Inexact Adaptive Finite Element Methods for Elliptic PDE Eigenvalue Problems

vorgelegt von
Magister Ingenieurin Agnieszka Międlar
aus Wrocław, Polen

Von der Fakultät II - Mathematik und Naturwissenschaften
der Technischen Universität Berlin
zur Erlangung des akademischen Grades

Doktor der Naturwissenschaften
– Dr. rer. nat. –

genehmigte Dissertation

Promotionsausschuss:

Vorsitzender:	Prof. Dr. Martin Skutella
Gutachter:	Prof. Dr. Volker Mehrmann
Gutachter:	Prof. Dr. Carsten Carstensen (Humboldt-Universität zu Berlin)
zusätzliche Gutachter:	Prof. Ing. Zdeněk Strakoš, DrSc. (Charles University of Prague)

Tag der wissenschaftlichen Aussprache: 18.03.2011

Berlin 2011
D 83

Contents

Acknowledgements

When eating a fruit, think of the person who planted the tree.

Vietnamese Proverb

Someone said that you need passion in your heart and oil in your head to write a Ph.D. thesis. After this three years I realized these are the things which put you on the track, but there is something else what keeps you on going and prevent from derailing on many turns on the way. These are people around you, those who are there every single day, who believe in you, often much stronger than you own do.

I would like to thank my advisor Prof. V. Mehrmann for taking a risk three years ago and accepting me under his wings. For long hour discussions, answering questions even at least appropriate moments and for showing that research can be a fascinating way of life. Moreover, for continued involvement in making this group so special and supporting us not only as great teacher but as an authority.

A very special thanks to my co-authors Prof. C. Carstensen who agreed to be my BMS mentor and J. Gedicke for our long debates, for introducing an adaptive world to me and providing the openFFW [29] finite element framework for the numerical experiments. I would like to thank Dr. M. Arioli, Prof. S. Friedland, Prof. J. Liesen, Prof. L. Grubišić, Prof. U. Hetmaniuk, Prof. B. Parlett, Prof. R. Rannacher, Prof. R. Schneider, Prof. Z. Strakoš, Prof. L. N. Treffethen, for inspiring discussions about mathematics and life. Special thanks to my former advisor Prof. K. Ziętak who opened the world of the numerical methods in front of me.

Many thanks to my office-mates Lisa and Ann-Kristin, all present and previous colleagues and my friends in Berlin and Wrocław for help, kind words which made this time nice, preparing a cup of tee in the morning or simply for being there.

I appreciate the Berlin Mathematical School and the DFG Research Center Matheon for the financial support and the BMS One-Stop-Office team for administrative assistance.

Last but not least I would like to thank my parents for their love, constant support, long-hours Skype calls and faith in me. You thought me that in life everything is possible. You mean so much to me.

Without You all, I would never be where I am now.
Thank you.

Abstract

Since decades modern technological applications lead to challenging PDE eigenvalue problems, e.g., vibrations of structures, modeling of photonic gap materials, analysis of the hydrodynamic stability, or calculations of energy levels in quantum mechanics [6, 58, 78, 95]. Recently, a lot of research is devoted to the so-called Adaptive Finite Element Methods (AFEM) [10]. In most AFEM approaches it is assumed that the resulting finite dimensional algebraic problem (linear system or eigenvalue problem) is solved exactly and computational costs for this part of the method as well as the fact that they are solved in finite precision arithmetic are typically ignored.
The goal of this work is to analyze the influence of the accuracy of the algebraic approximation on the adaptivity process. Efficient and reliable adaptive algorithms should take into consideration not only discretization errors, but also iteration errors and especially for non-symmetric problems the conditioning of the eigenvalues.
Our new AFEMLA algorithm extends the standard AFEM approaches to incorporate approximation errors into the adaptation process. Furthermore, we show that the adaptive mesh refinement may be steered by the discrete residual vector, e.g., when the problem is stated in a discrete formulation where only the underlying matrices and meshes are available. Moreover, we discuss how to reduce the computational effort of the iterative solver by adapting the size of the Krylov subspace. With classical perturbation results we prove upper bounds for the eigenvalue and the eigenfunction error. Under certain assumptions similar results are obtained for convection-diffusion problems. Following [9], we introduce functional perturbation results for PDE eigenvalue problems including the functional backward error and the functional condition number. These results are used to establish a combined a posteriori error estimator embodying the discretization and the approximation error. Based on perturbation results in the $H^1(\Omega)$- and $H^{-1}(\Omega)$-norm derived in [65] and a standard residual a posteriori error estimator a balanced AFEM algorithm is proposed. The eigensolver stopping criterion is based on the equilibrating strategy, i.e., iterations proceed as long as the discrete part of the error estimator dominates the continuous part. A completely new approach combining the adaptive finite element method with the homotopy method is introduced to determine the particular eigenvalue of the convection-diffusion problem. The adaptive homotopy approach derived here emphasizes the need of the multi-way adaptation based on three different errors, the homotopy, the discretization and the iteration error. All our statements are illustrated with several numerical examples.

