorem 3.4.4. [113, Theorem 8.3] *Consider the sequence of damped hyperbolic problems (3.4.1) under the Assumptions 3.4.1. Then,*

$$\begin{aligned} u^\varepsilon &\rightharpoonup u^{hom} && \text{in } L^2([0,T];\mathbf{V}) \\ a^\varepsilon \nabla u^\varepsilon &\rightharpoonup a^{hom}\nabla u^{hom} && \text{in } L^2([0,T];\mathbf{H}) \\ \mathcal{G}(\partial_t u^\varepsilon) &\rightharpoonup \mathcal{G}(\partial_t u^{hom}) && \text{in } L^2([0,T];\mathbf{H}),\end{aligned}$$

where u^{hom} is the solution of (3.4.6).

$$\begin{cases} \partial_{tt} u^{hom} - \nabla \cdot (a^{hom} \nabla u^{hom}) \\ \qquad -\nabla \cdot (\beta \nabla \partial_t u^{hom}) = \mathcal{G}(\partial_t u^{hom}) + f \quad \text{in } (0,T) \times \Omega \\ \qquad\qquad u^{hom} = 0 \quad \text{on } (0,T) \times \partial\Omega \\ \qquad u^{hom}(0) = u^0 \quad , \quad \partial_t u^{hom}(0) = v^0 \quad \text{in } \Omega. \end{cases} \tag{3.4.6}$$

Remark 3.4.5. *In [113], the authors assumed $\beta = 1$ and $\mathcal{G}$ as in the example 3.4.3 for $\sigma = 0$ and $\theta > 0$. However, it is straightforward to extend the proof of the theorem to the more general assumptions, where essentially only constants change.*

Remark 3.4.6. *It can be seen in the proof that the homogenized coefficient a^{hom} coincides with the homogenized coefficient from an auxiliary parabolic problem. Therefore, we know that the coefficient satisfies Assumption 3.4.1(i) by parabolic compactness [135, Theorem 3.1]. Lemma 3.4.2 shows that also the weak formation of the homogenized equation (3.4.6) fulfills Assumption 3.2.1.*

3.4.3 Heterogeneous multiscale method

We formulate the Finite element heterogeneous multiscale method (FE-HMM). For the macroscopic model we take the homogeneous equation (3.4.6) which we derived in Section 3.4.2. Based on this model, we want to compute an approximation of u^{hom} by discretization in space. We start with the macroscopic problem. We use a partition $\mathcal{T}_H$ into simplicial or quadrilateral elements, where we denote the elements of $\mathcal{T}_H$ by K. As the finite element space we choose

$$\mathbf{W}_H := \{v_H \in \mathbf{V} \mid v_{H|_K} \in \mathbb{P}_p, \forall K \in \mathcal{T}_H\},$$

where $\mathbb{P}_p$ is the space of polynomials of maximal degree p. Next we want to approximate the homogenized coefficient a^{hom}. Theorem 3.4.4 does not provide an explicit representation a^{hom}. However, if we assume more structure on the microscopic scale, i.e.,

$$a^{\varepsilon}(x) = a\Big(x, \frac{x}{\epsilon}\Big), \tag{3.4.7}$$

and $a(x,y)$ is Y-periodic w.r.t y, we can derive the same well-known formula for the homogenized coefficient as in the elliptic case [18, Chapter 1, Eq. 2.20] with the help of Remark 3.4.6 and [135, Theorem 8.1]

$$a_{ij}^{hom}(x) = \int_Y \sum_{k=1}^{d} a_{ik}(x,y)\big(\delta_{jk} + \frac{\partial \chi^j}{\partial y_k}(x,y)\big)\, \mathrm{d}y,$$

where δ_{jk} are the Kronecker delta and χ^j are the solutions of the following cell problems on Y :
Find $\chi^j \in H^1_\#(Y)$:

$$\int_Y a(x,y)(e^j + \nabla\chi^j(x,y)) \cdot \nabla z(y)\ \mathrm{d}y = 0 \tag{3.4.8}$$

for all $z \in H^1_\#(Y)$. To approximate the resulting bilinear form we use a suitable quadrature formula $\{\omega^j_K, x^j_K\}$ with positive weights ω^j_K in the following:

$$d^{hom}(\Phi_H, \Psi_H) = \sum_{K\in\mathcal{T}_H} \int_K a^{hom}(x)\nabla\Phi_H \cdot \nabla\Psi_H \approx \sum_{K\in\mathcal{T}_H} \sum_{j=1}^{J} \omega^j_K a^{hom}_{K,j} \nabla\Phi_H \cdot \nabla\Psi_H \tag{3.4.9}$$

with the coefficient

$$a^{hom}_{K,j}\nabla\Phi_H \cdot \nabla\Psi_H := a^{hom}(x^j_K)\nabla\Phi_H \cdot \nabla\Psi_H = \frac{1}{|Y_\varepsilon|} \int_{Y_{\varepsilon,K,j}} a\Big(x^j_K, \frac{x}{\epsilon}\Big)\nabla\phi_j(x) \cdot \nabla\psi_j(x)\ \mathrm{d}x,$$

where ϕ_j, ψ_j are the solutions of the shifted cell problems on $Y_{\varepsilon,K,j} = x^j_K + Y_\varepsilon$:
Find $\phi_j - \Phi^{lin}_{H,K,j} \in H^1_\#(Y_{\varepsilon,K,j})$:

$$\int_{Y_{\varepsilon,K,j}} a\Big(x^j_K, \frac{x}{\epsilon}\Big)\nabla\phi_j(x) \cdot \nabla z(x)\ \mathrm{d}x = 0 \tag{3.4.10}$$

for all $z \in H^1_\#(Y_{\varepsilon,K,j})$, with

$$\Phi^{lin}_{H,K,j}(x) = \Phi_H(x^j_K) + (x - x^j_K) \cdot \nabla\Phi_H(x^j_K).$$

ψ_j solves a similar cell problem, using Ψ_H instead of Φ_H.

Remark 3.4.7. *To ensure the optimal convergence rates of the FE-HMM we have to use a suitable quadrature formula $\{\omega^j_K, x^j_K\}$ of sufficient order. A detailed discussion including estimates of the quadrature error can be founded [2, Chp. 4.1].*

In a next step we discretize the cell problems. Since in applications the finescale parameter ε may be unknown, we consider as cell domain $Y_{\delta,K,j}$ with $\delta \geq \varepsilon$. Further, let $\mathcal{T}_h$ be a partition for each cell domain $Y_{\delta,K,j} \supseteq Y_{\varepsilon,K,j}$ consisting of simplicial or quadrilateral elements and a microscopic finite element space consisting of piecewise polynomial functions

$$V^q_h(Y_{\delta,K,j}) := \{v_h \in W(Y_{\delta,K,j}) \mid v_{h|_\kappa} \in \mathbb{P}_q, \forall\kappa \in \mathcal{T}_h\},$$

where $\mathbb{P}_q$ consist of all polynomials of order q and $W(Y_{\delta,K,j})$ is a subspace of the Sobolev space $H^1(Y_{\delta,K,j})$. This space describes the coupling between the macroscopical and microscopical problem. Common choices are

(1) periodic boundary conditions: $W(Y_{\delta,K,j}) := H^1_{\#}(Y_{\delta,K,j})$

(2) Dirichlet boundary conditions: $W(Y_{\delta,K,j}) := H^1_0(Y_{\delta,K,j})$

(3) Neumann boundary conditions: $W(Y_{\delta,K,j}) := H^1_N(Y_{\delta,K,j}) = \{u \in H^1(\Omega) \mid \int_{Y_{\delta,K,j}} \nabla u = \mathbf{0}\}$.

We consider the discrete micro problems: find $\phi_{h,K,j} - \Phi^{lin}_{H,K,j} \in V^q_h(Y_{\delta,K,j})$:

$$\int_{Y_{\delta,K,j}} a^\varepsilon(x)\nabla\phi_{h,K,j}(x)\cdot\nabla z_h(x)\,\mathrm{d}x = 0 \tag{3.4.11}$$

for all $z_h \in V^q_h(Y_{\delta,K,j})$.
Using this, we formulate the discrete form of (3.4.9) as

$$d^{hom}_h(\Phi_H,\Psi_H) = \sum_{K\in\mathcal{T}_H}\sum_{j=1}^{J}\omega^j_K\, a^{hom}_{K,j,h}\nabla\Phi_H\cdot\nabla\Psi_H \tag{3.4.12}$$

with the discrete coefficient

$$a^{hom}_{K,j,h}\nabla\Phi_H\cdot\nabla\Psi_H = \frac{1}{|Y_\delta|}\int_{Y_{\delta,K,j}} a^\varepsilon(x)\nabla\phi_{h,K,j}(x)\cdot\nabla\psi_{h,K,j}(x)\,\mathrm{d}x. \tag{3.4.13}$$

Remark 3.4.8. *In the discrete formulation (3.4.11), (3.4.13) we replaced* $a\Big(x^j_K, \frac{x}{\epsilon}\Big)$ *by* $a^\varepsilon(x)$ *since the structured form may be unknown in applications.*

We now can formulate the semi-discrete equation of the homogenized problem (3.4.6). We set $u^0_H := \mathcal{I}_H u^0$ and $v^0_H := \mathcal{I}_H v^0$, where $\mathcal{I}_H$ is the interpolation operator mapping from $H^2(\Omega)$ to $\mathbf{W}_H$. Further let

$$\beta_H = \mathcal{I}_H\beta \quad , \quad b_H(\cdot,\cdot) = (\beta_H\,\cdot,\cdot)_{\mathbf{V}} \quad , \quad \mathcal{G}_H = \mathcal{I}_H\mathcal{G} \quad , \quad f_H = \mathcal{I}_H f. \tag{3.4.14}$$

Let u^{hom}_H be the solution of the semi-discrete equation

$$\begin{aligned}\big(\partial_{tt}u^{hom}_H,\Phi_H\big)_{\mathbf{H}} + d^{hom}(u^{hom}_H,\Phi_H) + b_H(\partial_t u^{hom}_H,\Phi_H)& \\ = (\mathcal{G}_H(\cdot,\partial_t u^{hom}_H),\Phi_H)_{\mathbf{H}} + \big(f_H(t),\Phi_H\big)_{\mathbf{H}}&\end{aligned} \tag{3.4.15}$$

for all $\Phi_H \in \mathbf{W}_H$, with $u^{hom}_H(0) = u^0_H$ and $\partial_t u^{hom}_H(0) = v^0_H$.

3.4.4 Error bounds for the fully discrete FE-HMM

In this section we want to apply the abstract spatial error bounds to the discrete homogenized problem.

Lemma 3.4.9. *Let Assumption 3.4.1 hold true. Then the FE-HMM satisfies Assumptions 3.3.1 and 3.3.2.*

Proof. We set

$$(\cdot,\cdot)_{\mathbf{H}_H} = (\cdot,\cdot)_{\mathbf{H}} \quad , \quad d_H = d_h^{hom}, \quad (\cdot,\cdot)_{\mathbf{V}_H} = d_H(\cdot,\cdot) + (\cdot,\cdot)_{\mathbf{H}_H}.$$

Hence, $\mathcal{P}_H^{\mathbf{H}}$ is the L^2-projection on $\mathbf{H}_H = (\mathbf{W}_H, (\cdot,\cdot)_{\mathbf{H}_H})$ with $C_{\mathbf{H}} = 1$ and $\mathbf{V}_H = (\mathbf{W}_H, (\cdot,\cdot)_{\mathbf{V}_H})$. Using [1, Lemma 5], d_h^{hom} is coercive with constant λ_H and bounded w.r.t. $\|\cdot\|_{\mathbf{V}}$ and therefore Assumption 3.3.1(i) is fulfilled, where $\widehat{c}_{d_H} = 0$ and

$$C_{\mathbf{H}_H,\mathbf{V}_H} = \frac{c_\Omega}{\sqrt{\lambda_H + c_\Omega^2}}.$$

Then $\mathcal{P}_H^{\mathbf{V}}$ exists by the Theorem of Lax-Migram and $C_{\mathbf{V}} = \sqrt{\frac{1+c_\Omega^2}{\lambda_H}}$. Consequently, Assumption 3.3.2(i)(iii) is fulfilled. Assumption 3.3.2(ii) is fulfilled for a constant $C_{\mathbf{Z}^{\mathbf{V}}}$ due to standard interpolation error estimates for each subspace $\mathbf{Z}^{\mathbf{V}}$ of $C^0(\overline{\Omega})$. Assumption 3.3.1(ii),(iv) follows directly from Assumption 3.4.1 and boundedness of the interpolation operator. In addition, for all $v_H, w_H \in \mathbf{H}_H$ with $\|v_H\|_{\mathbf{H}_H}, \|w_H\|_{\mathbf{H}_H} \leq \rho_H$ we obtain

$$\|\mathcal{G}_H(\cdot, v_H) - \mathcal{G}_H(\cdot, w_H)\|_{\mathbf{H}_H} \leq L_\rho \|v_H - w_H\|_{\mathbf{H}_H}$$

and therefore 3.3.1(iii) is fulfilled. □

Next, we specify the required regularity of the exact solution in order to derive a priori error bounds.

Assumption 3.4.10.

(i) *Let $C_1, C_2 > 0$ be constants independent of ε and δ. For the small-scale parameter we assume*

$$\varepsilon \leq C_1 \delta^2 \leq C_2 H.$$

(ii) *For the discrete initial conditions we assume*

$$\|u_H^0 - \mathcal{I}_H u^0\|_{\mathbf{V}_H} + \|v_H^0 - \mathcal{I}_H v^0\|_{\mathbf{H}_H} \leq C H^p.$$

(iii) *The right-hand side and the coefficients satisfy*

$$f \in L^\infty([0,T]; H^{\max\{2,p\}}(\Omega)), \quad \beta \in W^{p+1,\infty}(\Omega), \quad a \in W^{1,\infty}(\Omega, W_\#^{1,\infty}(Y)).$$

(iv) *Let $T > 0$ and $p \geq 1$. Then we assume that the solution of the homogenized problem (3.4.6) satisfies the following regularity assumptions*

$$\begin{aligned} u^{hom} &\in L^\infty([0,T]; H^{p+1}(\Omega)), \\ \partial_t u^{hom} &\in L^\infty([0,T]; H^{p+2}(\Omega)), \\ \partial_{tt} u^{hom} &\in L^\infty([0,T]; H^{\max\{2,p\}}(\Omega)). \end{aligned}$$

We set

$$\rho = 3\max\Big\{1, \sqrt{\frac{\Lambda}{\lambda_H}}\Big\}\Big(\|u^{hom}\|_{L^\infty([0,T];\mathbf{V})} + \|\partial_t u^{hom}\|_{L^\infty([0,T];\mathbf{H})}\Big) + C_{\mathbf{Z}^\mathbf{V}}\|\partial_t u^{hom}\|_{L^\infty([0,T];H^{\max\{2,p\}})}$$

Further, for the nonlinearity we assume

$$\mathcal{G}(x, \partial_t u^{hom}) \in L^\infty([0,T]; H^p(\Omega) \cap C(\overline{\Omega})). \qquad \bigcirc$$

We now estimate the error terms occurring in Theorem 3.3.6 in our specific setting.

Lemma 3.4.11. *Under the Assumption 3.4.1 and 3.4.10, the spatial discretization error bounds of FE-HMM defined in Theorem 3.3.6 can be bounded by*

$$E_{H,i} \leq C\Big(H^p + \Big(\frac{h}{\varepsilon}\Big)^{2q} + e_{\text{mod}}\Big).$$

for $i = 1, \dots 4$, where p and q are the macroscopic and microscopic polynomial degree, respectively, and e_{mod} is a modeling error. The constant C does not depend on H, h, and ε.

Proof. The terms $E_{H,i}$ for $i = 1, 3$ can be bounded using standard interpolation bounds (see ,e.g., [28, Corollary 4.4.20]) by

$$E_{H,i} \leq CH^p.$$

For the errors of the nonlinearity $\mathcal{G}$ we obtain by local Lipschitz continuity

$$\begin{aligned}\|\mathcal{P}_H^{\mathbf{H}}\mathcal{G}(\partial_t u^{hom}) - \mathcal{G}_H(\mathcal{I}_H\partial_t u^{hom})\|_{\mathbf{H}_H} &\leq \|\mathcal{G}(\partial_t u^{hom}) - \mathcal{G}_H(\partial_t u^{hom})\|_{\mathbf{H}} \\ &\quad + \|\mathcal{G}_H(\partial_t u^{hom}) - \mathcal{G}_H(\mathcal{I}_H\partial_t u^{hom})\|_{\mathbf{H}} \\ &\leq C\max\{1, L_\rho\}H^p(\|\mathcal{G}(\partial_t u^{hom})\|_{H^p} + \|\partial_t u^{hom}\|_{H^p}).\end{aligned}$$

Further, for the strong damping term we consider

$$\begin{aligned}\|\mathcal{P}_H^{\mathbf{H}}\mathcal{B}u' - \mathcal{B}_H\mathcal{I}_H u'\|_{\mathbf{H}_H} &\leq C\big(\|(\beta - \mathcal{I}_H\beta)\|_{W^{1,\infty}}\|u'\|_{H^2} + \|(\mathcal{I}_H\beta)\|_{W^{1,\infty}}\|(u' - \mathcal{I}_H u')\|_{H^2}\big) \\ &\leq CH^p((\|\beta\|_{W^{p+1,\infty}} + \|\beta\|_{W^{1,\infty}})\|u'\|_{H^{p+2}}).\end{aligned}$$

We obtain

$$E_{H,4} \leq CH^p.$$

It remains to bound the conformity error of the scalar products in $E_{H,2}$. Let $\Phi_H, \Psi_H \in \mathbf{V}_H$. Then

$$|d^{hom}(\Phi_H, \Psi_H) - d_h^{hom}(\Phi_H, \Psi_H)| \leq \big(e_{\text{mod}} + e_{\text{mic}} + e_{\text{qf}}\big)\|\Phi_H\|_{\mathbf{V}}\|\Psi_H\|_{\mathbf{V}},$$

where $\|\cdot\|_F$ is the Frobenius norm and e_{qf} is a quadrature error arising from the quadrature in (3.4.9). The first term e_{mod} represents the model error caused by the use of wrong cell sizes or wrong boundary conditions. The second term e_{mic} represents the discretization error of the cell problems and can be estimated as follows using [2, Lemma 4.7] and Remark 3.4.7:

$$e_{\text{mic}} \leq C\Big(\frac{h}{\epsilon}\Big)^{2q}.$$

□

Remark 3.4.12. *The model error e_{mod} can be bounded depending on the type of boundary conditions selected. If a^ε is sufficiently regular*

$$e_{mod} \leq C \begin{cases} \delta, & \text{if } \frac{\delta}{\epsilon} \in \mathbb{N} \text{ and periodic b.c.,} \\ \delta + \frac{\varepsilon}{\delta}, & \text{else.} \end{cases}$$

The proofs can be found in [1, Proposition 14] and [49, Theorem 3.4]. Note that

$$W^{1,\infty}(\Omega, W^{1,\infty}_{\#}(Y)) \hookrightarrow C^{0,1}(\Omega, W^{1,\infty}_{\#}(Y))$$

(see [9, Theorem 5.2]).

We now apply the abstract time error bounds to the homogenized equation (3.4.6).

Theorem 3.4.13. *Let Assumption 3.4.1 be fulfilled and assume that the solution u^{hom} of (3.4.6) satisfies*

$$u \in C^4([0,T];\mathbf{H}) \cap C^3([0,T];\mathbf{V}) \cap C^2([0,T];H^2(\Omega)).$$

(a) (Semi-discrete): There exists $\tau^\star > 0$, such that for all $\tau < \tau^\star$ the error of the IMEX applied on (3.4.6) can be bounded by

$$\|u^{hom,n} - u^{hom}(t_n)\|_{\mathbf{V}} + \|v^{hom,n} - \partial_t u^{hom}(t_n)\|_{\mathbf{H}} \leq C t_n \, \mathrm{e}^{M t_n} \, \tau^2,$$

where $M := \left(\frac{1}{2}\lambda c_\Omega + L_\rho(2 + \frac{\tau}{2}L_\rho)\right)$. The constant C depends only on the derivatives of u^{hom} and L_ρ, T.

