Restarting and error estimation in polynomial and extended Krylov subspace methods for the approximation of matrix functions

Dissertation

Bergische Universität Wuppertal
Fakultät für Mathematik und Naturwissenschaften

eingereicht von
Marcel Schweitzer, M. Sc.
zur Erlangung des Grades eines Doktors der Naturwissenschaften

Betreut durch Prof. Dr. Andreas Frommer und Dr. Stefan Güttel

Angefertigt in der Zeit vom

01.12.2011 – 22.10.2015

Wuppertal, 22.10.2015

Die Dissertation kann wie folgt zitiert werden:

urn:nbn:de:hbz:468-20160212-112106-7
[http://nbn-resolving.de/urn/resolver.pl?urn=urn%3Anbn%3Ade%3Ahbz%3A468-20160212-112106-7]

ACKNOWLEDGMENTS

I wish to thank Prof. Dr. Andreas Frommer for giving me the opportunity to write this thesis, for his support while doing so, and for raising my interest in numerical linear algebra in the first place.

I also wish to thank Dr. Stefan Güttel for his support and for two very nice and fruitful stays in Manchester, and Prof. Dr. Bruno Lang and Prof. Dr. Birgit Jacob for agreeing to be members of my examination board.

In addition, I would like to thank all those people around me who accompanied me on my path to finishing this thesis and somehow managed to survive my everyday madness. Among these people, a very, very special "Thank you" goes to Sonja, as without her support and patience it would have been impossible for me to accomplish all this.

CONTENTS

CHAPTER 1

INTRODUCTION

Given a matrix $A \in \mathbb{C}^{n\times n}$, a vector $\boldsymbol{b} \in \mathbb{C}^n$ and a sufficiently smooth function f defined on $\operatorname{spec}(A)$, an increasingly important task in many areas of numerical linear algebra and scientific computing is the computation of

$$f(A)\boldsymbol{b}, \tag{1.1}$$

the action of the *matrix function* $f(A)$ on the vector $\boldsymbol{b}$. Important examples of matrix functions include the matrix exponential $f(A) = e^A$ which is used, e.g., in exponential integrators for the solution of differential equations [88–90] and in network analysis [49], the matrix sign function $f(A) = \operatorname{sign}(A)$ which has important applications in lattice quantum chromodynamics [18, 48], or the (inverse) fractional powers $f(A) = A^{\pm\alpha}$ for $\alpha \in (0, 1)$ which are, e.g., used in fractional differential equations [23] and statistical sampling [93].

The presumably most widely known special case of the computation of a matrix function times a vector is the solution of a linear system of equations, i.e., the computation of $\boldsymbol{x} \in \mathbb{C}^n$ such that

$$A\boldsymbol{x} = \boldsymbol{b}, \tag{1.2}$$

which corresponds to evaluating (1.1) with $f(A) = A^{-1}$.

While both (1.1) and its special case (1.2) are often solved by the same or very closely related iterative methods, specifically *Krylov subspace methods* [37,67,88, 98,114], the special structure of (1.2) and the simple nature of the function $f(A) = A^{-1}$ allow for many theoretical and algorithmic simplifications and advantages which are not available in the more general case of an arbitrary function f.

The main goal of this thesis is to fill some of these gaps by transferring or generalizing techniques and results which are well-known in the linear system case to the case of more general matrix functions. Many of the results of this thesis deal with the class of so-called *Stieltjes functions* [14, 15, 83] (but are in some cases also applicable to broader classes of functions) which can be characterized by a Riemann–Stieltjes integral representation of the form

$$f(z) = \int_0^\infty \frac{1}{z+t} \, \mathrm{d}\mu(t), \quad z \in \mathbb{C} \setminus \mathbb{R}_0^-, \tag{1.3}$$

where μ is a nonnegative, monotonically increasing function defined on $\mathbb{R}_0^+$. Substituting the matrix A for z in (1.3) and applying this matrix function to a vector $\boldsymbol{b}$ yields

$$f(A)\boldsymbol{b} = \int_0^\infty (A+tI)^{-1}\boldsymbol{b} \, \mathrm{d}\mu(t), \tag{1.4}$$

which already reveals the intimate relation between Stieltjes matrix functions and (shifted) linear systems. This connection is the main building block of most of the ideas employed in this thesis.

There are two main concepts investigated in this thesis. On the one hand, we consider *restarting* of Krylov subspace methods, a technique well-known in the linear system context for methods such as GMRES [116] or FOM [113] for limiting memory requirements of these methods. On the other hand, we deal with the efficient computation of error bounds and estimates, which is of special importance in case of the approximation of matrix functions, because in contrast to the linear system case, no quantity like a residual is available to easily monitor the progress of the method.

The remainder of this thesis is organized as follows. In Chapter 2, basic material necessary for making this thesis self-contained is presented. We begin with the precise definition and important properties of matrix functions in general and Stieltjes functions in particular. This is followed by a review of Krylov subspace methods, both for matrix functions and for the special case of linear systems, and a short overview of numerical quadrature rules (Gauss quadrature, in particular) which will be extensively used in the computational methods presented in this thesis for evaluating integral representations of matrix functions such as (1.4). In addition, we introduce a few model problems which will be used throughout the thesis for illustrating the developed results by numerical experiments. In Chapter 3, we derive an integral representation for the error $f(A)\boldsymbol{b} - \boldsymbol{f}_m$ of the iterate $\boldsymbol{f}_m$ produced by m steps of a Krylov subspace method. This representation constitutes the common basis for the restart approach and the error estimates in the later chapters of the thesis. In Chapter 4 we first give an overview of the restart

approaches for Krylov subspace methods for $f(A)\boldsymbol{b}$ available in the literature so far. Afterwards, we investigate the possibility of using the error representation from the previous chapter for a new implementation of the restart approach and comment on the differences and advantages in comparison to existing methods. We already published the resulting method in [58]. Chapter 5 deals with convergence of restarted Krylov subspace methods for the approximation of matrix functions. After reviewing the few previously known convergence results, we prove convergence of the restarted Arnoldi method for f a Stieltjes function and A Hermitian positive definite (for all restart lengths). In addition, we propose a variation of Arnoldi's method based on interpolation in harmonic Ritz values which allows to prove convergence for a larger class of matrices, the so-called positive real matrices. We published these results in [57]. We conclude the chapter by investigating the linear system case and presenting results on the convergence behavior of restarted FOM and restarted GMRES. We presented some of these results (partially in a weaker form) already in the technical report [119]. Chapter 6 deals with the estimation and bounding of the norm of the error $\|f(A)\boldsymbol{b} - \boldsymbol{f}_m\|_2$ in Krylov subspace methods by making use of the error representation from Chapter 3 combined with techniques developed in [72–74] for error estimation in the iterative solution of linear systems. As these error bounds rely on the relation between the Lanczos process and Gauss quadrature, evaluating the integral representation of the error in this context gives rise to a *nested* quadrature approach with an inner and an outer quadrature rule. Special care is devoted to the task of combining inner and outer quadrature rules in such a way that (in certain situations, e.g., for Hermitian positive definite matrices A and f a Stieltjes function) the error estimates are guaranteed to be upper or lower bounds for the exact error norm. In addition, we show how it is possible to compute these error bounds with negligible computational cost, which is independent both of the matrix dimension and the number of iterations performed in the Krylov subspace method, when A is Hermitian positive definite, or at least independent of the matrix dimension for non-Hermitian A. Most of the results from this chapter (those applying to Stieltjes functions) can also be found in our preprint [63]. In Chapter 7, results similar to the ones from Chapter 6 are presented in the context of *extended Krylov subspace methods* [38, 99, 124, 125]. These subspaces are built not only by using powers of the matrix A but also powers of A^{-1}. Thus, they result in rational approximations to $f(A)\boldsymbol{b}$ instead of polynomial approximations, therefore making the situation slightly more involved to analyze. We demonstrate how to transfer the techniques from the previous chapter to this situation, comment on the possibility to also use *rational Gauss quadrature rules* for computing error estimates and investigate in which situations one can still expect to obtain lower and upper bounds for the error. In Chapter 8, the results of this thesis are summarized and concluding remarks and topics for future research are given.

CHAPTER 2

REVIEW OF BASIC MATERIAL

In this chapter, we introduce and review the basic terminology and classical results on which the remainder of this thesis is based. We begin by presenting different possible definitions for matrix functions $f(A)$ and important properties which directly follow from these definitions in Section 2.1. In Section 2.2 we review the definition of the Riemann–Stieltjes integral and use it to define the class of *Stieltjes functions*. These are the functions which we will mostly investigate throughout the remainder of the thesis, as their special structure gives rise to a lot of computational and theoretical advantages. We present some examples of Stieltjes functions and give an overview of classical results from the literature which will become useful in later chapters. Next, Krylov subspace methods for approximating $f(A)\boldsymbol{b}$ are described in Section 2.3. We do not only cover the case of approximating a general matrix function f but also present some of the simplifications and theoretical results arising when $f(z) = z^{-1}$ in Section 2.4, i.e., when the solution of a linear system $A\boldsymbol{x} = \boldsymbol{b}$ is approximated. These results will later become beneficial when we investigate the intimate relation between the solution of *shifted* linear systems and approximating certain functions of matrices. In Section 2.5, we give a short overview of numerical quadrature rules, with a special emphasis on Gauss quadrature. Gauss quadrature will be important in two ways in this thesis. First, we will often work with (Riemann–Stieltjes) integral representations of functions for which no closed form is known, so that the integrals have to be evaluated numerically, and second, we will use the strong relation between Gauss quadrature and the Lanczos process for computing error estimates and error bounds in (extended) Krylov subspace methods in Chapter 6 and Chapter 7. In the final section of this chapter, we introduce different model problems which involve the approximation of a matrix function times a vector

and which will be used as benchmarks at various places throughout this thesis to illustrate and evaluate the developed methods and results.

2.1 Functions of matrices

In this section, we review the definition of a matrix function $f(A)$ and basic properties of matrix functions which we will use throughout this thesis. Most of our presentation, including the three classical (and, if applicable, equivalent) definitions of a matrix function, mainly follows [85, Chapter 1], with additional material and inspiration drawn from [64, 71, 91]. We focus solely on theory of matrix functions in this section, deferring computational and algorithmic issues to Section 2.3.

Each of the three definitions of a matrix function presented in the following has different advantages in different situations and most notably provides different angles of insight concerning the nature and behavior of matrix functions.

Throughout the remainder of this section, we use the following notation. We denote the spectrum of A by $\operatorname{spec}(A) = \{\lambda_1, \dots, \lambda_s\}$, where $\lambda_1, \dots, \lambda_s$ are the *distinct* eigenvalues of A. In addition, we denote by n_i the *index* of the eigenvalue λ_i, i.e., the size of the largest Jordan block $J_k(\lambda_i)$ corresponding to λ_i in the Jordan canonical form $A = WJW^{-1}$, where $J = \operatorname{diag}(J_1(\lambda_{i_1}), \dots, J_p(\lambda_{i_p}))$ with Jordan blocks

$$J_k(\lambda_{i_k}) = \begin{bmatrix} \lambda_{i_k} & 1 & & \\ & \lambda_{i_k} & \ddots & \\ & & \ddots & 1 \\ & & & \lambda_{i_k} \end{bmatrix} \in \mathbb{C}^{m_k \times m_k}.$$

Recall that one eigenvalue may correspond to more than one Jordan block of A. We say that a function f is defined on the spectrum of A if the values

$$f^{(j)}(\lambda_i), \qquad j = 0, \dots, n_i - 1, \quad i = 1, \dots, s \tag{2.1}$$

all exist. If this requirement is fulfilled, the matrix function $f(A)$ in the sense of the following definition is well-defined.

Definition 2.1. Let $A \in \mathbb{C}^{n \times n}$ with Jordan canonical form $A = WJW^{-1}$ and let f be defined on the spectrum of A. Then

$$f(A) := Wf(J)W^{-1} := W \operatorname{diag}\big(f(J_1(\lambda_{i_1})), \dots, f(J_p(\lambda_{i_p}))\big)W^{-1}, \tag{2.2}$$

where the function f evaluated at the Jordan blocks $J_k(\lambda_{i_k})$ is defined by

$$f(J_k(\lambda_{i_k})) := \begin{bmatrix} f(\lambda_{i_k}) & f'(\lambda_{i_k}) & \dots & \frac{f^{(m_k-1)}(\lambda_{i_k})}{(m_k-1)!} \\ & f(\lambda_{i_k}) & \ddots & \vdots \\ & & \ddots & f'(\lambda_{i_k}) \\ & & & f(\lambda_{i_k}) \end{bmatrix}.$$

A particular special case, which is very important in practice, is given for diagonalizable A, i.e., when the Jordan canonical form of A reduces to $A = W\Lambda W^{-1}$ with a diagonal matrix $\Lambda = \operatorname{diag}(\lambda_1, \dots, \lambda_n)$ (where this time, we count multiple eigenvalues individually). In this case,

$$f(A) = W f(\Lambda) W^{-1} \text{ where } f(\Lambda) = \operatorname{diag}(f(\lambda_1), \dots, f(\lambda_n)), \tag{2.3}$$

i.e., no derivatives of f are needed. This relation can indeed be used in practice to compute $f(A)$ for small matrices A, where it is feasible to compute a full eigenvalue decomposition (like it is, e.g., the case for the Hessenberg matrices H_m from Arnoldi's method, cf. Section 2.3, after a moderate number m of steps). However, for a general diagonalizable matrix A the eigenvector basis may be ill-conditioned, making (2.3) unstable in the presence of round-off error. When A is Hermitian, there exists an orthonormal eigenvector basis, so that W can be chosen as a unitary matrix, i.e., $W^{-1} = W^H$ and (2.3) can be evaluated in a numerically stable way. An immediate consequence of Definition 2.1 is that the eigenvalues of $f(A)$ are just $f(\lambda_i)$, as $f(A)$ is similar to $f(J)$ from (2.2).

Another way of defining a function of a matrix is based on polynomial interpolation and provides the main motivation for using Krylov subspace methods for approximating the action of a matrix function on a vector. It again requires f to be defined on $\operatorname{spec}(A)$ in the sense of (2.1).

Definition 2.2. Let $A \in \mathbb{C}^{n\times n}$, let f be defined on the spectrum of A and let ψ be the minimal polynomial of A. Then $f(A) := p(A)$, where p is the unique polynomial of degree less than $\deg \psi$ that interpolates f on $\operatorname{spec}(A)$, i.e.,

$$p^{(j)}(\lambda_i) = f^{(j)}(\lambda_i), \qquad j = 0, \dots, n_i - 1, \quad i = 1, \dots, s, \tag{2.4}$$

the so-called *Hermite interpolating polynomial*.

Definition 2.2 sheds light on some interesting properties of matrix functions. Immediate consequences are that *every* matrix function is a polynomial in A and that a matrix function is already uniquely defined by its values on a discrete, finite set, the spectrum of A. This in turn means that if two functions f and g coincide on the spectrum of A, then $f(A) = g(A)$, no matter which values f and g attain outside of $\operatorname{spec}(A)$. It is important to note, however, that $f(A) = p(A)$ for some fixed polynomial p does not hold independently of A, but that the polynomial p depends on A (or, to be precise, the Jordan structure of A) as well as on f, through the Hermite interpolation conditions (2.4).

The characterization of f as a polynomial in A is, in addition to the consequences mentioned above, especially useful because it directly implies a lot of important properties of matrix functions which are collected in the following lemma.

Lemma 2.3. *Let $A \in \mathbb{C}^{n\times n}$ and let f be defined on* $\operatorname{spec}(A)$*. Then the following properties hold.*

(i) $f(A)$ commutes with A,

(ii) if $X \in \mathbb{C}^{n\times n}$ commutes with A, then X commutes with $f(A)$,

(iii) if $X \in \mathbb{C}^{n\times n}$ is nonsingular, then $f(XAX^{-1}) = Xf(A)X^{-1}$.

Proof. All properties directly follow from the fact that $f(A) = p(A)$ for some polynomial p, see, e.g., [85, Theorem 1.13]. □

A third possible, and particularly elegant, way of defining a matrix function is given by the Cauchy integral formula. While it requires f to be analytic (where the other two definitions do not even require f to be continuous or defined outside of a finite set as long as A has no multiple eigenvalues) it has the advantage of allowing to generalize the notion of matrix functions to *operator functions* on infinite dimensional vector spaces, cf., e.g., [80]. Although we will not further pursue this approach in this thesis, the following definition (and variants thereof) will nonetheless prove useful.

Definition 2.4. Let $A \in \mathbb{C}^{n\times n}$ and let f be analytic on and inside a closed contour Γ that winds around $\operatorname{spec}(A)$ exactly once. Then

$$f(A) = \frac{1}{2\pi i}\int_\Gamma f(t)(tI - A)^{-1}\,\mathrm{d}t. \tag{2.5}$$

This definition of a matrix function is not restricted to the case of Cauchy integral representations but can also be used for other integral representations of f, for example for Stieltjes integral representations which will be discussed in Section 2.2 and will be the foundation of most of the results developed throughout the remainder of this thesis.

Of course, using different definitions of functions of matrices for developing the ideas of this thesis is only reasonable if all of these definitions agree (when applicable). This is indeed the case.

Theorem 2.5. *Let $A \in \mathbb{C}^{n\times n}$ and let f be defined on* $\operatorname{spec}(A)$ *in the sense of* (2.1). *Then Definition 2.1 and Definition 2.2 for $f(A)$ are equivalent. If f is in addition analytic in a region $\Omega \supset \operatorname{spec}(A)$, then Definition 2.4 for $f(A)$ is equivalent to Definition 2.1 and Definition 2.2.*

Proof. See, e.g., [85, Theorem 1.12] and [91, Theorem 6.2.28]. □

2.2 Stieltjes functions

In this section, we introduce the class of *Stieltjes functions*, which contains many functions of practical interest, like inverse fractional powers or rational functions of the logarithm. As this class of functions is defined by means of a Riemann–Stieltjes integral representation in the classical literature, we first review the basics of this integral concept. Afterwards, we define the class of Stieltjes functions, give some examples of functions from this class and present some basic properties which we will need for developing our results in later chapters of this thesis.

2.2.1 The Riemann–Stieltjes integral

The Riemann–Stieltjes integral can be seen as a generalization of the Riemann integral, in which integration of a function g is performed with respect to some other function μ (with the Riemann integral as special case when the function μ is chosen as the identity function $\mu(t) = t$) and was first introduced in [130]. To properly define the Riemann–Stieltjes integral, we first need the following prerequisites.

Definition 2.6. Let $[a,b] \subset \mathbb{R}$ be a finite interval. A *subdivision* of $[a,b]$ is a finite sequence $(\tau_i)_{i=0,\dots,m}$ of real numbers that satisfy

$$a = \tau_0 < \tau_1 < \cdots < \tau_m = b.$$

The *norm of* $(\tau_i)_{i=1,\dots,m}$ is defined as

$$|(\tau_i)_{i=0,\dots,m}| := \max_{1\le i\le m} \{\tau_i - \tau_{i-1}\}.$$

A sequence $(\sigma_i)_{i=1,\dots,m}$ of real numbers is called *sequence of pivotal points consistent with* $(\tau_i)_{i=0,\dots,m}$ if it satisfies

$$\tau_{i-1} \le \sigma_i \le \tau_i \text{ for } i = 1,\dots,m.$$

The Riemann–Stieltjes integral of g with respect to μ can now be defined analogously to the Riemann integral.

Definition 2.7. Let $[a,b] \subset \mathbb{R}$ be a finite interval, let g be a complex-valued function and let μ be a real-valued function, both defined on $[a,b]$. Further, let $(\tau_i)_{i=0,\dots,m}$ be a subdivision of $[a,b]$ and let $(\sigma_i)_{i=1,\dots,m}$ be a sequence of pivotal points consistent with $(\tau_i)_{i=0,\dots,m}$. Then the *Riemann–Stieltjes sum of* g *and* μ *corresponding to* $(\tau_i)_{i=0,\dots,m}$ *and* $(\sigma_i)_{i=1,\dots,m}$ is defined as

$$S\big((\tau_i)_{i=0,\dots,m}, (\sigma_i)_{i=1,\dots,m}\big) = \sum_{i=1}^{m} g(\sigma_i)\big(\mu(\tau_i) - \mu(\tau_{i-1})\big).$$

If there exists $S \in \mathbb{C}$ such that for any $\varepsilon > 0$ there exists $\delta > 0$ satisfying

$$\big|S\big((\tau_i)_{i=0,\dots,m}, (\sigma_i)_{i=1,\dots,m}\big) - S\big| < \varepsilon$$

for all subdivisions $(\tau_i)_{i=0,\dots,m}$ and consistent choices of $(\sigma_i)_{i=1,\dots,m}$ with $|(\tau_i)_{i=0,\dots,m}| < \delta$, then S is called the *Riemann–Stieltjes integral of* g *with respect to* μ *on* $[a,b]$ and is denoted by

$$S =: \int_a^b g(t)\,\mathrm{d}\mu(t). \tag{2.6}$$

The function g is called the *integrand* and μ is called the *integrator* of the Riemann–Stieltjes integral (2.6).

Note that for $\mu(t) = t$ (or $\mu(t) = t + c$ for some constant $c \in \mathbb{R}$), Definition 2.7 reduces to the definition of the ordinary Riemann integral. Another connection between Riemann and Riemann–Stieltjes integrals is given by the following, classical result.

Lemma 2.8. *Let $[a, b] \subset \mathbb{R}$ be a finite interval, let g be continuous on $[a, b]$ and let μ be continuously differentiable on $[a, b]$. Then*

$$\int_a^b g(t)\,\mathrm{d}\mu(t) = \int_a^b g(t)\mu'(t)\,\mathrm{d}t.$$

Proof. See [121, Theorem 9.55b]. □

For a continuously differentiable integrator μ, the Riemann–Stieltjes integral thus reduces to an ordinary Riemann integral.

Example 2.9. A special case of a Riemann–Stieltjes integral corresponding to a nondifferentiable integrator is given when μ is a step function with jumps of size $\mu_1, \ldots, \mu_\ell$ at the points $t_1, \ldots, t_\ell$, i.e.,

$$\mu(t) = \begin{cases} 0 & a \le t \le t_1 \\ \mu_1 & t_1 < t \le t_2 \\ \mu_1 + \mu_2 & t_2 < t \le t_3 \\ \vdots & \vdots \\ \mu_1 + \cdots + \mu_\ell & t_\ell < t \le b. \end{cases}$$

In this case, the Riemann–Stieltjes integral of a continuous function g reduces to a finite sum, cf. [83, Section 12.9, Example 3],

$$\int_a^b g(t)\,\mathrm{d}\mu(t) = \sum_{i=1}^{\ell} g(t_i)\mu_i.$$

This observation will later prove useful for establishing a connection between rational functions in partial fraction form and Stieltjes functions, cf. Example 2.14.

We proceed by collecting some basic, easy to prove properties of the Riemann–Stieltjes integral which mostly generalize well-known properties of the Riemann integral.

Proposition 2.10. *Let $[a,b] \subset \mathbb{R}$ be a finite interval, let g, g_1, g_2 be complex-valued functions on $[a,b]$ and let μ, μ_1, μ_2 be real-valued functions on $[a,b]$. Then*

(i) The Riemann–Stieltjes integral is linear in the integrand, i.e.,

$$\int_a^b g_1(t) + g_2(t)\,\mathrm{d}\mu(t) = \int_a^b g_1(t)\,\mathrm{d}\mu(t) + \int_a^b g_2(t)\,\mathrm{d}\mu(t), \tag{2.7}$$

and, for a constant $c \in \mathbb{C}$,

$$\int_a^b cg(t)\,\mathrm{d}\mu(t) = c\int_a^b g(t)\,\mathrm{d}\mu(t). \tag{2.8}$$

(ii) The Riemann–Stieltjes integral is linear in the integrator, i.e.,

$$\int_a^b g(t)\,\mathrm{d}(\mu_1(t) + \mu_2(t)) = \int_a^b g(t)\,\mathrm{d}\mu_1(t) + \int_a^b g(t)\,\mathrm{d}\mu_2(t), \tag{2.9}$$

and, for a constant $c \in \mathbb{R}$,

$$\int_a^b g(t)\,\mathrm{d}(c\mu(t)) = c\int_a^b g(t)\,\mathrm{d}\mu(t). \tag{2.10}$$

(iii) For $a < c < b$ it holds

$$\int_a^b g(t)\,\mathrm{d}\mu(t) = \int_a^c g(t)\,\mathrm{d}\mu(t) + \int_c^b g(t)\,\mathrm{d}\mu(t) \tag{2.11}$$

provided that all integrals in (2.11) exist.

(iv) If μ is monotonically increasing on $[a,b]$, then

$$\int_a^b \mathrm{d}\mu(t) := \int_a^b 1\,\mathrm{d}\mu(t) = \mu(b) - \mu(a).$$

(v) If μ is monotonically increasing on $[a,b]$ and g_1, g_2 are real-valued with $g_1(t) \le g_2(t)$ for all $t \in [a,b]$, then

$$\int_a^b g_1(t)\,\mathrm{d}\mu(t) \le \int_a^b g_2(t)\,\mathrm{d}\mu(t).$$

Proof. See [26, Theorem 5.1.5], [108, Section VIII.6] and [121, Section 9.55c]. □

Note that in assertion (i) and (ii) of Proposition 2.10, the existence of the integrals on the right-hand sides of equations (2.7), (2.8), (2.9) and (2.10) imply the existence of the integrals on the left-hand sides.

Just as for Riemann integrals, improper Riemann–Stieltjes integrals may be defined.

Definition 2.11. Let $a \in \mathbb{R}$, let g be a continuous, complex-valued function and let μ be a real-valued function on $[a, \infty)$. Then the *improper Riemann–Stieltjes integral of g with respect to μ on $[a, \infty)$* is defined as

$$\int_a^\infty g(t)\,\mathrm{d}\mu(t) := \lim_{b\to\infty} \int_a^b g(t)\,\mathrm{d}\mu(t),$$

provided that the limit exists.

There is a wide variety of results on assumptions necessary for the existence of (proper and improper) Riemann–Stieltjes integrals; see, e.g., [108, 121]. We will not go into detail on this in general rather important topic, as we are primarily interested in Stieltjes integrals with integrand $g(t) = \frac{1}{z+t}$ for $z \in \mathbb{C} \setminus \mathbb{R}_0^-$, for which the question of existence is easier to analyze than in the general case; cf. Section 2.2.2.

Before proceeding, we state one additional result which will prove useful for estimating error norms when investigating the convergence behavior of Krylov subspace methods for Stieltjes matrix functions.

Lemma 2.12. *Let $a \in \mathbb{R}$, let $g : [a, \infty) \longrightarrow \mathbb{C}^n$ be a vector-valued function, i.e.,*

$$g(t) = [g_1(t), g_2(t), \ldots, g_n(t)]^T$$

with $g_i : [a, \infty) \longrightarrow \mathbb{C}$. Further, let μ be real-valued and monotonically increasing on $[a, \infty)$, such that all integrals

$$\int_a^\infty g_i(t)\,\mathrm{d}\mu(t)$$

exist and let $\|\cdot\|$ be a norm on $\mathbb{C}^n$. Then

$$\left\| \int_a^\infty g(t)\,\mathrm{d}\mu(t) \right\| \leq \int_a^\infty \|g(t)\|\,\mathrm{d}\mu(t), \tag{2.12}$$

where the integral on the left-hand side of (2.12) is understood component-wise.

Proof. Let $b > a$, let $T^{(j)} = (\tau_i^{(j)})_{i=0,\dots,m_j}$ be a sequence of subdivisions of $[a,b]$ with $|T^{(j)}| \to 0$ for $j \to \infty$ and let $\Sigma^{(j)} = (\sigma_i^{(j)})_{i=1,\dots,m_j}$ be a sequence of consistent sequences of pivotal points. We define the vector-valued analogue to the Riemann–Stieltjes sum from Definition 2.7 as

$$S(T^{(j)}, \Sigma^{(j)}) = \sum_{i=1}^{m_j} \left[g_1(\sigma_i^{(j)}), \dots, g_n(\sigma_i^{(j)}) \right]^T \left(\mu(\tau_i^{(j)}) - \mu(\tau_{i-1}^{(j)}) \right),$$

i.e., the pivotal points are inserted into each individual component g_i of g. Then, by applying Definition 2.7 to each component individually, we have

$$\lim_{j\to\infty} S(T^{(j)}, \Sigma^{(j)}) = \int_a^b g(t)\,\mathrm{d}\mu(t), \tag{2.13}$$

We further have for any j

$$\begin{aligned} \|S(T^{(j)}, \Sigma^{(j)})\| &= \left\| \sum_{i=1}^{m_j} g(\sigma_i^{(j)}) \left(\mu(\tau_i^{(j)}) - \mu(\tau_{i-1}^{(j)}) \right) \right\| \\ &\le \sum_{i=1}^{m_j} \left\| g(\sigma_i^{(j)}) \left(\mu(\tau_i^{(j)}) - \mu(\tau_{i-1}^{(j)}) \right) \right\| \\ &= \sum_{i=1}^{m_j} \|g(\sigma_i^{(j)})\| \left(\mu(\tau_i^{(j)}) - \mu(\tau_{i-1}^{(j)}) \right), \end{aligned} \tag{2.14}$$

where the inequality holds due to the triangle inequality and the last equality holds because μ is monotonically increasing on $[a,b]$. By taking the norm on both sides of (2.13) and inserting (2.14), we obtain

$$\left\| \int_a^b g(t)\,\mathrm{d}\mu(t) \right\| = \left\| \lim_{j\to\infty} S(T^{(j)}, \Sigma^{(j)}) \right\| = \int_a^b \|g(t)\|\,\mathrm{d}\mu(t). \tag{2.15}$$

By taking the limit $b \to \infty$ inside the norm on the left-hand side of (2.15) and using the fact that $\|\cdot\|$ is continuous, we obtain the desired result. □

We remark that we do not make any statement about the existence of the integral on the right-hand side of (2.12), in the sense that if the integral is infinite, then infinity is taken as (trivial) upper bound for the left-hand side. At all places where we use the result of Lemma 2.12 in this thesis, we will individually investigate whether this integral is finite, as we do not need any general result about the finiteness of such integrals.

We will now turn our attention to the class of *Stieltjes functions* which are defined by means of a Riemann–Stieltjes integral of the resolvent function $g(t) = \frac{1}{z+t}$.

2.2.2 The Stieltjes cone

We first define the class of Stieltjes functions.

Definition 2.13. Let μ be a monotonically increasing, real-valued function on $\mathbb{R}_0^+$ such that

$$\int_0^\infty \frac{1}{1+t}\,\mathrm{d}\mu(t) < \infty, \tag{2.16}$$

and let $a \geq 0$. Then the function $f : \mathbb{C} \setminus \mathbb{R}_0^- \longrightarrow \mathbb{C}$ defined via

$$f(z) = a + \int_0^\infty \frac{1}{z+t}\,\mathrm{d}\mu(t) \tag{2.17}$$

is called *Stieltjes function corresponding to* μ. The function μ is also called *generating function* of f.

Note that the condition (2.16) imposed on μ is sufficient for f being defined (and holomorphic) in all $z \in \mathbb{C} \setminus \mathbb{R}_0^-$. The set of all Stieltjes functions forms a convex cone, i.e., it is closed under addition and under multiplication by nonnegative scalars. For both properties, see, e.g., [14, Section 3]. From now on we will, without loss of generality, always assume $a = 0$ in (2.17).

Before discussing useful properties of Stieltjes functions, we first list a few examples of important functions belonging to this class.

Example 2.14. The following functions are Stieltjes functions.

(i) The function $f(z) = z^{-1}$, generated by the step function

$$\mu(t) = \begin{cases} 0 & t = 0, \\ 1 & t > 0. \end{cases}$$

(ii) Rational functions in partial fraction form with poles on the negative real axis,

$$f(z) = \sum_{i=0}^{\ell} \frac{\mu_i}{z + t_i},$$

generated by the step function

$$\mu(t) = \begin{cases} 0 & 0 \le t \le t_1 \\ \mu_1 & t_1 < t \le t_2 \\ \mu_1 + \mu_2 & t_2 < t \le t_3 \\ \vdots & \vdots \\ \mu_1 + \cdots + \mu_\ell & t_\ell < t \end{cases}$$

with $t_i \ge 0, \mu_i > 0, i = 1, \ldots, \ell$.

(iii) The function $f(z) = z^{-\alpha}$ for $\alpha \in (0, 1)$, because

$$z^{-\alpha} = \frac{\sin(\alpha\pi)}{\pi} \int_0^\infty \frac{t^{-\alpha}}{z+t} \, \mathrm{d}t. \tag{2.18}$$

(iv) The function $f(z) = \log(1+z)/z$, because

$$\frac{\log(1+z)}{z} = \int_1^\infty \frac{t^{-1}}{z+t} \, \mathrm{d}t. \tag{2.19}$$

Note that the functions in Example 2.14(iii) and (iv) correspond to continuously differentiable generating functions μ, so that they can be written as ordinary Riemann integrals by Lemma 2.8. For further examples of Stieltjes functions and proofs that the above functions indeed are Stieltjes functions in the sense of Definition 2.13, see, e.g., [14, 15, 55, 83, 130].

The following lemma gives a representation of the derivative of Stieltjes functions which will be useful for some results on error bounds in this thesis.

Lemma 2.15. *Let f be a Stieltjes function with generating function μ. Then f is infinitely many times continuously differentiable on $\mathbb{C} \setminus \mathbb{R}_0^-$ and*

$$f^{(k)}(z) = (-1)^k k! \int_0^\infty \frac{1}{(z+t)^{k+1}} \, \mathrm{d}\mu(t) \text{ for all } k \in \mathbb{N}_0.$$

Proof. See, e.g., [14, Section 3]. □

The class of Stieltjes functions is very closely related to the class of completely monotonic functions defined in the following.

Definition 2.16. A function $f : \mathbb{R}^+ \longrightarrow \mathbb{R}$ is called *completely monotonic* if it is infinitely many times continuously differentiable and satisfies

$$(-1)^k f^{(k)}(z) \geq 0 \text{ for } k \in \mathbb{N}_0 \text{ and } z \in \mathbb{R}^+.$$

The following result establishes the connection between Stieltjes functions and completely monotonic functions and gives another easy to prove but useful property of completely monotonic functions, see, e.g., [5, 14].

Proposition 2.17.

(i) Every Stieltjes function (or more precisely, its restriction to the positive real axis) is a completely monotonic function.

(ii) Let f_1, f_2 be completely monotonic functions. Then $f_1 \cdot f_2$ is a completely monotonic function.

Proof. Part (i) directly follows from Lemma 2.15 and Proposition 2.10(v) and part (ii) is a direct consequence of the Leibniz rule for product differentiation. □

We mention in passing that the set of Stieltjes functions is a proper subset of the class of completely monotonic functions, i.e., not every completely monotonic function is a Stieltjes function, as the following example, taken from [14], illustrates.

Example 2.18. Consider the function $f(z) = 1/(z(1 + z^2))$. One easily verifies that f is completely monotonic, but it has poles at $z = \pm i$, so that it cannot be a Stieltjes function.

The class of Stieltjes functions is of particular interest in our setting as the integral representation (2.17) directly transfers to the case of matrix functions, similar to the Cauchy integral representation (2.5) of analytic functions. For f a Stieltjes function with generating function μ and $A \in \mathbb{C}^{n\times n}$ with $\operatorname{spec}(A) \subseteq \mathbb{C} \setminus \mathbb{R}_0^-$, we directly have

$$f(A) = \int_0^\infty (A + tI)^{-1} \, \mathrm{d}\mu(t)$$

and thus

$$f(A)\boldsymbol{b} = \int_0^\infty (A + tI)^{-1}\boldsymbol{b}\,\mathrm{d}\mu(t). \tag{2.20}$$

According to (2.20), $f(A)\boldsymbol{b}$ for f a Stieltjes function can be interpreted as the integral over the solutions $\boldsymbol{x}(t)$ of the shifted linear systems

$$(A + tI)\boldsymbol{x}(t) = \boldsymbol{b}$$

for $t \geq 0$. This relation between the action of Stieltjes matrix functions on a vector and shifted linear systems with positive shifts is one of the building blocks of the results developed in this thesis. In particular, it allows to also establish a relation between Krylov subspace methods for approximating matrix functions and Krylov subspace methods for the approximate solution of linear systems. This in turn allows to transfer theoretical results from the latter (which are understood far better) to the former and will be the basis of the convergence analysis presented in Chapter 5. We continue by investigating Krylov subspace methods in detail in the next section.

2.3 Krylov subspace methods for $f(A)\boldsymbol{b}$

While Section 2.1 dealt with matrix functions $f(A)$, for the remainder of this thesis we will not focus on the computation of the matrix function $f(A)$ itself, but rather on the action of $f(A)$ on some vector $\boldsymbol{b} \in \mathbb{C}^n$, i.e.,

$$f(A)\boldsymbol{b}. \tag{2.21}$$

For techniques and algorithms related to the computation of $f(A)$ (for small and possibly dense matrices A), we refer to, e.g., [32,85,86] and the references therein.

One of the main computational difficulties when numerically evaluating (2.21) is that $f(A)$ is in general a full matrix, even when A is sparse or structured, with the one exception from this rule being that $f(A)$ is (block-)diagonal when A is (block-)diagonal. We just mention for the sake of completeness that when A is block upper (or lower) triangular, $f(A)$ will also inherit this property, but the upper (or lower) triangle will in general be completely filled, resulting in a matrix with $\mathcal{O}(n^2)$ nonzero entries, so that we consider this as a dense matrix in our setting. Therefore, even for moderate values of n, it may not even be possible to store the matrix $f(A)$, such that the naive approach of first computing $f(A)$ and then multiplying it to $\boldsymbol{b}$ is infeasible, notwithstanding the high computational cost.

Therefore, one typically tries to approximate the vector $f(A)\boldsymbol{b}$ directly by some iterative method. By far the most popular and most widely-used methods for this task belong to the class of *Krylov subspace methods* (or related classes like *extended* and general *rational Krylov subspace methods*).

2.3.1 The Arnoldi/Lanczos approximation for $f(A)\boldsymbol{b}$

We begin our exposition with the basic definition of a *Krylov subspace* corresponding to a matrix A and a vector $\boldsymbol{b}$, which is central to most of the results of this thesis.

Definition 2.19. Let $A \in \mathbb{C}^{n\times n}$ and let $\boldsymbol{b} \in \mathbb{C}^n$. The mth Krylov subspace with respect to A and $\boldsymbol{b}$ is defined as

$$\mathcal{K}_m(A, \boldsymbol{b}) = \{p_{m-1}(A)\boldsymbol{b} : p_{m-1} \in \Pi_{m-1}\}, \tag{2.22}$$

where Π_{m-1} denotes the set of all polynomials of degree at most $m-1$.

The idea of searching for an approximation to $f(A)\boldsymbol{b}$ in a Krylov subspace $\mathcal{K}_m(A, \boldsymbol{b})$ is quite obvious in light of Definition 2.2, as each matrix function is a polynomial (of degree at most $n-1$) in A, so that $f(A)\boldsymbol{b} \in \mathcal{K}_n(A, \boldsymbol{b})$. Approximations from Krylov subspaces of dimension $m < n$ can thus be interpreted as replacing the polynomial p from Definition 2.2 by another polynomial of lower degree. Before proceeding, we summarize some basic properties of $\mathcal{K}_m(A, \boldsymbol{b})$.

Proposition 2.20. *Let $A \in \mathbb{C}^{n\times n}$ and let $\boldsymbol{b} \in \mathbb{C}^n$. In addition, let m^* be the smallest integer such that there exists a polynomial $p_{m^*} \in \Pi_{m^*}$ which satisfies $p_{m^*}(A)\boldsymbol{b} = \boldsymbol{0}$. Then*

(i) $\mathcal{K}_m(A, \boldsymbol{b}) \subseteq \mathcal{K}_{m+1}(A, \boldsymbol{b})$ *for all* $m \geq 1$,

(ii) $\mathcal{K}_{m^*}(A, \boldsymbol{b})$ *is invariant under* A, *and* $\mathcal{K}_m(A, \boldsymbol{b}) = \mathcal{K}_{m^*}(A, \boldsymbol{b})$ *for all* $m \geq m^*$,

(iii) $\dim \mathcal{K}_m(A, \boldsymbol{b}) = \min\{m, m^*\}$.

Proof. Part (i) is directly obvious from (2.22). For part (ii), see, e.g., [115, Proposition 6.1] and for part (iii), see, e.g., [115, Proposition 6.2]. □

Property (i) from Proposition 2.20 means that Krylov subspaces are nested, and together with Property (iii) it follows that, as long as $m < m^*$, if $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_m$ is a basis of $\mathcal{K}_m(A, \boldsymbol{b})$, then there exists $\boldsymbol{v}_{m+1} \in \mathcal{K}_{m+1}(A, \boldsymbol{b}) \setminus \mathcal{K}_m(A, \boldsymbol{b})$ such that $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{m+1}$ is a basis of $\mathcal{K}_{m+1}(A, \boldsymbol{b})$.

This observation allows to iteratively construct a basis of the Krylov subspace $\mathcal{K}_m(A, \boldsymbol{b})$ by starting with a basis of $\mathcal{K}_1(A, \boldsymbol{b}) = \operatorname{span}\{\boldsymbol{b}\}$ and adding one basis

vector at a time. The most obvious choice for a basis of $\mathcal{K}_m(A, \boldsymbol{b})$ is the *Krylov basis*

$$\boldsymbol{b}, A\boldsymbol{b}, A^2\boldsymbol{b}, \ldots, A^{m-1}\boldsymbol{b},$$

but this basis can become severely ill-conditioned (the sequence of basis vectors converges to the dominant eigenvector of A which has a nonzero contribution to $\boldsymbol{b}$, such that the vectors will become almost linearly dependent for higher values of m). To circumvent this problem, and because of general favorable properties with respect to numerical stability, one seeks to construct an *orthonormal* basis of $\mathcal{K}_m(A, \boldsymbol{b})$. In *Arnoldi's method* [6,115] this is done iteratively as described above. In each iteration, a new basis vector is generated by multiplying the last basis vector with A and orthogonalizing the resulting vector against all previous basis vectors by a modified Gram–Schmidt procedure [115]. The overall procedure is described in Algorithm 2.1.

Algorithm 2.1: Arnoldi's method

Given: A, $\boldsymbol{b}$, m

$\boldsymbol{v}_1 \leftarrow \frac{1}{\|\boldsymbol{b}\|_2}\boldsymbol{b}$
for $j = 1, \ldots, m$ **do**
  $\boldsymbol{w}_j \leftarrow A\boldsymbol{v}_j$
  for $i = 1, \ldots, j$ **do**
    $h_{i,j} \leftarrow \boldsymbol{v}_i^H \boldsymbol{w}_j$
    $\boldsymbol{w}_j \leftarrow \boldsymbol{w}_j - h_{i,j}\boldsymbol{v}_i$
  $h_{j+1,j} \leftarrow \|\boldsymbol{w}_j\|_2$
  if $h_{j+1,j} = 0$ **then**
    Stop.
  $\boldsymbol{v}_{j+1} \leftarrow \frac{1}{h_{j+1,j}}\boldsymbol{w}_j$

For practical computations there exist many variations of Arnoldi's method, e.g., using Householder reflections for orthogonalization or applying some number of reorthogonalization steps to account for the numerical loss of orthogonality in later iterations. We will, however, not consider this further in this thesis and instead refer to, e.g., [77, 115] for details.

The correctness of Arnoldi's method is guaranteed by the following lemma which is proven by showing that $\boldsymbol{v}_j = q_{j-1}(A)\boldsymbol{v}_1$, where q_{j-1} is a polynomial of *exact* degree $j - 1$.

Lemma 2.21. *Assume that Algorithm 2.1 does not stop before the mth step. Then the vectors $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_m$ form an orthonormal basis of the Krylov subspace $\mathcal{K}_m(A, \boldsymbol{b})$.*

Proof. See [115, Proposition 6.4]. □

If the condition $h_{j+1,j} = 0$ is fulfilled in line 8 of Algorithm 2.1, the algorithm breaks down. The following lemma assures that in this case the Krylov subspace $\mathcal{K}_j(A, \boldsymbol{b})$ has reached the maximum possible dimension and is invariant under A.

Lemma 2.22. *Arnoldi's method breaks down at step j if and only if $j = m^*$ (with m^* as defined in Proposition 2.20). In this case, $\mathcal{K}_j(A, \boldsymbol{b})$ is invariant under A.*

Proof. See [115, Proposition 6.6]. □

Collecting the orthonormal basis vectors computed by Algorithm 2.1 in a matrix $V_m = [\boldsymbol{v}_1, \ldots, \boldsymbol{v}_m] \in \mathbb{C}^{n\times m}$ and the orthogonalization coefficients in an unreduced upper Hessenberg matrix $H_m = [h_{i,j}]_{i,j=1,\ldots,m} \in \mathbb{C}^{m\times m}$ yields the *Arnoldi decomposition*

$$AV_m = V_m H_m + h_{m+1,m}\boldsymbol{v}_{m+1}\hat{\boldsymbol{e}}_m^H \tag{2.23}$$

where $\hat{\boldsymbol{e}}_m \in \mathbb{C}^m$ denotes the mth canonical unit vector. The following result guarantees that the Arnoldi decomposition (2.23) is *essentially unique*, which will be useful in Chapter 6 and 7, where we compute decompositions of the form (2.23) by other means than by applying Algorithm 2.1 and can still be sure to obtain the same result.

Lemma 2.23. *Let $A \in \mathbb{C}^{n\times n}$ and let $[V, \boldsymbol{v}] \in \mathbb{C}^{n\times(m+1)}$ have orthonormal columns. If there exist an upper Hessenberg matrix $H \in \mathbb{C}^{m\times m}$ and a scalar $h \in \mathbb{C}$ such that*

$$AV = VH + h\boldsymbol{v}\hat{\boldsymbol{e}}_m^H$$

is fulfilled, then $V = V_m D$ and $H = D^H H_m D$, where $D \in \mathbb{C}^{m\times m}$ is a unitary diagonal matrix and H_m and V_m are the matrices from the Arnoldi decomposition (2.23) corresponding to the Krylov subspace $\mathcal{K}_m(A, \boldsymbol{v}_1)$, where $\boldsymbol{v}_1$ is the first column of V. In particular, if all subdiagonal entries of H are real and positive, then $V = V_m$ and $H = H_m$.

Proof. See [129, Chapter 5, Theorem 1.3]. □

By multiplying both sides of the relation (2.23) by V_m^H and exploiting the orthogonality of the $\boldsymbol{v}_i, i = 1, \ldots, m+1$, one finds

$$V_m^H A V_m = H_m, \tag{2.24}$$

showing that H_m can be interpreted as the (orthogonal) projection of A onto the Krylov subspace $\mathcal{K}_m(A, \boldsymbol{b})$. The identity (2.24) also allows to easily prove that substantial algorithmic and computational simplifications are possible in Arnoldi's method when the matrix A is Hermitian. By (2.24) it directly follows that H_m is Hermitian whenever A is Hermitian, and because H_m is in addition upper Hessenberg by construction, it must be tridiagonal in this case. This in turn

means that it is known in advance that the orthogonalization coefficients $h_{i,j}$ for $i < j-1$ are zero, or in other words, that $A\boldsymbol{v}_j$ is already orthogonal to $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{j-2}$. This allows for a simplified version of Arnoldi's method (which, in particular, has constant computational cost across all iterations because the orthogonalization process does not get more expensive from one iteration to the next), known as the *Lanczos method* [102, 115]. The resulting method is given as Algorithm 2.2 (note that it is implicitly assumed that the assignment $h_{j,j+1} \leftarrow h_{j+1,j}$ is performed if the tridiagonal matrix H_m is needed). Let us explicitly note that throughout this thesis we will also denote the tridiagonal matrix resulting from the Lanczos process as H_m, while in the literature it is typically denoted by T_m. As many—but not all—of our results apply to Hermitian and non-Hermitian matrices alike, we do not make this distinction in notation in order to not change notation from one result to the next.

Algorithm 2.2: Lanczos method

Given: A, $\boldsymbol{b}$, m

$\boldsymbol{v}_1 \leftarrow \frac{1}{\|\boldsymbol{b}\|_2}\boldsymbol{b}$
$h_{1,0} \leftarrow 0$
for $j = 1, \ldots, m$ **do**
  $\boldsymbol{w}_j \leftarrow A\boldsymbol{v}_j - h_{j,j-1}\boldsymbol{v}_{j-1}$
  $h_{j,j} \leftarrow \boldsymbol{v}_j^H \boldsymbol{w}_j$
  $\boldsymbol{w}_j \leftarrow \boldsymbol{w}_j - h_{j,j}\boldsymbol{v}_j$
  $h_{j+1,j} \leftarrow \|\boldsymbol{w}_j\|_2$
  if $h_{j+1,j} = 0$ **then**
    Stop.
  $\boldsymbol{v}_{j+1} \leftarrow \frac{1}{h_{j+1,j}}\boldsymbol{w}_j$

From the above considerations, it is clear that Algorithm 2.2 computes an orthonormal basis of $\mathcal{K}_m(A, \boldsymbol{b})$ if A is Hermitian and that it is mathematically equivalent to Arnoldi's method (however, in practice one observes a severe loss of orthogonality of the basis vectors after some iterations, such that in some applications, reorthogonalization strategies have to be applied, see, e.g., [79, 105, 122]).

By Algorithm 2.1 (or Algorithm 2.2 for Hermitian A), we can compute an orthonormal basis of $\mathcal{K}_m(A, \boldsymbol{b})$. The next question we have to answer is, given such a basis V_m, how to find an approximation

$$f(A)\boldsymbol{b} \approx \boldsymbol{f}_m \in \mathcal{K}_m(A, \boldsymbol{b})$$

by imposing some suitable condition on $\boldsymbol{f}_m$. To answer this question, consider the following. The main motivation for using Krylov subspace methods is given by Definition 2.2. In view of this definition, the idea of any Krylov subspace method

can be summarized as approximating the polynomial p from Definition 2.2 (which may be of degree up to $n-1$) by a polynomial of smaller degree $m-1$. We can thus rephrase the above question as how to choose a polynomial $p_{m-1} \in \Pi_{m-1}$, such that $p_{m-1}(A)\boldsymbol{b} \approx p(A)\boldsymbol{b}$. A straightforward approach, considering the fact that p interpolates f at $\mathrm{spec}(A)$, is to choose p_{m-1} as a polynomial which interpolates f at m suitably chosen points. One such choice are the eigenvalues of H_m, the so-called *Ritz values* corresponding to $\mathcal{K}_m(A, \boldsymbol{b})$. The following, classical result relates the Ritz values to eigenvalues of A, thus giving a first motivation for why one can consider them to be sensible interpolation points.

Proposition 2.24. *Let H_m be the upper Hessenberg matrix from the Arnoldi decomposition* (2.23) *corresponding to $\mathcal{K}_m(A, \boldsymbol{b})$ and let* $\mathrm{spec}(H_m) = \{\theta_1, \ldots, \theta_m\}$. *Then*

$$\theta_i \in \mathcal{W}(A) \text{ for } i = 1, \ldots, m,$$

where

$$\mathcal{W}(A) := \left\{ \frac{\boldsymbol{v}^H A \boldsymbol{v}}{\boldsymbol{v}^H \boldsymbol{v}} : \boldsymbol{v} \neq \boldsymbol{0} \right\}$$

denotes the field of values of A. If, in addition, $\mathcal{K}_m(A, \boldsymbol{b})$ is A-invariant, i.e., $A\mathcal{K}_m(A, \boldsymbol{b}) \subseteq \mathcal{K}_m(A, \boldsymbol{b})$, then

$$\theta_i \in \mathrm{spec}(A) \text{ for } i = 1, \ldots, m.$$

Proof. The first part of the assertion follows directly from the relation $H_m = V_m^H A V_m$ and the fact that V_m has orthonormal columns. The second part of the statement follows, e.g., directly from [129, Chapter 4, Theorem 4.1]. □

Proposition 2.24 guarantees that the Ritz values corresponding to $\mathcal{K}_m(A, \boldsymbol{b})$ are always related to some kind of spectral information of A as they lie in its field of values (which reduces to the spectral interval $[\lambda_{\min}, \lambda_{\max}]$ in the Hermitian case), and that they even become exact eigenvalues of A once the Krylov subspace reaches its maximum possible dimension. Of course, $\mathcal{K}_m(A, \boldsymbol{b})$ will in general not become A-invariant in practical computations, where one uses only small values of m, but the result at least shows that there is a relation between Ritz values and eigenvalues of A. In case that A is Hermitian, one can show further results on the behavior of the Ritz values (which can be more or less arbitrary in the general, non-Hermitian case) *before* $\mathcal{K}_m(A, \boldsymbol{b})$ becomes A-invariant, e.g., that "outliers" at the left or right end of the spectrum are well approximated first, cf., e.g., [101,134].

In addition to the reasoning stated above, choosing p_{m-1} as the polynomial which interpolates f at the Ritz values corresponding to $\mathcal{K}_m(A, \boldsymbol{b})$ has the additional advantage that $p_{m-1}(A)\boldsymbol{b}$ is readily available without needing to explicitly compute p_{m-1} (the numerical computation of high-degree interpolating polynomials can become highly unstable [131]). The precise result is stated in the following lemma, first proven in [47] and [114].

Lemma 2.25. *Let $A \in \mathbb{C}^{n\times n}$ and let $\boldsymbol{b} \in \mathbb{C}^n$. Let V_m, H_m fulfill the relation* (2.23) *and let*

$$\boldsymbol{f}_m = V_m f(V_m^H A V_m) V_m^H \boldsymbol{b} = \|\boldsymbol{b}\|_2 V_m f(H_m)\hat{\boldsymbol{e}}_1. \tag{2.25}$$

Then

$$\boldsymbol{f}_m = \widetilde{p}_{m-1}(A)\boldsymbol{b},$$

where $\widetilde{p}_{m-1} \in \Pi_{m-1}$ is the unique polynomial interpolating f at the eigenvalues of H_m in the Hermite sense, provided that f is defined on $\operatorname{spec}(H_m)$.

Proof. See, e.g., [85, Theorem 13.5]. □

The approximation defined by (2.25) is commonly referred to as *Arnoldi (or Lanczos) approximation* to $f(A)\boldsymbol{b}$ and is the standard choice for an approximation from the Krylov subspace $\mathcal{K}_m(A, \boldsymbol{b})$. Another possible motivation for using (2.25), without even considering the interpolating polynomial characterization, is that (2.25) is a projection of the original problem (2.21) onto the smaller space $\mathcal{K}_m(A, \boldsymbol{b})$. Of course, for (2.25) to be well-defined, $f(H_m)$ must exist, i.e., f must be defined on $\operatorname{spec}(H_m)$. For this, it is *not* sufficient that $f(A)$ is defined, as the following example illustrates.

Example 2.26. Consider the symmetric indefinite matrix

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

the vector $\boldsymbol{b} = [1, 1]^H$ and the function $f(z) = z^{-1}$. As $\operatorname{spec}(A) = \{-1, 1\}$, the matrix function $f(A) = A^{-1}$ is well-defined and we have $f(A)\boldsymbol{b} = A^{-1}\boldsymbol{b} = [1, -1]^H$. However, one step of the Lanczos method computes $\boldsymbol{v}_1 = [1/\sqrt{2}, 1/\sqrt{2}]^H$ and $\boldsymbol{w}_1 = A\boldsymbol{v}_1 = [1/\sqrt{2}, -1/\sqrt{2}]^H$, which is already orthogonal to $\boldsymbol{v}_1$, so that $h_{1,1} = \boldsymbol{v}_1^H \boldsymbol{w}_1 = 0$ and thus $H_1 = 0$. Therefore, $\boldsymbol{f}_1 = \|\boldsymbol{b}\|_2 V_1 f(H_1)\hat{\boldsymbol{e}}_1$ is not defined.

Example 2.26 motivates, amongst other reasons we will come across in later parts of this thesis, that it may under some circumstances be reasonable to extract other approximations than the Arnoldi approximation (2.25) from a given Krylov subspace. The following result from [48] is a generalization of Lemma 2.25 which shows that the polynomial interpolation characterization also holds when H_m in (2.25) is replaced by a suitable rank-one modification.

Lemma 2.27. *Let $A \in \mathbb{C}^{n\times n}$ and let $\boldsymbol{b} \in \mathbb{C}^n$. Let V_m, H_m fulfill the relation* (2.23)*, let $\boldsymbol{z} \in \mathbb{C}^n$ and let*

$$\hat{\boldsymbol{f}}_m = \|\boldsymbol{b}\|_2 V_m f(H_m + \boldsymbol{z}\hat{\boldsymbol{e}}_m^H)\hat{\boldsymbol{e}}_1. \tag{2.26}$$

Then

$$\hat{\boldsymbol{f}}_m = \widehat{p}_{m-1}(A)\boldsymbol{b},$$

where $\widehat{p}_{m-1} \in \Pi_{m-1}$ is the unique polynomial interpolating f at the eigenvalues of $H_m + \boldsymbol{z}\hat{\boldsymbol{e}}_m^H$ in the Hermite sense, provided that f is defined on $\operatorname{spec}(H_m + \boldsymbol{z}\hat{\boldsymbol{e}}_m^H)$.

Proof. See [48, Lemma 3 and Corollary 4]. □

Before we proceed, we give some further comments on the advantages and disadvantages of the Arnoldi approximation (and the related approximations (2.26)). An important advantageous feature of Arnoldi's method for matrix functions is that (at least in exact arithmetic), finite termination is guaranteed as long as all approximations are defined. By Lemma 2.22, the method breaks down after m steps if and only if $\mathcal{K}_m(A, \boldsymbol{b})$ is invariant under A. This in turn means that $f(A)\boldsymbol{b} = p(A)\boldsymbol{b}$ is already contained in $\mathcal{K}_m(A, \boldsymbol{b})$ and the projection (2.25) will yield the exact value of $f(A)\boldsymbol{b}$ (therefore, such a breakdown is sometimes also referred to as a *lucky breakdown*). However, using Arnoldi's method for matrix functions also has several disadvantages. As already illustrated by Example 2.26, the Arnoldi approximations need not exist even when $f(A)\boldsymbol{b}$ is defined. Other disadvantages are mainly of practical, computational nature. For evaluating (2.25), one needs to store the whole Arnoldi basis V_m. As the Arnoldi vectors will in general be full vectors, this means storing a dense $n \times m$ matrix. As A is often very large and sparse in practical applications, n will frequently be large. In this case, the number m of steps that can be performed is often limited by the available memory and may not be large enough to compute an approximation of the desired accuracy. In addition, even if the available memory does not limit the number of steps that can be performed, the evaluation of $f(H_m)\hat{\boldsymbol{e}}_1$, the action of function of a matrix of size $m \times m$ on a vector, becomes increasingly expensive with growing number of iterations. If the number of iterations necessary to reach a sufficiently accurate approximation lies in the order of magnitude of n, evaluating $f(H_m)\hat{\boldsymbol{e}}_1$ may be about as difficult as evaluating $f(A)\boldsymbol{b}$ itself, which can make the method infeasible for some problems. There are different approaches for overcoming these difficulties. On the one hand, *restarting techniques* are proposed, in which a certain (small) number m of steps is performed, $\boldsymbol{f}_m$ is computed by (2.25) and then, in a new Arnoldi iteration, one tries to approximate the remaining *error* $f(A)\boldsymbol{b} - \boldsymbol{f}_m$. This technique is more often studied and better understood in the context of linear systems, see, e.g., [56, 115, 116, 123] and we will at this point not go into detail concerning this topic. Chapter 4 is devoted

to restarting techniques, containing a review of existing approaches from the literature and new developments and extensions of these approaches. The other established approach for overcoming the disadvantages of the Arnoldi approximation is using other subspaces than Krylov subspaces $\mathcal{K}_m(A, \boldsymbol{b})$ which (hopefully) have better approximation properties, in the sense that a smaller dimension m is needed to reach an accurate enough approximation. Popular choices for these richer subspaces are *rational Krylov subspaces* and, as a special case of the former, *extended Krylov subspaces*. We discuss extended Krylov subspace methods and their properties in Chapter 7, for further details and the treatment of general rational Krylov subspaces, we refer to, e.g., [38, 80, 81, 94–96, 99] and the references therein.

2.4 The special case $f(z) = z^{-1}$

Krylov subspace methods are frequently used for the solution of linear systems, i.e., the special case of (2.21) with $f(z) = z^{-1}$. As we will exploit the relation between the approximation of Stieltjes matrix functions by the Arnoldi approximation (2.25) and the solution of linear systems at several points throughout this thesis, we will briefly cover some of the basic terminology and results arising in this setting in the following. We do not in any way strive for completeness, especially as there is a broad variety of Krylov subspace methods for linear systems like, e.g., BiCGStab [128, 138] or QMR [52], to name just two, which do not have a direct connection to the Arnoldi approximation (2.25). They are therefore not of relevance for the developments of this thesis, although some of them are widely used in practical applications.

The method arising when the Arnoldi approximation (2.25) is applied to the linear system

$$A\boldsymbol{x} = \boldsymbol{b} \Leftrightarrow \boldsymbol{x} = A^{-1}\boldsymbol{b},$$

i.e, the computation of the approximation

$$\boldsymbol{x}_m = \|\boldsymbol{b}\|_2 V_m H_m^{-1} \hat{\boldsymbol{e}}_1, \tag{2.27}$$

where V_m, H_m are the matrices resulting from Arnoldi's method for A and $\boldsymbol{b}$, is known as the *full orthogonalization method (FOM)* [113, 115] for linear systems. Note that when solving linear systems by a Krylov subspace method, it is common practice to provide the method with an *initial guess* $\boldsymbol{x}_0$. In this case, one only needs to approximate the remaining *error* $\boldsymbol{x}^* - \boldsymbol{x}_0$ of the initial guess. The following well-known result (which we state as a proposition despite its simple nature, as it will be used extensively throughout this thesis) gives an easy way to do so.

Proposition 2.28. *Let $A \in \mathbb{C}^{n\times n}$, let $\boldsymbol{b} \in \mathbb{C}^n$ and let $\boldsymbol{x}^*$ be the solution of the linear system $A\boldsymbol{x} = \boldsymbol{b}$. Further, let $\boldsymbol{x}_0 \in \mathbb{C}^n$ and define the residual $\boldsymbol{r}_0 = \boldsymbol{b} - A\boldsymbol{x}_0$. Then the error $\boldsymbol{e}_0 = \boldsymbol{x}^* - \boldsymbol{x}_0$ satisfies the* residual equation

$$A\boldsymbol{e}_0 = \boldsymbol{r}_0. \tag{2.28}$$

Proof. A direct computation yields $A(\boldsymbol{x}^* - \boldsymbol{x}_0) = A\boldsymbol{x}^* - A\boldsymbol{x}_0 = \boldsymbol{b} - A\boldsymbol{x}_0 = \boldsymbol{r}_0$. □

According to Proposition 2.28, one can compute the *residual* $\boldsymbol{r}_0 = \boldsymbol{b} - A\boldsymbol{x}_0$ and then find an Arnoldi approximation for $A^{-1}\boldsymbol{r}_0$, the solution of the residual equation (2.28), i.e., one generates iterates in the affine Krylov subspace

$$\boldsymbol{x}_0 + \mathcal{K}_m(A, \boldsymbol{r}_0).$$

The following result gives an explicit expression for the residual generated by applying m steps of FOM to the linear system $A\boldsymbol{x} = \boldsymbol{b}$.

Proposition 2.29. *Let $A \in \mathbb{C}^{n\times n}, \boldsymbol{b}, \boldsymbol{x}_0 \in \mathbb{C}^n$ and let $\boldsymbol{x}_m$ be the approximation from m steps of FOM (with initial guess $\boldsymbol{x}_0$) applied to the linear system $A\boldsymbol{x} = \boldsymbol{b}$. Then the residual $\boldsymbol{r}_m = \boldsymbol{b} - A\boldsymbol{x}_m$ satisfies*

$$\boldsymbol{r}_m = -h_{m+1,m}\hat{\boldsymbol{e}}_m^H \boldsymbol{y}_m \boldsymbol{v}_{m+1}, \tag{2.29}$$

where $\boldsymbol{y}_m = \|\boldsymbol{r}_0\|_2 H_m^{-1}\hat{\boldsymbol{e}}_1$, with $H_m, h_{m+1,m}$ and $\boldsymbol{v}_{m+1}$ from the Arnoldi decomposition (2.23)*. Thus, its Euclidean norm is given by*

$$\|\boldsymbol{r}_m\|_2 = h_{m+1,m}|\hat{\boldsymbol{e}}_m^H \boldsymbol{y}_m|. \tag{2.30}$$

Proof. See, e.g., [115, Proposition 6.7]. □

By recalling the definition of $\boldsymbol{y}_m$ in (2.30), we see that the Euclidean norm of the FOM residual can be found by computing the bottom left entry of the inverse of H_m, a relation which we will (implicitly and explicitly) exploit in later chapters. An important implication of Proposition 2.28, besides allowing to provide Krylov subspace methods for linear systems with an initial guess, is the possibility to *restart* them easily. After some number m of steps of FOM (or any other Krylov subspace method for $A\boldsymbol{x} = \boldsymbol{b}$), one computes the residual $\boldsymbol{r}_m = \boldsymbol{b} - A\boldsymbol{x}_m$ and can then approximately solve the residual equation $A\boldsymbol{e}_m = \boldsymbol{r}_m$ by m further steps of the same method, obtaining an approximation $\widetilde{\boldsymbol{e}}_m$ for the error $\boldsymbol{e}_m = \boldsymbol{x}^* - \boldsymbol{x}_m$. By an additive correction $\boldsymbol{x}_m^{(2)} = \boldsymbol{x}_m + \widetilde{\boldsymbol{e}}_m$ one then (hopefully) obtains a better approximation for $\boldsymbol{x}^*$. This procedure can then again be applied to the new residual equation corresponding to $\boldsymbol{x}_m^{(2)}$ and so on, yielding after k restart cycles

$$\boldsymbol{x}_m^{(k+1)} = \boldsymbol{x}_m^{(k)} + \widetilde{\boldsymbol{e}}_m^{(k)} \text{ with } \widetilde{\boldsymbol{e}}_m^{(k)} = \|\boldsymbol{r}_m^{(k-1)}\|_2 V_m^{(k)} (H_m^{(k)})^{-1}\hat{\boldsymbol{e}}_1, \tag{2.31}$$

where $\boldsymbol{r}_m^{(k-1)} = \boldsymbol{b} - A\boldsymbol{x}_m^{(k-1)}$ is the residual of the iterate from the $(k-1)$st cycle. This way, all quantities computed in the previous cycles of the method (in particular the matrices $V_m^{(i)}$ and $H_m^{(i)}$ for $i = 1, \ldots, k-1$) can be discarded, thus avoiding the growing storage requirements and computational cost which is associated with the unrestarted FOM approximation (2.27). We give a sketch of the resulting method (without going into detail on possible stopping criteria) in Algorithm 2.3, it is discussed in detail in [113, 115]. Another method for which restarting is frequently used in practice is GMRES [116].