Zusammenfassung

Seit Jahrzehnten führen technische Anwendungen, wie z.B. Strukturschwingungen, die Modellierung von photonische Bandlücke Materialien, Analyse von hydrodynamischer Stabilität oder die Berechnung von Energieleveln in der Quantenmechanik [6, 58, 78, 95], auf PDE Eigenwertprobleme. Zur Zeit konzentriert sich die Forschung auf sogenannte Adaptive Finite Elemente Methoden (AFEM) [10]. In den meisten AFEM Ansätzen wird angenommen, dass das resultierende endlichdimensionale algebraische Problem (lineares Gleichungssystem oder Eigenwertproblem) exakt gelöst wird und der Berechnungsaufwand, sowie die Tatsache, dass die Lösungen nur in endlicher Genauigkeit vorliegen, wird vernachlässigt.
Ziel dieser Arbeit ist es, den Einfluss der Genauigkeit der algebraischen Approximation auf den adaptiven Prozess zu analysieren. Effiziente und verlässliche adaptive Algorithmen sollen betrachtet werden, d.h. nicht nur die Diskretisierungsfehler sondern auch die Iterationsfehler und insbesondere die Kondition der Eigenwerte für unsymmetrische Probleme müssen berücksichtigt werden.
Unser neuer AFEMLA Algorithmus erweitert die üblichen AFEM Ansätze durch Berücksichtigung des Approximationsfehlers im adaptiven Prozess. Desweiteren zeigen wir, dass die adaptive Gitterverfeinerung durch den diskreten Residuenvektor gesteuert werden kann, z.B. wenn das Problem in diskreter Form gegeben ist und nur die zugrundeliegenden Matrizen und Gitter verfügbar sind. Wir zeigen, wie der Berechnungsaufwand des iterativen Lösers durch Anpassung der Dimension des Krylov-Unterraums reduziert werden kann. Mit Hilfe von klassischen Störungsresultaten beweisen wir obere Schranken für den Fehler in den Eigenwerten und Eigenfunktionen. Ähnliche Resultate werden für Konvektions-Diffusions Probleme angegeben. Wie in [9] betrachten wir funktionale Störungsresultate für PDE Eigenwertprobleme, d.h. funktionale Rückwärtsfehler und funktionale Konditionzahl. Diese Resultate werden verwendet um einen gemeinsamen a posteriori Fehlerschätzer zu entwickeln, der Diskretisierungs- und Approximationsfehler berücksichtigt. Basierend auf Störungsresultaten in der $H^1(\Omega)$- und $H^{-1}(\Omega)$-Norm aus [65] und den residuenbasierten a posteriori Fehlerschätzern wurde ein balancierter AFEM Algorithmus entwickelt. Das Abbruchkriterium des Eigenwertlösers basiert auf Gleichgewichtsstrategien, d.h. die Iteration erfolgt so lange wie der diskrete Anteil des Fehlerschätzers den kontinuierlichen Anteil dominiert. Ein neuer Ansatz, der die Adaptive Finite Elemente Methode mit Homotopie verbindet, wird vorgestellt, um den Eigenwert des Konvektions-Diffusions Problems zu berechnen. Die entwickelte adaptive Homotopie hebt die Notwendigkeit von mehrfacher Adaptivität hervor, d.h. basierend auf den Homotopie-, Diskretisierungs- und Iterationsfehlern. Alle unsere Ergebnisse werden mit verschiedenen numerischen Beispielen illustriert.