(b) (Full-discrete): Let Assumption 3.4.10 be fulfilled. Then there exist $\tau^\star, h^\star > 0$ such that for all $\tau < \tau^\star$ and $h < h^\star$ the errors of the fully discrete solution of the IMEX and the FE-HMM satisfy the following bounds

$$\|u_H^{hom,n} - u^{hom}(t_n)\|_{\mathbf{V}} + \|v_H^{hom,n} - \partial_t u^{hom}(t_n)\|_{\mathbf{H}} \leq C t_n \, \mathrm{e}^{M t_n} \left(\tau^2 + H^p + \left(\frac{h}{\varepsilon}\right)^{2q} + e_{\mathrm{mod}}\right),$$

where $M := \left(\frac{1}{2}\lambda_H c_\Omega + L_\rho(2 + \frac{\tau}{2}L_\rho)\right)$. The constant C depends only on the derivatives of u^{hom} and L_ρ, T.

3.5 Numerical Examples

In the following we present numerical results to demonstrate the accuracy of the error estimates. For this, we use an implementation in the $C++$ FEM-library *deal.II* [5]. We verify numerically the space and time convergence rates of Theorem 3.4.13 for periodic coefficients. In the following we define the error function for all $\tau > 0$ as

$$\mathbf{E}_\tau(t_n) := \frac{\|u_H^n - u(t_n)\|_{H^1(\Omega)} + \|v_H^n - u'(t_n)\|_{L^2(\Omega)}}{\|u(t_n)\|_{H^1(\Omega)} + \|u'(t_n)\|_{L^2(\Omega)}}, \quad \text{where } t_n = n\tau.$$

Note that in our example the denominator will never be zero.

3.5.1 Implementation

To explain the implementation we first introduce some notation. We denote by $\boldsymbol{\mu}, \boldsymbol{\nu} \in \mathbb{R}^{N_H}$ the coefficient vector corresponding to the solution $u_H, v_H \in \mathbf{V}_H$. Furthermore, we set $\mathbf{M} \in \mathbb{R}^{N_H \times N_H}$ as the mass matrix. We denote $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{N_H \times N_H}$ as the stiffness matrices and $\mathbf{G}, \mathbf{f} \in \mathbb{R}^{N_H}$ as the load vectors to the discrete quantities $\mathcal{A}_H, \mathcal{B}_H$ and $\mathcal{G}_H, f_H$. In the following, we set

$$\mathbf{G}^n(\boldsymbol{\mu}^n, \boldsymbol{\nu}^n) = \mathbf{G}(\boldsymbol{\mu}^n, \boldsymbol{\nu}^n) + \mathbf{f}^n.$$

From (3.3.5a) we obtain for the half step the following second-order formation

$$\mathbf{M}\boldsymbol{\nu}^{n+\frac{1}{2}} = \mathbf{M}\boldsymbol{\nu}^n - \frac{\tau}{2}\mathbf{A}\boldsymbol{\mu}^n - \frac{\tau^2}{4}\mathbf{A}\boldsymbol{\nu}^{n+\frac{1}{2}} - \frac{\tau}{2}\mathbf{B}\boldsymbol{\nu}^{n+\frac{1}{2}} + \frac{\tau}{2}\mathbf{G}^n(\boldsymbol{\mu}^n, \boldsymbol{\nu}^n), \tag{3.5.1a}$$

$$\boldsymbol{\mu}^{n+\frac{1}{2}} = \boldsymbol{\mu}^n + \frac{\tau}{2}\boldsymbol{\nu}^{n+\frac{1}{2}}. \tag{3.5.1b}$$

Using this, we can calculate $\boldsymbol{\nu}^{n+1}$ and $\boldsymbol{\mu}^{n+1}$ for the full-step (3.3.5b) by

$$\mathbf{M}\boldsymbol{\nu}^{n+1} = 2\mathbf{M}\boldsymbol{\nu}^{n+\frac{1}{2}} - \mathbf{M}\boldsymbol{\nu}^n + \tau\big(\mathbf{G}^{n+\frac{1}{2}}(\boldsymbol{\mu}^{n+\frac{1}{2}}, \boldsymbol{\nu}^{n+\frac{1}{2}}) - \mathbf{G}^n(\boldsymbol{\mu}^n, \boldsymbol{\nu}^n)\big), \tag{3.5.2a}$$

$$\boldsymbol{\mu}^{n+1} = \boldsymbol{\mu}^n + \tau\boldsymbol{\nu}^{n+\frac{1}{2}}, \tag{3.5.2b}$$

In total, we have to solve two linear systems (3.5.1a), (3.5.2a) in each time step with one application of $\mathbf{A}$, one of $\mathbf{B}$ and two evaluations of the nonlinearity $\mathbf{G}$.
In the special case $\mathbf{G}^n(\boldsymbol{\mu}^n, \boldsymbol{\nu}^n) = \mathbf{G}^n(\boldsymbol{\mu}^n)$ only one linear system, namely (3.5.1a), must be solved, since we do not need to calculate $\boldsymbol{\nu}^{n+1}$, as only $\mathbf{M}\boldsymbol{\nu}^{n+1}$ is required.

3.5.2 Setting

We set as our domain $\Omega = (0,1)^2$, $T = 1$ and consider the homogenized nonlinear wave equation

$$\begin{aligned}\partial_{tt}u^{hom} - \nabla \cdot (a^{hom}(x)\nabla u^{hom}) - 0.01\Delta(\partial_t u^{hom}) \\ + \operatorname{sgn}(\partial_t u^{hom})(|\partial_t u^{hom} + 0.0001|^{0.6} - 0.0001^{0.6}) = f\end{aligned} \tag{3.5.3}$$

with homogeneous Dirichlet boundary conditions. For the microscopic problem we consider periodic boundary conditions with $\varepsilon = 2^{-15}$ and $\delta = 2^{-13}$, where we use $\mathcal{P}_1$ linear elements on a uniform microscopic mesh with grid width h. We set

$$a^\epsilon(x_1, x_2) = \begin{pmatrix} 0.33 + 0.15(\sin(2\pi x_1) + \sin(2\pi\frac{x_1}{\epsilon})) & 0 \\ 0 & 0.33 + 0.15(\sin(2\pi x_1) + \sin(2\pi\frac{x_1}{\epsilon})) \end{pmatrix}$$

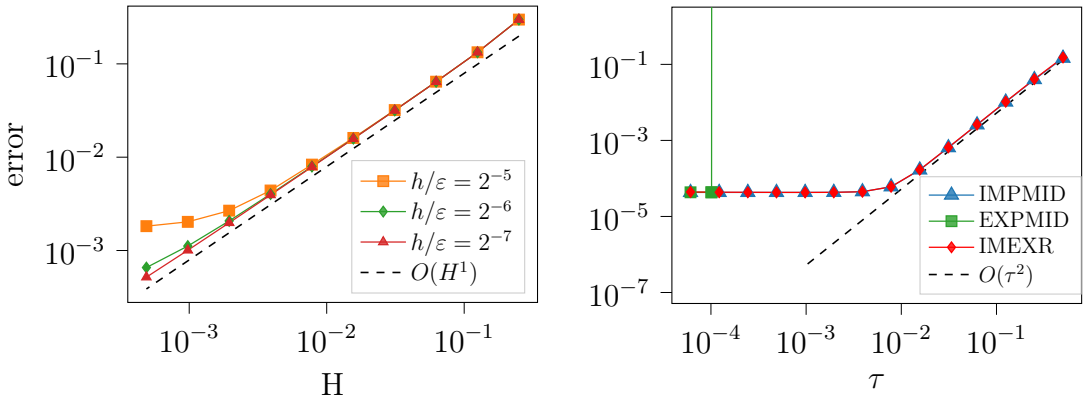


Figure 3.1: The error $\mathbf{E}_\tau(1)$ of finite element heterogeneous multiscale method plotted against the macroscopic mesh size H on the left and the time step-size τ on the right.

as coefficient. According to [4, Chapter 5.3], the homogenized coefficient can be calculated exactly as

$$a^{hom}(x_1, x_2) = \begin{pmatrix} 0.3 \cdot \sqrt{(1.1 + 0.5 \cdot \sin(2\pi x_1))^2 - 0.25} & 0 \\ 0 & 0.3 \cdot (1.1 + 0.5 \cdot \sin(2\pi x_1)) \end{pmatrix}. \tag{3.5.4}$$

As an exact solution of the homogenized equation we choose

$$u^{hom}(t, x_1, x_2) = \mathrm{e}^{\pi t} \sin(\pi x_1^2) \sin(\pi x_2^2)$$

with a corresponding right-hand side and initial conditions. For the macroscopic discretization we use finite elements on a uniform quadrilateral mesh with meshsizes $H = 2^{-k}$.

3.5.3 Results

In Figure 3.1a we consider the spatial error for different microscopic meshsizes for a fixed time step-size $\tau = 10^{-3}$. Here we use finite elements of order $p = 1$ as macroscopic space discretization. As predicted in Theorem 3.4.13, we obtain first order convergence till we reach the plateau of the space discretization of the cell problems.

In Figure 3.1b we compare the implicit and explicit midpoint rule with the IMEX scheme. For the macroscopic space discretization, we use piecewise quadratic elements with a fixed $H = 2^{-7}$. In addition, we use the formula (3.5.4) for the homogenized coefficient here, since we are focusing on the time discretization error. One can observe the predicted convergence rate for the IMEX scheme as well as the second-order convergence

for the implicit midpoint rule until the spatial discretization error is reached. The explicit midpoint rule is only stable under a strong CFL condition and the error immediately reaches the plateau of the spatial discretization error.

Data availability The codes to reproduce the experiments are available on https://doi.org/10.35097/gkHSEiWVXPBkLPxL.

CHAPTER 4

Matrix function approximation using a Chebyshev-based method

The content of the following chapter was published in [51]. The authors of this paper are Daniel Eckhardt, Marlis Hochbruck and Volker Grimm.

We present a Chebyshev-based method for approximating matrix functions or products of matrix functions with vectors. Our main interest is in matrix functions that arise in exponential integrators. The approach builds upon the construction of a Faber expansion of the function on an ellipse that encloses the field-of-values of the matrix. We derive error bounds for the proposed approximations and provide an efficient residual-based error estimator for the product of a matrix function with a vector, that can be computed efficiently using short recurrences. Since Faber polynomials rely on a priori knowledge of the field-of-values of the matrix, we propose a novel algorithm to determine a suitable ellipse from appropriate Ritz values. Numerical examples demonstrate the effectiveness of the proposed method.

4.1 Introduction

The numerical approximation of matrix functions

$$f(\mathbf{S}) \quad \text{or} \quad f(\mathbf{S})v, \qquad \text{where} \qquad \mathbf{S} \in \mathbb{C}^{N\times N}, \quad v \in \mathbb{C}^N,$$

where $f\colon \mathbb{C} \to \mathbb{C}$ is analytic in a neighborhood of the field-of-values of $\mathbf{S}$, plays a fundamental role in a wide range of scientific and engineering applications (e.g., Section 2 in [79]). Our prime interest is in φ_k-functions defined as

$$\varphi_0(z) = e^z, \qquad \varphi_k(z) = \int_0^1 e^{(1-s)z} \frac{s^{k-1}}{(k-1)!}\, ds, \quad k \geq 1\,. \tag{4.1.1}$$

These analytic functions arise in connection with exponential integrators, which are used to solve large systems of ordinary differential equations [87]. Exponential integrators are particularly effective in applications involving stiff or highly oscillatory problems. A common way to approximate matrix functions is to first construct a projection of the original matrix onto a lower-dimensional Krylov subspace. This is done by building a suitable basis of this subspace using the Arnoldi or Lanczos algorithm. The matrix function is then calculated using the significantly smaller projected matrix (cf., e.g., [72, 84, 114]). However, constructing such a basis is computationally expensive, particularly when weighted inner products are required, as it is necessary for matrices resulting from spatial discretizations of differential operators. In addition, even for the Lanczos method, which uses short recurrence, memory requirements can be high, since the entire basis must be stored.

To address these issues, we consider an approach for approximating matrix functions in the spirit of semiiterative methods [53], which relies on approximations of the field-of-values of the matrix. In particular, we employ Faber polynomials associated with suitable ellipses. These polynomials can be computed efficiently via a three-term recurrence, leading to a significant reduction in both computational cost and memory requirements. Since, for an ellipse with distinct semi-axes, Faber polynomials coincide with shifted and scaled Chebyshev polynomials, we refer to the proposed approach as *Chebyshev method*.

Chebyshev and Faber polynomials are widely used for approximating matrix functions, see, e.g., [87, 94, 100]. Error analyses, particularly for the matrix exponential, can be found in [19, 46–48, 84, 109, 119]. In this work, we obtain error bounds that apply to the broader class of φ_k-functions. Our approach builds on the results of [17] and extends them to general inner products and their corresponding norms. This extension is motivated by applications in finite element discretization of partial differential equations,

where discrete Sobolev norms naturally arise (see, e.g., [52, 83]). A popular alternative to Krylov subspace approximations is interpolation at Leja points [32, 34, 93], which provides superlinear convergence rates but also requires a prior information about the field-of-values.

The main contribution of this work is the development of an efficient algorithm for approximating the matrix φ_k-functions using the Chebyshev method, focusing on two main aspects: First, we provide a residual-based error estimator to terminate the Chebyshev method. Second, we develop a method for constructing suitable ellipses by approximating the field-of-values with the convex hull of suitable Ritz values. Our approach employs spectral preconditioning techniques in combination with the Arnoldi algorithm. This idea was originally used to compute eigenvalues with the largest (or smallest) real part, see, e.g., [50, 81, 105, 122]. In the context of exponential integrators, the proposed algorithm can be viewed as a preprocessing step within an exponential integrator using a fixed linearization, since the ellipse needs to be constructed only once and can then be reused at each time step. In contrast to [119], which determines ellipses for specific problems based on empirical results from numerical experiments, our approach is significantly more general.

The paper is structured as follows. In Section 4.2, we present the Chebyshev method for approximating matrix functions and derive the error estimates. In Section 4.3, we introduce a residual-based strategy for selecting the degree of the approximating polynomial. The algorithm for computing the enclosing ellipse is described in Section 4.4. Finally, in Section 4.5, we apply the method to systems of spatially discretized partial differential equations and report numerical results, illustrating its effectiveness compared to other methods.

4.2 Error bounds for truncated Faber/Chebyshev series

We are interested in the approximation of matrix functions for matrices $\mathbf{S}$ whose field-of-values satisfies

$$W(\mathbf{S}) = \big\{(\mathbf{S}v, v) \mid v \in \mathbb{C}^N, \|v\| = 1\big\} \subset K \tag{4.2.1}$$

where K is a compact set bounded by an ellipse of the form

$$\mathcal{E}_\gamma^{\alpha,\beta} = \{z \in \mathbb{C} \mid z = \alpha\cos\eta + \mathrm{i}\,\beta\sin\eta + \gamma, \eta \in [-\pi,\pi]\}, \qquad \alpha,\beta \geq 0,\ \alpha^2+\beta^2 > 0, \text{ and } \gamma \in \mathbb{R}.$$

Here, $\|\cdot\|$ denotes a suitable norm in $\mathbb{C}^N$ which is induced by an inner product $(\cdot,\cdot)$. The induced matrix norm is also designated by $\|\cdot\|$. Let $\mathbf{S}^\dagger$ be the adjoint matrix to $\mathbf{S}$ with respect to $(\cdot,\cdot)$. If $\mathbf{S}$ is normal, i.e., $\mathbf{S}^\dagger\mathbf{S} = \mathbf{S}\mathbf{S}^\dagger$, then the field-of-values is the convex hull of the eigenvalues of $\mathbf{S}$. We will use Faber polynomials with respect to the set K in order to approximate the function. For the convenience of the reader, we review some basic facts about Faber polynomials and Faber series. For more detailed properties of Faber polynomials, we refer to the book by Gaier [59], the survey by Suetin [134], and [53], respectively. Let $\mathbb{D}_\rho = \{z \in \mathbb{C} \;\mid\; |z| \le \rho\}$ be the closed disk with radius ρ and center zero. The mapping

$$\psi(w) = \frac{\alpha+\beta}{2}\, w + \gamma + \frac{\alpha-\beta}{2}\,\frac{1}{w}, \qquad \alpha,\beta \ge 0, \quad \alpha^2+\beta^2 > 0, \quad \gamma \in \mathbb{R}, \tag{4.2.2a}$$

maps the exterior of the unit disc $\mathbb{D}$ to the exterior of the ellipse $\mathcal{E}_\gamma^{\alpha,\beta}$. The map is holomorphic and one-to-one for $|w| > \sqrt{|\alpha-\beta|/(\alpha+\beta)}$ (cf. page 582 in [41]). The boundary of the unit circle is mapped to the ellipse $\mathcal{E}_\gamma^{\alpha,\beta}$, i.e., $\mathcal{E}_\gamma^{\alpha,\beta} = \psi(\partial\mathbb{D})$. The sets

$$K_\rho := \mathbb{C}\backslash\psi(\mathbb{C}\backslash\mathbb{D}_\rho), \qquad \rho \ge 1, \tag{4.2.2b}$$

are compact sets with ellipses as boundaries. We also use the short notation $\mathbb{D} = \mathbb{D}_1$ and $K = K_1$. The coefficient $(\alpha+\beta)/2$ is commonly referred to as the logarithmic capacity of K, p. 51 in [140]. The Faber polynomials F_ℓ, $\ell = 1, 2, \ldots$, for the set K are defined by the generating function

$$\frac{w\psi'(w)}{\psi(w)-z} = 1 + \sum_{\ell=1}^{\infty} F_\ell(z) w^{-\ell},$$

where $z \in K$ and $|w| > 1$. By equating the coefficients, one obtains the three-term recursion

$$\begin{aligned} F_0 &\equiv 2, \qquad F_1(z) = 2\,\frac{z-\gamma}{\alpha+\beta}, \\ F_{\ell+1}(z) &= 2\,\frac{z-\gamma}{\alpha+\beta}F_\ell(z) - \frac{\alpha-\beta}{\alpha+\beta}F_{\ell-1}(z), \qquad \ell = 1,2,\ldots \end{aligned} \tag{4.2.3}$$

It is well known that Faber polynomials for ellipses ($\alpha \neq \beta$) are shifted and scaled Chebyshev polynomials (cf. [17, 41, 134]). More exactly, the Faber polynomials (4.2.3), are given as

$$F_\ell(z) = 2\left(\frac{\alpha-\beta}{\alpha+\beta}\right)^{\frac{\ell}{2}} T_\ell\left(\frac{z-\gamma}{\sqrt{\alpha^2-\beta^2}}\right), \qquad \ell = 1,2,\ldots, \tag{4.2.4}$$

where T_ℓ are the Chebyshev polynomials of the first kind. The branch of the square root needs to be chosen appropriately. For circles, $\alpha = \beta$, the Faber polynomials are shifted

and scaled monomials

$$F_\ell(z) = \left(\frac{z-\gamma}{\alpha}\right)^\ell, \qquad \ell = 1, 2, \dots . \tag{4.2.5}$$

Functions holomorphic in a neighborhood of K can be given as a series of Faber polynomials. The following theorem is inspired by Faber's original ideas (cf. [55, 56]) and Section 5 of [41].