Algorithm 2.3: Restarted full orthogonalization method

Given: A, $\boldsymbol{b}$, m, $\boldsymbol{x}_0$
$\boldsymbol{r}_0 \leftarrow \boldsymbol{b} - A\boldsymbol{x}_0$
$\beta \leftarrow \|\boldsymbol{r}_0\|_2$
$\boldsymbol{v}_1 \leftarrow \frac{1}{\beta}\boldsymbol{r}_0$
`tol_reached` $\leftarrow 0$
while `tol_reached` $= 0$ **do**
  Compute V_m, H_m by Algorithm 2.1 applied to $A, \boldsymbol{r}_0$.
  $\boldsymbol{y}_m \leftarrow \beta H_m^{-1}\hat{\boldsymbol{e}}_1$
  $\boldsymbol{x}_m \leftarrow \boldsymbol{x}_0 + V_m\boldsymbol{y}_m$
  if *target accuracy reached* **then**
    `tol_reached` $\leftarrow 1$
  $\boldsymbol{x}_0 \leftarrow \boldsymbol{x}_m$
  $\boldsymbol{r}_0 \leftarrow -h_{m+1,m}\hat{\boldsymbol{e}}_m^H\boldsymbol{y}_m\boldsymbol{v}_{m+1}$
  $\beta \leftarrow \|\boldsymbol{r}_0\|_2$

However, restarting may slow down or even destroy convergence of a Krylov subspace method. The convergence behavior of restarted Krylov subspace methods is until now not fully understood, for a discussion of this topic we refer to, e.g., [39, 40, 97, 137] and also to Section 5.6 of this thesis. We demonstrate by an example (which we also presented in [57]) that restarted FOM may exhibit a cyclic behavior and may fail to converge even for the maximum restart length $m = n-1$ (the restart length $m = n$ corresponds to FOM without restarting, as termination after n steps is guaranteed by Lemma 2.22, at least in exact arithmetic).

Example 2.30. Consider the linear system $A\boldsymbol{x} = \boldsymbol{b}$ with the matrix

$$A = \begin{bmatrix} 1 & 0 & \cdots & 0 & 1 \\ 1 & 1 & 0 & \cdots & 0 \\ 0 & 1 & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & 1 & 1 \end{bmatrix} \in \mathbb{R}^{n\times n}$$

for odd n and the vector $\boldsymbol{b} = \hat{\boldsymbol{e}}_1$. The exact solution of this linear system is given by

$$\boldsymbol{x}(i) = \begin{cases} \frac{1}{2} & \text{if } i \text{ is odd,} \\ -\frac{1}{2} & \text{if } i \text{ is even.} \end{cases}$$

If restarted FOM with restart length $m = n - 1$ and $\boldsymbol{x}_0 = \boldsymbol{0}$ is applied to the linear system $A\boldsymbol{x} = \boldsymbol{b}$, the first Arnoldi basis is $V_m^{(1)} = [\hat{\boldsymbol{e}}_1, \hat{\boldsymbol{e}}_2, \ldots, \hat{\boldsymbol{e}}_{n-1}]$ and the upper Hessenberg matrix $H_m^{(1)}$ is given by

$$H_m^{(1)} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 1 & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 1 & 1 \end{bmatrix} \in \mathbb{R}^{(n-1)\times(n-1)}. \tag{2.32}$$

Obviously, $\operatorname{spec}(H_m^{(1)}) = \{1\}$ so that $H_m^{(1)}$ is nonsingular and the Arnoldi approximation $\boldsymbol{x}_m^{(1)} = V_m^{(1)}(H_m^{(1)})^{-1}\hat{\boldsymbol{e}}_1$ is defined. One directly checks that the corresponding residual $\boldsymbol{r}_m^{(1)} = \boldsymbol{b} - A\boldsymbol{x}_m^{(1)}$ satisfies $\boldsymbol{r}_m^{(1)} = \hat{\boldsymbol{e}}_n$. The second restart cycle computes the Arnoldi basis $V_m^{(2)} = [\hat{\boldsymbol{e}}_n, \hat{\boldsymbol{e}}_1, \ldots, \hat{\boldsymbol{e}}_{n-2}]$, the same Hessenberg matrix $H_m^{(2)} = H_m^{(1)}$ and the residual $\boldsymbol{r}_m^{(2)} = \hat{\boldsymbol{e}}_{n-1}$. Continuing in this manner, one sees that throughout all restart cycles, the Hessenberg matrices are identical to the one from (2.32) and that in the kth cycle ($k \leq n$), the Arnoldi basis consists of all canonical unit vectors except $\hat{\boldsymbol{e}}_{n+1-k}$, and $\boldsymbol{r}_m^{(k)} = \hat{\boldsymbol{e}}_{n+1-k}$. Thus, after n restart cycles, $\boldsymbol{r}_m^{(n)} = \hat{\boldsymbol{e}}_1$, so that from there on every sequence of n cycles is identical to the sequence of the first n cycles and no convergence is obtained. Similar cyclic behavior can also be observed for any other restart length $m < n$, so that the method in fact stagnates for all restart lengths.

If A is Hermitian positive definite (i.e., the Lanczos process may be used to compute the orthonormal basis V_m), the short recurrence for the basis vectors $\boldsymbol{v}_j$ translates into a short recurrence for the iterates $\boldsymbol{x}_j$ from (2.27). For a detailed derivation of this short recurrence, we refer to [115, Chapter 6.7]. The resulting method, given as Algorithm 2.4, is known as the *conjugate gradient method (CG)*, first introduced in [84], and is widely used for solving Hermitian positive definite linear systems in practice.

In addition to the computational advantages of the conjugate gradient method over FOM, the convergence behavior is also understood much better. Classical results on the convergence of the conjugate gradient method bound the *energy norm* of the error.

Algorithm 2.4: Conjugate gradient method

Given: A, $\boldsymbol{b}$, m, $\boldsymbol{x}_0$

$\boldsymbol{r}_0 \leftarrow \boldsymbol{b} - A\boldsymbol{x}_0$
$\boldsymbol{p}_0 \leftarrow \boldsymbol{r}_0$
for $j = 0, 1, \ldots, m$ **do**
  $\alpha_j \leftarrow (\boldsymbol{r}_j^H \boldsymbol{r}_j)/(\boldsymbol{p}_j^H A \boldsymbol{p}_j)$
  $\boldsymbol{x}_{j+1} \leftarrow \boldsymbol{x}_j + \alpha_j \boldsymbol{p}_j$
  $\boldsymbol{r}_{j+1} \leftarrow \boldsymbol{r}_j - \alpha_j A \boldsymbol{p}_j$
  $\beta_j \leftarrow (\boldsymbol{r}_{j+1}^H \boldsymbol{r}_{j+1})/(\boldsymbol{r}_j^H \boldsymbol{r}_j)$
  $\boldsymbol{p}_{j+1} \leftarrow \boldsymbol{r}_{j+1} + \beta_j \boldsymbol{p}_j$

Definition 2.31. Let $A \in \mathbb{C}^{n\times n}$ be Hermitian positive definite. Then the *energy norm* of a vector $\boldsymbol{v} \in \mathbb{C}^n$ with respect to A is defined as

$$\|\boldsymbol{v}\|_A = \sqrt{(\boldsymbol{v}, A\boldsymbol{v})}.$$

The fact that $\|\cdot\|_A$ is indeed a norm follows easily from the well-known property that the bilinear form $(x, y)_A = (x, Ay)$ is an inner product for Hermitian positive definite A.

The following classical result is derived by exploiting the approximation properties of Chebyshev polynomials. We state it here, as it will later be useful to investigate the convergence behavior of the restarted Arnoldi method for Stieltjes functions of Hermitian positive definite matrices.

Theorem 2.32. *Let $A \in \mathbb{C}^{n\times n}$ be Hermitian positive definite and let $\boldsymbol{x}_0, \boldsymbol{b} \in \mathbb{C}^n$. Further, let $\boldsymbol{x}^*$ denote the solution of the linear system $A\boldsymbol{x} = \boldsymbol{b}$ and let $\boldsymbol{x}_m$ be the mth iterate of the CG method with initial guess $\boldsymbol{x}_0$. Let $\kappa = \frac{\lambda_{\max}}{\lambda_{\min}}$, where $\lambda_{\min}$ and $\lambda_{\max}$ are the smallest and largest eigenvalue of A, respectively, denote the condition number of A and define*

$$c = \frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1} \quad \text{and} \quad \alpha_m = \frac{1}{\cosh(m \ln c)}$$

(where we set $\alpha_m = 0$ if $\kappa = 1$). Then the error in the CG method satisfies

$$\|\boldsymbol{x}^* - \boldsymbol{x}_m\|_A \leq \alpha_m \|\boldsymbol{x}^* - \boldsymbol{x}_0\|_A.$$

Proof. The result follows from [77, Theorem 3.1.1] by using $\cosh(m \ln c) = (c^m + c^{-m})/2$. □

Another important Krylov subspace method, which is typically the method of choice for solving large, sparse, non-Hermitian linear systems in practical applications, is GMRES [116]. GMRES differs from FOM (or CG in the Hermitian case) in the way the approximation is extracted from the affine Krylov subspace $\boldsymbol{x}_0 + \mathcal{K}_m(A, \boldsymbol{r}_0)$. The GMRES iterate $\boldsymbol{x}_m^{\mathrm{G}}$ is chosen such that the residual $\boldsymbol{r}_m^{\mathrm{G}} = \boldsymbol{b} - A\boldsymbol{x}_m^{\mathrm{G}}$ is minimal among all possible approximations from $\boldsymbol{x}_0 + \mathcal{K}_m(A, \boldsymbol{r}_0)$. Defining the extended Hessenberg matrix

$$\overline{H}_m = \begin{bmatrix} H_m \\ h_{m+1,m}\hat{\boldsymbol{e}}_m^H \end{bmatrix} \in \mathbb{C}^{(m+1)\times m},$$

every approximation of the form $\boldsymbol{x}_m = \boldsymbol{x}_0 + V_m\boldsymbol{y}_m$ fulfills

$$\boldsymbol{b} - A\boldsymbol{x}_m = \boldsymbol{r}_0 - AV_m\boldsymbol{y}_m = V_m(\|\boldsymbol{r}_0\|_2\hat{\boldsymbol{e}}_1 - \overline{H}_m\boldsymbol{y}_m)$$

so that

$$\|\boldsymbol{b} - A\boldsymbol{x}_m\|_2 = \left\|\|\boldsymbol{r}_0\|_2\hat{\boldsymbol{e}}_1 - \overline{H}_m\boldsymbol{y}_m\right\|_2.$$

This shows that the mth GMRES iterate, i.e., the vector which minimizes the residual norm among all approximations from $\boldsymbol{x}_0 + \mathcal{K}_m(A, \boldsymbol{r}_0)$, can be computed as

$$\boldsymbol{x}_m^{\mathrm{G}} = \boldsymbol{x}_0 + V_m\boldsymbol{y}_m^{\mathrm{G}}, \tag{2.33}$$

where $\boldsymbol{y}_m^{\mathrm{G}}$ solves the linear least squares problem

$$\left\|\|\boldsymbol{r}_0\|_2\hat{\boldsymbol{e}}_1 - \overline{H}_m\boldsymbol{y}\right\|_2 \to \min. \tag{2.34}$$

Interestingly, one can show that the GMRES approximation (2.33) also has a connection to polynomial interpolation, albeit in different interpolation nodes, the so-called *harmonic Ritz values*.

Definition 2.33. The *harmonic Ritz values* of $A \in \mathbb{C}^{n\times n}$ with respect to a subspace $\mathcal{U} \subseteq \mathbb{C}^n$ are those numbers $\vartheta \in \mathbb{C}$ for which there exists $\boldsymbol{x} \in \mathcal{U},\ \boldsymbol{x} \neq \boldsymbol{0}$ such that

$$A\boldsymbol{x} - \vartheta\boldsymbol{x} \perp A\mathcal{U}.$$

Although Definition 2.33 allows to define harmonic Ritz values corresponding to an arbitrary subspace $\mathcal{U}$, we will in the following restrict ourselves to the case $\mathcal{U} = \mathcal{K}_m(A, \boldsymbol{r}_0)$, as these are the harmonic Ritz values relevant in the context of GMRES.

Lemma 2.34. *Let $A \in \mathbb{C}^{n\times n}$, let $\boldsymbol{b} \in \mathbb{C}^n$ and let $\boldsymbol{x}_m^{\mathrm{G}}$ be the GMRES approximation (with initial guess $\boldsymbol{x}_0$) defined by* (2.33) *and* (2.34)*. Then*

$$\boldsymbol{x}_m^{\mathrm{G}} = \boldsymbol{x}_0 + \widehat{p}_{m-1}(A)\boldsymbol{r}_0,$$

where $\widehat{p}_{m-1}$ is the unique polynomial of degree at most $m-1$ which interpolates $f(z) = z^{-1}$ in the harmonic Ritz values of A with respect to $\mathcal{K}_m(A, \boldsymbol{r}_0)$.

Proof. See, e.g., [76, Theorem 5.1] and [111, Section 5]. □

Lemma 2.34 allows us to derive another characterization of the GMRES approximation, based on the result of Lemma 2.27. To do so, we need the following auxiliary result.

Proposition 2.35. *Consider the Arnoldi decomposition* (2.23)*. The harmonic Ritz values of A with respect to $\mathcal{K}_m(A, \boldsymbol{r}_0)$ are the eigenvalues of the matrix*

$$\widetilde{H}_m = H_m + \left(h_{m+1,m}H_m^{-1}\hat{\boldsymbol{e}}_m\right)\hat{\boldsymbol{e}}_m^H, \tag{2.35}$$

provided that H_m is nonsingular.

Proof. See [111, Section 7.1]. □

Lemma 2.27 and 2.34 together with Proposition 2.35 now allow us to conclude that the GMRES approximation can also be characterized as

$$\boldsymbol{x}_m^{\mathrm{G}} = \boldsymbol{x}_0 + \|\boldsymbol{r}_0\|_2 V_m \left(H_m + \left(h_{m+1,m}H_m^{-1}\hat{\boldsymbol{e}}_m\right)\hat{\boldsymbol{e}}_m^H\right)^{-1}\hat{\boldsymbol{e}}_1. \tag{2.36}$$

It is not advisable to use this representation for practical computations due to possible numerical instabilities, and in addition due to the fact that (2.36) is not defined when H_m is singular, while the computation of $\boldsymbol{x}_m^{\mathrm{G}}$ via (2.33) and (2.34) is always possible. Nonetheless, the relation (2.36) will later allow us to derive a method for the approximation of Stieltjes matrix functions $f(A)$ times a vector $\boldsymbol{b}$ which reduces to GMRES in the case $f(z) = z^{-1}$ and has some favorable theoretical properties; cf. Section 5.4 and 5.5.

Next, we give a result from [42] on the reduction of the residual norm in the GMRES method for the class of positive real matrices.

Theorem 2.36. *Let $A \in \mathbb{C}^{n\times n}$ be positive real, i.e., $\Re(\boldsymbol{v}^H A\boldsymbol{v}) > 0$ for all $\boldsymbol{v} \in \mathbb{C}^n, \boldsymbol{v} \neq \boldsymbol{0}$ and let $\boldsymbol{b} \in \mathbb{C}^n$. Define the quantities*

$$\delta := \min\left\{\left|\frac{\boldsymbol{v}^H A\boldsymbol{v}}{\boldsymbol{v}^H\boldsymbol{v}}\right| : \boldsymbol{v} \in \mathbb{C}^n, \boldsymbol{v} \neq 0\right\}, \tag{2.37}$$

$$\delta' := \min\left\{\left|\frac{\boldsymbol{v}^H A^{-1}\boldsymbol{v}}{\boldsymbol{v}^H\boldsymbol{v}}\right| : \boldsymbol{v} \in \mathbb{C}^n, \boldsymbol{v} \neq 0\right\}. \tag{2.38}$$

Then the residual $\boldsymbol{r}_m^{\mathrm{G}}$ corresponding to the GMRES iterate $\boldsymbol{x}_m^{\mathrm{G}}$ defined by (2.33) *and* (2.34) *with initial guess $\boldsymbol{x}_0$ satisfies*

$$\|\boldsymbol{r}_m^{\mathrm{G}}\|_2 \leq \left(1 - \delta\delta'\right)^{m/2} \|\boldsymbol{r}_0\|_2. \tag{2.39}$$

Proof. See, e.g., [42, Corollary 6.2]. □

Note that the quantities δ and δ' from (2.37) and (2.38) are positive if A is positive real (as in this case, A^{-1} is positive real as well, see [91, Chapter 1]) and satisfy $\delta\delta' \leq 1$, see, e.g., [42, Section 6]. In particular, one can directly conclude from (2.39) that the *restarted GMRES iteration* for $A\boldsymbol{x} = \boldsymbol{b}$ always converges to the solution $\boldsymbol{x}^*$ if A is positive real.

Corollary 2.37. *Let the assumptions of Theorem 2.36 hold, let $(\boldsymbol{x}_m^{\mathrm{G}})^{(k)}$ denote the iterate obtained by k cycles of restarted GMRES with restart length m and initial guess $\boldsymbol{x}_0$ and let $(\boldsymbol{r}_m^{\mathrm{G}})^{(k)}$ be the corresponding residual. Then*

$$\|(\boldsymbol{r}_m^{\mathrm{G}})^{(k)}\|_2 \leq \left(1 - \delta\delta'\right)^{km/2} \|\boldsymbol{r}_0\|_2. \tag{2.40}$$

In particular, the restarted GMRES method converges to the exact solution $\boldsymbol{x}^*$ of $A\boldsymbol{x} = \boldsymbol{b}$, because the right-hand side of (2.40) *goes to zero for $k \to \infty$.*

Proof. Equation (2.40) directly follows from Theorem 2.36 by noting that he kth cycle of restarted GMRES can be interpreted as performing m steps of GMRES with initial guess $(\boldsymbol{x}_m^{\mathrm{G}})^{(k-1)}$. As $0 < \delta\delta' \leq 1$, we have $|1 - \delta\delta'| \leq 1$, so that the right-hand side of (2.40) goes to zero for $k \to \infty$. □

2.4.1 Krylov subspace methods for shifted linear systems

Another important aspect central to many results of this thesis is the behavior of Krylov subspace methods for shifted linear systems of the form

$$(A + tI)\boldsymbol{x}(t) = \boldsymbol{b}, \tag{2.41}$$

i.e., families of systems with the same right-hand side and with the system matrices differing only by multiples of the identity matrix. By (2.20), these systems have a strong relation to Stieltjes matrix functions. The following result concerning Krylov subspaces for these systems holds.

Proposition 2.38. *Let $A \in \mathbb{C}^{n\times n}$, let $\boldsymbol{b} \in \mathbb{C}^n$ and let $t \in \mathbb{C}$. Then*

(i) $\mathcal{K}_m(A, \boldsymbol{b}) = \mathcal{K}_m(A + tI, \boldsymbol{b})$ for all $m > 0$,

(ii) Algorithm 2.1 applied to $A + tI$ *and* $\boldsymbol{b}$ *computes the Arnoldi decomposition*

$$(A + tI)V_m = V_m(H_m + tI) + h_{m+1,m}\boldsymbol{v}_{m+1}\hat{\boldsymbol{e}}_m^H,$$

where V_m *and* H_m *are the matrices from the Arnoldi decomposition* (2.23) *for* A *and* $\boldsymbol{b}$*,*

(iii) the m*th FOM approximation* $\boldsymbol{x}_m(t)$ *for the linear system* $(A + tI)\boldsymbol{x}(t) = \boldsymbol{b}$ *with initial guess* $\boldsymbol{x}_0(t) = \boldsymbol{0}$ *is given by*

$$\boldsymbol{x}_m(t) = \|\boldsymbol{b}\|_2 V_m(H_m + tI)^{-1}\hat{\boldsymbol{e}}_1.$$

Proof. Assertion (i) directly follows by investigating the structure of powers of the shifted matrix $A+tI$. Part (ii) can be concluded by inspecting the operations in Arnoldi's method, Algorithm 2.1. Part (iii) then follows directly from (ii). □

The assertions of Proposition 2.38 have been observed several times and for different Krylov subspace methods, see, e.g., [54, 62, 123]. These observations are typically used to implement methods which are capable of solving several shifted linear systems at once, while only needing to compute a single approximation subspace. This corresponds to only performing only a single matrix-vector multiplication per iteration, independent of the number of shifted systems to be solved, cf., e.g., [54, 62, 123].

A topic to which special care has to be devoted when dealing with the simultaneous solution of shifted linear systems is *restarting*. In the first cycle of a Krylov subspace method for a family of systems of the form (2.41), the same Krylov subspace can be constructed for all shifted systems (at least if all methods are started with initial guess $\boldsymbol{x}_0(t) = \boldsymbol{0}$) due to the fact that all systems have the same right-hand side $\boldsymbol{b}$. This need not be the case after restarting the method, as then one attempts to approximately solve the shifted residual equations

$$(A + tI)\boldsymbol{e}_m(t) = \boldsymbol{r}_m(t),$$

where $\boldsymbol{r}_m(t) = \boldsymbol{b} - (A + tI)\boldsymbol{x}_m(t)$, to compute an approximation for the error $\boldsymbol{e}_m(t) := \boldsymbol{x}^*(t) - \boldsymbol{x}_m(t)$ of the current iterate. Of course, it suffices that the right-hand sides of the two systems be collinear (instead of equal) for Proposition 2.38 to hold. Therefore, it is again possible to use the same Krylov subspace for all shifted systems if all residuals $\boldsymbol{r}_m(t)$ are collinear. For the full orthogonalization method, this is indeed the case, as by Proposition 2.29, the mth FOM residual is collinear to the $(m + 1)$st Arnoldi basis vector. Due to the shift invariance of the Arnoldi method stated by Proposition 2.38(ii), this basis vector $\boldsymbol{v}_{m+1}$ is the same for all systems, independent of the shift t. Thus, all shifted FOM residuals are collinear to $\boldsymbol{v}_{m+1}$ and one can compute the restarted shifted FOM approximations

of the second cycle (or later cycles) from one Krylov subspace for all systems again; see [123] for an in-depth treatment of the resulting method.

The GMRES method, however, does in general not produce collinear residuals, so that one cannot just compute GMRES approximations for systems of the form (2.41) with different shifts t and then use only one approximation space again after restarting. In [56], a variant of restarted GMRES for shifted linear systems has been proposed which overcomes these issues as follows: Only the approximate solution for one of the systems (the so-called *seed system*) is computed as a standard GMRES iterate as defined by (2.33) and (2.34), and then the approximations for the other systems are computed in a way that enforces collinearity to the residual of the seed system. This way, the iterates for the other systems are no true GMRES iterates (and therefore, e.g., do not have the residual norm minimization property) but restarting with one Krylov subspace for all systems is again possible. We do not go into detail concerning this topic here, as the precise construction is not of importance in our context. Theoretical results concerning the "shifted" GMRES method from [56] will be addressed in Chapter 5, where they are transferred to a method for approximating Stieltjes matrix functions.

2.5 Numerical quadrature

When dealing with integrals of functions for which no antiderivative is known or available in a numerical computation, one instead has to approximate the integral numerically by what is typically called a *quadrature rule*. As integral representations of functions for which no closed form is available will appear at many places throughout this thesis, due to the integral representation of the error in Arnoldi's method to be introduced in Chapter 3, quadrature rules are of vital importance for making the methods and results presented in this thesis feasible for numerical computations. We therefore briefly review the basic concepts of (mostly Gauss) quadrature, following the presentation in [33] and [74]. Other references for a basic treatment of quadrature rules include [46, 100, 131] (albeit sometimes in a slightly different setting than here).

We consider only quadrature rules on finite intervals in the following. For infinite intervals of integration, one either applies a suitable variable transformation which maps the interval of integration to a finite one or uses quadrature rules specifically designed for infinite intervals; see, e.g., [33, Chapter 3] or [68]. As we will pursue the first approach and comment on the choices of variable transformations in depth in Section 4.3, where they are actually applied in our setting, we do not go into detail concerning infinite intervals of integration here.

Gauss quadrature rules are typically introduced with respect to a nonnegative weight function $w(t) \geq 0$ in the literature. We will use a slightly more general approach in the following definition of quadrature rules, in the sense that we introduce Gauss rules for Riemann–Stieltjes integrals corresponding to a monotonically increasing function μ. By Lemma 2.8, if μ is differentiable, this can also be interpreted as an integral corresponding to the nonnegative weight function $w = \mu'$. When dealing with quadrature rules other than Gauss rules in the following, we will tacitly assume that $\mu(t) = t$, i.e, we are in the case of Riemann integrals.

Definition 2.39. Let $[a, b]$ be a finite interval, let $\mu : [a, b] \longrightarrow \mathbb{R}$ be monotonically increasing and let $g : [a, b] \longrightarrow \mathbb{C}$ be any function such that the integral

$$\int_a^b g(t)\,\mathrm{d}\mu(t) \tag{2.42}$$

exists and has a finite value. An *ℓ-point quadrature rule* for μ on $[a, b]$ is then given by a set of *weights* $\omega_i \in \mathbb{C}, i = 1, \ldots, \ell$ and a set of *nodes* $t_i \in [a, b], i = 1, \ldots, \ell$ such that

$$\sum_{i=1}^{\ell} \omega_i g(t_i),$$

approximates (2.42).

Two of the simplest quadrature rules are the *compound midpoint rule* and the *compound trapezoidal rule*. The compound midpoint rule is defined as

$$M_\ell(g) = \frac{b-a}{\ell} \sum_{i=1}^{\ell} g\left(a + \left(i - \frac{1}{2}\right)\frac{b-a}{\ell}\right), \tag{2.43}$$

i.e., all weights are equal to $\frac{b-a}{\ell}$ and the quadrature nodes are chosen as the centers of a subdivision of $[a, b]$ into ℓ intervals of equal length. The compound trapezoidal rule is given by

$$T_\ell(g) = \frac{b-a}{2\ell}\big(g(a) + g(b)\big) + \frac{b-a}{\ell} \sum_{i=1}^{\ell-1} g\left(a + i\frac{b-a}{\ell}\right), \tag{2.44}$$

i.e., the nodes are chosen equispaced in $[a, b]$, this time including the endpoints, and the weights are chosen to be $\frac{b-a}{2\ell}$ at the endpoints and $\frac{b-a}{\ell}$ for the interior nodes. An illustration of these simple rules (also called *primitive rules*) is given in Figure 2.1.

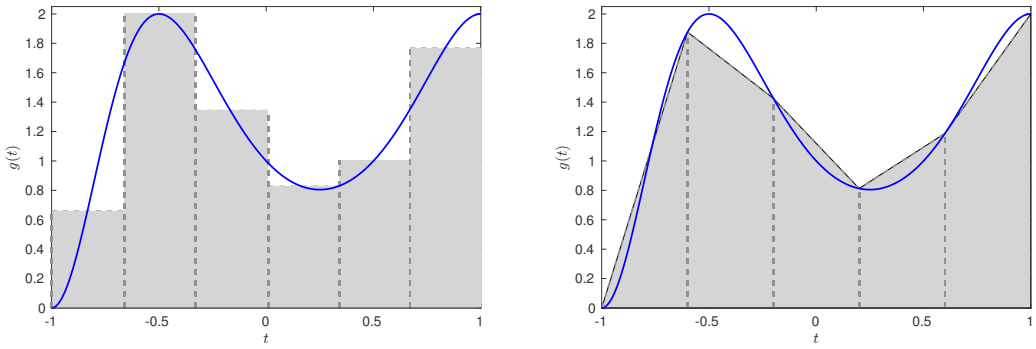


Figure 2.1: Illustration of the compound midpoint (left) and trapezoidal (right) rule for integrating the function $g(t) = 1 + \sin((t^2 - \frac{1}{2}t)\pi)$ on the interval $[-1, 1]$ with $\ell = 6$ quadrature points.

The following theorem gives an expression for the error of the primitive quadrature rules (2.43) and (2.44).

Theorem 2.40. *Let $g : [a, b] \to \mathbb{C}$ be two times continuously differentiable on (a, b) and let $M_\ell(g)$ and $T_\ell(g)$ be given by (2.43) and (2.44), respectively. Then*

$$\int_a^b g(t)\,\mathrm{d}t - M_\ell(g) = \frac{(b-a)^3}{24\ell^2} g''(\xi) \text{ for some } \xi \in (a, b)$$

and

$$\int_a^b g(t)\,\mathrm{d}t - T_\ell(g) = -\frac{(b-a)^3}{12\ell^2} g''(\xi) \text{ for some } \xi \in (a, b).$$

Proof. See [33, Section 4.3]. □

Theorem 2.40 gives some interesting insight into the properties of the midpoint and trapezoidal rule: Both rules are exact for linear functions (which is also clear from geometric intuition; cf. Figure 2.1), and if the second derivative of g is nonnegative on (a, b), we have

$$M_\ell(g) \leq \int_a^b g(t)\,\mathrm{d}t \leq T_\ell(g). \tag{2.45}$$

Properties of the type (2.45), also known as *bracketing properties*, will prove useful later in this thesis when using quadrature rules to compute bounds for the norm of the error in Arnoldi's method.

The primitive quadrature rules introduced so far are in general only exact for linear functions, i.e., polynomials of degree up to 1. As a quadrature rule, according to Definition 2.39, is defined by 2ℓ parameters, ℓ weights and ℓ nodes, it is a natural consideration to try to construct quadrature rules which are exact for polynomials of degree up to $2\ell - 1$. Quadrature rules of this type are called *Gauss quadrature rules*.

Definition 2.41. Let $[a, b]$ be a finite interval and let $\mu : [a, b] \longrightarrow \mathbb{R}$ be monotonically increasing. An ℓ-point quadrature rule for μ on $[a, b]$ defined by $(\omega_i, t_i), i = 1, \ldots, \ell$ is called *Gauss quadrature rule* if it satisfies

$$\int_a^b p_{2\ell-1}(t)\,\mathrm{d}\mu(t) = \sum_{i=1}^{\ell} \omega_i p_{2\ell-1}(t_i) \text{ for all } p_{2\ell-1} \in \Pi_{2\ell-1}. \tag{2.46}$$

The existence of rules which satisfy (2.46) is closely related to *orthonormal polynomials*. Note that a function μ which is nonnegative and monotonically increasing on $[a, b]$ induces a (not necessarily positive definite) inner product on polynomials $p, q \in \Pi_k$ via

$$(p, q) = \int_a^b p(t)\overline{q}(t)\,\mathrm{d}\mu(t). \tag{2.47}$$

Sequences of polynomials which are orthonormal with respect to this inner product play an important role in numerical quadrature.

Definition 2.42. A sequence of polynomials $p_i, i = 0, 1, \ldots$ is called orthonormal with respect to the inner product (2.47) if $\deg p_i = i$ and

$$(p_i, p_j) = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{otherwise.} \end{cases}$$

One can show that for a nonnegative, monotonically increasing function μ there exists a unique sequence of orthonormal polynomials if the inner product (2.47) is positive definite, see, e.g., [69] (more precisely, if (2.47) fulfills $(p, q) > 0$ for all $0 \neq p, q \in \Pi_k$ but not for $p, q \in \Pi_\ell$ for any $\ell > k$, then there exists a finite sequence of $k + 1$ orthonormal polynomials). We will in the following for ease of presentation restrict ourselves to the case that (2.47) is positive definite, which is

always fulfilled in the applications in this thesis. The following, central theorem states that orthonormal polynomials always obey a three-term recurrence.

Theorem 2.43. *Let μ be a nonnegative, monotonically increasing function on $[a, b]$ such that the inner product (2.47) is positive definite. Define*

$$p_{-1} \equiv 0 \text{ and } p_0 \equiv 1/\left(\int_a^b \mathrm{d}\mu(t)\right)^{1/2}.$$

Then there exist coefficients $a_i, b_i, i = 0, 1, 2, \ldots$ such that the sequence $p_i, i = 0, 1, \ldots$ defined by

$$b_i p_i(t) = (t - a_{i-1})p_{i-1}(t) - b_{i-1}p_{i-2}(t), \qquad i = 1, 2, \ldots \tag{2.48}$$

is the unique sequence of orthonormal polynomials corresponding to the inner product (2.47).

Proof. See [69, Theorem 1.27 and 1.29]. □

We just note at this point that the three-term recurrence (2.48) satisfied by orthonormal polynomials is very similar to the three-term recurrence relation satisfied by the Krylov basis vectors generated by the Lanczos process, Algorithm 2.2, for Hermitian A. Together with the intimate relation between the Lanczos/Arnoldi method and polynomials in A, this allows to also relate the Lanczos process to orthonormal polynomials and thus Gauss quadrature. This topic will be covered in depth in Section 6.1.

The relation between orthonormal polynomials and Gauss quadrature rules is given by the following result.

Theorem 2.44. *Let μ be a nonnegative, monotonically increasing function on $[a, b]$ such that the inner product (2.47) is positive definite and let $p_i, i = 0, 1, \ldots,$ be the sequence of orthonormal polynomials corresponding to μ and $[a, b]$. Then*

(i) The roots $t_i, i = 1, \ldots, \ell$ of $p_\ell(t)$ are real, simple and lie in (a, b) for all $\ell \geq 1$.

(ii) The quadrature rule defined by the nodes $t_i, i = 1, \ldots, \ell$ and the weights

$$\omega_i = -\frac{k_{\ell+1}}{k_\ell}\frac{1}{p_{\ell+1}(t_i)p'_\ell(t_i)}, \qquad i = 1, \ldots, \ell \tag{2.49}$$

where $k_\ell, k_{\ell+1}$ are the leading coefficients of p_ℓ and $p_{\ell+1}$, respectively, is an ℓ-point Gauss quadrature rule (which we denote by $\mathrm{G}_\ell^\mu(\cdot)$ in the following). In addition, the weights ω_i from (2.49) are all positive.

Proof. Part (i) is, e.g., shown in [74, Theorem 2.14]. For part (ii), see, e.g., [33, Section 2.7] □

Theorem 2.44 states that there always exists a Gauss quadrature rule for a given interval $[a, b]$ and nonnegative, monotonically increasing function μ if the inner product (2.47) is positive definite. In [100] is is shown that there exist no ℓ-point quadrature rules which integrate all polynomials of degree up to 2ℓ exactly, such that Gauss quadrature rules are optimal in this sense.

The representation of the nodes and weights of a Gauss rule given by Theorem 2.44 is often not well-suited for numerical computations. In [75], based on [143], a different approach was proposed, which we briefly mention here as it will be of importance in Chapter 6 and 7 of this thesis. Assuming we know the coefficients a_i, b_i in (2.48), we can rewrite the three-term recurrence in matrix notation as

$$t\boldsymbol{p}(t) = T_\ell \boldsymbol{p}(t) + b_\ell p_\ell(t)\hat{\boldsymbol{e}}_\ell, \tag{2.50}$$

where T_ℓ is the tridiagonal matrix defined by

$$T_\ell = \begin{bmatrix} a_1 & b_1 & & & \\ b_1 & a_2 & b_2 & & \\ & \ddots & \ddots & \ddots & \\ & & b_{\ell-1} & a_{\ell-1} & b_{\ell-1} \\ & & & b_\ell & a_\ell \end{bmatrix} \tag{2.51}$$

and

$$\boldsymbol{p}(t) = \left[p_0(t), p_1(t), \ldots, p_{\ell-1}(t)\right]^T.$$

The nodes $t_i, i = 1, \ldots, \ell$ of the ℓ-point Gauss quadrature rule are the zeros of $p_\ell(t)$, so that inserting one of these nodes as t in (2.50) yields

$$t_i \boldsymbol{p}(t_i) = T_\ell \boldsymbol{p}(t_i),$$

showing that t_i is an eigenvalue of T_ℓ with corresponding eigenvector $\boldsymbol{p}(t_i)$. Thus, the nodes of Gauss quadrature rules can be computed by solving a symmetric tridiagonal eigenvalue problem. One can further show that the corresponding weights are the squares of the first entries of the eigenvectors $\boldsymbol{p}(t_i)$ by using the so-called Christoffel–Darboux identity [74, Theorem 2.11], as proven, e.g., in [143]. This approach gives a more stable way of computing the nodes and weights of Gauss quadrature rules. We refer the reader to [74, 75] for an in-depth treatment of this approach and its implications for the structure and properties of Gauss rules.

Next, we give two examples of practically relevant Gauss quadrature rules, which we will also use in the computations presented in this thesis.

Example 2.45.

(i) The orthonormal polynomials corresponding to the function $\mu(t) = t$ on the interval $[-1, 1]$ are the *Legendre polynomials* given by the three-term recurrence

$$p_0 \equiv 1, \quad p_1(t) = t, \quad (i+1)p_{i+1}(t) = (2i+1)tp_i(t) - ip_{i-1}(t), i = 1, 2, \ldots.$$

The resulting quadrature rules are called *Gauss–Legendre* rules.

(ii) The orthonormal polynomials corresponding to the Jacobi weight function $w(t) = (1-t)^\alpha(1+t)^\beta$ with $\alpha, \beta > -1$ on the interval $[-1, 1]$ are the *Jacobi polynomials*

$$p_i(t) = \frac{1}{2^i} \sum_{j=0}^{i} \binom{i+\alpha}{j} \binom{i+\beta}{i-j} (t-1)^{i-j}(t+1)^j, \qquad i = 0, 1, \ldots.$$

The associated quadrature rules are called *Gauss–Jacobi* rules. In the special case $\alpha = \beta = -\frac{1}{2}$, the weight function simplifies to $w(t) = (1-t^2)^{-1/2}$ and the resulting quadrature rule is called *Gauss–Chebyshev* rule.

Note that we introduced Gauss–Jacobi quadrature with respect to a weight function in Example 2.45(ii) as this is the classical approach from the literature. Keep in mind that the integral with respect to the weight function can also be interpreted as a Riemann–Stieltjes integral.

For a sufficiently smooth function g, the error of a Gauss quadrature rule can again, similarly to the primitive quadrature rules, be expressed in terms of a derivative of g evaluated at some point in (a, b).

Theorem 2.46. *Let μ be a nonnegative, monotonically increasing function on $[a, b]$, let g be 2ℓ times continuously differentiable on (a, b) and let $\mathrm{G}_\ell^\mu(g)$ denote the corresponding ℓ-point Gauss quadrature rule. Then there exists $\xi \in (a, b)$ such that*

$$\int_a^b g(t)\,\mathrm{d}\mu(t) - \mathrm{G}_\ell^\mu(g) = c_\mathrm{G} \frac{g^{(2\ell)}(\xi)}{(2\ell)!},$$

with the constant

$$c_\mathrm{G} = \int_a^b p_\ell(t)^2\,\mathrm{d}\mu(t),$$

where p_ℓ is the ℓth orthonormal polynomial corresponding to μ and $[a, b]$.

Proof. See, e.g., [69, Corollary to Theorem 1.48]. □

By Theorem 2.46, if $g^{(2\ell)} \geq 0$ on (a, b), an ℓ-point Gauss quadrature rule will always give a lower bound for the exact value of the approximated integral. For certain applications (cf. Chapter 6 and 7) it is of value to have a bracketing property for Gauss rules, similar to the property (2.45) for the midpoint and trapezoidal rule. This is possible by considering quadrature rules in which one quadrature node is fixed a priori.

In principle, it is possible to fix any number of quadrature nodes at arbitrary points in $[a, b]$, but we will for brevity only consider the case in which one quadrature node is fixed to be at a, the left endpoint of the integration interval, as this is all that is needed in the remainder of this thesis. Rules of this kind with $\ell + 1$ nodes, which are exact for polynomials of degree up to 2ℓ are called Gauss–Radau rules. The nodes and weights of a Gauss–Radau quadrature rule can also be computed by solving a symmetric tridiagonal matrix eigenvalue problem as for ordinary Gauss rules, where the matrix $T_{\ell+1}$ is modified in such a way that one of its eigenvalues is prescribed to be a. Without going into detail, we denote the resulting quadrature rule by $\mathrm{GR}_{\ell+1}^{\mu}(\cdot)$. For its error, the following theorem holds.

Theorem 2.47. *Let μ be a nonnegative, monotonically increasing function on $[a, b]$, let g be $2\ell + 1$ times continuously differentiable on (a, b) and let $\mathrm{GR}_{\ell+1}^{\mu}(g)$ denote the corresponding $(\ell+1)$-point Gauss–Radau quadrature rule with one node fixed at a. Then there exists $\xi \in (a, b)$ such that*

$$\int_a^b g(t)\,\mathrm{d}\mu(t) - \mathrm{GR}_{\ell+1}^{\mu}(g) = c_{\mathrm{GR}}\frac{g^{(2\ell+1)}(\xi)}{(2\ell+1)!},$$

with the constant

$$c_{\mathrm{GR}} = \int_a^b p_\ell(t)^2(t-a)\,\mathrm{d}\mu(t),$$

where p_ℓ is the ℓth orthonormal polynomial corresponding to $\widehat{\mu}$ defined via

$$\mathrm{d}\widehat{\mu}(t) = (t-a)\,\mathrm{d}\mu(t)$$

on $[a, b]$.

Proof. See, e.g., [69, Theorem 3.3]. □

According to Theorem 2.46 and 2.47, we have the following version of the bracketing property for functions g which are $2\ell + 1$ times continuously differentiable and satisfy $g^{(2\ell)} \geq 0$ and $g^{(2\ell+1)} \leq 0$ on (a, b):

$$\mathrm{G}_{\ell}^{\mu}(g) \leq \int_a^b g(t)\,\mathrm{d}\mu(t) \leq \mathrm{GR}_{\ell+1}^{\mu}(g). \tag{2.52}$$

We end this section by pointing out that for the class of Stieltjes functions (or more generally completely monotonic functions), the following important result directly follows from (2.52).

Corollary 2.48. *Let μ be a nonnegative, monotonically increasing function on $[a, b]$ and let g be a completely monotonic function, cf. Definition 2.16. Then for all $\ell, \widetilde{\ell} \geq 1$ we have*

$$\mathrm{G}_{\ell}^{\mu}(g) \leq \int_a^b g(t)\,\mathrm{d}\mu(t) \leq \mathrm{GR}_{\widetilde{\ell}}^{\mu}(g), \tag{2.53}$$

i.e., the Gauss and Gauss–Radau rules for $\int_a^b g(t)\,\mathrm{d}\mu(t)$ always yield lower and upper bounds for the exact value of the integral, respectively, independently of the number of quadrature nodes.

2.6 Model problems

In this section we introduce various model problems from practical applications which will be used to demonstrate and gauge the various features of the different methods and results presented in this thesis in a realistic setting. All test cases will be considered at various places throughout this thesis. Note that some will be encountered more frequently than others, as not all methods and error bounds are applicable to all of the model problems (depending, e.g., on Hermiticity or definiteness of the resulting matrices).

2.6.1 Three-dimensional heat equation

The first model problem we consider is a standard test case which is frequently used for testing Krylov subspace methods for the matrix exponential; see, e.g., [3, 43,67]. We consider the initial boundary value problem with homogeneous Dirichlet boundary conditions (note that here and in the following, we use the nonstandard notation θ instead of t for the time parameter to avoid confusion with the integration variable in the integral representations of matrix functions or the shift in families of shifted linear systems)

$$\begin{aligned} \frac{\partial u}{\partial \theta} - \Delta u &= 0 \qquad \text{on } (0,1)^3 \times (0,T), \\ u(x,\theta) &= 0 \qquad \text{on } \partial(0,1)^3 \text{ for all } \theta \in [0,T], \\ u(x,0) &= u_0(x) \text{ for all } x \in (0,1)^3, \end{aligned} \tag{2.54}$$

where $u_0(x)$ is a function describing the initial conditions and Δ denotes the Laplace operator defined by

$$\Delta u = \frac{\partial^2 u}{\partial x_1^2} + \frac{\partial^2 u}{\partial x_2^2} + \frac{\partial^2 u}{\partial x_3^2}. \tag{2.55}$$

The system (2.54) describes the evolution of a heat distribution in the unit cube over a time interval $(0, T)$, starting from an initial distribution u_0 at time 0. Discretizing (2.55) by the standard seven-point finite difference stencil with $N+2$ equispaced grid points in each spatial direction, the system (2.54) reduces to a linear initial value ODE problem

$$\begin{aligned} \frac{\mathrm{d}\boldsymbol{u}(\theta)}{\mathrm{d}\theta} &= A\boldsymbol{u}(\theta), \text{ for } \theta \in (0, T) \\ \boldsymbol{u}(0) &= \boldsymbol{u}_0, \end{aligned} \tag{2.56}$$

where $A \in \mathbb{R}^{N^3 \times N^3}$ is Hermitian negative definite and $\boldsymbol{u}_0$ contains the values of the function $u_0(x)$ at the grid points. In our experiments, we choose $N = 50$, resulting in a matrix of dimension 125,000 which can be written as

$$A = A_{1\mathrm{D}} \otimes I \otimes I + I \otimes A_{1\mathrm{D}} \otimes I + I \otimes I \otimes A_{1\mathrm{D}},$$

where $A_{1\mathrm{D}}$ is the tridiagonal matrix

$$A_{1\mathrm{D}} = (N+1)^2 \begin{bmatrix} -2 & 1 & & \\ 1 & -2 & \ddots & \\ & \ddots & \ddots & 1 \\ & & 1 & -2 \end{bmatrix} \in \mathbb{R}^{N \times N}.$$

The solution of the ODE system (2.56) is then given by

$$\boldsymbol{u}(\theta) = e^{\theta A}\boldsymbol{u}_0, \tag{2.57}$$

i.e., the evaluation of the solution $\boldsymbol{u}$ at some point in time amounts to evaluating the action of a matrix exponential function on the vector of initial conditions. In our experiments, we approximate $\boldsymbol{u}(\theta)$ at $\theta = 0.1$ starting with a homogeneous initial heat distribution $\boldsymbol{u}_0 = \boldsymbol{1}$.

2.6.2 Three-dimensional convection diffusion equation

Our next model problem, taken from [43, 107], again deals with a PDE initial boundary value problem, which this time leads to a non-Hermitian matrix A. We

investigate the system

$$\begin{aligned} \frac{\partial u}{\partial \theta} - \Delta u + \tau_1 \frac{\partial u}{\partial x_1} + \tau_2 \frac{\partial u}{\partial x_2} &= 0 \qquad \text{on } (0,1)^3 \times (0,T), \\ u(x,\theta) &= 0 \qquad \text{on } \partial(0,1)^3 \text{ for all } \theta \in [0,T], \\ u(x,0) &= u_0(x) \text{ for all } x \in (0,1)^3. \end{aligned} \tag{2.58}$$

Again discretizing the Laplace operator by seven-point finite differences on a grid with $N+2$ points in each spatial direction and using centralized finite differences (on the same grid) for the first-order derivatives, we find, as in Section 2.6.1, a linear ODE system of the form (2.56), where this time the matrix $A \in \mathbb{R}^{N^3 \times N^3}$ is non-Hermitian and given by

$$A = A_{1\mathrm{D}} \otimes I \otimes I + I \otimes C_2 \otimes I + I \otimes I \otimes C_1 \tag{2.59}$$

with $N \times N$ matrices

$$C_i = (N+1)^2 \begin{bmatrix} -2 & 1-\nu_i & & \\ 1+\nu_i & -2 & \ddots & \\ & \ddots & \ddots & 1-\nu_i \\ & & 1+\nu_i & -2 \end{bmatrix}, i = 1, 2$$

where $\nu_i = \frac{\tau_i}{2(N+1)}$. Of course, the solution of the linear ODE system can again be represented by a matrix exponential applied to the vector of initial conditions as in (2.57) with the matrix A from (2.59). For our experiments we choose the parameters $N = 50$ and $\theta = 2 \cdot 10^{-3}$ and convection coefficients $\tau_1 = 4080, \tau_2 = 2040$, resulting in $\nu_1 = 40, \nu_2 = 20$, which leads to a highly non-Hermitian matrix A. The (discretized) initial conditions are again given by $\boldsymbol{u}_0 = \boldsymbol{1}$.

2.6.3 Three-dimensional wave equation

In this model problem, which is, e.g., considered in [38], we aim to approximate a function different from the matrix exponential. We consider solving the following initial boundary value problem for the three-dimensional wave equation on the unit cube and an infinite time interval

$$\begin{aligned} -\Delta u - \frac{\partial^2 u}{\partial \theta^2} &= 0 \qquad \text{on } (0,1)^3 \times \mathbb{R}^+ \\ u(x,\theta) &= 0 \qquad \text{on } \partial(0,1)^3 \text{ for all } \theta \in \mathbb{R}_0^+, \\ u(x,0) &= u_0(x) \text{ for all } x \in (0,1)^3. \end{aligned} \tag{2.60}$$

Discretizing the Laplace operator in the same way as in Section 2.6.1 and 2.6.2 leads to the system

$$\begin{aligned} \frac{\mathrm{d}^2\boldsymbol{u}(\theta)}{\mathrm{d}\theta^2} &= A\boldsymbol{u}(\theta), \text{ for } \theta \in \mathbb{R}^+ \\ \boldsymbol{u}(0) &= \boldsymbol{u}_0, \end{aligned}$$

where the matrix A is Hermitian positive definite in this case, as the sign of the Laplace operator and the sign of the time derivative are the same in (2.60), as opposed to (2.54) and (2.58). The solution of this system is given as

$$\boldsymbol{u}(\theta) = e^{-\theta\sqrt{A}}\boldsymbol{u}_0.$$

We rewrite this as

$$\boldsymbol{u}(\theta) = (Af(A) + I)\boldsymbol{u}_0, \tag{2.61}$$

with the function

$$f(z) = \frac{e^{-\theta\sqrt{z}} - 1}{z}. \tag{2.62}$$

The formulation (2.61) has the advantage that the function f from (2.62) has the integral representation

$$f(z) = -\int_0^\infty \frac{1}{z+t}\frac{\sin(\theta\sqrt{t})}{\pi t}\,\mathrm{d}t, \tag{2.63}$$

which makes many of the methods and results developed in the following applicable in this situation. We stress, however, that the function f from (2.62) is *not* a Stieltjes function as the corresponding generating function is not monotonically increasing. As in Section 2.6.1, we choose $N = 50, \theta = 0.1$ and initial conditions $\boldsymbol{u}_0 = \boldsymbol{1}$ in our computations.

2.6.4 Neuberger overlap operator in lattice QCD

The next problem we consider is from quantum chromodynamics (QCD), an area of Theoretical Physics which studies the strong interaction between quarks and gluons. In lattice QCD, this theory is simulated on a four-dimensional space-time lattice (where we, for ease of notation, denote the grid points by $x = (x_0, x_1, x_2, x_3)$, i.e., we do not distinguish space and time coordinates notationwise). For introducing the basics of this model problem, we follow the descriptions in [20, 117]. The most important relation for describing the interaction of quarks and gluons is the Dirac equation [35]

$$(\mathcal{D} + m)\psi(x) = \eta(x), \tag{2.64}$$

where m is a scalar parameter, ψ and η represent quark fields and $\mathcal{D}$ is the *Dirac operator* defined by

$$\mathcal{D} = \sum_{i=0}^{3} \gamma_i \otimes \left(\frac{\partial}{\partial x_i} + A_i \right) \tag{2.65}$$

where the matrices $A_i(x) \in \mathbb{C}^{3\times 3}$ are elements of the Lie algebra $\mathfrak{su}(3)$ of the special unitary group SU(3) and the matrices $\gamma_i \in \mathbb{C}^{4\times 4}, i = 0, \ldots, 3$ are generators of the Clifford algebra $C\ell_4(\mathbb{C})$. From the above description it is clear that the quark field $\psi(x)$ at a point x in space-time is a vector with 12 components (corresponding to three colors and four spins). For computer simulations, the Dirac equation (2.64) is discretized on an $N_t \times N_s^3$ grid (called the lattice from here on) with uniform lattice spacing a and N_t and N_s denoting the number of lattice points in the time dimension and each of the three spatial dimensions, respectively. We consider the Wilson discretization [144] with periodic boundary conditions in the following, in which the covariant derivatives in (2.65) are replaced by centralized covariant finite differences (and a stabilization term is added), resulting in the discretized operator

$$\begin{aligned} (D_W\phi)(x) = \quad & \frac{m_0+4}{a}\phi(x) - \frac{1}{2a}\sum_{i=0}^{3} \big((I_4 - \gamma_i) \otimes U_i(x)\big)\phi(x + a\hat{\boldsymbol{e}}_i) \\ - \quad & \frac{1}{2a}\sum_{i=0}^{3} \big((I_4 + \gamma_i) \otimes U_i^H(x - a\hat{\boldsymbol{e}}_i)\big)\phi(x - a\hat{\boldsymbol{e}}_i) \end{aligned} \tag{2.66}$$

where the mass parameter m_0 determines the quark mass, and the matrices $U_i(x) \in \mathbb{C}^{3\times 3}$ (the so-called gauge links) are elements of the Lie group SU(3). Consistent with the periodic boundary conditions, terms of the form $x - a\hat{\boldsymbol{e}}_i$ are to be understood on a torus, i.e., the boundaries of the lattice are "glued together".

A set of gauge links $U_i(x)$ for all grid points x is also referred to as a *configuration*.

The Wilson–Dirac operator (2.66) fulfills the so-called Γ_5-symmetry

$$(\Gamma_5 D_W)^H = \Gamma_5 D_W$$

with the Hermitian, unitary matrix

$$\Gamma_5 = I_{N_t N_s^3} \otimes \gamma_0\gamma_1\gamma_2\gamma_3 \otimes I_3; \tag{2.67}$$

see, e.g., [60]. For the simulation of some physical observables it is important that the discretized operator fulfills a (lattice variant of) the so-called chiral symmetry, which amounts to fulfilling the Ginsparg–Wilson relation [70]

$$\Gamma_5 D + D\Gamma_5 = aD\Gamma_5 D \tag{2.68}$$

with Γ_5 from (2.67). Unfortunately, the Wilson–Dirac operator does not fulfill this relation and is thus not suited for all simulations of interest. In [109], the Neuberger overlap operator

$$D_N = \rho I + \Gamma_5 \operatorname{sign}(\Gamma_5 D_W), \text{ where } \rho > 1, \tag{2.69}$$

which fulfills the relation (2.68), was introduced. In simulations involving the Neuberger overlap operator (2.69), one has to solve linear systems with D_N. As D_N is not explicitly available (it is not feasible to explicitly form $\operatorname{sign}(\Gamma_5 D_W)$ for realistic grid sizes) one typically uses Krylov subspace methods which only need to apply matrix vector products with D_N. Still, in each iteration of a Krylov subspace method, this amounts to approximating the action of the matrix sign function (of a Hermitian matrix) on the latest Krylov basis vector, i.e., $\operatorname{sign}(\Gamma_5 D_W)\boldsymbol{v}_i$.

The matrix sign function can be represented by the identity

$$\operatorname{sign}(A) = A(A^2)^{-1/2}, \tag{2.70}$$

involving the Stieltjes function $f(z) = z^{-1/2}$ (cf. Example 2.14), see, e.g., [31, 48], which is the representation typically used in computational practice. In our experiments we approximate the action of the matrix sign function involved in (2.69), using the formulation (2.70), on a vector for a discretization on an 8×8^3 lattice, thus yielding a matrix of dimension $12 \cdot 8^4 = 49{,}152$.

A further modification of the situation described above arises in the presence of a nonzero chemical potential ν. In this case, the links in time direction in the Wilson–Dirac operator change, and the discretization changes from (2.66) to

$$\begin{aligned}
(D_W^\nu \phi)(x) = \quad & \frac{m_0 + 4}{a}\phi(x) - \frac{1}{2a}\sum_{i=1}^{3}\big((I_4 - \gamma_i) \otimes U_i(x)\big)\phi(x + a\hat{\boldsymbol{e}}_i) \\
- \quad & \frac{1}{2a}\sum_{i=1}^{3}\big((I_4 + \gamma_i) \otimes U_i^H(x - a\hat{\boldsymbol{e}}_i)\big)\phi(x - a\hat{\boldsymbol{e}}_i) \\
- \quad & \frac{1}{2a}e^{\nu}\big((I_4 - \gamma_i) \otimes U_i(x)\big)\phi(x + a\hat{\boldsymbol{e}}_i) \\
- \quad & \frac{1}{2a}e^{-\nu}\big((I_4 + \gamma_i) \otimes U_i^H(x - a\hat{\boldsymbol{e}}_i)\big)\phi(x - a\hat{\boldsymbol{e}}_i),
\end{aligned} \tag{2.71}$$

which agrees with (2.66) for chemical potential $\nu = 0$. For $\nu \neq 0$, the operator D_W^ν from (2.71) is *not* Γ_5-symmetric, such that the solution of systems with the overlap operator (2.69) now involves approximating the matrix sign function (or inverse square root) of a non-Hermitian matrix. We also report experiments for this case, using the same 8×8^3 configuration as before, but adding a chemical potential $\nu = 1/20$, which is a physically reasonable value, given the temperature T (a quantity from statistical physics) used for generating the configuration.

2.6.5 Sampling from Gaussian Markov random fields

The last model problem we consider is taken from [93, 126] and arises from the statistical application of sampling from a *Gaussian Markov random field*. Given a set of n points $s_i \in \mathbb{R}^d, i = 1, \ldots, n$, one defines a Gaussian random variable $x_i, i = 1, \ldots, n$ at each point. The vector $\boldsymbol{x}$ of these random variables is called a Gaussian Markov random field. The so-called precision matrix $A \in \mathbb{R}^{n \times n}$ of the points s_i (with respect to two parameters δ, ϕ) is given by

$$a_{ij} = \begin{cases} 1 + \phi \sum_{k=1, k \neq i}^{n} \chi_{ij}^{\delta} & \text{if } i = j, \\ -\phi \chi_{ij}^{\delta} & \text{otherwise,} \end{cases} \tag{2.72}$$

where χ^{δ} is given by

$$\chi_{ij}^{\delta} = \begin{cases} 1 & \text{if } \|s_i - s_j\|_2 < \delta, \\ 0 & \text{otherwise.} \end{cases}$$

The matrix A from (2.72) is obviously Hermitian and diagonally dominant. By the Geršgorin disk theorem (see, e.g., [136]), all eigenvalues λ of A fulfill $\lambda \geq 1$, so that A is positive definite. In addition, as all row sums of A are 1, the vector $\boldsymbol{1}$ fulfills $A\boldsymbol{1} = \boldsymbol{1}$, showing that A must have an eigenvalue $\lambda = 1$ (as $\boldsymbol{1}$ is an eigenvector to this eigenvalue). A sample from the Gaussian Markov random field $\boldsymbol{x}$ can be obtained by computing $A^{-1/2}\boldsymbol{z}$, where $\boldsymbol{z}$ is a vector of independently and identically distributed standard normal random variables; see [92, 127]. For our experiments, we simulate $n = 50{,}000$ points in the unit square $(0, 1) \times (0, 1)$ and choose the parameters $\phi = 3, \delta = 0.01$, which results in a sparse, unstructured matrix $A \in \mathbb{R}^{50{,}000 \times 50{,}000}$ with $\operatorname{spec}(A) \subseteq [1, 109.6]$ and 830,626 nonzero entries. We are therefore, like for the Neuberger overlap operator at zero chemical potential, in the situation of approximating a Stieltjes function of a Hermitian positive definite matrix. The vector $\boldsymbol{z}$ is generated by the MATLAB function `randn`. For applications of Gaussian Markov random fields, see, e.g., [28, 112].

CHAPTER 3

AN INTEGRAL REPRESENTATION FOR THE ERROR IN ARNOLDI'S METHOD

In this chapter we consider representations for the error of the iterate $\boldsymbol{f}_m$ produced by m steps of Arnoldi's method for $f(A)\boldsymbol{b}$. These error representations form the basis for both the restarting approaches discussed in Chapter 4 and 5 as well as the computation of error bounds (mostly in unrestarted methods) which are investigated in Chapter 6 and 7. We begin by discussing previously known error representations from the literature in Section 3.1. In Section 3.2 we proceed by deriving new integral representations for the error for different classes of functions representable by contour integrals over resolvent functions. Important classes of functions to which our results apply are Stieltjes functions and holomorphic functions represented by the Cauchy integral formula.

3.1 Error representation via divided differences

In the special case $f(z) = z^{-1}$, i.e., when solving a linear system, a simple error representation is given by the residual equation (2.28). This equation can be rewritten as

$$\boldsymbol{e}_0 = A^{-1}\boldsymbol{r}_0 = f(A)\boldsymbol{r}_0,$$

showing that the error is representable as the action of the matrix function $f(A)$ on the vector $\boldsymbol{r}_0$. For general matrix functions, a similar result does unfortunately not hold. However, it is possible to represent the error of the restarted Arnoldi approximation as the action of a matrix function *different from* f based on divided differences (see, e.g., [34]), as the following result from [43] shows. It is

originally stated for *Arnoldi-like decompositions*, which are decompositions of the form (2.23) without the requirement that the columns of V_m are orthonormal. As we do not need this more general concept for the results of this thesis, we state the result in terms of standard Arnoldi decompositions.

Theorem 3.1. *Given $A \in \mathbb{C}^{n\times n}$, let $\boldsymbol{b} \in \mathbb{C}^n$, let V_m and H_m satisfy the Arnoldi relation (2.23) corresponding to A and $\boldsymbol{b}$ and let $w_m(z) = \prod_{i=1}^m (z-\theta_i)$ be the nodal polynomial associated with the Ritz values $\theta_1, \ldots, \theta_m$, i.e., the eigenvalues of H_m. Then the error of the Arnoldi approximation $\boldsymbol{f}_m$ from (2.25) is given by*

$$f(A)\boldsymbol{b} - \boldsymbol{f}_m = \|\boldsymbol{b}\|_2 \gamma_m [D_{w_m} f](A)\boldsymbol{v}_{m+1} =: e_m(A)\boldsymbol{v}_{m+1}, \tag{3.1}$$

where $[D_{w_m} f]$ denotes the mth divided difference of f with respect to the interpolation nodes $\theta_1, \ldots, \theta_m$, and $\gamma_m = \prod_{i=1}^m h_{i+1,i}$.

Proof. See [43, Theorem 2.6]. □

We note that an error representation based on mth order divided differences was independently found in [132].

From a theoretical point of view, Theorem 3.1 gives an answer to the question how the error after m steps of Arnoldi's method can be represented as the action of a matrix function on a vector again. However, it is not feasible for practical computations due to the well-known fact that the numerical evaluation of high-order divided differences is prone to instabilities, especially when interpolation nodes are close to each other, thereby causing subtractive cancellations and very small denominators in the divided difference table. This fact especially leads to problems when attempting to implement a restart approach based on this error function, as for Hermitian A it is known that the Ritz values of all restart cycles will asymptotically appear as a two-cyclic sequence [3], so that the interpolation nodes will form $2m$ clusters and the evaluation of the error function using (3.1) will necessarily become unstable.

Based on the results from [43], a different representation for the error in Arnoldi's method, which is also based on divided differences, was developed in [93] for Hermitian A.

Theorem 3.2. *Let the assumptions of Theorem 3.1 hold and let $A \in \mathbb{C}^{n\times n}$ be Hermitian. Let W_m be a unitary matrix whose columns are eigenvectors of H_m, and define $\alpha_i = \hat{\boldsymbol{e}}_m^H W_m \hat{\boldsymbol{e}}_i$ and $\beta_i = \hat{\boldsymbol{e}}_1^H W_m \hat{\boldsymbol{e}}_i, i = 1, \ldots, m$. Then*

$$f(A)\boldsymbol{b} - \boldsymbol{f}_m = \|\boldsymbol{b}\|_2 h_{m+1,m} g(A)\boldsymbol{v}_{m+1}$$

with

$$g(z) = \sum_{i=1}^m \alpha_i \beta_i [D_{\widetilde{w}_i} f](z), \text{ where } \widetilde{w}_i(z) = (z-\theta_i).$$

Proof. See [93, Theorem 2.1]. □

On first sight, one could expect the error representation given in Theorem 3.2 to be more stable in finite precision arithmetic than the one from Theorem 3.1 as it only involves first-order divided differences. However, as was stated in [93] and is also confirmed by our numerical experiments presented in Chapter 4, the representation is still unstable and therefore not usable in practice, especially in cases where the requirements on accuracy and reliability are high.

3.2 Integral representation of the error function

As explained in the previous section, the error representations considered in the literature so far are numerically infeasible (e.g., for implementing a restarted Arnoldi method) due to the need of evaluating divided differences. In this section, we derive integral representations for the error of the Arnoldi approximation for different classes of functions. Our results presented in this chapter have been published in [57–59]. We begin by investigating "Cauchy-type" integrals.

Due to the intimate relation between Arnoldi's method and polynomial interpolation, cf. Lemma 2.25 and Lemma 2.27, we first give an integral representation for interpolating polynomials of functions of Cauchy-type.

Lemma 3.3. *Let $\Omega \subset \mathbb{C}$ be a region and let $f : \Omega \to \mathbb{C}$ be analytic, with integral representation*

$$f(z) = \int_\Gamma \frac{g(t)}{t - z} \, \mathrm{d}t, \quad z \in \Omega, \tag{3.2}$$

with a path $\Gamma \subset \mathbb{C} \setminus \Omega$ and a function $g : \Gamma \to \mathbb{C}$. The Hermite interpolating polynomial p_{m-1} of f with interpolation nodes $\{\theta_1, \ldots, \theta_m\} \subset \Omega$ is given as

$$p_{m-1}(z) = \int_\Gamma \left(1 - \frac{w_m(z)}{w_m(t)}\right) \frac{g(t)}{t - z} \, \mathrm{d}t, \tag{3.3}$$

where $w_m(z) = \prod_{i=1}^m (z - \theta_i)$, provided that the integral in (3.3) exists.

Proof. Observe that for fixed t, the function $1 - w_m(z)/w_m(t)$ is a polynomial of degree m in z with a root at t. Therefore it contains a linear factor $t - z$, showing that $(1 - w_m(z)/w_m(t))/(t - z)$ is a polynomial of degree $m - 1$ in z, and so is the whole right-hand side of (3.3). By definition of w_m we have

$$p_{m-1}(\theta_i) = \int_\Gamma \left(1 - \frac{w_m(\theta_i)}{w_m(t)}\right) \frac{g(t)}{t - \theta_i} \, \mathrm{d}t = \int_\Gamma \frac{g(t)}{t - \theta_i} \, \mathrm{d}t = f(\theta_i)$$

for $i = 1, \ldots, m$, showing that the interpolation conditions for f are satisfied. In the case of coalescent interpolation nodes θ_i, we also demand that certain derivatives of f are interpolated by p_{m-1}. Assume that $\theta_i = \theta_j$ for $i < j$, which then amounts to the interpolation condition $p'_{m-1}(\theta_i) = f'(\theta_i)$. For $\varepsilon > 0$, define the sequence of interpolating polynomials $p^{\varepsilon}_{m-1}(z)$ corresponding to the interpolation nodes $\theta_1, \ldots, \theta_{j-1}, \theta_i + \varepsilon, \theta_{j+1}, \ldots, \theta_m$, which are pairwise distinct for ε sufficiently small (assuming for simplicity that no other interpolation nodes than θ_i and θ_j coincide). Due to the fact that interpolating polynomials depend analytically on the interpolation nodes and because f is analytic in Ω, we have that

$$\begin{aligned} f'(\theta_i) &= \lim_{\varepsilon \to 0} \frac{f(\theta_i + \varepsilon) - f(\theta_i)}{\varepsilon} \\ &= \lim_{\varepsilon \to 0} \frac{p^{\varepsilon}_{m-1}(\theta_i + \varepsilon) - p^{\varepsilon}_{m-1}(\theta_i)}{\varepsilon} \\ &= p'(\theta_i). \end{aligned}$$

For more than two coincident interpolation nodes and higher derivatives, the result follows analogously. □

Lemma 3.3 does not assert the existence of the integral on the right-hand side of (3.3). Since $f(z)$ is assumed to be representable by the integral (3.2), the integral in (3.3) exists if and only if the integral

$$\int_{\Gamma} \frac{1}{w_m(t)} \frac{g(t)}{t - z} \, \mathrm{d}t, \tag{3.4}$$

exists. At this point we just caution the reader to be aware of this fact and postpone a discussion of sufficient conditions guaranteeing the existence of (3.4) to the end of this section. In the following we derive an integral representation for the error of the Arnoldi approximation to $f(A)\boldsymbol{b}$ under the assumption that all necessary integrals exists.