Chapter 1

Introduction

Nothing is as important to the success of a finite element analysis as the accuracy of the elements. Indeed, in a linear static analysis, the finite elements embody all of the discretizing assumptions, the rest of the calculations are exact except for the lack of precision.

A proposed standard set of problems to test finite element accuracy,
March 1984
R. H. McNeal and R. L. Harder

Several times in our life we were asked *"What is an eigenvalue?"*. Well, since it is not difficult to answer this question to a person which have some basic knowledge about the linear algebra, however, preparing a clear answer for average Joe, Kowalski or Schmidt is a challenge. *"We live in an eigen world"* this is how Bob Broughton [1] answers a question *"Why Solve the Eigen Problem?"* and this is probably the simplest way to introduce this problem to the world.

Vibrations of structures, computation of acoustic fields in cars or trains, analysis of automobile brakes, resonance problems or nuclear reactor criticality, modeling of photonic gap materials, analysis of the hydrodynamic stability, or calculations of energy levels in quantum mechanics [6, 58, 78, 95]. These are all very important practical applications and they all influence our life more than we imagine. When we live in a seismically active area we would appreciate if the natural frequencies of our building lie outside the earthquake band, driving a car we expect that the interior noises will be as low as possible or passing the London millennium footbridge next time we do not want it to wobble or collapse.

These every day life problems are included in this abstract terminology which we often do not know and understand, but definitely we do need. All those phenomena involve the solution of the eigenvalue problems for partial differential equations (PDEs).

[1] B. Broughton, *"Why Solve the Eigen Problem?"*
`http://www.math.canterbury.ac.nz/php/resources/podcast/eigen-problem/`

In this work we investigate a second-order elliptic partial differential eigenvalue problems of a general form

$$\mathcal{L}(u) = \lambda u,$$

where $\mathcal{L}$ is a second-order elliptic partial differential operator, λ is an eigenvalue and u the corresponding eigenfunction.

It is well understood that numerical methods for PDEs, such as the Finite Element Method (FEM) with very fine meshes, give good approximations of the solutions, but they typically lead to a very high computational effort. Recently, a lot of research is devoted to the so-called Adaptive Finite Element Methods (AFEM) [10] which, based on the quality of the numerical approximation (a posteriori error estimator), automatically adjust the finite element space to reduce the computational complexity, while retaining the overall accuracy. An adaptation of the mesh requires to determine the regions where the solution deviates from a regular behavior and concentrating grid points in these regions. To do this, a priori and a posteriori error estimates for the error between the exact solution and the computational solution are determined and used to control the mesh refinement [2, 111].

First a priori error estimates for eigenvalues and eigenvectors were developed for elliptic operators in [107]. Further improvements were established for self-adjoint operators in [37, 72, 98] and for compact operators in [11, 12]. A two-grid adaptation approach was suggested in [115]. All these approaches, although optimal, contain mesh size restrictions, which cannot be verified or quantified, neither a priori nor a posteriori.
First truly a priori error estimates for symmetric eigenvalue problems, based on angles between subspaces, were presented in [77]. It was shown, both for single and multiple eigenvalues, that the eigenvalue error depends on the approximability of the eigenfunctions in the corresponding invariant subspace. Other works [7, 76] take advantage of this technique to obtain a priori Rayleigh-Ritz majorization error bounds and apply them in the context of the finite element method. Further results on a priori error estimates can be found in [80, 98].
Since a priori error estimates yield information about theoretical properties such as asymptotic convergence rates or stability, one needs some fully computable lower and upper error bounds to control an adaptive mesh refinement procedure. On the other hand a posteriori error estimators, based on the numerical solution and initial data, control the whole adaptive process by indicating the error distribution.