Theorem 4.2.1. *Let F_ℓ be the Faber polynomials for the set $K = K_1$ defined in (4.2.2). Let $\Omega \supset K$ be an open set and f be holomorphic on Ω. Then there is a $\rho^* > 1$ such that $K_\rho \subset \Omega$ for $1 \le \rho < \rho^*$. The Faber series for the function f*

$$f(z) = c_0 + \sum_{\ell=1}^{\infty} c_\ell F_\ell(z), \qquad c_\ell = \frac{1}{2\pi i}\int_{\partial\mathbb{D}} f(\psi(w)) w^{-\ell-1}\, dw\,,$$

converges absolutely and uniformly on all sets K_ρ, $1 \le \rho < \rho^*$. For the truncated series

$$p_m(z) := c_0 + \sum_{\ell=1}^{m} c_\ell F_\ell(z)\,,$$

we have the estimate

$$\|f - p_{m-1}\|_{L^\infty(K)} \le 2\sum_{\ell=m}^{\infty} |c_\ell| \le 2\|f\|_{L^\infty(K_\rho)} \frac{\rho^{-m}}{1-\rho^{-1}}, \qquad \rho^* > \rho > 1\,,$$

where

$$\|f\|_{L^\infty(K)} := \sup_{z\in K} |f(z)|\,.$$

Proof. The existence of $\rho^* > 1$ follows from the fact that Ω is open and K is compact. Pick ρ such that $1 < \rho < \rho^*$. Let z be in the interior of K_ρ. Then, the Cauchy integral theorem gives

$$\begin{aligned} f(z) &= \frac{1}{2\pi i}\int_{\psi(\partial\mathbb{D}_\rho)} f(\xi)\frac{1}{\xi - z}\, d\xi = \frac{1}{2\pi i}\int_{\partial\mathbb{D}_\rho} f(\psi(w))\frac{\psi'(w)}{\psi(w)-z}\, dw \\ &= \frac{1}{2\pi i}\int_{\partial\mathbb{D}_\rho} \frac{f(\psi(w))}{w}\frac{w\psi'(w)}{\psi(w)-z}\, dw = \frac{1}{2\pi i}\int_{\partial\mathbb{D}_\rho} \frac{f(\psi(w))}{w}\left(1 + \sum_{\ell=1}^{\infty} F_\ell(z) w^{-\ell}\right) dw\,. \end{aligned}$$

Since the series is uniformly convergent on $\partial\mathbb{D}_\rho$, the series and the integral can be exchanged. This gives the series with the coefficients

$$c_\ell = \frac{1}{2\pi i}\int_{\partial\mathbb{D}_\rho} f(\psi(w)) w^{-\ell-1}\, dw\,.$$

Since $f\circ\psi$ is holomorphic for $\rho^* > |w| > \rho_*$, $\rho_* < 1$, the contour can be changed to $\partial\mathbb{D}$. For $\alpha \ne 0$ and $\beta \ne 0$, one might choose $\rho_* = \sqrt{|\alpha-\beta|/(\alpha+\beta)}$, for the remaining cases, one might choose $\rho_* = 1/\rho^*$. The coefficient c_ℓ can be estimated by

$$|c_\ell| = \left|\frac{1}{2\pi i}\int_{\partial\mathbb{D}_\rho} f(\psi(w)) w^{-\ell-1}\, dw\right| \le \frac{1}{\rho^\ell}\|f\circ\psi\|_{L^\infty(\partial\mathbb{D}_\rho)} = \frac{1}{\rho^\ell}\|f\|_{L^\infty(K_\rho)}\,,$$

where the last equality follows from the maximum modulus principle. By Bernstein's lemma (cf. Lemma 3.6.2 in [112]) and Theorem 2.1 in [17], we have that

$$\|F_\ell\|_{L^\infty(K_{\tilde\rho})} \le \tilde\rho^\ell \|F_\ell\|_{L^\infty(K)} \le 2\tilde\rho^\ell\,, \qquad \tilde\rho \ge 1\,.$$

Let now $1 \le \tilde\rho < \rho$, then

$$\|c_\ell F_\ell\|_{L^\infty(K_{\tilde\rho})} \le |c_\ell| \|F_\ell\|_{L^\infty(K_{\tilde\rho})} \le 2\left(\frac{\tilde\rho}{\rho}\right)^\ell \|f\|_{L^\infty(K_\rho)}\,.$$

From this, the absolute and uniform convergence on $K_{\tilde\rho}$ follows. For $\tilde\rho = 1$, the estimate of the truncated series on K is deduced. □

Theorem 4.2.1 can be transferred to matrix functions for matrices $\mathbf{S}$ with a field-of-values contained in the set K, i.e. $W(\mathbf{S}) \subset K$. The following result is a corollary of Theorem 4.2.1 and of Theorem 2.1 in [17].

Corollary 4.2.2. *Let F_ℓ be the Faber polynomials for the set K, cf. (4.2.2b). Let $\Omega \supset K$ be an open set and f be holomorphic on Ω. Further, let $\mathbf{S}$ be a matrix such that $W(\mathbf{S}) \subset K$. Then, the matrix series*

$$f(\mathbf{S}) = c_0 I + \sum_{\ell=1}^{\infty} c_\ell F_\ell(\mathbf{S}), \qquad c_\ell = \frac{1}{2\pi i}\int_{\partial\mathbb{D}} f(\psi(w)) w^{-\ell-1}\,dw\,,$$

converges in the operator norm. Here, $\mathbf{I}$ designates the identity matrix. For the truncated series

$$p_m(\mathbf{S}) := c_0\mathbf{I} + \sum_{\ell=1}^{m} c_\ell F_\ell(\mathbf{S})\,,$$

we have the estimate

$$\|f(\mathbf{S}) - p_{m-1}(\mathbf{S})\| \le 2\sum_{\ell=m}^{\infty} |c_\ell|\,,$$

where the last series is convergent with the same bound as in Theorem 4.2.1.

Proof. By Theorem 2.1 in [17], we find the bound

$$\|F_\ell(\mathbf{S})\| \le 2, \qquad \ell = 1, 2, \dots\,,$$

for the Euclidean inner product and its corresponding operator norm. The proof of Theorem 2.1 in [17] can be extended straightforwardly to an arbitrary inner product in $\mathbb{C}^N$, by replacing A^T in the proof with the complex conjugate of the adjoint matrix with respect to the inner product, $\overline{A^\dagger}$, and by replacing the Euclidean inner product with the general one. Hence,

$$\|f(\mathbf{S}) - p_{m-1}(\mathbf{S})\| \le \sum_{\ell=m}^{\infty} |c_\ell| \|F_\ell(\mathbf{S})\| \le 2\sum_{\ell=m}^{\infty} |c_\ell| \le 2\|f\|_{L^\infty(K_\rho)} \frac{\rho^{-m}}{1-\rho^{-1}}\,,$$

where $1 < \rho < \rho^*$ has been chosen according to Theorem 4.2.1. □

For the Faber series of the φ_k-functions, cf. (4.1.1), we obtain the following theorem.

Theorem 4.2.3. *Let* $\mathbf{S}$ *satisfy* (4.2.1) *with* $K = K_1$ *defined in* (4.2.2). *Let* F_ℓ *be the Faber polynomials for the set* K, *cf.* (4.2.2b). *Then, the matrix series*

$$\varphi_k(\mathbf{S}) = c_0^{(k)}\mathbf{I} + \sum_{\ell=1}^{\infty} c_\ell^{(k)} F_\ell(\mathbf{S}), \qquad c_\ell^{(k)} = \frac{1}{2\pi i}\int_{\partial\mathbb{D}} \varphi_k(\psi(w)) w^{-\ell-1}\, dw\,,$$

converges in the matrix norm. For the truncated series

$$p_m^{(k)}(\mathbf{S}) := c_0^{(k)}\mathbf{I} + \sum_{\ell=1}^{m} c_\ell^{(k)} F_\ell(\mathbf{S})\,, \tag{4.2.6}$$

we have the estimates

$$\|\varphi_k(\mathbf{S}) - p_{m-1}^{(k)}(\mathbf{S})\| \le \frac{4\sqrt{\pi}}{\sqrt{2m}}\frac{m!}{(m+k)!} e^{\alpha+\gamma}\left(\frac{(\alpha+\beta)e}{2m}\right)^m, \qquad m \ge \alpha+\beta\,.$$

Proof. For the exponential φ_0, analogously to the proof of Corollary 4.1 in [17], we find the bound

$$|c_\ell^{(0)}| \le \frac{\sqrt{\pi}}{\sqrt{2\ell}} e^{\alpha+\gamma}\left(\frac{(\alpha+\beta)e}{2\ell}\right)^\ell, \qquad \ell > \alpha+\beta\,.$$

With the representation of the coefficients of the Faber series for the φ_k-functions in Example 4.3 in [17], we find for $k \ge 1$ the bounds

$$|c_\ell^{(k)}| \le \frac{\sqrt{\pi}}{\sqrt{2\ell}}\frac{\ell!}{(\ell+k)!} e^{\alpha+\gamma}\left(\frac{(\alpha+\beta)e}{2\ell}\right)^\ell, \qquad \ell > \alpha+\beta\,.$$

The last formula is the correct estimate for $k = 0$, too. For $k = 0, 1, 2, \ldots$ and $m > \alpha+\beta$, we therefore obtain the estimate

$$\begin{aligned}
\sum_{\ell=m}^{\infty} |c_\ell^{(k)}| &\le \frac{\sqrt{\pi}}{\sqrt{2m}}\frac{m!}{(m+k)!} e^{\alpha+\gamma} \sum_{\ell=m}^{\infty}\left(\frac{(\alpha+\beta)e}{2\ell}\right)^\ell \\
&\le \frac{\sqrt{\pi}}{\sqrt{2m}}\frac{m!}{(m+k)!} e^{\alpha+\gamma} \sum_{\ell=m}^{\infty} e^m\left(\frac{(\alpha+\beta)}{2m}\right)^\ell \\
&= \frac{\sqrt{\pi}}{\sqrt{2m}}\frac{m!}{(m+k)!} e^{\alpha+\gamma}\left(\frac{(\alpha+\beta)e}{2m}\right)^m \sum_{\ell=0}^{\infty}\rho^{-\ell} \\
&\le 2\frac{\sqrt{\pi}}{\sqrt{2m}}\frac{m!}{(m+k)!} e^{\alpha+\gamma}\left(\frac{(\alpha+\beta)e}{2m}\right)^m,
\end{aligned}$$

where $\rho = \frac{2m}{\alpha+\beta} > 2$. Corollary 4.2.2 applied to the φ_k-functions now proves the statement of Theorem 4.2.3. □

Theorem 4.2.3 is our main result for the error of the approximation of the φ_k-functions by Faber polynomials. Note that the coefficients of the Faber series are Fourier coefficients of the periodic smooth function $\varphi_k(\psi(\exp(i\theta))) = \varphi_k(\alpha\cos\theta + \mathrm{i}\,\beta\sin\theta + \gamma)$, $\theta \in [-\pi, \pi]$, which can be approximated efficiently by fast Fourier transforms (e.g. [137]). The evaluation of the Faber polynomials is efficient due to the three-term recursion (4.2.3).

4.3 Residual-based error estimator

To achieve the desired accuracy, we propose a strategy for estimating the degree m of the Faber approximation $p_m^{(k)}$ defined in (4.2.6). Although the error bounds in Theorem 4.2.3 provide a theoretical estimate, they are not practical for determining the degree m in real applications. Instead, we employ a residual-based approach, which is more suitable for practical computations. Here, the matrix functions $\varphi_k(\tau\mathbf{S})$ are considered for some time step $\tau > 0$. As in [27, 93], we consider the initial value problems

$$(Y^{(k)})'(\tau) = \begin{cases} \mathbf{S}Y^{(0)}(\tau), & Y^{(0)}(0) = \mathbf{I}, \quad \text{if } k = 0, \\ \mathbf{S}Y^{(k)}(\tau) + \frac{\tau^{k-1}}{(k-1)!}\mathbf{I}, & Y^{(k)}(0) = 0, \quad \text{if } k \geq 1, \end{cases}$$

whose solutions are given by

$$Y^{(k)}(\tau) = \tau^k \varphi_k(\tau\mathbf{S}), \qquad k \geq 0.$$

For the corresponding Faber approximations $Y_m^{(k)}(\tau) = \tau^k p_m^{(k)}(\tau\mathbf{S})$, $k \geq 0$ it can be shown that the residuals defined by

$$\begin{aligned} R_m^{(0)}(\tau) &= \rho_m^{(0)}(\tau) && \text{if } k = 0, \\ R_m^{(k)}(\tau) &= \rho_m^{(k)}(\tau) + \frac{\tau^{k-1}}{(k-1)!} && \text{if } k \geq 1, \\ \rho_m^{(k)}(\tau) &= \mathbf{S}Y_m^{(k)}(\tau) - (Y_m^{(k)})'(\tau) \end{aligned} \tag{4.3.1}$$

can be used to estimate the error $\|Y^{(k)}(\tau) - Y_m^{(k)}(\tau)\|$, and it holds

$$\begin{aligned} R_m^{(0)}(\tau) &= \Big(\mathbf{S}p_m^{(k)}(\tau\mathbf{S}) - \frac{d}{d\tau}p_m^{(k)}(\tau\mathbf{S})\Big), && \text{if } k = 0, \\ R_m^{(k)}(\tau) &= \tau^{k-1}\Big(\tau R_m^{(0)}(\tau) + \frac{\mathbf{I}}{(k-1)!} - kp_m^{(k)}(\tau\mathbf{S})\Big), && \text{if } k \geq 1. \end{aligned} \tag{4.3.2}$$

(cf. [27, 93]). To compute the required quantities, we use the fact that Faber polynomials reduce to shifted and scaled Chebyshev polynomials when ellipses $\mathcal{E}_\gamma^{\alpha,\beta}$ with $\alpha \neq \beta$ are considered (cf. (4.2.4)). Consequently, the same technique as in [27] can be applied, where Chebyshev polynomials of first and second kind are used. We define a weighted version of these polynomials as

$$\widetilde{U}_j = \vartheta^{\frac{j}{2}} U_j, \qquad \vartheta = \frac{\alpha - \beta}{\alpha + \beta}, \tag{4.3.3}$$

where U_j is the standard jth Chebyshev polynomial of second kind (see [27, eq.(3.8)]). A simple calculation shows that $\widetilde{U}_j$ satisfies the three-term recurrence

$$\widetilde{U}_j(z) = 2z\sqrt{\vartheta}\widetilde{U}_{j-1}(z) - \vartheta\widetilde{U}_{j-2}(z), \quad j \geq 2$$

with

$$\widetilde{U}_1(z) = 2\sqrt{\vartheta}z, \quad \widetilde{U}_0(z) = 1, \quad \widetilde{U}_{-1}(z) = 0, \quad \widetilde{U}_{-2}(z) = -\tfrac{1}{\vartheta}.$$

Moreover, the following relations can be shown.

Lemma 4.3.1. *Let*

$$\widetilde{T}_j(z) = \vartheta^{\frac{j}{2}} T_j(z),$$

where T_j is the jth Chebyshev polynomial of first kind. For the weighted Chebyshev polynomials the following identities hold true:

$$(\widetilde{T}_j)'(z) = j\sqrt{\vartheta}\widetilde{U}_{j-1}(z), \tag{4.3.4a}$$

$$z\widetilde{T}_j(z) = \frac{1}{2}(\tfrac{1}{\sqrt{\vartheta}}\widetilde{T}_{j+1}(z) + \sqrt{\vartheta}\widetilde{T}_{j-1}(z)), \tag{4.3.4b}$$

$$z\widetilde{U}_j(z) = \frac{1}{2}(\tfrac{1}{\sqrt{\vartheta}}\widetilde{U}_{j+1}(z) + \sqrt{\vartheta}\widetilde{U}_{j-1}(z)), \tag{4.3.4c}$$

$$\widetilde{T}_j(z) = \frac{1}{2}(\widetilde{U}_j(z) - \vartheta\widetilde{U}_{j-2}(z)), \tag{4.3.4d}$$

Proof. Follows directly from [27, eq.(3.4)–(3.7)] and (4.3.3). □

The following Lemma shows that the residual (4.3.2) can be efficiently approximated using the short-term Chebyshev recurrences.

Lemma 4.3.2. *Let $p_m^{(k)}$ be defined as in Theorem 4.2.3 and*

$$\widehat{\tau\mathbf{S}} = \frac{\tau\mathbf{S} - \gamma\mathbf{I}}{\sqrt{\alpha^2 - \beta^2}}, \qquad \alpha \neq \beta.$$

Then, we have

$$p_m^{(k)}(\tau\mathbf{S}) = c_0^{(k)}\mathbf{I} + \sum_{j=1}^{m} c_j^{(k)}\Big(\widetilde{U}_j(\widehat{\tau\mathbf{S}}) - \frac{\alpha-\beta}{\alpha+\beta}\,\widetilde{U}_{j-2}(\widehat{\tau\mathbf{S}})\Big). \tag{4.3.5}$$

Further, the residual terms in (4.3.2) can be expressed by

$$\tau\frac{d}{d\tau}p_m^{(k)}(\tau\mathbf{S}) = \sum_{j=1}^{m} c_j^{(k)} j\Big(\widetilde{U}_j(\widehat{\tau\mathbf{S}}) + \frac{2\gamma}{\alpha+\beta}\widetilde{U}_{j-1}(\widehat{\tau\mathbf{S}}) + \frac{\alpha-\beta}{\alpha+\beta}\widetilde{U}_{j-2}(\widehat{\tau\mathbf{S}})\Big), \tag{4.3.6}$$

and

$$\begin{aligned}\tau\mathbf{S}p_m^{(k)}(\tau\mathbf{S}) = c_0^{(k)}\tau\mathbf{S} + \sum_{j=1}^{m}\frac{c_j^{(k)}}{2}\Big((\alpha+\beta)\widetilde{U}_{j+1}(\widehat{\tau\mathbf{S}}) + 2\gamma\Big(\widetilde{U}_j(\widehat{\tau\mathbf{S}}) - \frac{\alpha-\beta}{\alpha+\beta}\,\widetilde{U}_{j-2}(\widehat{\tau\mathbf{S}})\Big)\\ + \frac{(\alpha-\beta)^2}{\alpha+\beta}\,\widetilde{U}_{j-3}(\widehat{\tau\mathbf{S}})\Big).\end{aligned} \tag{4.3.7}$$

Proof. By (4.2.4) we have $2\widetilde{T}_j(\widehat{\tau\mathbf{S}}) = F_j(\tau\mathbf{S})$. Then, (4.3.5) follows directly from (4.3.4d). To derive (4.3.6) we use (4.3.4a) and (4.3.4c):

$$\begin{aligned}
\tau\frac{d}{d\tau}p_m^{(k)}(\tau\mathbf{S}) &= 2\sum_{j=1}^{m} c_j^{(k)}\tau\frac{d}{d\tau}\widetilde{T}_j(\widehat{\tau\mathbf{S}}) \\
&= 2\sum_{j=1}^{m} c_j^{(k)}\frac{\tau\mathbf{S}}{\sqrt{\alpha^2-\beta^2}}(\widetilde{T}_j)'(\widehat{\tau\mathbf{S}}) \\
&= 2\sum_{j=1}^{m} c_j^{(k)} j\Big(\sqrt{\vartheta}\widehat{\tau\mathbf{S}}\widetilde{U}_{j-1}(\widehat{\tau\mathbf{S}}) + \frac{\gamma}{\sqrt{\alpha^2-\beta^2}}\sqrt{\vartheta}\widetilde{U}_{j-1}(\widehat{\tau\mathbf{S}})\Big) \\
&= 2\sum_{j=1}^{m} c_j^{(k)} j\Big(\frac{1}{2}\widetilde{U}_j(\widehat{\tau\mathbf{S}}) + \frac{\gamma}{\alpha+\beta}\widetilde{U}_{j-1}(\widehat{\tau\mathbf{S}}) + \frac{1}{2}\vartheta\widetilde{U}_{j-2}(\widehat{\tau\mathbf{S}})\Big).
\end{aligned}$$