Theorem 3.4. *Let $\Omega \subset \mathbb{C}$ be a region, let f have an integral representation as in Lemma 3.3 with $\Gamma \subset \mathbb{C} \setminus \Omega$, and let $A \in \mathbb{C}^{n \times n}$ with $\operatorname{spec}(A) \subset \Omega$ and $\boldsymbol{b} \in \mathbb{C}^n$ be given. Denote by $\boldsymbol{f}_m$ the mth Arnoldi approximation (2.25) to $f(A)\boldsymbol{b}$ with $\operatorname{spec}(H_m) = \{\theta_1, \ldots, \theta_m\} \subset \Omega$. Then, provided that the integral (3.4) with $w_m(t) = \prod_{i=1}^{m}(t - \theta_i)$ exists,*

$$f(A)\boldsymbol{b} - \boldsymbol{f}_m = \|\boldsymbol{b}\|_2 \gamma_m \int_{\Gamma} \frac{g(t)}{w_m(t)} (tI - A)^{-1} \boldsymbol{v}_{m+1} \, \mathrm{d}t =: e_m(A)\boldsymbol{v}_{m+1}, \tag{3.5}$$

where $\gamma_m = \prod_{i=1}^{m} h_{i+1,i}$.

Proof. Let p_{m-1} denote the interpolating polynomial of f with respect to the interpolation nodes $\theta_1, \ldots, \theta_m$. By subtracting p_{m-1} from f and using the representations (3.2) and (3.3) we have

$$f(z) - p_{m-1}(z) = \int_\Gamma \frac{w_m(z)}{w_m(t)} \frac{g(t)}{t-z} \, \mathrm{d}t. \tag{3.6}$$

Substituting A for z in (3.6), post-multiplying by $\boldsymbol{b}$, and noting that $p_{m-1}(A)\boldsymbol{b} = \boldsymbol{f}_m$ by Lemma 2.25 then leads to

$$f(A)\boldsymbol{b} - \boldsymbol{f}_m = \int_\Gamma \frac{g(t)}{w_m(t)} (tI - A)^{-1} w_m(A)\boldsymbol{b} \, \mathrm{d}t.$$

The assertion follows from the fact that $w_m(A)\boldsymbol{b} = \|\boldsymbol{b}\|_2 \gamma_m \boldsymbol{v}_{m+1}$, see [111, Corollary 2.1]. Note that in the result from [111] only symmetric matrices A are considered, but the result and its proof also apply to non-Hermitian A in exactly the same way. □

The most prominent examples of functions with a representation of the form (3.2) are holomorphic functions given by the Cauchy integral formula

$$f(z) = \frac{1}{2\pi i} \int_\Gamma \frac{f(t)}{t-z} \, \mathrm{d}t,$$

where Γ is a path that winds around z exactly once. In this case, $g(t) = \frac{1}{2\pi i} f(t)$ in (3.2). Our more general approach allows to also consider other classes of functions, like, e.g., Stieltjes functions generated by a differentiable function μ, where $g(t) = \mu'(t)$ (after performing a simple variable transformation $t \to -t$); cf. Lemma 2.8.

It is also possible to derive a result similar to the one of Theorem 3.4 for general Stieltjes functions corresponding to a possibly nondifferentiable measure. We omit a proof of this result, as it is almost identical to the one of Theorem 3.4.

Theorem 3.5. *Let $A \in \mathbb{C}^{n\times n}$, let $\boldsymbol{b} \in \mathbb{C}^n$ and let f be a Stieltjes function of the form* (3.15)*. Assume that* $\operatorname{spec}(A) \subset \mathbb{C} \setminus \mathbb{R}_0^-$ *and denote by $\boldsymbol{f}_m$ the mth Arnoldi approximation* (2.25) *to $f(A)\boldsymbol{b}$. Assume that* $\operatorname{spec}(H_m) = \{\theta_1, \ldots, \theta_m\}$ *satisfies* $\operatorname{spec}(H_m) \subset \mathbb{C} \setminus \mathbb{R}_0^-$ *and define*

$$e_m(z) = (-1)^{m+1} \|\boldsymbol{b}\|_2 \gamma_m \int_0^\infty \frac{1}{w_m(t)} \frac{1}{z+t} \, \mathrm{d}\mu(t), \quad z \in \mathbb{C} \setminus \mathbb{R}_0^-, \tag{3.7}$$

where $w_m(t) = \prod_{i=1}^m (t + \theta_i)$ and $\gamma_m = \prod_{i=1}^m h_{i+1,i}$. Then

$$f(A)\boldsymbol{b} - \boldsymbol{f}_m = e_m(A)\boldsymbol{v}_{m+1}, \tag{3.8}$$

where $\boldsymbol{v}_{m+1}$ is the $(m+1)$st Arnoldi vector.

Some functions of practical interest, like, e.g., $\widetilde{f}(z) = z^\alpha$ for $\alpha \in (0,1)$, including the square root as the most important special case, or $\widetilde{f}(z) = \log(1+z)$, do not have an integral representation (3.2) but can be written as $\widetilde{f}(z) = zf(z)$, where f is of the form (3.2). In this case, the result of Theorem 3.4 does not directly apply. One could possibly overcome this problem by using the fact that

$$\widetilde{f}(A)\boldsymbol{b} = Af(A)\boldsymbol{b} = f(A)\widetilde{\boldsymbol{b}}, \text{ where } \widetilde{\boldsymbol{b}} = A\boldsymbol{b}$$

and then apply Arnoldi's method to A and $\widetilde{\boldsymbol{b}}$. However, this approach has the disadvantage that $\|\widetilde{\boldsymbol{b}}\|_2$ may be significantly larger than $\|\boldsymbol{b}\|_2$ (by a factor of up to $\|A\|_2$) which may result in larger absolute errors of the Arnoldi approximations. Therefore, one should try to work with $\widetilde{f}$ directly. Fortunately, it is possible to modify the result from Theorem 3.4 to accommodate for such functions.

Corollary 3.6. *Let the assumptions of Theorem 3.4 hold and let $\widetilde{f}(z) = zf(z)$. Denote by $\widetilde{\boldsymbol{f}}_m$ the mth Arnoldi approximation* (2.25) *to $\widetilde{f}(A)\boldsymbol{b}$. Then*

$$\widetilde{f}(A)\boldsymbol{b} - \widetilde{\boldsymbol{f}}_m = \|\boldsymbol{b}\|_2\gamma_m A\int_\Gamma \frac{g(t)}{w_m(t)}(tI-A)^{-1}\boldsymbol{v}_{m+1}\,\mathrm{d}t - h_{m+1,m}\left(\hat{\boldsymbol{e}}_m^H f(H_m)\hat{\boldsymbol{e}}_1\right)\boldsymbol{v}_{m+1}, \tag{3.9}$$

provided that the integral in (3.9) *exists.*

Proof. By (2.25) we have

$$\widetilde{\boldsymbol{f}}_m = V_m\widetilde{f}(H_m)\hat{\boldsymbol{e}}_1 = V_m H_m f(H_m)\hat{\boldsymbol{e}}_1. \tag{3.10}$$

Inserting the Arnoldi decomposition (2.23) into 3.10 gives

$$\widetilde{\boldsymbol{f}}_m = AV_m f(H_m)\hat{\boldsymbol{e}}_1 - h_{m+1,m}\left(\hat{\boldsymbol{e}}_m^H f(H_m)\hat{\boldsymbol{e}}_1\right)\boldsymbol{v}_{m+1}. \tag{3.11}$$

By subtracting (3.11) from $\widetilde{f}(A)\boldsymbol{b}$ we arrive at

$$\widetilde{f}(A)\boldsymbol{b} - \widetilde{\boldsymbol{f}}_m = A\left(f(A)\boldsymbol{b} - V_m f(H_m)\hat{\boldsymbol{e}}_1\right) - h_{m+1,m}\left(\hat{\boldsymbol{e}}_m^H f(H_m)\hat{\boldsymbol{e}}_1\right)\boldsymbol{v}_{m+1}. \tag{3.12}$$

The assertion now follows by applying Theorem 3.4 to the first term on the right-hand side of (3.12). □

Corollary 3.6 can easily be generalized to functions of the form $\widetilde{f}(z) = z^\ell f(z)$ by repeated application of (2.23). We just state the result only for $zf(z)$ for the sake of notational simplicity and because it appears to be the most important case in practice. Ignoring for a moment the term

$$-h_{m+1,m}\left(\hat{\boldsymbol{e}}_m^H f(H_m)\hat{\boldsymbol{e}}_1\right)\boldsymbol{v}_{m+1} \tag{3.13}$$

in (3.9), we observe that the error function on the right-hand side of (3.9) is of a similar form as the original function $\widetilde{f}(z) = zf(z)$, in the sense that it is of the form $ze_m(z)$, where $e_m(z)$ denotes the error function for $f(z)$ from (3.5). The remaining term (3.13) in the error representation can be explicitly evaluated along with $\widetilde{\boldsymbol{f}}_m$ from (3.10) at almost no cost because all necessary quantities are readily available. Doing this corresponds to the "corrected" Arnoldi approximation introduced in [114] in the context of approximating so-called φ-functions.

Of course, the above discussion and the result of Corollary 3.6 also apply to general Stieltjes functions in light of Theorem 3.5, but we refrain from restating it for this case, as the necessary modifications are obvious.

In the remainder of this section, we will investigate sufficient conditions for guaranteeing the existence of the integrals appearing in Theorem 3.4 and 3.5. As a tool, we need a result from classical analysis, the Abel–Dirichlet test for improper integrals.

Theorem 3.7. *Let $h_1(t)$ be piecewise continuously differentiable on every interval $[t_0, 0] \subset \mathbb{R}_0^-$ and suppose $h_1(t) \to 0$ as $t \to -\infty$, while $h_1'(t)$ is absolutely integrable on $\mathbb{R}_0^-$. Moreover, let $h_2(t)$ be piecewise continuous on every interval $[t_0, 0] \subset \mathbb{R}_0^-$ and suppose*

$$|H_2(t)| \leq C \quad \text{for } t \in \mathbb{R}_0^-, \quad \text{where } H_2(t) = \int_t^0 h_2(\zeta)\,\mathrm{d}\zeta \tag{3.14}$$

with C independent of $t \in \mathbb{R}_0^-$. Then the integral $\int_{-\infty}^0 h_1(t)h_2(t)\,\mathrm{d}t$ exists and is finite.

Proof. See [121, Theorem 11.23a]. □

Using this result, we can prove the existence of the integral (3.4) for two important classes of functions.

Proposition 3.8. *Assume that f, g, Ω, and Γ in (3.2) satisfy one of the two following conditions:*

(i) f is holomorphic in a region $\Omega' \supset \Omega$, and $\Gamma \subset \Omega'$ is a closed contour winding around each $z \in \Omega$ exactly once, such that by the Cauchy integral formula

$$f(z) = \frac{1}{2\pi i} \int_\Gamma \frac{f(t)}{t - z}\,\mathrm{d}t,$$

i.e., $g(t) = \frac{1}{2\pi i} f(t)$ in (3.2).

(ii) $\Gamma = \mathbb{R}_0^-$, $\Omega = \mathbb{C} \setminus \mathbb{R}_0^-$ *and* f *is of the form*

$$f(z) = \int_{-\infty}^{0} \frac{g(t)}{t-z}\,\mathrm{d}t, \quad z \in \mathbb{C} \setminus \mathbb{R}_0^- \tag{3.15}$$

where $g(t)$ *is a function which is piecewise continuous on every interval* $[t_0, 0] \subset \mathbb{R}_0^-$.

Moreover, assume that $w_m(t) = \prod_{i=1}^m (t-\theta_i)$ *with* $\theta_i \in \Omega, i = 1, \ldots, m$. *Then the integral*

$$\int_\Gamma \frac{1}{w_m(t)} \frac{g(t)}{t-z}\,\mathrm{d}t$$

exists for all $z \in \Omega$.

Proof. Part (i) is trivial, since in this case the function $\frac{1}{w_m(t)}\frac{g(t)}{t-z}$ is continuous on the closed contour Γ for all $z \in \Omega$. We note in passing that in this case the integral representation (3.3) of the interpolating polynomial is a well-known classical result, cf., e.g., [140]. The proof for part (ii) is a bit more involved. We define the auxiliary functions

$$h_1(t) = \frac{1}{w_m(t)}, \quad h_2(t) = \frac{g(t)}{t-z}.$$

For $z \in \mathbb{C} \setminus \mathbb{R}_0^-$ the function $h_2(t)$ is piecewise continuous on every finite subinterval of $\mathbb{R}_0^-$ and the condition (3.14) from Theorem 3.7 is fulfilled with $C = |f(z)|$. Since all roots of $w_m(t)$ lie outside of $\Gamma = \mathbb{R}_0^-$ by assumption and the degree of the denominator of $h_1'(t)$ exceeds the degree of the numerator by at least two, h_1' is absolutely integrable over Γ. All other conditions on h_1 from Theorem 3.7 are obviously fulfilled, so that the integral (3.4) exists. This concludes the proof of the proposition. □

Proposition 3.8 guarantees the existence of the error function for Stieltjes functions generated by a differentiable function μ in a very general setting (i.e., as long as no Ritz value lies on the negative real axis, a case in which $f(H_m)$ is not defined). For general Stieltjes functions (3.15), we can at least guarantee the existence of the error function (3.7) if all Ritz values are real and positive, as it is, e.g., the case when the matrix A is Hermitian positive definite. In this case, the conditions $\mathrm{spec}(A) \subset \mathbb{C} \setminus \mathbb{R}_0^-$ and $\mathrm{spec}(H_m) \subset \mathbb{C} \setminus \mathbb{R}_0^-$ are always fulfilled. In addition, the nodal polynomial $\prod_{i=1}^m w_m(t) = (t+\theta_i)$ is positive for $t \geq 0$, and thus is $1/w_m(t)$, so that there exists a constant $\alpha > 0$ such that $1/w_m(t) \leq \frac{\alpha}{1+t}$ for $t \geq 0$. Using this fact together with the condition (2.16) imposed on μ we find

$$\widetilde{\mu}(t) := \int_0^t \frac{1}{w_m(\tau)}\,\mathrm{d}\mu(\tau) \leq \alpha \int_0^t \frac{1}{1+\tau}\,\mathrm{d}\mu(\tau) < \infty \tag{3.16}$$

for all $t \geq 0$. Since

$$\mathrm{d}\widetilde{\mu}(t) = \frac{1}{w_m(t)}\,\mathrm{d}\mu(t),$$

this yields the following proposition.

Proposition 3.9. *Let $A \in \mathbb{C}^{n\times n}$ be Hermitian positive definite, let $\boldsymbol{b} \in \mathbb{C}^n$ and let f be a Stieltjes function of the form* (3.15)*. Then the error function $e_m(z)$ from* (3.7) *is a scalar multiple of another Stieltjes function,*

$$e_m(z) = (-1)^{m+1}\|\boldsymbol{b}\|_2\gamma_m \int_0^\infty \frac{1}{z+t}\,\mathrm{d}\widetilde{\mu}(t),$$

generated by the function $\widetilde{\mu}$ from (3.16)*. In particular, the integral on the right-hand side of* (3.7) *exists and is finite.*

We note that the conditions given here for the existence of the integrals in the error function representation are sufficient, but *not* necessary and the integrals may exist under much weaker conditions.

CHAPTER 4

IMPLEMENTATION OF A QUADRATURE-BASED RESTARTED ARNOLDI METHOD

This chapter deals with the development of an efficient and numerically stable restarted Arnoldi method for functions with integral representation, based on the error function representation from Chapter 3. We first recapitulate the previously proposed restart procedures for Krylov subspace methods for matrix functions from the literature in Section 4.1 before presenting our new method based on adaptive quadrature in Section 4.2. Section 4.3 is devoted to specifics about the choice of quadrature rule for some important functions. In addition, we present results which reveal that these quadrature rules correspond to certain Padé approximants in case of the Stieltjes functions $f(z) = z^{-\alpha}$ and $f(z) = \log(1+z)/z$. Numerical experiments demonstrating the efficiency and stability of the proposed restart procedure in comparison to other approaches for the model problems from Section 2.6 are reported in Section 4.4.

4.1 Previously known restart approaches

Several approaches for restarting Arnoldi's method have been proposed in the literature so far. The simplest and most straightforward ones are based on the error function representations from Theorem 3.1 and Theorem 3.2, which directly allow (at least in exact arithmetic) to perform restarts like in the linear system case, with the only difference being that the function f is replaced by the error function e_m after restarting; see [93, 132]. In Algorithm 4.1, we summarize a generic version of such a restarted Arnoldi method (with constant restart length m) without going into detail on how the error function $e_m^{(k-1)}$ in line 4 is determined. This

Algorithm 4.1: Restarted Arnoldi method for $f(A)\boldsymbol{b}$ (generic version).

Given: A, $\boldsymbol{b}$, f, m

1 Compute the Arnoldi decomposition $AV_m^{(1)} = V_m^{(1)} H_m^{(1)} + h_{m+1,m}^{(1)} \boldsymbol{v}_{m+1}^{(1)} \hat{\boldsymbol{e}}_m^H$ with respect to A and $\boldsymbol{b}$.

2 $\boldsymbol{f}_m^{(1)} \leftarrow \|\boldsymbol{b}\|_2 V_m^{(1)} f(H_m^{(1)}) \boldsymbol{e}_1$.

3 **for** $k = 2, 3, \ldots$ until convergence **do**

4 Determine the error function $e_m^{(k-1)}$.

5 Compute the Arnoldi decomposition $AV_m^{(k)} = V_m^{(k)} H_m^{(k)} + h_{m+1,m}^{(k)} \boldsymbol{v}_{m+1}^{(k)} \hat{\boldsymbol{e}}_m^H$ with respect to A and $\boldsymbol{v}_{m+1}^{(k-1)}$.

6 $\boldsymbol{f}_m^{(k)} \leftarrow \boldsymbol{f}_m^{(k-1)} + V_m^{(k)} e_m^{(k-1)}(H_m^{(k)}) \hat{\boldsymbol{e}}_1$.

way, we can later also use this algorithm as a building block of our new restarted method based on the error representations from Theorem 3.4 and 3.5.

As discussed in Chapter 3, when using the previously known error function representations from Theorem 3.1 and 3.2, Algorithm 4.1 becomes unstable in floating point arithmetic and the presence of round-off errors, especially in later restart cycles (see [43, 93] and our experiments in Section 4.4).

As this instability was already recognized in [43], the authors proposed an alternative, mathematically equivalent restarted Arnoldi procedure, which is numerically stable but has computational complexity growing with the number of restart cycles (while Algorithm 4.1 requires constant work per cycle under the assumption that the evaluation of the error functions $e_m^{(k-1)}$ has the same cost for all values of k). For achieving a stable method, one first needs to define the accumulated Hessenberg matrices

$$H_{km} = \begin{bmatrix} H_{(k-1)m} & O \\ h_{m+1,m}^{(k-1)} \hat{\boldsymbol{e}}_1 \hat{\boldsymbol{e}}_{(k-1)m}^H & H_m^{(k)} \end{bmatrix} \in \mathbb{C}^{km \times km} \tag{4.1}$$

starting with $H_m = H_m^{(1)}$. One can then show that the iterates produced by Algorithm 4.1 satisfy the update formula

$$\boldsymbol{f}_m^{(k)} = \boldsymbol{f}_m^{(k-1)} + \|\boldsymbol{b}\|_2 V_m^{(k)} \boldsymbol{y}_{km}((k-1)m + 1 : km), \text{ where } \boldsymbol{y}_{km} = f(H_{km}) \hat{\boldsymbol{e}}_1 \tag{4.2}$$

when $k \geq 2$; see [43]. This way, one only ever needs to apply the original function f and circumvents the need of evaluating the error function $e_m^{(k-1)}$ for computing $\boldsymbol{f}_m^{(k)}$, so that the instability caused by the divided differences in the error function representation (3.1) is avoided. Therefore, computing the restarted Arnoldi iterates by means of (4.2) results in a stable method but requires the evaluation of f on the accumulated Hessenberg matrix H_{km} which is of size $km \times km$, i.e., growing from one restart cycle to the next. Thus, the resulting method solves

the storage problems of Arnoldi's method, as only the last Arnoldi basis $V_m^{(k)}$ is needed to evaluate (4.2), but its computational cost grows with k (often and typically cubically, depending on the function to be approximated). In fact, in a setting where not only storage requirements limit the applicability of Arnoldi's method, but also unacceptably high computational work is required to reach the targeted accuracy, problems typically become more severe when using a method based on (4.2). This is due to the fact that in most cases, a restarted method will need more iterations to reach a prescribed accuracy than the corresponding unrestarted method, so that the dimension of the matrix H_{km} needed in (4.2) will typically be larger than the dimension of the matrix H_m needed for computing a standard Arnoldi approximation (2.25) which gives a comparable accuracy. We note however, that while the preceding statements hold true for almost all practical problems, they may not be true in general, as there, e.g., also exist (academic) examples of matrices for which restarted GMRES converges more slowly when the restart length is increased; cf. [45].

To solve the problem of growing computational work in the method from [43] based on (4.2), while keeping its advantageous stability properties, a modification of the method was proposed in [3]. It requires that one wants to approximate the action of a rational function in partial fraction form on a vector, i.e., $r(A)\boldsymbol{b}$ with

$$r(z) = \sum_{i=1}^{\ell} \frac{\alpha_i}{t_i - z}, \tag{4.3}$$

or, in a more general setting, that one is interested in a function $f \approx r$, where r is of the form (4.3), i.e., f must be well approximable by a rational function (on the spectrum of A). In this case, one applies the algorithm to r instead of f and computes an approximation $\boldsymbol{f}_m^{(k)} \approx r(A)\boldsymbol{b} \approx f(A)\boldsymbol{b}$. One then obtains constant computational work per restart cycle as follows: Evaluating the update formula (4.2) with f replaced by r of the form (4.3) amounts to computing

$$r(H_{km})\hat{\boldsymbol{e}}_1 = \sum_{i=1}^{\ell} \alpha_i (t_i I - H_{km})^{-1} \hat{\boldsymbol{e}}_1, \tag{4.4}$$

which requires solving ℓ shifted linear systems

$$(t_i I - H_{km})\boldsymbol{r}(t_i) = \hat{\boldsymbol{e}}_1, \qquad i = 1, \ldots, \ell. \tag{4.5}$$

Partitioning the solutions as

$$\boldsymbol{r}(t_i) = \begin{bmatrix} \boldsymbol{r}_1(t_i) \\ \boldsymbol{r}_2(t_i) \\ \vdots \\ \boldsymbol{r}_k(t_i) \end{bmatrix}$$

and exploiting the block structure (4.1) of H_{km} together with the fact that all right-hand sides of (4.5) are equal to $\hat{\boldsymbol{e}}_1$, one finds that each system in (4.5) decouples into small linear systems of dimension m; cf. [3]. To be specific, one has for $i = 1, \ldots, \ell$

$$\begin{aligned} (t_i I - H_m^{(1)})\boldsymbol{r}_1(t_i) &= \hat{\boldsymbol{e}}_1, & (4.6)\\ (t_i I - H_m^{(j)})\boldsymbol{r}_j(t_i) &= h_{m+1,m}^{(j-1)}(\hat{\boldsymbol{e}}_m^H \boldsymbol{r}_{j-1}(t_i))\hat{\boldsymbol{e}}_1, \quad j = 2, \ldots, k. & (4.7) \end{aligned}$$

A further simplification arises from the fact that evaluating (4.2) requires only the last m entries of (4.4), which are given by

$$\boldsymbol{y}_{km}((k-1)m + 1 : km) = \sum_{i=1}^{\ell} \alpha_i \boldsymbol{r}_k(t_i),$$

such that not all, but only ℓ of the small $m \times m$ systems in (4.7) have to be solved. The resulting method is summarized in Algorithm 4.2.

Algorithm 4.2: Restarted Arnoldi method for $f(A)\boldsymbol{b}$ from [3].

Given: A, $\boldsymbol{b}$, m, rational approximation $r \approx f$ of the form (4.3)

$\boldsymbol{f}_m^{(0)} \leftarrow \boldsymbol{0}$
$\boldsymbol{v}_{m+1}^{(0)} \leftarrow \boldsymbol{b}$
for $k = 1, 2, \ldots$ until convergence **do**
  Compute the Arnoldi decomposition $AV_m^{(k)} = V_m^{(k)} H_m^{(k)} + h_{m+1,m}^{(k)} \boldsymbol{v}_{m+1}^{(k)} \hat{\boldsymbol{e}}_m^H$ with respect to A and $\boldsymbol{v}_{m+1}^{(k-1)}$.
  if $k = 1$ **then**
    for $i = 1, \ldots, \ell$ **do**
      Solve $(t_i I - H_m^{(k)})\boldsymbol{r}_1(t_i) = \hat{\boldsymbol{e}}_1$.
  else
    for $i = 1, \ldots, \ell$ **do**
      Solve $(t_i I - H_m^{(k)})\boldsymbol{r}_k(t_i) = h_{m+1,m}^{(k-1)}(\hat{\boldsymbol{e}}_m^H \boldsymbol{r}_{k-1}(t_i))\hat{\boldsymbol{e}}_1$.
  $\boldsymbol{u}_m^{(k)} \leftarrow \sum_{i=1}^{\ell} \alpha_i \boldsymbol{r}_k(t_i)$.
  Set $\boldsymbol{f}_m^{(k)} \leftarrow \boldsymbol{f}_m^{(k-1)} + \|\boldsymbol{b}\|_2 V_m^{(k)} \boldsymbol{u}_m^{(k)}$.

Algorithm 4.2 allows to approximate $f(A)\boldsymbol{b}$ both storage and cost efficiently, but it requires the knowledge of a suitable and accurate rational approximation of f (which has to be known a priori and needs to stay fixed throughout all cycles of the method). This requires information on the spectrum of A, so that r can be constructed in such a way that it approximates f accurately enough in all eigenvalues of A. In the Hermitian case, it therefore suffices to know $\lambda_{\min}$ and

$\lambda_{\max}$, the smallest and largest eigenvalue of A, respectively, and to construct r as a good approximation on $[\lambda_{\min}, \lambda_{\max}]$. In the non-Hermitian case, however, it is far more complicated to find a suitable region which contains $\mathrm{spec}(A)$ and to construct an accurate rational approximation on such a general region, which is no interval on the real line. In addition, Ritz values may appear anywhere in the field of values $\mathcal{W}(A)$ of A, such that it may be necessary that f be well approximated on an even larger set. Another limitation to the applicability of the approach from [3] is that for certain functions, no simple to construct rational approximations which give a certain accuracy may be known, even for "simple" spectral regions.

Thus, while the method from [3] has very advantageous properties when applicable (e.g., for approximating the exponential of a Hermitian negative definite matrix), it is no black-box method in general, as it often requires spectral information on A and it is only feasible for a rather narrow class of functions. Therefore, in the next section, by combining the error representations from Chapter 3 with numerical quadrature, we introduce another implementation of the restarted Arnoldi method, which inherits the stability properties and constant computational cost per cycle from Algorithm 4.2 and is applicable to a broad class of functions without requiring spectral information.

To end this section, we mention that Algorithm 4.2 is mathematically equivalent to restarted FOM for the shifted linear systems

$$(t_i I - A)\boldsymbol{x}(t_i) = \boldsymbol{b}, \tag{4.8}$$

a method introduced in [123], with the only difference being that the approximate solutions of the individual systems (4.8) do not need to be computed or stored explicitly, as one is only interested in the approximation to $r(A)\boldsymbol{b} \approx f(A)\boldsymbol{b}$. A similar point of view was recently discussed in [17], where this approach was used to save computational work when solving families of shifted (block) linear systems having their origin in a partial fraction expansion (4.3) of a rational function.

4.2 Restarts based on numerical quadrature

In this section, we describe how to use the integral representation of the Arnoldi error for deriving a numerically stable version of Algorithm 4.1. The key observation for achieving stability in this context is that the numerical evaluation of integrals is much more stable than the numerical evaluation of difference quotients (and thus, e.g., the computation of divided differences), see, e.g., [41], where this is discussed in the context of the numerical solution of differential equations.

To be able to use the error function representations from Theorem 3.4 and 3.5 in Algorithm 4.1, we require not only an integral representation for the approximation $\boldsymbol{f}_m$ obtained by m steps of Arnoldi's method, but also for the restarted approximations $\boldsymbol{f}_m^{(k)}$. Fortunately, recursively replacing f by $e_m^{(1)}, e_m^{(2)}, \dots$ in the statements of the theorems directly gives rise to such a representation. We summarize this result in the following corollary.

Corollary 4.1. *Let the assumptions of Theorem 3.4 hold and let $\boldsymbol{f}_m^{(k)}$ be the restarted Arnoldi approximation to $f(A)\boldsymbol{b}$ after k restart cycles of Algorithm 4.1. Denote by $w_m^{(j)}$ the nodal polynomial of the jth restart cycle, i.e., the monic polynomial of degree m with its roots given by the eigenvalues of $H_m^{(j)}$, and assume that all these roots do not lie on Γ and let $\gamma_m^{(j)} = \prod_{i=1}^{m} h_{i+1,i}^{(j)}$. Then the error of $\boldsymbol{f}_m^{(k)}$ satisfies*

$$\begin{aligned} f(A)\boldsymbol{b} - \boldsymbol{f}_m^{(k)} &= \|\boldsymbol{b}\|_2 \left(\prod_{j=1}^{k} \gamma_m^{(j)} \right) \int_\Gamma \frac{g(t)}{\prod_{j=1}^{k} w_m^{(j)}(t)} (tI - A)^{-1} \boldsymbol{v}_{m+1}^{(k)} \, \mathrm{d}t \quad (4.9) \\ &=: e_m^{(k)}(A)\boldsymbol{v}_{m+1}^{(k)}, \end{aligned}$$

provided that the integral (3.4)*, with $w_m(t) = \prod_{j=1}^{k} w_m^{(j)}(t)$, exists.*

In the same way, if f is a Stieltjes function (3.15), we have a representation of the form

$$f(A)\boldsymbol{b} - \boldsymbol{f}_m^{(k)} = e_m^{(k)}(A)\boldsymbol{v}_{m+1}^{(k)}, \tag{4.10}$$

where

$$e_m^{(k)}(z) = (-1)^{k(m+1)} \|\boldsymbol{b}\|_2 \left(\prod_{j=1}^{k} \gamma_m^{(j)} \right) \int_0^\infty \frac{1}{\prod_{j=1}^{k} w_m^{(j)}(t)} \frac{1}{z+t} \, \mathrm{d}\mu(t). \tag{4.11}$$

Instead of recursively inserting the error functions into Theorem 3.4, one also finds these results directly by using the fact that $\boldsymbol{f}_m^{(k)} = p_{km-1}(A)\boldsymbol{b}$, where p_{km-1} is the polynomial of degree $km-1$ that interpolates f on $\operatorname{spec}(H_{km})$, with H_{km} from (4.1), see [43], and reproducing the proof of Theorem 3.4 for this case. Note that the existence of the integrals in (4.9) and (4.11) can of course be guaranteed under the same assumptions as in Proposition 3.8 and 3.9, respectively.

With Corollary 4.1, we are in a position to formulate our new, quadrature-based restarted Arnoldi method. It mainly consists of using the error function representation (4.9) or (4.11) in Algorithm 4.1 and approximating $e_m^{(k-1)}(A)\boldsymbol{b}$ by a (suitably chosen) quadrature rule, as it is in general not possible to evaluate (4.9) or (4.11) exactly. As we do not know a priori how many quadrature nodes are necessary to approximate the error functions with sufficient accuracy (as even the error functions themselves are not known in advance) and as this number may

vary from one cycle to the next, we use a simple form of adaptive quadrature. Of course one can also use more sophisticated techniques than what is described in the following, as in general all forms of adaptive quadrature are suitable, but we found that this simple approach was sufficient in our setting, as the efficient evaluation of quadrature rules is not the bottleneck of the method.

At each restart cycle, we choose two sets of $\widetilde{\ell}$ and $\ell := \lfloor \sqrt{2} \cdot \widetilde{\ell} \rfloor$ quadrature nodes and weights, and approximate $e_m^{(k-1)}(H_m^{(k)})\hat{\boldsymbol{e}}_1$ by quadrature of these two different orders. If the norm of the difference between the two resulting approximations $\widetilde{\boldsymbol{u}}_m^{(k)}$ and $\boldsymbol{u}_m^{(k)}$ is smaller than a prescribed tolerance `tol`, the approximation $\boldsymbol{u}_m^{(k)}$ of higher order is accepted, otherwise we increase the number of quadrature points by another factor of $\sqrt{2}$ and continue this way until the desired accuracy is reached. This approach has two advantages. First (and obviously), if the initially chosen number of quadrature points is too small to reach the prescribed accuracy, it automatically increases the number as much as it is needed in the current cycle. On the other hand, the approach does in the same way allow to *decrease* the number of quadrature points, if fewer points suffice to obtain the required tolerance. Therefore, if in a restart cycle the number of quadrature points is not increased, we decrease the number of quadrature points for the next cycle by a factor of $\sqrt{2}$ and first test whether this lower number is already sufficient. This way, later restart cycles may indeed be *less expensive* than earlier cycles in our method. We go into more detail concerning this topic in the numerical experiments reported in Section 4.4. The resulting method is given as Algorithm 4.3 and was first introduced in [58], an implementation being provided in [59].

On first sight, one may assume that our approach may at one point be more prone to numerical instability than the one from [3], as it requires the evaluation of the nodal polynomial $w_m(t)$ which is of (possibly very high) degree m. However, note that

$$\frac{\gamma_m}{w_m(t)} = h_{m+1,m}\hat{\boldsymbol{e}}_m^H \left(tI_m - H_m\right)^{-1} \hat{\boldsymbol{e}}_1, \tag{4.12}$$

see, e.g., [115], so that the necessary scalar quantities can be computed by solving a shifted linear system of dimension m which can be done in a stable way. Another technique for reliably evaluating $w_m(t)$ in factored form is to use a suitable reordering of its zeros while computing the product. These approaches together with the numerical experiments reported in Section 4.4 suggest that our method is indeed numerically stable.

We proceed by further commenting on the relation between Algorithm 4.3 and Algorithm 4.2, the approach from [3], as this is the only other approach from the literature which also guarantees constant work per cycle as well as numerical stability. A further similarity of the two approaches is revealed by noting that an arbitrary quadrature rule with nodes t_i and weights ω_i for approximating the

Algorithm 4.3: Quadrature-based restarted Arnoldi method for $f(A)\boldsymbol{b}$.

Given: A, $\boldsymbol{b}$, f, m, `tol`

Compute the Arnoldi decomposition $AV_m^{(1)} = V_m^{(1)} H_m^{(1)} + h_{m+1,m}^{(1)} \boldsymbol{v}_{m+1}^{(1)} \hat{\boldsymbol{e}}_m^H$ with respect to A and $\boldsymbol{b}$.

$\boldsymbol{f}_m^{(1)} \leftarrow \|\boldsymbol{b}\|_2 V_m^{(1)} f(H_m^{(1)}) \hat{\boldsymbol{e}}_1$

$\widetilde{\ell} \leftarrow 8$, $\ell \leftarrow \text{round}(\sqrt{2} \cdot \widetilde{\ell})$

for $k = 2, 3, \ldots$ until convergence **do**

  Compute the Arnoldi decomposition $AV_m^{(k)} = V_m^{(k)} H_m^{(k)} + h_{m+1,m}^{(k)} \boldsymbol{v}_{m+1}^{(k)} \hat{\boldsymbol{e}}_m^H$ with respect to A and $\boldsymbol{v}_{m+1}^{(k-1)}$.

  Choose sets $(\widetilde{t}_i, \widetilde{\omega}_i)_{i=1,\ldots,\widetilde{\ell}}$, $(t_i, \omega_i)_{i=1,\ldots,\ell}$ of quadrature nodes/weights.

  `accurate` $\leftarrow$ `false`

  `refined` $\leftarrow$ `false`

  while `accurate` = `false` **do**

    Compute $\widetilde{\boldsymbol{u}}_m^{(k)} \approx e_m^{(k-1)}(H_m^{(k)}) \hat{\boldsymbol{e}}_1$ by quadrature of order $\widetilde{\ell}$.

    Compute $\boldsymbol{u}_m^{(k)} \approx e_m^{(k-1)}(H_m^{(k)}) \hat{\boldsymbol{e}}_1$ by quadrature of order ℓ.

    if $\|\boldsymbol{u}_m^{(k)} - \widetilde{\boldsymbol{u}}_m^{(k)}\|_2 <$ `tol` **then**

      `accurate` $\leftarrow$ `true`.

    else

      $\widetilde{\ell} \leftarrow \ell$

      $\ell \leftarrow \text{round}(\sqrt{2} \cdot \widetilde{\ell})$

      `refined` $\leftarrow$ `true`.

  $\boldsymbol{f}_m^{(k)} \leftarrow \boldsymbol{f}_m^{(k-1)} + \|\boldsymbol{b}\|_2 V_m^{(k)} \boldsymbol{u}_m^{(k)}$.

  if `refined` = `false` **then**

    $\ell \leftarrow \widetilde{\ell}$

    $\widetilde{\ell} \leftarrow \text{round}(\ell/\sqrt{2})$

error function $e_m(z)$ from (3.5) gives rise to an approximation of the form

$$\widehat{e}_m(z) = \|\boldsymbol{b}\|_2 \gamma_m \sum_{i=1}^{\ell} \omega_i \frac{g(t_i)}{w_m(t_i)} \frac{1}{t_i - z} \tag{4.13}$$

which clearly is a rational approximation (of type $(\ell - 1/\ell)$) for $e_m(z)$. Therefore, in a sense, both methods rely on using rational approximations for the error functions. In fact, we can show that under a few assumptions, both approaches are equivalent (at least assuming exact arithmetic). The precise result is given in the following lemma, where we refer to Algorithm 4.1 (using quadrature to evaluate the integral representation of the error) instead of Algorithm 4.3 to make clear that a non-adaptive approach is applied, in which the quadrature nodes and

weights are chosen a priori.

Lemma 4.2. *Let the quadrature nodes t_i and weights ω_i in* (4.13) *be fixed throughout all restart cycles in Algorithm 4.1. Let Algorithm 4.2 utilize a rational approximation in partial fraction form* (4.3) *with poles t_i and weights $\alpha_i = \omega_i g(t_i)$. Assume that this quadrature formula is also used to evaluate f in the first restart cycle of Algorithm 4.1. Then both algorithms produce the same approximations $\boldsymbol{f}_m^{(k)}$ at each restart cycle $k \geq 1$.*

Proof. From (4.6) and (4.13) (with $w_m \equiv 1$ in the first restart cycle) it immediately follows that both algorithms produce the same first Arnoldi approximation

$$\boldsymbol{f}_m^{(1)} = \|\boldsymbol{b}\|_2 V_m^{(1)} \sum_{i=1}^{\ell} \omega_i g(t_i)(t_i I - H_m^{(1)})^{-1} \hat{\boldsymbol{e}}_1.$$

In subsequent restart cycles $k \geq 2$ of Algorithm 4.1, using the error function representation (4.13), the approximations are computed as

$$\boldsymbol{f}_m^{(k)} = \boldsymbol{f}_m^{(k-1)} + \|\boldsymbol{b}\|_2 V_m^{(k)} \sum_{i=1}^{\ell} \frac{\omega_i g(t_i) \prod_{j=1}^{k-1} \gamma_m^{(j)}}{\prod_{j=1}^{k-1} w_m^{(j)}(t_i)} (t_i I - H_m^{(k)})^{-1} \hat{\boldsymbol{e}}_1. \tag{4.14}$$

From (4.7) we find $\boldsymbol{r}_k(t_i) = h_{m+1,m}^{(k-1)}(\hat{\boldsymbol{e}}_m^H \boldsymbol{r}_{k-1}(t_i))(t_i I - H_m^{(k)})^{-1} \hat{\boldsymbol{e}}_1$. Repeated application of (4.12) yields

$$h_{m+1,m}^{(k-1)}(\hat{\boldsymbol{e}}_m^H \boldsymbol{r}_{k-1}(t_i)) = \frac{\prod_{j=1}^{k-1} \gamma_m^{(j)}}{\prod_{j=1}^{k-1} w_m^{(j)}(t_i)},$$

so that (4.14) is equivalent to

$$\boldsymbol{f}_m^{(k)} = \boldsymbol{f}_m^{(k-1)} + \|\boldsymbol{b}\|_2 V_m^{(k)} \sum_{i=1}^{\ell} \omega_i g(t_i) \boldsymbol{r}_k(t_i),$$

which is precisely the update formula of Algorithm 4.2 when $\alpha_i = \omega_i g(t_i)$. □

In light of Lemma 4.2, one may ask which advantages our new approach gives in comparison to the one from [3]. To answer this question, we stress again that the result of Lemma 4.2 only holds in the very specific case that the quadrature rule is fixed once and for all before starting the method. While it is indeed necessary to fix the rational approximation in the approach from [3] in the beginning, this is not the case for our method. In addition, as long as the integration path Γ does not depend on the spectrum of A (which is, e.g., the case for Stieltjes functions), we need no additional information for the choice of quadrature rule,

in contrast to bounds for the spectral region necessary for constructing a rational approximation. Even in cases where the path Γ depends on $\mathrm{spec}(A)$, like it is, e.g., the case when approximating the matrix exponential, one can in many cases easily construct sufficiently accurate rational approximations by exploiting information obtained from Ritz values. This is not possible in Algorithm 4.2, as the rational approximation has to be chosen a priori and needs to stay fixed throughout all cycles, so that the spectral information available from the matrices $H_m^{(j)}$ cannot be exploited in any way. We go into detail concerning this topic in Section 4.3. Another advantage is the potential for adaptivity not only for guaranteeing that the prescribed accuracy is reached, but also for not investing more computational work than necessary in later restart cycles and therefore in some cases making the method even more efficient.

We just briefly mention here that it is also possible to combine our quadrature-based restart approach with the *deflated restarting* technique from [44] in a straightforward way. This technique is also included in our implementation [59] of the restarted Arnoldi method, but we do not give the details for this here and refer to [58] for numerical experiments illustrating the behavior of the resulting method.

An important point influencing the performance of our method which we have not yet discussed in detail is the choice of quadrature rules for evaluating the error function $e_m^{(k-1)}$. While in principle, we can use any convergent quadrature rule in our adaptive algorithm, making it a black-box method, there are natural choices of quadrature rules for certain functions which allow to improve the performance of the method even further and also reveal interesting theoretical connections to certain types of optimal rational approximants. This is the topic of the next section.

4.3 Choice of quadrature rules and connection to Padé approximation

In this section, we will exemplarily go into more detail concerning the choice of quadrature rules for three different functions, namely the Stieltjes functions $f(z) = z^{-\alpha}, \alpha \in (0,1)$ and $f(z) = \log(1+z)/z$, and the exponential function $f(z) = e^z$. As stated at the end of the last section, if the path of integration is known (as, e.g., for Stieltjes functions) using any of the convergent quadrature rules presented in Section 2.5 is in principle possible, but there are often better choices available when exploiting specific properties of the function at hand. Concerning the exponential function, the path Γ is not known in advance if no

spectral information on A is available, as it is typically the case when A is non-Hermitian (in the Hermitian negative definite case, Hankel contours that enclose the negative real axis are suitable integration paths) and we will also comment on how to adaptively construct a suitable contour in this case.

As it turns out that certain choices of quadrature rules for Stieltjes functions correspond to certain (optimal) rational approximants of these functions, the so-called *Padé approximants* [8–10, 53, 110], we review the basic definition of these approximants before proceeding.

Definition 4.3. Let f be a function and let $m \geq 0, \ell \geq 1$ be given. An (m/ℓ) *Padé approximant* of f with expansion point a is a rational function

$$r_{m,\ell}(z) = \frac{p_m(z)}{q_\ell(z)} \text{ where } \deg p_m \leq m, \deg q_\ell \leq \ell,$$

such that

$$\frac{\mathrm{d}^j}{\mathrm{d}z^j} f(z)\bigg|_{z=a} = \frac{\mathrm{d}^j}{\mathrm{d}z^j} r_{m,\ell}(z)\bigg|_{z=a} \text{ for } j = 0, \ldots, m+\ell. \tag{4.15}$$

We note that, by a classical result from [53,110], if an (m/ℓ) Padé approximant to a function f exists, then it is unique, so that we will in the following refer to $r_{m,\ell}$ as *the* (m/ℓ) Padé approximant of f at a. We also note that there exist other (more general) definitions of Padé approximants, cf. [8], which agree with Definition 4.3 when both are applicable, but use other matching conditions than (4.15). As this is not of importance in our situation, we do not go into detail concerning this topic.

We begin by considering quadrature rules for the integral representation (2.18) of the inverse fractional powers $f(z) = z^{-\alpha}, \alpha \in (0,1)$. In this Stieltjes representation, the integration interval $\Gamma = \mathbb{R}_0^+$ is known a priori but infinite. Instead of using a quadrature rule for infinite intervals, one can also apply a suitable variable transformation.

Lemma 4.4. *Let $z \in \mathbb{C} \setminus \mathbb{R}_0^-$ and $\alpha \in (0,1)$. Then for all $\beta > 0$*

$$z^{-\alpha} = \frac{2\beta^{1-\alpha}\sin(\alpha\pi)}{\pi} \int_{-1}^{1} \frac{(1-x)^{\alpha-1}(1+x)^{-\alpha}}{\beta(1+x)+z(1-x)}\,\mathrm{d}x. \tag{4.16}$$

Proof. We apply the variable transformation $t = \beta\frac{1+x}{1-x}$ to (2.18). Noting that $\frac{\mathrm{d}t}{\mathrm{d}x} = \frac{2\beta}{(1-x)^2}$ and integrating by substitution, we find

$$\begin{aligned} z^{-\alpha} &= \frac{\sin(\alpha\pi)}{\pi}\int_{-1}^{1}\frac{\left(\beta\frac{1+x}{1-x}\right)^{-\alpha}}{\beta\frac{1+x}{1-x}+z}\cdot\frac{2\beta}{(1-x)^2}\,\mathrm{d}x \\ &= \frac{2\beta^{1-\alpha}\sin(\alpha\pi)}{\pi}\int_{-1}^{1}\frac{(1-x)^{\alpha}(1+x)^{-\alpha}}{\beta(1+x)(1-x)+z(1-x)^2}\,\mathrm{d}x, \end{aligned}$$

from which the assertion follows. □

The integrand in (4.16) has singularities at both endpoints -1 and 1. However, these singularities are contained only in the numerator, which exactly corresponds to the $(\alpha-1,-\alpha)$ Jacobi weight function; cf. Example 2.45. Therefore, we can use Gauss–Jacobi quadrature to resolve it exactly. The remaining integrand has no singularities as long as $z \in \mathbb{C}\backslash\mathbb{R}_0^-$ (for $z \in \mathbb{R}_0^-$, the original integral representation also has a singularity). The following result reveals a connection between Gauss–Jacobi quadrature for (4.16) and Padé approximants.

Lemma 4.5. *Let $\beta > 0$ and let x_i and $\omega_i, i = 1,\dots,\ell$ be the nodes and weights of the ℓ-point $(\alpha-1,-\alpha)$ Gauss–Jacobi quadrature rule on $[-1,1]$. Then*

$$r_{\ell-1,\ell}(z) = \frac{2\beta^{1-\alpha}\sin(\alpha\pi)}{\pi}\sum_{i=1}^{\ell}\frac{\omega_i}{\beta(1+x_i)+z(1-x_i)} \tag{4.17}$$

is the $(\ell-1/\ell)$ Padé approximant of $z^{-\alpha}, \alpha \in (0,1)$, with expansion point β.

Proof. Note that (4.17) clearly is a rational function of type $(\ell-1/\ell)$ in partial fraction form. Therefore we only have to verify the Padé matching conditions

$$\frac{\mathrm{d}^j}{\mathrm{d}z^j}z^{-\alpha}\bigg|_{z=\beta} = \frac{\mathrm{d}^j}{\mathrm{d}z^j}r_{\ell-1,\ell}(z)\bigg|_{z=\beta} \quad \text{for } j = 0,\dots,2\ell-1.$$

The derivatives of $r_{\ell-1,\ell}(z)$ are given by

$$\frac{\mathrm{d}^j}{\mathrm{d}z^j}r_{\ell-1,\ell}(z) = -\frac{2\beta^{1-\alpha}\sin(\alpha\pi)}{\pi}\sum_{i=1}^{\ell}(-1)^j\frac{j!\cdot(1-x_i)^j\cdot\omega_i}{(\beta(1+x_i)+z(1-x_i))^{j+1}}. \tag{4.18}$$

For $z = \beta$ all denominators in (4.18) become independent of x_i and we arrive at

$$\frac{\mathrm{d}^j}{\mathrm{d}z^j}r_{\ell-1,\ell}(z)\bigg|_{z=\beta} = -\frac{2\beta^{1-\alpha}\sin(\alpha\pi)}{\pi}\sum_{i=1}^{\ell}(-1)^j\frac{j!\cdot(1-x_i)^j\cdot\omega_i}{(2\beta)^{j+1}}.$$

As Gauss–Jacobi quadrature with ℓ nodes is exact for polynomials up to degree $2\ell - 1$, we have the relation

$$\frac{\mathrm{d}^j}{\mathrm{d}z^j} r_{\ell-1,\ell}(z)\bigg|_{z=\beta} = \frac{2\beta^{1-\alpha} j! \cdot \sin(\alpha\pi)}{(2\beta)^{j+1}\pi}(-1)^j \int_{-1}^{1} (1-x)^j (1-x)^{\alpha-1}(1+x)^{-\alpha}\,\mathrm{d}x.$$

for $j = 0, \ldots, 2\ell - 1$. Differentiating the right-hand side of (4.16) and evaluating at β gives the same result, which completes the proof. □

We note that it is known that the rational functions generated by certain Gauss quadrature rules coincide with certain Padé approximants, see, e.g., [4, 21], but the precise result of Lemma 4.5 was not given in this explicit form before to the best of our knowledge. As the approximation quality of Padé approximants is typically highest close to the expansion point, it seems reasonable to choose the transformation parameter β such that the eigenvalues of A are clustered around it. A straightforward choice therefore is the arithmetic mean of the eigenvalues of A, which is readily available as $\operatorname{trace}(A)/n$. Other, more sophisticated choices of β are of course possible in our setting due to the availability of Ritz value information, but we observed in our experiments that this has no large influence on the overall behavior of the method, especially as the cost of evaluating the quadrature rules in Algorithm 4.3 is typically negligible compared to the cost of matrix vector products and orthogonalization.

In Algorithm 4.3, the quadrature formula is of course not applied to evaluate f, but instead to evaluate the error functions $e_m^{(k-1)}$, for which the quadrature rule for the transformed integral does not correspond to a Padé approximant, but can still be expected to yield good approximations. Applying the same variable transformation as in Lemma 4.4 to the integral representation of the error function (3.7) corresponding to $z^{-\alpha}$ results in the transformed integral

$$(-1)^{m+1}\frac{2\sin(\alpha\pi)\beta^{1-\alpha}\|\boldsymbol{b}\|_2\gamma_m}{\pi}\int_{-1}^{1}\frac{1}{w_m(\beta\frac{1+x}{1-x})}\frac{(1-x)^{\alpha-1}(1+x)^{-\alpha}}{\beta(1+x)+z(1-x)}\,\mathrm{d}x. \tag{4.19}$$

The integrand again has singularities at both endpoints of the interval of integration which can be resolved exactly by Gauss–Jacobi quadrature, but the reciprocal of the nodal polynomial introduces m additional singularities. Obviously, the singularities of the reciprocal of w_m prior to the variable transformation are exactly the Ritz values with switched sign. As the variable transformation bijectively maps $\mathbb{R}_0^+$ to $[-1, 1]$, the transformed integrand therefore has a singularity in the integration interval if and only if there is a Ritz value on the negative real axis. As all Ritz values lie in the field of values $\mathcal{W}(A)$ of A, a sufficient condition for the integrand in (4.19) having no singularities in the interval of integration is that the field of values of A is disjoint from the negative real axis. This is, e.g., the case, when A is Hermitian positive definite, or, more generally, when A is positive real.

If these conditions on A are not fulfilled it may well happen that Algorithm 4.3 generates Ritz values on the negative real axis. This is, however, no shortcoming of our method, but rather a general problem. None of the restart approaches will work in this case, as the function $z^{-\alpha}$ and therefore also the error functions are not defined on the branch-cut along $\mathbb{R}_0^-$.

Another Stieltjes function for which a similar analysis as for $z^{-\alpha}$ can be performed is $f(z) = \log(1+z)/z$ with the integral representation (2.19). We again begin by transforming the infinite integration interval to a finite one.

Lemma 4.6. *Let $z \in \mathbb{C} \setminus (-\infty, -1]$. Then*

$$\frac{\log(1+z)}{z} = \int_{-1}^{1} \frac{1}{z(1-x)+2}\,\mathrm{d}x. \tag{4.20}$$

Proof. We apply the transformation $t = 2/(1-x)$, which satisfies $\frac{\mathrm{d}t}{\mathrm{d}x} = \frac{2}{(1-x)^2}$, to (2.19). Integrating by substitution gives

$$\begin{aligned} \frac{\log(1+z)}{z} &= \int_{-1}^{1} \frac{\frac{1-x}{2}}{\frac{2}{1-x}+z} \cdot \frac{2}{(1-x)^2}\,\mathrm{d}x \\ &= \int_{-1}^{1} \frac{1}{\frac{2}{1-x}+z} \cdot \frac{1}{1-x}\,\mathrm{d}x, \end{aligned}$$

which proves the lemma. □

When using Gauss–Legendre quadrature for approximating the integral (4.20), we again find a connection to Padé approximants.

Lemma 4.7. *Let x_i and $\omega_i, i = 1, \ldots, \ell$ be the nodes and weights of the ℓ-point Gauss–Legendre quadrature rule on $[-1, 1]$. Then*

$$r_{\ell-1,\ell}(z) = \sum_{i=1}^{\ell} \frac{\omega_i}{z(1-x_i)+2}$$

is the $(\ell - 1/\ell)$ Padé approximant of $\log(1+z)/z$ with expansion point 0.

Proof. The proof proceeds analogously to the proof of Lemma 4.5 by noting that ℓ-point Gauss–Legendre quadrature is exact for polynomials of degree up to $2\ell-1$, and using the formula

$$\frac{\mathrm{d}^j}{\mathrm{d}z^j} r_{\ell-1,\ell}(z) = \sum_{i=1}^{\ell} (-1)^j \frac{j! \cdot (1-x_i)^j \cdot \omega_i}{(z(1-x_i)+2)^{j+1}}$$

for the derivatives of $r_{\ell-1,\ell}(z)$. □

In contrast to Lemma 4.5, the result of Lemma 4.7 does not allow to freely choose the expansion point of the Padé approximant. Instead, it is always fixed at the origin. As already mentioned in the discussion after Lemma 4.5, the method does not behave very sensitively with respect to the expansion point anyway, so that we can expect the method to work efficiently also in cases where the matrix A has eigenvalues far away from the origin. Apart from this minor difference, most of the discussion for $z^{-\alpha}$ also applies in the case of $\log(1+z)/z$ with obvious modifications, so that we refrain from restating this here.

The last function we discuss in detail in this section is the exponential function, $f(z) = e^z$. We use the Cauchy integral representation

$$e^z = \frac{1}{2\pi i} \int_\Gamma \frac{e^t}{t - z} \, \mathrm{d}t \tag{4.21}$$

where Γ is a closed contour in the extended complex plane winding around z exactly once, which directly translates into an integral representation for the matrix exponential

$$e^A = \frac{1}{2\pi i} \int_\Gamma e^t \, (tI - A)^{-1} \, \mathrm{d}t,$$

where Γ now has to wind around $\mathrm{spec}(A)$ exactly once; cf. Definition 2.4. An important special case arises when A is Hermitian negative semi-definite, i.e., its eigenvalues lie in $\mathbb{R}_0^-$. It was shown in [135, 141, 142] that the trapezoidal rule on suitably chosen parabolic, hyperbolic or cotangent Hankel contours (so-called Talbot contours, introduced in [133]) gives very good approximation results for the exponential function on the negative real axis (and thus also for the matrix exponential of Hermitian negative semi-definite matrices). Therefore, these contours seem well suited to be used in Algorithm 4.3 in this case. In the following, we only discuss parabolic contours, as the results are very similar for all three types of contours in our setting.

The optimized parabolic contour proposed in [135] is given as

$$\gamma(\zeta) = \ell(0.1309 - 0.1194\zeta^2 + 0.25i\zeta) \tag{4.22}$$

and the ℓ-point trapezoidal rule applied to (4.21) on the contour (4.22) gives a convergence rate of $\mathcal{O}(2.85^{-\ell})$. The resulting contour for different values of ℓ is given in Figure 4.1. While in the experiments reported in [135] at most $\ell = 32$ quadrature nodes were necessary to approximate the exponential function on $\mathbb{R}_0^-$ to machine precision, the situation is different in the context of our restarted Arnoldi method. On the one hand, we are not interested in approximating the exponential itself, but rather the error function (3.5) and on the other hand, we are not only interested in achieving a high accuracy on $\mathbb{R}_0^-$, but, when A is non-Hermitian with field of values in the left half-plane, in a larger region of the left half-plane which contains all Ritz values. Because of these two reasons, higher

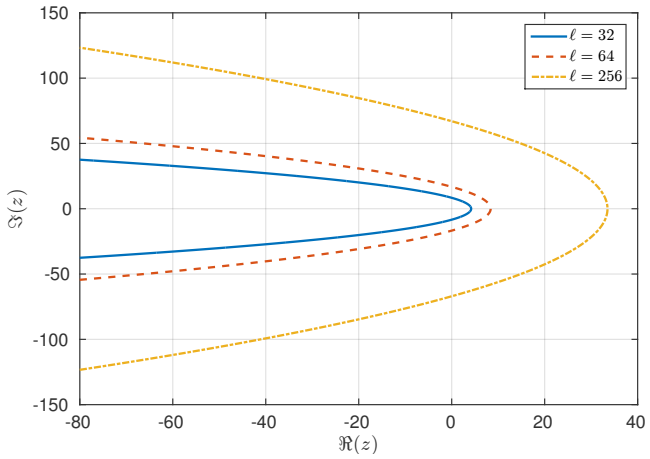


Figure 4.1: Parabolic contour from [135] for $\ell = 32, 64$ and 256. For larger values of ℓ, the intersections of the contour and the imaginary axis move farther away from the origin.

numbers of quadrature points were necessary in our experiments. In these cases, the corresponding contour (4.22) intersected the imaginary axis far away from the origin, and the integrand is highly oscillatory along the parts of the contour near the imaginary axis, which resulted in numerical instabilities.

We therefore use a different (non-optimal) parabolic contour, which is fixed in the sense that it does not depend on the number of quadrature points used. Specifically, we use

$$\gamma(\zeta) = a + i\zeta - c\zeta^2, \quad \zeta \in \mathbb{R}, \tag{4.23}$$

defined by two parameters $a, c > 0$. By varying the value of a one can shift the contour from left to right while the parameter c controls the "width" of the contour. Figure 4.2 shows the resulting contours for different values of the two parameters. In Algorithm 4.3, we adaptively choose the parameters a and c in such a way that all Ritz values are contained in the interior of the contour, i.e., the contour possibly changes after each restart. This is done as follows. Let Θ denote the set of Ritz values accumulated throughout all restart cycles performed thus far. Then, we choose

$$a = \max\left(\{\Re(\theta) + 1 \mid \theta \in \Theta\} \cup \{1\}\right) \tag{4.24}$$

and

$$c = \min\left(\{(a - \Re(\theta) - 1)/\Im(\theta)^2\} \mid \theta \in \Theta\} \cup \{0.25\}\right). \tag{4.25}$$

Note that the addition of one to the real part of θ in (4.24) is a "safety measure" to ensure that all Ritz values have a positive distance from the contour (and thus, any other positive value different from one could be used in principle). In the

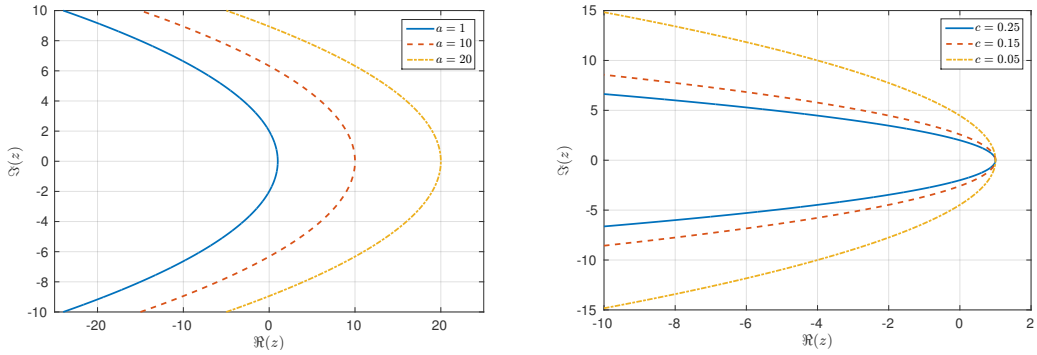


Figure 4.2: Parabolic contour used in our restarted Arnoldi method for the exponential function. On the left, contours resulting for $c = 0.25$ and $a = 1, 10$ and 20 are shown. On the right, contours resulting for $a = 1$ and $c = 0.25, 0.15$ and 0.05 are shown.

same way, the values one in (4.24) and 0.25 in (4.25) are chosen such that the "right endpoint" of the contour does not come too close to the origin and that the contour has a prescribed minimal width (and thus distance to the negative real axis). With this choice of a and c, we can guarantee that all Ritz values lie in the interior of the corresponding parabolic contour, from which it follows that quadrature on this contour really approximates the error function in question.

Proposition 4.8. *Let $\Theta \subseteq \mathbb{C}$ be compact and let a and c be given by* (4.24) *and* (4.25)*, respectively. Then all $\theta \in \Theta$ lie in the interior of the contour γ from* (4.23) *defined by these parameters.*

Proof. Let $\theta \in \Theta$. Then by the definition of γ in (4.23), we have $\Re(\theta) = \Re(\gamma(\zeta_\theta))$ for

$$\zeta_\theta = \pm\sqrt{(a - \Re(\theta))/c}. \tag{4.26}$$

Note that $(a - \Re(\theta))/c$ is positive due to the choice of a and the fact that $c > 0$. On the other hand, by the choice of c, we have

$$\begin{aligned} & c < (a - \Re(\theta))/(\Im(\theta)^2) \\ \Leftrightarrow \quad & \Im(\theta)^2 < (a - \Re(\theta))/c \\ \Leftrightarrow \quad & |\Im(\theta)| < |\sqrt{(a - \Re(\theta)/c}|. \end{aligned} \tag{4.27}$$

By (4.26) and the fact that $\Im(\gamma(\zeta_\theta)) = \zeta_\theta$, the inequality in (4.27) is equivalent to

$$|\Im(\gamma(\zeta_\theta))| > |\Im(\theta)|,$$

which shows that θ lies inside the contour γ. □

The contour (4.23) is infinite, so we have to truncate it for practical computations. Given a prescribed tolerance $\texttt{tol}$, we compute a truncation parameter

$$\zeta_{\texttt{tol}} = \sqrt{(a - \log(\texttt{tol}))/c}. \tag{4.28}$$

A straightforward calculation shows that for the parameter $\zeta_{\texttt{tol}}$ from (4.28) one has $|e^{\gamma(\pm\zeta_{\texttt{tol}})}| = \texttt{tol}$, so that the values of the integrand for $|\zeta| > \zeta_{\texttt{tol}}$ are negligibly small. Using the contour (4.23) as integration path Γ in (4.21) and truncating at $\pm\zeta_{\texttt{tol}}$, we have

$$e^z = \frac{1}{2\pi i}\int_{-\infty}^{\infty} \frac{e^{\gamma(\zeta)}\gamma'(\zeta)}{\gamma(\zeta) - z}\,\mathrm{d}\zeta \approx \frac{1}{2\pi i}\int_{-\zeta_{\texttt{tol}}}^{\zeta_{\texttt{tol}}} \frac{e^{\gamma(\zeta)}\gamma'(\zeta)}{\gamma(\zeta) - z}\,\mathrm{d}\zeta. \tag{4.29}$$

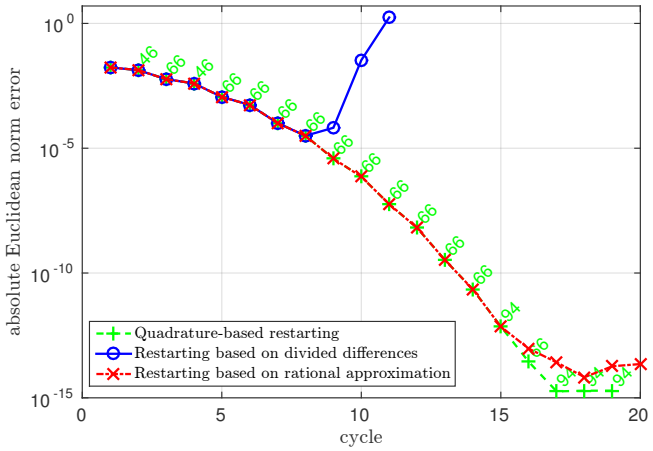
Applying the ℓ-point compound midpoint rule with equidistantly spaced quadrature nodes $\zeta_j = \zeta_{\texttt{tol}} \cdot \left(\frac{2j-1}{\ell} - 1\right), j = 1, \ldots, \ell$ to (4.29) then gives the quadrature approximation

$$\frac{2\zeta_{\texttt{tol}}}{\ell} \sum_{j=1}^{\ell} \frac{e^{\gamma(\zeta_j)}\gamma'(\zeta_j)}{\gamma(\zeta_j) - z},$$

which we use in our restarted Arnoldi method (of course applied to the error function $e_m^{(k-1)}(z)$ instead of e^z). Numerical experiments performed with this approach are reported in the next section.