A first approach on a posteriori error analysis for symmetric second-order elliptic eigenvalue problems can be found in [111, §3.4]. A combination of a posteriori and a priori analysis was used in [79] to prove a posteriori estimates of optimal order. Under the assumption that the mesh is fine enough, to guarantee that the computed eigenvalue is close to the exact, and that the eigenfunction possess an appropriate regularity it was proved that for smooth eigenvectors the error in eigenvalues and eigenvectors is bounded in terms of the mesh size, a stability factor, and the residual. In [50] it is shown that the edge residual (i.e., the residual on the edges of the mesh) is an upper bound for the volumetric part of the residual and an explicit residual-based estimator is constructed and proved to be equivalent to the error

up to higher order terms. In [88] similar results are achieved by applying a local averaging technique. Based on the analysis of the residual equations, in [92] an a posteriori error estimator is presented that works on a subspace of eigenvector approximations obtained by the preconditioned inverse iteration. Recently in [31] the results of [50, 88] were improved by showing that, for all eigenvalues, refinement is possible without the volume contribution in the estimator and that the higher order terms can in fact be neglected.

An approach for non-symmetric elliptic eigenvalue problems was presented in [67]. Using the general optimal control framework of Galerkin approximations of nonlinear variational equations of [18] a residual-based estimator with explicitly given remainder terms was presented. Unfortunately, also this result, as it needs the knowledge of the exact solution of the adjoint problem and provides only upper bounds of the error, is not a true a posteriori error estimate. Surveys about a posteriori error estimation can be found in [2, 111].

In [59] the convergence of an adaptive linear finite element method for computing eigenpairs is proved with a refinement procedure that considers both a standard a posteriori error estimator and eigenfunction oscillations. The global convergence result of [31] requires no additional mesh size assumptions and inner node properties. Additionally, the presented AFEM method with an averaging scheme has optimal empirical convergence rate. In [43] based on the relation between the finite element eigenvalue approximation and the associated boundary value approximation the uniform convergence and optimal complexity of the adaptive finite element eigenvalue approximation are proved. Nearly at the same time, in [56], the convergence of AFEM for any reasonable marking strategy and any initial mesh was shown. Complexity estimates for adaptive eigenvalue computations were obtained in [42] and further eigenvalue/eigenvector estimates are obtained in [62, 74, 75]. The analysis of the FEM for elliptic eigenvalue problems is also studied in [102, 115].

From a scientific computing point of view, a posteriori error estimates and convergence results give some theoretical background for the actual computational schemes (algorithms) which are of particular interest, e.g., the preconditioned eigensolvers PINVIT [92], LOBPCG [73] or a two-grid discretization scheme [115], to mention only few.

1.1 Motivation

A natural question is whether there is still something to be done in the direction of adaptive finite element methods for eigenvalue problems. Figure **??** shows that numerical models of the real-world problems from the very beginning are corrupted with different types of errors. Although, we can not avoid them completely, we should at least try to reduce their influence on the numerical simulations, since solving inaccurate problems up to high accuracy leads to **high computational effort**.
In most AFEM approaches it is assumed that the resulting finite dimensional algebraic problem (linear system or eigenvalue problem) is **solved exactly** and the computational costs for this part of the method as well as the fact that these problems are solved in finite

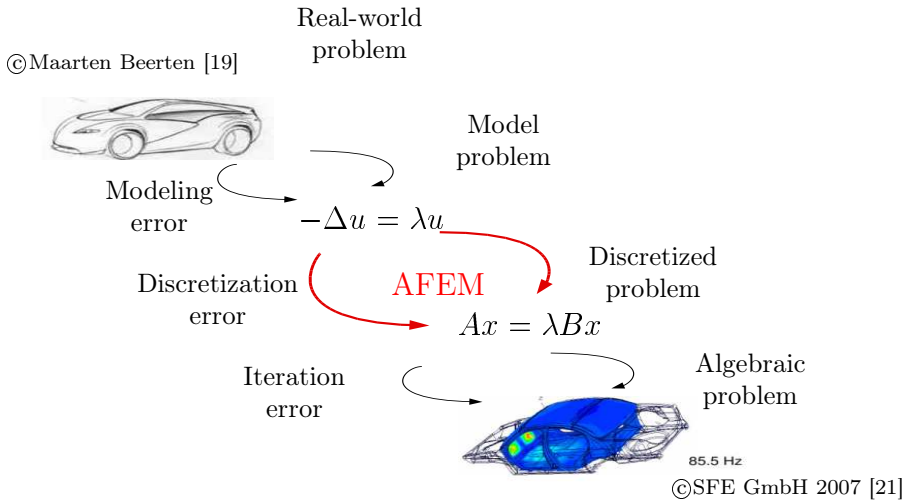


Figure 1.1: A real-world problem versus a numerical model. Car picture [19]. Acoustic field picture [21].