We obtain (4.3.7) by writing

$$\tau\mathbf{S}p_m^{(k)}(\tau\mathbf{S}) = c_0^{(k)}\tau\mathbf{S} + 2\sum_{j=1}^{m} c_j^{(k)}\sqrt{\alpha^2-\beta^2}\frac{\tau\mathbf{S}}{\sqrt{\alpha^2-\beta^2}}\widetilde{T}_j(\widehat{\tau\mathbf{S}})v$$

and using (4.3.4b) and (4.3.4d) to express

$$\begin{aligned}
\frac{\tau\mathbf{S}}{\sqrt{\alpha^2-\beta^2}}\widetilde{T}_j(\widehat{\tau\mathbf{S}}) &= \widehat{\tau\mathbf{S}}\widetilde{T}_j(\widehat{\tau\mathbf{S}}) + \frac{\gamma}{\sqrt{\alpha^2-\beta^2}}\widetilde{T}_j(\widehat{\tau\mathbf{S}}) \\
&= \frac{1}{2}\frac{1}{\sqrt{\vartheta}}\widetilde{T}_{j+1}(\widehat{\tau\mathbf{S}}) + \frac{\gamma}{\sqrt{\alpha^2-\beta^2}}\widetilde{T}_j(\widehat{\tau\mathbf{S}}) + \frac{1}{2}\sqrt{\vartheta}\widetilde{T}_{j-1}(\widehat{\tau\mathbf{S}}) \\
&= \frac{1}{4}\Big(\frac{1}{\sqrt{\vartheta}}\widetilde{U}_{j+1}(\widehat{\tau\mathbf{S}}) - \sqrt{\vartheta}\widetilde{U}_{j-1}(\widehat{\tau\mathbf{S}})\Big) \\
&\quad + \frac{\gamma}{2\sqrt{\alpha^2-\beta^2}}\Big(\widetilde{U}_j(\widehat{\tau\mathbf{S}}) - \vartheta\widetilde{U}_{j-2}(\widehat{\tau\mathbf{S}})\Big) \\
&\quad + \frac{1}{4}\Big(\sqrt{\vartheta}\widetilde{U}_{j-1}(\widehat{\tau\mathbf{S}}) - \vartheta^{\frac{3}{2}}\widetilde{U}_{j-3}(\widehat{\tau\mathbf{S}})\Big) \\
&= \frac{1}{4}\frac{1}{\sqrt{\vartheta}}\widetilde{U}_{j+1}(\widehat{\tau\mathbf{S}}) + \frac{\gamma}{2\sqrt{\alpha^2-\beta^2}}\Big(\widetilde{U}_j(\widehat{\tau\mathbf{S}}) - \vartheta\widetilde{U}_{j-2}(\widehat{\tau\mathbf{S}})\Big) \\
&\quad - \frac{1}{4}\vartheta^{\frac{3}{2}}\widetilde{U}_{j-3}(\widehat{\tau\mathbf{S}}),
\end{aligned}$$

which leads to

$$\begin{aligned}
&2\sum_{j=1}^{m} c_j^{(k)}\sqrt{\alpha^2-\beta^2}\frac{\tau\mathbf{S}}{\sqrt{\alpha^2-\beta^2}}\widetilde{T}_j(\widehat{\tau\mathbf{S}}) \\
=&2\sum_{j=1}^{m}\frac{c_j^{(k)}}{4}\sqrt{\alpha^2-\beta^2}\Big(\frac{1}{\sqrt{\vartheta}}\widetilde{U}_{j+1}(\widehat{\tau\mathbf{S}}) + \frac{2\gamma}{\sqrt{\alpha^2-\beta^2}}\Big(\widetilde{U}_j(\widehat{\tau\mathbf{S}}) - \vartheta\widetilde{U}_{j-2}(\widehat{\tau\mathbf{S}})\Big) - \vartheta^{\frac{3}{2}}\widetilde{U}_{j-3}(\widehat{\tau\mathbf{S}})\Big) \\
=&\sum_{j=1}^{m}\frac{c_j^{(k)}}{2}\Big((\alpha+\beta)\widetilde{U}_{j+1}(\widehat{\tau\mathbf{S}}) + 2\gamma\big(\widetilde{U}_j(\widehat{\tau\mathbf{S}}) - \frac{\alpha-\beta}{\alpha+\beta}\widetilde{U}_{j-2}(\widehat{\tau\mathbf{S}})\big) - \frac{(\alpha-\beta)^2}{\alpha+\beta}\widetilde{U}_{j-3}(\widehat{\tau\mathbf{S}})\Big).
\end{aligned}$$

This completes the proof. □

In order to estimate the error of the Faber approximation to $\varphi_k(\tau\mathbf{S})v$ for a vector $v \in \mathbb{C}^N$, we compute $R_m^{(k)}(\tau)v$ defined in (4.3.2) using the short-term recurrence of Lemma 4.3.2. Note that although more vectors must be stored, the additional computational effort increases only moderately, since these vectors can also be reused for the evaluation of (4.3.5). Finally, [27] provides pseudocode for a related but simpler case, in which only the matrix exponential is considered and where $\mathbf{S}$ is (skew-)symmetric.

4.4 Efficient calculation of an optimal ellipse

We seek a *best-fitting* ellipse $\mathcal{E}$ that encloses the field-of-values $W(\mathbf{S})$ in order to provide a reliable approximation, but which is also "small" enough to guarantee convergence with optimal order. In this section we restrict ourselves to real matrices $\mathbf{S}$. Since the field-of-values is generally difficult to compute, we instead approximate it by the convex hull of a few "outer" Ritz values which are computed by a preconditioned Arnoldi iteration. Note that this is a preprocessing step which is necessary for all semiiterative methods, see, e.g., [53]. If $\mathbf{S}$ is nearly normal, using an ellipse computed from appropriate Ritz values in the method of Section 4.2 provides a reliable approximation, as demonstrated in the numerical experiments in Section 4.5.

4.4.1 Construction of the best-fitting ellipse from a discrete complex set

We begin by constructing a best-fitting ellipse for a finite set of points M in the complex plane. Following the approach of [103], we assume w.l.o.g. that $M \subset \mathbb{C}_+ = \{z \in \mathbb{C} \mid \mathrm{Re} z > 0\}$. Although our primary interest is in the case where all points lie in the left half-plane, the method can be easily adapted by first reflecting the points across the imaginary axis, computing the ellipse $\mathcal{E}_\gamma^{\alpha,\beta}$ in the right half-plane, and return $\mathcal{E}_{-\gamma}^{\alpha,\beta}$.

4.4.1.1 Notation and assumptions

We assume that M is symmetric with respect to the real axis. The linear eccentricity of the ellipse $\mathcal{E}_\gamma^{\alpha,\beta}$ is defined as $\epsilon = \sqrt{\alpha^2 - \beta^2} \in \mathbb{C}$. Typically, the linear eccentricity is defined as a real quantity. However, in our setting we allow ϵ to be complex in order to distinguish between the two possible cases that may arise: the ellipse is *horizontal*, with

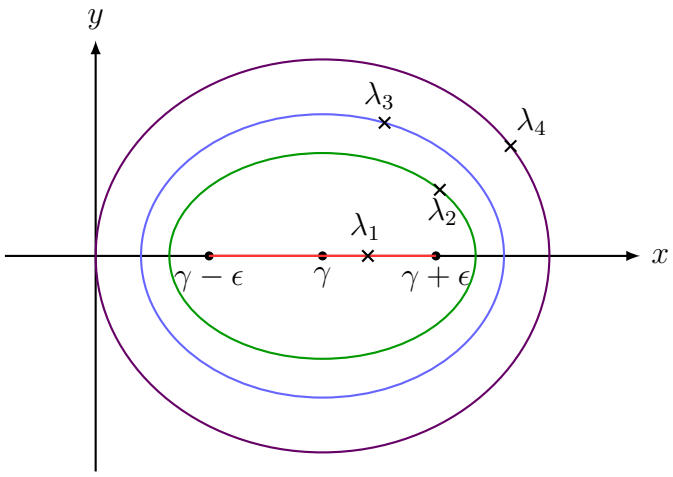


Figure 4.1: Ellipses from $\mathcal{F}_\gamma^\epsilon$, each uniquely determined by the corresponding λ_i. The ellipse determined by λ_4 is the member of the family that passes through the origin.

the major semi-axis α parallel to the real axis, in which case $\epsilon \in \mathbb{R}_{\geq 0}$, or the ellipse is *vertical*, with the major semi-axis β parallel to the imaginary axis, in which case $\epsilon \in i\,\mathbb{R}_{>0}$.

Note that these ellipses can be parametrized either by the α, β, and γ or ϵ, γ, and an arbitrary point λ on the ellipse. Therefore, we define the family of ellipses (see Figure 4.1) with the same linear eccentricity and center as $\mathcal{F}_\gamma^\epsilon = \{\mathcal{E}_\gamma^{\alpha,\beta} \mid \epsilon = \sqrt{\alpha^2 - \beta^2}\}$.

4.4.1.2 The Min-Max problem

As proposed in [103] we consider the map

$$s(\epsilon, \gamma, \lambda) = \left| \frac{(\gamma - \lambda) + ((\gamma - \lambda)^2 - \epsilon^2)^{1/2}}{\gamma + (\gamma^2 - \epsilon^2)^{1/2}} \right|, \tag{4.4.1}$$

which satisfies the following properties: For two ellipses $\mathcal{E}_1, \mathcal{E}_2 \in \mathcal{F}_\gamma^\epsilon$ and two points $\lambda_1 \in \mathcal{E}_1$ and $\lambda_2 \in \mathcal{E}_2$ it holds that

$$\begin{aligned} s(\epsilon, \gamma, \lambda_1) > s(\epsilon, \gamma, \lambda_2) \quad &\iff \mathcal{E}_1 \text{ encloses } \mathcal{E}_2, \\ s(\epsilon, \gamma, \lambda_1) = s(\epsilon, \gamma, \lambda_2) \quad &\iff \mathcal{E}_1 = \mathcal{E}_2, \\ s(\epsilon, \gamma, \lambda_1) = 1 \quad &\iff 0 \in \mathcal{E}_1, \end{aligned} \tag{4.4.2}$$

(cf. [103, eq. (2.13) and (3.1)]). These properties lead two important observations. First, let ϵ and γ be fixed, and let s be maximized over $\lambda \in M$. Then the resulting ellipse encloses all points in M.

Second, we can rewrite (4.4.1) as follows. Let $\mathcal{E}_\gamma^{\alpha,\beta} \in \mathcal{F}_\gamma^\epsilon$ be an ellipse with $\lambda \in \mathcal{E}_\gamma^{\alpha,\beta}$ for some $\lambda \in M$. Furthermore, let $\mathcal{E}_\gamma^{\alpha_0,\beta_0} \in \mathcal{F}_\gamma^\epsilon$ be the ellipse that passes through the origin,

i.e., $0 \in \mathcal{E}_\gamma^{\alpha_0,\beta_0}$ (see Figure 4.1). By (4.4.2) and [81, Eq. (7)], we have

$$s(\epsilon,\gamma,\lambda) = s(\epsilon,\gamma,\alpha+\gamma) = \frac{\alpha+\beta}{\alpha_0+\beta_0}, \tag{4.4.3}$$

where

$$\beta = \sqrt{\alpha^2-\epsilon^2}, \quad \alpha_0 = |\gamma|, \quad \beta_0 = \sqrt{\alpha_0^2-\epsilon^2}.$$

Thus, minimizing (4.4.1) with respect to ϵ and γ leads, heuristically speaking, to an ellipse with a small value of $\alpha+\beta$ and a large value of $\alpha_0+\beta_0$. The latter means that the ellipse is as far as possible into the right half-plane, while still keeping λ on the ellipse.

This motivates the following definition of a *best-fitting* ellipse for M (cf. [103, Eq. (3.2)]).

Definition 4.4.1 (Best-fitting ellipse)**.** Let M be a finite set in $\mathbb{C}_+$. Then a *best-fitting* ellipse is a solution $\mathcal{E}_{\gamma^\star}^{\alpha^\star,\beta^\star} \in \mathcal{F}_{\gamma^\star}^{\epsilon^\star}$ of the min-max problem

$$\min_{(\epsilon,\gamma)\in R} \max_{\lambda\in M} s(\epsilon,\gamma,\lambda) = s(\epsilon^\star,\gamma^\star,\lambda^\star), \quad \text{for all } \lambda^\star \in \mathcal{E}_{\gamma^\star}^{\alpha^\star,\beta^\star}, \tag{4.4.4}$$

where $R := \Big((\mathbb{R}_{\geq 0} \cup \mathrm{i}\,\mathbb{R}_{>0}) \times \mathbb{R}\Big) \setminus \{(0,0)\}$.

Lemma 4.4.2. *Let M be a finite set of points in $\mathbb{C}_+$ for which the assumptions of Section 4.4.1.1 hold. If there exists a solution $\mathcal{E}_{\gamma^\star}^{\alpha^\star,\beta^\star}$ of the min-max problem (4.4.4), then $\mathcal{E}_{\gamma^\star}^{\alpha^\star,\beta^\star}$ encloses all points of M. Further, $\mathcal{E}_{\gamma^\star}^{\alpha^\star,\beta^\star} \subset \mathbb{C}_+$ is optimal in the sense that no ellipse enclosing all points of M attains a smaller value for (4.4.3).*

Proof. By the assumption, there exist $\gamma > 0$ and $\epsilon \in \mathbb{C}$ such that the ellipse $\tilde{\mathcal{E}} \in \mathcal{F}_\gamma^\epsilon$ which is contained in the right half-plane and encloses all points in M. We can assume that at least one point $\widetilde{\lambda} \in M$ lies on the boundary of this ellipse. For all $\lambda \in M$ there exists $\mathcal{E}_\lambda \in \mathcal{F}_\gamma^\epsilon$ with $\lambda \in \mathcal{E}_\lambda$ given by (cf. Lemma 4.6.2),

$$\mathcal{E}_\lambda = \begin{cases} \{\epsilon\cosh\omega_\lambda\cos\psi_\lambda + \mathrm{i}\,\epsilon\sinh\omega_\lambda\sin\psi_\lambda + \gamma \quad | \, \omega_\lambda \in \mathbb{R}_{\geq 0}, \psi_\lambda \in [0,2\pi]\}, & \text{if } \epsilon \in \mathbb{R}, \\ \{|\epsilon|\sin\omega_\lambda\cos\psi_\lambda + \mathrm{i}\,|\epsilon|\cosh\omega_\lambda\sin\psi_\lambda + \gamma \mid \omega_\lambda \in \mathbb{R}_{\geq 0}, \psi_\lambda \in [0,2\pi]\}, & \text{if } \epsilon \in \mathrm{i}\,\mathbb{R}. \end{cases}$$

Since all λ are contained in $\tilde{\mathcal{E}} = \mathcal{E}_{\widetilde{\lambda}}$, it holds that $\omega_\lambda \leq \omega_{\widetilde{\lambda}}$. Therefore, all $\mathcal{E}_\lambda$ are enclosed by $\widetilde{\mathcal{E}}$. By (4.4.2) it follows that

$$s(\epsilon,\gamma,\lambda) \leq s(\epsilon,\gamma,\widetilde{\lambda}) < 1$$

for all $\lambda \in M$. Then, for $\mathcal{E}_{\gamma^\star}^{\alpha^\star,\beta^\star}$ with eccentricity $\epsilon^\star$ and center $\gamma^\star$ and $\lambda^\star \in M$ with $\lambda^\star \in \mathcal{E}_{\gamma^\star}^{\alpha^\star,\beta^\star}$ it holds

$$s(\epsilon^\star,\gamma^\star,\lambda) \leq s(\epsilon^\star,\gamma^\star,\lambda^\star) \leq s(\epsilon,\gamma,\lambda^\star) < 1 \tag{4.4.5}$$

for all $\lambda \in M$. Using (4.4.2) again, $\mathcal{E}_{\gamma^\star}^{\alpha^\star,\beta^\star}$ enclosed all $\lambda \in M$ by the first inequality in (4.4.5) and $\mathcal{E}_{\gamma^\star}^{\alpha^\star,\beta^\star}$ lies in the right half-plane.

Finally, for any ellipse $\widehat{\mathcal{E}} \in \mathcal{F}_\gamma^\epsilon$ with $\widehat{\lambda} \in \widehat{\mathcal{E}}$ that encloses M, we have

$$s(\epsilon^\star, \gamma^\star, \lambda^\star) \leq s(\epsilon, \gamma, \lambda^\star) \leq s(\epsilon, \gamma, \widehat{\lambda}).$$

Since $\widehat{\mathcal{E}}$ was arbitrary $\mathcal{E}_{\gamma^\star}^{\alpha^\star, \beta^\star}$ is the ellipse with the smallest value (4.4.3) that enclose M. □

In [103] it was shown that there exists a solution to the min–max problem (4.4.4) on a compact set if we minimize with respect to $\widetilde{R} = \{(\epsilon, \gamma) \in R \mid \gamma > 0,\ \gamma^2 > \epsilon^2\}$ (see also [102, Chp. 4] for more details).

4.4.1.3 Finding the ellipse

In the following, we present how to construct the *best-fitting* ellipses by explicitly solving the corresponding min-max problems. The calculations can be found in [102]. Because the ellipses are symmetric with respect to the real axis, λ is contained in the ellipse if and only if $\overline{\lambda}$ is also contained in the ellipse. Consequently, the min-max problem reduces to a minimization with respect to

$$M_+ := \{\lambda \in M \mid \mathrm{Im}\lambda \geq 0\}.$$

As described in [103], the following three cases must be considered.

Case 1: Exactly one point $\lambda^\star \in M_+$ is on the boundary of the best-fitting ellipse:
In this case, we have

$$\min_{(\epsilon,\gamma)\in R} \max_{\lambda \in M_+} s(\epsilon, \gamma, \lambda) = \min_{(\epsilon,\gamma)\in R} s(\epsilon, \gamma, \lambda^\star).$$

The *best-fitting* (degenerate) ellipse is

$$\alpha = 0, \quad \beta^2 = \mathrm{Im}^2\lambda^\star, \quad \gamma = \mathrm{Re}\lambda^\star.$$

(cf. [103]). This means that all points in M have the same real part and $\lambda^\star$ is the point with the largest imaginary part.

Case 2: Exactly two points $\lambda_1, \lambda_2 \in M_+$ are on the boundary of the best-fitting ellipse:
In this case, we have

$$\min_{(\epsilon,\gamma)\in R} \max_{\lambda \in M_+} s(\epsilon, \gamma, \lambda) = \min_{(\epsilon,\gamma)\in R} s(\epsilon, \gamma, \lambda_1) = \min_{(\epsilon,\gamma)\in R} s(\epsilon, \gamma, \lambda_2).$$

We assume that $\mathrm{Re}\lambda_1 < \mathrm{Re}\lambda_2$. Further, we define

$$\begin{aligned} A &= \frac{\mathrm{Re}\lambda_2 - \mathrm{Re}\lambda_1}{2}, \quad B = \frac{\mathrm{Re}\lambda_2 + \mathrm{Re}\lambda_1}{2}, \\ V &= \frac{\mathrm{Im}\lambda_2 - \mathrm{Im}\lambda_1}{2}, \quad W = \frac{\mathrm{Im}\lambda_2 + \mathrm{Im}\lambda_1}{2}. \end{aligned} \tag{4.4.6}$$

By the assumptions of this section, we have $A, B, W \geq 0$.

i) If $V = 0$, according to [103], we obtain

$$\gamma = B, \qquad \epsilon^2 = \frac{\alpha^2(\alpha^2 - (A^2 + W^2))}{\alpha^2 - A^2}, \tag{4.4.7}$$

by inserting λ_1 and λ_2 into equation

$$\frac{(\mathrm{Re}\lambda - \gamma)^2}{\alpha^2} + \frac{\mathrm{Im}^2\lambda}{\alpha^2 - \epsilon^2} = 1. \tag{4.4.8}$$

It remains to calculate α by minimizing the constraint function (4.4.1). By [103] we obtain that α^2 is the only real root of $q(z)$ with

$$q(z) = q_1 z^3 + q_2 z^2 + q_3 z + q_4,$$

where the coefficient q_i, $i = 1 \ldots 4$ can be found in [103, eq.(4.8)].

ii) Else if $V \neq 0$, then using again (4.4.8) [103] obtained

$$\begin{aligned} \alpha^2 &= \left(\gamma - \left(B - \tfrac{AW}{V}\right)\right)\left(\gamma - \left(B - \tfrac{AV}{W}\right)\right), \\ \epsilon^2 &= \frac{\left(\gamma - \left(B + \tfrac{VW}{A}\right)\right)\left(\gamma - \left(B - \tfrac{AW}{V}\right)\right)\left(\gamma - \left(B - \tfrac{AV}{W}\right)\right)}{\gamma - B}. \end{aligned} \tag{4.4.9}$$

Note that there is a small typo in [103] in the last equation. It remains to minimize the constraint function (4.4.1) with respect to γ. This leads to

$$z = \gamma - B \text{ is the root of } q(z) \text{ in } \begin{cases} (0, A) & \text{for } V > 0, \\ (-A, 0) & \text{for } V < 0, \end{cases}$$

where

$$q(z) = q_1 z^5 + q_2 z^4 + q_3 z^3 + q_4 z^2 + q_5 z + q_6.$$

Again the coefficients can be found in [103, eq.(4.9)].