4.4 Numerical experiments

In this section, we illustrate the performance of the quadrature-based restarted Arnoldi method when applied to the model problems from Section 2.6. We focus on numerical efficiency (i.e., execution time) and stability when compared to the other restarting approaches described in Section 4.1, and will not investigate the dependency on parameters like, e.g., the restart length in detail. We refer to, e.g., [3, 43] for a thorough treatment of these issues. All experiments are performed in MATLAB R2013a using our implementation `FUNM_QUAD` of the restarted Arnoldi method [59]. The methods from [3, 43] were tested using the `FUNM_KRYL` implementation [44] and the method from [93,126] was implemented based on the same version of the Lanczos process. We stress that MATLAB codes are not always best suited for comparing running times of algorithms (in large parts due to the fact that part of the code is interpreted and not pre-compiled) but that in our setting, where most of the time in all methods is spent in performing matrix vector products (which are calls to precompiled MATLAB routines) and all algorithms rely on the same implementation of the Arnoldi/Lanczos process, significant differences in running time can be trusted to be meaningful. In all tests, we use the tolerance $\texttt{tol} = 10^{-13}$ for the adaptive quadrature in Algorithm 4.3, which was

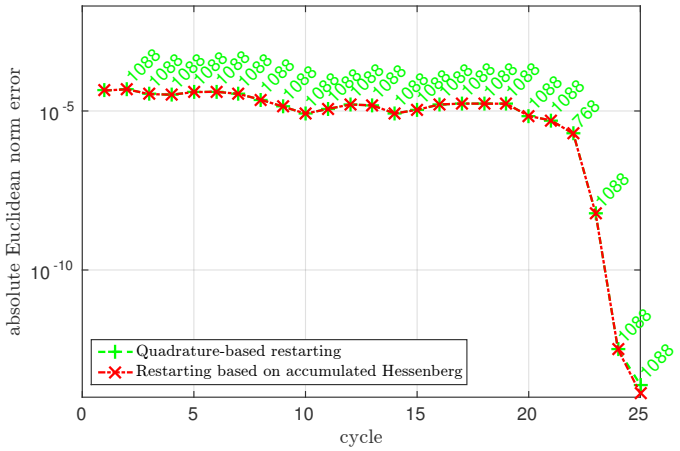


method	time
quadrature	1.25 s
rational approx.	1.16 s
divided diff.	0.8 s

Figure 4.3: Approximating $e^{\theta A}\boldsymbol{b}$: Convergence history (left) and running times (right) of different restarting algorithms for the semi-discretization of a three-dimensional heat equation. The numbers next to the curve for the quadrature-based method indicate the number of quadrature nodes used for evaluating the error function in the corresponding restart cycle. The restart length is $m = 20$ in all cases.

enough for the Arnoldi approximation to converge to sufficient accuracy for all test cases. In addition to the running time of our algorithm, we also report the number of quadrature nodes necessary to reach the prescribed accuracy `tol` in each restart cycle. The initial number of quadrature nodes used was $\ell = 8$. The "exact solutions" used in the experiments in this chapter as well as at all other places in this thesis (except for some small examples where they are easily computable via a full eigenvalue decomposition of A) are in reality approximations which have been precomputed to an accuracy of about 10^{-15} (with guaranteed error bounds when possible) by unrestarted Krylov subspace methods. This allows us to consider model problems of large size (for which an explicit computation of $f(A)\boldsymbol{b}$ is infeasible) while still being sure that the used solutions are accurate enough to behave exactly as the exact solution in our experiments.

We begin by reporting the results for the three-dimensional heat equation, our first model problem. We compare our quadrature-based restart approach, Algorithm 4.3, to the method based on divided differences from [93], which is applicable because A is Hermitian, and Algorithm 4.2 using the best uniform rational approximation of degree 16 for the exponential function on the negative real axis [25,27], which can be constructed *without* knowledge of the spectral interval of A. We use restart length $m = 20$ in all three methods. On the left-hand side of Figure 4.3, the convergence curves of the three methods are depicted, the corresponding execution times are given on the right-hand side. In the first eight restart cycles, all three methods behave exactly the same. From the ninth cycle on, the instability of the divided difference based method becomes visible, such that the

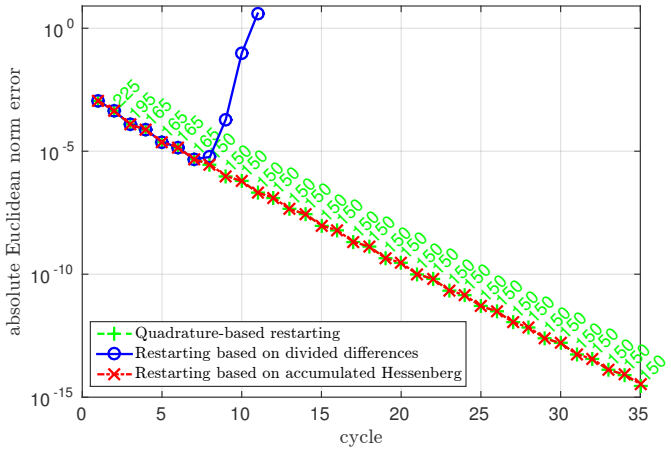


method	time
quadrature	3.6 s
acc. Hessenberg	5.2 s

Figure 4.4: Approximating $e^{\theta A}\boldsymbol{b}$: Convergence history (left) and running times (right) of different restarting algorithms for the semi-discretization of a three-dimensional convection diffusion equation. The numbers next to the curve for the quadrature-based method indicate the number of quadrature nodes used for evaluating the error function in the corresponding restart cycle. The restart length is $m = 20$ in both cases.

method only reaches an absolute error norm of about 10^{-5} before diverging. The other two methods continue to behave exactly the same, apart from the fact that the final error norm reached in the quadrature-based method is about one order of magnitude lower than when using the fixed rational approximation, which is most probably due to the accuracy of the rational approximation. The running time of the two converging methods is also about the same, with the rational approximation method being slightly faster. This can be explained by the lower degree of rational approximation used (16 in contrast to a degree between 46 and 94 for the quadrature-based method) and the additional overhead for the adaptive quadrature in our method. However, the difference in running time is almost negligible and may as such have no real meaning at all (cf. the discussion of MATLAB timings at the beginning of this section). The running time for the divided difference based method is slightly lower, but only 10 iterations were performed until the instability was detected, meaning that the other two methods perform slightly faster per iteration. However, the difference is again not large enough to be significant. This example illustrates that our quadrature-based method works and is stable in a case where the method based on divided differences becomes unstable. However, it does not show any superiority in comparison to the rational approximation method so far. Due to A being Hermitian negative definite and the availability of a low-degree rational approximation for e^z which does not depend on spec(A), both are black-box methods in this case and have very similar running times.

A first advantage of our restarted method can be demonstrated when considering



method	time
quadrature	3 s
divided diff.	0.75 s
acc. Hessenberg	32 s

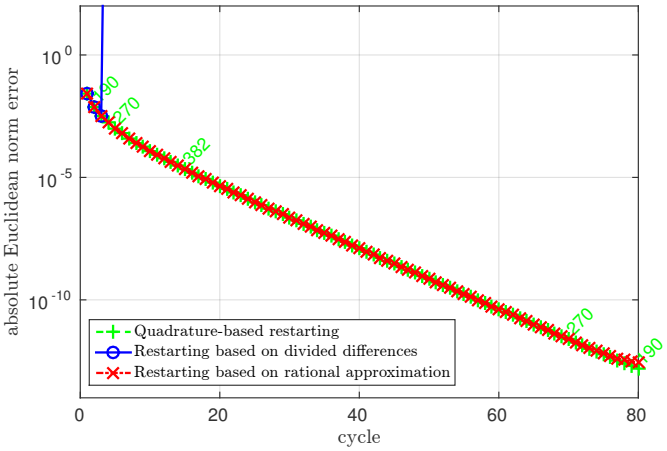
Figure 4.5: Approximating $f(z) = \frac{e^{-\theta\sqrt{z}}-1}{z}$: Convergence history (left) and running times (right) of different restarting algorithms for the semi-discretization of a three-dimensional wave equation. The numbers next to the curves for the quadrature-based method indicate the number of quadrature nodes used for evaluating the error function in the corresponding restart cycle. The restart length is $m = 20$ in all cases.

the second model problem, the three-dimensional convection diffusion equation. Here, the matrix A resulting from the semi-discretization is non-Hermitian, so that no good rational approximation for e^z on $\mathcal{W}(A)$ is available in a straightforward way. Therefore, the approach using the accumulated Hessenberg matrix (4.1) from [43] was the only restart method available for this test problem so far (the method from [93] is not applicable as A is non-Hermitian). In the quadrature-based method, we use adaptively constructed parabolic Hankel contours enclosing all Ritz values as described in Section 4.3, and we again use the restart length $m = 20$ in both methods. The convergence curves are given on the left of Figure 4.4, while the running times are reported on the right. Both methods behave the same again, showing an initial phase in which almost no progress is made, before rapid convergence takes place in the last few cycles. Algorithm 4.3 uses 1088 or 1540 quadrature nodes in all cycles for this problem, showing that the error function is much more difficult to evaluate than in the previous example, where at most 94 nodes were used. Still our method is about one third faster than the method proposed in [43] due to the fact that we only work with $m \times m$ matrices throughout all restart cycles, while the evaluation of $e^{\theta H_{km}}$ becomes increasingly expensive in later restart cycles. Therefore, the more restart cycles are necessary, the larger the benefit of using our new method will be (as long as one has no good rational approximation at hand). This is illustrated in the next experiment.

When considering the semi-discretization of the three-dimensional wave equation (2.60), one has to deal with the function $f(z) = \frac{e^{-\theta\sqrt{z}}-1}{z}$, which is neither an entire function nor a Stieltjes function (albeit closely related, just generated by

an oscillating, nonmonotonic function μ). Still, Proposition 3.8 applies to it and guarantees the existence of the integral representation of the error function. For this function, it is again not trivial to construct a good rational approximation of sufficient accuracy, even though A is Hermitian positive definite. We therefore compare our method to the method based on accumulated Hessenberg matrices and to the method based on divided differences, which is applicable because A is Hermitian. We again use restart length $m = 20$ and, as it is difficult to find a suitable quadrature rule for the integral (2.63) due to the oscillatory behavior of the integrand (which in case one uses a variable transformation onto a finite interval, leads to increasingly high-frequent oscillations when approaching one endpoint of the integration interval), we simply use the MATLAB routine `quadgk` [120] to evaluate the integral representation of the error function. In contrast to the other model problems, where we use the hand-tailored quadrature rules from Section 4.3, this means that we are, e.g., not able to exploit update formulas for the values of the reciprocal nodal polynomial $1/w_m(t_i)$ at the quadrature nodes, resulting in many superfluous computations. In Figure 4.5, convergence curves are again given on the left, while timings are reported on the right. Our method and the method from [43] again behave exactly the same, while the method from [93] again only reaches an accuracy of about 10^{-5} before it starts to diverge. The running times show a clear superiority of our method this time. Even though a more efficient implementation would be possible by constructing a suitable quadrature rule by hand, the method is still faster by a factor of about ten. The running time of the divided difference based method is only reported for the sake of completeness. Considering the number of restart cycles performed, one sees that one cycle of this method has roughly the same cost as one cycle of the quadrature-based method.

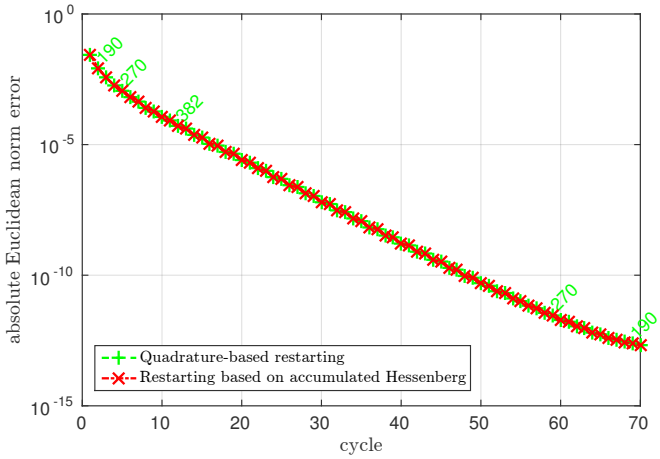
Next, we consider the model problems arising from lattice QCD computations. We begin by computing the sign function of the Wilson–Dirac operator at zero chemical potential, which corresponds to evaluating the inverse square root (i.e., a Stieltjes function) of a Hermitian positive definite matrix. We again compare our method to the divided difference based approach and the method employing a rational approximation. In the latter approach, we use the best relative Zolotarev approximation of degree 32 on the spectral interval of A [145]. Constructing this approximation requires knowledge of the largest and smallest eigenvalue of A, such that the rational approximation method is not a black-box method in this case, but requires spectral information on A. The time consumed for computing these eigenvalues and constructing the approximation is not included in the reported timings. Our quadrature-based method uses Gauss–Jacobi quadrature as explained in Section 4.3. As $\alpha = 1/2$ in this model problem, we have that $-\alpha = \alpha - 1 = -1/2$, so that the Gauss–Jacobi quadrature rule reduces to a Gauss–Chebyshev rule (cf. also Example 2.45), for which there are closed formulas for the quadrature nodes and weights, see, e.g., [1, Chapter 22], so that they



method	time
quadrature	23.5 s
divided diff.	5.3 s
rational approx.	22.4 s

Figure 4.6: Approximating $\left((\Gamma_5 D_W)^2\right)^{-1/2}\Gamma_5 D_W \boldsymbol{b}$: Convergence history (left) and running times (right) of different restarting algorithms for computing the sign function of the Wilson–Dirac operator at zero chemical potential. The numbers next to the curves for the quadrature-based method indicate the number of quadrature nodes used for evaluating the error function in the corresponding restart cycle (due to the large number of restart cycles, we only give numbers in those iterations where the number of quadrature nodes has changed compared to the previous cycle). The restart length is $m = 20$ in all cases.

can be computed essentially for free. We report convergence curves and execution times for all three algorithms in Figure 4.6. We again observe monotone convergence of the quadrature and rational approximation based methods while the method based on divided differences begins to diverge starting already from the third restart cycle (so that the running time for this method is only given for the sake of completeness and is not really meaningful). Concerning running time, the two stable methods behave very similarly again, but one should keep in mind that the construction of the Zolotarev rational approximation requires the computation of the smallest and largest eigenvalue of A, which, using the MATLAB function `eigs`, takes about four and a half minutes in this experiment. This is no problem in realistic lattice QCD computations, as the sign function needs to be approximated many times in a single simulation, but it indicates that the independence from spectral information of our method can be a big advantage in situations in which the action of a matrix function on a single vector (or only a few vectors) needs to be approximated and the computation of eigenvalue information of A is costly. An interesting observation about the accuracy of the quadrature rules to be made in this experiment is that the number of quadrature nodes which is necessary for reaching the prescribed accuracy `tol` is (after a slight increase in the first few cycles) monotonically decreasing from one cycle to the next, making later restart cycles less computationally expensive than earlier cycles. This behavior is typical when approximating Stieltjes functions of Hermitian

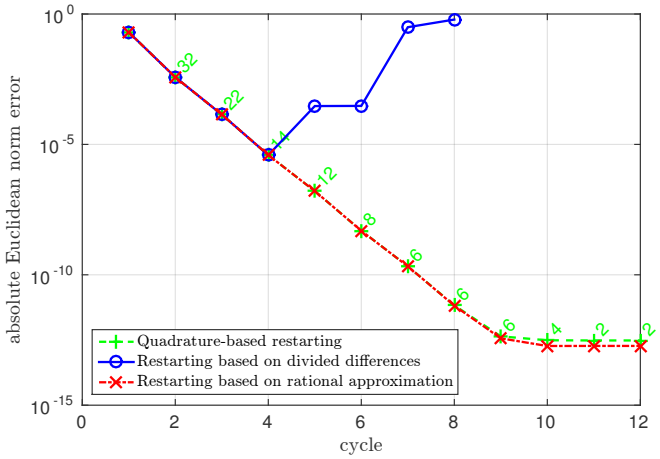


method	time
quadrature	34.5 s
acc. Hessenberg	804 s

Figure 4.7: Approximating $\left((\Gamma_5 D_W)^2\right)^{-1/2}\Gamma_5 D_W \boldsymbol{b}$: Convergence history (left) and running times (right) of different restarting algorithms for computing the sign function of the Wilson–Dirac operator at nonzero chemical potential. The numbers next to the curves for the quadrature-based method indicate the number of quadrature nodes used for evaluating the error function in the corresponding restart cycle (due to the large number of restart cycles, we only give numbers in those iterations where the number of quadrature nodes has changed compared to the previous cycle). The restart length is $m = 20$ in both cases.

positive definite matrices (see also the Gaussian Markov random field experiment later in this section and the discussion in [58]) and can be explained as follows. The order of magnitude of the computed corrections becomes smaller and smaller the longer the iteration goes on (see also the results on monotone convergence presented in Chapter 5) so that the *relative accuracy* that is required to reach the *absolute accuracy* `tol` is lower in later restart cycles. This is also in line with the error function representation (3.7), where the integrand contains the reciprocal of $\prod_{j=1}^{k-1} w_m^{(j)}$, a polynomial of degree $(k-1)m$ with roots in $\mathbb{R}_0^-$ in the kth restart cycle. The higher the degree of this polynomial gets, the closer the integrand is to the zero function (which still holds true after applying a variable transformation as in Section 4.3, as some term involving the reciprocal of $\prod_{j=1}^{k-1} w_m^{(j)}$ is always present).

Next, we turn our attention to the case of nonzero chemical potential, which corresponds to approximating the action of the inverse square root of a non-Hermitian matrix. Therefore, we cannot use the Zolotarev rational approximation here (as it is not accurate enough for eigenvalues which do not lie close to the real axis), and we thus compare our method to the approach of [43] again. The results of this experiment are given in Figure 4.7. The convergence behavior is very similar to the one observed for the Neuberger operator at zero chemical potential, but the running times reported on the right-hand side of Figure 4.7



method	time
quadrature	1.0 s
rational approx.	0.9 s
divided diff.	1.1 s

Figure 4.8: Approximating $A^{-1/2}\boldsymbol{z}$: Convergence history (left) and running times (right) of different restarting algorithms for sampling from a Gaussian Markov random field. The numbers next to the curves for the quadrature-based method indicate the number of quadrature nodes used for evaluating the error function in the corresponding restart cycle. The restart length is $m = 20$ in all cases.

show that in this case, as already observed in the wave equation model problem, the constant work per cycle that our quadrature-based method requires gives a very significant advantage over the approach from [43], which requires about 23 times as much time as our method.

To conclude the numerical experiments in this chapter, we investigate the problem of sampling from a Gaussian Markov random field. As we are again interested in approximating the action of the inverse square root of a Hermitian positive definite matrix on a vector, we compare the same methods and use the same rational approximation (but of degree 16, which is sufficient this time) and quadrature rule as for the (Hermitian) lattice QCD model problem. Again, the time consumed for computing the largest eigenvalue of A—the smallest one is explicitly known—for the construction of the Zolotarev rational approximation (which is about one second in this case) is not included in the timings reported on the right-hand side of Figure 4.8. Convergence curves and the number of quadrature nodes used by our algorithm are again shown on the left-hand side of the figure. All methods perform about the same in terms of running time, and the quadrature-based method and the method based on rational approximation again show the same convergence behavior, while the divided difference based method fails to reach an accuracy higher than 10^{-5}. As before, we observe that the number of quadrature nodes needed to reach the tolerance `tol` in our method decreases in later restart cycles. This experiment once again demonstrates that our algorithm exhibits the same stability and efficiency as the method of [3] with the advantage of being completely black-box (at least for Stieltjes and related functions).

CHAPTER 5

CONVERGENCE OF RESTARTED KRYLOV SUBSPACE METHODS

In this chapter we investigate the convergence behavior of the restarted Arnoldi method introduced in Chapter 4. We begin by shortly reviewing the few existing convergence results for restarted Krylov subspace methods for approximating $f(A)\boldsymbol{b}$ in Section 5.1 before developing a new approach for the convergence analysis of the restarted Arnoldi method for Stieltjes functions of Hermitian positive definite matrices in Section 5.2. In Section 5.3 we briefly comment on limitations of the applicability of our theory in case of non-Hermitian matrices and use this as a motivation for proposing a slight modification of Arnoldi's method in Section 5.4 for which we can extend our convergence analysis to the class of positive real matrices in Section 5.5. In Section 5.6 we give some results on a related topic, namely the arbitrary convergence behavior of Krylov subspace methods for non-Hermitian linear systems. We conclude the chapter by presenting some numerical experiments which illustrate the quality of the developed convergence bounds in Section 5.7.

5.1 Known convergence results

We begin by presenting previously known convergence results for restarted Krylov subspace methods for matrix functions. The one case in which convergence of the restarted Arnoldi method is well understood is when f is an entire function of order one; cf. [19].

Definition 5.1. The function f is called entire of order one if it is analytic in all $z \in \mathbb{C}$ and

$$\limsup_{r\to\infty} \frac{\log(\log(M(r)))}{\log r} = 1,$$

where $M(r) = \max_{|z|=r} |f(z)|$.

The most prominent example of an entire function of order one is the exponential $f(z) = e^z$ as clearly

$$\frac{\log(\log(M(r)))}{\log r} = 1 \text{ for all } r > 0$$

in this case. A convergence analysis for the restarted Arnoldi method for entire functions of order one was given in [43].

Theorem 5.2. *Let $A \in \mathbb{C}^{n\times n}$, let $\boldsymbol{b} \in \mathbb{C}^n$, let f be an entire function of order one and denote by $\boldsymbol{f}_m^{(k)}$ the approximation to $f(A)\boldsymbol{b}$ resulting from k cycles of the restarted Arnoldi method with restart length m. Then there exist constants C and γ independent of m and k such that*

$$\|f(A)\boldsymbol{b} - \boldsymbol{f}_m^{(k)}\|_2 \leq C\frac{\gamma^{km-1}}{(km-1)!}\|\boldsymbol{b}\|_2 \quad \textit{for all } k \geq 1.$$

Proof. See [43, Theorem 4.2 and Corollary 4.3]. □

In particular, Theorem 5.2 guarantees that the restarted Arnoldi method converges superlinearly to $f(A)\boldsymbol{b}$ for all restart lengths m when f is an entire function of order one. The proof of this result relies on convergence results for polynomials of best uniform approximation to entire functions of order one; see [50]. As similar results from approximation theory are not available for larger classes of functions (especially not for functions with singularities, such as Stieltjes functions), it seems difficult to generalize or extend the result of Theorem 5.2 to other functions.

There is one other known result from the literature, given in [2], which guarantees linear convergence of the restarted Arnoldi method and is applicable to a class of functions which contains the Stieltjes functions, but it is restricted to the case of restart length $m = 1$. This can be interpreted as a generalization of the method of steepest descent for matrix functions, but is seldom used in practice.

Theorem 5.3. *Let $A \in \mathbb{C}^{n\times n}$ be Hermitian positive definite, let $\boldsymbol{b} \in \mathbb{C}^n$, and let $\lambda_{\min}$ and $\lambda_{\max}$ denote the smallest and largest eigenvalue of A, respectively. Let f be a function analytic in $(\mathbb{C} \setminus \mathbb{R}) \cup [\lambda_{\min}, \lambda_{\max}]$. Then the restarted Arnoldi method with restart length $m = 1$ converges to $f(A)\boldsymbol{b}$ with asymptotic convergence factor at least*

$$\frac{\lambda_{\max} - \lambda_{\min}}{|\zeta - \lambda_{\max}| + |\zeta - \lambda_{\min}|},$$

where ζ is a singularity of f which is closest to $[\lambda_{\min}, \lambda_{\max}]$.

Proof. See [2, Corollary 5.5]. □

The technique of proof used to derive this result in [2] is different from the one used to prove Theorem 5.2 in [43]. It relies on the fact, also proved in [2], that the sequence of Ritz values generated by the restarted Arnoldi method with $m = 1$ asymptotically alternates between only two values θ_1 and θ_2, which allows to asymptotically characterize the corresponding Arnoldi approximations as resulting from a simple interpolation process with only two nodes, a situation analyzed, e.g., in [140]. While it holds true for larger restart lengths m that the sequence of Ritz values generated by the restarted Arnoldi method asymptotically alternates between two sets of m values, there are several other tools used in the proof of Theorem 5.3 for which a generalization to $m > 1$ is currently unknown, so that at least no straightforward generalization of the result of Theorem 5.3 is available.

In the next section, we therefore derive convergence results for Stieltjes matrix functions and arbitrary restart length $m \geq 1$ using a different technique which exploits the intimate relation between the restarted Arnoldi method for $f(A)\boldsymbol{b}$ and restarted FOM for shifted linear systems.

5.2 Convergence of restarted Arnoldi for Stieltjes functions

In this section, we prove convergence of the restarted Arnoldi method for Stieltjes functions of Hermitian positive definite matrices. We already published these results in [57]. We begin by pointing out a different interpretation of the error function representation (3.8) which is crucial for the following analysis. In the following, let $\boldsymbol{x}_m(t)$ denote the mth FOM iterate (2.27) for the shifted linear system

$$(A + tI)\boldsymbol{x}(t) = \boldsymbol{b} \tag{5.1}$$

with initial guess $\boldsymbol{x}_0(t) = \mathbf{0}$. We assume from here on that $t \geq 0$ and $\operatorname{spec}(A) \subset \mathbb{C} \setminus \mathbb{R}_0^-$, so that each shifted matrix $A + tI$ is nonsingular and therefore each

system (5.1) has a unique solution $\boldsymbol{x}^*(t)$. The residuals $\boldsymbol{r}_m(t) = \boldsymbol{b} - (A+tI)\boldsymbol{x}_m(t)$ corresponding to the FOM iterates $\boldsymbol{x}_m(t)$ then satisfy

$$\boldsymbol{r}_m(t) = \frac{(-1)^{m+1}\|\boldsymbol{b}\|_2\gamma_m}{w_m(t)}\boldsymbol{v}_{m+1}. \tag{5.2}$$

with $w_m(t) = \prod_{i=1}^m (t+\theta_i)$ and $\gamma_m = \prod_{i=1}^m h_{i+1,i}$, which is merely a rewritten and shifted version of the representation (2.29) for the FOM residual.

Now consider the mth Arnoldi approximation $\boldsymbol{f}_m$ for f a Stieltjes function (3.15), which can be rewritten as

$$\boldsymbol{f}_m = \|\boldsymbol{b}\|_2 V_m f(H_m)\hat{\boldsymbol{e}}_1 = \int_0^\infty \|\boldsymbol{b}\|_2 V_m (H_m + tI)^{-1}\hat{\boldsymbol{e}}_1 \,\mathrm{d}\mu(t) = \int_0^\infty \boldsymbol{x}_m(t)\,\mathrm{d}\mu(t). \tag{5.3}$$

Combining the error representation (3.7) for $f(A)\boldsymbol{b} - \boldsymbol{f}_m$ with (5.2) similarly gives

$$f(A)\boldsymbol{b} - \boldsymbol{f}_m = e_m(A)\boldsymbol{v}_{m+1} = \int_0^\infty (A+tI)^{-1}\boldsymbol{r}_m(t)\,\mathrm{d}\mu(t) = \int_0^\infty \boldsymbol{e}_m(t)\,\mathrm{d}\mu(t), \tag{5.4}$$

where $\boldsymbol{e}_m(t) = \boldsymbol{x}^*(t) - \boldsymbol{x}_m(t)$ is the error of the mth FOM iterate for the system $(A+tI)\boldsymbol{x}(t) = \boldsymbol{b}$. We note that a similar result is known for analytic functions using the Cauchy integral representation and was observed, e.g., in [44, 88, 114]. Equations (5.3) and (5.4) allow to interprete performing Arnoldi's method for $f(A)\boldsymbol{b}$ as implicitly applying FOM to the shifted linear systems (5.1) for all values $t \geq 0$ and integrating the corresponding approximations when f is a Stieltjes function. Consequently, the error of the Arnoldi approximation $\boldsymbol{f}_m$ is the integral over the errors for all these linear systems.

Although (5.4) already reveals the relation between approximating $f(A)\boldsymbol{b}$ by Arnoldi's method and the solution of shifted linear systems by FOM, we need to generalize this representation to the case of the respective restarted methods to be able to use it in our convergence analysis. To do so, we follow a similar analysis of the restarted Arnoldi approximation in case of analytic functions represented by the Cauchy integral formula performed in [44]. Recalling the definition (2.31) of the restarted FOM approximation and applying this to the shifted linear systems (5.1), we have

$$\boldsymbol{x}_m^{(k+1)}(t) = \boldsymbol{x}_m^{(k)}(t) + \boldsymbol{e}_m^{(k)}(t) \text{ with } \boldsymbol{e}_m^{(k)}(t) = \|\boldsymbol{b}\|_2 V_m^{(k)}(H_m^{(k)} + tI)^{-1}\hat{\boldsymbol{e}}_1. \tag{5.5}$$

Inductively applying (5.2) to (5.5), we find that the residuals of the restarted shifted FOM iterates satisfy

$$\boldsymbol{r}_m^{(k)}(t) = (-1)^{k(m+1)}\|\boldsymbol{b}\|_2 \frac{\prod_{j=1}^k \gamma_m^{(j)}}{\prod_{j=1}^k w_m^{(j)}(t)}\boldsymbol{v}_{m+1}^{(k)}. \tag{5.6}$$

Using (4.10) and (4.11) together with (5.6) then gives the representation

$$e_m^{(k)}(A)\boldsymbol{v}_{m+1}^{(k)} = \int_0^\infty (A+tI)^{-1}\boldsymbol{r}_m^{(k)}(t)\,\mathrm{d}\mu(t) = \int_0^\infty \boldsymbol{e}_m^{(k)}(t)\,\mathrm{d}\mu(t). \tag{5.7}$$

for the error of the restarted Arnoldi approximation. This representation will be the basis of our convergence analysis, as it allows to transfer known results on the linear system errors $\boldsymbol{e}_m^{(k)}(t)$ to the matrix function setting.

For the remainder of this section, we restrict ourselves to Hermitian positive definite matrices A, as FOM reduces to the conjugate gradient method in this case and we can use Theorem 2.32 for bounding the norm of the right-hand side of (5.7). The following results are therefore based on convergence results for *restarted CG*, a method which one would under normal circumstances not use in practice, as the unrestarted CG method, Algorithm 2.4, only needs constant storage and computational work per iteration and restarting therefore has no benefit. It is nonetheless a convergent method and therefore suited for building the basis of the analysis to come.

Before proceeding, we summarize a few obvious but important facts about the shifted linear systems (5.1).

Proposition 5.4. *Let $A \in \mathbb{C}^{n\times n}$ be Hermitian positive definite and let $\lambda_{\min}$ and $\lambda_{\max}$ denote its smallest and largest eigenvalue respectively. Then*

1. *the matrix $A + tI$ is Hermitian positive definite for all $t \geq 0$,*
2. *the condition number of $A + tI$ is $\kappa(t) = \frac{\lambda_{\max}+t}{\lambda_{\min}+t}$.*

With these prerequisites we are now in a position to derive a first bound for the (energy) norm of the error of $\boldsymbol{f}_m^{(k)}$.

Lemma 5.5. *Let $A \in \mathbb{C}^{n\times n}$ be Hermitian positive definite, let $\boldsymbol{b} \in \mathbb{C}^n$, let f be a Stieltjes function of the form (3.15) and let $\boldsymbol{f}_m^{(k)}$ be the approximation to $f(A)\boldsymbol{b}$ from k cycles of the restarted Arnoldi method with restart length m. Let $\lambda_{\min}$ and $\lambda_{\max}$ denote the smallest and largest eigenvalue of A, respectively, and define the functions*

$$\kappa(t) = \frac{\lambda_{\max}+t}{\lambda_{\min}+t},\ c(t) = \frac{\sqrt{\kappa(t)}-1}{\sqrt{\kappa(t)}+1},\ \text{ and } \alpha_m(t) = \frac{1}{\cosh(m\ln c(t))}. \tag{5.8}$$

The energy norm of the error of $\boldsymbol{f}_m^{(k)}$ is then bounded by

$$\|f(A)\boldsymbol{b} - \boldsymbol{f}_m^{(k)}\|_A \leq \|\boldsymbol{b}\|_2\sqrt{\lambda_{\max}}\int_0^\infty \frac{\alpha_m(t)^k}{\sqrt{\lambda_{\min}+t}\cdot\sqrt{\lambda_{\max}+t}}\,\mathrm{d}\mu(t). \tag{5.9}$$

Proof. We first note that at this stage we allow ∞ as a value for the integral on the right-hand side of (5.9). Its finiteness and convergence to zero for $k \to \infty$ will be discussed at a later point. By using (5.7), we can write

$$f(A)\boldsymbol{b} - \boldsymbol{f}_m^{(k)} = \int_0^\infty \boldsymbol{e}_m^{(k)}(t)\,\mathrm{d}\mu(t), \tag{5.10}$$

where $\boldsymbol{e}_m^{(k)}(t)$ denotes the error of the approximation $\boldsymbol{x}_m^{(k)}(t)$ from k cycles of restarted CG with restart length m for the shifted linear system $(A+tI)\boldsymbol{x}(t) = \boldsymbol{b}$. Taking the energy norm on both sides of (5.10) and using Lemma 2.12 gives

$$\begin{aligned}\|f(A)\boldsymbol{b} - \boldsymbol{f}_m^{(k)}\|_A &\leq \int_0^\infty \|\boldsymbol{e}_m^{(k)}(t)\|_A\,\mathrm{d}\mu(t) \\ &\leq \int_0^\infty \frac{\sqrt{\lambda_{\max}}}{\sqrt{\lambda_{\max}+t}}\|\boldsymbol{e}_m^{(k)}(t)\|_{A+tI}\,\mathrm{d}\mu(t),\end{aligned}$$

where we used that $\|\boldsymbol{v}\|_A \leq \sqrt{\lambda_{\max}/(\lambda_{\max}+t)}\|\boldsymbol{v}\|_{A+tI}$ holds for all $t \geq 0$ since $\boldsymbol{v}^H(A+tI)\boldsymbol{v} = \boldsymbol{v}^HA\boldsymbol{v}+t\boldsymbol{v}^H\boldsymbol{v}$ and $\boldsymbol{v}^HA\boldsymbol{v} \leq \lambda_{\max}\boldsymbol{v}^H\boldsymbol{v}$. According to Proposition 5.4 we can now apply Theorem 2.32 for the shifted matrices $A + tI$, where $\alpha_m(t)$ from (5.8) is exactly the factor from the CG convergence bound for $A+tI$. Using the fact that the kth cycle of restarted CG can be interpreted as performing m iterations of CG with the approximation $\boldsymbol{x}_m^{(k-1)}(t)$ from the previous cycle as initial guess, we obtain

$$\begin{aligned}\|f(A)\boldsymbol{b} - \boldsymbol{f}_m^{(k)}\|_A &\leq \sqrt{\lambda_{\max}}\int_0^\infty \frac{\alpha_m(t)}{\sqrt{\lambda_{\max}+t}}\|\boldsymbol{x}^*(t) - \boldsymbol{x}_m^{(k-1)}(t)\|_{A+tI}\,\mathrm{d}\mu(t) \\ &= \sqrt{\lambda_{\max}}\int_0^\infty \frac{\alpha_m(t)}{\sqrt{\lambda_{\max}+t}}\|\boldsymbol{e}_m^{(k-1)}(t)\|_{A+tI}\,\mathrm{d}\mu(t),\end{aligned}$$

with $\alpha_m(t)$ from (5.8). Repeatedly applying the CG estimate for all t throughout all restart cycles and using the fact that the initial guess of the first restart cycle is $\boldsymbol{x}_0(t) = \boldsymbol{0}$ for all t, we conclude that

$$\|f(A)\boldsymbol{b} - \boldsymbol{f}_m^{(k)}\|_A \leq \sqrt{\lambda_{\max}}\int_0^\infty \frac{\alpha_m(t)^k}{\sqrt{\lambda_{\max}+t}}\|\boldsymbol{x}^*(t)\|_{A+tI}\,\mathrm{d}\mu(t). \tag{5.11}$$

As $\boldsymbol{x}^*(t) = (A+tI)^{-1}\boldsymbol{b}$, a straightforward calculation shows that

$$\|\boldsymbol{x}^*(t)\|_{A+tI} \leq \frac{\|\boldsymbol{b}\|_2}{\sqrt{\lambda_{\min}+t}}. \tag{5.12}$$

Inserting (5.12) into (5.11) completes the proof. □

As mentioned at the beginning of the proof of Lemma 5.5, it is not immediately clear whether the integral on the right-hand side of (5.9) has a finite value. Therefore, this upper bound by itself is of no great use. Using the following result on the monotonicity of the function $\alpha_m(t)$ from (5.8) will allow us to derive an upper bound which is finite and goes to zero as $k \to \infty$.

Proposition 5.6. *The function $\alpha_m(t)$ from (5.8) is monotonically decreasing on $\mathbb{R}_0^+$.*

Proof. As a function of $t \in \mathbb{R}_0^+$, κ from (5.8) decreases monotonically from $\kappa(0)$ to 1, c increases monotonically from $c(\kappa(0))$ to 1 as a function of $\kappa \in [\kappa(0), \infty)$, and α_m increases monotonically as a function of $c \in [c(\kappa(0)), 1)$. Altogether, thus, α_m decreases monotonically as a function of t. □

We continue with the main result of this section.

Theorem 5.7. *Let $A \in \mathbb{C}^{n\times n}$ be Hermitian positive definite, let $\boldsymbol{b} \in \mathbb{C}^n$, let f be a Stieltjes function of the form (3.15), and let $\boldsymbol{f}_m^{(k)}$ be the approximation from k cycles of Arnoldi's method with restart length m. Further, let $\alpha_m(t)$ be defined as in (5.8) and let $t_0 \geq 0$ be the left endpoint of the support of μ. The energy norm of the error of $\boldsymbol{f}_m^{(k)}$ can then be bounded as*

$$\|f(A)\boldsymbol{b} - \boldsymbol{f}_m^{(k)}\|_A \leq C\alpha_m(t_0)^k, \tag{5.13}$$

where

$$C = \|\boldsymbol{b}\|_2\sqrt{\lambda_{\max}} \cdot f\big(\sqrt{\lambda_{\min}\lambda_{\max}}\big) \tag{5.14}$$

is a constant independent of m and k, and $0 \leq \alpha_m(t_0) < 1$. In particular, the restarted Arnoldi method converges for all restart lengths $m \geq 1$.

Proof. We begin by using Lemma 5.5 and Proposition 5.6 to estimate

$$\begin{aligned}\|f(A)\boldsymbol{b} - \boldsymbol{f}_m^{(k)}\|_A &\leq \|\boldsymbol{b}\|_2\sqrt{\lambda_{\max}}\int_0^\infty \frac{\alpha_m(t)^k}{\sqrt{\lambda_{\min}+t}\cdot\sqrt{\lambda_{\max}+t}}\,\mathrm{d}\mu(t)\\ &\leq \|\boldsymbol{b}\|_2\alpha_m(t_0)^k\sqrt{\lambda_{\max}}\int_0^\infty \frac{1}{\sqrt{\lambda_{\min}+t}\cdot\sqrt{\lambda_{\max}+t}}\,\mathrm{d}\mu(t). \end{aligned}\tag{5.15}$$

Due to the inequality

$$\sqrt{\lambda_{\min}\lambda_{\max}} \leq \frac{1}{2}(\lambda_{\min}+\lambda_{\max})$$

for the geometric and arithmetic mean, we have

$$\begin{aligned}\sqrt{\lambda_{\min}+t}\cdot\sqrt{\lambda_{\max}+t} &= \sqrt{\lambda_{\min}\lambda_{\max} + (\lambda_{\min}+\lambda_{\max})t + t^2}\\ &\geq \sqrt{\lambda_{\min}\lambda_{\max} + 2\sqrt{\lambda_{\min}\lambda_{\max}}t + t^2}\\ &= \sqrt{\lambda_{\min}\lambda_{\max}} + t.\end{aligned}$$

Therefore,

$$\frac{1}{\sqrt{\lambda_{\min}+t}\cdot\sqrt{\lambda_{\max}+t}} \leq \frac{1}{\sqrt{\lambda_{\min}\lambda_{\max}}+t}. \tag{5.16}$$

Inserting (5.16) into (5.15), we obtain

$$\|f(A)\boldsymbol{b} - \boldsymbol{f}_m^{(k)}\|_A \leq \|\boldsymbol{b}\|_2 \alpha_m(t_0)^k \sqrt{\lambda_{\max}} \int_0^\infty \frac{1}{\sqrt{\lambda_{\min}\lambda_{\max}} + t} \, \mathrm{d}\mu(t). \tag{5.17}$$

The integral on the right-hand side of (5.17) is $f(\sqrt{\lambda_{\min}\lambda_{\max}})$, which completes the proof. □

Theorem 5.7 proves that the restarted Arnoldi method for Hermitian positive definite A and f a Stieltjes function converges to $f(A)\boldsymbol{b}$ for all restart lengths $m \geq 1$. This qualitative statement does of course not depend on the norm in which the error is measured, but as one is typically interested in the Euclidean norm of the error when approximating matrix functions, we give another error bound for this norm before proceeding.

Of course, as long as one only uses the equivalence of norms on $\mathbb{C}^n$, only the constant in front of the convergence factor $\alpha_m(t_0)$ changes when switching from one norm to another.

Corollary 5.8. *Let the assumptions of Theorem 5.7 hold. The Euclidean norm of the error of $\boldsymbol{f}_m^{(k)}$ can then be bounded as*

$$\|f(A)\boldsymbol{b} - \boldsymbol{f}_m^{(k)}\|_2 \leq \widetilde{C}\alpha_m(t_0)^k,$$

where

$$\widetilde{C} = \|\boldsymbol{b}\|_2 \sqrt{\kappa(0)} f(\sqrt{\lambda_{\min}\lambda_{\max}}).$$

Proof. For all $\boldsymbol{v} \in \mathbb{C}^n$ one has $\|\boldsymbol{v}\|_2 \leq \frac{1}{\lambda_{\min}}\|\boldsymbol{v}\|_A$. Inserting this into (5.13) and noting that $\kappa(0) = \lambda_{\max}/\lambda_{\min}$ concludes the proof. □

It is interesting to investigate two special cases of the bound (5.13), namely the "extremal" cases of restart length $m = 1$ and restart length $m = n$ (i.e., the unrestarted Arnoldi method). For the case $m = 1$, the convergence factor is given by

$$\alpha_1(t_0) = \frac{1}{\cosh(\ln c(t_0))}. \tag{5.18}$$

Using the definition of the hyperbolic cosine and the definition of $c(t)$ from (5.8), we find

$$\begin{aligned} \cosh(\ln c(t_0)) &= \frac{1}{2}\left(e^{\ln c(t_0)} + e^{-\ln c(t_0)}\right) \\ &= \frac{1}{2}\left(c(t_0) + \frac{1}{c(t_0)}\right) \\ &= \frac{1}{2} \cdot \frac{2\kappa(t_0) + 2}{\kappa(t_0) - 1}. \end{aligned} \tag{5.19}$$

Inserting (5.19) into (5.18) together with the definition of $\kappa(t)$ then gives

$$\alpha_1(t_0) = \frac{\lambda_{\max} - \lambda_{\min}}{\lambda_{\max} + \lambda_{\min} + 2t_0} = \frac{\lambda_{\max} - \lambda_{\min}}{|-t_0 - \lambda_{\max}| + |-t_0 - \lambda_{\min}|}.$$

For f a Stieltjes function and A Hermitian positive definite, the singularity of f closest to $[\lambda_{\min}, \lambda_{\max}]$ is clearly $\zeta = -t_0$, so that we exactly recover the asymptotic convergence factor from Theorem 5.3. This is indeed interesting as two completely different techniques of proof were used to derive these results.

The other special case of our bound which we study in more detail is the unrestarted Arnoldi method. Of course, due to the finite termination property of Arnoldi's method (cf. Section 2.3), the method will (at least in exact arithmetic) terminate after at most n steps. It is still interesting to have an estimate for the energy norm of the error throughout the iterations of the method. The modification necessary to obtain a bound for the Euclidean norm is obvious, so we forego stating it here.

Corollary 5.9. *Let the assumptions of Theorem 5.7 hold and let $\boldsymbol{f}_m$ be the approximation to $f(A)\boldsymbol{b}$ after m iterations of the unrestarted Arnoldi method. The energy norm of the error of $\boldsymbol{f}_m$ can then be bounded as*

$$\|f(A)\boldsymbol{b} - \boldsymbol{f}_m\|_A \leq C\alpha_m(t_0), \tag{5.20}$$

where C is the constant from (5.14).

Proof. The result directly follows by taking $k = 1$ in Theorem 5.7. □

Ignoring the constant factor C, the bound (5.20) is the same bound as the standard bound for the energy norm of the error in the CG method for the shifted system $(A + t_0 I)\boldsymbol{x}(t_0) = \boldsymbol{b}$, cf. Theorem 2.32. This is not necessarily a surprise, as the discussion so far already revealed that convergence of Arnoldi's method for Hermitian positive definite A is closely tied to convergence of CG for shifted linear systems. As the shifted matrices become more and more well-conditioned for growing t (cf. also the proof of Proposition 5.6), the system with smallest shift t_0 is the worst-conditioned of all systems and can therefore be expected to be the one dominating the convergence behavior of the method.

We stress here that all bounds presented so far, in particular (5.20), do not take into account superlinear convergence effects observed in later iterations of CG due to spectral adaption; see [7, 12, 13]. For the restarted Arnoldi method, this is not really relevant in practical situations, as these effects typically only take place in the unrestarted method or if the restart length m is rather large compared to the matrix size n, a fact which we will further comment on in the numerical experiments reported in Section 5.7. Later in this chapter, in Theorem 5.21, we

also present a result which accounts for superlinear convergence effects, but needs techniques of proof which are different from the ones used so far.

In the numerical experiments reported in Section 4.4, we observed that the (Euclidean) norm of the error was monotonically decreasing in cases where we approximated a Stieltjes function of a Hermitian positive definite matrix. While Theorem 5.7 guarantees that the norm of the error in the restarted Arnoldi method converges to zero for all restart lengths, it does not make any statements about monotonicity. The following result from [55] (see also [36]) guarantees monotone convergence of the standard, unrestarted Arnoldi method for approximating Stieltjes matrix functions.

Theorem 5.10. *Let $A \in \mathbb{C}^{n\times n}$ be Hermitian positive definite, let $\boldsymbol{b} \in \mathbb{C}^n$, let f be a Stieltjes function, and let $\boldsymbol{f}_m$ denote the approximation for $f(A)\boldsymbol{b}$ obtained by m iterations of Arnoldi's method. Then*

$$\|f(A)\boldsymbol{b} - \boldsymbol{f}_{m+1}\|_2 \leq \|f(A)\boldsymbol{b} - \boldsymbol{f}_m\|_2 \quad \text{for all } m \geq 1,$$

i.e., the Euclidean norm of the error decreases monotonically.

We already have all tools at hand to easily transfer the result of Theorem 5.10 to the restarted case.

Corollary 5.11. *Under the assumptions of Theorem 5.10, the approximations $\boldsymbol{f}_m^{(k)}$ obtained via the restarted Arnoldi method satisfy*

$$\|f(A)\boldsymbol{b} - \boldsymbol{f}_m^{(k+1)}\|_2 \leq \|f(A)\boldsymbol{b} - \boldsymbol{f}_m^{(k)}\|_2 \quad \text{for all } k \geq 1.$$

Proof. The approximation $\boldsymbol{f}_m^{(k+1)}$ from the $(k+1)$st Arnoldi cycle can be written as

$$\boldsymbol{f}_m^{(k+1)} = \boldsymbol{f}_m^{(k)} + \boldsymbol{d}_m^{(k)},$$

where $\boldsymbol{d}_m^{(k)}$ is the approximation obtained by applying m steps of Arnoldi's method for approximating $e_m^{(k)}(A)\boldsymbol{v}_{m+1}^{(k)}$. As the error function $e_m^{(k)}(z)$ is again a (multiple of a) Stieltjes function according to Proposition 3.9, we can apply Theorem 5.10 and find the desired result. □

Another obvious extension of the results presented so far is the transfer to functions of the type $\widetilde{f}(z) = zf(z)$ for f a Stieltjes function, as already considered in Corollary 3.6. Using the error representation from Corollary 3.6, we can easily derive an error bound similar to the one from Theorem 5.7 for the restarted corrected Arnoldi approximation. Note that this bound is not sharp and does not reflect the advantage of directly working with $\widetilde{f}$ as mentioned in the discussion preceding Corollary 3.6.

Theorem 5.12. *Let $A \in \mathbb{C}^{n\times n}$ be Hermitian positive definite, let $\boldsymbol{b} \in \mathbb{C}^n$ and let $\widetilde{f}(z) = zf(z)$ where f is a Stieltjes function as in (3.15). Let $\hat{\boldsymbol{f}}_m^{(k)}$ be the corrected approximation from k cycles of the restarted Arnoldi method with restart length m for $\widetilde{f}(A)\boldsymbol{b}$, i.e.,*

$$\hat{\boldsymbol{f}}_m^{(k)} = \widetilde{\boldsymbol{f}}_m^{(k)} + h_{m+1,m}^{(k)} \left(\hat{\boldsymbol{e}}_m^H e_m^{(k-1)}(H_m^{(k)})\hat{\boldsymbol{e}}_1\right) \boldsymbol{v}_{m+1}^{(k)},$$

where $\widetilde{\boldsymbol{f}}_m^{(k)}$ denotes the Arnoldi approximation for the error $\widetilde{e}_m^{(k-1)}(A)\boldsymbol{v}_{m+1}^{(k-1)} = Ae_m^{(k-1)}(A)\boldsymbol{v}_{m+1}^{(k-1)} = \widetilde{f}(A)\boldsymbol{b} - \hat{\boldsymbol{f}}_m^{(k-1)}$ (starting with $\hat{\boldsymbol{f}}_m^{(0)} = \boldsymbol{0}$, i.e., $e_m^{(0)}(z) = f(z)$). Further, let $\alpha_m(t)$ be defined as in (5.8), and let $t_0 \geq 0$ be the left endpoint of the support of μ. Then

$$\|\widetilde{f}(A)\boldsymbol{b} - \hat{\boldsymbol{f}}_m^{(k)}\|_A \leq \lambda_{\max} C \alpha_m(t_0)^k$$

and

$$\|\widetilde{f}(A)\boldsymbol{b} - \hat{\boldsymbol{f}}_m^{(k)}\|_2 \leq \lambda_{\max} \widetilde{C} \alpha_m(t_0)^k,$$

where C and $\widetilde{C}$ are the constants from Theorem 5.7 and Corollary 5.8, respectively. In particular, the restarted corrected Arnoldi method for $\widetilde{f}(z) = zf(z)$ converges for all restart lengths $m \geq 1$.

5.3 Limitations for non-Hermitian matrices

In the last section, we proved convergence of the restarted Arnoldi method for Stieltjes functions of Hermitian matrices. A natural question is of course whether these results are generalizable to larger classes of matrices. To this end, we first note that it is sensible to require the field of values $\mathcal{W}(A)$ of A to lie in the right half-plane (i.e., that A is *positive real*), as any convergence proof must guarantee that $\mathcal{W}(A) \cap \mathbb{R}_0^- = \emptyset$, as otherwise it can happen that a Ritz value occurs on $\mathbb{R}_0^-$ and the restarted Arnoldi approximations are not even defined. Therefore, a reasonable choice for the next larger class of matrices (containing the class of Hermitian positive definite matrices), are normal matrices with field of values in the right half-plane. One can, however, construct matrices which belong to this class but for which the restarted Arnoldi method fails to converge, showing that a generalization of Theorem 5.7 seems impossible for (meaningful) larger classes of matrices. We illustrate this by investigating the matrix

$$A = \begin{bmatrix} \alpha & 0 & \cdots & 0 & 1 \\ 1 & \alpha & 0 & \cdots & 0 \\ 0 & 1 & \alpha & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & 1 & \alpha \end{bmatrix} \in \mathbb{R}^{n\times n}, \tag{5.21}$$

where n is odd and $\alpha \in \mathbb{R}$ is a real parameter. Some properties of this matrix are summarized in Proposition 5.13.

Proposition 5.13. *Let A be the matrix from (5.21) where n is odd and $\alpha \in \mathbb{R}$ is arbitrary. Then*

(i) A is normal and

(ii) the eigenvalues of A are $\lambda_k = \alpha + e^{2\pi i k/n}, k = 1, \ldots, n$.

Proof. A straightforward calculation shows that

$$A^H A = \begin{bmatrix} 2\alpha & \alpha & 0 & \cdots & 0 & \alpha \\ \alpha & 2\alpha & \alpha & \ddots & \cdots & 0 \\ 0 & \alpha & 2\alpha & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & \ddots & \ddots & \ddots & 2\alpha & \alpha \\ \alpha & 0 & \cdots & 0 & \alpha & 2\alpha \end{bmatrix} = AA^H$$

so that A is normal. For part (ii), we observe that the characteristic polynomial of A is given by

$$\chi_A(\lambda) = (\lambda - \alpha)^n - 1,$$

such that its roots λ_k must be nth roots of unity shifted by α. This proves the result. □

Example 5.14. Consider a matrix A of the form (5.21) where $n = 21$ and $\alpha = 0.995$. According to Proposition 5.13(i), A is normal, such that $\mathcal{W}(A)$ is the convex hull of its eigenvalues, which by Proposition 5.13 are the nth roots of unity shifted by α. The smallest real part among $e^{2\pi i k/21}, k = 1, \ldots, 21$ is $\cos(22\pi/21) > -0.995$, such that the real parts of all eigenvalues of A are positive and $\mathcal{W}(A)$ lies in the right half-plane.

When approximating the action of the Stieltjes matrix function $A^{-1/2}$ on the first canonical unit vector $\hat{\boldsymbol{e}}_1$ with the restarted Arnoldi method with restart length $m = 10$, we observe that the method diverges (after a short initial phase in which the norm of the error is reduced), cf. Figure 5.1.

A thorough explanation of the behavior observed in Example 5.14 will be given in Section 5.6, in which the possible convergence curves of restarted Krylov subspace methods for linear systems with matrices with sparsity pattern as in (5.21) are

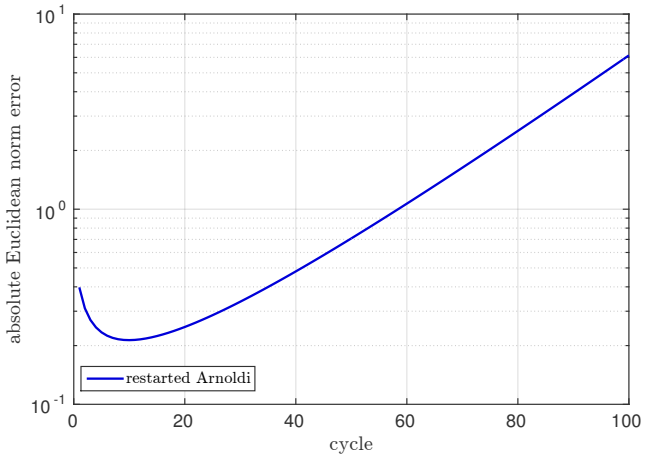


Figure 5.1: Norms of the error of the iterates produced by the restarted Arnoldi method with restart length $m = 10$ applied to the matrix A from (5.21) with $n = 21$ and $\alpha = 0.995$ for approximating $A^{-1/2}\hat{e}_1$.

investigated in detail. This will in turn allow to understand the behavior of restarted Krylov subspace methods for approximating Stieltjes functions of such matrices by studying the underlying shifted linear systems, cf. Remark 5.25. For the time being, we will skip this analysis and just use Example 5.14 for showing that there indeed exist normal, positive real matrices for which Arnoldi's method for approximating Stieltjes functions diverges. This motivates to consider a slight modification of the method in the next section, for which convergence for this class of matrices can be guaranteed (in fact, it can even be guaranteed without requiring normality).

5.4 The restarted harmonic Arnoldi method

In the preceding section, we illustrated that one cannot expect the convergence results proven for the restarted Arnoldi method for approximating Stieltjes matrix functions to hold for larger classes of matrices than Hermitian positive definite ones. Considering Corollary 2.37 which guarantees that the restarted GMRES method for linear systems converges if A is positive real, it is a natural approach to try to find a generalization of restarted GMRES for Stieltjes matrix functions, hoping that convergence results transfer to the matrix function case. In the same way, the restarted Arnoldi method for Hermitian positive definite A can be seen as a matrix function analogue of restarted CG. In light of Lemma 2.34 a sensible approach to reach this goal is to use a variant of Arnoldi's method in which the approximation is defined by the polynomial interpolating f at the harmonic Ritz

values corresponding to the Krylov subspace $\mathcal{K}_m(A, \boldsymbol{b})$ instead of the standard Ritz values. The resulting approximation, which we will call *(restarted) harmonic Arnoldi approximation* here and in the following, was already considered in the context of approximating matrix functions (albeit in unrestarted methods only and without presenting a convergence analysis) in [48, 87]. Our analysis of this method in the restarted context presented in this and the subsequent section was already published in [57].

By combining Lemma 2.27 and Proposition 2.35, it is immediately clear that the harmonic Arnoldi approximation for $f(A)\boldsymbol{b}$ can be computed as

$$\widetilde{\boldsymbol{f}}_m = \|\boldsymbol{b}\|_2 V_m f\left(H_m + (h_{m+1,m} H_m^{-1} \hat{\boldsymbol{e}}_m)\hat{\boldsymbol{e}}_m^H\right)\hat{\boldsymbol{e}}_1 = \|\boldsymbol{b}\|_2 V_m f(\widetilde{H}_m)\hat{\boldsymbol{e}}_1, \tag{5.22}$$

provided that H_m is nonsingular. When f is a Stieltjes function of the form (3.15), the harmonic Arnoldi approximation (5.22) can be rewritten as

$$\widetilde{\boldsymbol{f}}_m = \int_0^\infty \|\boldsymbol{b}\|_2 V_m (\widetilde{H}_m + tI)^{-1}\hat{\boldsymbol{e}}_1 \,\mathrm{d}\mu(t) =: \int_0^\infty \widetilde{\boldsymbol{x}}_m(t)\,\mathrm{d}\mu(t). \tag{5.23}$$

We note straightaway, to avoid confusion, that while $\widetilde{\boldsymbol{x}}_m(0)$ is the mth GMRES iterate for the system $A\boldsymbol{x} = \boldsymbol{b}$, the other vectors $\widetilde{\boldsymbol{x}}_m(t)$ for $t > 0$ are *not* the GMRES iterates for $(A+tI)\boldsymbol{x}(t) = \boldsymbol{b}$. This is due to the fact that the eigenvalues of $\widetilde{H}_m + tI$ are not the harmonic Ritz values of $A+tI$ corresponding to $\mathcal{K}_m(A, \boldsymbol{b})$. We will in the following show that this does not hinder a convergence analysis for the resulting method, and reveal connections to the shifted GMRES method from [56]. The following result shows that the residuals corresponding to the vectors $\widetilde{\boldsymbol{x}}_m(t)$ are collinear.

Lemma 5.15. *Let $A \in \mathbb{C}^{n\times n}$, let $\boldsymbol{b} \in \mathbb{C}^n$ and let V_m, H_m be the matrices from the Arnoldi decomposition* (2.23) *for A and $\boldsymbol{b}$ and let $\widetilde{H}_m$ be defined as in* (2.35) *and $\widetilde{\boldsymbol{x}}_m(t)$ as in* (5.23)*. For $f(z) = z^{-1}$, let $q_{\widetilde{H}_m,()^{-1}}$ be the polynomial interpolating f at $\operatorname{spec}(\widetilde{H}_m)$, and let $p_m(z) = 1 - zq_{\widetilde{H}_m,()^{-1}}(z)$. Then*

$$\widetilde{\boldsymbol{r}}_m(t) := \boldsymbol{b} - (A + tI)\widetilde{\boldsymbol{x}}_m(t) = \eta_m(t)\widetilde{\boldsymbol{r}}_m(0),$$

where

$$\eta_m(t) = \frac{1}{p_m(-t)}, \quad \widetilde{\boldsymbol{r}}_m(0) = p_m(A)\boldsymbol{b}. \tag{5.24}$$

Proof. Let $q_{\widetilde{H}_m+tI,()^{-1}}(z)$ interpolate $f(z) = z^{-1}$ at $\operatorname{spec}(\widetilde{H}_m + tI) = \{\vartheta_1, \ldots, \vartheta_m\}$ in the Hermite sense. Then, by Lemma 2.27, we have $\widetilde{\boldsymbol{x}}_m(t) = q_{\widetilde{H}_m+tI,()^{-1}}(A+tI)\boldsymbol{b}$. Define the polynomial

$$p_{m,t}(z) = 1 - zq_{\widetilde{H}_m+tI,()^{-1}}(z)$$

of exact degree m, which satisfies $p_{m,t}(0) = 1$. Then $\widetilde{\boldsymbol{r}}_m(t) = p_{m,t}(A + tI)\boldsymbol{b}$. Due to

$$p_{m,t}(\vartheta_i + t) = 1 - (\vartheta_i + t) \cdot \frac{1}{\vartheta_i + t} = 0,$$

the polynomial $p_{m,t}$ interpolates the zero function at $\operatorname{spec}(\widetilde{H}_m + tI)$. In particular, we have

$$p_{m,0}(z) = \prod_{i=1}^{m} \left(1 - \frac{z}{\vartheta_i}\right)$$

and

$$p_{m,t}(z) = \prod_{i=1}^{m} \left(1 - \frac{z}{\vartheta_i + t}\right) = \frac{1}{p_{m,0}(-t)} p_{m,0}(z - t). \tag{5.25}$$

The last equality in (5.25) holds because the polynomial on the right-hand side has the same zeros as $p_{m,t}$ and attains the value 1 at $z = 0$. We thus find

$$\widetilde{\boldsymbol{r}}_m(t) = p_{m,t}(A + tI)\boldsymbol{b} = \frac{1}{p_{m,0}(-t)} p_{m,0}(A) = \eta_m(t) p_{m,0}(A)\boldsymbol{b} = \eta_m(t)\widetilde{\boldsymbol{r}}_m(0),$$

which proves the assertion of the lemma. □

The result of Lemma 5.15 shows that the residuals of the iterates $\widetilde{\boldsymbol{x}}_m(t)$ are collinear to the residual produced by GMRES for the system $A\boldsymbol{x} = \boldsymbol{b}$. This already suggests to conjecture that there is a connection between the iterates $\widetilde{\boldsymbol{x}}_m(t)$ generated by the shifted GMRES method from [56], which was briefly described in Section 2.4.1, as there the residuals of the shifted systems are also enforced to be collinear to the GMRES residual of the seed system. By comparing the collinearity factor $\eta_m(t)$ given in (5.24) with the one from [56], one discovers that they are indeed the same (if the seed system is chosen as the system with shift $t = 0$), and that thus also the approximations $\widetilde{\boldsymbol{x}}_m(t)$ are the same (as long as A is nonsingular, as then the residual uniquely determines the approximation due to the residual equation). Besides being an interesting observation, this is also useful because it allows to determine the approximations $\widetilde{\boldsymbol{x}}_m(t)$ in the way proposed in [56] without needing to form the matrix $\widetilde{H}_m$, which could in some cases lead to numerical instabilities. We keep this in mind while still using $\widetilde{H}_m$ in the following to avoid unnecessary notational overhead.

5.5 Convergence of restarted harmonic Arnoldi for Stieltjes functions

In this section, we show how to transfer results on the convergence of restarted, shifted GMRES for positive real matrices A to the restarted harmonic Arnoldi approximation.

Proceeding similarly to (5.4), this time for the error representation (5.23), and using the result of Lemma 5.15, the error of the harmonic Arnoldi approximation can be written as

$$f(A)\boldsymbol{b} - \widetilde{\boldsymbol{f}}_m = \int_0^\infty (A+tI)^{-1}\widetilde{\boldsymbol{r}}_m(t)\,\mathrm{d}\mu(t) = \int_0^\infty \eta_m(t)(A+tI)^{-1}\,\mathrm{d}\mu(t)\cdot\widetilde{\boldsymbol{r}}_m(0)$$

with $\eta_m(t)$ from (5.24). We thus have $f(A)\boldsymbol{b} - \widetilde{\boldsymbol{f}}_m = \widetilde{e}_m(A)\widetilde{\boldsymbol{r}}_m(0)$ with the error function

$$\widetilde{e}_m(z) = \int_0^\infty \frac{\eta_m(t)}{z+t}\,\mathrm{d}\mu(t).$$

We will show that the residual collinearity factors $\eta_m(t)$ are bounded from above by 1. This is similar to the analysis performed in [56] for proving that the iterates of the restarted shifted GMRES method are convergent for A positive real.

Lemma 5.16. *Let $A \in \mathbb{C}^{n\times n}$ be positive real and let $\eta_m(t)$ be defined as in* (5.24)*. Then*

$$|\eta_m(t)| \le \left(\frac{1}{1+t\rho}\right)^m \le 1, \tag{5.26}$$

where

$$\rho := \min\left\{\Re\left(\frac{\boldsymbol{v}^H A^{-1}\boldsymbol{v}}{\boldsymbol{v}^H\boldsymbol{v}}\right) : \boldsymbol{v}\in\mathbb{C}^n, \boldsymbol{v}\neq 0\right\}. \tag{5.27}$$

Proof. From the definition of $\eta_m(t)$ in Lemma 5.15 we have

$$\eta_m(t) = \frac{1}{\prod_{i=1}^m (1+\frac{t}{\vartheta_i})},$$

with ϑ_i being the harmonic Ritz values of A with respect to $\mathcal{K}_m(A,\boldsymbol{b})$. Since the harmonic Ritz values of A are the inverses of the Ritz values of A^{-1} with respect to $A\mathcal{K}_m(A,\boldsymbol{b})$, see [111], we have $\vartheta_i^{-1} = (\boldsymbol{w}_i^H A^{-1}\boldsymbol{w}_i)/(\boldsymbol{w}_i^H\boldsymbol{w}_i)$ for some vector $\boldsymbol{w}_i \in \mathbb{C}^n$ and thus $\Re(\vartheta_i^{-1}) \ge \rho$. Therefore, for any $t \ge 0$ we have, using that $\Re(\vartheta_i) \ge 0, i = 1,\ldots,m$,

$$|1+t\vartheta_i^{-1}| \ge 1 + t\Re(\vartheta_i^{-1}) \ge 1+t\rho \quad \text{for } i = 1,\ldots,m,$$

which gives (5.26). □

The natural choice of norm for bounding the error of a Krylov subspace method for non-Hermitian positive real A is the energy norm induced by the matrix A^HA (which is Hermitian positive definite when A is positive real), as most known results bound the Euclidean norm of the residual (see, e.g., Theorem 2.36) and we have

$$\|\boldsymbol{r}\|_2 = \sqrt{\boldsymbol{r}^H\boldsymbol{r}} = \sqrt{(A\boldsymbol{e})^H(A\boldsymbol{e})} = \|\boldsymbol{e}\|_{A^HA}.$$

In the proof of Lemma 5.5, we used the relation between the energy norms induced by the Hermitian positive definite matrices A and $A+tI$, and in the following we need a similar result for the norms induced by A^HA and $(A+tI)^H(A+tI)$. As the situation is a little bit more involved to analyze, we state the precise result in a separate lemma.

Lemma 5.17. *Let $A \in \mathbb{C}^{n\times n}$ be positive real.*

(i) For all $\boldsymbol{v} \in \mathbb{C}^n$ and $t \geq 0$ we have

$$\|\boldsymbol{v}\|_{A^HA}^2 \leq \frac{1}{\nu_{\max}^{-1}t^2 + 2\rho t + 1}\|\boldsymbol{v}\|_{(A+tI)^H(A+tI)}^2\,, \tag{5.28}$$

where ρ is defined in (5.27) *and*

$$\nu_{\max} := \max\left\{\frac{(A\boldsymbol{v})^H(A\boldsymbol{v})}{\boldsymbol{v}^H\boldsymbol{v}} : \boldsymbol{v} \in \mathbb{C}^n, \boldsymbol{v} \neq 0\right\} = \|A\|_2^2. \tag{5.29}$$

(ii) For $t \geq 0$ we have

$$\frac{1}{\nu_{\max}^{-1}t^2 + 2\rho t + 1} \leq \frac{\nu_{\max}}{(t+\rho\nu_{\max})^2}. \tag{5.30}$$

Proof. For part (i) we expand

$$\|\boldsymbol{v}\|_{(A+tI)^H(A+tI)}^2 = \|\boldsymbol{v}\|_{A^HA}^2 + 2t\Re(\boldsymbol{v}^HA^H\boldsymbol{v}) + t^2\|\boldsymbol{v}\|_2^2\,.$$

The inequality now follows from $\|\boldsymbol{v}\|_2^2 \geq \frac{1}{\nu_{\max}}\|\boldsymbol{v}\|_{A^HA}^2$ and

$$\Re(\boldsymbol{v}^HA^H\boldsymbol{v})/(\boldsymbol{v}^HA^HA\boldsymbol{v}) = \Re(\boldsymbol{w}^HA^{-1}\boldsymbol{w})/(\boldsymbol{w}^H\boldsymbol{w}) \geq \rho, \text{ where } A\boldsymbol{v} = \boldsymbol{w}.$$

The inequality in part (ii) is equivalent to $(t+\rho\nu_{\max})^2 \leq t^2 + 2\rho\nu_{\max}t + \nu_{\max}$, i.e., to $\rho^2\nu_{\max} \leq 1$, which can be established as follows. Let $\boldsymbol{v}$ be the normalized eigenvector of $(AA^H)^{-1}$ corresponding to the smallest eigenvalue, which is $1/\nu_{\max}$. Then, by the Cauchy–Schwarz inequality,

$$\rho \leq |\boldsymbol{v}^HA^{-1}\boldsymbol{v}| \leq \|\boldsymbol{v}\|_2 \cdot \|A^{-1}\boldsymbol{v}\|_2 = \sqrt{\boldsymbol{v}^H(AA^H)^{-1}\boldsymbol{v}} = \frac{1}{\nu_{\max}^{1/2}}\,,$$

which concludes the proof of the lemma. □

With these prerequisites, we are in a position to prove the following theorem on the convergence of the restarted harmonic Arnoldi method for positive real matrices.