precision arithmetic are typically ignored. Both a priori and a posteriori error estimates derived for PDE eigenvalue problems **do not consider iteration errors** (errors connected to solving algebraic eigenvalue problems using iterative methods). Iterative solvers used for solving corresponding algebraic eigenvalue problems, i.e., Krylov space methods, are used as a **'black box'** often wasting computational time and resources. In particular, the contribution of the complexity of the algebraic solvers into the total AFEM complexity is ignored and even not analyzed. Besides, **using the solution** of the algebraic eigenvalue problem **only to determine where the grid should be refined**, is a complete waste of computational work.

This is acceptable if the costs for the algebraic problems are small and the problems are well-conditioned, such that reaching full accuracy is possible. For real-world problems adaptive finite element methods, in itself, lead to **problems of large dimension**, and therefore costs of solving the corresponding algebraic eigenvalue problem start to significantly dominate over the total costs. Of course ignoring that fact and working further with well-established methods is possible, but in an era of intense competition, waiting for the computational results even few days may be a serious drawback. Also storage demands may be an issue. An upcoming century of multi-linear algebra requires designing new algorithms which exploit techniques like model reduction or adaptive methods instead of counting on the endless computational power limits. Moreover, there is a second, more significant danger. Ignoring computational error may lead to **incorrect results** [106].

1.2 Hope for changes

Although, the need of designing the adaptive finite element methods concerning the iterative errors was already noticed in [114] not much attention was dedicated to this issue. In the course of our work several new results were established.
In [68] fully computable a posteriori error estimates which take into account an inexact solution of the linear algebraic system are derived together with efficient stopping criteria for iterative solvers in the context of a second-order elliptic pure diffusion model problem. Recently the convergence of inexact adaptive finite element boundary value solvers for elliptic self-adjoint problems was proved in [8]. Moreover, a practical stopping criterion for the conjugate gradient method was proposed. Most of the existing convergence results for AFEM both for boundary and eigenvalue problems are based on perturbation arguments assuming that the a posteriori error estimator evaluated at an approximate solution does not differ much from the estimator evaluated at the exact solution for which the convergence was proved. An AFEM algorithm without Galerkin orthogonality is introduced in [64]. The residual type estimator is constructed directly with the inexact solution of the boundary problem and controls an algebraic error in terms of the BPX precoditioner [25].
For eigenvalue problems a combined adaptive finite element method with an iterative algebraic eigenvalue of quasi-optimal computational complexity (AFEMES) was introduced in [32]. In [116] the convergence and a quasi-optimality of the inexact inverse iteration coupled with adaptive finite element methods exploiting the well-known results from boundary value problems are studied.

1.3 Content of this work

Adaptive methods and error estimates for boundary value problems associated with PDEs have been studied in great detail. Research for eigenvalue problems have increased during the last few years, however they are not well established yet. In general the non-linear nature of eigenvalue problems does not allow to simply extend results from the boundary value problems. Most of the results are still concentrated on deriving new a posteriori error estimators or improving convergence results. The goal of this work is to analyze the influence of the accuracy of the algebraic approximation on the adaptivity process. Efficient and reliable adaptive algorithms should take into consideration not only the discretization errors, but also iteration errors and especially for the non-symmetric problems the conditioning of the eigenvalues. In this work we discuss the main difficulties which one has to take into consideration while working with PDE eigenvalue problems. We explain what criteria have to be fulfilled to construct PDE eigenvalue solvers satisfying real life expectations in the context of efficiency, optimal complexity, overall accuracy and parameter variations. For the class of elliptic self-adjoint eigenvalue problems we introduce an appropriate error estimator combining adaptive mesh refinement with an adaptation of the iterative eigensolvers. Moreover, we discuss how to reduce the computational effort of the iterative solver by adapting the size of the Krylov subspace. One of the advantages of this method is their direct application