Case 3: Three or more pairs in M_+ are on the boundary

We start with the case where we only have three points λ_1, λ_2, and λ_3, where the construction from case 2 does not yield the *best-fitting* ellipse. In this case, we have

$$\min_{(\epsilon,\gamma)\in R} \max_{\lambda\in M_+} s(\epsilon, \gamma, \lambda) = \min_{(\epsilon,\gamma)\in R} s(\epsilon, \gamma, \lambda_1) = \min_{(\epsilon,\gamma)\in R} s(\epsilon, \gamma, \lambda_2) = \min_{(\epsilon,\gamma)\in R} s(\epsilon, \gamma, \lambda_3).$$

We assume that $\mathrm{Re}\lambda_1 < \mathrm{Re}\lambda_2 < \mathrm{Re}\lambda_3$ and

$$(\mathrm{Re}\lambda_2 - \mathrm{Re}\lambda_1)(\mathrm{Im}\lambda_3^2 - \mathrm{Im}\lambda_1^2) < (\mathrm{Re}\lambda_3 - \mathrm{Re}\lambda_1)(\mathrm{Im}\lambda_2^2 - \mathrm{Im}\lambda_1^2). \tag{4.4.10}$$

Condition (4.4.10) ensures that there exists an ellipse such that $\lambda_1, \lambda_2, \lambda_3$ lie on its boundary. By inserting the three points into equation (4.4.8) we obtain a system of three equations with three unknowns α, ϵ, γ. The details can be found in [103, eq. (4.11) and (4.12)]. In the general case with more than three pairs, where the construction from Case 2 does not yield the *best-fitting* ellipse, we must consider all combinations of three pairs of points that satisfy (4.4.10) and selecting the ellipse that minimizes (4.4.1). The algorithm for calculating the method for determining the *best-fitting* ellipse consists of the following steps (see also [103] for more details).

Algorithm 1 Best-fitting ellipse

Input: Set of points $M_+ = \{\lambda_i\}_i$ with $\mathrm{Im}\lambda_i \geq 0$
Output: ellipse parameters α, β, γ

Set $\mathcal{P}_n = \{\Pi \in \mathcal{P}(M_+) \mid |\Pi| = n\}$ for $n = 2, 3$ with $\mathcal{P}(M_+) = \{U \mid U \subset M_+\}$
if All pairs lie on a vertical line **then**
    Set $\lambda = \mathrm{argmax}_{\lambda \in M_+} |\mathrm{Im}\lambda|$
    Calculate α, β, γ by Case 1 w.r.t λ
    return α, β, γ
end if
for $\{\lambda_i, \lambda_j\} \in \mathcal{P}_2$ **do**
    Calculate α, β, γ by Case 2 w.r.t $\{\lambda_i, \lambda_j\}$
    if $\nexists \lambda \in M_+$ outside $\mathcal{E}_\gamma^{\alpha,\beta}$ **then**
        return α, β, γ
    end if
end for
Set $s_{best} = \infty$
for $\{\lambda_i, \lambda_j, \lambda_l\} \in \mathcal{P}_3$ **do**
    if (4.4.10) is true for $\{\lambda_i, \lambda_j, \lambda_l\}$ **then**
        Calculate $\alpha, \beta, \gamma, \epsilon$ by Case 3 w.r.t $\{\lambda_i, \lambda_j, \lambda_l\}$ and set $s_{tmp} = s(\epsilon, \gamma, \lambda_i)$
        if $\nexists \lambda \in M_+$ outside $\mathcal{E}_\gamma^{\alpha,\beta}$ **and** $s_{tmp} < s_{best}$ **then**
            $s_{best} = s_{tmp}$
            $\alpha^{best} = \alpha,\ \beta^{best} = \beta,\ \gamma^{best} = \gamma$
        end if
    end if
end for
return $\alpha^{best}, \beta^{best}, \gamma^{best}$

It is clear that, for the computation of a best-fitting ellipse for a set M, only the outer points are relevant (see Figure 4.2). In the following, we therefore derive a method

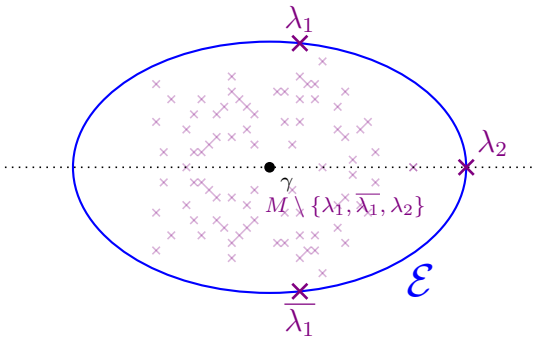


Figure 4.2: Best-fitting ellipse for a set M. For its calculation only λ_1 and λ_2 are required.

for approximating the best-fitting ellipse of the set of eigenvalues by approximating the "outer" eigenvalues with suitable Ritz values.

4.4.2 Arnoldi method

The Arnoldi method is used to construct an orthonormal basis for the Krylov subspace

$$\mathcal{K}_m(\mathbf{S}, \nu) = \operatorname{span}\{\nu, \mathbf{S}\nu, \mathbf{S}^2\nu, \ldots, \mathbf{S}^{m-1}\nu\}.$$

The basis vectors $\{\nu_i\}_i$ of $\mathcal{K}_m(\mathbf{S}, \nu)$ are then stored columnwise in the matrix V_m. The eigenvalues of the projection matrix $H_m = \big((\nu_i, \mathbf{S}\nu_j)\big)_{i,j}$ are Ritz values (w.r.t. $\mathcal{K}_m(\mathbf{S}, \nu)$) and approximate the eigenvalues of the original matrix $\mathbf{S}$ (see, e.g., [36, 127]). Here, $(\cdot, \cdot)$ denotes the scalar product used in the method (see Section 4.2). We are interested in approximating eigenvalues, which lie on the outmost part of the spectrum (see Figure 4.2). The Arnoldi algorithm naturally approximates these eigenvalues (see, e.g. [127, p.156 ff]). However, the convergence speed of the Arnoldi method strongly depends on the distribution of the eigenvalues. If the eigenvalues are clustered, convergence can be very slow. In addition, we are looking for a stopping criterion that will allow us to determine whether we have found enough Ritz values for a good approximation of the best-fitting ellipse. We will address this issues in the next sections.

4.4.3 Transformation of the spectrum

We follow [122] to improve the convergence of the Arnoldi method. Suppose we are given an initial estimate of the best-fitting ellipse $\mathcal{E}$ that encloses all eigenvalues of $\mathbf{S}$ except for one eigenvalue λ. Our goal is to construct a polynomial p such that $|p(\lambda)|$ is large and $|p|$ is small on all other eigenvalues. We then apply the Arnoldi method to the polynomial $p(\mathbf{S})$.

By the Arnoldi method w.r.t. $p(\mathbf{S})$, we obtain a basis $\widehat{V}_m = \{\widehat{\nu}_i\}_i$ of the Krylov subspace $\mathcal{K}_m(p(\mathbf{S}), \nu)$ and an approximation of the corresponding eigenvalue of $\mathbf{S}$ from the projected matrix

$$G_m = \big((\widehat{\nu}_i, \mathbf{S}\widehat{\nu}_j)\big)_{i,j}, \tag{4.4.11}$$

to recompute $\mathcal{E}$. A widely used polynomial in this context is the following (see, e.g., [50, 81, 105, 122, 128]).

Definition 4.4.3. Let $\mathcal{E}_\gamma^{\alpha,\beta}$ be an ellipse and $z_0 \in \mathcal{E}_\gamma^{\alpha,\beta}$. For $\alpha \neq \beta$ and $z_0 \neq \gamma$ we define

$$r_\ell^{\alpha,\beta,\gamma}(z, z_0) := \frac{T_\ell(\iota(z))}{T_\ell(\iota(z_0))}, \qquad \iota(z) = \frac{z-\gamma}{\sqrt{\alpha^2-\beta^2}}. \tag{4.4.12}$$

We obtain by [122] the following relation for efficiently computing $r_\ell^{\alpha,\beta,\gamma}$:

$$\begin{aligned} &r_{\ell+1}^{\alpha,\beta,\gamma}(z, z_0) = 2\sigma_{\ell+1}\iota(z) r_\ell^{\alpha,\beta,\gamma}(z, z_0) - \sigma_\ell\sigma_{\ell+1} r_{\ell-1}^{\alpha,\beta,\gamma}(z, z_0), \quad \ell \geq 1, \\ &\sigma_{\ell+1} = \frac{1}{\iota(z_0) - \sigma_\ell}, \qquad \sigma_1 = \frac{1}{\iota(z_0)}, \qquad r_0^{\alpha,\beta,\gamma}(z, z_0) = 1, \qquad r_1^{\alpha,\beta,\gamma}(z, z_0) = \frac{z-\gamma}{z_0-\gamma}. \end{aligned} \tag{4.4.13}$$

The contour lines of $|r_\ell^{\alpha,\beta,\gamma}(z, z_0)|$ are confocal and concentric ellipses, in particular, the larger ellipses approach circles (see the discussion in [105, p. 307 ff.] and [103, p. 312 ff.]). The reference point z_0 in (4.4.12) determines the behavior of $r_\ell^{\alpha,\beta,\gamma}$, in particular whether $|r_\ell^{\alpha,\beta,\gamma}(z, z_0)|$ is large or small for $z \in W(\mathbf{S})$. This leads to the following result, (see again [105, p. 307 ff.]).

Lemma 4.4.4. *Let $\mathcal{E}_\gamma^{\alpha,\beta}$ be an ellipse with $z_0 \in \mathcal{E}_\gamma^{\alpha,\beta}$ and $\alpha, \beta > 0$. Then,*

$$\lim_{\ell\to\infty} \left| r_\ell^{\alpha,\beta,\gamma}(z, z_0) \right|^{1/\ell} \begin{cases} > 1 & \text{if } z \text{ is outside of } \mathcal{E}_\gamma^{\alpha,\beta}, \\ = 1 & \text{if } z \in \mathcal{E}_\gamma^{\alpha,\beta}, \\ < 1 & \text{if } z \text{ is inside of } \mathcal{E}_\gamma^{\alpha,\beta}, \end{cases}$$

for all $z \in \mathbb{C}$.

Remark 4.4.5. *A similar result holds in the degenerate cases $\alpha = 0$ or $\beta = 0$, which means that z is outside of the degenerate ellipse if and only if $\iota(z) \notin [-1, 1]$. We choose z_0 such that $\iota(z_0) \in \{-1, 1\}$. Then*

$$|r_\ell^{\alpha,\beta,\gamma}(z, z_0)| = |T_\ell(\iota(z))| \quad \text{for all } z \in \mathbb{C}.$$

It is well-known that $T_\ell(z) = \frac{1}{2}\big(\chi(z)^\ell + \chi(z)^{-\ell}\big)$, where χ is the (inverse) Joukowski transformation [100, p.86 ff], which maps the exterior of $[-1, 1]$ to the exterior of the unit disk. Consequently, $\lim_{\ell\to\infty} |r_\ell^{\alpha,\beta,\gamma}(z, z_0)|^{1/\ell} > 1$ if $\iota(z) \notin [-1, 1]$, whereas $|r_\ell^{\alpha,\beta,\gamma}(z, z_0)| \leq 1$ for all $\ell \in \mathbb{N}$ otherwise.

4.4.4 Find best-fitting ellipse for eigenvalues of S

Using the ideas from Section 4.4.3, we propose the following algorithm for calculating the best-fitting ellipse. We begin with an initial guess of the ellipse $\mathcal{E}_\gamma^{\alpha,\beta}$. This can be obtained by approximating a small set of eigenvalues, for example those with the largest magnitude, by standard methods. If eigenvalues with small magnitudes are expected, the origin can also be used to construct the ellipse.

Next, we apply the Arnoldi method to an initial vector ν and $r_\ell^{\alpha,\beta,\gamma}(\mathbf{S}, z)$, where $z \in \mathcal{E}_\gamma^{\alpha,\beta}$ and $\ell \in \mathbb{N}$ and calculate the Ritz values of this matrix. To assess the convergence of the Ritz values, we employ a standard residual-based error estimator [125, Proposition 6.8]. Specifically, at each iteration, we check whether the residual of the Ritz pair corresponding to the Ritz value of the largest magnitude is smaller than some tolerance. Additionally, a maximum number of iterations is enforced.

Once the Arnoldi process terminates, we check whether the ℓth root of norm of the Ritz value of the largest magnitude is less than one. If this condition is satisfied, the current ellipse is accepted as the best-fitting ellipse.

Otherwise, we calculate the Ritz values of $\mathbf{S}$ w.r.t. $\mathcal{K}_m(r_\ell^{\alpha,\beta,\gamma}(\mathbf{S}, z), \nu)$ by calculating the eigenvalues of G_m (see (4.4.11)) and determine whether they approximate eigenvalues of $\mathbf{S}$ (see Algorithm 3 line 9). If so, these are added to the current set of approximated eigenvalues, and a new ellipse is constructed based on the updated set. If no Ritz value converges, we choose as new starting vector for the Arnoldi method the Ritz vector ν corresponding to the maximum Ritz value of this matrix with respect to the function

$$\varrho(\lambda) = \begin{cases} \frac{(\mathrm{Re}\lambda-\gamma)^2}{\alpha^2} + \frac{\mathrm{Im}\lambda^2}{\beta^2}, & \text{if } \alpha, \beta > 0, \\ \frac{(\mathrm{Re}\lambda-\gamma)^2}{\alpha^2}, & \text{if } \beta = 0 \text{ and } \mathrm{Im}\lambda = 0, \\ \frac{\mathrm{Im}\lambda^2}{\beta^2}, & \text{if } \alpha = 0 \text{ and } \mathrm{Re}\lambda = \gamma, \\ 2, & \text{else}, \end{cases} \tag{4.4.14}$$

which indicates how "far" a point λ lies outside the ellipse $\mathcal{E}_\gamma^{\alpha,\beta}$. If two Ritz values have the same value, we choose the one with the Ritz values of larger magnitude. Otherwise, if some Ritz values converge, we choose a random initial vector that is orthogonal to all converged Ritz vectors with respect to $(\cdot,\cdot)$, see Algorithm 3, line 19. Since we consider real matrix $\mathbf{S}$, we can restart the Arnoldi algorithm with $\mathrm{Re}(\nu)$ (see [105, p. 303]). The algorithm is summarized in Algorithm 2 and 3.

Remark 4.4.6. *A more detailed discussion of implementation aspects, in particular the choice of the parameter ℓ_n in Algorithm 3, and additional strategies for improving the convergence speed can be found in Section 4.5.2.*

Algorithm 2 Arnoldi method with spectral preconditioning

Input: Matrix $\mathbf{S} \in \mathbb{R}^{N\times N}$ and initial vector $\widehat{\nu}_1 \in \mathbb{R}^N$ with $\|\widehat{\nu}_1\| = 1$,
Point $z \in \mathbb{R}$, degree $\ell \in \mathbb{N}$, ellipse parameters α, β, γ
Maximum iterations $m_{\max} \in \mathbb{N}$, tolerance $\mathrm{TOL}_{\mathrm{res}} > 0$
Output: $\mu_{\max}$ and Ritz pairs (λ_i, w_i)

Initialize $R = 1$, $m = 1$
while $R > \mathrm{TOL}_{\mathrm{res}}$ **and** $m < m_{\max}$ **do**
  Compute $\eta_m = r_\ell^{\alpha,\beta,\gamma}(\mathbf{S}, z)\widehat{\nu}_m$ by (4.4.13)
  for $j = 1, \ldots, m$ **do**
    $\widehat{h}_{j,m} = (\widehat{\nu}_j, \eta_m)$
  end for
  $v = \eta_m - \sum_{j=1}^{m} \widehat{h}_{j,m}\widehat{\nu}_j$
  $\widehat{h}_{m+1,m} = \|v\|$
  if $\widehat{h}_{m+1,m} = 0$ **then**
    break
  end if
  $\widehat{\nu}_{m+1} = v/\widehat{h}_{m+1,m}$
  Compute eigenpairs $(\mu_i^{(m)}, y_i^{(m)})$, $i = 1, \ldots, m$ of $\widehat{H}_m = (\widehat{h}_{i,j})_{i,j=1}^m$
  Select $(\mu_{\max}^{(m)}, y_{\max}^{(m)})$ such that $|\mu_{\max}^{(m)}| = \max_i |\mu_i^{(m)}|$
  $R = \widehat{h}_{m+1,m}\,|e_m^T y_{\max}^{(m)}|$ and $m \leftarrow m + 1$
end while
Form $\widehat{V}_m = (\widehat{\nu}_1, \ldots, \widehat{\nu}_m)$
Compute eigenpairs $(\lambda_i^{(m)}, x_i^{(m)})$, $i = 1, \ldots, m$ of $G_m = \big((\widehat{\nu}_i, \mathbf{S}\widehat{\nu}_j)\big)_{i,j}$
return $\mu_{\max}^{(m)}$ and $(\lambda_i^{(m)}, \widehat{V}_m x_i^{(m)})$, $i = 1, \ldots, m$

Algorithm 3 Method for calculating the ellipse

Input: Matrix $\mathbf{S} \in \mathbb{R}^{N\times N}$ and initial vector $\nu_1 \in \mathbb{R}^N$ with $\|\nu_1\| = 1$,
degree $\ell_1 \in \mathbb{N}$, initial set M^1, maximum iterations $m_{\max}, n_{\max} \in \mathbb{N}$
Tolerance $\mathrm{TOL}_{\mathrm{eig}}, \mathrm{TOL}_{\mathrm{res}} > 0$, user-defined parameter $\mathrm{EPS} > 0$
Output: Ellipse $\mathcal{E}_\gamma^{\alpha,\beta}$

Compute $\mathcal{E}_{\gamma_1}^{\alpha_1,\beta_1}$ with M^1 and Algorithm 1
Set $z_1 = \max(\mathcal{E}_{\gamma_1}^{\alpha_1,\beta_1} \cap \mathbb{R})$ and $n = 1$ and $W^1 = \{\}$
while $n < n_{\max}$ **do**
  $\mu_n, (\lambda_i^{(n)}, w_i^{(n)}) \leftarrow$ Algorithm 2 with $\left(\mathbf{S}, \nu_n, z_n, \ell_n, \alpha_n, \beta_n, \gamma_n, m_{max}, \mathrm{TOL}_{\mathrm{res}}\right)$
  if $\left|\mu_n\right|^{1/\ell_n} \leq 1$ **then**
    return $\mathcal{E}_{\gamma_n}^{\alpha_n,\beta_n}$
  end if
  for each $(\lambda_i^{(n)}, w_i^{(n)})$ **do**
    if $\|\lambda_i^{(n)} w_i^{(n)} - \mathbf{S} w_i^{(n)}\| \leq \mathrm{TOL}_{\mathrm{eig}} \cdot \|\mathbf{S} w_i^{(n)}\|$ **then**
      $M^{n+1} \leftarrow M^n \cup \{\lambda_i^{(n)}\}$
      $W^{n+1} \leftarrow W^n \cup \{w_i^{(n)}\}$
    end if
  end for
  Compute $\mathcal{E}_{\gamma_{n+1}}^{\alpha_{n+1},\beta_{n+1}}$ with M^{n+1} and Algorithm 1
  if $\alpha_{n+1} = \beta_{n+1}$ **then**
    $\beta_{n+1} \leftarrow \beta_{n+1} + \mathrm{EPS}$
  end if
  if $|M^{n+1}| > |M^n|$ **then**
    ν_{n+1} is a random unit vector that is orthogonal to all $w \in W^{n+1}$ w.r.t. $(\cdot,\cdot)$
  else
    $\omega = w_i^{(n)}$ with $i = \mathrm{argmax}_i\, |\varrho(\lambda_i^{(n)})|$
    $\nu_{n+1} = \mathrm{Re}(\omega)/\|\omega\|$
  end if
  Determine new ℓ_{n+1} and set $z_{n+1} = \max(\mathcal{E}_{\gamma_{n+1}}^{\alpha_{n+1},\beta_{n+1}} \cap \mathbb{R})$ and $n \leftarrow n+1$
end while

4.5 Application for discrete semilinear damped wave equation

This section describes how the theory from the previous sections can be applied to systems of ordinary differential equations (ODEs) arising from space discretizations of semilinear damped wave equations. First, in Section 4.5.1, we introduce a second-order model problem. Rewriting this in first-order form leads to a system matrix $\mathbf{S}$, to which we apply the proposed methods to approximate the matrix φ_k-functions, as required when applying exponential integrators to the model problem. We focus here on the matrix exponential. In Section 4.5.2, we provide details of the implementation. Furthermore, we compare our method with other approaches described in Section 4.5.3 and present numerical experiments in Section 4.5.4.