Theorem 5.18. *Let $A \in \mathbb{C}^{n\times n}$ be positive real, let $\boldsymbol{b} \in \mathbb{C}^n$, let f be a Stieltjes function of the form (3.15), and let $\widetilde{\boldsymbol{f}}_m^{(k)}$ be the approximation from k cycles of the restarted harmonic Arnoldi method with restart length m. Further, let ρ be as defined in (5.27) and let δ, δ' be defined as in (2.37) and (2.38), respectively. For $t \geq 0$ define*

$$\widetilde{\alpha}_m(t) := \left(\frac{\sqrt{1-\delta\delta'}}{1+t\rho}\right)^m.$$

Let $t_0 \geq 0$ be the left endpoint of the support of μ. Then the A^HA-energy norm of the error of $\widetilde{\boldsymbol{f}}_m^{(k)}$ satisfies

$$\begin{aligned}
\|f(A)\boldsymbol{b} - \widetilde{\boldsymbol{f}}_m^{(k)}\|_{A^HA} &\leq \|\boldsymbol{r}_m^{(k)}(0)\|_2 \int_0^\infty \frac{(1+t\rho)^{-mk}}{\sqrt{\nu_{\max}^{-1}t^2 + 2\rho t + 1}}\,\mathrm{d}\mu(t) && (5.31)\\
&\leq \|\boldsymbol{b}\|_2 \int_0^\infty \frac{\widetilde{\alpha}_m(t)^k}{\sqrt{\nu_{\max}^{-1}t^2 + 2\rho t + 1}}\,\mathrm{d}\mu(t) && (5.32)\\
&\leq C\widetilde{\alpha}_m(t_0)^k, && (5.33)
\end{aligned}$$

where $0 \leq \widetilde{\alpha}_m(t_0) < 1$ and

$$C = \|\boldsymbol{b}\|_2\sqrt{\nu_{\max}}f(\rho\nu_{\max}) \tag{5.34}$$

with $\nu_{\max}$ defined as in (5.29). In particular, the restarted harmonic Arnoldi method converges for all restart lengths $m \geq 1$.

Proof. As the proof is very similar to that of Lemma 5.5 and Theorem 5.7 we only give a sketch. Using an upper index, as before, to distinguish the quantities belonging to different restart cycles we have

$$f(A)\boldsymbol{b} - \widetilde{\boldsymbol{f}}_m^{(k)} = \int_0^\infty \widetilde{\boldsymbol{e}}_m^{(k)}(t)\,\mathrm{d}\mu(t) = \int_0^\infty (A+tI)^{-1}\widetilde{\boldsymbol{r}}_m^{(k)}(t)\,\mathrm{d}\mu(t).$$

Using Lemma 5.17(i) together with the equality $\|\widetilde{\boldsymbol{e}}_m^{(k)}(t)\|_{(A+tI)^H(A+tI)} = \|\widetilde{\boldsymbol{r}}_m^{(k)}(t)\|_2$ and the collinearity of these residuals as stated in Lemma 5.15, one obtains

$$\|f(A)\boldsymbol{b} - \widetilde{\boldsymbol{f}}_m^{(k)}\|_{A^HA} \leq \int_0^\infty \frac{|\eta_m^{(1)}(t)\cdots\eta_m^{(k)}(t)|}{\sqrt{\nu_{\max}^{-1}t^2 + 2\rho t + 1}}\|\widetilde{\boldsymbol{r}}_m^{(k)}(0)\|_2\,\mathrm{d}\mu(t).$$

Inequality (5.31) now follows by bounding each factor $|\eta_m^{(j)}(t)|$ via (5.26). The second relation (5.32) is obtained by using the bound for $\|\widetilde{\boldsymbol{r}}_m^{(k)}(0)\|_2$ from Theorem 2.36. To get (5.33) and (5.34) one then uses the fact that $\widetilde{\alpha}_m(t)$ is monotonically decreasing as a function of t and the bound (5.30). □

We just note that it is of course again possible to replace the bound (5.33) for the A^HA-energy norm by a bound for the Euclidean norm of the error by suitably modifying the constant C, similarly to what we have done for the convergence bounds in the standard restarted Arnoldi method. We do not give the details here, as this is completely analogous. Convergence for functions of the type $\widetilde{f}(z) = zf(z)$ for f a Stieltjes function, i.e., an analogue to Theorem 5.12 for the restarted harmonic Arnoldi method is also possible, again using exactly the same tools as before, so that we omit it here and just state that the restarted (corrected) harmonic Arnoldi method also converges for functions of this type when A is positive real.

Theorem 5.18 guarantees the convergence of the restarted harmonic Arnoldi method for all restart lengths m, but we give an additional result, which gives a little more insight into the behavior of the method in comparison to restarted GMRES, as it gives an in a sense more immediate relation.

Corollary 5.19. *Let the assumptions of Theorem 5.18 hold. Then*

$$\|f(A)\boldsymbol{b} - \widetilde{\boldsymbol{f}}_m^{(k)}\|_{A^HA} \leq C_1 \|\boldsymbol{r}_m^{(k)}(0)\|_2, \tag{5.35}$$

where $C_1 = \sqrt{\nu_{\max}} f(\rho\nu_{\max})$.

Proof. We insert the relation $(1 + t\rho) \geq 1$ for all $t \geq 0$ into (5.31), which yields

$$\|f(A)\boldsymbol{b} - \widetilde{\boldsymbol{f}}_m^{(k)}\|_{A^HA} \leq \|\boldsymbol{r}_m^{(k)}(0)\|_2 \int_0^\infty \frac{1}{\sqrt{\nu_{\max}^{-1} t^2 + 2\rho t + 1}} \, \mathrm{d}\mu(t).$$

The assertion of the corollary then follows by applying (5.30). □

Corollary 5.19 is especially interesting in the context of superlinear convergence of the GMRES method, see, e.g., [106, 139]. In this setting, the statement of the corollary can be rephrased as: If (restarted) GMRES for the positive real linear system $A\boldsymbol{x} = \boldsymbol{b}$ exhibits superlinear convergence behavior, then so does the (restarted) harmonic Arnoldi method for approximating $f(A)\boldsymbol{b}$ when f is a Stieltjes function.

We revisit Example 5.14, in which the standard restarted Arnoldi method failed to converge for a (normal) positive real matrix A. In Figure 5.2, we give the convergence curves of the restarted Arnoldi and restarted harmonic Arnoldi method for the same problem (and with the same parameters) as considered in Example 5.14. As predicted by Theorem 5.18, we observe that the restarted harmonic Arnoldi method converges linearly to $f(A)\boldsymbol{b}$.

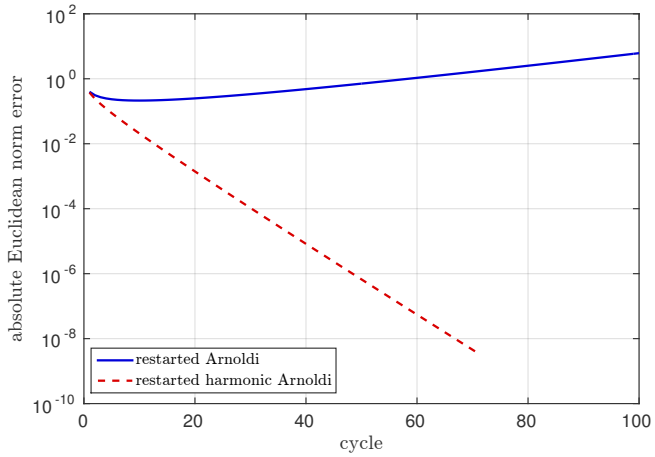


Figure 5.2: Norms of the error of the iterates produced by the restarted Arnoldi and restarted harmonic Arnoldi method with restart length $m = 10$ applied to the matrix A from (5.21) with $n = 21$ and $\alpha = 0.995$ for approximating $A^{-1/2}\hat{\boldsymbol{e}}_1$.

Remark 5.20. We only considered "standard" harmonic Ritz values in this section, so that some of the results on the convergence of the restarted harmonic Arnoldi method, in particular Corollary 5.19, involve quantities corresponding to the underlying linear system with shift $t = 0$. It is also possible to define *shifted harmonic Ritz values* ϑ_i with respect to a subspace $\mathcal{U} \subseteq \mathbb{C}^n$ and a "target" t_0 other than 0. These shifted harmonic Ritz values satisfy

$$(A + t_0 I)\boldsymbol{x}_i - (\vartheta_i + t_0)\boldsymbol{x}_i \perp (A + t_0 I)\mathcal{U}$$

with $\mathbf{0} \neq \boldsymbol{x}_i \in \mathbb{C}^n$, see, e.g., [87]. If the left endpoint t_0 of the support of μ is different from zero, then these shifted harmonic Ritz values allow to refine the analysis such that this fact can be taken into account. All results presented in this section can be modified accordingly, but we refrain from explicitly doing so for the sake of brevity and notational simplicity.

We end this section by stating a further result on the standard restarted Arnoldi method for A Hermitian positive definite. It can be derived in the same way as (5.35) by using the fact that all (restarted) CG residuals are collinear according to the shift invariance stated by Proposition 2.38 and replacing the harmonic Ritz values by the standard Ritz values, which are known to all lie in $[\lambda_{\min}, \lambda_{\max}]$. This way, one obtains the following result.

Theorem 5.21. *Let $A \in \mathbb{C}^{n\times n}$ be Hermitian positive definite, let $\boldsymbol{b} \in \mathbb{C}^n$, let f be a Stieltjes function of the form (3.15), and let $\boldsymbol{f}_m^{(k)}$ be the approximation from k cycles of the restarted Arnoldi method with restart length m. Further, let $t_0 \geq 0$*

be the left endpoint of the support of μ. The A^2-energy norm of the error of $\boldsymbol{f}_m^{(k)}$ can then be bounded as

$$\|f(A)\boldsymbol{b} - \boldsymbol{f}_m^{(k)}\|_{A^2} \leq C\|\boldsymbol{r}_m^{(k)}(t_0)\|_2, \tag{5.36}$$

where $\|\boldsymbol{r}_m^{(k)}(t_0)\|_2$ is the Euclidean norm of the residual of the (restarted) CG iterate for the system $(A + t_0 I)\boldsymbol{x}(t_0) = \boldsymbol{b}$ and

$$C = \lambda_{\max} f\big(\sqrt{\lambda_{\max}}\big).$$

Consequently,

$$\|f(A)\boldsymbol{b} - \boldsymbol{f}_m^{(k)}\|_A \leq \frac{C}{\sqrt{\lambda_{\min}}}\|\boldsymbol{r}_m^{(k)}(t_0)\|_2. \tag{5.37}$$

Again, this result essentially states that we can expect the restarted Arnoldi method to exhibit superlinear convergence behavior whenever restarted CG for the system $(A + t_0 I)\boldsymbol{x} = \boldsymbol{b}$ converges superlinearly [12, 13]. Therefore, this result is, just as Corollary 5.19, especially interesting in the unrestarted case (or for restart lengths which are very large in relation to n, which are seldom used in practice), as one typically observes superlinear convergence only in these cases; cf. also the experiments reported in Section 5.7. We just briefly remark that, while all other results stated in this chapter for Hermitian positive definite matrices use the energy norm corresponding to A, it is natural to initially arrive at an estimate in the energy norm corresponding to A^2 in (5.36), as the proof relies on the relation

$$\|\boldsymbol{r}_m^{(k)}(t)\|_2 = \|\boldsymbol{e}_m^{(k)}(t)\|_{A^2}.$$

5.6 Convergence of restarted FOM for linear systems

In this section, we give a few results concerning the convergence behavior of restarted FOM (and restarted GMRES) for the solution of non-Hermitian linear systems. This topic is only remotely related to the other results presented in this section (in the sense that $f(z) = z^{-1}$ is also a special case of a Stieltjes function) and can be regarded as a by-product of investigating matrices like the one from Example 5.14. The results in this section are also presented in [119].

As already illustrated, e.g., by Example 2.30, the restarted full orthogonalization method can stagnate at some point, without ever reaching the desired solution $A^{-1}\boldsymbol{b}$, even in exact arithmetic. Approximately solving a linear system with the matrix from Example 5.14 and the right-hand side $\hat{\boldsymbol{e}}_1$ with restarted FOM would lead to a sequence of residual norms which exhibit a similar exponential growth

as the one depicted for the error norms for approximating $A^{-1/2}\boldsymbol{b}$ in Figure 5.1. What kinds of behavior restarted FOM can exhibit under which circumstances is an issue that is not fully understood by now apart from the Hermitian positive definite case, when restarted FOM reduces to restarted CG, which is known to converge to $A^{-1}\boldsymbol{b}$; cf. also Section 5.2. The asymptotic speed of convergence in the Hermitian positive definite case is closely tied to spectral information of A, e.g., the smallest and largest eigenvalue for estimates like in Theorem 2.32, or information on clusters formed by the eigenvalues and outliers from the spectrum for a more intricate analysis. This could lead one to believe that eigenvalue information can also be used to gain insight into the behavior of FOM in the non-Hermitian case. We will show that this is not true and that restarted FOM can attain any behavior completely independent of the spectrum of A. Results of this type are known for (restarted) GMRES, see, e.g., [39, 78, 137]. The result from [78] essentially states that unrestarted GMRES can generate any prescribed, monotonically decreasing sequence of residual norms for a matrix which has any desired eigenvalues. In [39], a refined result of this type is presented, in which also the Ritz values in each iteration can be freely prescribed. In [137], a similar result is presented for restarted GMRES, where the residual norms at the end of each restart cycle can be prescribed for the first $\lfloor \frac{n}{m} \rfloor$ iterations (where m is the restart length), again for a matrix with any desired eigenvalues. A more general result, allowing to prescribe the residual norms also in the iterations within each cycle (and with the possibility to additionally prescribe all Ritz values) was recently given in [40].

We present similar results for restarted FOM, where the residual norm in each iteration can be prescribed for the first n iterations (at most). Towards the end of this section, we will further comment on the relation and differences of our results for FOM in comparison to the ones for GMRES.

To make notation not overly complicated, we will, in contrast to most of the other parts of this thesis, number the residuals consecutively, i.e., $\boldsymbol{r}_1, \ldots, \boldsymbol{r}_m$ are the iterates from the first restart cycle, $\boldsymbol{r}_{m+1}, \ldots, \boldsymbol{r}_{2m}$ are the iterates from the second restart cycle and so on.

Theorem 5.22. *Let $m, n, q \in \mathbb{N}$ with $m \leq n-1$ and $q \leq n$, let $r_1, \ldots, r_q \in \mathbb{R}_0^+$ be given with $r_1, \ldots, r_{q-1} > 0$ and $r_q \geq 0$ and let $\mu_1, \ldots, \mu_n \in \mathbb{C} \setminus \{0\}$. Then there exist a matrix $A \in \mathbb{C}^{n \times n}$ with $\operatorname{spec}(A) = \{\mu_1, \ldots \mu_n\}$ and vectors $\boldsymbol{b}, \boldsymbol{x}_0 \in \mathbb{C}^n$ such that the residuals $\boldsymbol{r}_1, \ldots, \boldsymbol{r}_q$ generated by q steps of restarted FOM with restart length m for $A\boldsymbol{x} = \boldsymbol{b}$ with initial guess $\boldsymbol{x}_0$ satisfy*

$$\|\boldsymbol{r}_j\|_2 = r_j \text{ for } j = 1, \ldots, q.$$

The proof of Theorem 5.22 is quite lengthy and will require a few auxiliary results

which are given next. We investigate matrices of the form

$$A(\boldsymbol{d},\boldsymbol{s}) = \begin{bmatrix} d_1 & 0 & \cdots & 0 & s_n \\ s_1 & d_2 & 0 & \cdots & 0 \\ 0 & s_2 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & d_{n-1} & 0 \\ 0 & \cdots & 0 & s_{n-1} & d_n \end{bmatrix} \tag{5.38}$$

defined by two vectors $\boldsymbol{d},\boldsymbol{s} \in \mathbb{C}^n$ in the following (note that the matrices from Example 2.30 and 5.14 are both special cases of (5.38)).

For the sake of simplicity we assume that $q = n$ and $r_n > 0$ in Theorem 5.22. This is no essential restriction, as we will point out at the end of the proof of Theorem 5.22. We first examine the results of applying Arnoldi's method to $A(\boldsymbol{d},\boldsymbol{s})$ and a (multiple of a) canonical unit vector.

Proposition 5.23. *Let $A(\boldsymbol{d},\boldsymbol{s}) \in \mathbb{C}^{n\times n}$ be of the form (5.38), let $m \leq n-1$, $\xi_0 \in \mathbb{C}$ with $|\xi_0| = 1$, and let $c > 0$. Let $\boldsymbol{x}_0, \boldsymbol{b} \in \mathbb{C}^n$ be given such that the residual $\boldsymbol{r}_0 = \boldsymbol{b} - A\boldsymbol{x}_0$ satisfies $\boldsymbol{r}_0 = \xi_0 c\hat{\boldsymbol{e}}_i$. Then the basis V_{j+1} generated by $j \leq m$ steps of Arnoldi's method, Algorithm 2.1, for $A(\boldsymbol{d},\boldsymbol{s})$ and $\boldsymbol{b}$ with initial guess $\boldsymbol{x}_0$ is given by*

$$V_{j+1} = [\xi_0\hat{\boldsymbol{e}}_i, \xi_1\hat{\boldsymbol{e}}_{i+1}, \ldots, \xi_j\hat{\boldsymbol{e}}_{i+j}] \tag{5.39}$$

(where, like everywhere in the following, for ease of notation, the indices are to be understood cyclically, i.e., $\hat{\boldsymbol{e}}_{n+1} := \hat{\boldsymbol{e}}_1, \hat{\boldsymbol{e}}_{n+2} := \hat{\boldsymbol{e}}_2, \ldots$) with $\xi_k = \frac{s_{i+k-1}\xi_{k-1}}{|s_{i+k-1}|}, k = 1,\ldots,j$. The corresponding upper Hessenberg matrix is given by

$$H_j = \begin{bmatrix} d_i & 0 & \cdots & 0 & 0 \\ |s_i| & d_{i+1} & 0 & \cdots & 0 \\ 0 & |s_{i+1}| & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & d_{i+j-2} & 0 \\ 0 & \cdots & 0 & |s_{i+j-2}| & d_{i+j-1} \end{bmatrix}, \quad h_{j+1,j} = |s_{i+j-1}|. \tag{5.40}$$

Proof. One verifies by a direct computation that V_{j+1} and $H_j, h_{j+1,j}$ from (5.39) and (5.40), respectively, satisfy the Arnoldi relation (2.23) for $A(\boldsymbol{d},\boldsymbol{s})$. The assertion then follows from the essential uniqueness of the Arnoldi decomposition, see Lemma 2.23, because all subdiagonal entries of H_j are real and positive. □

Given the orthonormal basis and Hessenberg matrix resulting from Arnoldi's method, we can easily give an explicit expression for the residuals generated by applying FOM to the linear system $A(\boldsymbol{d},\boldsymbol{s})\boldsymbol{x} = \boldsymbol{b}$. Using Proposition 5.23, one obtains the following result.

Proposition 5.24. *Let the assumptions of Proposition 5.23 hold. Then the residual generated by $j \leq m$ steps of FOM is given by*

$$\boldsymbol{r}_j = (-1)^j \xi_j c \frac{|s_i \cdot s_{i+1} \cdots s_{i+j-1}|}{d_i \cdot d_{i+1} \cdots d_{i+j-1}} \hat{\boldsymbol{e}}_{i+j}.$$

In particular,

$$\|\boldsymbol{r}_j\|_2 = \left| \frac{s_{i+j-1}}{d_{i+j-1}} \right| \cdot \|\boldsymbol{r}_{j-1}\|_2.$$

Proof. The FOM residual satisfies $\boldsymbol{r}_j = -h_{j+1,j} \|\boldsymbol{r}_0\|_2 \left(\hat{\boldsymbol{e}}_j^H H_j^{-1} \hat{\boldsymbol{e}}_1 \right) \boldsymbol{v}_{j+1}$ according to Proposition 2.29. In our setting we have

$$\|\boldsymbol{r}_0\|_2 = c, \quad h_{j+1,j} = |s_{i+j-1}|, \quad \boldsymbol{v}_{j+1} = \xi_j \hat{\boldsymbol{e}}_{i+j},$$

and the lower left entry $\hat{\boldsymbol{e}}_j^H H_j^{-1} \hat{\boldsymbol{e}}_1$ of H_j^{-1} is $(-1)^{j-1} \frac{|s_{i+1}| \cdots |s_{i+j-2}|}{d_{i+1} \cdots d_{i+j-1}}$ due to the simple, bidiagonal structure of H_j. Putting these quantities together proves the proposition. □

When applying restarted FOM with restart length m to the linear system

$$A(\boldsymbol{d}, \boldsymbol{s}) \boldsymbol{x} = \hat{\boldsymbol{e}}_1$$

with initial guess $\boldsymbol{x}_0 = \boldsymbol{0}$ it is now easy to use Proposition 5.24 to choose some of the coefficients in $\boldsymbol{d}$ and $\boldsymbol{s}$ in such a way that the prescribed residual norms $r_1, \ldots, r_m$ are generated. More precisely, one just needs to choose the first m entries of the coefficient vectors $\boldsymbol{d}, \boldsymbol{s}$ such that they satisfy

$$s_j = \frac{r_j}{r_{j-1}} d_j, \quad j = 1, \ldots, m \tag{5.41}$$

(where we set $r_0 = 1$) to produce the desired residual norm sequence. Assuming that $d_1, \ldots, d_m$ are already fixed to arbitrary nonzero values, it is always possible to choose $s_1, \ldots, s_m$ in such a way that (5.41) is satisfied. After restarting, the situation is very similar. After the first m steps of FOM, the residual is, according to Proposition 5.24, given by

$$\boldsymbol{r}_m = (-1)^m \xi_m \frac{|s_1 \cdot s_2 \cdots s_m|}{d_1 \cdot d_2 \cdots d_m} \hat{\boldsymbol{e}}_{m+1}.$$

Therefore, using the new initial guess $\boldsymbol{x}_m$ after restarting, we are again in a situation in which the assumptions of Proposition 5.24 are fulfilled, with $c = |\frac{s_1 \cdots s_m}{d_1 \cdots d_m}| = r_m$. From this, it is immediately clear that choosing the next m values in $\boldsymbol{d}$ and $\boldsymbol{s}$ (analogously to (5.41)) such that

$$s_j = \frac{r_j}{r_{j-1}} d_j, \quad j = m+1, \ldots, \min\{2m, n\}$$

is fulfilled will produce the desired residual norms $r_{m+1}, \ldots, r_{\min\{2m,n\}}$ in the next cycle of restarted FOM (or in the first $n-m$ iterations of this cycle, if $2m > n$). By continuing this construction, one can prescribe residual norms for further iterations until all values in $\boldsymbol{s}$ are fixed (i.e., for n iterations).

We assumed that the coefficients in $\boldsymbol{d}$ are already fixed to some arbitrary nonzero values. We will now show how to fix them in such a way that the matrix $A(\boldsymbol{d}, \boldsymbol{s})$ has any desired (nonzero) eigenvalues $\mu_1, \ldots, \mu_n$. One immediately sees that the characteristic polynomial of $A(\boldsymbol{d}, \boldsymbol{s})$ is given by

$$\chi_{A(\boldsymbol{d},\boldsymbol{s})}(\lambda) = (\lambda - d_1) \cdots (\lambda - d_n) - s_1 \cdots s_n. \tag{5.42}$$

Assuming that the matrix $A(\boldsymbol{d}, \boldsymbol{s})$ produces the desired sequence of residual norms, we can eliminate the dependency of $\chi_{A(\boldsymbol{d},\boldsymbol{s})}$ on the values $s_1, \ldots, s_n$. Multiplying all equations in (5.41) (and its counterparts from later restart cycles), we find the relation

$$s_1 \cdots s_n = r_n \cdot d_1 \cdots d_n. \tag{5.43}$$

Inserting this into (5.42), we can rewrite the characteristic polynomial as

$$\chi_{A(\boldsymbol{d},\boldsymbol{s})}(\lambda) = (\lambda - d_1) \cdots (\lambda - d_n) - r_n \cdot d_1 \cdots d_n. \tag{5.44}$$

There exist coefficients $\beta_0, \ldots, \beta_{n-1} \in \mathbb{C}$ such that the values $\mu_1, \ldots, \mu_n$ are the roots of the corresponding monic polynomial, i.e.,

$$(\lambda - \mu_1) \cdots (\lambda - \mu_n) = \lambda^n + \beta_{n-1}\lambda^{n-1} + \cdots + \beta_1\lambda + \beta_0. \tag{5.45}$$

Note that the following construction breaks down if $r_n = (-1)^n$ (which can of course only happen for even n, as $r_n \geq 0$). However, we may assume that $r_n \neq (-1)^n$ without loss of generality. This can be seen as follows. If $r_n = (-1)^n$, we can choose an arbitrary value $\alpha \notin \{0, 1\}$, replace all values r_i by αr_i and start the FOM iteration with right-hand side $\frac{1}{\alpha}\hat{\boldsymbol{e}}_1$, which will produce the same sequence of residual norms.

We now choose the values $d_1, \ldots, d_n$ such that they are the n roots of the polynomial

$$\lambda^n + \beta_{n-1}\lambda^{n-1} + \cdots \beta_1\lambda + \widetilde{\beta}_0 \text{ with } \widetilde{\beta}_0 = \frac{\beta_0}{1 + (-1)^{n+1} r_n}.$$

These exist due to the fundamental theorem of algebra. With this choice of the roots d_i it obviously holds

$$(-1)^n d_1 \cdots d_n = \widetilde{\beta}_0. \tag{5.46}$$

Inserting this into the characteristic polynomial (5.44), we find

$$\begin{aligned}
\chi_{A(\boldsymbol{d},\boldsymbol{s})}(\lambda) &= \lambda^n + \beta_{n-1}\lambda^{n-1} + \cdots + \beta_1\lambda + \widetilde{\beta}_0 - r_n \cdot d_1 \cdots d_n \\
&= \lambda^n + \beta_{n-1}\lambda^{n-1} + \cdots + \beta_1\lambda + \widetilde{\beta}_0 + (-1)^{n+1} r_n \widetilde{\beta}_0 \\
&= \lambda^n + \beta_{n-1}\lambda^{n-1} + \cdots + \beta_1\lambda + (1 + (-1)^{n+1} r_n)\widetilde{\beta}_0 \\
&= \lambda^n + \beta_{n-1}\lambda^{n-1} + \cdots + \beta_1\lambda + \beta_0,
\end{aligned}$$

showing that $A(\boldsymbol{d}, \boldsymbol{s})$ has the desired eigenvalues according to (5.45). Equation (5.46) together with the fact that $\widetilde{\beta}_0 \neq 0$ (because $\beta_0 = \mu_1 \cdots \mu_n \neq 0$) implies that all entries of $\boldsymbol{d}$ are nonzero, such that all Hessenberg matrices (5.40) are nonsingular and all restarted Arnoldi approximations are therefore defined. This proves Theorem 5.22 in case that $q = n$ and $r_n > 0$. If $q < n$, we can use the same construction as above and just fix the "unused" coefficients $s_{q+1}, \ldots, s_n$ in such a way that (5.43) still holds (where r_n is of course replaced by r_q), so this situation does not cause any difficulties. Now consider the case that $r_q = 0$. This implies $s_q = 0$ and the characteristic polynomial (5.42) of $A(\boldsymbol{d}, \boldsymbol{s})$ is therefore given by

$$\chi_{A(\boldsymbol{d},\boldsymbol{s})}(\lambda) = (\lambda - d_1) \cdots (\lambda - d_n),$$

showing that the eigenvalues of $A(\boldsymbol{d}, \boldsymbol{s})$ are just the entries of $\boldsymbol{d}$ in this situation. Therefore, the eigenvalues of $A(\boldsymbol{d}, \boldsymbol{s})$ can again freely be prescribed through the choice of the coefficients in $\boldsymbol{d}$. If $q \neq n$, the coefficients $s_{q+1}, \ldots, s_n$ can attain any values (e.g., all zero) as FOM terminates after finding the exact solution in the qth iteration in this case, and they are therefore of no importance. This concludes the proof of Theorem 5.22.

An interesting observation concerning the construction from the proof of Theorem 5.22 is the following. The result only allows to prescribe the residual norms for the first n iterations of restarted FOM, but due to the simple nature of the matrices $A(\boldsymbol{d}, \boldsymbol{s})$ we have full information on the residual norms in later iterations (exceeding n). Consider using FOM with restart length m for the linear system $A(\boldsymbol{d}, \boldsymbol{s})\boldsymbol{x} = \hat{\boldsymbol{e}}_1$ with initial guess $\boldsymbol{x}_0 = \boldsymbol{0}$ again (where we assume that $A(\boldsymbol{d}, \boldsymbol{s})$ was constructed with $q = n$ and $r_n > 0$). As the residual generated after n iterations is again a multiple of a canonical unit vector, Proposition 5.24 still applies in this situation. One thus finds that the residuals in iterations exceeding n satisfy

$$\|\boldsymbol{r}_{n+j}\|_2 = \frac{r_{j \bmod n}}{r_{j-1 \bmod n}} \|\boldsymbol{r}_{n+j-1}\|_2, \tag{5.47}$$

i.e., the ratios of consecutive residuals are repeated cyclically. This full information on the behavior of the method in "later" iterations is a feature that distinguishes our construction from the one in [40, 137] for restarted GMRES, in which no information at all is available on the behavior of the method after more than n iterations (or more than $\lfloor \frac{n}{m} \rfloor$ restart cycles).

Remark 5.25. The insight gained from the proof of Theorem 5.22 allows to better explain the behavior observed when trying to approximate the inverse square root in Example 5.14, cf. also Figure 5.1, which can be interpreted as implicitly solving shifted linear systems with A. In the first few restart cycles, the error norm actually decreases, as the (implicitly computed) iterates for the underlying linear systems belonging to large shifts converge. Divergence only takes place for systems belonging to some interval $[0, t']$ close to the origin, and

once the systems belonging to larger shifts are converged, the divergence for the other systems slowly becomes visible.

Another observation one can make about our construction is that it applies in almost the same way to unrestarted FOM (in fact, the method behaves exactly the same for all choices of restart length $m \leq n-1$), apart from the fact that unrestarted FOM must terminate with the exact solution at the nth step. This gives the following result.

Corollary 5.26. *Let $n \in \mathbb{N}$, $1 \leq q \leq n$, $r_1, \ldots, r_{q-1} \in \mathbb{R}^+$, $r_q = 0$ and let $\mu_1, \ldots, \mu_n \in \mathbb{C} \setminus \{0\}$. Then there exist a matrix $A \in \mathbb{C}^{n \times n}$ with $\operatorname{spec}(A) = \{\mu_1, \ldots \mu_n\}$ and vectors $\boldsymbol{b}, \boldsymbol{x}_0 \in \mathbb{C}^n$ such that the residuals $\boldsymbol{r}_j$ generated by j steps of FOM for $A\boldsymbol{x} = \boldsymbol{b}$ with initial guess $\boldsymbol{x}_0$ satisfy*

$$\|\boldsymbol{r}_j\|_2 = r_j \text{ for } j = 1, \ldots, q.$$

The proof of Corollary 5.26 is almost identical to the one of Theorem 5.22 apart from the fact that $\boldsymbol{r}_n$ must be the zero vector due to the finite termination property of unrestarted FOM.

We now discuss the relation between our results for (restarted) FOM and results on (restarted) GMRES from [39, 40, 78, 137]. The FOM residual norm and the GMRES residual norm are not independent of each other, but fulfill the following relation (where $\boldsymbol{r}_j^{\mathrm{F}}$ and $\boldsymbol{r}_j^{\mathrm{G}}$ denote the residual generated by m steps of FOM and GMRES, respectively)

$$\|\boldsymbol{r}_j^{\mathrm{F}}\|_2 = \frac{\|\boldsymbol{r}_j^{\mathrm{G}}\|_2}{\sqrt{1 - (\|\boldsymbol{r}_j^{\mathrm{G}}\|_2 / \|\boldsymbol{r}_{j-1}^{\mathrm{G}}\|_2)^2}}, \tag{5.48}$$

see, e.g., [29, 30], or [22, 104] for other relations between FOM and GMRES. Relation (5.48) allows to prove Corollary 5.26 directly as a corollary of the results from [39, 78] by constructing a matrix A and a vector $\boldsymbol{b}$ such that GMRES generates the sequence

$$r_j^{\mathrm{G}} := \frac{r_j^{\mathrm{F}}}{\sqrt{1 + (r_j^{\mathrm{F}} / r_{j-1}^{\mathrm{G}})^2}} \text{ with } r_0^{\mathrm{G}} = 1.$$

of residual norms, where r_j^{F} are the FOM residual norms to be prescribed. By virtue of (5.48), A and $\boldsymbol{b}$ will then produce the desired FOM residual norms. The result of Theorem 5.22, however, can not be derived in such a simple way from the older GMRES result of [137], as this result does not allow the residual norm at each iteration to be prescribed, but only from the more recent analysis of [40].

The other way around, our result can be used to construct A and $\boldsymbol{b}$ in such a way that they produce an arbitrary admissible convergence curve in restarted GMRES where the norm after each iteration can be prescribed. The only limitation in this case is that our construction does not allow for stagnation in GMRES, as stagnation from step j to step $j+1$ in GMRES corresponds to the $(j+1)$st FOM iterate not being defined; see [22]. This would require the value d_{j+1} to be zero, which is not possible in our construction. Therefore, we can conclude that one can construct a matrix A and a vector $\boldsymbol{b}$ with arbitrary nonzero eigenvalues which produce any *strictly* monotonically decreasing sequence of residual norms in restarted GMRES, which gives an alternative proof for a result slightly weaker than what was recently presented in [40].

Another result concerning restarted GMRES is given in the following. It is related to an open question from the conclusions section of [137], where the authors ask whether it is possible to give bounds on the residual norms generated by restarted GMRES based on eigenvalue information once the iteration number exceeds n. As our approach provides information on the residual norms also in these later iterations, cf. (5.47), we can negatively answer this question (for both FOM and GMRES). For FOM, this is directly obvious from (5.47), for GMRES we give the precise result (and its rather technical proof) in the following.

Simply put, the following theorem states that restarted GMRES can, independently of the eigenvalues of A, converge arbitrarily slowly for any number k (possibly larger than n) of iterations, in the sense that the norm of the residual is reduced only by a prescribed margin which can be chosen arbitrarily close to zero.

Theorem 5.27. *Let $n, m, k \in \mathbb{N}$, $m \leq n-1$, let $\mu_1, \ldots, \mu_n \in \mathbb{C} \setminus \{0\}$ and let $0 \leq \delta < 1$. Then there exist a matrix $A \in \mathbb{C}^{n\times n}$ with $\operatorname{spec}(A) = \{\mu_1, \ldots \mu_n\}$ and vectors $\boldsymbol{x}_0, \boldsymbol{b} \in \mathbb{C}^n$ such that the residual $\boldsymbol{r}_k^{\mathrm{G}} = \boldsymbol{b} - A\boldsymbol{x}_k^{\mathrm{G}}$ generated by k iterations of restarted GMRES with restart length m for $A\boldsymbol{x} = \boldsymbol{b}$ with initial guess $\boldsymbol{x}_0$ satisfies*

$$\|\boldsymbol{r}_k^{\mathrm{G}}\|_2 / \|\boldsymbol{r}_0^{\mathrm{G}}\|_2 \geq \delta.$$

Proof. According to Theorem 5.22 there exist a matrix $A \in \mathbb{C}^{n\times n}$ with eigenvalues $\mu_1, \ldots, \mu_n$ and vectors $\boldsymbol{b}, \boldsymbol{x}_0 \in \mathbb{C}^n$ such that the residuals $\boldsymbol{r}_j^{\mathrm{F}}$ produced by the first n iterations of restarted FOM with restart length m fulfill

$$\|\boldsymbol{r}_j^{\mathrm{F}}\|_2 = \rho^j \text{ with } \rho = \frac{\delta^{1/k}}{(1-\delta^{2/k})^{1/2}} \text{ for } j = 1, \ldots, n. \tag{5.49}$$

Due to (5.47), we then have that (5.49) also holds for $j > n$. We rephrase this relation as

$$\|\boldsymbol{r}_{j-1}^{\mathrm{F}}\|_2 = \frac{1}{\rho}\|\boldsymbol{r}_j^{\mathrm{F}}\|_2 \text{ for all } j \in \mathbb{N}. \tag{5.50}$$

By relation (5.48), we have that two consecutive residual norms generated by restarted GMRES for $A, \boldsymbol{b}$ and $\boldsymbol{x}_0$ fulfill

$$\begin{aligned}
\frac{\|\boldsymbol{r}_j^{\mathrm{G}}\|_2}{\|\boldsymbol{r}_{j-1}^{\mathrm{G}}\|_2} &= \frac{\|\boldsymbol{r}_j^{\mathrm{F}}\|_2}{\|\boldsymbol{r}_{j-1}^{\mathrm{G}}\|_2\sqrt{1+(\|\boldsymbol{r}_j^{\mathrm{F}}\|_2/\|\boldsymbol{r}_{j-1}^{\mathrm{G}}\|_2)^2}} \\
&= \frac{\|\boldsymbol{r}_j^{\mathrm{F}}\|_2}{\sqrt{\|\boldsymbol{r}_{j-1}^{\mathrm{G}}\|_2^2+\|\boldsymbol{r}_j^{\mathrm{F}}\|_2^2}} \\
&= \frac{\|\boldsymbol{r}_j^{\mathrm{F}}\|_2}{\sqrt{\frac{\|\boldsymbol{r}_{j-1}^{\mathrm{F}}\|_2^2}{\sqrt{1+(\|\boldsymbol{r}_{j-1}^{\mathrm{F}}\|_2/\|\boldsymbol{r}_{j-2}^{\mathrm{G}}\|_2)^2}}+\|\boldsymbol{r}_j^{\mathrm{F}}\|_2^2}} \\
&\geq \frac{\|\boldsymbol{r}_j^{\mathrm{F}}\|_2}{\sqrt{\|\boldsymbol{r}_{j-1}^{\mathrm{F}}\|_2^2+\|\boldsymbol{r}_j^{\mathrm{F}}\|_2^2}}. \qquad (5.51)
\end{aligned}$$

Inserting (5.50) into the right-hand side of (5.51), we find

$$\frac{\|\boldsymbol{r}_j^{\mathrm{G}}\|_2}{\|\boldsymbol{r}_{j-1}^{\mathrm{G}}\|_2} \geq \frac{\|\boldsymbol{r}_j^{\mathrm{F}}\|_2}{\sqrt{\frac{1}{\rho^2}\|\boldsymbol{r}_j^{\mathrm{F}}\|_2^2+\|\boldsymbol{r}_j^{\mathrm{F}}\|_2^2}} = \frac{1}{\sqrt{\frac{1}{\rho^2}+1}}. \qquad (5.52)$$

Repeated application of (5.52) for all $j \leq k$ yields

$$\|\boldsymbol{r}_k^{\mathrm{G}}\|_2/\|\boldsymbol{r}_0^{\mathrm{G}}\|_2 = \big(\|\boldsymbol{r}_k^{\mathrm{G}}\|_2/\|\boldsymbol{r}_{k-1}^{\mathrm{G}}\|_2\big)\cdots\big(\|\boldsymbol{r}_1^{\mathrm{G}}\|_2/\|\boldsymbol{r}_0^{\mathrm{G}}\|_2\big) \geq \frac{1}{\big(\frac{1}{\rho^2}+1\big)^{k/2}}. \qquad (5.53)$$

The result follows from (5.53) by noting that $(\frac{1}{\rho^2}+1)^{k/2} = \frac{1}{\delta}$. □

5.7 Numerical experiments

In this section we report a few experiments which illustrate the convergence theory developed in this chapter. As the results are more of theoretical importance and the proven error bounds cannot be expected to be sharp, we mainly use simple, academic examples involving (block) diagonal matrices instead of the model problems from Section 2.6, as these best allow to discuss the influence of, e.g., the eigenvalue distribution of the matrix on the quality of the error bounds. Again, all experiments are performed in MATLAB using the implementation `FUNM_QUAD` [59] of the restarted Arnoldi method (and a modification thereof for the restarted harmonic Arnoldi method). Most of the experiments in this section have already been presented in the same or a similar form in [57].

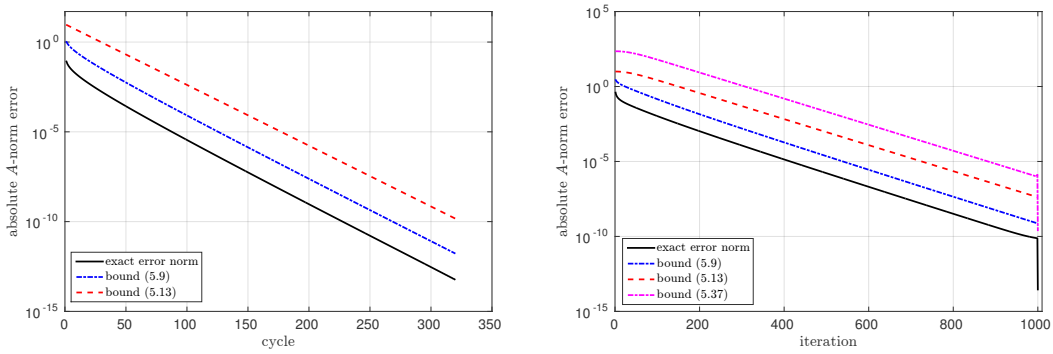


Figure 5.3: Comparison of the norm of the error and the error bounds from Lemma 5.5, Theorem 5.7 and Theorem 5.21 for diagonal A with eigenvalues chosen as Chebyshev points in $[10^{-2}, 10^{2}]$, $f(z) = z^{-1/2}$, restart length $m = 20$ (left) and unrestarted Arnoldi (right).

The first problem we consider is approximating $A^{-1/2}\boldsymbol{b}$, where $A \in \mathbb{C}^{1{,}000\times 1{,}000}$ is a diagonal matrix with eigenvalues chosen as Chebyshev points (i.e., zeros of the scaled Chebyshev polynomial of degree 1,000) in $[10^{-2}, 10^{2}]$ and $\boldsymbol{b}$ is the normalized vector of all ones. Obviously, A is Hermitian positive definite, so that the theory from Section 5.2 applies in this case. We report the convergence bound (5.13) from Theorem 5.7 as well as the bound (5.9) (where the integral is evaluated by adaptive Gauss–Kronrod quadrature, see, e.g., [74]) which can be expected to be sharper. In Figure 5.3 we report the results for restart length $m = 20$ and for the unrestarted Arnoldi method. In the second case, we also report the bound (5.37) from Theorem 5.21, which is not of interest for $m = 20$, as no superlinear convergence effects are expected to take place. We observe that all bounds capture the *rate* of convergence very accurately, while the magnitude of the error is overestimated by one (bound (5.9)) or two (bound (5.13)) orders of magnitude. The bound (5.37) even overestimates the error norm by three orders of magnitude, but is the only bound which can in a way capture the convergence to (approximately) machine precision in the last iteration of the unrestarted Arnoldi method. This can of course not be the case for the other two bounds which only ever predict linear convergence.

The standard bound for the error in CG which we used to prove Theorem 5.7 is obtained by bounding the CG polynomials by means of Chebyshev polynomials (see, e.g., [115]). When the eigenvalues of A are Chebyshev points (or lie close to these points), it is known that the speed of convergence of CG method is close to its worst case behavior, see, e.g. [103], and $\alpha_m(0)$ can thus be expected to be a very close estimate for the actual convergence factor. As the system $A\boldsymbol{x} = \boldsymbol{b}$ corresponding to the smallest shift $t = 0$ dominates the convergence of

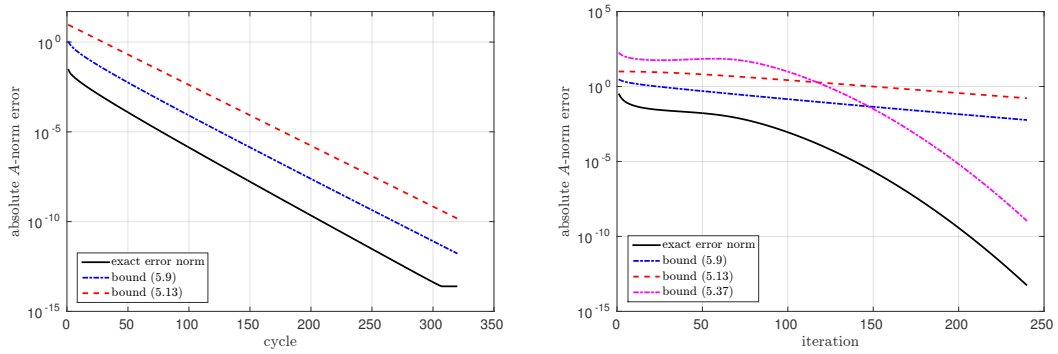


Figure 5.4: Comparison of the norm of the error and the error bounds from Lemma 5.5, Theorem 5.7 and Theorem 5.21 for diagonal A with eigenvalues chosen equidistantly spaced in $[10^{-2}, 10^2]$, $f(z) = z^{-1/2}$, restart length $m = 20$ (left) and unrestarted Arnoldi (right).

the restarted Arnoldi method, it is therefore expected that the overall convergence rate is predicted accurately. Next, we therefore modify the diagonal matrix A in such a way that the CG bound $\alpha_m(0)$ is no longer close to optimal. We do so by choosing the eigenvalues of A in the same interval $[10^{-2}, 10^2]$, but equidistantly spaced this time. All other parameters and the quantities we report stay the same as before. The resulting convergence curves and error bounds are depicted in Figure 5.4. For the restarted method, the behavior is very similar to what was observed for the matrix with Chebyshev eigenvalues, although the convergence slope is a little bit steeper than predicted by our bounds this time, but only very moderately so. For the unrestarted method however, we observe very different behavior. In an initial phase (until after about 75 iterations), the convergence rate is approximately as predicted by our bounds, but after that, the superlinear convergence behavior of Arnoldi's method starts to take place (see also [11]), and the error bounds (except for the bound (5.37) based on the CG residual norm) do not capture the actual behavior of the method anymore. This, together with the fact that the error is again overestimated by several orders of magnitude (and unfortunately the most by the bound which does capture the convergence slope accurately), already suggests that the bounds developed in this chapter are mainly of theoretical value and it is not advisable to use them as stopping criteria in practical computations.

In the next experiment, we compare the behavior of the restarted Arnoldi and restarted harmonic Arnoldi method for a positive real matrix. We do not report the error bounds from Section 5.5 for the restarted harmonic Arnoldi method here, as they are even worse than the bounds for the standard restarted Arnoldi method, both severely overestimating the error and the convergence slope. This comes as

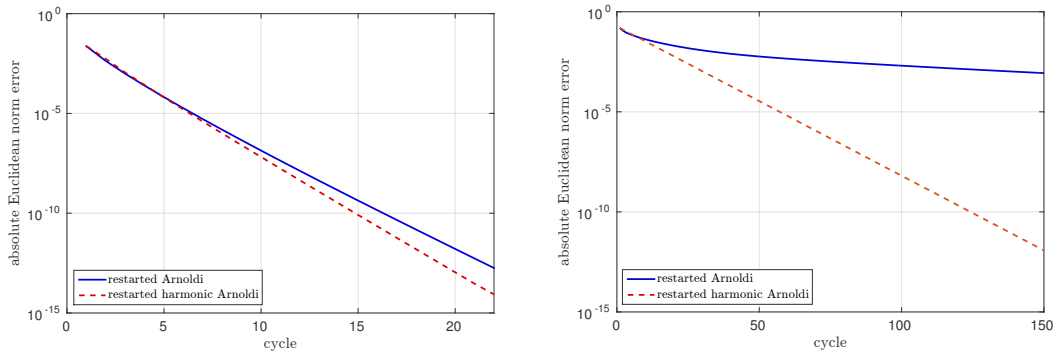


Figure 5.5: Convergence curves of the restarted Arnoldi method and the restarted harmonic Arnoldi method for a diagonal matrix A and $f(z) = z^{-1/2}$. The eigenvalues of A are chosen randomly in a disk of radius 1 centered at $1 + 10^{-1}$ (left) and $1 + 10^{-5}$ (right).

no surprise, as this phenomenon is also known for the bound for the convergence of (restarted) GMRES it is based upon. We therefore cannot expect our bounds to show better behavior (as we additionally overestimate the exact error norm, e.g., when only considering the convergence of the dominating linear system with $t = 0$). We again consider computing the inverse square root of a diagonal matrix $A \in \mathbb{C}^{1{,}000 \times 1{,}000}$ applied to the normalized vector of all ones. This time, we choose the diagonal entries of A of the form $\lambda_j = \alpha + r_j e^{2\pi i \theta_j}, j = 1, \ldots, 1{,}000$, where $\alpha > 1$ and the parameters r_j and θ_j are random variables chosen independently and uniformly distributed in $[0, 1]$. Therefore, all eigenvalues of A lie in a disk of radius one with center α. As $\alpha > 1$, all eigenvalues are contained in the right half-plane. As A is diagonal it is in particular normal, and therefore $\mathcal{W}(A)$ also lies in the right half-plane when all eigenvalues do. Thus, A is positive real. We test both methods for the two choices $\alpha = 1 + 10^{-1}$ and $\alpha = 1 + 10^{-5}$. The results of our experiment are given in Figure 5.5. Note that both methods converge, while our theory only guarantees this for the restarted harmonic Arnoldi method. For $\alpha = 1 + 10^{-1}$, the matrix A is quite well-conditioned, as no eigenvalues close to the origin can appear, and both methods converge very fast and behave almost the same. For the choice $\alpha = 1 + 10^{-5}$, the spectrum of A moves closer to the origin and the matrix is much worse conditioned than before. In this case, convergence of the restarted Arnoldi method critically slows down, while the restarted harmonic Arnoldi method still converges reasonably fast (albeit slower than for the better conditioned matrix corresponding to $\alpha = 1 + 10^{-1}$, as has to be expected). This example illustrates that in case of non-Hermitian positive real matrices, the restarted harmonic Arnoldi method may indeed behave substantially better than the standard restarted Arnoldi method, even when both

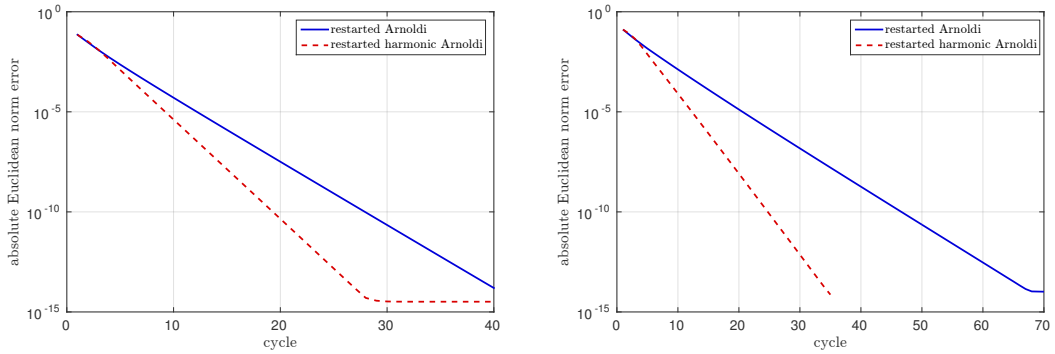


Figure 5.6: Convergence curves of the restarted Arnoldi method and the restarted harmonic Arnoldi method, where A is not diagonalizable with 2×2 Jordan blocks, and $f(z) = z^{-1/2}$. The details on the spectrum of A for the left and right plots are given in the text.

converge to $f(A)\boldsymbol{b}$. This shows that the advantage of this method is not only of theoretical nature for some pathological examples.

All matrices considered so far have been normal and in particular diagonalizable (even diagonal). We therefore compare the two methods in one last experiment involving a matrix which is *not* diagonalizable. We choose $A \in \mathbb{C}^{1,000\times 1,000}$ again, this time block-diagonal with 2×2 Jordan blocks

$$\begin{bmatrix} \lambda & 0 \\ 1 & \lambda \end{bmatrix}$$

on the diagonal. One easily checks that such a block is positive real if $\Re(\lambda) > 0.5$ (and thus A is positive real if this is fulfilled for all blocks). We again approximate the inverse square root of A applied to the normalized vector of all ones. We again tested both methods for two different (randomly produced) matrices A. In both cases, the imaginary parts of the values λ are chosen uniformly distributed in $[-10, 10]$. The real parts are chosen uniformly distributed in $[0.6, 0.8]$ and $[0.5001, 0.5099]$, respectively. The results of both experiments are given in Figure 5.6. We observe reasonably fast convergence for both methods, especially for the first, better conditioned system. For the second system, the harmonic Arnoldi method outperforms the standard method by a factor of about two, concerning iteration numbers.

CHAPTER 6

ERROR ESTIMATES IN KRYLOV METHODS

In Chapter 5 we have shown how to derive a priori error bounds for Arnoldi's method for Stieltjes matrix functions. However, as the numerical experiments in Section 5.7 reveal (and as can be expected from similar experience with the error bounds for linear systems our results are based upon), these bounds tend to severely overestimate the order of magnitude of the error and (depending on the eigenvalue distribution of the matrix A) may also wrongly predict the convergence slope. Therefore, they are not feasible for use as stopping criteria for iterative methods in most practical situations, as this would typically lead to a relatively high number of unnecessary Arnoldi iterations which are performed in spite of the error norm already being well below the desired tolerance. Therefore, we now show how to compute error estimates during Arnoldi's method which better capture the actual behavior of the error (and can be shown to be lower and upper bounds for the exact error norm in certain situations) and are therefore better suited as stopping criteria. In [65,66], approaches for computing such error estimates (or bounds) for matrix function computations have been presented, but the results given there only apply to rational functions in partial fraction form (another approach for rational functions, which we investigate in more detail later in this chapter is presented in [61]). Therefore, it is of interest to construct such error bounds for more general classes of matrix functions. The bounds we present here for this purpose are largely based on the close relation between Gauss quadrature and the Lanczos process, see [72–74], which we briefly review in Section 6.1. In Section 6.2, we describe how this relation can be used to compute error bounds for bilinear forms $\boldsymbol{u}^H h(A)\boldsymbol{v}$ defined by a matrix function $h(A)$. By rewriting the error norm in Arnoldi's method for Stieltjes functions as such a bilinear form, we show how these techniques can be used for bounding the Arnoldi error norm in Section 6.3. As the naive application of this approach

leads to very high additional computational cost for the computation of the error bounds (multiple additional matrix-vector products per Arnoldi iteration), we describe how to compute the bounds with cost independent of the matrix size n and iteration number m in Section 6.4. In Section 6.5 we briefly describe how the techniques from the previous sections can be transferred to non-Hermitian matrices (and functions which are not Stieltjes functions, like, e.g., the matrix exponential) although in this case one obtains only estimates for the error (instead of bounds) in general, and the cost for the computation of these bounds grows with the number m of iterations in Arnoldi's method (but is still independent of the matrix size n). In Section 6.6, we apply the developed techniques to the model problems from Section 2.6 to illustrate the quality of our error estimates and investigate their dependence on certain parameters.

The results presented in this section related to Stieltjes functions are submitted for publication; see [63].

6.1 Relation between Gauss quadrature and the Lanczos process

In this section we briefly review the connection between Gauss quadrature and the Lanczos process. For this, consider the sequence of *Lanczos polynomials*, i.e., the polynomials p_{k-1} with

$$p_{k-1}(A)\boldsymbol{v}_1 = \boldsymbol{v}_k \text{ and } \deg p_{k-1} = k-1. \tag{6.1}$$

One can show that these polynomials form an orthonormal set with respect to an inner product depending on $\operatorname{spec}(A)$. For ease of presentation, we assume that all eigenvalues of A are distinct in the following. We will, however, briefly touch on the necessary modifications in case that A has multiple eigenvalues right after the statement of the central Theorem 6.1.

Theorem 6.1. *Let $A \in \mathbb{C}^{n\times n}$ be Hermitian with eigenvalue decomposition $A = Q\Lambda Q^H$ and let $p_{k-1}, k = 1, 2, \dots$ be the Lanczos polynomials* (6.1)*. Define the function*

$$\alpha(t) = \begin{cases} 0, & \text{if } t < \lambda_1, \\ \sum_{j=1}^{i} |\hat{\boldsymbol{v}}(j)|^2, & \text{if } \lambda_i \le t < \lambda_{i+1}, \\ \sum_{j=1}^{n} |\hat{\boldsymbol{v}}(j)|^2, & \text{if } \lambda_n \le t, \end{cases} \tag{6.2}$$

where $\lambda_{\min} = \lambda_1 < \lambda_2 < \cdots < \lambda_n = \lambda_{\max}$ denote the (sorted) eigenvalues of A and $\hat{\boldsymbol{v}} = Q^H\boldsymbol{v}_1$. Then the polynomials p_{k-1} are orthonormal with respect to the

inner product

$$\begin{aligned}(p,q)_\alpha &= \int_a^b p(t)q(t)d\alpha(t) \\ &= \hat{\boldsymbol{v}}^H p(A)^H q(A)\hat{\boldsymbol{v}}, \end{aligned} \tag{6.3}$$

where $a \leq \lambda_{\min}$ *and* $b \geq \lambda_{\max}$.

Proof. See [74, Theorem 4.2]. □

Remark 6.2. In case that the eigenvalues of A are not pairwise distinct, the notation in Theorem 6.1 has to be adapted as follows. Denoting by $\widehat{\lambda}_1 < \cdots < \widehat{\lambda}_{\widehat{n}}$ the distinct eigenvalues, there exist corresponding eigenvectors $\widehat{\boldsymbol{v}}_j$ and coefficients $\eta_j, j = 1, \ldots, \widehat{n}$ such that

$$\boldsymbol{v}_1 = \sum_{j=1}^{\widehat{n}} \eta_j \widehat{\boldsymbol{v}}_j,$$

as A is Hermitian positive definite and thus, in particular, its eigenvectors form a basis of $\mathbb{C}^n$. The step function α from (6.2) is then changed to

$$\alpha(t) = \begin{cases} 0, & \text{if } t < \widehat{\lambda}_1, \\ \sum_{j=1}^{i} |\eta_j|^2, & \text{if } \widehat{\lambda}_i \leq t < \widehat{\lambda}_{i+1}, \\ \sum_{j=1}^{\widehat{n}} |\eta_j|^2, & \text{if } \widehat{\lambda}_{\widehat{n}} < t, \end{cases}$$

and all results presented in the following apply in a straightforward way.

Theorem 6.1 states that the Lanczos process generates a (finite) sequence of orthonormal polynomials with respect to the inner product (6.3) defined via the function α. In practical situations, α is not known explicitly, as it requires knowledge of all eigenvalues and eigenvectors of A. Interestingly, by using the Lanczos process, it is possible to find Gauss quadrature rules corresponding to $[a, b]$ and α without the explicit knowledge of α. The following theorem is derived by exploiting the relationship between Gauss quadrature and the eigenvalues and eigenvectors of tridiagonal matrices, see Section 2.5.

Theorem 6.3. *Let* $A \in \mathbb{C}^{n\times n}$ *be Hermitian positive definite with smallest and largest eigenvalue* $\lambda_{\min}$ *and* $\lambda_{\max}$*, respectively, let* $a \leq \lambda_{\min}$ *and* $b \geq \lambda_{\max}$*, let* $\boldsymbol{v}_1 \in \mathbb{C}^n$ *with* $\|\boldsymbol{v}_1\|_2 = 1$*, let* h *be a function defined on* $[a, b]$ *and let* α *be defined as in* (6.2)*. Let* $t_\ell, \omega_\ell, \ell = 1, \ldots, k$ *be the nodes and weights of the* k*-point Gauss quadrature rule for approximating*

$$\int_a^b h(t)d\alpha(t). \tag{6.4}$$

Then

$$\sum_{\ell=1}^{k} \omega_\ell h(t_\ell) = \hat{\boldsymbol{e}}_1^H h(H_k)\hat{\boldsymbol{e}}_1, \tag{6.5}$$

where H_k is the tridiagonal matrix obtained by k steps of the Lanczos process, Algorithm 2.2, applied to A and $\boldsymbol{v}_1$.

Proof. See [74, Theorem 6.6] □

In the same way, the $(k+1)$-point Gauss–Radau quadrature rule (with one node fixed at $\lambda_{\min}$) can be evaluated as $\hat{\boldsymbol{e}}_1^H h(\widetilde{H}_{k+1})\hat{\boldsymbol{e}}_1$, with the modified tridiagonal matrix

$$\widetilde{H}_{k+1} = \begin{bmatrix} H_k & h_{k+1,k}\hat{\boldsymbol{e}}_k \\ h_{k+1,k}\hat{\boldsymbol{e}}_k^H & \boldsymbol{d}(k) \end{bmatrix}, \text{ where } \boldsymbol{d} = h_{k+1,k}^2 (H_k - \lambda_{\min} I)^{-1}\hat{\boldsymbol{e}}_k; \tag{6.6}$$

see [74].

By Theorem 6.3, we can evaluate a Gauss quadrature rule for the function h without even explicitly computing the nodes and weights of the corresponding rule, by evaluating h on a tridiagonal matrix. In the next section we will show why and how bilinear forms $\boldsymbol{u}^H h(A)\boldsymbol{v}$ can be interpreted as Riemann–Stieltjes integrals of the form (6.4).

6.2 Bounds and estimates for bilinear forms $\boldsymbol{u}^H h(A)\boldsymbol{v}$

Let $A \in \mathbb{C}^{n\times n}$ be Hermitian positive definite with smallest and largest eigenvalue $\lambda_{\min}$ and $\lambda_{\max}$, respectively, let $[a, b]$ with $a \le \lambda_{\min}$ and $b \ge \lambda_{\max}$ and let h be a function defined on $[a, b]$ as before. Given vectors $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{C}^n$, we are interested in approximating the bilinear form $\boldsymbol{u}^H h(A)\boldsymbol{v}$. To do so, we mainly follow the presentation in [74, Chapter 7]. With $\boldsymbol{q}_1, \ldots, \boldsymbol{q}_n$ denoting an orthonormal eigenbasis of A, we decompose $\boldsymbol{u}$ and $\boldsymbol{v}$ as

$$\boldsymbol{u} = \sum_{i=1}^{n} \beta_i \boldsymbol{q}_i \text{ and } \boldsymbol{v} = \sum_{i=1}^{n} \eta_i \boldsymbol{q}_i.$$

Inserting this relation into $\boldsymbol{u}^H h(A)\boldsymbol{v}$ and using the eigendecomposition $A = Q\Lambda Q^H$, we find

$$\boldsymbol{u}^H h(A)\boldsymbol{v} = \boldsymbol{u}^H Q h(\Lambda) Q^H \boldsymbol{v} = \boldsymbol{\beta}^H h(\Lambda)\boldsymbol{\eta} = \sum_{i=1}^{n} \overline{\beta}_i \eta_i h(\lambda_i). \tag{6.7}$$

The sum on the right-hand side of (6.7) can be interpreted as a Riemann–Stieltjes integral

$$\int_a^b h(t)\,\mathrm{d}\widetilde{\alpha}(t)$$

with respect to a piecewise constant step function $\widetilde{\alpha}$ (cf. also Example 2.9) given by

$$\widetilde{\alpha}(t) = \begin{cases} 0, & \text{if } t < \lambda_1, \\ \sum_{j=1}^{i} \overline{\beta}_j \eta_j, & \text{if } \lambda_i \leq t < \lambda_{i+1}, \\ \sum_{j=1}^{n} \overline{\beta}_j \eta_j, & \text{if } \lambda_n \leq t. \end{cases}$$

In case of a quadratic form $\boldsymbol{v}^H h(A)\boldsymbol{v}$, i.e., $\boldsymbol{u} = \boldsymbol{v}$ in (6.7), the function $\widetilde{\alpha}$ simplifies to

$$\widetilde{\alpha}(t) = \begin{cases} 0, & \text{if } t < \lambda_1, \\ \sum_{j=1}^{i} |\eta_j|^2, & \text{if } \lambda_i \leq t < \lambda_{i+1}, \\ \sum_{j=1}^{n} |\eta_j|^2, & \text{if } \lambda_n \leq t. \end{cases} \tag{6.8}$$

If we use $\boldsymbol{v}$ as starting vector for the Lanczos process (for notational simplicity assuming that $\|\boldsymbol{v}\|_2 = 1$), we have $\boldsymbol{\eta} = \hat{\boldsymbol{v}}$ in Theorem 6.1, and the step function $\widetilde{\alpha}$ from (6.8) coincides with α from (6.2). In other words, the quadratic form $\boldsymbol{v}^H h(A)\boldsymbol{v}$ can be interpreted as a Riemann–Stieltjes integral with respect to the function α for which the Lanczos polynomials form an orthonormal sequence. Therefore, the corresponding Gauss quadrature rule (6.5) from Theorem 6.3 (or the Gauss–Radau rule obtained from replacing H_k by $\widetilde{H}_{k+1}$) is a natural choice for approximating it. A simple way of approximating quadratic forms $\boldsymbol{v}^H h(A)\boldsymbol{v}$ is therefore as follows. First, normalize $\boldsymbol{v}$ if necessary, yielding $\boldsymbol{v}_1$. Perform m steps of Algorithm 2.2 to obtain the tridiagonal matrix H_k (and modify this matrix according to (6.6) if Gauss–Radau quadrature is to be used), and then evaluate (6.5) to obtain an estimate corresponding to a k-point Gauss rule (or $(k+1)$-point Gauss–Radau rule). In the next section, we will show that the norm of the error in Arnoldi's method can be expressed as a quadratic form, so that the results from this and the preceding section apply in this case. Afterwards, in Section 6.4, we will show how to modify the simple approach described above so that it does not require k additional multiplications with A for performing a secondary Lanczos process.

6.3 Error bounds for Stieltjes functions of positive definite matrices

We begin this section by giving a straightforward characterization of the error norm in Arnoldi's method for Stieltjes matrix functions.