to problems stated directly in a discrete form. The adaptive homotopy approach derived here emphasizes the need of the multi-way adaptation taking into consideration changing homotopy or optimization parameters.
The work is organized as follows: In Chapter 2 we introduce the theoretical background of this work. Section 2.1 reviews some basic facts from functional analysis and (numerical) linear algebra. In Section 2.2 we introduce an elliptic second-order PDE eigenvalue problem which is a main object of our interest throughout this work. The Galerkin method as well as the Finite Element Method (FEM) for approximating the solution of PDE eigenvalue problem are presented in Section 2.2.2. Some state of the art results in a priori and a posteriori error analysis are introduced in Section 2.2.3. The Adaptive Finite Element Method (AFEM) together with several refinement and marking strategies is described in Section 2.2.4. After that, in Section 2.3, we consider the Generalized Algebraic Eigenvalue Problem, we introduce the iterative methods for large, sparse eigenvalue problems, namely the Arnoldi/Lanczos method and briefly discuss homotopy methods. We conclude this section by recalling perturbation results for algebraic eigenvalue problems and discuss relations between continuous and discrete inner products and norms on Sobolev spaces.

All new ideas introduced in subsequent sections will be demonstrated on two model problems introduced in Chapter 3. The Laplace and the convection-diffusion eigenvalue problem are chosen to illustrate the properties and difficulties of the PDE eigenvalue problems.

Chapter 4 is dedicated to self-adjoint eigenvalue problems. At first, in Section 4.1, we show how dangerous it may be relying on the assumed exactness of the approximate solution. We introduce the adaptive finite element algorithm called AFEMLA which incorporates the information obtained during the iterative process of solving the algebraic eigenvalue problems into the error estimation and the adaptive mesh refinement in Section 4.2. Furthermore, under the saturation assumption, we determine error estimates for the eigenvalues and the corresponding eigenfunctions. We conclude this section with several numerical experiments including domains with singularities as well as the subspace version of our algorithm where we estimate few eigenvalues simultaneously. The results presented in this section are published in [91]. The notion of the functional backward error and condition number introduced in [9] are generalized in Section 4.3 where we derive functional perturbation results for PDE eigenvalue problems in their variational formulation. A combined a posteriori error estimator which measures directly the size of the iteration error is introduced in Section 4.4. Based on the perturbation results in $H^1(\Omega)$- and $H^{-1}(\Omega)$-norm derived in [65] and a standard residual a posteriori error estimator the AFEM algorithm which balances both sources of errors is proposed.

Non-self-adjoint problems are studied in Chapter 5. In Section 5.1 an extension of the AFEMLA algorithm for the class of the convection-diffusion problems is introduced. Under the assumption that the convection coefficient is small enough, such that the operator is close to normal, the bounds for the error in the simple eigenvalue and the corresponding eigenfunction are obtained. Several numerical experiments are presented to highlight the influence of the non-self-adjoint component on the approximations. A completely new approach combining the adaptive finite element method with the homotopy methods is a

subject of Section 5.2. We introduce three different algorithms to analyze the influence of the homotopy, the discretization and the iteration error on the performance of the AFEM algorithm. Although, the numerical experiments are dedicated only to approximating the eigenvalues with the smallest real part of the convection-diffusion operator, the generalization for more complicated problems seems to be possible. The results presented in this section are a joint work with C. Carstensen, J. Gedicke and V. Mehrmann and are published in [33]. We finish this work with Chapter 6 where, except of general conclusions, we discuss some new research ideas and future goals.

Chapter 2

Preliminaries

In order to make this work self-consistent, in this chapter we present a short outline of the mathematical framework needed in the subsequent chapters. First, we summarize some results from functional analysis like partial differential operators, linear functionals, sesquilinear (bilinear) forms or Sobolev spaces. We briefly discuss some facts from matrix theory and introduce the standard algebraic eigenvalue problem. In Section 2.2 we define an elliptic second-order PDE eigenvalue problem in classical and variational formulation. We describe the Galerkin method as well as the Finite Element Method (FEM) for approximating the solution of PDE eigenvalue problem. Afterwards, we discuss the quality of approximate solutions based on *a priori* and *a posteriori* error analysis and introduce the Adaptive Finite Element Method (AFEM) which reduces the computational complexity of standard FEM, while retaining the overall accuracy. In Section 2.3 we consider Generalized Algebraic Eigenvalue Problems. We introduce iterative methods for large, sparse eigenvalue problems, namely the Arnoldi/Lanczos method. Moreover, we discuss homotopy methods. We conclude this section by recalling perturbation results for the algebraic eigenvalue problems. Last but not least, we expose some relations between continuous and discrete inner products and norms on Sobolev spaces.