4.5.1 Model problem

We consider the following semilinear damped wave equation in a domain $\Omega \subset \mathbb{R}^d, d = 1, 2, 3$ with Dirichlet boundary conditions

$$\begin{aligned} \partial_{tt} u(t,x) - a\Delta u(t,x) + \mathcal{B}\partial_t u(t,x) &= \mathcal{G}(u, \partial_t u(t,x)), \quad t > 0, \\ u(0) = u_0, \qquad \partial_t u(0) &= v_0, \end{aligned} \tag{4.5.1}$$

where $a > 0$, $\mathcal{B}$ is a linear differential operator and $\mathcal{G}$ is nonlinear and locally Lipschitz continuous form $H^1(\Omega) \times L^2(\Omega) \to L^2(\Omega)$ and u_0, v_0 sufficiently smooth. We consider two different operators $\mathcal{B}$ in the following

$$\mathcal{B} = -\delta\Delta, \tag{4.5.2a}$$

$$\mathcal{B} = \delta b^T \cdot \nabla \tag{4.5.2b}$$

with $\delta > 0$ and $b(x) \in (C^1(\mathbb{R}^d))^d$. The operator (4.5.2a) appears in, e.g., viscoelasticity, structural mechanics, and thermoelasticity and represents strong/structural damping (cf. [24, 101]). On the other hand (4.5.2b) arises in aeroacoustics [23, 37, 45], and moving-medium wave propagation [15]. Discretizing (4.5.1) using linear finite elements yields a system of ODEs of the form

$$\begin{aligned} \mathbf{M}\boldsymbol{\mu}''(t) + \mathbf{A}\boldsymbol{\mu}(t) + \mathbf{B}\boldsymbol{\mu}'(t) &= \mathbf{G}(\boldsymbol{\mu}(t), \boldsymbol{\mu}'(t)), \quad t > 0, \\ \boldsymbol{\mu}(0) = \boldsymbol{\mu}^0, \quad \boldsymbol{\mu}'(0) &= \boldsymbol{\nu}^0, \end{aligned} \tag{4.5.3}$$

where $\mathbf{M}, \mathbf{A}, \mathbf{B} \in \mathbb{R}^{N\times N}$, $\mathbf{M}, \mathbf{A}$ are symmetric and positive definite, and $\mathbf{G} : \mathbb{R}^N \times \mathbb{R}^N \to \mathbb{R}^N$ is locally Lipschitz continuous w.r.t both arguments. Rewriting (4.5.3) in first-order

formulation gives

$$\begin{aligned} \boldsymbol{x}' - \mathbf{S}\boldsymbol{x} &= \mathbf{F}(\boldsymbol{x}), \\ \boldsymbol{x}(0) &= \boldsymbol{x}^0, \end{aligned} \tag{4.5.4}$$

where

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{\mu} \\ \boldsymbol{\nu} \end{bmatrix}, \qquad \mathbf{S} = \begin{bmatrix} 0 & \mathbf{I} \\ -\mathbf{M}^{-1}\mathbf{A} & -\mathbf{M}^{-1}\mathbf{B} \end{bmatrix},$$
$$\boldsymbol{x}^0 = \begin{bmatrix} \boldsymbol{\mu}^0 \\ \boldsymbol{\nu}^0 \end{bmatrix}, \qquad \mathbf{F}(\boldsymbol{x}) = \begin{bmatrix} 0 \\ \mathbf{M}^{-1}\mathbf{G}(\boldsymbol{\mu}, \boldsymbol{\nu}) \end{bmatrix}.$$

Note that if $\mathbf{B} = 0$ the matrix $\mathbf{S}$ is skew-symmetric w.r.t. the inner product

$$\langle \cdot, \cdot \rangle_{\mathbf{A}\times\mathbf{M}} = \left(\cdot, \begin{bmatrix} \mathbf{A} & 0 \\ 0 & \mathbf{M} \end{bmatrix} \cdot \right), \tag{4.5.5}$$

where $\langle \cdot, \cdot \rangle$ denotes here the Euclidean inner product in $\mathbb{C}^{2N}$. This choice is natural, as it represents a discrete analogue of the $H^1 \times L^2$ inner product, which is commonly used in this setting [52, 83].

When multiplying with $\mathbf{S}$, we solve a linear system involving $\mathbf{M}$ using the CG method with a tolerance of $\mathrm{TOL}_{\mathrm{sol}} = 0.01 \cdot \mathrm{TOL}_R$, where TOL_R is a tolerance used for the error estimator (Section 4.3). This tolerance should be determined by the ODE solver, for example through local error estimators. We use an incomplete Cholesky factorization without fill-in (IC(0)) factorization as a preconditioner (cf. [126, Chp. 10]). Although the mass matrix $\mathbf{M}$ is generally well-conditioned, the frequent multiplications of $\mathbf{S}$, when applying our method, requires repeatedly solving such linear systems. It is therefore advantageous to compute the preconditioner once and reuse it throughout the computation. In our numerical experiments, we find that the number of CG iterations is reduced to only 1 or 2 when using IC(0) factorization, whereas without preconditioning it ranges from 5 to 20 depending on the chosen tolerance.

4.5.2 Implementation details of Algorithms 2 and 3

We now describe the details of the Algorithms proposed in Section 4.4.4. We begin by selecting a random initial vector ν_1 with $\|\nu_1\| = 1$, generated using a uniform distribution on $(0, 1)$. As the first set M^1, we choose $\{\lambda^\star, 0\}$, where $\lambda^\star$ is an approximation of the eigenvalue with the largest magnitude of $\mathbf{S}$, calculated with the C++ library *Spectra*

https://spectralib.org/. Based on this set, the initial ellipse is constructed using Algorithm 1. We stop the algorithm if $\left|\mu_n\right|^{1/\ell_n} \leq 1 + \mathrm{TOL}_{\mathrm{stop}}$ (see Algorithm 3, line 5), where $\mathrm{TOL}_{\mathrm{stop}}$ is a small tolerance that ensures numerical errors in computation do not destabilize the algorithm.

In the following experiments we choose $\mathrm{TOL}_{\mathrm{stop}} = \mathrm{TOL}_{\mathrm{eig}} = 10^{-4}$. The initial polynomial degree is set to $\ell_0 = \sqrt{\lfloor \epsilon \rfloor}$, where ϵ denotes the linear eccentricity of the initial ellipse. Based on numerical experiments, this appears to be a good choice in many cases. $n_{\max}$ is set to 30.

Error estimator of Arnoldi method For the error estimator in the Arnoldi method, we use a moderate tolerance $\mathrm{TOL}_{\mathrm{res}} = 10^{-2}$ to keep the number of iterations low while still achieving good convergence. We set the maximum number of iterations to $m_{\max} = 200$.

Degree ℓ of $r_\ell(z, \cdot)$ We calculate the degree ℓ_n by the following procedure, which is inspired by [128, eq.(23),(24)]. In the first steps, we take into account that our ellipses may not be good, which means that there are eigenvalues that are likely to lie far outside. Therefore, we do not need a high polynomial degree, as these converge relatively quickly. This is because the convergence rate to an eigenvalue λ outside of the current ellipse $\mathcal{E}$ depends on the "gap" between $\mathcal{E}$ and an ellipse, which passes through λ and has the same center and linear eccentricity as $\mathcal{E}$ (see [105, Theorem 4.1]). A larger gap leads to better convergence. Later, the polynomial degree must be increased to ensure good convergence of eigenvalues that are not so far outside. We therefore choose

$$\ell_{n+1} = \max\left\{ \left| \frac{\ln(\mathrm{TOL}_{\mathrm{stop}})}{2\ln(\mu_n)} \right| + 10, \ell_n \right\},$$

where μ_n is computed in Algorithm 3 line 5. Close to convergence, $\ln(\mu_n)$ can become very small, causing ℓ_n to grow unnecessarily large. For this reason, we propose that if $\mu_n - 1 < 100 \cdot \mathrm{TOL}_{\mathrm{stop}}$ then

$$\ell_{n+1} = 1 + \left(0.1 \left| \ln\left(\frac{\mu_n}{1 + \mathrm{TOL}_{\mathrm{stop}}} \right) \right| + 1\right) \ell_n,$$

which ensures moderate growth while avoiding an excessive number of matrix-vector multiplications. We also set a maximum degree $\ell_{\max}$ to prevent excessive computational effort. In the numerical experiments, we choose $\ell_{\max} = 1000$.

Further strategies To improve the convergence, we propose several strategies. Firstly, if the maximum degree $\ell_{\max}$ is reached and the algorithm has not yet converged, we reduce the tolerance $\mathrm{TOL}_{\mathrm{res}}$ by the factor 0.1.

In practice, the ellipse can become quite large when the eigenvalues lie close to the imaginary axis. To avoid this, we shift all computed eigenvalues by 1. After completing the calculation, we shift the resulting ellipse back.

Since numerical instabilities can arise when the semi-axes have nearly equal lengths, we slightly increase one of the semi-axes in this case. In the numerical experiments, we increase β by 10^{-9} if $\left|\alpha - \beta\right| < 10^{-10}$.

4.5.3 Krylov subspace methods

We compare our method with two classical approaches for computing matrix functions: the standard Krylov subspace method and the shift-and-invert method with shift $s \in \mathbb{R}$.

Standard Krylov subspace method Similar as in Section 4.4.2 we can use the Arnoldi method to construct an orthonormal basis $V_m \in \mathbb{R}^{2N\times m}$ (w.r.t. $(\cdot,\cdot)_{\mathbf{A}\times\mathbf{M}}$) and a projection matrix $H_m \in \mathbb{R}^{m\times m}$ of the Krylov subspace $\mathcal{K}_m(\mathbf{S},\nu)$, such that

$$\mathbf{S}V_m = V_m H_m + h_{m+1,m}\nu_{m+1}e_m^T, \qquad V_m^T \begin{bmatrix} \mathbf{A} & 0 \\ 0 & \mathbf{M} \end{bmatrix} V_m = \mathbf{I}_m.$$

Here, $e_m = (0,\dots,0,1)^T \in \mathbb{R}^m$ and I_m denotes the $m \times m$ identity matrix. An approximation of the exponential matrix function is computed by

$$\exp(\tau\mathbf{S})\nu \approx \|\nu\|_{\mathbf{A}\times\mathbf{M}} V_m \exp(\tau H_m)e_1. \tag{4.5.6}$$

(see e.g., [27, eq.(2.2)]). By [27, eq.(2.5)] we get the following expression for the residual

$$|R_m(\tau)| = \|\nu\|_{\mathbf{A}\times\mathbf{M}} |h_{m+1,m}e_m^T \exp(\tau H_m)e_1| < \mathrm{TOL}_R, \tag{4.5.7}$$

which provides an error estimate controlled by a user-defined tolerance TOL_R.

Shift-and-Invert Arnoldi method Here the Krylov subspace is constructed with respect to the matrix $(\mathbf{I} - s\mathbf{S})^{-1}$ with $s > 0$ being the shift parameter. Then, the Krylov relation reads

$$(\mathbf{I} - s\mathbf{S})^{-1}V_m = V_m\widetilde{H}_m + \widetilde{h}_{m+1,m}\nu_{m+1}e_m^T. \tag{4.5.8}$$

An approximation of the matrix function is then calculated using (4.5.6), where H_m is defined by

$$H_m = \frac{1}{s}(\mathbf{I} - \widetilde{H}_m^{-1}),$$

c.f. [138]. Using [27, eq.(2.10)] gives $\|R_m(\tau)\|_{\mathbf{A}\times\mathbf{M}} = |\beta_m(\tau)|\|w_{m+1}\|_{\mathbf{A}\times\mathbf{M}} < \mathrm{TOL}_R$ with

$$|\beta_m(\tau)| = \|v\|_{\mathbf{A}\times\mathbf{M}}|\frac{h_{m+1,m}}{s}e_m^T\widetilde{H}_m^{-1}\exp(\tau H_m)e_1|, \qquad w_{m+1} = (\mathbf{I}-s\mathbf{S})\nu_{m+1},$$

which provides an error estimate controlled by a user-defined tolerance TOL_R. Note, that we never calculate the inverse $(\mathbf{I}-s\mathbf{S})^{-1}$ explicitly. Instead we solve

$$(\mathbf{I}-s\mathbf{S})x = y \Leftrightarrow \left(\begin{bmatrix}\mathbf{A} & 0\\ 0 & \mathbf{M}\end{bmatrix} - s\begin{bmatrix}0 & \mathbf{A}\\ -\mathbf{A} & \mathbf{B}\end{bmatrix}\right)\begin{bmatrix}x_1\\ x_2\end{bmatrix} = \begin{bmatrix}\mathbf{A} & 0\\ 0 & \mathbf{M}\end{bmatrix}\begin{bmatrix}y_1\\ y_2\end{bmatrix}$$

for $x = \begin{bmatrix}x_1\\ x_2\end{bmatrix}$ and $y = \begin{bmatrix}y_1\\ y_2\end{bmatrix}$ for $x_1, x_2, y_1, y_2 \in \mathbb{R}^N$. This together with the Schur complement lead to following system of equations

$$\begin{aligned} y_1 &= x_1 + sy_2,\\ \mathbf{T}y_2 &= \mathbf{M}x_2 - s\mathbf{A}x_1 \end{aligned}$$

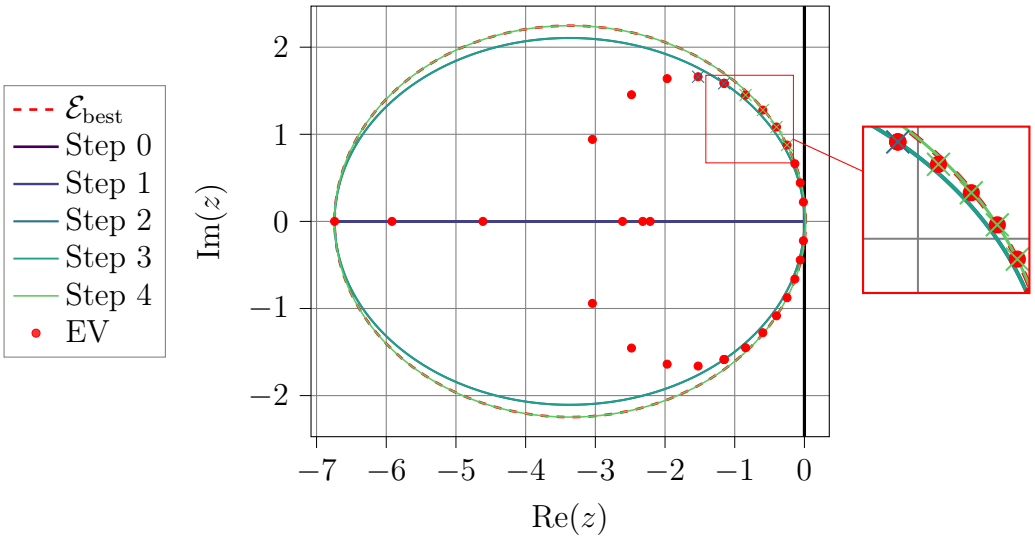
with $\mathbf{T} = \mathbf{M} + s^2\mathbf{A} + s\mathbf{B}$. We solve this system using the CG method with tolerance $\mathrm{TOL}_{\mathrm{sol}} = 0.01 \cdot \mathrm{TOL}_R$ if $\mathbf{T}$ is symmetric. Otherwise, we use the GMRES method with the same tolerance $\mathrm{TOL}_{\mathrm{sol}}$. In the symmetric case, we use the IC(0) as a preconditioner, whereas in the nonsymmetric case, we employ the incomplete LU factorization without fill-in (ILU(0)) (cf. [126, Chp. 10]).

4.5.4 Numerical examples

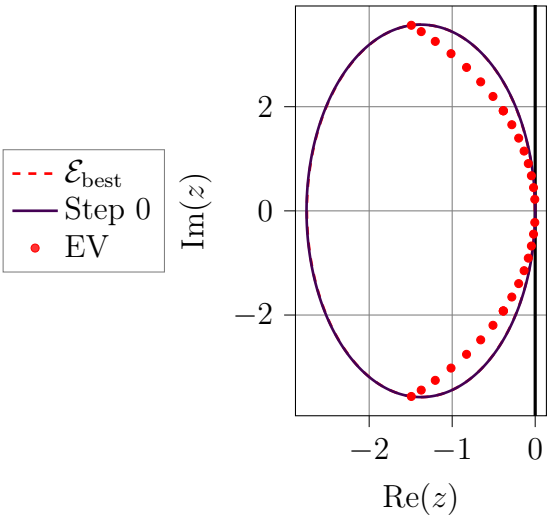
In this section we present numerical experiments for the model problem (4.5.3) using the Chebyshev method from Section 4.2. All experiments are performed in the C++-library *deal.II* [10] and run on an Intel i5-12500K CPU with 32 GB RAM.

Example 1 In order to show how the Arnoldi method with spectral preconditioning (Algorithm 3) works in practice, we first consider a small-scale example. We set $\Omega = (0,1)$ and $\mathcal{B}$ as in (4.5.2a) with $\delta \in \{0.01, 0.03, 0.4\}$ and $a = 0.5$. For the discretization we use an uniform grid with grid width $h = 2^{-4}$ which lead to $\mathbf{A}, \mathbf{M}, \mathbf{B} \in \mathbb{R}^{17\times 17}$.

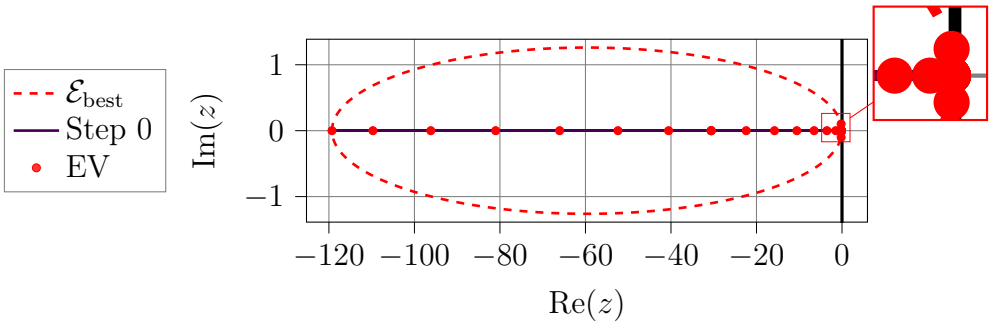
We first compute the eigenvalues of the matrix $\mathbf{S}$ using a standard dense eigenvalue solver from LAPACK++ (`https://lapackpp.sourceforge.net/html/index.html`) and calculate the *best-fitting* ellipse $\mathcal{E}_{\mathrm{best}}$ containing all eigenvalues using the algorithm from Section 4.4.1 for reference.



(a) $\delta = 0.03$



(b) $\delta = 0.01$



(c) $\delta = 0.4$

Figure 4.3: Eigenvalues of **S** in Example 1, best-fitting reference ellipse, and the calculated ellipse of the Arnoldi method with spectral preconditioning for each step and for different coefficients δ.

In Figure 4.3, we show $\mathcal{E}_{\text{best}}$ and the eigenvalues of $\mathbf{S}$ together with the result of the Arnoldi method with spectral preconditioning, which was applied to $\mathbf{S}$ for different δ. Note that the axes are scaled differently for better visualization.

The first Figure 4.3a shows the case, where the method performed several steps to find a best-fitting ellipse. Each detected eigenvalue is marked with a cross in the color corresponding to the step in which it was found. It can be seen that after a few steps, the method finds the best-fitting ellipse.

The second Figure 4.3b shows the case where the largest eigenvalue is sufficient to obtain the best-fitting ellipse, so that the algorithm converges in one step.

The last Figure 4.3c shows the case where there are eigenvalues that lie slightly outside the computed ellipse, as they were not found with the chosen tolerance and ℓ_0. Theoretically, this can lead to problems, but in practice we have not encountered any issues with the selected tolerance and initial degree, as the computed ellipse differs relatively little from the best-fitting ellipse.

Example 2 We set $\Omega = (0,1)^2$ and $\mathcal{B}$ with $\delta \in \{0.1, 0.01, 0.001\}$ and $a = 0.5$ as in (4.5.2a). We use a 2D-finite element discretization on a regular rectangular grid with grid width $h = 2^{-7}$, which lead to $\mathbf{A}, \mathbf{M}, \mathbf{B} \in \mathbb{R}^{N\times N}$ with $N = 16641$.