Lemma 6.4. *Let $A \in \mathbb{C}^{n\times n}$ be Hermitian positive definite, let $\boldsymbol{b} \in \mathbb{C}^n$, let f be a Stieltjes function of the form* (3.15) *and let $\boldsymbol{f}_m$ be the approximation for $f(A)\boldsymbol{b}$ obtained from m steps of Arnoldi's method. Then*

$$\|f(A)\boldsymbol{b} - \boldsymbol{f}_m\|_2^2 = \|\boldsymbol{b}\|_2^2\gamma_m^2 \boldsymbol{v}_{m+1}^H \widetilde{e}_m(A)^2 \boldsymbol{v}_{m+1}, \tag{6.9}$$

where $\widetilde{e}_m(z)$ is given by

$$\widetilde{e}_m(z) = \int_0^\infty \frac{1}{z+t}\,\mathrm{d}\widetilde{\mu}(t) \text{ with } \mathrm{d}\widetilde{\mu}(t) = \frac{1}{w_m(t)}\,\mathrm{d}\mu(t) \tag{6.10}$$

and γ_m, w_m are as defined in Theorem 3.5.

Proof. By Theorem 3.5 and Proposition 3.9 we have the representation

$$f(A)\boldsymbol{b} - \boldsymbol{f}_m = (-1)^{m+1}\|\boldsymbol{b}\|_2\gamma_m \int_0^\infty (A+tI)^{-1}\,\mathrm{d}\widetilde{\mu}(t)\boldsymbol{v}_{m+1} \tag{6.11}$$

for the error in Arnoldi's method. Taking (squared) norms in (6.11) gives

$$\begin{aligned}
\|f(A)\boldsymbol{b} - \boldsymbol{f}_m\|_2^2 &= \left(\|\boldsymbol{b}\|_2\gamma_m\widetilde{e}_m(A)\boldsymbol{v}_{m+1}\right)^H \left(\|\boldsymbol{b}\|_2\gamma_m\widetilde{e}_m(A)\boldsymbol{v}_{m+1}\right) \\
&= \|\boldsymbol{b}\|_2^2\gamma_m^2 \boldsymbol{v}_{m+1}^H \widetilde{e}_m(A)^H \widetilde{e}_m(A)\boldsymbol{v}_{m+1} \\
&= \|\boldsymbol{b}\|_2^2\gamma_m^2 \boldsymbol{v}_{m+1}^H \widetilde{e}_m(A)^2 \boldsymbol{v}_{m+1},
\end{aligned}$$

where the last equality holds because $\widetilde{e}_m(A)$ is Hermitian if A is Hermitian. □

The representation (6.9) of the Arnoldi error norm as a quadratic form allows to use Gauss (and Gauss–Radau) quadrature in the sense of (6.5) to compute approximations for it. In our situation, i.e., f a Stieltjes function and A Hermitian positive definite, we can prove that the approximations obtained this way are lower and upper bounds for the exact norm of the error.

Theorem 6.5. *Let $A \in \mathbb{C}^{n\times n}$ be Hermitian positive definite, let $\boldsymbol{b} \in \mathbb{C}^n$, let f be a Stieltjes function of the form* (3.15) *and let $\boldsymbol{f}_m$ be the mth Arnoldi approximation* (2.25) *to $f(A)\boldsymbol{b}$. Denote by $H_k^{(2)}$ the tridiagonal matrix resulting from k steps of the Lanczos process applied to A and $\boldsymbol{v}_{m+1}$ and by $\widetilde{H}_{k+1}^{(2)}$ the modification of $H_k^{(2)}$ according to* (6.6)*. Then*

$$\|\boldsymbol{b}\|_2^2\gamma_m^2 \hat{\boldsymbol{e}}_1^H \widetilde{e}_m\left(H_k^{(2)}\right)^2 \hat{\boldsymbol{e}}_1 \le \|f(A)\boldsymbol{b} - \boldsymbol{f}_m\|_2^2 \le \|\boldsymbol{b}\|_2^2\gamma_m^2 \hat{\boldsymbol{e}}_1^H \widetilde{e}_m\left(\widetilde{H}_{k+1}^{(2)}\right)^2 \hat{\boldsymbol{e}}_1, \tag{6.12}$$

where $\widetilde{e}_m(z)$ is the error function given in (6.10) *and γ_m is as defined in Theorem 3.5.*

Proof. By Theorem 6.3, the quantity $\hat{\boldsymbol{e}}_1^H \widetilde{e}_m \left(H_k^{(2)}\right)^2 \hat{\boldsymbol{e}}_1$ corresponds to a k-point Gauss quadrature rule for the Riemann–Stieltjes integral

$$\int_a^b \widetilde{e}_m(z)^2 \, \mathrm{d}\alpha(z),$$

where α is given by (6.2). In the same way $\hat{\boldsymbol{e}}_1^H \widetilde{e}_m \big(\widetilde{H}_{k+1}^{(2)}\big)^2 \hat{\boldsymbol{e}}_1$ corresponds to a $(k+1)$-point Gauss–Radau rule. As $\widetilde{e}_m(z)$ is a Stieltjes function by Proposition 3.9, and thus, according to Proposition 2.17(i), completely monotonic, $\widetilde{e}_m(z)^2$ is also completely monotonic by Proposition 2.17(ii), although it is not a Stieltjes function in general. The bracketing property (2.53) from Corollary 2.48 then gives the desired result. □

Theorem 6.5 suggests a simple way of computing error bounds for the Arnoldi approximation to $f(A)\boldsymbol{b}$, albeit one that is hardly feasible in practice. Directly evaluating the leftmost and rightmost expression in (6.12) to bound the error in the mth step of Arnoldi's method demands an *additional* k matrix vector products for computing $H_k^{(2)}$. Therefore, the computation of the error bounds requires the same amount of computational work as advancing the Arnoldi iteration from step m to step $m+k$. This is an unacceptably high cost, especially if one wants to compute error bounds in each iteration of Arnoldi's method to monitor at what point the desired accuracy is reached. How this can be circumvented will be the topic of Section 6.4. Prior to that, we address another issue that arises when trying to use (6.12) for computing error bounds. The function $\widetilde{e}_m(z)$ is not available in an explicit closed form, so that it has to be evaluated, e.g, by numerical quadrature, just as in the implementation of the restarted Arnoldi method in Chapter 4. While in Chapter 4 we were mainly interested in using a convergent quadrature rule which gives an accurate enough representation of the error to be approximated, this time it is also important to know whether these approximations give lower or upper bounds for the exact value of the integral, because otherwise one cannot be sure that the approximations computed for the quantities on the left and right of (6.12) are still bounds (rather than only estimates) for the error. In the following we show that it suffices to choose a quadrature rule that gives lower (or upper) bounds for the value of $\widetilde{e}_m(z)$ in the scalar case, because this property will carry over to the matrix case. Note that this result is not as trivial as it may appear at first sight, as it relies on the fact that A (and thus H_k) is Hermitian positive definite and does in general not hold in the non-Hermitian case.

Proposition 6.6. *Let the assumptions of Theorem 3.4 hold and let $H \in \mathbb{C}^{m\times m}$ be any Hermitian positive definite matrix. Further, let $t_\ell, \omega_\ell \in \mathbb{R}, \ell = 1, \ldots, k$ be the nodes and weights of a quadrature rule for which*

$$\sum_{\ell=1}^{k} \frac{\omega_\ell}{z + t_\ell} \leq \widetilde{e}_m(z) \text{ for } z \in \mathbb{R}^+. \tag{6.13}$$

Then

$$\hat{\boldsymbol{e}}_1^H \left(\sum_{\ell=1}^{k} \omega_\ell (H + t_\ell I)^{-1} \right)^2 \hat{\boldsymbol{e}}_1 \leq \hat{\boldsymbol{e}}_1^H \widetilde{e}_m(H)^2 \hat{\boldsymbol{e}}_1.$$

The result holds analogously for quadrature rules which give upper bounds. In particular, the result applies to the matrices $H_k^{(2)}$ and $\widetilde{H}_{k+1}^{(2)}$ from Theorem 6.5.

Proof. Using the spectral decomposition $H = UDU^H$ with unitary U and diagonal D and defining the shorthand notation $\boldsymbol{u} = U^H \hat{\boldsymbol{e}}_1$, we have

$$\begin{aligned} \hat{\boldsymbol{e}}_1^H \left(\sum_{\ell=1}^{k} \omega_\ell (H + t_\ell I)^{-1} \right)^2 \hat{\boldsymbol{e}}_1 &= \boldsymbol{u}^H \left(\sum_{\ell=1}^{k} \omega_\ell (D + t_\ell I)^{-1} \right)^2 \boldsymbol{u} \\ &= \sum_{i=1}^{m} |\boldsymbol{u}_i|^2 \left(\sum_{\ell=1}^{k} \frac{\omega_\ell}{d_{ii} + t_\ell} \right)^2. \end{aligned} \tag{6.14}$$

Using (6.13) we can bound the right-hand side of (6.14) by

$$\sum_{i=1}^{m} |\boldsymbol{u}_i|^2 \left(\sum_{\ell=1}^{k} \frac{\omega_\ell}{d_{ii} + t_\ell} \right)^2 \leq \sum_{i=1}^{m} |\boldsymbol{u}_i|^2 \widetilde{e}_m(d_{ii})^2 = \boldsymbol{u}^H \widetilde{e}_m(D)^2 \boldsymbol{u} = \hat{\boldsymbol{e}}_1^H \widetilde{e}_m(H)^2 \hat{\boldsymbol{e}}_1,$$

which concludes the proof for lower bounds. The modifications necessary for proving the result for upper bounds are straightforward. The matrix $H_k^{(2)}$ is obviously Hermitian positive definite, as A is Hermitian positive definite and $H_k^{(2)} = V^H A V$ for a matrix V of full (column) rank. For $\widetilde{H}_{k+1}^{(2)}$, note that the modification (6.6) again results in a Hermitian matrix. As its eigenvalues are the nodes of a Gauss–Radau quadrature rule, they are known to lie in $[\lambda_{\min}, \lambda_{\max}]$ (when the fixed quadrature node a is chosen such that $a \leq \lambda_{\min}$ or $a \geq \lambda_{\max}$, which is the case in our setting), see, e.g., [69]. Therefore, $\widetilde{H}_{k+1}^{(2)}$ is also a Hermitian positive definite matrix and the result of the proposition applies. □

We note that while the result of Proposition 6.6 is important in the sense that it guarantees that it is possible to really compute lower and upper bounds for the error in Arnoldi's method by properly combining two rules that each give a lower (or upper) bound in (6.12), the numerical experiments reported in Section 6.6 show that the error in the *inner* quadrature rule, i.e., the rule for approximating $\widetilde{e}_m(z)$, is typically negligible compared to the error of the *outer* Gauss quadrature rule used to approximate the bilinear form (6.9) for reasonable choices of parameters, so that in most situations, the computed quantities can still be trusted to be bounds for the error norm, even when no special care is devoted to choosing the inner quadrature rule properly.

We end this section by commenting on the relation of the results presented here to the results from [61]. In [61], an approach very similar to what is presented here is established for linear systems and, building on this, for rational functions in partial fraction form (with poles on the negative real axis). These rational functions belong to the class of Stieltjes functions, cf. also Example 2.14, and the techniques presented in this thesis and in [61] in fact lead to exactly the same results in this case. Instead of using the results from this chapter, one could also apply the approach of [61] to other functions by first approximating f by a suitable rational function $r \approx f$ and then working with r for computing error estimates (similar to Algorithm 4.2 from [3], where one replaces f by a rational function to allow restarting). Using our approach, however, has the advantage that it circumvents the (potentially costly or complicated) a priori construction of a rational approximation for f and just works with f directly. In addition, when one is interested in computing guaranteed error bounds in the first place, one typically wants these to bound the error corresponding to $f(A)\boldsymbol{b}$. Using the approach from [61] would only yield bounds for the error corresponding to $r(A)\boldsymbol{b}$. As long as one does not have information on the sign of the remainder term in the rational approximation (and, in general, the remainder will change sign on the interval $[\lambda_{\min}, \lambda_{\max}]$), one can therefore not relate these bounds to $f(A)\boldsymbol{b}$ directly. Therefore, our approach, while similar in spirit to what was done in [61], has additional advantages which warrant its closer investigation in this thesis.

6.4 Computing error bounds with low computational cost

In this section, we show how to compute the quantities from (6.12) with computational cost independent of the number m of iterations performed thus far in Arnoldi's method and the dimension n of the matrix A, thus making it feasible to evaluate the resulting bounds in each iteration of Arnoldi's method for monitoring progress of the method. The main idea to reach this goal relies on the fact that we only need to know the tridiagonal matrix $H_k^{(2)}$ resulting from applying k steps of Algorithm 2.2 to A and $\boldsymbol{v}_{m+1}$, but not the corresponding Arnoldi basis vectors. The tridiagonal matrix can be computed efficiently, as stated by the following theorem from [61]. We state the result in its original form here, as this is all we need right now. We will present several more general versions of it in Section 6.5 and Chapter 7, but giving the lengthy proofs of the more general versions here would deviate from the main topic of this section.

Theorem 6.7. *Let $A \in \mathbb{C}^{n\times n}$ be Hermitian positive definite, let $\boldsymbol{v}_1 \in \mathbb{C}^n$ and let H_{m+k+1} be the tridiagonal matrix resulting from $m+k+1$ steps of the Lanczos process for A and $\boldsymbol{v}_1$. Let $\hat{k} = \min\{m, k\}$ and denote by $\widetilde{H}$ the lower right $(k+$*

$\hat{k}+1) \times (k+\hat{k}+1)$ *sub-block of* H_{m+k+1}. *Further, let* $\hat{H}$ *denote the tridiagonal matrix resulting from* k *steps of the Lanczos process applied to* $\widetilde{H}$ *and* $\hat{\boldsymbol{e}}_{\hat{k}+1}$. *Then* $\hat{H} = H_k^{(2)}$, *where* $H_k^{(2)}$ *denotes the matrix resulting from* k *iterations of the Lanczos process for* A *and* $\boldsymbol{v}_{m+1}$.

Proof. See [61, Theorem 4.1]. □

Theorem 6.7 states that the matrix $H_k^{(2)}$ can be computed by performing k steps of a secondary Lanczos process with a tridiagonal matrix of size at most $(2k+1) \times (2k+1)$, i.e., with computational cost $\mathcal{O}(k^2)$, independent of the size of the original matrix. The price one pays for this reduction in computational complexity is that the error bounds for step m are not available immediately when performing this step, but only later at step $m+k+1$ (if Gauss quadrature with k nodes is used for computing the bounds). Therefore, there is a trade-off between accuracy of the error bound (which implies a higher number k of quadrature nodes) and timely availability of the bounds (which implies using as few nodes as possible). We further comment on this trade-off when investigating the dependency of the method on the number of quadrature nodes used in the numerical experiments reported in Section 6.6.

Apart from $H_k^{(2)}$, the computation of the error bounds by (6.12) requires one to be able to evaluate $\widetilde{e}_m$ and thus the nodal polynomial $w_m(t)$, at least at the quadrature nodes used for approximating $\widetilde{e}_m$. As $w_m(t)$ is a polynomial of degree m, a naive approach of evaluating it at each of the ℓ quadrature nodes used in the inner quadrature rule would require $\mathcal{O}(m\ell)$ arithmetic operations (notwithstanding the fact that the explicit formula for $w_m(t)$ requires computing the eigenvalues of H_m in the first place), so that the cost of evaluating $\widetilde{e}_m$ would grow with the iteration number m. In addition, this approach could potentially become numerically unstable for larger values of m. We can circumvent this problem by choosing the quadrature nodes $t_i, i = 1, \ldots, \ell$ in advance and fixing them throughout all iterations, as $1/w_m(t_i)$ can be updated from $1/w_{m-1}(t_i)$ with computational cost $\mathcal{O}(1)$ as follows. By exploiting the relation (4.12) (adapted to our situation, i.e., replacing $tI - H_m$ by $H_m + tI$), one immediately sees that $1/w_m(t)$ is a multiple of the bottom left entry of the inverse of the shifted matrix $H_m + tI$. As H_m is Hermitian and tridiagonal (and so are its shifted versions), one can easily give a recurrence relation for this entry. When performing Gaussian elimination to solve

$$(H_m + t_i I)\boldsymbol{z}_m(t_i) = \hat{\boldsymbol{e}}_1, \tag{6.15}$$

with L_m denoting the resulting lower triangular matrix and $\boldsymbol{u}_m$ the vector on which the same elimination steps have been performed (starting from $\hat{\boldsymbol{e}}_1$), one obviously has

$$L_m = \begin{bmatrix} L_{m-1} & * \\ * & * \end{bmatrix} \text{ and } \boldsymbol{u}_m = \begin{bmatrix} \boldsymbol{u}_{m-1} \\ * \end{bmatrix},$$

with $*$ denoting unknown quantities. The quantities with lower index $m-1$ result from the elimination for the system (6.15) from the previous Arnoldi step. Therefore, the last entry of the solution of (6.15) can be constructed when the two quantities $\ell_{m-1,m-1}$ and $\boldsymbol{u}_{m-1}(m-1)$ are kept track of from the last iteration. One then has

$$\boldsymbol{u}_m(m) = \boldsymbol{u}_{m-1}(m-1) \cdot \frac{h_{m+1,m}}{\ell_{m,m}} \text{ where } \ell_{m,m} = (h_{m,m} + t_i) - \frac{h_{m+1,m}^2}{\ell_{m-1,m-1}},$$

so that all quantities necessary for computing $1/w_m(t_i)$ from (4.12) with cost $\mathcal{O}(1)$ are available. We note that this approach in a way corresponds to what is done in certain formulations of the (shifted) CG algorithm, see, e.g., [61, 62, 115], with the difference that the necessary quantities do not arise naturally during the computations in our case, as we are only implicitly working with shifted linear systems. This approach only requires to store $2(k+1)$ scalar values for each quadrature point t_i (as one needs to be able to retrieve the values from iteration m in iteration $m+k+1$ for computing the retrospective error bound), thus resulting in overall additional computational cost of order $\mathcal{O}(\ell)$ and storage cost of order $\mathcal{O}(k\ell)$, which is negligible in comparison to the computational cost and storage requirements of Algorithm 2.2 for reasonable values of k and ℓ. The requirement that the nodes for the inner quadrature rule have to be fixed in advance and cannot be changed during the execution of the algorithm (otherwise, one needs to recompute the values $1/w_m(t_i)$ from scratch for all new quadrature nodes, resulting in a cost of $\mathcal{O}(m\ell)$) is not a big disadvantage. One generally does not need a very high number ℓ of inner quadrature nodes to obtain an approximation error in the same order of magnitude as the one of the outer quadrature rule; see also the results presented in Section 6.6. In addition, even the recomputation of all values with cost $\mathcal{O}(m\ell)$ is in general affordable should it really become necessary.

Algorithm 6.1 summarizes our approach of computing retrospective error bounds in the Lanczos method for $f(A)\boldsymbol{b}$. The iteration stops once the upper bound of the error lies below the specified tolerance and the current iterate $\boldsymbol{f}_m$ is formed and returned (even though the error bound corresponds to the iterate $m-k-1$, we know by Theorem 5.10 that the (Euclidean) error norm of the Arnoldi approximations is monotonically decreasing for Hermitian positive definite A, so that the error norm of $\boldsymbol{f}_m$ will also lie below the specified tolerance and will in general be smaller than the one of $\boldsymbol{f}_{m-k-1}$). In order to not make the notation overly complicated we assume that the quantities d_i and ρ_i which keep track of the values necessary to retrieve the values of $1/w_m(t_i)$ are stored as full size vectors, although it suffices to keep the values from the last $k+1$ iterations in an actual implementation. Algorithm 6.1 is given in such a way that it only terminates when the desired tolerance is reached. It is of course possible to also specify an upper bound $m_{\max}$ for the number of iterations to be performed.

Algorithm 6.1: Lanczos method for $f(A)\boldsymbol{b}$ with error bounds

Given: A, $\boldsymbol{b}$, f, k, ℓ, `tol`, $\lambda_{\min}$

Choose quadrature nodes/weights $(t_i, \omega_i)_{i=1,\dots,\ell}$ for inner quadrature.
$d_i(0) \leftarrow 1, \rho_i(0) \leftarrow 1, i = 1, \dots, \ell$.
$h_{1,0} \leftarrow 0$.
$\boldsymbol{v}_1 \leftarrow \frac{1}{\|\boldsymbol{b}\|_2}\boldsymbol{b}$
for $m = 1, 2, \dots$ **do**
  $\boldsymbol{w}_m \leftarrow A\boldsymbol{v}_m - h_{m,m-1}\boldsymbol{v}_{m-1}$
  $h_{m,m} \leftarrow \boldsymbol{v}_m^H \boldsymbol{w}_m$
  $\boldsymbol{w}_m \leftarrow \boldsymbol{w}_m - h_{m,m}\boldsymbol{v}_m$
  $h_{m+1,m} \leftarrow \|\boldsymbol{w}_m\|_2$
  if $h_{m+1,m} = 0$ **then**
    $\boldsymbol{f}_m \leftarrow \|\boldsymbol{b}\|_2 V_m f(H_m)\hat{\boldsymbol{e}}_1$.
    Stop.
  $\boldsymbol{v}_{m+1} \leftarrow \frac{1}{h_{m+1,m}}\boldsymbol{w}_m$
  for $i = 1, \dots, \ell$ **do**
    $d_i(m) \leftarrow (h_{m,m} + t_i) - \frac{h_{m+1,m}^2}{d_i(m-1)}$
    $\rho_i(m) \leftarrow \rho_i(m-1) \cdot \frac{h_{m+1,m}}{d_i(m)}$
  $\hat{k} \leftarrow \min\{m+1, k+1\}$.
  Let $\widetilde{H}$ be the lower right $(k+\hat{k}) \times (k+\hat{k})$ sub-block of H.
  Perform k steps of Algorithm 2.2 for $\widetilde{H}$ and $\hat{\boldsymbol{e}}_{\hat{k}}$, yielding $\hat{H}$.
  Modify $\hat{H}$ according to (6.6), yielding $\bar{H}$.
  $\texttt{lower\_bound} \leftarrow \|\boldsymbol{b}\|_2^2 \hat{\boldsymbol{e}}_1^H \left(\sum_{i=1}^{\ell} \omega_i \rho_i (m-k-1)(\hat{H} + t_i I)^{-1}\right)^2 \hat{\boldsymbol{e}}_1$
  $\texttt{upper\_bound} \leftarrow \|\boldsymbol{b}\|_2^2 \hat{\boldsymbol{e}}_1^H \left(\sum_{i=1}^{\ell} \omega_i \rho_i (m-k-1)(\bar{H} + t_i I)^{-1}\right)^2 \hat{\boldsymbol{e}}_1$
  if `upper_bound` $\leq$ `tol` **then**
    $\boldsymbol{f}_m \leftarrow \|\boldsymbol{b}\|_2 V_m f(H_m)\hat{\boldsymbol{e}}_1$.
    Stop.

The next result summarizes the additional computational cost of Algorithm 6.1 in comparison to Algorithm 2.2.

Lemma 6.8. *Performing Algorithm 6.1 instead of Algorithm 2.2 (plus the computation of $\boldsymbol{f}_m$) for $A \in \mathbb{C}^{n\times n}$ and $\boldsymbol{b} \in \mathbb{C}^n$ requires an additional computational cost of the order $\mathcal{O}(k^2 + k\ell)$ per iteration and thus an overall additional work of $\mathcal{O}(m_{\max}k^2 + m_{\max}k\ell)$, if $m_{\max}$ iterations are necessary to reach the desired accuracy. In particular, the additional cost in the mth iteration is independent of both m and n.*

Proof. The initializations in line 1 and 2 of Algorithm 6.1 have cost $\mathcal{O}(\ell)$, assuming that the nodes and weights of the quadrature rule are available and do not need to be computed by a separate algorithm. Line 3–13 (ignoring the computation of $\boldsymbol{f}_m$ in line 11) exactly correspond to the Lanczos process given in Algorithm 2.2. The **for** loop in line 14–16 has computational cost $\mathcal{O}(\ell)$, as the update formulas for d_i and ρ_i only require a fixed number of scalar operations. Line 17 has cost $\mathcal{O}(1)$. Line 18 has cost $\mathcal{O}(k)$ and line 19 has cost $\mathcal{O}(k^2)$, as $\widetilde{H}$ is tridiagonal and matrix vector products with it therefore have cost $\mathcal{O}(k)$. Line 20 again has cost $\mathcal{O}(k)$ for solving the linear system with $\widetilde{H}$. The computation of the lower and upper bounds in line 21 and 22, respectively, requires $\mathcal{O}(k\ell)$ operations. Adding up the cost of all individual lines and noting that $\mathcal{O}(\ell), \mathcal{O}(k) \subset \mathcal{O}(k\ell)$ gives the desired result. □

Lemma 6.8 shows that the cost of computing error bounds for the Arnoldi approximation for Hermitian positive definite A by Algorithm 6.1 is independent of n. If n is large and k and ℓ are small in comparison, the additional cost is completely negligible. In the numerical experiments reported in Section 6.6 we demonstrate that values of k between 5 and 20 and values of ℓ below 100 are typically sufficient to compute accurate error bounds, also for large matrix dimension n. Although it is rather difficult to make any precise statement on the accuracy of the computed error bounds, one can of course expect their quality to depend on $\kappa(A)$, the condition number of the matrix A. Therefore, for large $\kappa(A)$, higher values of k and ℓ might be necessary to obtain satisfactory results.

We conclude this section by briefly commenting on the situation when using a restarted Arnoldi method (like, e.g., the one from Chapter 4) instead of the full Arnoldi method. In this case, one cannot use the Lanczos restart recovery from Theorem 6.7 to compute error bounds for all iterations of the method, but only for those iterations for which the next $k+1$ iterations belong to the same restart cycle, because the construction from Theorem 6.7 requires the tridiagonal matrix resulting from $k+1$ further steps of the Lanczos method and the result does not hold any longer if one restarts the method in between. Therefore, if m denotes the restart length, we can only compute error bounds for the first $m-k-1$ iterations of each cycle by the approach described before. However, in the restarted case, one is not that reliant on error bounds as for the full Arnoldi method, as one already computes error estimates in a natural way: In each restart cycle, one aims to approximate the error of the iterate from the last restart cycle. Therefore, assuming that the method computes sufficiently accurate approximations, the norm of the additive correction computed in cycle j can be interpreted as an estimate for the norm of the error of the iterate from cycle $j-1$ and thus gives a first hint at the progress of the method. However, if one just uses the approach from Chapter 4 to compute a correction for $\boldsymbol{f}_m^{(j-1)}$ and then computes its norm, one has no guarantee that the resulting value is an upper or lower bound for the

error norm or that it is a sufficiently accurate approximation at all. To obtain bounds, at least for the iterate computed at the end of the last restart cycle, we can use the result of Theorem 6.5 directly, without need for restart recovery. When restarting the Arnoldi method, we perform the Lanczos process in the next, jth restart cycle with the matrix A and the vector $\boldsymbol{v}_{m+1}^{(j-1)}$. This is exactly what is needed for computing the tridiagonal matrix used in (6.12). While in the full Arnoldi method, using Theorem 6.5 directly means many additional matrix vector multiplications which do not advance the primary iteration, we are now in a situation where the matrix vector multiplications performed by the next cycle of the primary Lanczos process and those needed for the computation of the error bounds are exactly the same. Therefore, if we compute the values on the left- and right-hand side of (6.12) at the end of restart cycle j (with restart length m), this corresponds to an approximation of the norm of the error after cycle $j-1$ by an m-point Gauss and $(m+1)$-point Gauss–Radau rule, respectively. This can also be interpreted the other way around: One performs k iterations of a secondary Lanczos process to bound the error of the current iterate. If the bound shows that the iterate does not yet fulfill the accuracy requirement, the iterations performed in the secondary method are not "lost", but can be used to begin the next restart cycle. We will also give some examples for the bounds obtained this way in the numerical experiments reported in Section 6.6.

6.5 Extension to non-Hermitian matrices

In this section, we will briefly sketch how it is possibly to transfer the basic techniques used in the previous sections also to the case of non-Hermitian matrices. Most of the theoretical results concerning, e.g., the sign of the error in the inner and outer quadrature rules, do not hold any longer in this case so that one cannot obtain guaranteed lower or upper bounds for the error in general. The basis for being able to also use a similar approach in the non-Hermitian case is given in [24, 51], where it is shown that Arnoldi's method can also be related to quadrature rules, similar to what was done for the Lanczos process in Section 6.1. In this context, one investigates bilinear forms

$$(h_1, h_2)_{A,\boldsymbol{v}} = \boldsymbol{v}^H h_1(A)^H h_2(A)\boldsymbol{v} \tag{6.16}$$

induced by A and $\boldsymbol{v}$ for functions h_1, h_2 defined on $\operatorname{spec}(A)$. When the functions h_1 and h_2 are both analytic in a neighborhood of $\operatorname{spec}(A)$, one can use the Cauchy integral formula (as in Definition 2.4) to rewrite (6.16) as a double integral along a path Γ that winds around $\operatorname{spec}(A)$ exactly once

$$(h_1, h_2)_{A,\boldsymbol{v}} = \frac{1}{4\pi^2} \int_\Gamma \int_\Gamma \overline{h_1(z_1)} h_2(z_2) \boldsymbol{v}^H (\overline{z_1} I - A^H)^{-1} (z_2 I - A)^{-1} \boldsymbol{v}\, \overline{\mathrm{d}z_1}\, \mathrm{d}z_2. \tag{6.17}$$

Using the quantities from the Arnoldi decomposition (2.23) to approximate (6.16), i.e.,

$$\boldsymbol{v}^H h_1(A)^H h_2(A)\boldsymbol{v} \approx \|\boldsymbol{v}\|_2^2 \hat{\boldsymbol{e}}_1^H h_1(H_k)^H h_2(H_k)\hat{\boldsymbol{e}}_1 \tag{6.18}$$

can then be interpreted as a k-point quadrature rule for (6.17). One can also show that the polynomials p_i defining the Arnoldi basis vectors are orthonormal with respect to the bilinear form (6.16), i.e.,

$$(p_i, p_j)_{A,\boldsymbol{v}} = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{otherwise.} \end{cases}$$

From this, one can prove that the resulting k-point quadrature rules are exact for $(h_1, h_2) \in \mathbb{W}_{k-1}$, where

$$\mathbb{W}_{k-1} = (\Pi_{k-1} \oplus \Pi_k) \cup (\Pi_k \oplus \Pi_{k-1});$$

see [24]. We do not go into detail concerning the theoretical analysis of the resulting quadrature rules, as most of this theory is not important or not applicable for the developments presented in this section (one can, e.g., show that under some conditions, these *Arnoldi quadrature rules* give upper or lower bounds for the bilinear form (6.16), see [24], but these conditions are not fulfilled or cannot easily be verified in our setting).

In this manner, one can use the upper Hessenberg matrix H_k resulting from k steps of Arnoldi's method applied to A and $\boldsymbol{v}_{m+1}$ to compute error estimates for the mth Arnoldi approximation to $f(A)\boldsymbol{b}$, where f is a Stieltjes function, just as in the Hermitian case, by setting $h_1 = h_2 = \widetilde{e}_m$ in (6.18). However, the key for being able to do so with affordable additional computational cost in the Hermitian case was given by Theorem 6.7, which allows to perform the secondary Lanczos process on a $(2k+1) \times (2k+1)$ matrix instead of an $n \times n$ matrix. Unfortunately, the result given in [61] holds only in the Hermitian case and has to be modified accordingly in case of non-Hermitian A. Its proof, however, is almost the same as for the original result from [61].

Theorem 6.9. *Let $A \in \mathbb{C}^{n\times n}$ be Hermitian positive definite, let $\boldsymbol{v}_1 \in \mathbb{C}^n$ and let H_{m+k+1} be the upper Hessenberg matrix resulting from $m+k+1$ steps of Arnoldi's method for A and $\boldsymbol{v}_1$. Further, let $\hat{H}$ denote the upper Hessenberg matrix resulting from k steps of Arnoldi's method applied to H_{m+k+1} and $\hat{\boldsymbol{e}}_{m+1}$. Then $\hat{H} = H_k^{(2)}$, where $H_k^{(2)}$ denotes the matrix resulting from k iterations of Arnoldi's method for A and $\boldsymbol{v}_{m+1}$.*

Proof. Let the Arnoldi decomposition arising from k steps of Arnoldi's method for A and $\boldsymbol{v}_{m+1}$ be given as

$$A\widetilde{V}_k = \widetilde{V}_k H_k^{(2)} + h_{k+1,k}^{(2)} \widetilde{\boldsymbol{v}}_{k+1} \hat{\boldsymbol{e}}_k^H. \tag{6.19}$$

As $\boldsymbol{v}_{m+1} \in \mathcal{K}_{m+1}(A, \boldsymbol{v}_1)$, we obviously have that

$$\mathcal{K}_{k+1}(A, \boldsymbol{v}_{m+1}) \subseteq \mathcal{K}_{m+k+1}(A, \boldsymbol{v}_1).$$

Therefore, the basis vectors $\widetilde{\boldsymbol{v}}_1, \ldots, \widetilde{\boldsymbol{v}}_{k+1}$ generated by Arnoldi's method for A and $\boldsymbol{v}_{m+1}$ all lie in $\mathcal{K}_{m+k+1}(A, \boldsymbol{v}_1)$ and can thus be written as linear combinations of the basis vectors $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{m+k+1}$, i.e.,

$$[\widetilde{V}_k, \widetilde{\boldsymbol{v}}_{k+1}] = V_{m+k+1}[Q_k, \boldsymbol{q}_{k+1}] \tag{6.20}$$

for some matrix $[Q_k, \boldsymbol{q}_{k+1}] \in \mathbb{C}^{(m+k+1)\times(k+1)}$. As $[\widetilde{V}_k, \widetilde{\boldsymbol{v}}_{k+1}]$ and V_{m+k+1} both have orthonormal columns, $[Q_k, \boldsymbol{q}_{k+1}]$ must have orthonormal columns as well. Inserting (6.20) into the Arnoldi decomposition (6.19) gives

$$AV_{m+k+1}Q_k = V_{m+k+1}Q_k H_k^{(2)} + h_{k+1,k}^{(2)} V_{m+k+1}\boldsymbol{q}_{k+1}\hat{\boldsymbol{e}}_k^H. \tag{6.21}$$

Left-multiplying both sides of (6.21) by the orthogonal projector $V_{m+k+1}V_{m+k+1}^H$ onto the space $\mathcal{K}_{m+k+1}(A, \boldsymbol{v}_1)$ gives

$$V_{m+k+1}V_{m+k+1}^H AV_{m+k+1}Q_k = V_{m+k+1}Q_k H_k^{(2)} + h_{k+1,k}^{(2)} V_{m+k+1}\boldsymbol{q}_{k+1}\hat{\boldsymbol{e}}_k^H.$$

which by (2.24) simplifies to

$$V_{m+k+1}H_{m+k+1}Q_k = V_{m+k+1}Q_k H_k^{(2)} + h_{k+1,k}^{(2)} V_{m+k+1}\boldsymbol{q}_{k+1}\hat{\boldsymbol{e}}_k^H.$$

Noting that V_{m+k+1} has full (column) rank, this implies

$$H_{m+k+1}Q_k = Q_k H_k^{(2)} + h_{k+1,k}^{(2)} \boldsymbol{q}_{k+1}\hat{\boldsymbol{e}}_k^H. \tag{6.22}$$

Due to Lemma 2.23 and the fact that all subdiagonal entries of $H_k^{(2)}$ are positive (as it was computed by Arnoldi's method), it follows that (6.22) is the Arnoldi decomposition corresponding to H_{m+k+1} and $\boldsymbol{q}_1$. As $\widetilde{\boldsymbol{v}}_1 = \boldsymbol{v}_{m+1}$, we have that $\boldsymbol{q}_1 = \hat{\boldsymbol{e}}_{m+1}$, which proves the result. □

Theorem 6.7 can directly be derived from Theorem 6.9 by noting that in the Hermitian case, large parts of the coefficients in Q_k from (6.20) are zero due to the tridiagonal structure of H_{m+k+1}, thus allowing to only use a small part of the tridiagonal matrix for the computations, see [61]. As this is not the case in presence of a non-Hermitian matrix A, it is not possible to only use a small subblock of H_{m+k+1} for retrieving the matrix $H_k^{(2)}$. This in turn means that, while we can circumvent additional multiplications with A, we cannot avoid multiplications with H_{m+k+1}, a matrix growing from one iteration to the next. Apart from this fact, the approach of Algorithm 6.1 can be used in the same way as in the Hermitian case (replacing the Lanczos process by Arnoldi's method and the computation of the bounds in line 21 and 22 by a quadrature-based approximation

for (6.18)) and replacing the recurrence relations for the entries of the inverse of shifted versions of H_m with the explicit computation of these values, as the simple update formulas do not hold any longer, because H_m is not tridiagonal. We just briefly mention here that it is still possible to obtain the necessary quantities in a more efficient way by applying successive rotations, similar to what is done in GMRES for cheaply computing the residual norm, see, e.g., [116]. This does not, however, change the overall cost of the algorithm, at least in $\mathcal{O}$-sense, cf. also the proof of Lemma 6.10, so that we do not go into detail concerning this here. The resulting method is given in Algorithm 6.2.

Algorithm 6.2: Arnoldi's method for $f(A)\boldsymbol{b}$ with error estimate

Given: A, $\boldsymbol{b}$, f, k, ℓ, `tol`
Choose quadrature nodes/weights $(t_i, \omega_i)_{i=1,\dots,\ell}$ for inner quadrature.
$\boldsymbol{v}_1 \leftarrow \frac{1}{\|\boldsymbol{b}\|_2}\boldsymbol{b}$
for $m = 1, 2, \dots$ **do**
  $\boldsymbol{w}_m \leftarrow A\boldsymbol{v}_m$
  for $i = 1, \dots, m$ **do**
    $h_{i,m} \leftarrow \boldsymbol{v}_i^H \boldsymbol{w}_m$
    $\boldsymbol{w}_m \leftarrow \boldsymbol{w}_m - h_{i,m}\boldsymbol{v}_i$
  $h_{m+1,m} \leftarrow \|\boldsymbol{w}_m\|_2$
  if $h_{m+1,m} = 0$ **then**
    $\boldsymbol{f}_m \leftarrow \|\boldsymbol{b}\|_2 V_m f(H_m)\hat{\boldsymbol{e}}_1$.
    Stop.
  $\boldsymbol{v}_{m+1} \leftarrow \frac{1}{h_{m+1,m}}\boldsymbol{w}_m$
  for $i = 1, \dots, \ell$ **do**
    $\rho_i(m) \leftarrow h_{m+1,m}\hat{\boldsymbol{e}}_m^H (H_m + tI)^{-1}\hat{\boldsymbol{e}}_1$
  if $m \geq k + 1$ **then**
    Perform k steps of Algorithm 2.1 for H and $\hat{\boldsymbol{e}}_{m-k}$, yielding $\hat{H}$.
    `estimate` $\leftarrow \|\boldsymbol{b}\|_2^2 \hat{\boldsymbol{e}}_1^H \sum_{i=1}^{\ell} |\omega_i \rho_i(m-k-1)|^2 (\hat{H} + t_i I)^{-H} (\hat{H} + t_i I)^{-1} \hat{\boldsymbol{e}}_1$
    if `estimate` $\leq$ `tol` **then**
      $\boldsymbol{f}_m \leftarrow \|\boldsymbol{b}\|_2 V_m f(H_m)\hat{\boldsymbol{e}}_1$.
      Stop.

Note that, in contrast to Algorithm 6.1, Algorithm 6.2 does not guarantee that the exact error norm of the iterate $\boldsymbol{f}_m$ lies below the prescribed tolerance `tol` upon termination. We note that the results of the numerical experiments in Section 6.6 suggest that the error estimates are rather accurate and reliable in many situations, at least for Stieltjes functions. It can, however, be useful to include a "safety factor" $\varepsilon < 1$ in the computations and run Algorithm 6.2 with the tolerance $\varepsilon \cdot$`tol` instead of `tol` if it is important that the prescribed tolerance is not

only approximately reached. The additional cost of Algorithm 6.2 in comparison to the standard Arnoldi method without computation of error estimates is given in the next lemma.

Lemma 6.10. *Performing Algorithm 6.2 instead of Algorithm 2.1 (plus the computation of $\boldsymbol{f}_m$) for $A \in \mathbb{C}^{n\times n}$ and $\boldsymbol{b} \in \mathbb{C}^n$ requires an additional computational cost of the order $\mathcal{O}(m^2(k+\ell)+k\ell)$ in the mth iteration and thus $\mathcal{O}(m_{\max}^3(k+\ell)+m_{\max}k\ell)$, if $m_{\max}$ iterations are performed in total. In particular, the additional cost per iteration is independent of n, but not of m.*

Proof. The proof is very similar to the one of Lemma 6.8, with the following differences. The secondary Arnoldi method for step m now has a cost of $\mathcal{O}(m^2k)$, as each multiplication with H_{m+k+1} has cost $\mathcal{O}(m^2)$ since the upper triangle of this matrix is in general dense (and we assume $k \in \mathcal{O}(m)$). The solution of each linear system in line 14 has cost $\mathcal{O}(m^2)$ due to the Hessenberg structure of H_m, which results in $\mathcal{O}(m^2\ell)$ for all systems. □

According to Lemma 6.10, the cost of computing error estimates in Algorithm 6.2 grows with the number of iterations performed. Therefore, if a large number m of iterations is necessary, the cost of computing the estimates may become prohibitively large. On the other hand, the cost of the orthogonalization in Arnoldi's method also grows from one iteration to the next and is in fact of order $\mathcal{O}(mn)$. For k fixed (independently of m and n), we have that $\mathcal{O}(m^2k) \subset \mathcal{O}(mn)$, so that the cost of Algorithm 6.2 (ignoring the matrix vector multiplication) is still not dominated by the cost of computing the error estimates, at least in $\mathcal{O}$-sense. Nonetheless, computing the error estimates is more costly than in the Hermitian case so that computing error estimates in the non-Hermitian case seems particularly attractive in situations where only a low number of very costly iterations is necessary for reaching the desired accuracy, as it is, e.g., typically the case in extended Krylov methods. Extending the results of this chapter to these related iterative methods is the topic of Chapter 7.

In analogy to what was discussed at the end of Section 6.4, we again mention that the possibility of restart recovery is not given for all iterations in case one uses a restarted Arnoldi method. One can, however, again use the norm of the approximation computed at the end of each cycle as an estimate of the error norm. In contrast to the Hermitian positive definite case, there is no provable benefit of specifically using estimates computed by quadrature rules instead of just using $\|\boldsymbol{e}_m^{(k)}\|_2$, as for both alternatives, one does not have any information on the sign of the remainder or the quality of the estimate. We therefore just mention this here briefly, but will not further investigate it in the numerical experiments in Section 6.6.

Another obvious extension of the approaches presented in this chapter, as long as one is not interested in computing guaranteed bounds, is to apply them to Cauchy-type integral representations (3.2), so that it is also possible to compute error estimates for functions like the matrix exponential. We provide numerical results for this approach in the next section (along with the results for Stieltjes functions) without presenting the (obvious) modifications to the algorithms in detail here.

6.6 Numerical experiments

In this section, we compute error bounds and estimates for Arnoldi approximations for the model problems from Section 2.6, both for the full (based on restart recovery) and the restarted (based directly on Theorem 6.5) Arnoldi method. We begin by illustrating the results from Section 6.3 and 6.4, i.e., we consider approximating Stieltjes matrix functions of Hermitian positive definite matrices.

The first model problem we consider is sampling from a Gaussian Markov random field. As the matrix function under consideration is the inverse square root and A is positive definite, Theorem 6.5 guarantees that we can compute lower and upper bounds for the error norm in Algorithm 6.1. Note that the computation of upper bounds requires knowledge of (a good lower bound a for) the smallest eigenvalue of A. In this special application, the smallest eigenvalue of the precision matrix is known to be 1. In other cases, one can either precompute an approximation for the smallest eigenvalue, if this is feasible, or directly use the approximation obtained from the Lanczos iteration (i.e., the smallest Ritz value, possibly multiplied by a safety factor $\varepsilon < 1$, see [61]). In this case, as long as one does not know whether the eigenvalue is represented accurately enough, one has again no guarantee that the computed estimate is really an upper bound. We touch on this topic again when discussing the Hermitian lattice QCD model problem, in which we use this approach, as the computation of eigenvalues is very costly (cf. Section 4.4).

Figure 6.1 presents the bounds computed by Algorithm 6.1 for the Gaussian Markov random field model problem for 150 Lanczos iterations and different values of k. We show the exact error norm as well as the lower and upper bounds computed with $k = 2, 5$ and 10 quadrature nodes for the outer Gauss and Gauss–Radau rule. For the inner quadrature rule used for evaluating the error function we use $\ell = 20$ nodes of a Gauss and Gauss–Radau rule, respectively, chosen such that the sign of the error in the inner and outer quadrature rule is the same and we compute guaranteed bounds, cf. also Proposition 6.6. Keep in mind that the bounds are computed in retrospect, so that, e.g., when $k = 10$, the bounds for the 140th to 150th iteration are not available when 150 iterations are performed in total. For all values of k, we see that the qualitative behavior of the error is

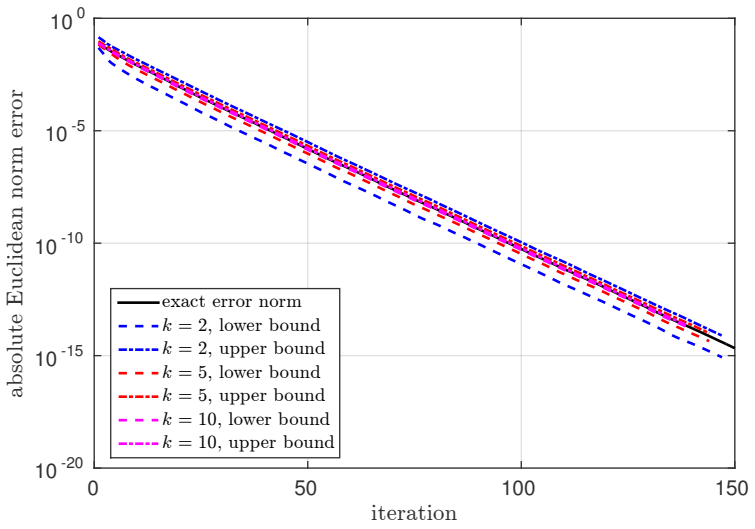


Figure 6.1: Exact error norm and bounds computed by Algorithm 6.1 for approximating $A^{-1/2}\boldsymbol{z}$ in the Gaussian Markov random field model problem. The inner quadrature rule uses $\ell = 20$ nodes, while the number of nodes in the outer quadrature rule is varied between $k = 2, 5$ and 10.

captured accurately (in particular for $k = 5$ and $k = 10$) and that even for the very small number of $k = 2$ quadrature nodes, the error is only overestimated (respectively underestimated) by about one order of magnitude, which is already an improvement over the asymptotic bounds from Chapter 5. For $k = 5$ and $k = 10$, the error bounds lie very close to the exact value of the integral, however they are only available three (respectively eight) iterations later than the bounds for $k = 2$. An interesting question (for which there will be no single answer for all possible cases) is the following. Assume we are given a prescribed accuracy `tol` for the absolute error norm to be reached by the computed approximation and we use the upper bounds from Section 6.3 as stopping criterion, as it is done in Algorithm 6.1. Then, for which value of k do we require the smallest number of iterations until convergence to the desired tolerance is detected (this is of course what one typically wants to have in practical computations, assuming that the cost for computing the error bounds is negligible for all considered values of k). For small values of k the bounds are inaccurate but available early, and for large values of k the bounds are accurate but available late. Therefore, it is not at all clear which value of k in this trade-off between accuracy and early availability is optimal. The answer depends on different factors, not only on the quality of the bounds as a function of k, but also, e.g, on the steepness of the convergence slope for the function, matrix and right-hand side at hand. We experimentally determine the "optimal" value of k for the Gaussian Markov random field model

ℓ	ratio lower bound	ratio upper bound
5	1.01	1.03
10	1.003	1.002
20	1.00004	1.00001

Table 6.1: Maximum ratio of the bounds computed by Algorithm 6.1 for the values $\ell = 5, 10, 20$ of inner quadratures nodes and the bounds computed for $\ell = 50$ for the Gaussian Markov random field model problem and $k = 5$.

problem, when trying to reach an accuracy of $\texttt{tol} = 10^{-9}$. For $k = 2$, the iteration would have been terminated after 92 iterations, just as with $k = 5$. For $k = 10$, one would require 96 iterations before detecting convergence to the desired tolerance, so that this value, despite the very high accuracy of the computed bounds, would result in four unnecessary Lanczos iterations in comparison to the lower values of k (which may seem surprising because the bounds for $k = 2$ look quite more inaccurate than those for the other values at first glance). Another interesting question is how to choose the value ℓ of inner quadrature nodes, especially considering the fact that this number has to be set to a pre-chosen value if one wants to avoid superfluous computations. Our experiments revealed that the computed error bounds are not very sensitive with respect to the value of ℓ, to the extent that the differences in the resulting bounds are not visible to the eye in the respective convergence graphs. We therefore give the results of our experiments, this time fixing $k = 5$ and varying $\ell = 5, 10, 20$ and 50 in form of a table which reports the maximal ratio between the bounds computed for $\ell = 5, 10, 20$ and the most accurate value computed for $\ell = 50$. These ratios are given in Table 6.1 and show that the resulting bounds are almost the same for all values, in particular when considering that even a difference of a factor of two in the resulting bounds would be acceptable in most cases. Therefore, and in addition keeping in mind that the cost of Algorithm 6.1 depends only mildly on ℓ (cf. Lemma 6.8), one may choose some not too low number, like, e.g., $\ell = 20$ or 50 and will most probably obtain satisfactory results. There exist cases, however, in which the error function is harder to approximate and the computed bounds are more sensitive with respect to the number ℓ of inner quadrature nodes as well. It might therefore be advisable to use a simple form of adaptive quadrature again as a safety measure. This is similar to what is done in our quadrature-based restarted Arnoldi method, Algorithm 4.3, albeit in a "relaxed form" (as one only wants to recompute the values of the nodal polynomial at the quadrature points if it is really necessary, to avoid computational cost depending on the iteration number m). A straightforward approach is to use two quadrature rules of different order again, but to reject the result only if the two computed approximations differ by a factor of more than two, e.g. This way, one can still detect whether the inner approxima-

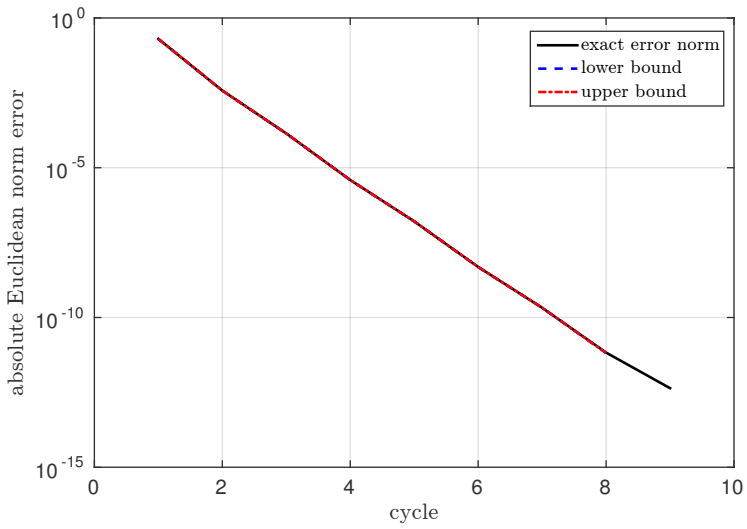


Figure 6.2: Exact error norm and bounds computed in Algorithm 4.3 for approximating $A^{-1/2}\boldsymbol{z}$ in the Gaussian Markov random field model problem. The inner quadrature rule uses $\ell = 20$ nodes, the number of outer quadrature nodes corresponds to the restart length $m = 20$.

tion is good enough but avoid to recompute the values of the nodal polynomial from scratch too frequently.

In Figure 6.2, we report the bounds resulting from using the Hessenberg matrix from cycle j of the restarted Arnoldi method to compute bounds for the error after cycle $j-1$, as described at the end of Section 6.4. We use the same parameters in the restarted Arnoldi method as in the experiments from Section 4.4, in particular restart length $m = 20$, so that the computed bounds correspond to 20-point Gauss and 21-point Gauss–Radau rules, respectively. Both bounds are almost indistinguishable from the exact error norm, which can be expected due to the rather high number of quadrature nodes used. This example illustrates that it is very attractive to use the developed error bounds also in the restarted Arnoldi method, especially considering that the additional work which is necessary for computing them is even less than for the unrestarted Arnoldi method, as no secondary Lanczos process is necessary.

Next, we consider approximating the Neuberger overlap operator at zero chemical potential, i.e., computing the inverse square root of a Hermitian positive definite matrix again. We use the same parameters as in the previous experiment and report the exact error norm and the computed bounds in Figure 6.3, this time also for $k = 20$ outer quadrature nodes. For computing the upper bounds for the error, it is necessary to have a good approximation of the smallest eigenvalue of $\Gamma_5 D_W$ at

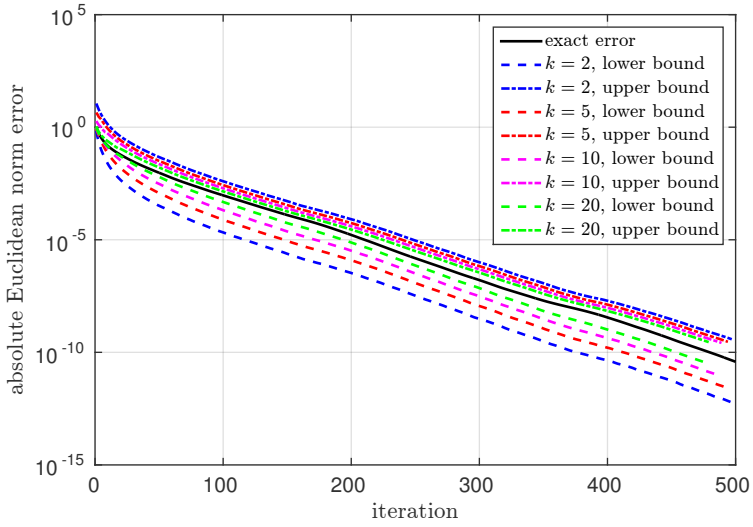


Figure 6.3: Exact error norm and bounds computed by Algorithm 6.1 for approximating $\left((\Gamma_5 D_W)^2\right)^{-1/2}\Gamma_5 D_W \boldsymbol{b}$ in the Hermitian QCD model problem. The highest number of nodes used in the inner quadrature rule is $\ell = 100$, while the number of nodes in the outer quadrature rule is varied between $k = 2, 5, 10$ and 20.

hand. In contrast to the previous example, we do not know the smallest eigenvalue explicitly. As it is very costly to approximate it before starting the Arnoldi iteration (cf. Section 4.4) we use the smallest Ritz value as an approximation to $\lambda_{\min}$. As soon as the smallest Ritz value does not change substantially any longer from one iteration to the next, we assume that it has converged to $\lambda_{\min}$ to sufficient accuracy and use it (multiplied by the safety factor $\varepsilon = 0.99$) as the fixed quadrature node at the left of the interval of integration for the Gauss–Radau rule. This approach was suggested in [61]. The qualitative results obtained in this experiment are similar to those for the Gaussian Markov random field model problem, i.e., the bounds capture the behavior of the error norm very well. This time, the bounds are not as close to the exact error norm as before, especially the lower bounds underestimate the error norm by a quite large margin for smaller values of k. For $k = 2$, the error norm is underestimated by about two orders of magnitude, a value which drastically improves as k increases. The upper bounds (which are typically more important for a stopping criterion) are closer to the actual error norm also for small values of k and do not improve by such a large margin if k is increased. We again determine the value of k for which the iteration is terminated earliest when an accuracy of 10^{-9} is desired. For $k = 2$, the stopping criterion is fulfilled in iteration 478, for $k = 5$ in iteration 472, for $k = 10$ in iteration 470 and for $k = 20$ in iteration 471, so that this time, $k = 10$ is the optimal value (out of those that were tested). Still, we again see that the

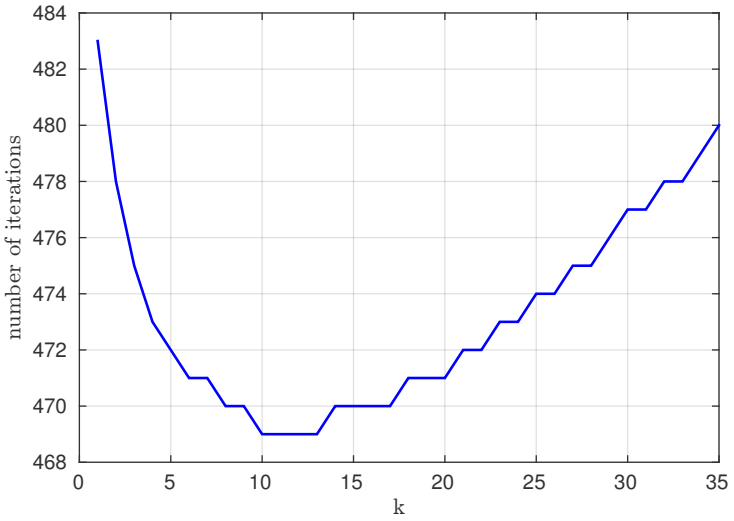


Figure 6.4: Iteration number at which the stopping criterion in Algorithm 6.1 is fulfilled (with $\texttt{tol} = 10^{-9}$) for the Hermitian QCD model problem and different values of k.

difference between the best and worst of the tested values is only eight iterations, such that even a "non-optimal" choice of k does not result in too much wasted computation time. The exception from this rule is that one should avoid a very large number of quadrature points, as from some value on, a higher number k of quadrature nodes will not make the bounds substantially more accurate. It will, however, increase the number of additional iterations which are necessary before convergence to the desired tolerance is detected. We illustrate this by comparing the iteration number at which convergence to the accuracy 10^{-9} is detected for all values of k between 1 and 35. The results are given in Figure 6.4. The optimal values in this case are found to be $k = 12$ or 13. For smaller or larger values, a higher number of iterations is necessary. For values of k larger than 20, the increase in the number of necessary iterations is almost proportional to the increase in k, thus confirming the intuitive conjecture that for too high values of k, no additional accuracy in the bounds is obtained and the additional quadrature nodes only delay the availability of the computed bound. Therefore, it seems like a reasonable, albeit heuristic, guideline for practical computations to choose not more than $k = 20$ nodes for the outer quadrature rule (for a more precise guideline, one would also, e.g., need to take into account the speed of convergence of the method for the problem at hand).

Another difference between this and the previous experiment which is worth mentioning is that larger numbers ℓ of inner quadrature nodes are necessary to obtain satisfactory bounds. The results reported in Figure 6.3 were produced using an adaptive approach as described in the discussion of the previous experiment. We

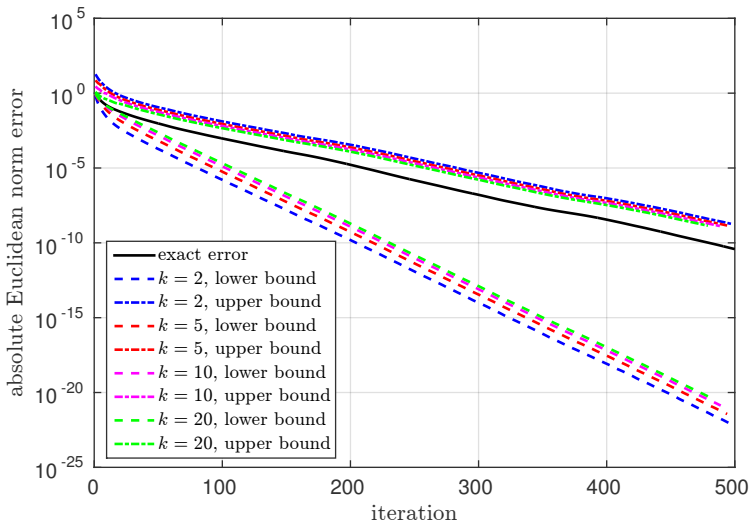


Figure 6.5: Exact error norm and bounds computed by Algorithm 6.1 for approximating $\left((\Gamma_5 D_W)^2\right)^{-1/2}\Gamma_5 D_W \boldsymbol{b}$ in the Hermitian QCD model problem. The number of nodes used in the inner quadrature rule is fixed to $\ell = 20$, while the number of nodes in the outer quadrature rule is varied between $k = 2, 5, 10$ and 20.

ℓ	ratio lower bound	ratio upper bound
10	5500	6.18
20	10.47	3.17
50	1.12	1.33

Table 6.2: Maximum ratio between the of the bounds computed by Algorithm 6.1 for the values $\ell = 10, 20, 50$ of inner quadratures nodes and the bounds computed for $\ell = 100$ for the Hermitian lattice QCD model problem and $k = 5$ in the first 100 iterations.

illustrate the influence of the value of ℓ by also giving the results for $\ell = 20$, a value which was largely sufficient for the previous model problem, in Figure 6.5. In addition, we again give a comparison of different values of ℓ in Table 6.2. The ratios between the quadrature rules for $\ell = 10$ and $\ell = 20$ compared to the most accurate tested rule for $\ell = 100$ are very large, showing that the approximations computed for these values are not accurate and one can expect the error in the inner quadrature rule to make a non-negligible contribution to the quality of the computed bounds. The ratio between $\ell = 50$ and $\ell = 100$ is still not as small as what was observed in the previous model problem, but it will in general be accurate enough so that the deviation of the bound from the exact error norm is not dominated by the error of the inner quadrature rule.

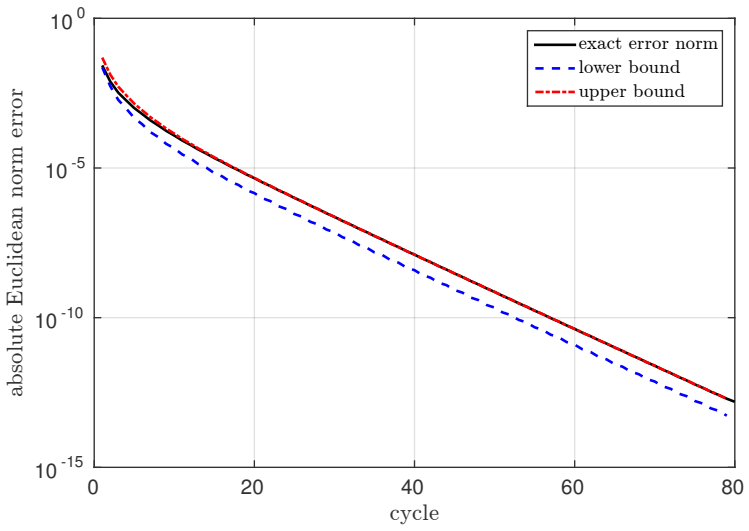


Figure 6.6: Exact error norm and bounds computed in Algorithm 4.3 for approximating $\left((\Gamma_5 D_W)^2\right)^{-1/2}\Gamma_5 D_W \boldsymbol{b}$ in the Hermitian lattice QCD model problem. The inner quadrature rule uses at most $\ell = 50$ nodes, the number of outer quadrature nodes corresponds to the restart length $m = 20$.

Before proceeding with model problems corresponding to functions other than Stieltjes functions or non-Hermitian matrices, we present error bounds computed in the *restarted* Arnoldi method for the Neuberger overlap operator at zero chemical potential in Figure 6.6. The upper bound again almost completely agrees with the exact error norm, the lower bound slightly underestimates it, but by less than one order of magnitude, again demonstrating that this approach gives very accurate estimates for the error norm in the restarted Arnoldi setting. We stress, however, that one has to be cautious when using the approach for approximating $\lambda_{\min}$ described before in the restarted Arnoldi case. For small values of m, it may well happen that no Ritz value approximates $\lambda_{\min}$ accurately enough. Thus, we advise to mainly use the described approach for computing error bounds in the restarted Arnoldi method if $\lambda_{\min}$ is explicitly known (like, e.g., for the Gaussian Markov random field model problem). We therefore re-used the approximation to $\lambda_{\min}$ obtained from the experiment involving the unrestarted Arnoldi method to obtain the results given in Figure 6.6.

Note that for the numerical experiments to come, we will not report results for the restarted Arnoldi method, as in the model problems considered in the following, one cannot compute guaranteed bounds for the error norm. The simple approach of just using the norm of the update computed in cycle j as an estimate for the error norm in cycle $j-1$ can be used in these cases, as the more complicated quadrature-based approach does not have any (provable) advantages over it.

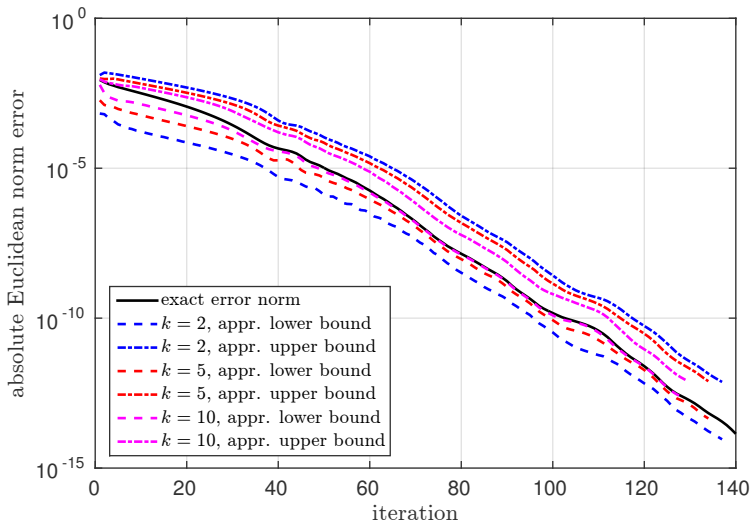


Figure 6.7: Exact error norm and (approximate) bounds computed by Algorithm 6.1 for approximating $(e^{-\theta\sqrt{A}} - I)A^{-1}\boldsymbol{b}$ for the semi-discretization of the wave equation. The inner quadrature rule uses $\ell = 20$ nodes, while the number of nodes in the outer quadrature rule is varied between $k = 2, 5$ and 10.

In the semi-discretization of the wave equation, the matrix A is Hermitian positive definite, but the function f is not a Stieltjes function, as it is generated by an oscillating function μ, so that we cannot guarantee the estimates to be error bounds by our theory. Nonetheless, we provide results for both Gauss and Gauss–Radau quadrature, which experimentally show that we still get bounds in this case. For the Gauss–Radau rule, we use the fact that the smallest eigenvalue of the three-dimensional Laplacian is explicitly known. Otherwise one could again use the smallest Ritz value after some iterations as an approximation. The quality of the estimates is again very similar to what was observed in the two previous experiments, with the (approximate) lower bounds being slightly more accurate than the (approximate) upper bounds in this example, especially in later iterations. The (approximate) upper error bound decreases below 10^{-9} in iteration 108 for $k = 2$ and $k = 5$ and in iteration 109 for $k = 10$, so that again all values lie closely together and are reasonable choices. We stress that one has to keep in mind that in this situation, the error estimate decreasing below 10^{-9} is not a guarantee that the exact error norm lies below the tolerance (although it is the case in the example presented here).