2.1 Basic facts

The definitions and notation we adopt here are taken from [6, 13, 36, 46, 52, 61, 93, 95, 99, 101, 112].

2.1.1 Normed and inner product vector spaces

Definition 2.1. A linear space V equipped with a mapping $\|\cdot\|_V : V \to \mathbb{R}$ (called *norm*) such that:

1. $\|u\|_V \geq 0 \quad$ for all $u \in V$,
2. $\|u\|_V = 0 \quad \Leftrightarrow u = 0$,

3. $\|\alpha u\| = |\alpha|\|u\|$ for all $\alpha \in \mathbb{F}$ ($\mathbb{R}$ or $\mathbb{C}$) and for all $u \in V$,

4. $\|u+v\|_V \leq \|u\|_V + \|v\|_V$ for all $u, v \in V$,

is a *normed vector space*.

Definition 2.2. A sequence $u_n, n = 1, 2, \ldots$ in the normed vector space V for which

$$\lim_{m,n\to\infty} \|u_m - u_n\|_V = 0$$

is a *Cauchy sequence*.

Definition 2.3. Any normed vector space V is *complete* if every Cauchy sequence $u_n, n = 1, 2, \ldots$ in V has the limit in this space.

Definition 2.4. A complete, normed vector space is a *Banach space*.

Definition 2.5. A linear space V together with a mapping $(\cdot,\cdot)_V : V \times V \to \mathbb{F}$ ($\mathbb{R}$ or $\mathbb{C}$) (called *inner product*) such that:

1. $(u, v)_V = \overline{(v, u)_V}$ for all $u, v \in V$,

2. $(u, u)_V \geq 0$ for all $u \in V$,

3. $(u, u)_V = 0 \quad \Leftrightarrow u = 0$,

4. $(\alpha u + \beta v, w)_V = \alpha(v, w)_V + \beta(v, w)_V \quad \forall \alpha, \beta \in \mathbb{F}$ ($\mathbb{R}$ or $\mathbb{C}$) and $u, v, w \in V$,

is an *inner product space*.

Remark 2.6. An inner product space V is a normed space, where the norm is defined

$$\|u\|_V := \sqrt{(u, u)_V}.$$

Definition 2.7. A complete inner product space is a *Hilbert space*.

Remark 2.8. Every Hilbert space is a Banach space.

Definition 2.9. Let u, v be elements of the normed vector space V with inner product $(\cdot,\cdot)_V$ and norm $\|\cdot\|_V$. The acute angle $\angle(u, v)$ is a real number $0 \leq \angle(u, v) \leq \frac{\pi}{2}$ satisfying

$$\cos_V \angle(u, v) := \frac{|(u, v)_V|}{\|u\|_V \|v\|_V}.$$

Moreover, if $X, Y \subset V$. Then, the angle between subspaces X, Y, denoted by $\angle(X, Y)$, is defined as

$$\sin_V \angle(X, Y) := \sup_{x\in X} \inf_{y\in Y} \frac{\|y - x\|_V}{\|x\|_V}.$$

2.1.2 Partial differential operators

Definition 2.10. Let Ω be a bounded open set $\Omega \subset \mathbb{F}^n$ and $u : \Omega \to \mathbb{F}$. The transformation $\mathcal{L} : C^k(\Omega) \to C(\Omega)$ is a *linear partial differential operator of order* k and is given by

$$\mathcal{L}u := \sum_{|\alpha| \leq k} a_\alpha(x) \partial^\alpha u \tag{2.1}$$

with a multi-index $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_n)$, $|\alpha| = \alpha_1 + \alpha_2 + \ldots + \alpha_n$ and $\partial^\alpha = \partial_1^{\alpha_1} \partial_2^{\alpha_2} \ldots \partial_n^{\alpha_n}$, where $\partial_i^{\alpha_i} = \frac{\partial^{\alpha_i}}{\partial x_i^{\alpha_i}}$.