Further, we set ν_c as the coefficient vector that is obtained from the interpolation of the function $g(x,y) = \sin(\pi x^2)\sin(\pi y^2)$ on the grid. As initial vector we choose

$$\nu = \begin{bmatrix} \nu_1 \\ \nu_2 \end{bmatrix} \quad \text{with} \quad \nu_1 = \nu_2 = \nu_c \in \mathbb{R}^N.$$

In the following we compute the matrix function $\exp(\tau\mathbf{S})\nu$ with $\tau = 0.1$. We compare the Chebyshev method with the standard Krylov subspace method and the shift-and-invert method. Since the computational cost is dominated by sparse matrix–vector multiplications, we measure the computational cost in terms of matrix–vector multiplications with $\mathbf{A}, \mathbf{B}, \mathbf{M}$ or similar matrices. In CG or GMRES with IC(0)/ILU(0) preconditioning, each iteration requires one matrix–vector multiplication, and one forward and one backward substitution with the triangular matrices obtained from the IC(0)/ILU(0) factorization. Since, for suitable sparse storage formats and memory layouts, the cost of these two substitutions is comparable to one sparse matrix–vector multiplication (see, e.g., [131] for ILU(0)), we approximate the cost of one preconditioned CG/GMRES iteration by two sparse matrix–vector multiplications.

In Figure 4.4, we examine the error with respect to the number of matrix-vector multiplications for different δ values and different tolerances TOL_R applied to the respective residual error estimators.

We calculate a reference solution using the standard Krylov subspace method for the matrix function with a very low tolerance 10^{-10} and the residual calculated by (4.5.7). Furthermore, we set $s = \frac{\tau}{2}$.

We observe that the Chebyshev method performs significantly better than the standard Krylov subspace method. Compared to the shift-and-invert method, the Chebyshev method is less efficient for relatively large tolerances. However, for smaller tolerances, the Chebyshev method benefits from the exponential decay of the error combined with the low computational cost per iteration, which considerably improves its efficiency compared to the shift-and-invert method. This advantage diminishes for larger δ (Figure 4.4b), so that the shift-and-invert method performs slightly better (Figure 4.4c) if δ is sufficiently large. This behavior can be explained by the fact that for small δ, the eigenvalues are clustered which is advantageous for the Chebyshev method because the ellipses stay relatively small (see Figure 4.3). For larger δ, the eigenvalues are more dispersed, which makes the advantage of an ellipse-based approach less significant.

Example 3 We use the same setting as in Example 2, but $\mathcal{B}$ with $\delta \in \{1.0, 0.1, 0.01\}$ and

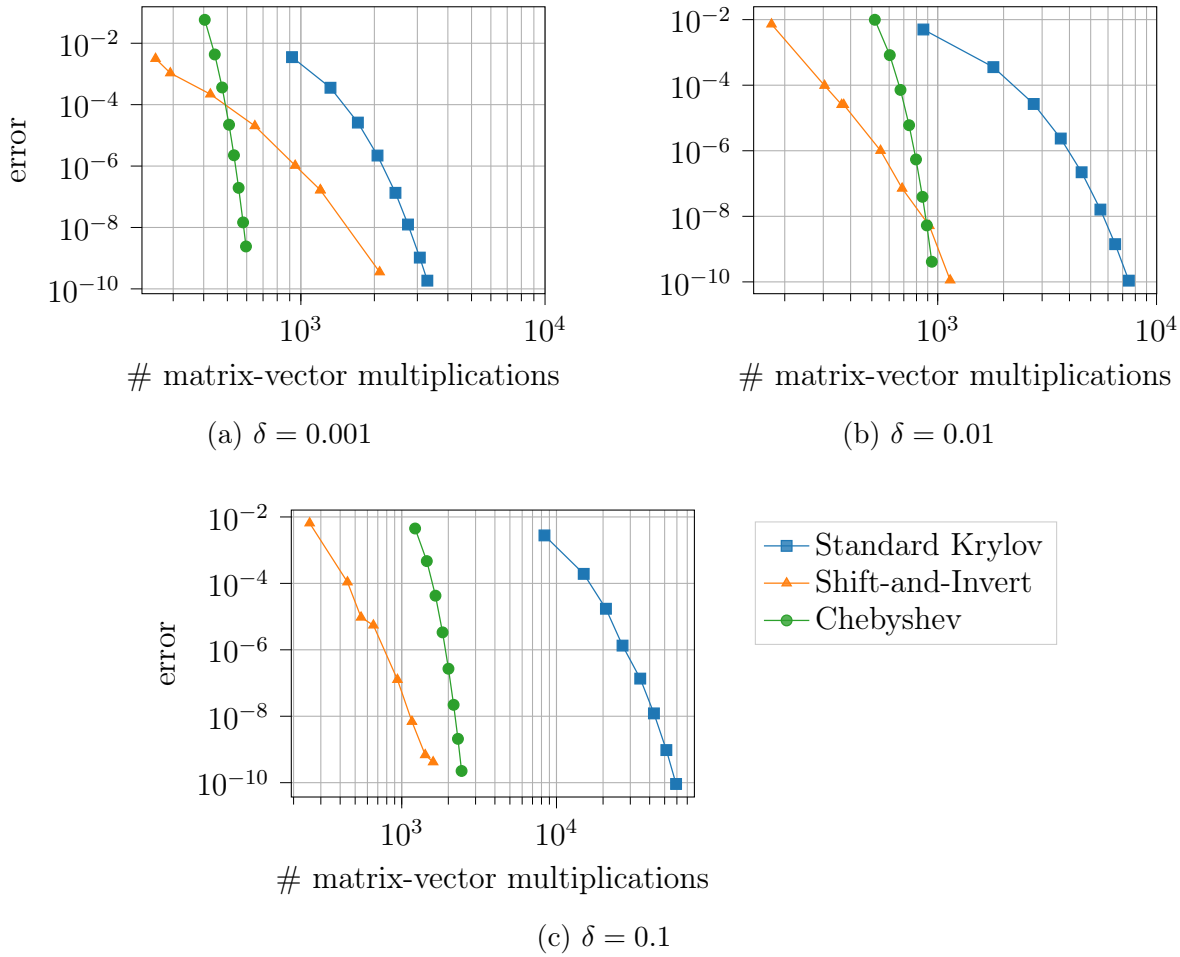
$$b(x_1, x_2) = \begin{bmatrix} -x_1 \\ 0 \end{bmatrix}$$

is defined as in (4.5.2b).

Again, we compute the matrix function $\exp(\tau \mathbf{S})\nu$ with $\tau = 0.1$. In Figure 4.5, we examine the error with respect to the number of matrix-vector multiplications for different δ values and different tolerances TOL_R applied to the respective residual error estimators. Again, we calculate a reference solution using the standard Krylov subspace method for the matrix function with a very low tolerance 10^{-10} and choose $s = \frac{\tau}{2}$.

We observe that the Chebyshev method performs significantly better than the standard Krylov subspace method. Compared to the shift-and-invert method, it can be seen, as in Example 2, that the shift-and-invert method is particularly efficient for relatively large tolerances, while the Chebyshev method is more advantageous for smaller tolerances.

Data availability The codes to reproduce the experiments are available on https://github.com/danieleckhardt/chebychev



(a) $\delta = 0.001$

(b) $\delta = 0.01$

(c) $\delta = 0.1$

Figure 4.4: Approximation error in $\|\cdot\|_{\mathbf{A}\times\mathbf{M}}$ of $\exp(\tau\mathbf{S})v$ with $\tau = 0.1$ in Example 2 for different δ. The error is shown w.r.t. the number of matrix-vector multiplications, for $tol = 10^{-k}$, $k = 1, \dots 8$ as tolerance for the residual error estimators.

4.6 Useful identities

Lemma 4.6.1. *For $\theta_\psi^\omega = \psi - \mathrm{i}\,\omega$, $\psi \in [0, 2\pi]$ and $\omega \geq 0$ it holds that*

$$\cos(\theta_\psi^\omega) = \cosh\omega\cos\psi + \mathrm{i}\sinh\omega\sin\psi.$$

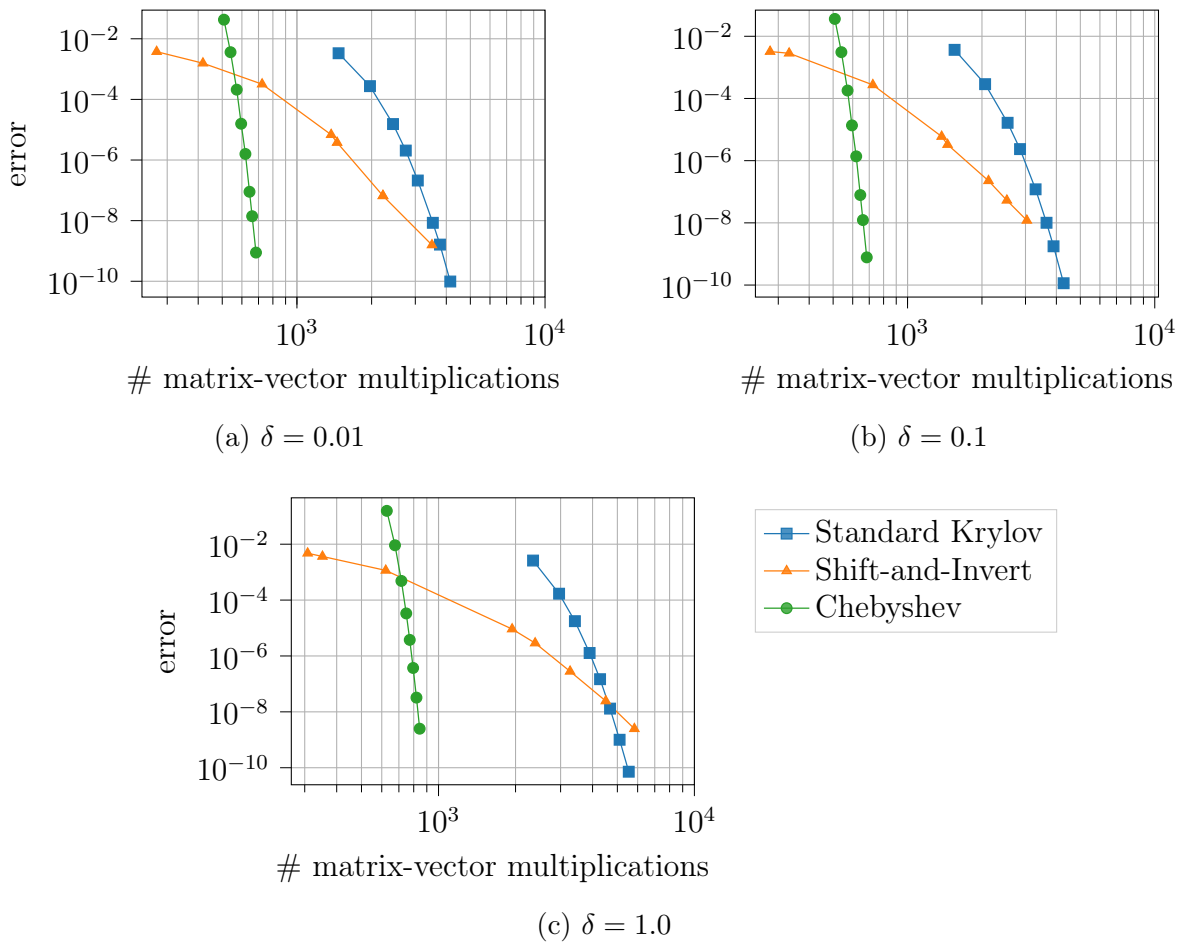
Proof.

$$\begin{aligned}\cos(\theta_\psi^\omega) &= \frac{1}{2}(\mathrm{e}^{\mathrm{i}\psi+\omega} + \mathrm{e}^{-\mathrm{i}\psi-\omega}) = \frac{1}{2}((\cos\psi + \mathrm{i}\sin\psi)\,\mathrm{e}^{\omega} + (\cos\psi - \mathrm{i}\sin\psi)\,\mathrm{e}^{-\omega}) \\ &= \cosh\omega\cos\psi + \mathrm{i}\sinh\omega\sin\psi.\end{aligned}$$

□

Lemma 4.6.2. *Let*

$$z = \alpha\cos\psi + \mathrm{i}\,\beta\sin\psi + \gamma \in \mathcal{E}_\gamma^{\alpha,\beta}.$$



(a) $\delta = 0.01$ (b) $\delta = 0.1$

(c) $\delta = 1.0$

Figure 4.5: Approximation error in $\|\cdot\|_{\mathbf{A}\times\mathbf{M}}$ of $\exp(\tau\mathbf{S})v$ with $\tau = 0.1$ in Example 3 for different δ. The error is shown w.r.t. the number of matrix-vector multiplications, for $tol = 10^{-k}$, $k = 1, \dots 8$ as tolerance for the residual error estimators.

Then it holds that if $\alpha \neq \beta$

$$\frac{z-\gamma}{\epsilon} = \cos\theta_\psi^{\alpha,\beta} = \frac{1}{2}(\mathrm{e}^{\mathrm{i}\,\theta_\psi^{\alpha,\beta}} + \mathrm{e}^{-\mathrm{i}\,\theta_\psi^{\alpha,\beta}}),$$

where $\epsilon = \sqrt{\alpha^2 - \beta^2}$ *and*

$$\theta_\psi^{\alpha,\beta} = \psi - \mathrm{i}\log\Big(\sqrt{\frac{\alpha+\beta}{\alpha-\beta}}\Big) \qquad \text{if } \epsilon \in \mathbb{R}$$

$$\theta_\psi^{\alpha,\beta} = \psi - \frac{\pi}{2} - \mathrm{i}\log\Big(\sqrt{\frac{\beta+\alpha}{\beta-\alpha}}\Big) \qquad \text{if } \epsilon \in \mathrm{i}\,\mathbb{R}$$

with $\psi \in [-\pi, \pi]$.

Proof. Follows directly from Lemma 4.6.1. □

Bibliography

[1] A. Abdulle. The finite element heterogeneous multiscale method: a computational strategy for multiscale PDEs. In *Multiple scales problems in biomathematics, mechanics, physics and numerics*, volume 31 of *GAKUTO Internat. Ser. Math. Sci. Appl.*, pages 133–181. Gakkotosho, Tokyo, 2009.

[2] A. Abdulle, W. E, B. Engquist, and E. Vanden-Eijnden. The heterogeneous multiscale method. *Acta Numer.*, 21:1–87, 2012.

[3] A. Abdulle and M. J. Grote. Finite element heterogeneous multiscale method for the wave equation. *Multiscale Model. Simul.*, 9(2):766–792, 2011.

[4] A. Abdulle, M. J. Grote, and C. Stohrer. Finite element heterogeneous multiscale method for the wave equation: long-time effects. *Multiscale Model. Simul.*, 12(3):1230–1257, 2014.

[5] P. C. Africa, D. Arndt, W. Bangerth, B. Blais, M. Fehling, R. Gassmöller, T. Heister, L. Heltai, S. Kinnewig, M. Kronbichler, M. Maier, P. Munch, M. Schreter-Fleischhacker, J. P. Thiele, B. Turcksin, D. Wells, and V. Yushutin. The deal.ii library, version 9.6. *Journal of Numerical Mathematics*, 32(4):369–380, 2024.

[6] L. V. Ahlfors. *Complex analysis*. International Series in Pure and Applied Mathematics. McGraw-Hill Book Co., New York, third edition, 1978. An introduction to the theory of analytic functions of one complex variable.

[7] G. Akrivis, M. Crouzeix, and C. Makridakis. Implicit-explicit multistep methods for quasilinear parabolic equations. *Numer. Math.*, 82(4):521–541, 1999.

[8] D. D. Ang and A. Pham Ngoc Dinh. *Some viscoelastic wave equations*, pages 35–43. Springer Netherlands, Dordrecht, 1989.

[9] W. Arendt and M. Kreuter. Mapping theorems for Sobolev spaces of vector-valued functions. *Studia Math.*, 240(3):275–299, 2018.

[10] D. Arndt, W. Bangerth, M. Bergbauer, B. Blais, M. Fehling, R. Gassmöller, T. Heister, L. Heltai, M. Kronbichler, M. Maier, P. Munch, S. Scheuerman, B. Turcksin, S. Uzunbajakau, D. Wells, and M. Wichrowski. The deal.ii library, version 9.7. *Journal of Numerical Mathematics*, 33(4):403–415, 2025.

[11] K. R. Arun, A. J. Das Gupta, and S. Samantaray. Analysis of an asymptotic preserving low Mach number accurate IMEX-RK scheme for the wave equation system. *Appl. Math. Comput.*, 411:Paper No. 126469, 20, 2021.

[12] U. M. Ascher, S. J. Ruuth, and R. J. Spiteri. Implicit-explicit Runge-Kutta methods for time-dependent partial differential equations. *Appl. Numer. Math.*, 25(2-3):151–167, 1997. Special issue on time integration (Amsterdam, 1996).

[13] U. M. Ascher, S. J. Ruuth, and B. T. R. Wetton. Implicit-explicit methods for time-dependent partial differential equations. *SIAM J. Numer. Anal.*, 32(3):797–823, 1995.

[14] P. Aviles and J. Sandefur. Nonlinear second order equations with applications to partial differential equations. *J. Differential Equations*, 58(3):404–427, 1985.

[15] N. Banichuk, J. Jeronen, P. Neittaanmäki, T. Saksa, and T. Tuovinen. *Mechanics of Moving Materials*, volume 207 of *Solid Mechanics and Its Applications*. Springer International Publishing, 2014. First edition.

[16] S. Baumstark, E. Faou, and K. Schratz. Uniformly accurate exponential-type integrators for Klein-Gordon equations with asymptotic convergence to the classical NLS splitting. *Math. Comp.*, 87(311):1227–1254, 2018.

[17] B. Beckermann and L. Reichel. Error estimates and evaluation of matrix functions via the Faber transform. *SIAM J. Numer. Anal.*, 47(5):3849–3883, 2009.

[18] A. Bensoussan, J.-L. Lions, and G. Papanicolaou. *Asymptotic analysis for periodic structures*. AMS Chelsea Publishing, Providence, RI, 2011. Corrected reprint of the 1978 original [MR0503330].

[19] L. Bergamaschi and M. Vianello. Efficient computation of the exponential operator for large, sparse, symmetric matrices. *Numer. Linear Algebra Appl.*, 7(1):27–45, 2000.

[20] M. Berljafa and S. Güttel. Generalized rational Krylov decompositions with an application to rational approximation. *SIAM J. Matrix Anal. Appl.*, 36(2):894–916, 2015.

[21] M. Berljafa and S. Güttel. Generalized rational Krylov decompositions with an application to rational approximation. *SIAM J. Matrix Anal. Appl.*, 36(2):894–916, 2015.

[22] M. Berljafa and S. Güttel. Parallelization of the rational Arnoldi algorithm. *SIAM J. Sci. Comput.*, 39(5):S197–S221, 2017.

[23] R. L. Bisplinghoff and H. Ashley. *Principles of aeroelasticity.* John Wiley & Sons, Inc., New York-London, 1962.

[24] E. Bonetti, E. Rocca, R. Scala, and G. Schimperna. On the strongly damped wave equation with constraint. *Comm. Partial Differential Equations*, 42(7):1042–1064, 2017.

[25] S. Boscarino. Error analysis of IMEX Runge-Kutta methods derived from differential-algebraic systems. *SIAM J. Numer. Anal.*, 45(4):1600–1621, 2007.

[26] S. Boscarino, L. Pareschi, and G. Russo. *Implicit-explicit methods for evolutionary partial differential equations*, volume 24 of *Mathematical Modeling and Computation.* Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2025.

[27] M. A. Botchev, V. Grimm, and M. Hochbruck. Residual, restarting, and Richardson iteration for the matrix exponential. *SIAM J. Sci. Comput.*, 35(3):A1376–A1397, 2013.

[28] S. C. Brenner and L. R. Scott. *The mathematical theory of finite element methods*, volume 15 of *Texts in Applied Mathematics.* Springer, New York, third edition, 2008.

[29] S. Buchholz, B. Dörich, and M. Hochbruck. On averaged exponential integrators for semilinear wave equations with solutions of low-regularity. *Partial Differ. Equ. Appl.*, 2(2):Paper No. 23, 27, 2021.

[30] S. Burkhard, M. Hochbruck, and M. Scheifinger. An implicit-explicit time discretization scheme for second-order semilinear wave equations with a nonlocal material law and kinetic boundary conditions. *CRC 1173 Preprint*, 2025.

[31] J. C. Butcher and G. Wanner. Runge-Kutta methods: some historical notes. *Appl. Numer. Math.*, 22(1-3):113–151, 1996. Special issue celebrating the centenary of Runge-Kutta methods.