When approximating the exponential function of a Hermitian negative definite matrix for solving the heat equation, one is in another situation where one cannot compute guaranteed error bounds although the system matrix is Hermitian and definite, because the exponential is not a Stieltjes function. For approximating the

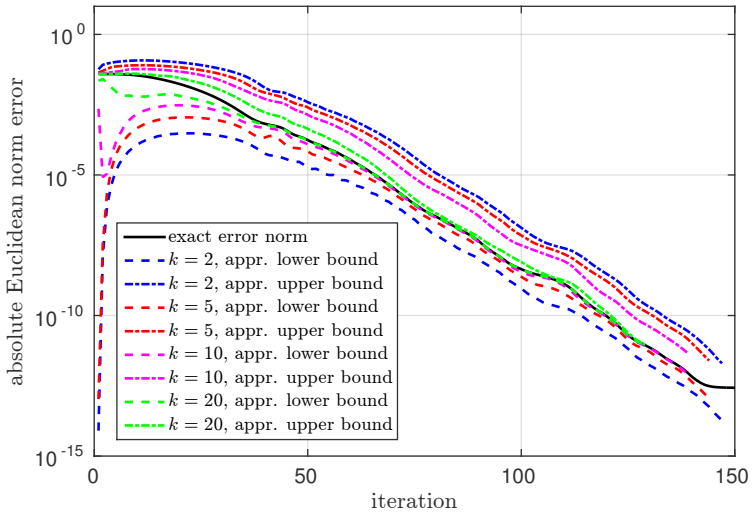


Figure 6.8: Exact error norm and (approximate) bounds computed by Algorithm 6.1 for approximating $e^{\theta A}\boldsymbol{b}$ for the semi-discretization of the heat equation. The inner quadrature rule uses $\ell = 20$ nodes on a parabolic Hankel contour, while the number of nodes in the outer quadrature rule is varied between $k = 2, 5, 10$ and 20.

error function in the inner quadrature rule, we use a fixed number of quadrature nodes on the parabolic Hankel contour (4.22). Although we found this approach to lead to instabilities in the context of our restarted Arnoldi implementation in Chapter 4, we could safely use it here. We only need a small number of quadrature nodes for finding a rather rough estimate of the value of the error function, so that the problems mentioned for higher numbers of quadrature nodes in Section 4.3 do not occur here, and in addition, the matrix A is Hermitian negative definite, so that we know that all Ritz values lie on the negative real axis, where the contour (4.22) gives very accurate approximations. The results of our experiment are reported in Figure 6.8. While the behavior of the approximate bounds for all values of k looks about the same as before after the first 25 iterations, there is a significant difference in the first few iterations. For the smaller number of quadrature nodes, the approximate lower bound severely underestimates the error norm (even estimating it to only lie slightly above the order of magnitude of the machine precision in the first iterations). For $k = 10$ this effect becomes less severe, and for $k = 20$, estimates of sufficient accuracy are computed from the first iteration on. Again we observe that also in this case, which is not covered by our theory, we obtain bounds for the error. The optimal values of k for identifying an error norm below 10^{-9} are again $k = 2$ and $k = 5$, for which 128 iterations are necessary. For $k = 10$ one needs 129 iterations and $k = 20$ results in 135 iterations before termination, again showing the characteristic increase for large

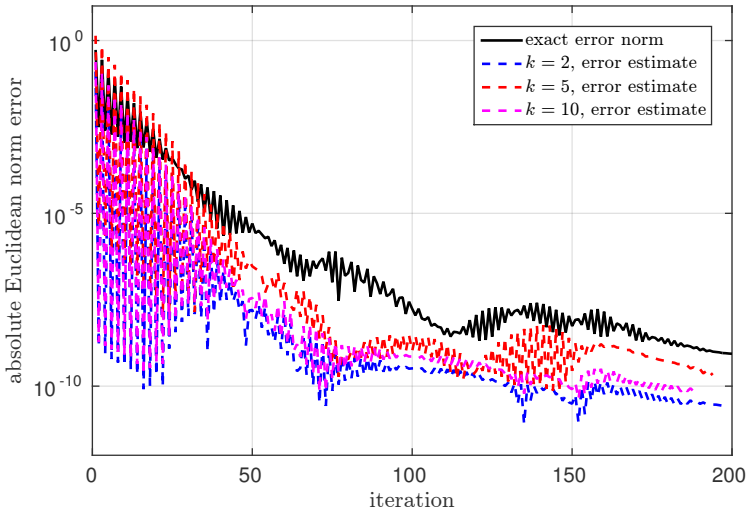


Figure 6.9: Exact error norm and estimates computed by Algorithm 6.2 for approximating $e^{\theta A}\boldsymbol{b}$ for the semi-discretization of a convection diffusion equation. The inner quadrature rule uses $\ell = 20$ nodes on an adaptively constructed parabolic Hankel contour, while the number of nodes in the outer quadrature rule is varied between $k = 2, 5$ and 10.

values of k observed in Figure 6.4 for the Hermitian lattice QCD model problem.

In the following, we turn our attention to the non-Hermitian model problems, namely the semi-discretization of a convection diffusion equation and the Neuberger overlap operator at nonzero chemical potential. For these problems, we use Algorithm 6.2, i.e., we only compute one error estimate, which will in general neither be an upper nor a lower bound for the norm of the error. We test the same parameters as for the Hermitian problems. For computing error estimates for the convection diffusion equation, we need to approximate the error function arising from the Cauchy integral representation of the exponential function. In contrast to the semi-discretized heat equation, the matrix A is not Hermitian in this case, so that the Ritz values can lie outside of the negative real axis. Therefore using the parabolic contour (4.22) can lead to useless approximations, especially when Ritz values lie on the outside of the contour, meaning that one not only has to expect inaccuracies, but the approximated integral is just plain wrong and no representation of the error function. We therefore use an approach similar to the one employed in our restarting algorithm for approximating the exponential of a non-Hermitian matrix. We again use an adaptively constructed Hankel contour (4.23), which we construct for the default parameters $a = 1, c = 0.25$ in the beginning. If a Ritz value outside the integration contour is detected, we adjust the contour accordingly (which requires recomputation of the value of the nodal polynomial

at all quadrature points, but as already reasoned earlier, the computational cost for doing this is negligible compared to the other parts of the algorithm in most cases). We note that this approach requires explicitly computing the Ritz values in each iteration which increases the additional cost of Algorithm 6.2. This may make this approach prohibitively expensive in cases where the required number m of iterations is large (which is, however, a situation in which one typically should avoid using the standard Arnoldi method and instead turn to restarted or rational Krylov methods). The results of our experiment are given in Figure 6.9. In this example, the convergence of the Arnoldi approximations to $f(A)\boldsymbol{b}$ is not monotone and the error estimates inherit the oscillating behavior of the exact error. In the first about 50 iterations, the oscillations in the error norm are heavily amplified in the error estimates, making them not reliable here. After the 50th iteration, the oscillations in the error estimates become less strong and lie in about the same order of magnitude as the oscillations in the exact error norm. However, the norm of the error is underestimated by about two to three orders of magnitude between the 70th and 100th iteration. Interestingly, the error estimate computed for $k = 5$ is more accurate for this model problem than the estimate for $k = 10$. While the estimates give an idea on how the error behaves, especially in later iterations, this experiment also illustrates the danger of using the estimates as sole stopping criteria in the non-Hermitian case (or more generally in any situation in which one has no guarantee that the computed estimates are upper bounds). When demanding an accuracy of, e.g., 10^{-9}, a stopping criterion based on the estimates produced by Algorithm 6.2 would already stop after a few iterations, or even if one would ignore the largely oscillating error estimates computed in the initial phase, after about 70 iterations, at a point where the exact error is far away from reaching the demanded accuracy. We do therefore not recommend to use the error estimates of Algorithm 6.2 as sole stopping criteria in practical situations. Even using a safety factor so that one only stops the iteration when the estimate lies several orders of magnitudes below the desired accuracy does not solve this problem. On the one hand this may result in a large number of unnecessary iterations, on the other hand, no matter how small the safety factor is chosen, there can never be a guarantee that it suffices to reach the desired accuracy, as we have no theoretical results at all on the quality of the computed error estimates. The next experiment, however, will show that there also exist situations in which the error estimates from Algorithm 6.2 are of much better quality and do not always show the behavior observed for the "highly non-Hermitian" convection diffusion problem (and the oscillating exponential function).

When approximating the action of the Neuberger operator at nonzero chemical potential, the error norm decreases much more smoothly than for the exponential function in the semi-discretization of the convection diffusion equation, as can be seen from Figure 6.10. The error estimates behave in the same way as the exact error norm, so that no oscillations like in the previous experiment are observed.

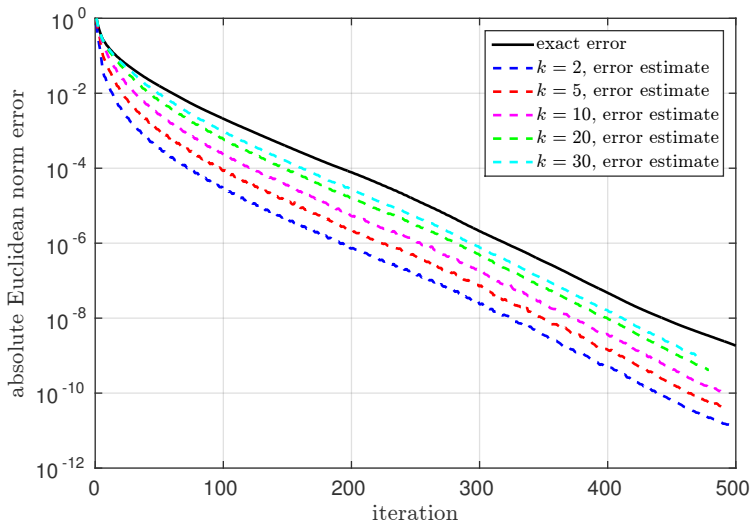


Figure 6.10: Exact error norm and estimates computed by Algorithm 6.2 for approximating $\left((\Gamma_5 D_W)^2\right)^{-1/2}\Gamma_5 D_W \boldsymbol{b}$ in the non-Hermitian QCD model problem. The highest number of nodes used in the inner quadrature rule is $\ell = 100$, while the number of nodes in the outer quadrature rule is varied between $k = 2, 5, 10, 20$ and 30.

Nonetheless, the error is again severely underestimated for smaller values of k. The estimate for $k = 2$ differs from the exact value by about two orders of magnitude, for $k = 10$, the error is underestimated by about one order of magnitude. After an initial phase of about 20 iterations, the relative difference between the estimate and the exact value stays almost constant, which is different to the behavior for the exponential. Although we do not have theoretical results on the behavior or quality of the error estimates in the non-Hermitian case, this experiment at least illustrates that there are situations in which one can expect the estimates to capture the convergence behavior of the method quite accurately.

CHAPTER 7

ERROR ESTIMATES IN EXTENDED KRYLOV METHODS

In this chapter, we show how to transfer some of the techniques developed so far to extended Krylov subspaces. These subspaces are constructed by not only applying positive, but also *negative powers* of A to the vector $\boldsymbol{b}$. This amounts to approximating f by a rational function (a *Laurent polynomial*, to be precise) instead of a polynomial. Of course, constructing a basis for these subspaces is much more costly than in the standard Krylov case and in return one hopes to obtain accurate approximations for much smaller subspace dimension (as often rational functions of rather small degree are much better suited for approximating a given function than high-degree polynomials). Many of the properties of polynomial Krylov subspaces carry over to extended Krylov subspaces and we begin this chapter by providing some basic material on these spaces and highlighting similarities and differences to the polynomial case in Section 7.1. Afterwards, in Section 7.2, we show how to transfer the integral representation for the error derived in Chapter 3 to approximations from extended Krylov spaces. This would in principle allow to use a restarting method similar to the one from Chapter 4 for extended Krylov subspace approximations. However, one typically uses extended Krylov spaces only in situations where a low-dimensional approximation space suffices for achieving the desired accuracy, so that memory constraints seldom become an issue and we do not go into detail on this topic and only mention in passing that it would of course be possible to implement such a method. Instead, we focus on computing error estimates. To do so, we present a generalization of Arnoldi/Lanczos restart recovery (Theorem 6.7 and 6.9) to extended Krylov spaces in section 7.3. We can then use these tools to compute estimates for the error in extended Krylov subspace methods. We illustrate the quality of these

estimates by performing numerical experiments on some of our model problems in Section 7.4.

7.1 Extended Krylov subspaces

In the following, we introduce extended Krylov subspaces, which, in the context of approximating matrix functions, were first considered in [38] and further investigated in, e.g., [94–96, 99].

The main idea of extended Krylov subspace methods (or also general rational Krylov subspace methods, which we do not cover here) is that oftentimes when using rational functions for approximating a given function, the degree of the numerator and denominator necessary to reach a certain accuracy is substantially smaller than the degree of a polynomial exhibiting the same approximation quality, see, e.g., [80]. Therefore it seems reasonable to use approximations to $f(A)\boldsymbol{b}$ which are characterized by an underlying rational approximant instead of a polynomial, especially when the behavior of f is difficult to capture by low-degree polynomials. Different variants of general rational Krylov subspace methods arise through the choice of the poles of the rational functions, see, e.g., [80–82].

One simple, black-box choice of poles, which results in the so-called extended Krylov subspaces, is to only choose the poles 0 and ∞ (often alternatingly), i.e., building a Krylov subspace with respect to powers of A and A^{-1}.

Definition 7.1. Let $A \in \mathbb{C}^{n\times n}$ be nonsingular and let $\boldsymbol{b} \in \mathbb{C}^n$. Then the (k, m)th *extended Krylov subspace* with respect to A and $\boldsymbol{b}$ is

$$\mathcal{E}_{k,m}(A, \boldsymbol{b}) := A^{-k}\mathcal{K}_{k+m}(A, \boldsymbol{b}) = \{\ell_{k,m-1}(A)\boldsymbol{b} : \ell_{k,m-1} \in \mathcal{L}_{k,m-1}\},$$

where

$$\mathcal{L}_{k,m-1} = \operatorname{span}\{t^{-k}, \dots, t^{-1}, 1, t, t^2, \dots, t^{m-1}\}$$

denotes the space of *Laurent polynomials* of denominator degree at most k and numerator degree at most $m - 1$.

We begin our exposition by collecting some evident and useful properties and different characterizations of extended Krylov subspaces. Results of this type have also been observed in, e.g., [38, 95, 124].

Proposition 7.2. *Let $A \in \mathbb{C}^{n\times n}$ be nonsingular and let $\boldsymbol{b} \in \mathbb{C}^n$. Then*

(i) $\mathcal{E}_{k,m}(A, \boldsymbol{b}) \subseteq \mathcal{E}_{k+k_0,m+m_0}(A, \boldsymbol{b})$ *for all* $k_0, m_0 \geq 0$,

(ii) $\mathcal{E}_{k,m}(A, \boldsymbol{b}) = \mathcal{K}_k(A^{-1}, A^{-1}\boldsymbol{b}) + \mathcal{K}_m(A, \boldsymbol{b}) = \mathcal{K}_{k+m}(A, A^{-k}\boldsymbol{b})$.

Proof. Property (i) directly follows from the definition of extended Krylov subspaces, similarly to the nestedness of polynomial Krylov subspaces. Both equalities in (ii) can be derived by using the representation

$$\mathcal{E}_{k,m}(A, \boldsymbol{b}) = \operatorname{span}\{A^{-k}\boldsymbol{b}, \ldots, A^{-1}\boldsymbol{b}, \boldsymbol{b}, A\boldsymbol{b}, A^2\boldsymbol{b}, \ldots, A^{m-1}\boldsymbol{b}\}$$

and the definition of polynomial Krylov subspaces. □

Property (i) from Proposition 7.2 is again a nestedness property, which holds for every increase of the order of the subspace, be it an increase of the numerator degree, the denominator degree, or both. Property (ii) relates extended Krylov subspaces to polynomial Krylov spaces in two different ways. The first characterization allows to write $\mathcal{E}_{k,m}(A, \boldsymbol{b})$ as the *sum* of two polynomial Krylov subspaces, one of them corresponding to A and one corresponding to A^{-1}, while the second one shows that $\mathcal{E}_{k,m}(A, \boldsymbol{b})$ is in fact a polynomial Krylov subspace corresponding to A, albeit with a different starting vector.

In the following we restrict ourselves to the case of $k = m$, sometimes called *diagonal* extended Krylov subspaces, to avoid unnecessary notational overhead. All results apply to general extended Krylov subspaces with $k \neq m$ with obvious modifications. A nested orthonormal basis for $\mathcal{E}_{m,m}(A, \boldsymbol{b})$ can be computed by a method similar to Arnoldi's method. There are, however, different ways to generate the basis vectors. One way is to generate them sequentially, one-by-one, by alternatingly applying A and A^{-1} to the respective last basis vector. Another approach, first introduced in [124], is to compute the basis in a "block-wise" fashion, two vectors at a time, by multiplying the last basis vector with A^{-1} and the second to last basis vector with A. This way, odd-numbered basis vectors advance the basis corresponding to powers of A and even-numbered basis vectors advance the basis corresponding to powers of A^{-1}. We will use the second, block-wise approach in this chapter, given in Algorithm 7.1.

The simple choice of poles in an extended Krylov subspace method gives rise to several theoretical and computational simplifications in contrast to general rational Krylov subspace methods. If it is feasible to solve the linear systems with A by a direct method, it is sufficient to compute an LU (or Cholesky) factorization [131] of A once and reuse it in each iteration of the method, while for varying poles one has to compute a factorization for each of the poles used. Another advantage is that for extended Krylov subspaces corresponding to a Hermitian matrix A it is again possible to derive an analogue to the short-recurrence Lanczos process, the only difference to the polynomial Lanczos algorithm being that the three-term

Algorithm 7.1: Block-wise extended Arnoldi method

Input: $m \in \mathbb{N}$, $A \in \mathbb{C}^{n\times n}$ nonsingular, $\boldsymbol{b} \in \mathbb{C}^n$

Output: Orthonormal basis $V_{m,m} = [\boldsymbol{v}_1, \dots, \boldsymbol{v}_{2m}]$ of $\mathcal{E}_{m,m}(A, \boldsymbol{b})$

$\boldsymbol{v}_1 \leftarrow \frac{1}{\|\boldsymbol{b}\|_2}\boldsymbol{b}$
$\boldsymbol{w}_2 \leftarrow A^{-1}\boldsymbol{b}$
$\boldsymbol{w}_2 \leftarrow \boldsymbol{w}_2 - (\boldsymbol{v}_1^H \boldsymbol{w}_2)\boldsymbol{v}_1$
$\boldsymbol{v}_2 \leftarrow \frac{1}{\|\boldsymbol{w}_2\|_2}\boldsymbol{w}_2$
for $j = 1, 2, \dots, m$ **do**
  $\boldsymbol{w}_{2j+1} \leftarrow A\boldsymbol{v}_{2j-1}$
  for $i = 1, \dots, 2j$ **do**
    $h_{i,2j-1} \leftarrow \boldsymbol{v}_i^H \boldsymbol{w}_{2j+1}$
    $\boldsymbol{w}_{2j+1} \leftarrow \boldsymbol{w}_{2j+1} - h_{i,2j-1}\boldsymbol{v}_i$
  $h_{2j+1,2j-1} \leftarrow \|\boldsymbol{w}_{2j+1}\|_2$
  $\boldsymbol{v}_{2j+1} \leftarrow \frac{1}{h_{2j+1,2j-1}}\boldsymbol{w}_{2j+1}$
  $\boldsymbol{w}_{2j+2} \leftarrow A^{-1}\boldsymbol{v}_{2j}$
  for $i = 1, \dots, 2j+1$ **do**
    $h_{i,2j} \leftarrow \boldsymbol{v}_i^H \boldsymbol{w}_{2j+2}$
    $\boldsymbol{w}_{2j+2} \leftarrow \boldsymbol{w}_{2j+2} - h_{i,2j}\boldsymbol{v}_i$
  $h_{2j+2,2j} \leftarrow \|\boldsymbol{w}_{2j+2}\|_2$
  $\boldsymbol{v}_{2j+2} \leftarrow \frac{1}{h_{2j+2,2j}}\boldsymbol{w}_{2j+2}$

recurrence turns into a five-term recurrence. The matrix of orthogonalization coefficients thus becomes pentadiagonal instead of tridiagonal, see, e.g., [94–96,124]. We do not present this method in a separate algorithm, as it suffices to modify the two "**for** i" loops in Algorithm 7.1 to run from $\max\{2j-3, 1\}, \dots, 2j$ and $\max\{2j-2, 1\}, \dots, 2j+1$, respectively.

The *extended Arnoldi approximation* is defined completely analogously to the standard Arnoldi approximation for polynomial Krylov subspaces and can be related to interpolation by Laurent polynomials.

Lemma 7.3. *Let $A \in \mathbb{C}^{n\times n}$ be nonsingular, let $\boldsymbol{b} \in \mathbb{C}^n$, let $V_{m,m}$ be the matrix computed by Algorithm 7.1 whose columns form an orthonormal basis of $\mathcal{E}_{m,m}(A, \boldsymbol{b})$, let $A_{m,m} = V_{m,m}^H A V_{m,m}$, let f be a function defined on* $\operatorname{spec}(A_{m,m})$ *and let*

$$\boldsymbol{f}_{m,m} = V_{m,m} f(A_{m,m}) V_{m,m}^H \boldsymbol{b} = \|\boldsymbol{b}\|_2 V_{m,m} f(A_{m,m}) \hat{\boldsymbol{e}}_1. \tag{7.1}$$

Then

$$\boldsymbol{f}_{m,m} = \ell_{m,m-1}(A)\boldsymbol{b},$$

where $\ell_{m,m-1} \in \mathcal{L}_{m,m-1}$ interpolates f at the eigenvalues of $A_{m,m}$.

Proof. The result follows, e.g., from the more general result of [80, Theorem 4.8] which gives a rational interpolation characterization for general rational Arnoldi approximations, of which the extended Arnoldi approximation (7.1) is a special case (with minor modifications to account for the block-wise generation of the basis vectors, which is not considered in [80]). □

The projected matrix $A_{m,m}$ which is needed to evaluate the extended Arnoldi approximation (7.1), in contrast to the polynomial Krylov case, does not coincide with the matrix of orthogonalization coefficients $h_{i,j}$. However, when A is Hermitian, one can show that it is pentadiagonal as well, and one can derive recursion formulas for the entries of $A_{m,m}$ based on the orthogonalization coefficients from the extended Arnoldi method [94–96, 118, 124], so that it is not necessary to explicitly compute the matrix as $A_{m,m} = V_{m,m}^H A V_{m,m}$.

To conclude this section, we mention that the orthonormal basis $V_{m,m}$ and the compressed matrix $A_{m,m}$ fulfill the following *extended Arnoldi relation*

$$AV_{m,m} = V_{m,m}A_{m,m} + [\boldsymbol{v}_{2m+1}, \boldsymbol{v}_{2m+2}]\tau_{m,m}[\hat{\boldsymbol{e}}_{2m-1}, \hat{\boldsymbol{e}}_{2m}]^H, \tag{7.2}$$

where $\tau_{m,m} = [\boldsymbol{v}_{2m+1}, \boldsymbol{v}_{2m+2}]^H A[\boldsymbol{v}_{2m-1}, \boldsymbol{v}_{2m}] \in \mathbb{C}^{2\times 2}$, which is a natural analogue to the polynomial Arnoldi decomposition (2.23), see, e.g., [124].

7.2 Generalization of the integral representation of the error to extended Krylov methods

In this section, we show how it is possible to generalize the error representation from Chapter 3 to the case of extended Krylov subspaces. For the sake of brevity, we restrict ourselves to the case of Stieltjes functions here and just mention in passing that all results hold (with obvious modifications) for "Cauchy-type" integral representations (3.2). There are again, like for polynomial Krylov spaces, two ways of deriving an integral representation for the error of extended Krylov approximations of Stieltjes matrix functions. One approach is using the interpolation characterization from Lemma 7.3 to derive a representation similar to the one from Lemma 3.3 for the interpolating Laurent polynomials, the other one is using the relation to shifted linear systems, as done in Chapter 5. We will cover both approaches here for the following reasons. On the one hand, the integral representation of the interpolating polynomial will allow to prove that it is again possible to compute lower and upper bounds for the error in extended Krylov subspace methods. On the other hand, working with extended Arnoldi approximations for shifted linear systems, similarly to what was done at the beginning of Section 5.2, allows to gain additional insight into the behavior of these methods

for linear systems in general. At first sight, it may not seem reasonable at all to use extended Krylov methods for the solution of linear systems, as each iteration of such a method requires the application of A^{-1} to a vector, i.e., the solution of a linear system. In fact, an extended Krylov subspace method for $A\boldsymbol{x} = \boldsymbol{b}$ would yield the exact solution $\boldsymbol{x}^*$ after just one step. However, if one has to solve a large number of shifted systems simultaneously, using an extended (or general rational) Krylov subspace method may become attractive, see, e.g., [81, 118, 125]. Clearly, if the total number of iterations in a rational Krylov subspace method needed for all systems to converge is lower than the number of systems to be solved, one has already gained something in terms of linear system solves. This gain can be even larger if one uses an extended Krylov subspace method, as this allows to re-use a single LU-decomposition in all iterations, making the subsequent linear system solves even cheaper. Therefore, investigating properties of extended Krylov methods for (shifted) linear systems is of interest in its own right and also allows to identify similarities and differences between polynomial and extended methods which are of interest for the applicability of our theory.

Lemma 3.3, which gave an integral representation for the interpolating polynomial of "Cauchy-type" functions, can easily be transferred to Laurent polynomials and Stieltjes functions. In fact, this is again a slight modification of a classical result for analytic functions given in Cauchy integral representation, see, e.g., [140, Theorem VIII.2]

Lemma 7.4. *Let f be a Stieltjes function of the form* (3.15)*. The interpolating Laurent polynomial $\ell_{m,m-1} \in \mathcal{L}_{m,m-1}$ of f with interpolation nodes $\{\theta_1, \dots, \theta_{2m}\} \subset \mathbb{C} \setminus \mathbb{R}_0^-$ is given as*

$$\ell_{m,m-1}(z) = \int_0^\infty \left(1 - \frac{w_{2m}(-z)t^m}{w_{2m}(t)(-z)^m}\right) \frac{1}{z+t} \, \mathrm{d}\mu(t), \tag{7.3}$$

where $w_{2m}(z) = \prod_{i=1}^{2m}(z+\theta_i)$, provided that the integral in (7.3) *exists.*

Proof. The function $1 - \frac{w_{2m}(-z)t^m}{w_{2m}(t)(-z)^m}$ is a rational function in z with numerator degree $2m$ and denominator degree m. Moreover, it has a root at $-t$, which means that its numerator must contain a linear factor $z+t$, showing that the integrand in (7.3) is a rational function in z with numerator degree $2m-1$ and denominator degree m. As the denominator is given by a multiple of $(-z)^m$, it directly follows that the integrand is a Laurent polynomial from $\mathcal{L}_{m,m-1}$. Integration with respect to t does not change this, so that $\ell_{m,m-1}(z)$ from (7.3) is indeed a Laurent polynomial of the required degrees. By definition of $w_{2m}(z)$ we have

$$\ell_{m,m-1}(\theta_i) = \int_0^\infty \left(1 - \frac{w_{2m}(-\theta_i)t^m}{w_{2m}(t)(-\theta_i)^m}\right) \frac{1}{t+\theta_i} \, \mathrm{d}\mu(t) = \int_0^\infty \frac{1}{t+\theta_i} \, \mathrm{d}\mu(t) = f(\theta_i)$$

for $i = 1, \dots, 2m$, showing that the interpolation conditions for f are satisfied. The case of coinciding interpolation nodes θ_i can be treated in the same way as in the proof of Lemma 3.3. □

Lemma 7.4 allows to easily derive a variant of Theorem 3.5 for extended Krylov subspaces. We omit the proof of the following result, as it is completely analogous to the one of Theorem 3.5.

Theorem 7.5. *Let $A \in \mathbb{C}^{n\times n}$ be nonsingular, let $\boldsymbol{b} \in \mathbb{C}^n$ and let f be a Stieltjes function of the form* (3.15)*. Assume that* $\operatorname{spec}(A) \subset \mathbb{C} \setminus \mathbb{R}_0^-$ *and denote by $\boldsymbol{f}_{m,m}$ the (m,m)th extended Arnoldi approximation* (7.1) *to $f(A)\boldsymbol{b}$. Assume that* $\operatorname{spec}(A_{m,m}) = \{\theta_1, \dots, \theta_{2m}\}$ *satisfies* $\operatorname{spec}(A_{m,m}) \subset \mathbb{C} \setminus \mathbb{R}_0^-$ *and define*

$$e_{m,m}(z) = \int_0^\infty \frac{t^m}{w_{2m}(t)} \cdot \frac{(-z)^{-m} w_{2m}(-z)}{z+t} \, \mathrm{d}\mu(t), \quad z \in \mathbb{C} \setminus \mathbb{R}_0^-, \tag{7.4}$$

where $w_{2m}(z) = \prod_{i=1}^{2m}(z+\theta_i)$. Then

$$f(A)\boldsymbol{b} - \boldsymbol{f}_{m,m} = e_{m,m}(A)\boldsymbol{b}. \tag{7.5}$$

The result of Theorem 7.5 is stated in a slightly different way than the one of Theorem 3.5 for polynomial Krylov spaces. While we could easily conclude that $w_m(-A)\boldsymbol{b} = (-1)^m \|\boldsymbol{b}\|_2 \gamma_m \boldsymbol{v}_{m+1}$ in the polynomial case, such a relation is not readily available for extended Krylov spaces, so that the term $(-z)^{-m} w_{2m}(-z)$ is still present in the error function representation given in (7.4). When investigating extended Krylov subspace methods for shifted linear systems in the following, we will show that a similar characterization of $(-A)^{-m} w_{2m}(-A)\boldsymbol{b}$ as for polynomial Krylov spaces is possible, allowing us to give a representation in which the error function is applied to the extended Arnoldi basis vector $\boldsymbol{v}_{2m+1}$ instead of $\boldsymbol{b}$.

Consider the shifted linear system

$$(A + tI)\boldsymbol{x}(t) = \boldsymbol{b} \tag{7.6}$$

and denote by

$$\boldsymbol{x}_{m,m}(t) = \|\boldsymbol{b}\|_2 V_{m,m}(A_{m,m} + tI)^{-1} \hat{\boldsymbol{e}}_1 \tag{7.7}$$

the extended Arnoldi approximation (7.1) for (7.6), computed from the extended Krylov space $\mathcal{E}_{m,m}(A, \boldsymbol{b})$. Note that extended Krylov subspaces are, in contrast to polynomial Krylov spaces, *not shift invariant*. It is still justified to compute approximations for different shifts t from the subspace built with A, as at least the "polynomial part" is shift invariant and it still holds that

$$V_{m,m}^H (A + tI) V_{m,m,} = A_{m,m} + tI,$$

as a straightforward calculation shows. The shifted extended Arnoldi approximations $\boldsymbol{x}_{m,m}(t)$ from (7.7) are exactly the vectors implicitly generated for all t when approximating the action of a Stieltjes matrix function on a vector, as

$$\boldsymbol{f}_{m,m} = \int_0^\infty \|\boldsymbol{b}\|_2 V_{m,m}(A_{m,m} + tI)^{-1}\hat{\boldsymbol{e}}_1 \,\mathrm{d}\mu(t) = \int_0^\infty \boldsymbol{x}_{m,m}(t)\,\mathrm{d}\mu(t).$$

Representing $f(A)\boldsymbol{b}$ as the integral over the solutions $\boldsymbol{x}^*(t)$ of (7.6) again, we thus find the representation

$$f(A)\boldsymbol{b} - \boldsymbol{f}_{m,m} = \int_0^\infty \boldsymbol{x}^*(t) - \boldsymbol{x}_{m,m}(t)\,\mathrm{d}\mu(t) = \int_0^\infty \boldsymbol{e}_{m,m}(t)\,\mathrm{d}\mu(t) \tag{7.8}$$

for the error of the extended Arnoldi approximation $\boldsymbol{f}_{m,m}$, where

$$\boldsymbol{e}_{m,m}(t) = \boldsymbol{x}^*(t) - \boldsymbol{x}_{m,m}(t)$$

denotes the error of the approximation (7.7). Using the fact that the errors $\boldsymbol{e}_{m,m}(t)$ fulfill (shifted versions) of the residual equation (2.28), we can rewrite (7.8) as

$$f(A)\boldsymbol{b} - \boldsymbol{f}_{m,m} = \int_0^\infty (A + tI)^{-1}\boldsymbol{r}_{m,m}(t)\,\mathrm{d}\mu(t) \tag{7.9}$$

where

$$\boldsymbol{r}_{m,m}(t) = \boldsymbol{b} - (A + tI)\boldsymbol{x}_{m,m}(t).$$

While it was obvious from Proposition 2.38(ii) in the polynomial Krylov case, it is not straightforwardly clear whether all residuals $\boldsymbol{r}_{m,m}(t)$ of the shifted extended Arnoldi iterates are collinear, so that (7.9) can be interpreted as the action of a matrix function on a single vector. To prove that this is indeed the case (which will later allow us to relate the error representation (7.9) to (7.5)), we revisit the extended Arnoldi decomposition (7.2).

Proposition 7.6. *Let $\boldsymbol{x}_{m,m}(t)$ be the extended Arnoldi approximation (7.7) for the shifted linear system (7.6) and let $\boldsymbol{r}_{m,m}(t) = \boldsymbol{b} - (A + tI)\boldsymbol{x}_{m,m}(t)$ be the corresponding residual. Then*

$$\boldsymbol{r}_{m,m}(t) = -\|\boldsymbol{b}\|_2[\boldsymbol{v}_{2m+1}, \boldsymbol{v}_{2m+2}]\tau_{m,m}[\hat{\boldsymbol{e}}_{2m-1}, \hat{\boldsymbol{e}}_{2m}]^H(A_{m,m} + tI)^{-1}\hat{\boldsymbol{e}}_1 \tag{7.10}$$

where $\tau_{m,m} = [\boldsymbol{v}_{2m+1}, \boldsymbol{v}_{2m+2}]^H A[\boldsymbol{v}_{2m-1}, \boldsymbol{v}_{2m}]$.

Proof. By adding the term $tV_{m,m}$ on both sides of (7.2), we directly get

$$(A + tI)V_{m,m} = V_{m,m}(A_{m,m} + tI) + [\boldsymbol{v}_{2m+1}\boldsymbol{v}_{2m+2}]\tau_{m,m}[\hat{\boldsymbol{e}}_{2m-1}, \hat{\boldsymbol{e}}_{2m}]^H. \tag{7.11}$$

Inserting (7.11) into the definition (7.7) of $\boldsymbol{x}_{m,m}(t)$, we get

$$\begin{aligned}\boldsymbol{r}_{m,m}(t) &= \boldsymbol{b} - (A+tI)(\|\boldsymbol{b}\|_2 V_{m,m}(A_{m,m}+tI)^{-1}\hat{\boldsymbol{e}}_1) \\ &= \boldsymbol{b} - \|\boldsymbol{b}\|_2 V_{m,m}(A_{m,m}+tI)(A_{m,m}+tI)^{-1}\hat{\boldsymbol{e}}_1 \\ &\quad - \|\boldsymbol{b}\|_2[\boldsymbol{v}_{2m+1},\boldsymbol{v}_{2m+2}]\tau_{m,m}[\hat{\boldsymbol{e}}_{2m-1},\hat{\boldsymbol{e}}_{2m}]^H(A_{m,m}+tI)^{-1}\hat{\boldsymbol{e}}_1 \\ &= -\|\boldsymbol{b}\|_2[\boldsymbol{v}_{2m+1},\boldsymbol{v}_{2m+2}]\tau_{m,m}[\hat{\boldsymbol{e}}_{2m-1},\hat{\boldsymbol{e}}_{2m}]^H(A_{m,m}+tI)^{-1}\hat{\boldsymbol{e}}_1,\end{aligned}$$

where the last equality follows because $\|\boldsymbol{b}\|_2 V_{m,m}\hat{\boldsymbol{e}}_1 = \boldsymbol{b}$. □

The representation (7.10) of the shifted residuals $\boldsymbol{r}_{m,m}(t)$ shows that they are linear combinations of the two basis vectors $\boldsymbol{v}_{2m+1}$ and $\boldsymbol{v}_{2m+2}$. Carefully inspecting the coefficient matrix $\tau_{m,m}$ however shows that only the vector $\boldsymbol{v}_{2m+1}$ contributes to the linear combination nontrivially.

Proposition 7.7. *Let* $\tau_{m,m} = [\boldsymbol{v}_{2m+1},\boldsymbol{v}_{2m+2}]^H A[\boldsymbol{v}_{2m-1},\boldsymbol{v}_{2m}]$. *Then*

$$\tau_{m,m}(2,1) = \tau_{m,m}(2,2) = 0. \tag{7.12}$$

In particular, the shifted residuals $\boldsymbol{r}_{m,m}(t) = \boldsymbol{b} - (A+tI)\boldsymbol{x}_{m,m}(t)$ *satisfy*

$$\boldsymbol{r}_{m,m}(t) = -\|\boldsymbol{b}\|_2[\tau_{m,m}(1,1)\boldsymbol{v}_{2m+1}, \tau_{m,m}(1,2)\boldsymbol{v}_{2m+1}][\hat{\boldsymbol{e}}_{2m-1},\hat{\boldsymbol{e}}_{2m}]^H(A_{m,m}+tI)^{-1}\hat{\boldsymbol{e}}_1 \tag{7.13}$$

and are therefore collinear to the basis vector $\boldsymbol{v}_{2m+1}$.

Proof. From the definition of $\tau_{m,m}$, we have $\tau_{m,m}(2,1) = \boldsymbol{v}_{2m+2}^H A\boldsymbol{v}_{2m-1}$. Now $A\boldsymbol{v}_{2m-1} \in A\mathcal{E}_{m,m}(A,\boldsymbol{b}) \subseteq \mathcal{E}_{m+1,m}(A,\boldsymbol{b})$. This is exactly the space against which $\boldsymbol{v}_{2m+2}$ is orthogonalized, so that $\boldsymbol{v}_{2m+2}^H A\boldsymbol{v}_{2m-1} = 0$. Similarly, we have $\tau_{m,m}(2,2) = \boldsymbol{v}_{2m+2}^H A\boldsymbol{v}_{2m}$ and $A\boldsymbol{v}_{2m} \in A\mathcal{E}_{m,m}(A,\boldsymbol{b}) \subseteq \mathcal{E}_{m+1,m}(A,\boldsymbol{b})$, so that the same argument as above can be applied to show that $\tau_{m,m}(2,2)$ is zero as well. This proves (7.12). Equation (7.13) then follows directly by inserting (7.12) into (7.10). Abbreviating $\boldsymbol{u} := [\hat{\boldsymbol{e}}_{2m-1},\hat{\boldsymbol{e}}_{2m}]^H(A_{m,m}+tI)^{-1}\hat{\boldsymbol{e}}_1$, the representation (7.13) becomes

$$\boldsymbol{r}_{m,m}(t) = -\|\boldsymbol{b}\|_2\left(\boldsymbol{u}(1)\tau_{m,m}(1,1) + \boldsymbol{u}(2)\tau_{m,m}(1,2)\right)\boldsymbol{v}_{2m+1}, \tag{7.14}$$

proving that all $\boldsymbol{r}_{m,m}(t)$ are collinear to $\boldsymbol{v}_{2m+1}$. Note that the collinearity factor of course depends on t, which is not directly visible from (7.14), as $\boldsymbol{u}$ implicitly depends on t. □

The result of Proposition 7.7 shows that all shifted extended Arnoldi residuals are again collinear to the next basis vector. Putting Proposition 7.6 and 7.7 in relation to Proposition 2.29, one sees that while the norm of the FOM residual depends only on the last entry of the first column of the inverse of the compressed matrix H_m, the norm of the extended Arnoldi iterate depends on the last and second to last entry of the first column of the inverse of $A_{m,m}$ (or shifted versions thereof).

When A is Hermitian, $A_{m,m} + tI$ is pentadiagonal and it is again possible to derive efficient update formulas for the lower left entries of its inverse and thus the norms of the shifted residuals, similarly to what was done in Section 6.4 for the tridiagonal matrix from the Lanczos process, by thoroughly investigating Gaussian elimination for pentadiagonal matrices. We do not give the details of this rather technical matter here, but just mention that such recursion relations for the residual norms in extended Krylov subspace methods which can be evaluated with cost $\mathcal{O}(1)$ per iteration have been presented in [118, Satz 4.4 & Satz 4.8].

Denoting the collinearity factor from (7.14) by $\psi_{m,m}(t)$, we can rewrite (7.9) as

$$f(A)\boldsymbol{b} - \boldsymbol{f}_{m,m} = \int_0^\infty \psi_{m,m}(t)(A + tI)^{-1} \,\mathrm{d}\mu(t)\boldsymbol{v}_{2m+1}. \tag{7.15}$$

Remark 7.8. Applying both (7.14) and (7.4)–(7.5) to the Stieltjes function $f(A) = (A + tI)^{-1}$, we find

$$\psi_{m,m}(t)(A + tI)^{-1}\boldsymbol{v}_{2m+1} = \frac{t^m}{w_{2m}(t)}(A + tI)^{-1}(-A)^{-m}w_{2m}(-A)\boldsymbol{b},$$

which shows that

$$\psi_{m,m}(t)\boldsymbol{v}_{2m+1} = \frac{t^m}{w_{2m}(t)}(-A)^{-m}w_{2m}(-A)\boldsymbol{b},$$

i.e., that the nodal Laurent polynomial $\frac{w_{2m}(-z)}{(-z)^m}$ evaluated in A maps $\boldsymbol{b}$ to a multiple of the next basis vector $\boldsymbol{v}_{2m+1}$, just as in the polynomial Krylov case. In particular, we find

$$\psi_{m,m}(t) = c \cdot \frac{t^m}{w_{2m}(t)}, \tag{7.16}$$

where $c \in \mathbb{C}$ is a constant independent of t, a relation which will be useful later on.

With the representation (7.15) we made a first step towards being able to use an algorithm similar to Algorithm 6.1 for computing error estimates for the extended Arnoldi approximation, as it gives rise to a natural analogue of Lemma 6.4, expressing the squared error norm as a quadratic form.

Lemma 7.9. *Let $A \in \mathbb{C}^{n\times n}$ be nonsingular, let $\boldsymbol{b} \in \mathbb{C}^n$, let f be a Stieltjes function of the form* (3.15)*, and let $\boldsymbol{f}_{m,m}$ be the extended Arnoldi approximation for $f(A)\boldsymbol{b}$. Then*

$$\|f(A)\boldsymbol{b} - \boldsymbol{f}_{m,m}\|_2^2 = \boldsymbol{v}_{2m+1}^H \widetilde{e}_{m,m}(A)^H \widetilde{e}_{m,m}(A)\boldsymbol{v}_{2m+1}, \tag{7.17}$$

where $\widetilde{e}_{m,m}(z)$ is given by

$$\widetilde{e}_{m,m}(z) = \int_0^\infty \frac{\psi_{m,m}(t)}{z + t} \,\mathrm{d}\mu(t). \tag{7.18}$$

Proof. The proof is completely analogous to that of Lemma 6.4, just using the error representation (7.15) and not exploiting Hermiticity of A. □

Using relation (7.16), we can prove the following result for the error function (7.18) in the Hermitian positive definite case.

Theorem 7.10. *Let the assumptions of Lemma 7.9 hold and let A be Hermitian positive definite in addition. Then, the error function $\widetilde{e}_{m,m}(z)$ from (7.18) is a multiple of another Stieltjes function, i.e.,*

$$\widetilde{e}_{m,m}(z) = c \cdot \int_0^\infty \frac{1}{z+t} \, \mathrm{d}\widetilde{\mu}(t)$$

for a nonnegative, monotonically increasing function $\widetilde{\mu}$ and a constant $c \in \mathbb{C}$.

Proof. We proceed similarly to the polynomial Krylov case, where the error function also was a multiple of a Stieltjes function. We define the function

$$\widetilde{\mu}(t) = \int_0^t \frac{\tau^m}{w_{2m}(\tau)} \, \mathrm{d}\mu(\tau), \tag{7.19}$$

where $w_{2m}(\tau) = \prod_{i=1}^{2m}(\tau + \theta_i)$ is again the nodal polynomial corresponding to the Ritz values $\theta_1, \ldots, \theta_{2m}$. As all Ritz values are positive when A is Hermitian positive definite, the function $\tau^m / w_{2m}(\tau)$ is nonnegative on $\mathbb{R}_0^+$. As μ is nonnegative and monotonically increasing, the integral on the right-hand side of (7.19), and thus the function $\widetilde{\mu}$, is nonnegative for all $t \geq 0$. Further, for $t_1 > t_0 \geq 0$, we have

$$\begin{aligned}
\widetilde{\mu}(t_1) &= \int_0^{t_1} \frac{\tau^m}{w_{2m}(\tau)} \, \mathrm{d}\mu(\tau) \\
&= \int_0^{t_0} \frac{\tau^m}{w_{2m}(\tau)} \, \mathrm{d}\mu(\tau) + \int_{t_0}^{t_1} \frac{\tau^m}{w_{2m}(\tau)} \, \mathrm{d}\mu(\tau) \\
&= \widetilde{\mu}(t_0) + \int_{t_0}^{t_1} \frac{\tau^m}{w_{2m}(\tau)} \, \mathrm{d}\mu(\tau) \\
&\geq \widetilde{\mu}(t_0).
\end{aligned}$$

This shows that $\widetilde{\mu}$ is nonnegative and monotonically increasing. To show that $\widetilde{\mu}$ generates a Stieltjes function, we have to check the condition (2.16), i.e., whether

$$\int_0^\infty \frac{1}{1+t} \, \mathrm{d}\widetilde{\mu}(t) < \infty.$$

For this, first note that (7.19) implies

$$\mathrm{d}\widetilde{\mu}(t) = \frac{t^m}{w_{2m}(t)} \, \mathrm{d}\mu(t),$$

and that the function $t^m/w_{2m}(t)$ is bounded on $\mathbb{R}_0^+$ by some constant $d > 0$, as the degree of its denominator exceeds the degree of its numerator. Therefore, we have

$$\int_0^\infty \frac{1}{1+t}\,\mathrm{d}\widetilde{\mu}(t) = \int_0^\infty \frac{1}{1+t}\frac{t^m}{w_{2m}(t)}\,\mathrm{d}\mu(t) \leq d \cdot \int_0^\infty \frac{1}{1+t}\,\mathrm{d}\mu(t) < \infty, \tag{7.20}$$

where the last inequality in (7.20) follows because μ satisfies the condition (2.16). Summarizing, we have shown that the function

$$\int_0^\infty \frac{1}{z+t}\,\mathrm{d}\widetilde{\mu}(t)$$

is a Stieltjes function. Inserting the relation (7.16) into the error function representation (7.18), we have that

$$\widetilde{e}_{m,m}(z) = c \cdot \int_0^\infty \frac{1}{z+t}\,\mathrm{d}\widetilde{\mu}(t),$$

with the constant c from (7.16). This completes the proof of the theorem. □

The result of Theorem 7.10 serves two purposes. On the one hand, it again guarantees that the integral in the error function representation (7.18) is always finite, and on the other hand, it shows that it is in principle also possible to compute error bounds for extended Krylov subspace approximations by pairs of Gauss and Gauss–Radau quadrature rules when A is Hermitian positive definite. In this case, the error norm representation (7.17) becomes

$$\|f(A)\boldsymbol{b} - \boldsymbol{f}_{m,m}\|_2^2 = \boldsymbol{v}_{2m+1}^H \widetilde{e}_{m,m}(A)^2 \boldsymbol{v}_{2m+1},$$

and the function $\widetilde{e}_{m,m}(z)^2$ is completely monotonic, as it is the product of two (multiples of) Stieltjes functions, just as in the polynomial Krylov case. We have therefore just proven the following analogue to Theorem 6.5 for extended Krylov subspace methods.

Theorem 7.11. *Let $A \in \mathbb{C}^{n\times n}$ be Hermitian positive definite, let $\boldsymbol{b} \in \mathbb{C}^n$, let f be a Stieltjes function of the form* (3.15)*, and let $\boldsymbol{f}_{m,m}$ be the extended Arnoldi approximation to $f(A)\boldsymbol{b}$. Let $\boldsymbol{v}_{2m+1}$ be the $(2m+1)$st extended Arnoldi basis vector. Denote by $H_k^{(2)}$ the tridiagonal matrix resulting from k steps of the Lanczos process applied to A and $\boldsymbol{v}_{2m+1}$ and by $\widetilde{H}_{k+1}^{(2)}$ the modification of $H_k^{(2)}$ according to* (6.6)*. Then*

$$\hat{\boldsymbol{e}}_1^H \widetilde{e}_{m,m}\left(H_k^{(2)}\right)^2 \hat{\boldsymbol{e}}_1 \leq \|f(A)\boldsymbol{b} - \boldsymbol{f}_{m,m}\|_2^2 \leq \hat{\boldsymbol{e}}_1^H \widetilde{e}_{m,m}\left(\widetilde{H}_{k+1}^{(2)}\right)^2 \hat{\boldsymbol{e}}_1, \tag{7.21}$$

where $\widetilde{e}_{m,m}(z)$ is the error function from (7.18)*.*

Just as in the polynomial Krylov case, we are of course again in the situation that we are not able to exactly evaluate the error function $\widetilde{e}_{m,m}(z)$ but have to use numerical quadrature instead. For computing guaranteed bounds for the extended Arnoldi error via (7.21), we thus also have to be careful about the sign of the error in this inner quadrature rule. The result of Proposition 6.6 applies in the extended case as well, so that it is sufficient to find a quadrature rule which computes lower or upper bounds for the scalar function $g(t) = \frac{t^m}{w_{2m}(t)}\frac{1}{z+t}$. However, the integrand in this case is less well-behaved than in the polynomial Krylov case, where $\frac{1}{w_m(t)}\frac{1}{z+t}$ is monotonically decreasing as a function of t on $\mathbb{R}_0^+$. This is not true in the extended Krylov case in general. Instead, the function $\frac{t^m}{w_{2m}(t)}$ has the value zero at $t = 0$, tends to zero for $t \to \infty$, and has exactly one local maximum in between. Noting that the value of the function is closely related to the residual norms produced by the extended Krylov subspace method for linear systems, cf. Remark 7.8, this behavior is indeed quite natural: The linear system (7.6) corresponding to shift $t = 0$ is solved exactly in the very first step of the extended Arnoldi method (as one applies a multiplication with A^{-1}), being in line with the function $\frac{t^m}{w_{2m}(t)}$ attaining the value zero at $t = 0$. When increasing the shift, the solutions of the corresponding shifted systems increasingly differ from the solution for shift $t = 0$, so that they are harder to find for the method and the residual norms increase. At some point, however, the better conditioning of the matrices $A + tI$ for large shifts t becomes noticeable and the method again finds iterates with smaller residual norms as the systems become easier to solve (the point at which this change happens is exactly the local maximum of $\frac{t^m}{w_{2m}(t)}$ on $\mathbb{R}_0^+$). An illustration of the function $\frac{t^m}{w_{2m}(t)}\frac{1}{z+t}$ which we need to approximate by quadrature when computing error bounds in the extended Arnoldi method for the Gaussian Markov random field model problem is given in Figure 7.1.

This structure of the integrand makes it much harder to find suitable integration rules which provide lower and upper bounds, but as we already mentioned when discussing the polynomial Krylov case, the error in the inner quadrature rule is typically much smaller than the error in the outer quadrature rule, so that it seldom dominates the overall error and does in general not prevent the computed estimates from being bounds. This is also demonstrated in the numerical experiments presented in Section 7.4, but we stress that one has no guarantee for this to be true.

To be able to generalize Algorithm 6.1 to the extended Arnoldi approximation by computing error norm estimates via Gauss quadrature for (7.17), we still need to resolve one issue. We again need a way to find the matrix $H_k^{(2)}$ from a secondary Lanczos process without performing additional matrix vector multiplications with A, i.e., a result similar to that of Theorem 6.7. As the proofs of Theorem 6.7 and 6.9 largely relied on the nestedness properties of Krylov subspaces, and the extended Krylov space $\mathcal{E}_{m,m}(A, \boldsymbol{b})$ contains $\mathcal{K}_m(A, \boldsymbol{b})$, we can expect a similar

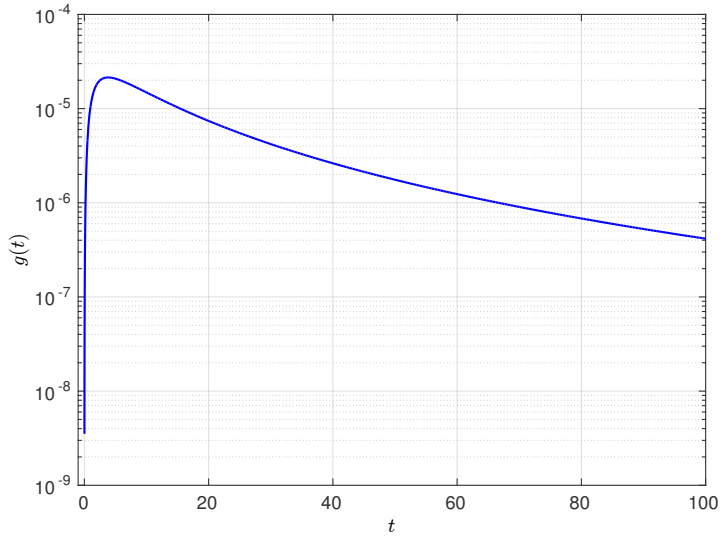


Figure 7.1: Nonmonotonic integrand $g(t) = \frac{t^m}{w_{2m}(t)} \frac{1}{z+t}$ from the second step of the extended Arnoldi method for approximating $A^{-1/2}\boldsymbol{z}$ in the Gaussian Markov random field model problem (for $z = 2$).

result to hold in this situation as well. This will be the topic of Section 7.3.

Another obvious idea arising in the context of approximating the quadratic form on the right-hand side of (7.17) is to use *rational Gauss quadrature rules* [95, 96], i.e., quadrature rules which are exact for Laurent polynomials of a certain degree, instead of standard Gauss rules. In [95, 96] it is shown that these rational Gauss rules are intimately related to extended Krylov subspace methods for Hermitian A. To be precise, it is shown that

$$\hat{\boldsymbol{e}}_1^H h(A_{k,k}) \hat{\boldsymbol{e}}_1 \approx \boldsymbol{v}^H h(A) \boldsymbol{v}, \tag{7.22}$$

where $A_{k,k}$ is the pentadiagonal matrix arising from k steps of the extended Arnoldi method for A and $\boldsymbol{v}$, can be interpreted as a rational Gauss quadrature rule with $2k$ nodes (the eigenvalues of $A_{k,k}$) and that it provides lower bounds for the right-hand side of (7.22) if h satisfies the condition

$$\frac{\mathrm{d}^{4k}}{\mathrm{d}z^{4k}} \left(z^{2(k-1)} h(z) \right) \geq 0.$$

Unfortunately, this condition is in general not fulfilled for the error function $\widetilde{e}_{m,m}(z)$, so that we cannot expect to compute error bounds by rational Gauss quadrature rules. On the other hand, one can hope that rational Gauss rules are more accurate than standard Gauss rules in many situations. We will compare both kinds of rules for estimating the error in extended Krylov subspace methods in the experiments presented in Section 7.4. We just mention that in [96] also

rational Gauss–Radau rules are constructed, but we will not go into detail concerning this topic here, as the main use of these rules was to bracket $\boldsymbol{v}^H h(A)\boldsymbol{v}$, which is not possible for $h = \widetilde{e}_{m,m}$ as just explained.

Naively approximating (7.17) by a rational Gauss rule would not only require additional multiplications with A but also additional linear system solves. To avoid this, we will also consider the possibility of using restart recovery for cheaply constructing the matrix $A^{(2)}_{k,k}$ corresponding to a secondary extended Arnoldi method for A and $\boldsymbol{v}_{2m+1}$ in the next section.

7.3 Restart recovery in extended Krylov methods

In this section we investigate how to compute error estimates with low computational cost based on the representation (7.17) of the error norm in the extended Arnoldi method. We begin by proving a direct generalization of Theorem 6.9 to the case of extended Krylov methods, which makes use of the fact that polynomial Krylov spaces are subspaces of (suitably chosen) extended Krylov spaces.

Theorem 7.12. *Let the columns of $V_{m+k+1,m+k+1}$ be the orthonormal basis of $\mathcal{E}_{m+k+1,m+k+1}(A, \boldsymbol{v}_1)$ from $m+k+1$ steps of the extended Arnoldi method for A and $\boldsymbol{v}_1$ and let $A_{m+k+1,m+k+1} = V^H_{m+k+1,m+k+1} A V_{m+k+1,m+k+1}$. Further, let $\hat{H}$ denote the matrix resulting from k steps of Arnoldi's method applied to $A_{m+k+1,m+k+1}$ and $\hat{\boldsymbol{e}}_{2m+1}$. Then $\hat{H} = H^{(2)}_k$, where $H^{(2)}_k$ denotes the matrix resulting from k iterations of Arnoldi's method for A and $\boldsymbol{v}_{2m+1}$.*

Proof. The proof is completely analogous to the one of Theorem 6.9, using the fact that $\mathcal{K}_{k+1}(A, \boldsymbol{v}_{2m+1}) \subseteq \mathcal{E}_{m+k+1,m+1}(A, \boldsymbol{v}_1) \subseteq \mathcal{E}_{m+k+1,m+k+1}(A, \boldsymbol{v}_1)$ because $\boldsymbol{v}_{2m+1} \in \mathcal{E}_{m+1,m+1}(A, \boldsymbol{v}_1)$. Therefore it is again possible to represent the Arnoldi basis vectors of $\mathcal{K}_{k+1}(A, \boldsymbol{v}_{2m+1})$ in terms of the basis $V_{m+k+1,m+k+1}$ and proceed to construct an Arnoldi relation involving the corresponding coefficient matrix Q_k and the matrix $A_{m+k+1,m+k+1}$. □

The result of Theorem 7.12 shows that restart recovery similar to the polynomial Krylov case is possible in extended methods. The following proposition shows that in the Hermitian case, it is again not necessary to perform the secondary Lanczos process with the full matrix $A_{m+k+1,m+k+1}$, but only with a sub-block of constant size.

Proposition 7.13. *Let the assumptions of Theorem 7.12 hold and let A be Hermitian positive definite in addition. Then, k iterations of the Lanczos process applied to the lower right $(4k+1) \times (4k+1)$ sub-block of $A_{m+k+1,m+k+1}$ and*

$\hat{\boldsymbol{e}}_{2k+1}$ produce the same matrix $H_k^{(2)}$ as k iterations of the Lanczos process for $A_{m+k+1,m+k+1}$ and $\hat{\boldsymbol{e}}_{2m+1}$.

Proof. The result can be proven by carefully investigating the nonzero structure of the matrix $A_{m+k+1,m+k+1}$. The decomposition corresponding to k steps of Lanczos for $A_{m+k+1,m+k+1}$ and $\hat{\boldsymbol{e}}_{2m+1}$ is given by

$$A_{m+k+1,m+k+1}\widetilde{V}_k = \widetilde{V}_k H_k^{(2)} + h_{k+1,k}^{(2)} \widetilde{\boldsymbol{v}}_{k+1} \hat{\boldsymbol{e}}_k^H. \tag{7.23}$$

Given that $A_{m+k+1,m+k+1}$ is pentadiagonal, $H_k^{(2)}$ is tridiagonal and $\widetilde{\boldsymbol{v}}_1 = \hat{\boldsymbol{e}}_{2m+1}$ has its only nonzero entry at position $2m+1$, by comparing nonzero structures, we find that $\widetilde{\boldsymbol{v}}_j, j = 2, \ldots, k$ has nonzero entries only in position $2m+1-2j, \ldots, 2m+1+2j$. In particular, the rows $1, \ldots, 2m-2k$ of $\widetilde{V}_k$ are all zero and do not make a contribution to (7.23) and we can thus omit the corresponding rows and columns of $A_{m+k+1,m+k+1}$. Given that the matrix is of size $2(m+k+1) \times 2(m+k+1)$, the remaining lower right sub-block is thus of size $(2(m+k+1)-(2m-2k)) \times (2(m+k+1)-(2m-2k)) = (4k+1) \times (4k+1)$. In addition, omitting the first $2m-2k$ rows in $\hat{\boldsymbol{e}}_{2m+1} \in \mathbb{R}^{2(m+k+1)}$ results in $\hat{\boldsymbol{e}}_{2k+1} \in \mathbb{R}^{4k+1}$, thus proving the assertion of the proposition. □

We now have all the tools available to be able to use an extended Krylov subspace analogue to Algorithm 6.1. We do not give a detailed algorithm here, as it is just a straightforward adaption of the techniques from Algorithm 6.1 to Algorithm 7.1.

It is also possible to use restart recovery to construct the matrix $A_{k,k}^{(2)}$ corresponding to k steps of the *extended* Arnoldi method for A and $\boldsymbol{v}_{2m+1}$, instead of the matrix $H_k^{(2)}$ corresponding to *standard* Arnoldi, which can then be used for computing error estimates based on rational Gauss quadrature via (7.22). Before we can prove this, we need a few auxiliary results. First, we need to show that the matrix $B_{m,m} = V_{m,m}^H A^{-1} V_{m,m}$ fulfills a relation similar to (7.2). A similar result was shown in [96] (where the authors refer to the matrix $B_{m,m}$ as the *inverse projection matrix*), but exclusively for the Hermitian case, and for a non-blockwise generation of the basis vectors, which leads to a slightly different nonzero structure of $B_{m,m}$. We therefore give a sketch of the proof of the following result which is adapted to our situation.

Lemma 7.14. *Let $A \in \mathbb{C}^{n\times n}$ be nonsingular, let $\boldsymbol{b} \in \mathbb{C}^n$ and let the columns of $[V_{m,m}, \boldsymbol{v}_{2m+1}, \boldsymbol{v}_{2m+2}]$ be the orthonormal basis of $\mathcal{E}_{m,m}(A, \boldsymbol{b})$ computed by Algorithm 7.1. Then the matrix $B_{m,m} = V_{m,m}^H A^{-1} V_{m,m}$ satisfies*

$$A^{-1}V_{m,m} = V_{m,m}B_{m,m} + [\boldsymbol{v}_{2m+1}, \boldsymbol{v}_{2m+2}]\sigma_{m,m}[\hat{\boldsymbol{e}}_{2m-1}, \hat{\boldsymbol{e}}_{2m}]^H, \tag{7.24}$$

where $\sigma_{m,m} = [\boldsymbol{v}_{2m+1}, \boldsymbol{v}_{2m+2}]A^{-1}[\boldsymbol{v}_{2m-1}, \boldsymbol{v}_{2m}]$.

Proof. For $j \leq 2(m-1)$, we have $\boldsymbol{v}_j \in \mathcal{E}_{m-1,m-1}(A, \boldsymbol{b})$ and thus $A^{-1}\boldsymbol{v}_j \in \mathcal{E}_{m,m}(A, \boldsymbol{b})$. As $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{2m}$ form an orthonormal basis of $\mathcal{E}_{m,m}(A, \boldsymbol{b})$, we can thus express $A^{-1}\boldsymbol{v}_j$ as

$$A^{-1}\boldsymbol{v}_j = \sum_{i=1}^{2m} \left(\boldsymbol{v}_i^H A^{-1}\boldsymbol{v}_j\right) \boldsymbol{v}_i. \tag{7.25}$$

Analogously, for $2m-1 \leq j \leq 2m$, we have that $A^{-1}\boldsymbol{v}_j \in \mathcal{E}_{m+1,m+1}(A, \boldsymbol{b})$ and thus

$$A^{-1}\boldsymbol{v}_j = \sum_{i=1}^{2m+2} \left(\boldsymbol{v}_i^H A^{-1}\boldsymbol{v}_j\right) \boldsymbol{v}_i. \tag{7.26}$$

Recasting (7.25) and (7.26) into matrix form proves the assertion. □

Next, we investigate the nonzero structure of $A_{m,m}$ and $B_{m,m}$. This is again very similar to results already known for the Hermitian case [96, 124], and we refrain from giving a proof this time, as it is completely straightforward. One just needs to carefully examine which values of the form $\boldsymbol{v}_i^H A\boldsymbol{v}_j$ or $\boldsymbol{v}_i^H A^{-1}\boldsymbol{v}_j$ are known to be zero, because $A\boldsymbol{v}_j$ or $A^{-1}\boldsymbol{v}_j$ lies in $\operatorname{span}\{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{i-1}\}$, similarly to what was done in the proof of Proposition 7.7. By doing so, one finds that

$$A_{m,m} = \begin{bmatrix} \alpha_{1,1} & \alpha_{1,2} & \alpha_{1,3} & \alpha_{1,4} & \alpha_{1,5} & \alpha_{1,6} & \alpha_{1,7} & \cdots \\ \alpha_{2,1} & \alpha_{2,2} & \alpha_{2,3} & \alpha_{2,4} & \alpha_{2,5} & \alpha_{2,6} & \alpha_{2,7} & \cdots \\ \alpha_{3,1} & \alpha_{3,2} & \alpha_{3,3} & \alpha_{3,4} & \alpha_{3,5} & \alpha_{3,6} & \alpha_{3,7} & \cdots \\ & & \alpha_{4,3} & \alpha_{4,4} & \alpha_{4,5} & \alpha_{4,6} & \alpha_{4,7} & \cdots \\ & & \alpha_{5,3} & \alpha_{5,4} & \alpha_{5,5} & \alpha_{5,6} & \alpha_{5,7} & \cdots \\ & & & & \alpha_{6,5} & \alpha_{6,6} & \alpha_{6,7} & \cdots \\ & & & & \alpha_{7,5} & \alpha_{7,6} & \alpha_{7,7} & \cdots \\ & & & & & & \vdots & \ddots \end{bmatrix} \tag{7.27}$$

and

$$B_{m,m} = \begin{bmatrix} \beta_{1,1} & \beta_{1,2} & \beta_{1,3} & \beta_{1,4} & \beta_{1,5} & \beta_{1,6} & \beta_{1,7} & \cdots \\ \beta_{2,1} & \beta_{2,2} & \beta_{2,3} & \beta_{2,4} & \beta_{2,5} & \beta_{2,6} & \beta_{2,7} & \cdots \\ & \beta_{3,2} & \beta_{3,3} & \beta_{3,4} & \beta_{3,5} & \beta_{3,6} & \beta_{3,7} & \cdots \\ & \beta_{4,2} & \beta_{4,3} & \beta_{4,4} & \beta_{4,5} & \beta_{4,6} & \beta_{4,7} & \cdots \\ & & & \beta_{5,4} & \beta_{5,5} & \beta_{5,6} & \beta_{5,7} & \cdots \\ & & & \beta_{6,4} & \beta_{6,5} & \beta_{6,6} & \beta_{6,7} & \cdots \\ & & & & & \beta_{7,6} & \beta_{7,7} & \cdots \\ & & & & & \vdots & \vdots & \ddots \end{bmatrix}. \tag{7.28}$$

We are now in a position to prove a result which can be seen as an extended Krylov analogue to Lemma 2.23. As Lemma 2.23 has a strong relation to the *implicit Q theorem*, cf., e.g, [129, Chapter 2, Theorem 3.3], we can think of the following result as an *extended implicit Q theorem*. We briefly mention here that

a version of the implicit Q theorem for general rational Krylov subspaces was recently proven in [16]. However, the formulation of the theorem given in [16] is not compatible with the block-wise generation of the extended Krylov subspace we use. In addition, the notion of "essential uniqueness" used in [16] is weaker than what we can use here, as extended Krylov subspaces allow for fewer additional degrees of freedom than general rational Krylov subspaces. We can thus formulate the result in a more concise way, which is more reminiscent of the situation faced when dealing with polynomial Krylov spaces.

Theorem 7.15. *Let $A \in \mathbb{C}^{n\times n}$ be nonsingular. Assume there exist matrices $[V_{m,m}, \boldsymbol{v}_{2m+1}, \boldsymbol{v}_{2m+2}], [\widetilde{V}_{m,m}, \widetilde{\boldsymbol{v}}_{2m+1}, \widetilde{\boldsymbol{v}}_{2m+2}] \in \mathbb{C}^{n\times 2(m+1)}$ with orthonormal columns and $\boldsymbol{v}_1 = \widetilde{\boldsymbol{v}}_1$ as well as matrices $A_{m,m}, \widetilde{A}_{m,m}$ and $B_{m,m}, \widetilde{B}_{m,m}$ with nonzero structure (7.27) and (7.28), respectively, and $\tau_{m,m}, \widetilde{\tau}_{m,m}, \sigma_{m,m}, \widetilde{\sigma}_{m,m} \in \mathbb{C}^{2\times 2}$ such that the relations*

$$\begin{aligned} AV_{m,m} &= V_{m,m}A_{m,m} + [\boldsymbol{v}_{2m+1}, \boldsymbol{v}_{2m+2}]\tau_{m,m}[\hat{\boldsymbol{e}}_{2m-1}, \hat{\boldsymbol{e}}_{2m}]^H, && (7.29)\\ A^{-1}V_{m,m} &= V_{m,m}B_{m,m} + [\boldsymbol{v}_{2m+1}, \boldsymbol{v}_{2m+2}]\sigma_{m,m}[\hat{\boldsymbol{e}}_{2m-1}, \hat{\boldsymbol{e}}_{2m}]^H, && (7.30)\\ A\widetilde{V}_{m,m} &= \widetilde{V}_{m,m}\widetilde{A}_{m,m} + [\widetilde{\boldsymbol{v}}_{2m+1}, \widetilde{\boldsymbol{v}}_{2m+2}]\widetilde{\tau}_{m,m}[\hat{\boldsymbol{e}}_{2m-1}, \hat{\boldsymbol{e}}_{2m}]^H, && (7.31)\\ A^{-1}\widetilde{V}_{m,m} &= \widetilde{V}_{m,m}\widetilde{B}_{m,m} + [\widetilde{\boldsymbol{v}}_{2m+1}, \widetilde{\boldsymbol{v}}_{2m+2}]\widetilde{\sigma}_{m,m}[\hat{\boldsymbol{e}}_{2m-1}, \hat{\boldsymbol{e}}_{2m}]^H && (7.32) \end{aligned}$$

hold. Then $V_{m,m}, A_{m,m}, B_{m,m}$ are essentially equal to $\widetilde{V}_{m,m}, \widetilde{A}_{m,m}, \widetilde{B}_{m,m}$ in the sense that there exists a unitary diagonal matrix $D_m = \operatorname{diag}(d_1, \ldots, d_m) \in \mathbb{C}^{m\times m}$ with $d_1 = 1$ such that $\widetilde{V}_{m,m} = V_{m,m}D_m$, $\widetilde{A}_{m,m} = D_m^H A_{m,m} D_m$ and $\widetilde{B}_{m,m} = D_m^H B_{m,m} D_m$.

Proof. The proof of the theorem, which is constructive, proceeds column by column through the relations (7.29)–(7.32) and defines the values d_i according to the assertion of the theorem. We begin by putting $d_1 = 1$. We denote the entries of $A_{m,m}, \widetilde{A}_{m,m}, B_{m,m}$ and $\widetilde{B}_{m,m}$ by $\alpha_{i,j}, \widetilde{\alpha}_{i,j}, \beta_{i,j}$ and $\widetilde{\beta}_{i,j}$, respectively.

The first column of (7.30) reads (taking the nonzero structure (7.28) into account)

$$A^{-1}\boldsymbol{v}_1 = \beta_{1,1}\boldsymbol{v}_1 + \beta_{2,1}\boldsymbol{v}_2,$$

which directly implies that $\beta_{1,1} = \boldsymbol{v}_1^H A^{-1}\boldsymbol{v}_1$ and $\beta_{2,1} = \boldsymbol{v}_2^H A^{-1}\boldsymbol{v}_1$. In the same way, the first column of (7.32) implies $\widetilde{\beta}_{1,1} = \widetilde{\boldsymbol{v}}_1^H A^{-1}\widetilde{\boldsymbol{v}}_1$. As $\boldsymbol{v}_1 = \widetilde{\boldsymbol{v}}_1$ by assumption, we thus have $\beta_{1,1} = \widetilde{\beta}_{1,1}$. Using this fact and rearranging the first columns of (7.30) and (7.32) gives

$$\boldsymbol{v}_2 = (A^{-1}\boldsymbol{v}_1 - \beta_{1,1}\boldsymbol{v}_1)/\beta_{2,1} \text{ and } \widetilde{\boldsymbol{v}}_2 = (A^{-1}\boldsymbol{v}_1 - \beta_{1,1}\boldsymbol{v}_1)/\widetilde{\beta}_{2,1}.$$

This directly shows that

$$\widetilde{\boldsymbol{v}}_2 = \beta_{2,1}/\widetilde{\beta}_{2,1}\boldsymbol{v}_2. \tag{7.33}$$

Therefore, we set $d_2 = \beta_{2,1}/\widetilde{\beta}_{2,1}$. As $\|\boldsymbol{v}_2\|_2 = \|\widetilde{\boldsymbol{v}}_2\|_2 = 1$ we have that $|d_2| = 1$ as well. Further, $\widetilde{\beta}_{2,1} = \beta_{2,1}/d_2 = \overline{d}_2\beta_{2,1}$.

Next, consider the first column of (7.29) and (7.31), again taking into account its nonzero structure given in (7.27), i.e.,

$$A\boldsymbol{v}_1 = \alpha_{1,1}\boldsymbol{v}_1 + \alpha_{2,1}\boldsymbol{v}_2 + \alpha_{3,1}\boldsymbol{v}_3 \text{ and } A\widetilde{\boldsymbol{v}}_1 = \widetilde{\alpha}_{1,1}\widetilde{\boldsymbol{v}}_1 + \widetilde{\alpha}_{2,1}\widetilde{\boldsymbol{v}}_2 + \widetilde{\alpha}_{3,1}\widetilde{\boldsymbol{v}}_3, \tag{7.34}$$

which, similarly to the above gives $\alpha_{1,1} = \widetilde{\alpha}_{1,1} = \boldsymbol{v}_1^H A\boldsymbol{v}_1$ and $\alpha_{2,1} = \boldsymbol{v}_2^H A\boldsymbol{v}_1$. Using (7.33) together with the definition of d_2, we further find

$$\widetilde{\alpha}_{2,1} = \widetilde{\boldsymbol{v}}_2^H A\boldsymbol{v}_1 = \overline{d}_2\alpha_{2,1}.$$

Rearranging (7.34) gives

$$\boldsymbol{v}_3 = (A\boldsymbol{v}_1 - \alpha_{1,1}\boldsymbol{v}_1 - \alpha_{2,1}\boldsymbol{v}_2)/\alpha_{3,1} \text{ and } \widetilde{\boldsymbol{v}}_3 = (A\widetilde{\boldsymbol{v}}_1 - \widetilde{\alpha}_{1,1}\widetilde{\boldsymbol{v}}_1 - \widetilde{\alpha}_{2,1}\widetilde{\boldsymbol{v}}_2)/\widetilde{\alpha}_{3,1}.$$

By inserting the relations $\widetilde{\alpha}_{2,1} = \overline{d}_2\alpha_{2,1}$, $\widetilde{\boldsymbol{v}}_2 = d_2\boldsymbol{v}_2$ and $\overline{d}_2 d_2 = 1$, we find

$$\widetilde{\boldsymbol{v}}_3 = (A\boldsymbol{v}_1 - \alpha_{1,1}\boldsymbol{v}_1 - \alpha_{2,1}\boldsymbol{v}_2)/\widetilde{\alpha}_{3,1}.$$

showing that $\widetilde{\boldsymbol{v}}_3 = \alpha_{3,1}/\widetilde{\alpha}_{3,1}\boldsymbol{v}_3$, so that we put $d_3 = \alpha_{3,1}/\widetilde{\alpha}_{3,1}$. With the same reasoning as for d_2, we have $|d_3| = 1$ and $\widetilde{\alpha}_{3,1} = \overline{d_3}\alpha_{3,1}$. Proceeding similarly with the second column of (7.29) and (7.31), exploiting the fact that

$$\widetilde{\boldsymbol{v}}_1 = d_1\boldsymbol{v}_1, \qquad \widetilde{\boldsymbol{v}}_2 = d_2\boldsymbol{v}_2 \quad \text{and} \quad \widetilde{\boldsymbol{v}}_3 = d_3\boldsymbol{v}_3, \tag{7.35}$$

direct calculations show that

$$\widetilde{\alpha}_{1,2} = \overline{d_1}d_2\alpha_{1,2}, \quad \widetilde{\alpha}_{2,2} = \overline{d_2}d_2\alpha_{2,2} = \alpha_{2,2} \quad \text{and} \quad \widetilde{\alpha}_{3,2} = \overline{d_3}d_2\alpha_{3,2}.$$

We have thus shown that, with the choices made for d_i so far, the first two columns of $D_m^H A_{m,m} D_m$ and $\widetilde{A}_{m,m}$ agree. We proceed with the second columns of (7.30) and (7.32), which give $\beta_{i,2} = \boldsymbol{v}_i^H A^{-1}\boldsymbol{v}_2$ and $\widetilde{\beta}_{i,2} = \widetilde{\boldsymbol{v}}_i^H A^{-1}\widetilde{\boldsymbol{v}}_2$ for $i = 1, \ldots, 4$. Inserting the relations (7.35), as before, yields

$$\widetilde{\beta}_{1,2} = \overline{d_1}d_2\beta_{1,2}, \quad \widetilde{\beta}_{2,2} = \overline{d_2}d_2\beta_{2,2} = \beta_{2,2} \quad \text{and} \quad \widetilde{\beta}_{3,2} = \overline{d_3}d_2\beta_{3,2}. \tag{7.36}$$

We rearrange the second columns of (7.30) and (7.32) to give

$$\boldsymbol{v}_4 = (A^{-1}\boldsymbol{v}_2 - \beta_{1,2}\boldsymbol{v}_1 - \beta_{2,2}\boldsymbol{v}_2 - \beta_{3,2}\boldsymbol{v}_3)/\beta_{4,2}$$

and

$$\widetilde{\boldsymbol{v}}_4 = (A^{-1}\widetilde{\boldsymbol{v}}_2 - \widetilde{\beta}_{1,2}\widetilde{\boldsymbol{v}}_1 - \widetilde{\beta}_{2,2}\widetilde{\boldsymbol{v}}_2 - \widetilde{\beta}_{3,2}\widetilde{\boldsymbol{v}}_3)/\widetilde{\beta}_{4,2}. \tag{7.37}$$

Using (7.35) and (7.36), we can rewrite (7.37) as

$$\widetilde{\boldsymbol{v}}_4 = (d_2A^{-1}\boldsymbol{v}_2 - \beta_{1,2}d_2\boldsymbol{v}_1 - \beta_{2,2}d_2\boldsymbol{v}_2 - \beta_{3,2}d_2\boldsymbol{v}_3)/\widetilde{\beta}_{4,2}.$$

Thus, $\widetilde{\boldsymbol{v}}_4 = d_2\beta_{4,2}/\widetilde{\beta}_{4,2}\boldsymbol{v}_4$. Putting $d_4 = d_2\beta_{4,2}/\widetilde{\beta}_{4,2}$, we have $\beta_{4,2} = \overline{d}_4 d_2 \beta_{4,2}$. We have thus also proven the relations from the assertion of the theorem for the first two columns of $B_{m,m}$ and $\widetilde{B}_{m,m}$. One can now continue in the same way as demonstrated until here as there is always either one column of (7.29) and (7.31) or one column of (7.30) and (7.32) where the relation $\widetilde{\boldsymbol{v}}_i = d_i\boldsymbol{v}_i$ has already been shown for all but one of the pairs $\boldsymbol{v}_i, \widetilde{\boldsymbol{v}}_i$ appearing in the equation. This then allows rearranging the equations such that they prove the assertion for the next basis vector. We refrain from explicitly presenting the inductive step here, as it is rather straightforward, but very technical and does not give any more insight than what was presented up to this point. □

With help of Theorem 7.15, we can now finally formulate the result needed for performing extended restart recovery.

Theorem 7.16. *Let $A \in \mathbb{C}^{n\times n}$ and let the columns of $V_{m+k+2,m+k+2}$ be the orthonormal basis of $\mathcal{E}_{m+k+2,m+k+2}(A, \boldsymbol{v}_1)$ from $m+k+2$ steps of the extended Arnoldi method for A and $\boldsymbol{v}_1$ and let $A_{m+k+2,m+k+2} = V_{m+k+2,m+k+2}^H A V_{m+k+2,m+k+2}$ be nonsingular. Further, let $\hat{A}_{k,k}$ denote the matrix resulting from k steps of the extended Arnoldi method applied to $A_{m+k+2,m+k+2}$ and $\hat{\boldsymbol{e}}_{2m+1}$. Then $\hat{A}_{k,k} = D_k^H A_{k,k}^{(2)} D_k$, where $A_{k,k}^{(2)}$ denotes the matrix resulting from k iterations of the extended Arnoldi method for A and $\boldsymbol{v}_{2m+1}$ and D_k is a unitary diagonal matrix with $d_{1,1} = 1$.*

Proof. The proof proceeds similarly to the one of Theorem 6.9, with the difference that we have to consider one additional "artificial" extended Arnoldi iteration in order to find the relation for the inverse projection matrix at the end of the proof. In the remainder of the proof, we use the shorthand notation $\widetilde{m} = m + k + 2$.

Let the extended Arnoldi decomposition arising from $k+1$ steps of the extended Arnoldi method for A and $\boldsymbol{v}_{2m+1}$ be given as

$$A\widetilde{V}_{k+1,k+1} = \widetilde{V}_{k+1,k+1}A_{k+1,k+1}^{(2)} + [\widetilde{\boldsymbol{v}}_{2k+3}, \widetilde{\boldsymbol{v}}_{2k+4}]\widetilde{\tau}_{k+1,k+1}[\hat{\boldsymbol{e}}_{2k+1}, \hat{\boldsymbol{e}}_{2k+2}]^H. \tag{7.38}$$

As $\boldsymbol{v}_{2m+1} \in \mathcal{E}_{m+1,m+1}(A, \boldsymbol{v}_1)$, we obviously have that

$$\mathcal{E}_{k+2,k+2}(A, \boldsymbol{v}_{2m+1}) \subseteq \mathcal{E}_{\widetilde{m},\widetilde{m}}(A, \boldsymbol{v}_1).$$

Therefore, the basis vectors $\widetilde{\boldsymbol{v}}_1, \ldots, \widetilde{\boldsymbol{v}}_{2k+4}$ generated by the extended Arnoldi method for A and $\boldsymbol{v}_{2m+1}$ all lie in $\mathcal{E}_{\widetilde{m},\widetilde{m}}(A, \boldsymbol{v}_1)$ and can thus be written as linear combinations of the basis vectors $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{2\widetilde{m}}$, i.e.,

$$[\widetilde{V}_{k+1,k+1}, \widetilde{\boldsymbol{v}}_{2k+3}, \widetilde{\boldsymbol{v}}_{2k+4}] = V_{\widetilde{m},\widetilde{m}}[Q_{k+1,k+1}, \boldsymbol{q}_{2k+3}, \boldsymbol{q}_{2k+4}] \tag{7.39}$$

for some matrix $Q_{k+1,k+1} \in \mathbb{C}^{2\widetilde{m}\times 2(k+1)}$. As $[\widetilde{V}_{k+1,k+1}, \widetilde{\boldsymbol{v}}_{2k+3}, \widetilde{\boldsymbol{v}}_{2k+4}]$ and $V_{\widetilde{m},\widetilde{m}}$ both have orthonormal columns, $[Q_{k+1,k+1}, \boldsymbol{q}_{2k+3}, \boldsymbol{q}_{2k+4}]$ must have orthonormal

columns as well. Inserting (7.39) into the extended Arnoldi decomposition (7.38) gives

$$AV_{\widetilde{m},\widetilde{m}}Q_{k+1,k+1} = V_{\widetilde{m},\widetilde{m}}Q_{k+1,k+1}A^{(2)}_{k+1,k+1} + V_{\widetilde{m},\widetilde{m}}[\boldsymbol{q}_{2k+3}, \boldsymbol{q}_{2k+4}]\widetilde{\tau}_{k+1,k+1}[\hat{\boldsymbol{e}}_{2k+1}, \hat{\boldsymbol{e}}_{2k+2}]^H.$$

Left-multiplying both sides of this equation by $V_{\widetilde{m},\widetilde{m}}V_{\widetilde{m},\widetilde{m}}^H$, the orthogonal projector onto the space $\mathcal{E}_{\widetilde{m},\widetilde{m}}(A, \boldsymbol{v}_1)$, and using

$$V_{\widetilde{m},\widetilde{m}}^H A V_{\widetilde{m},\widetilde{m}} = A_{\widetilde{m},\widetilde{m}}$$

allows to rewrite $V_{\widetilde{m},\widetilde{m}}A_{\widetilde{m},\widetilde{m}}Q_{k+1,k+1}$ as

$$V_{\widetilde{m},\widetilde{m}}Q_{k+1,k+1}A^{(2)}_{k+1,k+1} + V_{\widetilde{m},\widetilde{m}}[\boldsymbol{q}_{2k+3}, \boldsymbol{q}_{2k+4}]\widetilde{\tau}_{k+1,k+1}[\hat{\boldsymbol{e}}_{2k+1}, \hat{\boldsymbol{e}}_{2k+2}]^H.$$

Noting that $V_{\widetilde{m},\widetilde{m}}$ has full (column) rank, this implies

$$A_{\widetilde{m},\widetilde{m}}Q_{k+1,k+1} = Q_{k+1,k+1}A^{(2)}_{k+1,k+1} + [\boldsymbol{q}_{2k+3}, \boldsymbol{q}_{2k+4}]\widetilde{\tau}_{k+1,k+1}[\hat{\boldsymbol{e}}_{2k+1}, \hat{\boldsymbol{e}}_{2k+2}]^H. \quad (7.40)$$

Repeating the same line of argument starting from the relation (7.24), i.e.,

$$A^{-1}\widetilde{V}_{k+1,k+1} = \widetilde{V}_{k+1,k+1}B^{(2)}_{k+1,k+1} + [\widetilde{\boldsymbol{v}}_{2k+3}, \widetilde{\boldsymbol{v}}_{2k+4}]\widetilde{\sigma}_{k+1,k+1}[\hat{\boldsymbol{e}}_{2k+1}, \hat{\boldsymbol{e}}_{2k+2}]^H.$$

for the inverse projection matrix corresponding to $\mathcal{E}_{k+1,k+1}(A, \boldsymbol{v}_{2m+1})$ shows that we additionally have

$$B_{\widetilde{m},\widetilde{m}}Q_{k+1,k+1} = Q_{k+1,k+1}B^{(2)}_{k+1,k+1} + [\boldsymbol{q}_{2k+3}, \boldsymbol{q}_{2k+4}]\widetilde{\sigma}_{k+1,k+1}[\hat{\boldsymbol{e}}_{2k+1}, \hat{\boldsymbol{e}}_{2k+2}]^H. \quad (7.41)$$

We further note that we have the following relation involving $A_{\widetilde{m},\widetilde{m}}$ and $B_{\widetilde{m},\widetilde{m}}$ (a similar statement is shown in [96]), found by left-multiplying (7.24) (with m replaced by $\widetilde{m}$) by $V_{\widetilde{m},\widetilde{m}}^H A$.

$$I = A_{\widetilde{m},\widetilde{m}}B_{\widetilde{m},\widetilde{m}} + V_{\widetilde{m},\widetilde{m}}^H A[\boldsymbol{v}_{2\widetilde{m}+1}, \boldsymbol{v}_{2\widetilde{m}+2}]\sigma_{\widetilde{m},\widetilde{m}}[\hat{\boldsymbol{e}}_{2\widetilde{m}-1}, \hat{\boldsymbol{e}}_{2\widetilde{m}}]^H$$

which can be rearranged to

$$B_{\widetilde{m},\widetilde{m}} = A_{\widetilde{m},\widetilde{m}}^{-1}(I - V_{\widetilde{m},\widetilde{m}}^H A[\boldsymbol{v}_{2\widetilde{m}+1}, \boldsymbol{v}_{2\widetilde{m}+2}]\sigma_{\widetilde{m},\widetilde{m}}[\hat{\boldsymbol{e}}_{2\widetilde{m}-1}, \hat{\boldsymbol{e}}_{2\widetilde{m}}]^H), \quad (7.42)$$

because $A_{\widetilde{m},\widetilde{m}}$ is nonsingular by assumption. Inserting (7.42) into (7.41) and discarding the last two columns now finally gives

$$A^{-1}_{m+k+1,m+k+1}Q_{k,k} = Q_{k,k}B^{(2)}_{k,k} + [\boldsymbol{q}_{2k+1}, \boldsymbol{q}_{2k+2}]\widetilde{\sigma}_{k,k}[\hat{\boldsymbol{e}}_{2k-1}, \hat{\boldsymbol{e}}_{2k}]^H, \quad (7.43)$$

where we use that, due to the nonzero structure of $Q_{k+1,k+1}$, only the last two columns of $V_{\widetilde{m},\widetilde{m}}^H A[\boldsymbol{v}_{2\widetilde{m}+1}, \boldsymbol{v}_{2\widetilde{m}+2}]\sigma_{\widetilde{m},\widetilde{m}}[\hat{\boldsymbol{e}}_{2\widetilde{m}-1}, \hat{\boldsymbol{e}}_{2\widetilde{m}}]^H Q_{k+1,k+1}$ are nonzero. The relations (7.40) (after also dropping the last two columns) and (7.43) now allow us to use Theorem 7.15 and prove the assertion by noting that $\boldsymbol{q}_1 = \hat{\boldsymbol{e}}_{2m+1}$. □

Concerning the statement of Theorem 7.16, it is instructive to give some remarks.

Remark 7.17.

(i) While Theorem 7.16 does not guarantee that we exactly retrieve the matrix $A_{k,k}^{(2)}$, but $D_k^H A_{k,k}^{(2)} D_k$ instead, this does not have an influence on estimates computed for quadratic forms, as for any function h defined on $\operatorname{spec}(A_{k,k}^{(2)})$ we have

$$\hat{\boldsymbol{e}}_1^H h(D_k^H A_{k,k}^{(2)} D_k)\hat{\boldsymbol{e}}_1 = \hat{\boldsymbol{e}}_1^H D_k^H h(A_{k,k}^{(2)}) D_k \hat{\boldsymbol{e}}_1 = \hat{\boldsymbol{e}}_1^H h(A_{k,k}^{(2)})\hat{\boldsymbol{e}}_1,$$

using the fact that $D_k \hat{\boldsymbol{e}}_1 = \hat{\boldsymbol{e}}_1$ due to $d_{1,1} = 1$.

(ii) It is easily possible to derive a result analogous to Proposition 7.13 for Theorem 7.16. The proof follows in exactly the same way, as it only relies on properties of the basis vectors $\boldsymbol{v}_i$.

(iii) In the statement of Theorem 7.16, we assumed $A_{m+k+2,m+k+2}$ to be nonsingular. Cases in which this condition is always fulfilled are when A is Hermitian positive definite, or more general, when A is positive real. In other cases, it may well happen that $A_{m+k+2,m+k+2}$ is singular. If this happens, one can instead postpone the computation of error estimates to the next step (if $A_{m+k+3,m+k+3}$ happens to be nonsingular again) or just use estimates based on Gauss quadrature, which do not require the inversion of $A_{m+k+2,m+k+2}$.

We briefly summarize the results presented in this chapter before we proceed with numerical experiments illustrating them in the next section. We showed that it is possible to perform restart recovery in extended Krylov subspace methods, with the possibility to generate either the Hessenberg matrix from a secondary Arnoldi method or the block Hessenberg matrix from a secondary extended Arnoldi method.

In the Hermitian positive definite case, we could further show that this restart recovery can again be performed with matrices of constant size, so that the computation of error estimates is possible with cost independent of the matrix size and iteration number, and that the estimates from pairs of Gauss and Gauss–Radau quadrature form upper and lower bounds for the exact error norm.

7.4 Numerical experiments

In this section, we perform experiments for two of the model problems from Section 2.6 to illustrate the quality of the error estimates for the extended Arnoldi

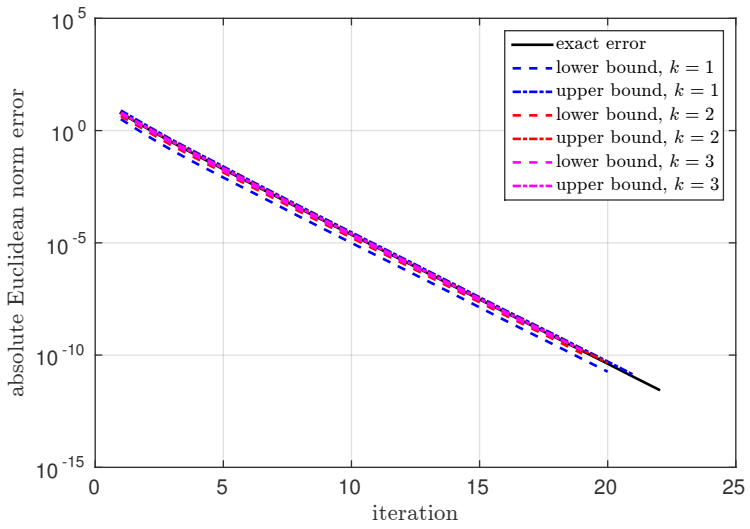


Figure 7.2: Exact error norm and error bounds computed by Gauss and Gauss–Radau quadrature rules for approximating $A^{-1/2}\boldsymbol{z}$ in the Gaussian Markov random field model problem by the extended Arnoldi method. The inner quadrature rule uses $\ell = 20$ nodes, the number of nodes in the outer quadrature rule is varied between $k = 1, 2$ and 3.

iterates obtained by (rational) Gauss quadrature as described in Sections 7.2 and 7.3. We only consider those model problems in which we need to approximate the action of a Stieltjes function of a Hermitian positive definite matrix on a vector, i.e., sampling from a Gaussian Markov random field and the Hermitian QCD model problem. Of course, all of the techniques developed in this chapter could also be applied to the other model problems (cf. also Chapter 6, where this is discussed for the polynomial Krylov case), but we refrain from doing so here, as the considered problems are sufficient for illustrating the quality of the estimates and allow to apply our theory concerning lower and upper bounds for the error.

We begin by investigating the model problem originating from sampling from a Gaussian Markov random field. As the precision matrix A of the Gaussian Markov random field can be reordered to have rather small bandwidth, the linear systems occurring in the extended Arnoldi method can efficiently be solved by Gaussian elimination after reordering. We begin by comparing the quality of the bounds obtained by these rules for the different values $k = 1, 2, 3$ of quadrature nodes. The results of this experiment are given in Figure 7.2. We again, as in the experiment presented in Section 6.6, use an inner quadrature rule with $\ell = 20$ nodes, which we found to be sufficient again (which is not completely self-evident, as the integrand in the inner quadrature has different properties here). We observe that we indeed obtain bounds for the exact error norm, and that already for $k = 3$, the computed bounds are essentially indistinguishable from

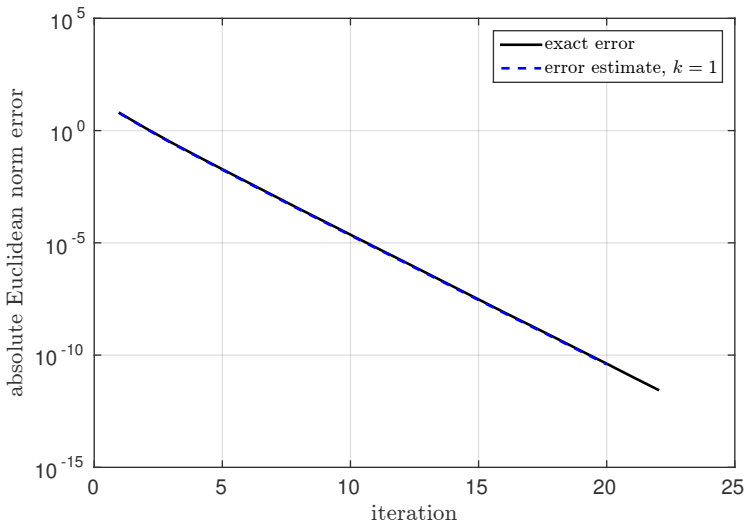


Figure 7.3: Exact error norm and error estimate computed by a rational Gauss quadrature rule for approximating $A^{-1/2}\boldsymbol{z}$ in the Gaussian Markov random field model problem by the extended Arnoldi method. The inner quadrature rule uses $\ell = 20$ nodes, while the number of nodes in the outer quadrature rule is fixed to 2 (i.e., $k = 1$).

the exact error norm to the eye. This is very important for making these error bounds usable in practical computations, as the number of iterations one needs to perform in extended Krylov methods is typically rather small, with each iteration being very costly (compared to, e.g., iterations of a polynomial Krylov method). Therefore it is even more crucial in extended Krylov methods to use a small number of quadrature nodes for computing the error bounds to avoid performing too many superfluous iterations.

Next, we investigate the error estimates computed by rational Gauss quadrature, using the extended restart recovery from Theorem 7.16 to retrieve the matrix $A^{(2)}_{k,k}$. We only present the estimate computed for $k = 1$ (keep in mind that k steps of extended Arnoldi correspond to a quadrature rule with $2k$ nodes) in Figure 7.3, as this is already very accurate, showing that rational Gauss rules can provide even better estimates than standard Gauss rules. However, both provide very good results for this rather well-conditioned model problem, so that it is hard to really judge the advantages of either one based solely on this experiment.

Therefore, we next consider the Hermitian QCD model problem, which is less well-conditioned and led to more varying quality of the bounds in the polynomial Krylov case in Section 6.6. Due to the structure of $\Gamma_5 D_W$ and $(\Gamma_5 D_W)^2$, it is difficult to solve the linear systems in the extended Arnoldi method by a direct solver, and we instead use the conjugate gradient method to approximately solve

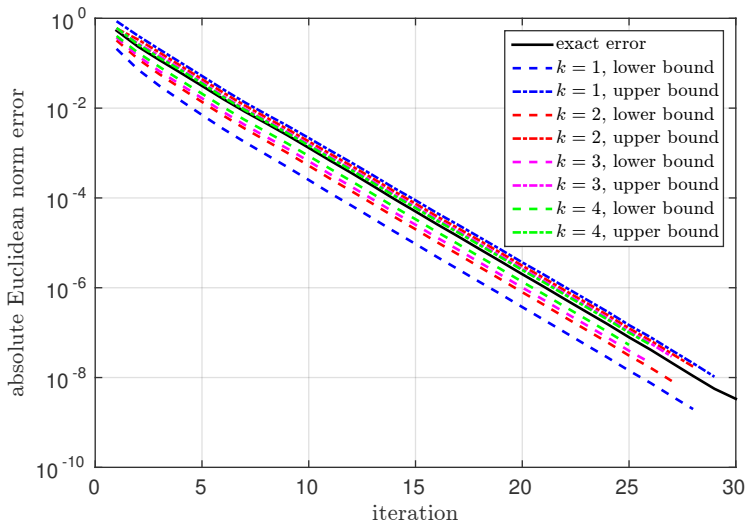


Figure 7.4: Exact error norm and error bounds computed by Gauss and Gauss–Radau quadrature rules for approximating $\left((\Gamma_5 D_W)^2\right)^{-1/2}\Gamma_5 D_W \boldsymbol{b}$ in the Hermitian QCD model problem by the extended Arnoldi method. The inner quadrature rule uses $\ell = 20$ nodes, while the number of nodes in the outer quadrature rule is varied between $k = 1, 2, 3$ and 4.

the systems. This is in a sense natural for this model problem, as for realistic lattice sizes, $\Gamma_5 D_W$ is typically not available explicitly as a matrix, but only through a routine which, given a vector $\boldsymbol{v}$, returns the result of the matrix vector product $\Gamma_5 D_W \boldsymbol{v}$. Figure 7.4 gives the results for error bounds computed via Gauss and Gauss–Radau quadrature with $k = 1, \ldots, 4$ quadrature nodes. The inner quadrature rule uses $\ell = 20$ quadrature nodes again, which this time is sufficient in comparison to the experiments in Section 6.6, cf. in particular Figure 6.5. Therefore, the integrand in the integral representation of the error function in the extended Arnoldi case seems to be easier to handle numerically than in the polynomial case, although it is more difficult to find quadrature rules which compute bounds for the integral. Like before, we again use the smallest Ritz value after a few iterations (multiplied by the safety factor 0.99) as an approximation to $\lambda_{\min}$ to be used as fixed node in the Gauss–Radau quadrature rule. We observe that also for this less well-conditioned problem, in which the bounds in the polynomial Krylov case were much worse than for the GMRF model problem, we obtain very accurate error estimates (and, as predicted by Theorem 7.11, they are indeed upper and lower bounds again) for very small numbers of outer quadrature nodes, with even the lower bound computed for $k = 1$ underestimating the exact error norm by less than one order of magnitude. The upper bounds (which are typically the more important ones, as they can be used as stopping criterion)

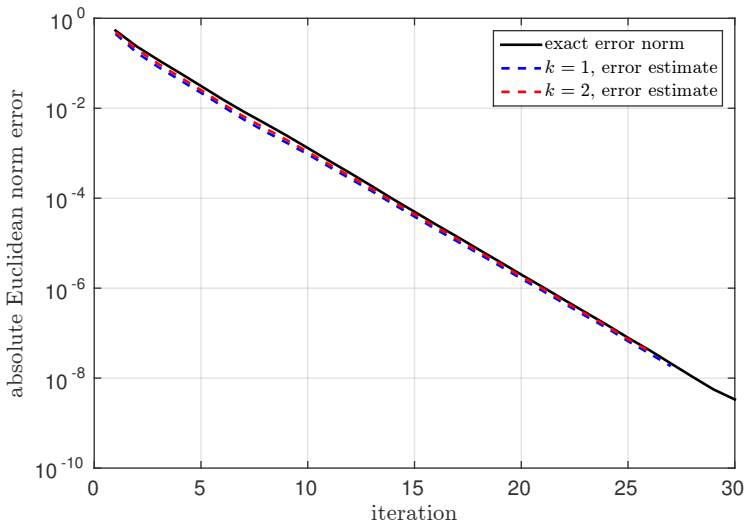


Figure 7.5: Exact error norm and error estimate computed by a rational Gauss quadrature rule for approximating $\left((\Gamma_5 D_W)^2\right)^{-1/2}\Gamma_5 D_W \boldsymbol{b}$ in the Hermitian QCD model problem by the extended Arnoldi method. The inner quadrature rule uses $\ell = 20$ nodes, while the number of nodes in the outer quadrature rule is varied between 2 and 4 (i.e., $k = 1, 2$).

lie very closely together for all numbers of quadrature nodes, so that we do not observe a real advantage in using more nodes (and thus having the bounds available later). As one can expect after observing the very high quality of the error estimates computed by standard Gauss and Gauss–Radau rules, the estimates computed by means of a rational Gauss rule are also very accurate again. We provide results for $k = 1$ and $k = 2$ (i.e., quadrature rules with 2 and 4 nodes) in Figure 7.5. While the results for $k = 2$ are slightly better than for $k = 1$, the difference is negligible, so that the value $k = 1$ is sufficient again, even for this much harder problem. While the estimates computed by rational Gauss rules are again a bit more accurate than those from standard Gauss rules, the difference is not that large, so that both methods seem equally valid for practical purposes (the standard rules having the advantage of being guaranteed bounds).

CHAPTER 8

CONCLUSIONS & OUTLOOK

In this thesis, we presented several new results arising from a new integral representation of the error in Arnoldi's method (and related methods such as the extended and harmonic Arnoldi method). This error representation allows to resolve some of the most prominent disadvantages from which Krylov subspace methods for the approximation of matrix functions typically suffer.

In particular, we presented a quadrature-based restart approach for Arnoldi's method, which allows to overcome the memory constraints that often prevent a sufficient number of iterations to be performed in the unrestarted case. The presented method is, to our knowledge, the only restarted Krylov subspace method for $f(A)\boldsymbol{b}$ proposed so far which combines numerical stability and constant computational work per cycle and at the same time acts as a black-box solver for a large class of functions.

Besides algorithmic questions concerning stability and efficient implementation, we presented a theoretical analysis of the convergence behavior of the restarted Arnoldi method for the approximation of Stieltjes matrix functions using as tools the intimate relation between Stieltjes functions and shifted linear systems, allowing to generalize convergence results for the (shifted restarted) conjugate gradient method to our setting. The main result of this analysis was that the restarted Arnoldi method converges to $f(A)\boldsymbol{b}$ for every restart length $m \geq 1$ when A is Hermitian positive definite. We also motivated that one cannot expect this result to be generalizable to larger classes of matrices. As a by-product of this analysis, we presented some results on the arbitrary convergence behavior of the restarted full orthogonalization method and the restarted GMRES method for linear systems. To overcome the limitations just mentioned, we proposed using a slight variation of Arnoldi's method, the restarted harmonic Arnoldi method (which reduces to

restarted GMRES in the linear system case) for which we could prove convergence for every restart length $m \geq 1$ for the larger class of (possibly non-Hermitian) positive real matrices.

The other main application of our integral representation for the Arnoldi error was the efficient computation of error bounds and estimates using Gauss quadrature. In particular, we showed that it is possible to compute guaranteed error bounds arising from nested quadrature rules when f is a Stieltjes function and A is Hermitian positive definite. These bounds can, e.g., be used as stopping criterion in Arnoldi's method. By making use of the so-called Lanczos restart recovery, we demonstrated that the construction of these bounds can be incorporated into the Lanczos method with computational cost independent of the dimension n of the matrix A and the iteration number m, such that they are available essentially for free in cases where n is large. We also briefly sketched that it is possible to transfer the error estimation approach to the case of non-Hermitian matrices and/or functions other than Stieltjes functions, although one has no guarantee that one computes bounds in these cases and the cost of computing the estimates increases proportionally to the iteration number in Arnoldi's method.

In the final chapter of this thesis, we showed how to transfer some of our results to extended Krylov subspace methods. For these methods, a similar integral representation for the error as in the polynomial Krylov case exists and it is therefore possible to transfer most of the results of this thesis to this related class of methods. As restarting is not very relevant in the extended Krylov case (as one typically only uses extended Krylov methods if they converge to the desired accuracy in a small number of iterations), we mainly focused on the computation of error estimates. We showed that it is again possible to compute lower and upper bounds for the error norm in these methods by Gauss and Gauss–Radau quadrature when A is Hermitian positive definite. We also investigated the possibility to use rational Gauss quadrature rules for error estimation, which led to very accurate results but does not allow to compute guaranteed error bounds.

Topics for future research include a more thorough and in-depth treatment of integral representations for the error for extended and especially general rational Krylov methods and the possibilities they offer. Another topic that could be covered is a convergence analysis for (restarted or unrestarted) extended or rational Krylov methods based on our error representation, which could maybe complement other analysis approaches available in the literature so far which typically rely on other tools than the ones used in this thesis.

A further topic which seems very relevant and appealing, especially from a practitioner's point of view, is the comparison of the efficiency of our restart approach to other techniques frequently used to overcome the memory limitations of Arnoldi's

method. These techniques include, e.g., the already mentioned extended and rational Krylov methods or the shifted conjugate gradient method applied to a rational approximation of f in partial fraction form (when A is Hermitian positive definite). As these methods all have (at least partially) the same goal but reach it in different ways, it is not at all clear whether one method is superior to the others in general or whether this depends on some (and which) properties of the problem at hand. A comparison of this kind should include meaningful numerical experiments as well as theoretical evidence for the observed behavior and give guidelines for deciding in which cases the presented methods should really be used in practice.

LIST OF FIGURES

LIST OF TABLES

LIST OF ALGORITHMS

LIST OF NOTATIONS

Throughout this thesis, scalars are denoted by lower-case letters, matrices are denoted by upper-case letters and vectors are denoted by lower-case, bold-face letters. In addition, the following notations are used.

$\mathbb{R}$	the field of real numbers
$\mathbb{R}^+$	the positive real axis $(0, \infty)$
$\mathbb{R}^-$	the negative real axis $(-\infty, 0)$
$\mathbb{R}_0^{+/-}$	the positive/negative real axis including 0
$\mathbb{C}$	the field of complex numbers
$\overline{\alpha}$	the complex conjugate of the scalar $\alpha \in \mathbb{C}$
$\mathbb{K}^n$	the n-dimensional Euclidean vector space over the field $\mathbb{K}$
$\mathbb{K}^{m \times n}$	the space of $m \times n$ matrices over the field $\mathbb{K}$
$\boldsymbol{v}_m$	a vector related to the mth iteration of an iterative method
$\boldsymbol{v}_m^{(k)}$	a vector related to the mth iteration of the kth cycle of a restarted iterative method
$\boldsymbol{v}(i)$	the ith entry of the vector $\boldsymbol{v}$
$\boldsymbol{v}(i:j)$	the entries $i, i+1, \ldots, j$ of the vector $\boldsymbol{v}$
$\|\boldsymbol{v}\|_2$	the Euclidean norm of the vector $\boldsymbol{v}$
$\mathbf{0}$	the vector of all zeros
$\boldsymbol{1}$	the vector of all ones
I	the identity matrix
A^H	the complex adjoint of the matrix A
A^{-1}	the inverse of the nonsingular matrix A
a_{ij}	the (i,j)th entry of the matrix A
$\mathcal{W}(A)$	the field of values of the matrix A
$\mathrm{diag}(\alpha_1, \ldots, \alpha_n)$	The diagonal matrix with diagonal entries $\alpha_1, \ldots, \alpha_n$
$\mathrm{diag}(A_1, \ldots, A_n)$	The block-diagonal matrix with diagonal blocks $A_1, \ldots, A_n$
Δ	the Laplace differential operator
$\partial\Omega$	the boundary of the domain Ω
$A \otimes B$	the Kronecker product of the matrices A and B

BIBLIOGRAPHY

[1] M. Abramowitz and I. A. Stegun, *Handbook of Mathematical Functions: With Formulas, Graphs, and Mathematical Tables*, Dover Publications, New York, 1964.

[2] M. Afanasjew, M. Eiermann, O. G. Ernst, and S. Güttel, *A generalization of the steepest descent method for matrix functions*, Electron. Trans. Numer. Anal., 28 (2008), pp. 206–222.

[3] ——, *Implementation of a restarted Krylov subspace method for the evaluation of matrix functions*, Linear Algebra Appl., 429 (2008), pp. 2293–2314.

[4] G. D. Allen, C. K. Chui, W. R. Madych, F. J. Narcowich, and P. W. Smith, *Padé approximation and Gaussian quadrature*, Bull. Aust. Math. Soc., 11 (1974), pp. 63–69.

[5] H. Alzer and C. Berg, *Some classes of completely monotonic functions*, Ann. Acad. Sci. Fenn., Math., 27 (2002), pp. 445–460.

[6] W. E. Arnoldi, *The principle of minimized iteration in the solution of the matrix eigenvalue problem*, Q. Appl. Math., 9 (1951), pp. 17–29.

[7] O. Axelsson and J. Karátson, *Reaching the superlinear convergence phase of the CG method*, J. Comput. Appl. Math., 260 (2014), pp. 244–257.

[8] G. A. Baker, *The existence and convergence of subsequences of Padé approximants*, J. Math. Anal. Appl., 43 (1973), pp. 498–528.

[9] ——, *Essentials of Padé Approximants*, Academic Press, New York, 1975.

[10] G. A. Baker and P. Graves-Morris, *Padé Approximants, 2nd edition*, Cambridge University Press, Cambridge, 1996.

[11] B. Beckermann and S. Güttel, *Superlinear convergence of the rational Arnoldi method for the approximation of matrix functions*, Numer. Math., 121 (2012), pp. 205–236.

[12] B. Beckermann and A. B. J. Kuijlaars, *Superlinear convergence of conjugate gradients*, SIAM J. Numer. Anal., 39 (2001), pp. 300–329.

[13] ——, *Superlinear CG convergence for special right-hand sides*, Electron. Trans. Numer. Anal., 14 (2002), pp. 1–19.

[14] C. Berg, *Stieltjes-Pick-Bernstein-Schoenberg and their connection to complete monotonicity*, in Positive Definite Functions. From Schoenberg to Space-Time Challenges, J. Mateu and E. Porcu, eds., Dept. of Mathematics, University Jaume I, Castellón de la Plana, Spain, 2008.

[15] C. Berg and G. Forst, *Potential Theory on Locally Compact Abelian Groups*, Springer, Berlin Heidelberg, 1975.

[16] M. Berljafa and S. Güttel, *Generalized rational Krylov decompositions with an application to rational approximation*, SIAM J. Matrix Anal. Appl., 36 (2015), pp. 894–916.

[17] S. Birk, *Deflated Shifted Block Krylov Subspace Methods for Hermitian Positive Definite Matrices*, PhD thesis, Bergische Universität Wuppertal, 2015.

[18] J. Bloch, A. Frommer, B. Lang, and T. Wettig, *An iterative method to compute the sign function of a non-Hermitian matrix and its application to the overlap Dirac operator at nonzero chemical potential*, Comput. Phys. Commun., 177 (2007), pp. 933–943.

[19] R. P. Boas, *Entire Functions*, Academic Press, New York, 1954.

[20] J. Brannick, A. Frommer, K. Kahl, B. Leder, M. Rottmann, and A. Strebel, *Multigrid preconditioning for the overlap operator in lattice QCD*, Numer. Math., (2015). to appear.

[21] C. Brezinski, *From numerical quadrature to Padé approximation*, Appl. Numer. Math., 60 (2010), pp. 1209–1220.

[22] P. Brown, *A theoretical comparison of the Arnoldi and GMRES algorithms*, SIAM J. Sci. Stat. Comput., 12 (1991), pp. 58–78.

[23] K. Burrage, N. Hale, and D. Kay, *An efficient implicit FEM scheme for fractional-in-space reaction-diffusion equations*, SIAM J. Sci. Comput., 34 (2012), pp. A2145–A2172.

[24] D. Calvetti, S. Kim, and L. Reichel, *Quadrature rules based on the Arnoldi process*, SIAM J. Matrix Anal. Appl., 26 (2005), pp. 765–781.

[25] A. J. Carpenter, A. Ruttan, and R. S. Varga, *Extended numerical computations on the "1/9" conjecture in rational approximation theory*, in Rational Approximation and Interpolation, Proceedings, Tampa, Florida 1983, Lecture Notes in Mathematics, vol. 1105, P. Graves-Morris, E. B. Saff, and R. S. Varga, eds., Springer, Berlin Heidelberg, 1984, pp. 383–411.

[26] M. Carter and B. van Brunt, *The Lebesgue–Stieltjes Integral — A Practical Introduction*, Springer, New York, 2000.

[27] W. J. Cody, G. Meinardus, and R. S. Varga, *Chebyshev rational approximation to e^{-x} in $[0, +\infty)$ and applications to heat-conduction problems*, J. Approx. Theory, 2 (1969), pp. 50–65.

[28] N. A. C. Cressie, *Statistics for Spatial Data*, Wiley Series in Probability and Mathematical Statistics. John Wiley & Sons, New York, 1993.

[29] J. Cullum, *Iterative methods for solving $Ax = b$, GMRES/FOM versus QMR/BiCG*, Adv. Comput. Math., 6 (1996), pp. 1–24.

[30] J. Cullum and A. Greenbaum, *Relations between Galerkin and norm-minimizing iterative methods for solving linear systems*, SIAM J. Matrix Anal. Appl., 17 (1996), pp. 223–247.

[31] N. Cundy, J. van den Eshof, A. Frommer, S. Krieg, Th. Lippert, and K. Schäfer, *Numerical methods for the QCD overlap operator: III. Nested iterations*, Comput. Phys. Commun., 165 (2005), pp. 221–242.

[32] P. I. Davies and N. J. Higham, *A Schur–Parlett algorithm for computing matrix functions*, SIAM J. Matrix Anal. Appl., 25 (2003), pp. 464–485.

[33] P. J. Davis and P. Rabinowitz, *Methods of Numerical Integration*, Academic Press, New York, 1975.

[34] C. de Boor, *Divided differences*, Surv. Approx. Theory, 1 (2005), pp. 46–69.

[35] P. A. M. Dirac, *The Principles of Quantum Mechanics, 4th edition*, Oxford University Press, Oxford, 1958.

[36] V. Druskin, *On monotonicity of the Lanczos approximation to the matrix exponential*, Linear Algebra Appl., 429 (2008), pp. 1679–1683.

[37] V. Druskin and L. Knizhnerman, *Two polynomial methods of calculating functions of symmetric matrices*, U.S.S.R. Comput. Math. Math. Phys., 29 (1989), pp. 112–121.

[38] V. Druskin and L. Knizhnerman, *Extended Krylov subspaces: Approximation of the matrix square root and related functions*, SIAM J. Matrix Anal. Appl., 19 (1998), pp. 755–771.

[39] J. Duintjer Tebbens and G. Meurant, *Any Ritz value behavior is possible for Arnoldi and for GMRES*, SIAM J. Matrix Anal. Appl., 33 (2012), pp. 958–978.

[40] ——, *On the admissible convergence curves for restarted GMRES*, tech. rep., Department of Computational Methods, Institute of Computer Science, Academy of Sciences of the Czech Republic, 2014.

[41] A. Dutt, L. Greengard, and V. Rokhlin, *Spectral deferred correction methods for ordinary differential equations*, BIT, 40 (2000), pp. 241–266.

[42] M. Eiermann and O. G. Ernst, *Geometric aspects of the theory of Krylov subspace methods*, Acta Numerica, 10 (2001), pp. 251–312.

[43] M. Eiermann and O. G. Ernst, *A restarted Krylov subspace method for the evaluation of matrix functions*, SIAM J. Numer. Anal., 44 (2006), pp. 2481–2504.

[44] M. Eiermann, O. G. Ernst, and S. Güttel, *Deflated restarting for matrix functions*, SIAM J. Matrix Anal. Appl., 32 (2011), pp. 621–641.

[45] M. Embree, *The tortoise and the hare restart GMRES*, SIAM Rev., 45 (2003), pp. 259–266.

[46] H. Engels, *Numerical Quadrature and Cubature*, Academic Press, London, 1980.

[47] Th. Ericsson, *Computing functions of matrices using Krylov subspace methods*, tech. rep., Chalmers University of Technology, Göteborg, Sweden, 1990.

[48] J. van den Eshof, A. Frommer, Th. Lippert, K. Schilling, and H. A. van der Vorst, *Numerical methods for the QCD overlap operator. I. Sign-function and error bounds*, Comput. Phys. Commun., 146 (2002), pp. 203–224.

[49] E. Estrada and D. J. Higham, *Network properties revealed through matrix functions*, SIAM Rev., 52 (2010), pp. 696–714.

[50] M. Freund and E. Görlich, *Polynomial approximation of an entire function and rate of growth of Taylor coefficients*, Proc. Edinb. Math. Soc., 28 (1985), pp. 341–348.

[51] R. W. Freund and M. Hochbruck, *Gauss quadrature associated with the Arnoldi process and the Lanczos algorithm*, in Linear Algebra for Large Scale and Real-Time Applications, M. S. Moonen, G. H. Golub, and B. L. R. De Moor, eds., Kluwer Academic Publishers, Dordrecht, 1993, pp. 377–380.

[52] R. W. Freund and N. M. Nachtigal, *QMR: a quasi-minimal residual method for non-Hermitian linear systems*, Numer. Math., 60 (1991), pp. 315–339.

[53] G. Frobenius, *Ueber Relationen zwischen den Näherungsbrüchen von Potenzreihen*, J. Reine Angew. Math., 90 (1881), pp. 1–17.

[54] A. Frommer, *BiCGStab(ℓ) for families of shifted linear systems*, Computing, 70 (2003), pp. 87–109.

[55] ——, *Monotone convergence of the Lanczos approximations to matrix functions of Hermitian matrices*, Electron. Trans. Numer. Anal., 35 (2009), pp. 118–128.

[56] A. Frommer and U. Glässner, *Restarted GMRES for shifted linear systems*, SIAM J. Sci. Comput., 19 (1998), pp. 15–26.

[57] A. Frommer, S. Güttel, and M. Schweitzer, *Convergence of restarted Krylov subspace methods for Stieltjes functions of matrices*, SIAM J. Matrix Anal. Appl., 35 (2014), pp. 1602–1624.

[58] A. Frommer, S. Güttel, and M. Schweitzer, *Efficient and stable Arnoldi restarts for matrix functions based on quadrature*, SIAM J. Matrix Anal. Appl., 35 (2014), pp. 661–683.

[59] A. Frommer, S. Güttel, and M. Schweitzer, *FUNM_QUAD: An implementation of a stable, quadrature-based restarted Arnoldi method for matrix functions*, tech. rep., Bergische Universität Wuppertal, 2014. available at `http://www.imacm.uni-wuppertal.de/fileadmin/imacm/preprints/2014/imacm_14_04.pdf`.

[60] A. Frommer, K. Kahl, S. Krieg, B. Leder, and M. Rottmann, *Adaptive aggregation-based domain decomposition multigrid for the lattice Wilson–Dirac operator*, SIAM J. Sci. Comput., 36 (2014), pp. A1581–A1608.

[61] A. Frommer, K. Kahl, Th. Lippert, and H. Rittich, *2-norm error bounds and estimates for Lanczos approximations to linear systems and rational matrix functions*, SIAM J. Matrix Anal. Appl., 34 (2013), pp. 1046–1065.

[62] A. Frommer and P. Maass, *Fast CG-based methods for Tikhonov–Phillips regularization*, SIAM J. Sci. Comput., 20 (1999), pp. 1831–1850.

[63] A. Frommer and M. Schweitzer, *Error bounds and estimates for Krylov subspace approximations of Stieltjes matrix functions*, tech. rep., Bergische Universität Wuppertal, 2015. available at `http://www.imacm.uni-wuppertal.de/fileadmin/imacm/preprints/2015/imacm_15_21.pdf`.

[64] A. Frommer and V. Simoncini, *Matrix functions*, in Model Order Reduction: Theory, Research Aspects and Applications, W. H. A. Schilders, H. A. van der Vorst, and J. Rommes, eds., Springer, Berlin Heidelberg, 2008, pp. 275–303.

[65] ——, *Stopping criteria for rational matrix functions of Hermitian and symmetric matrices*, SIAM J. Sci. Comput., 30 (2008), pp. 1387–1412.

[66] ——, *Error bounds for Lanczos approximations of rational functions of matrices*, in Numerical Validation in Current Hardware Architectures, A. Cuyt, W. Krämer, W. Luther, and P. Markstein, eds., Springer, Berlin Heidelberg, 2009, pp. 203–216.

[67] E. Gallopoulos and Y. Saad, *Efficient solution of parabolic equations by Krylov approximation methods*, SIAM J. Sci. Stat. Comput., 13 (1992), pp. 1236–1264.

[68] W. Gautschi, *Quadrature formulae on half-infinite intervals*, BIT, 31 (1991), pp. 438–446.

[69] ——, *Orthogonal Polynomials: Computation and Approximation*, Oxford University Press, Oxford, 2004.

[70] P. H. Ginsparg and K. G. Wilson, *A remnant of chiral symmetry on the lattice*, Phys. Rev. D, 25 (1982), pp. 2649–2657.

[71] G. H. Golub and Ch. F. van Loan, *Matrix Computations, 3rd edition*, Johns Hopkins University Press, Baltimore and London, 1996.

[72] G. H. Golub and G. Meurant, *Matrices, moments and quadrature*, in Numerical Analysis 1993, D. F. Griffiths and G. A. Watson, eds., Essex, 1994, Longman Scientific & Technical, pp. 105–156.

[73] ——, *Matrices, moments and quadrature II; How to compute the norm of the error in iterative methods*, BIT, 37 (1997), pp. 687–705.

[74] ——, *Matrices, Moments and Quadrature with Applications*, Princeton University Press, Princeton and Oxford, 2010.

[75] G. H. Golub and J. H. Welsch, *Calculation of Gauss quadrature rules*, Math. Comput., 23 (1969), pp. 221–230+s1–s10.

[76] S. Goossens and D. Roose, *Ritz and harmonic Ritz values and the convergence of FOM and GMRES*, Numer. Linear Algebra Appl., 6 (1999), pp. 281–293.

[77] A. Greenbaum, *Iterative Methods for Solving Linear Systems*, SIAM, Philadelphia, 1997.

[78] A. Greenbaum, V. Pták, and Z. Strakoš, *Any nonincreasing convergence curve is possible for GMRES*, SIAM J. Matrix Anal. Appl., 17 (1996), pp. 465–469.

[79] A. Greenbaum and Z. Strakoš, *Predicting the behavior of finite precision Lanczos and conjugate gradient computations*, SIAM J. Matrix Anal. Appl., 13 (1992), pp. 121–137.

[80] S. Güttel, *Rational Krylov Methods for Operator Functions*, PhD thesis, Fakultät für Mathematik und Informatik der Technischen Universität Bergakademie Freiberg, 2010.

[81] S. Güttel, *Rational Krylov approximation of matrix functions: Numerical methods and optimal pole selection*, GAMM-Mitt., 36 (2013), pp. 8–31.

[82] S. Güttel and L. Knizhnerman, *A black-box rational Arnoldi variant for Cauchy–Stieltjes matrix functions*, BIT, 53 (2013), pp. 595–616.

[83] P. Henrici, *Applied and Computational Complex Analysis, Vol. 2*, John Wiley & Sons, New York, 1977.

[84] M. R. Hestenes and E. Stiefel, *Methods of conjugate gradients for solving linear systems*, J. Res. Natl. Bur. Stand., 49 (1952), pp. 409–436.

[85] N. J. Higham, *Functions of Matrices: Theory and Computation*, SIAM, Philadelphia, 2008.

[86] N. J. Higham and A. H. Al-Mohy, *Computing matrix functions*, Acta Numerica, 19 (2010), pp. 159–208.

[87] M. Hochbruck and M. E. Hochstenbach, *Subspace extraction for matrix functions*, tech. rep., Case Western Reserve University, Department of Mathematics, Cleveland, 2005.

[88] M. Hochbruck and Ch. Lubich, *On Krylov subspace approximations to the matrix exponential operator*, SIAM J. Numer. Anal., 34 (1997), pp. 1911–1925.

[89] M. Hochbruck, Ch. Lubich, and H. Selhofer, *Exponential integrators for large systems of differential equations*, SIAM J. Sci. Comput., 19 (1998), pp. 1552–1574.

[90] M. Hochbruck and A. Ostermann, *Exponential integrators*, Acta Numerica, 19 (2010), pp. 209–286.

[91] R. A. Horn and Ch. R. Johnson, *Topics in Matrix Analysis*, Cambridge University Press, Cambridge, 1991.

[92] M. Ilić, I. W. Turner, and A. N. Pettitt, *Bayesian computations and efficient algorithms for computing functions of large, sparse matrices*, ANZIAM J., 45 (2004), pp. C504–C518.

[93] M. Ilić, I. W. Turner, and D. P. Simpson, *A restarted Lanczos approximation to functions of a symmetric matrix*, IMA J. Numer. Anal., 30 (2010), pp. 1044–1061.

[94] C. Jagels and L. Reichel, *The extended Krylov subspace method and orthogonal Laurent polynomials*, Linear Algebra Appl., 431 (2009), pp. 441–458.

[95] ——, *Recursion relations for the extended Krylov subspace method*, Linear Algebra Appl., 434 (2011), pp. 1716–1732.

[96] ——, *The structure of matrices in rational Gauss quadrature*, Math. Comput., 82 (2013), pp. 2035–2060.

[97] W. Joubert, *On the convergence behavior of the restarted GMRES algorithm for solving nonsymmetric linear systems*, Numer. Linear Algebra Appl., 1 (1994), pp. 427–447.

[98] L. Knizhnerman, *Calculation of functions of unsymmetric matrices using Arnoldi's method*, Zh. Vychisl. Mat. Mat. Fiz., 31 (1991), pp. 1–9.

[99] L. Knizhnerman and V. Simoncini, *A new investigation of the extended Krylov subspace method for matrix function evaluations*, Numer. Linear Algebra Appl., 17 (2010), pp. 615–638.

[100] A. R. Krommer and Ch. W. Ueberhuber, *Computational Integration*, SIAM, Philadelphia, 1998.

[101] A. B. J. Kuijlaars, *Which eigenvalues are found by the Lanczos method?*, SIAM J. Matrix Anal. Appl., 22 (2000), pp. 306–321.

[102] C. Lanczos, *An iteration method for the solution of the eigenvalue problem of linear differential and integral operators*, J. Res. Natl. Stand., 45 (1950), pp. 255–282.

[103] R. C. Li, *Sharpness in rates of convergence for CG and symmetric Lanczos methods*, tech. rep., University of Kentucky, 2005.

[104] G. Meurant, *On the residual norm in FOM and GMRES*, SIAM J. Matrix Anal. Appl., 32 (2011), pp. 394–411.

[105] G. Meurant and Z. Strakoš, *The Lanczos and conjugate gradient algorithms in finite precision arithmetic*, Acta Numerica, 15 (2006), pp. 471–542.

[106] I. Moret, *A note on the superlinear convergence of GMRES*, SIAM J. Numer. Anal., 34 (1997), pp. 513–516.

[107] I. Moret and P. Novati, *An interpolatory approximation of the matrix exponential based on Faber polynomials*, J. Comput. Appl. Math., 131 (2001), pp. 361–380.

[108] I. P. Natanson, *Theorie der Funktionen einer reellen Veränderlichen*, Akademie-Verlag, Berlin, 1975.

[109] H. Neuberger, *Exactly massless quarks on the lattice*, Phys. Lett., B, 417 (1998), pp. 141–144.

[110] H. Padé, *Sur la représentation approchée d'une fonction par des fractions rationelles*, Ann. Sci. Éc. Norm. Supér. (3), 9 (1892), pp. 3–93.

[111] C. C. Paige, B. N. Parlett, and H. A. van der Vorst, *Approximate solutions and eigenvalue bounds from Krylov subspaces*, Numer. Linear Algebra Appl., 2 (1995), pp. 115–133.

[112] A. N. Pettitt, I. S. Weir, and A. G. Hart, *A conditional autoregressive Gaussian process for irregularly spaced multivariate data with application to modelling large sets of binary data*, Stat. Comput., 12 (2002), pp. 353–367.

[113] Y. Saad, *Krylov subspace methods for solving large unsymmetric linear systems*, Math. Comput., 37 (1981), pp. 105–126.

[114] Y. Saad, *Analysis of some Krylov subspace approximations to the matrix exponential operator*, SIAM J. Numer. Anal., 29 (1992), pp. 209–228.

[115] ——, *Iterative Methods for Sparse Linear Systems, 2nd edition*, SIAM, Philadelphia, 2000.

[116] Y. Saad and M. Schultz, *GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM J. Sci. Stat. Comput., 7 (1986), pp. 856–869.

[117] K. Schäfer, *Krylov subspace methods for shifted unitary matrices and eigenvalue deflation applied to the Neuberger Operator and the matrix sign function*, PhD thesis, Bergische Universität Wuppertal, 2008.

[118] M. Schweitzer, *Erweiterte Krylow-Unterräume für Familien geshifteter Systeme*, Bachelor's thesis, Bergische Universität Wuppertal, 2009.

[119] ——, *Any cycle-convergence curve is possible for restarted FOM*, tech. rep., Bergische Universtität Wuppertal, 2014. available at `http://www.imacm.uni-wuppertal.de/fileadmin/imacm/preprints/2014/imacm_14_19.pdf`.

[120] L. F. Shampine, *Vectorized adaptive quadrature in MATLAB*, J. Comput. Appl. Math., 211 (2008), pp. 131–140.

[121] G. E. Shilov, *Elementary Real and Complex Analysis*, MIT Press, Cambridge, 1973.

[122] H. D. Simon, *Analysis of the symmetric Lanczos algorithm with reorthogonalization methods*, Linear Algebra Appl., 61 (1984), pp. 101–131.

[123] V. Simoncini, *Restarted full orthogonalization method for shifted linear systems*, BIT, 43 (2003), pp. 459–466.

[124] ——, *A new iterative method for solving large-scale Lyapunov matrix equations*, SIAM J. Sci. Comput., 29 (2007), pp. 1268–1288.

[125] ——, *Extended Krylov subspace for parameter dependent systems*, Appl. Numer. Math., 60 (2010), pp. 550–560.

[126] D. P. Simpson, *Krylov subspace methods for approximating functions of symmetric positive definite matrices with applications to applied statistics and anomalous diffusion*, PhD thesis, Queensland University of Technology, 2008.

[127] D. P. Simpson, I. W. Turner, and A. N. Pettitt, *Fast sampling from a Gaussian Markov random field using Krylov subspace approaches*, tech. rep., Queensland University of Technology, 2008.

[128] G. L. G. Sleijpen and D. R. Fokkema, *BiCGstab(l) for linear equations involving matrices with complex spectrum*, Electron. Trans. Numer. Anal., 1 (1993), pp. 11–32.

[129] G. W. Stewart, *Matrix Algorithms Volume II: Eigensystems*, SIAM, Philadelphia, 2001.

[130] Th. J. Stieltjes, *Recherches sur les fractions continues*, Toulouse Ann., 8 (1894), pp. J1–J122.

[131] J. Stoer and R. Bulirsch, *Introduction to Numerical Analysis, 3rd edition*, Springer, New York, 2002.

[132] H. Tal-Ezer, *On restart and error estimation for Krylov approximation of $w = f(A)v$*, SIAM J. Sci. Comput., 29 (2007), pp. 2426–2441.

[133] A. Talbot, *The accurate numerical inversion of Laplace transforms*, J. Inst. Math. Appl., 23 (1979), pp. 97–120.

[134] L. N. Trefethen and D. Bau, *Numerical Linear Algebra*, SIAM, Philadelphia, 2000.

[135] L. N. Trefethen, J. A. C. Weideman, and T. Schmelzer, *Talbot quadratures and rational approximations*, BIT, 46 (2006), pp. 653–670.

[136] R. S. Varga, *Geršgorin and His Circles*, Springer, Berlin, 2004.

[137] E. Vecharynski and J. Langou, *Any admissible cycle-convergence behavior is possible for restarted GMRES at its initial cycles*, Numer. Linear Algebra Appl., 18 (2011), pp. 499–511.

[138] H. A. van der Vorst, *BI-CGSTAB: A fast and smoothly converging variant of BI-CG for the solution of nonsymmetric linear systems*, SIAM J. Sci. Stat. Comput., 13 (1992), pp. 631–644.

[139] H. A. van der Vorst and C. Vuik, *The superlinear convergence behaviour of GMRES*, J. Comput. Appl. Math., 48 (1993), pp. 327–341.

[140] J. L. Walsh, *Interpolation and Approximation by Rational Functions in the Complex Domain, 5th edition*, American Mathematical Society, Providence, 1969.

[141] J. A. C. Weideman, *Optimizing Talbot's contours for the inversion of the Laplace transform*, SIAM J. Numer. Anal., 44 (2006), pp. 2342–2362.

[142] J. A. C. Weideman and L. N. Trefethen, *Parabolic and hyperbolic contours for computing the Bromwich integral*, Math. Comput., 76 (2007), pp. 1341–1356.

[143] H. S. Wilf, *Mathematics for the Physical Sciences*, John Wiley & Sons, New York, London, Sydney, 1962.

[144] K. G. Wilson, *Quarks and strings on a lattice*, in New Phenomena in Subnuclear Physics. Part A., A. Zichichi, ed., Plenum Press, New York, 1977, pp. 69–125.

[145] G. Zolotarev, *Application of elliptic functions to the problem of functions which vary the least or the most from zero*, Abh. St. Petersb., 30 (1877), pp. 1–59.