[32] M. Caliari, P. Kandolf, A. Ostermann, and S. Rainer. The Leja method revisited: backward error analysis for the matrix exponential. *SIAM J. Sci. Comput.*, 38(3):A1639–A1661, 2016.

[33] M. Caliari and A. Ostermann. Implementation of exponential Rosenbrock-type integrators. *Appl. Numer. Math.*, 59(3-4):568–581, 2009.

[34] M. Caliari, M. Vianello, and L. Bergamaschi. Interpolating discrete advection-diffusion propagators at Leja sequences. *J. Comput. Appl. Math.*, 172(1):79–99, 2004.

[35] T. Cazenave. Uniform estimates for solutions of nonlinear klein–gordon equations. *Journal of Functional Analysis*, 60(1):36–55, 1985.

[36] F. c. Chatelin. *Eigenvalues of matrices*, volume 71 of *Classics in Applied Mathematics*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2012. With exercises by Mario Ahués and the author, Translated with additional material by Walter Ledermann, Revised reprint of the 1993 edition [MR1232655].

[37] I. Chueshov, I. Lasiecka, and J. Webster. Flow-plate interactions: well-posedness and long-time behavior. *Discrete Contin. Dyn. Syst. Ser. S*, 7(5):925–965, 2014.

[38] P. G. Ciarlet. *The finite element method for elliptic problems*, volume 40 of *Classics in Applied Mathematics*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2002.

[39] G. Cohen, P. Joly, J. E. Roberts, and N. Tordjman. Higher order triangular finite elements with mass lumping for the wave equation. *SIAM J. Numer. Anal.*, 38(6):2047–2078, 2001.

[40] M. Crouzeix and C. Palencia. The numerical range is a $(1+\sqrt{2})$-spectral set. *SIAM J. Matrix Anal. Appl.*, 38(2):649–655, 2017.

[41] J. H. Curtiss. Faber polynomials and the Faber series. *Amer. Math. Monthly*, 78:577–596, 1971.

[42] G. Dahlquist. A special stability problem for linear multistep methods. *BIT Numerical Mathematics*, 3(1):27–43, 1963.

[43] D. A. Di Pietro and A. Ern. *Mathematical aspects of discontinuous Galerkin methods*, volume 69 of *Mathématiques & Applications (Berlin) [Mathematics & Applications]*. Springer, Heidelberg, 2012.

[44] B. Dörich and J. Leibold. Full discretization error analysis of exponential integrators for semilinear wave equations. *Math. Comp.*, 91(336):1687–1709, 2022.

[45] E. H. Dowell, K. C. Hall, J. P. Thomas, and D. A. Dowell, editors. *A Modern Course in Aeroelasticity*, volume 293 of *Solid Mechanics and Its Applications*. Springer, Cham, 6 edition, 2022.

[46] V. Druskin and L. Knizhnerman. Krylov subspace approximation of eigenpairs and matrix functions in exact and computer arithmetic. *Numer. Linear Algebra Appl.*, 2(3):205–217, 1995.

[47] V. L. Druskin and L. A. Knizhnerman. Two polynomial methods for calculating functions of symmetric matrices. *Zh. Vychisl. Mat. i Mat. Fiz.*, 29(12):1763–1775, 1989.

[48] V. L. Druskin and L. A. Knizhnerman. Error estimates in the simple Lanczos process in the calculation of functions of symmetric matrices and eigenvalues. *Zh. Vychisl. Mat. i Mat. Fiz.*, 31(7):970–983, 1991.

[49] R. Du and P. Ming. Convergence of the heterogeneous multiscale finite element method for elliptic problems with nonsmooth microstructures. *Multiscale Model. Simul.*, 8(5):1770–1783, 2010.

[50] I. S. Duff and J. A. Scott. Computing selected eigenvalues of sparse unsymmetric matrices using subspace iteration. *ACM Trans. Math. Software*, 19(2):137–159, 1993.

[51] D. Eckhardt, V. Grimm, and M. Hochbruck. A Chebychev method for matrix function approximation. *CRC 1173 Preprint*, 2026.

[52] D. Eckhardt, M. Hochbruck, and B. Verfürth. Error analysis of an implicit–explicit time discretization scheme for semilinear wave equations with application to multiscale problems. *IMA Journal of Numerical Analysis*, page draf092, 10 2025.

[53] M. Eiermann. On semiiterative methods generated by Faber polynomials. *Numer. Math.*, 56(2-3):139–156, 1989.

[54] A. Ern and J.-L. Guermond. *Finite elements II—Galerkin approximation, elliptic and mixed PDEs*, volume 73 of *Texts in Applied Mathematics*. Springer, Cham, 2021.

[55] G. Faber. Über polynomische Entwicklungen. *Math. Ann.*, 57:389–408, 1903.

[56] G. Faber. Über polynomische Entwicklungen II. *Math. Ann.*, 64(1):116–135, 1907.

[57] R. W. Freund, M. H. Gutknecht, and N. M. Nachtigal. An implementation of the look-ahead Lanczos algorithm for non-Hermitian matrices. *SIAM J. Sci. Comput.*, 14(1):137–158, 1993.

[58] M. Fujimoto. *Introduction to the Mathematical Physics of Nonlinear Waves (Second Edition)*. 2053-2563. IOP Publishing, 2021.

[59] D. Gaier, editor. *Lectures on complex approximation*. Birkhäuser Boston, Inc., Boston, MA, 1987.

[60] D. J. Gardner, J. E. Guerra, F. P. Hamon, D. R. Reynolds, P. A. Ullrich, and C. S. Woodward. Implicit–explicit (imex) runge–kutta methods for non-hydrostatic atmospheric models. *Geoscientific Model Development*, 11(4):1497–1515, 2018.

[61] L. Gauckler. Error analysis of trigonometric integrators for semilinear wave equations. *SIAM J. Numer. Anal.*, 53(2):1082–1106, 2015.

[62] F. Gazzola and M. Squassina. Global solutions and finite time blow up for damped semilinear wave equations. *Ann. Inst. H. Poincaré C Anal. Non Linéaire*, 23(2):185–207, 2006.

[63] K. O. Geddes. Near-minimax polynomial approximation in an elliptical region. *SIAM J. Numer. Anal.*, 15(6):1225–1233, 1978.

[64] S. Geevers, W. A. Mulder, and J. J. W. van der Vegt. New higher-order mass-lumped tetrahedral elements for wave propagation modelling. *SIAM J. Sci. Comput.*, 40(5):A2830–A2857, 2018.

[65] J.-M. Ghidaglia and A. Marzocchi. Longtime behaviour of strongly damped wave equations, global attractors and their dimension. *SIAM J. Math. Anal.*, 22(4):879–895, 1991.

[66] T. Göckler. *Rational Krylov subspace methods for phi-functions in exponential integrators*. PhD thesis, Karlsruhe, Karlsruher Institut für Technologie (KIT), PhD thesis, 2014, 2014.

[67] T. Göckler and V. Grimm. Convergence analysis of an extended Krylov subspace method for the approximation of operator functions in exponential integrators. *SIAM J. Numer. Anal.*, 51(4):2189–2213, 2013.

[68] T. Göckler and V. Grimm. Acceleration of contour integration techniques by rational Krylov subspace methods. *J. Comput. Appl. Math.*, 316:133–142, 2017.

[69] V. Grimm. Resolvent Krylov subspace approximation to operator functions. *BIT*, 52(3):639–659, 2012.

[70] M. J. Grote, S. Michel, and S. A. Sauter. Stabilized leapfrog based local time-stepping method for the wave equation. *Math. Comp.*, 90(332):2603–2643, 2021.

[71] S. Güttel. *Rational Krylov methods for operator functions*. PhD thesis, Technische Universität Bergakademie Freiberg, 2010.

[72] S. Güttel. Rational Krylov approximation of matrix functions: numerical methods and optimal pole selection. *GAMM-Mitt.*, 36(1):8–31, 2013.

[73] E. Hairer. Highest possible order of algebraically stable diagonally implicit runge-kutta methods. *BIT Numerical Mathematics*, 20(2):254–256, 1980.

[74] E. Hairer, C. Lubich, and G. Wanner. *Geometric numerical integration*, volume 31 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, second edition, 2006. Structure-preserving algorithms for ordinary differential equations.

[75] E. Hairer, S. P. Nørsett, and G. Wanner. *Solving ordinary differential equations. I*, volume 8 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, second edition, 1993. Nonstiff problems.

[76] E. Hairer and G. Wanner. *Solving ordinary differential equations. II*, volume 14 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, revised edition, 2010. Stiff and differential-algebraic problems.

[77] H. Han and Z. Zhang. Split local absorbing conditions for one-dimensional nonlinear Klein-Gordon equation on unbounded domain. *J. Comput. Phys.*, 227(20):8992–9004, 2008.

[78] E. Hansen and A. Ostermann. High-order splitting schemes for semilinear evolution equations. *BIT*, 56(4):1303–1316, 2016.

[79] N. J. Higham. *Functions of matrices*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2008. Theory and computation.

[80] D. Hipp, M. Hochbruck, and C. Stohrer. Unified error analysis for nonconforming space discretizations of wave-type equations. *IMA J. Numer. Anal.*, 39(3):1206–1245, 2019.

[81] D. Ho. Tchebychev acceleration technique for large scale nonsymmetric matrices. *Numer. Math.*, 56(7):721–734, 1990.

[82] M. Hochbruck and J. Leibold. Finite element discretization of semilinear acoustic wave equations with kinetic boundary conditions. *Electron. Trans. Numer. Anal.*, 53:522–540, 2020.

[83] M. Hochbruck and J. Leibold. An implicit-explicit time discretization scheme for second-order semilinear wave equations with application to dynamic boundary conditions. *Numer. Math.*, 147(4):869–899, 2021.

[84] M. Hochbruck and C. Lubich. On Krylov subspace approximations to the matrix exponential operator. *SIAM J. Numer. Anal.*, 34(5):1911–1925, 1997.

[85] M. Hochbruck, C. Lubich, and H. Selhofer. Exponential integrators for large systems of differential equations. *SIAM J. Sci. Comput.*, 19(5):1552–1574, 1998.

[86] M. Hochbruck and A. Ostermann. Explicit exponential Runge-Kutta methods for semilinear parabolic problems. *SIAM J. Numer. Anal.*, 43(3):1069–1090, 2005.

[87] M. Hochbruck and A. Ostermann. Exponential integrators. *Acta Numer.*, 19:209–286, 2010.

[88] F. Huang and J. Shen. A new class of implicit-explicit BDFk SAV schemes for general dissipative systems and their error analysis. *Comput. Methods Appl. Mech. Engrg.*, 392:Paper No. 114718, 25, 2022.

[89] W. Hundsdorfer and S. J. Ruuth. IMEX extensions of linear multistep methods with general monotonicity and boundedness properties. *J. Comput. Phys.*, 225(2):2016–2042, 2007.

[90] T. Jawecki. A study of defect-based error estimates for the Krylov approximation of φ-functions. *Numer. Algorithms*, 90(1):323–361, 2022.

[91] Z. Jia and H. Lv. A posteriori error estimates of Krylov subspace approximations to matrix functions. *Numer. Algorithms*, 69(1):1–28, 2015.

[92] S. Y. Kadioglu, D. A. Knoll, R. B. Lowrie, and R. M. Rauenzahn. A second order self-consistent IMEX method for radiation hydrodynamics. *J. Comput. Phys.*, 229(22):8313–8332, 2010.

[93] P. Kandolf, A. Ostermann, and S. Rainer. A residual based error estimate for Leja interpolation of matrix functions. *Linear Algebra Appl.*, 456:157–173, 2014.

[94] J. S. Kole. Solving seismic wave propagation in elastic media using the matrix exponential approach. *Wave Motion*, 38(4):279–293, 2003.

[95] T. Koto. IMEX Runge-Kutta schemes for reaction-diffusion equations. *J. Comput. Appl. Math.*, 215(1):182–195, 2008.

[96] C. Lanczos. An iteration method for the solution of the eigenvalue problem of linear differential and integral operators. *J. Research Nat. Bur. Standards*, 45:255–282, 1950.

[97] W. Layton, Y. Li, and C. Trenchea. Recent developments in IMEX methods with time filters for systems of evolution equations. *J. Comput. Appl. Math.*, 299:50–67, 2016.

[98] W. Layton and C. Trenchea. Stability of two IMEX methods, CNLF and BDF2-AB2, for uncoupling systems of evolution equations. *Appl. Numer. Math.*, 62(2):112–120, 2012.

[99] D. Li, C. Quan, and T. Tang. Stability and convergence analysis for the implicit-explicit method to the Cahn-Hilliard equation. *Math. Comp.*, 91(334):785–809, 2022.

[100] C. Lubich. *From quantum to classical molecular dynamics: reduced models and numerical analysis*, volume 12. European Mathematical Society, 2008.

[101] J. Lubliner and P. Papadopoulos. *Introduction to Solid Mechanics*. Springer, Cham, 2nd edition, 2017.

[102] T. A. Manteuffel. *An Iterative Method for Solving Nonsymmetric Linear Systems with Dynamic Estimation of Parameters*. PhD thesis, University of Illinois at Urbana-Champaign, Ann Arbor, MI, 1975. Ph.D. thesis.

[103] T. A. Manteuffel. The Tchebychev iteration for nonsymmetric linear systems. *Numer. Math.*, 28(3):307–327, 1977.

[104] J. C. Mason and D. C. Handscomb. *Chebyshev Polynomials*. Chapman & Hall/CRC, Boca Raton, FL, 2003.

[105] K. Meerbergen and D. Roose. Matrix transformations for computing rightmost eigenvalues of large sparse non-symmetric eigenvalue problems. *IMA J. Numer. Anal.*, 16(3):297–346, 1996.

[106] H. Methenni, A. Imperiale, and S. Imperiale. An implicit–explicit time discretization for elastic wave propagation problems in plates. *International Journal for Numerical Methods in Engineering*, 125(5):e7393, 2024.

[107] H. Mohamad. Energy asymptotics for the strongly damped Klein-Gordon equation. *Partial Differ. Equ. Appl.*, 3(6):Paper No. 71, 12, 2022.

[108] I. Moret. On RD-rational Krylov approximations to the core-functions of exponential integrators. *Numer. Linear Algebra Appl.*, 14(5):445–457, 2007.

[109] I. Moret and P. Novati. The computation of functions of matrices by truncated Faber series. *Numer. Funct. Anal. Optim.*, 22(5-6):697–719, 2001.

[110] I. Moret and P. Novati. RD-rational approximations of the matrix exponential. *BIT*, 44(3):595–615, 2004.

[111] N. J. Munch. A Chebyshev theorem for ellipses in the complex plane. *Amer. Math. Monthly*, 126(5):430–436, 2019.

[112] O. Nevanlinna, editor. *Convergence of iterations for linear equations*. Lectures in Mathematics ETH Zürich. Birkhäuser Verlag, Basel, 1993.

[113] G. Nguetseng, H. Nnang, and N. Svanstedt. G-convergence and homogenization of monotone damped hyperbolic equations. *Banach J. Math. Anal.*, 4(1):100–115, 2010.

[114] J. Niesen and W. M. Wright. Algorithm 919: a Krylov subspace algorithm for evaluating the ϕ-functions appearing in exponential integrators. *ACM Trans. Math. Software*, 38(3):Art. 22, 19, 2012.

[115] P. Novati. Using the restricted-denominator rational Arnoldi method for exponential integrators. *SIAM J. Matrix Anal. Appl.*, 32(4):1537–1558, 2011.

[116] A. Pazy. *Semigroups of linear operators and applications to partial differential equations*, volume 44 of *Applied Mathematical Sciences*. Springer-Verlag, New York, 1983.

[117] J. C. Pedro, M. K. Banda, and P. Sibanda. Implicit-explicit higher-order time integration schemes for computations of structural dynamics with fluid-structure interaction. *Appl. Appl. Math.*, 10(1):287–311, 2015.

[118] R. Rajaraman. *Solitons and Instantons: An Introduction to Solitons and Instantons in Quantum Field Theory*. North-Holland Personal Library. North-Holland, Amsterdam, 1982.

[119] F. V. Ravelo, P. S. Peixoto, and M. Schreiber. An explicit exponential integrator based on Faber polynomials and its application to seismic wave modeling. *J. Sci. Comput.*, 98(2):Paper No. 32, 39, 2024.

[120] T. J. Rivlin. *Chebyshev polynomials*. Pure and Applied Mathematics (New York). John Wiley & Sons, Inc., New York, second edition, 1990. From approximation theory to algebra and number theory.

[121] Y. Saad. Krylov subspace methods for solving large unsymmetric linear systems. *Math. Comp.*, 37(155):105–126, 1981.

[122] Y. Saad. Chebyshev acceleration techniques for solving nonsymmetric eigenvalue problems. *Math. Comp.*, 42(166):567–588, 1984.

[123] Y. Saad. Analysis of some Krylov subspace approximations to the matrix exponential operator. *SIAM J. Numer. Anal.*, 29(1):209–228, 1992.

[124] Y. Saad. Analysis of some Krylov subspace approximations to the matrix exponential operator. *SIAM J. Numer. Anal.*, 29(1):209–228, 1992.

[125] Y. Saad. *Numerical methods for large eigenvalue problems*. Algorithms and Architectures for Advanced Scientific Computing. Manchester University Press, Manchester; Halsted Press [John Wiley & Sons, Inc.], New York, 1992.

[126] Y. Saad. *Iterative methods for sparse linear systems*. Society for Industrial and Applied Mathematics, Philadelphia, PA, second edition, 2003.

[127] Y. Saad. *Numerical methods for large eigenvalue problems*, volume 66 of *Classics in Applied Mathematics*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, revised edition, 2011.

[128] J. A. Scott. An Arnoldi code for computing selected eigenvalues of sparse, real, unsymmetric matrices. *ACM Trans. Math. Software*, 21(4):432–475, 1995.

[129] V. Serov. *Fourier series, Fourier transform and their applications to mathematical physics*, volume 197 of *Applied Mathematical Sciences*. Springer, Cham, 2017.

[130] J. R. Silvester. Determinants of block matrices. *The Mathematical Gazette*, 84(501):460–467, 2000.

[131] B. Smith and H. Zhang. Sparse triangular solves for ILU revisited: data layout crucial to better performance. *The International Journal of High Performance Computing Applications*, 25(4):386–391, 2011.

[132] D. E. Stewart and T. S. Leyk. Error estimates for Krylov subspace approximations of matrix exponentials. *J. Comput. Appl. Math.*, 72(2):359–369, 1996.

[133] K. Strehmel, R. Weiner, and H. Podhaisky. *Numerik gewöhnlicher Differentialgleichungen: Nichtsteife, steife und differential-algebraische Gleichungen*. Vieweg+Teubner Verlag, Wiesbaden, Germany, 2 edition, 2012.

[134] P. Suetin. The basic properties of Faber poynomials. *Uspehi Mat. Nauk*, 19(4(118)):121–149, 1964.

[135] N. Svanstedt. G-convergence of parabolic operators. *Nonlinear Anal.*, 36(7):807–842, 1999.

[136] M. Tokman. Efficient integration of large stiff systems of ODEs with exponential propagation iterative (EPI) methods. *J. Comput. Phys.*, 213(2):748–776, 2006.

[137] L. N. Trefethen, editor. *Spectral methods in MATLAB*, volume 10 of *Software, Environments, and Tools*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2000.

[138] J. van den Eshof and M. Hochbruck. Preconditioning Lanczos approximations to the matrix exponential. *SIAM J. Sci. Comput.*, 27(4):1438–1457, 2006.

[139] J. G. Verwer, J. G. Blom, and W. Hundsdorfer. An implicit-explicit approach for atmospheric transport-chemistry problems. *Appl. Numer. Math.*, 20(1-2):191–209, 1996. Workshop on the method of lines for time-dependent problems (Lexington, KY, 1995).

[140] J. L. Walsh. *Approximation by polynomials in the complex domain*. Number 73 in Mémorial des sciences mathématiques. Gauthier-Villars, 1935.

[141] X. Wang and R. Xu. Global existence and finite time blowup for a nonlocal semilinear pseudo-parabolic equation. *Adv. Nonlinear Anal.*, 10(1):261–288, 2021.

[142] H. Yang and Y. Han. Initial boundary value problem for a strongly damped wave equation with a general nonlinearity. *Evol. Equ. Control Theory*, 11(3):635–648, 2022.