Given a linear partial differential operator $\mathcal{L}$ the *adjoint operator* $\mathcal{L}^\star$ is defined by

$$(\mathcal{L}u, v)_V = (u, \mathcal{L}^\star v)_V, \quad \text{for all } u, v \in V,$$

where $(\cdot, \cdot)_V$ is an inner product on V. A bounded operator $\mathcal{L}$ is *Hermitian* or *self-adjoint* if $\mathcal{L} = \mathcal{L}^\star$.
An operator $\mathcal{A}$ of the form

$$\mathcal{A}u := \sum_{i,j=1}^{n} a_{i,j}(x) \left(\frac{\partial^2 u}{\partial x_i \partial x_j}(x) \right) + \sum_{i=1} b_i(x) \frac{\partial u}{\partial x_i} + c(x)u \tag{2.2}$$

is a *linear second-order partial differential operator*. Let the coefficient matrix $A(x)$ be given as

$$A(x) = \begin{bmatrix} a_{11}(x) & \ldots & a_{1n}(x) \\ \vdots & \ddots & \vdots \\ a_{n1}(x) & \ldots & a_{nn}(x) \end{bmatrix}$$

The operator $\mathcal{A}$ is *elliptic* if all eigenvalues of $A(x)$ are positive or negative,
parabolic if $n-1$ eigenvalues are positive or negative and one eigenvalue is zero,
hyperbolic if all $n-1$ eigenvalues are positive and one is negative or vice versa,
ultrahyperbolic if there are more than one positive and one negative eigenvalue but there are no zero eigenvalues.

2.1.3 Linear functionals and sesquilinear (bilinear) forms

Let V be a linear vector space and $\mathbb{F}$ denote either $\mathbb{R}$ or $\mathbb{C}$. The linear transformation $\ell : V \to \mathbb{F}$ is a *linear functional* on V. A linear functional on a normed linear vector space V is said to be *bounded* or (*continuous*) if there exists a constant $c > 0$ such that

$$|\ell(v)| \leq c \|v\|_V, \quad \text{for all } v \in V.$$

The space of bounded linear functionals on V is called the *dual space* and denoted by V^*. Moreover, V^* is itself a normed vector space with a *dual norm* defined by

$$\|\ell(v)\|_{V^*} := \sup_{v \in V} \frac{|\ell(v)|}{\|v\|_V}, \quad v \neq 0. \tag{2.3}$$

Definition 2.11 ([112])**.** Let V, W be linear vector spaces over the field $\mathbb{F}$ ($\mathbb{R}$ or $\mathbb{C}$). Then the mapping $a : V \times W \to \mathbb{F}$ is a *sesquilinear form* (or *functional*) on $V \times W$ if, for a fixed w, $a(v, w)$ is linear in V and for a fixed v the complex conjugate $\overline{a(v, w)}$ is linear in W, i.e.,

$$a(v_1 + v_2, w_1 + w_2) = a(v_1, w_1) + a(v_1, w_2) + a(v_2, w_1) + a(v_2, w_2)$$

and

$$a(\alpha v, \beta w) = \alpha \bar{\beta} a(v, w)$$

for $v, v_1, v_2 \in V$, $w, w_1, w_2 \in W$ and $\alpha, \beta \in \mathbb{F}$.

Remark 2.12. If $\mathbb{F} = \mathbb{R}$ $a(x, y)$ is linear in V and W and therefore is a *bilinear form*.

Remark 2.13. Every bounded linear functional $\ell(u)$ can be written as $\ell(u) = a(u, v)$ for some $v \in V$.

A sesquilinear (or bilinear for $\mathbb{F} = \mathbb{R}$) form $a(u, v)$ on space V, i.e., $a : V \times V \to \mathbb{F}$, is said to be *Hermitian* (or *symmetric* for $\mathbb{F} = \mathbb{R}$) if

$$a(u, v) = \overline{a(v, u)} \quad \big(\text{ or } \ a(u, v) = a(v, u) \big) \qquad \text{for all } u, v \in V, \tag{2.4}$$

continuous or bounded on V if there exist a positive constant C such that

$$|a(u, v)| \leq C \|u\|_V \|v\|_V \qquad \text{for all } u, v \in V, \tag{2.5}$$

coercive on V or V-*elliptic* if there exist a positive constant c such that

$$\text{Re } a(u, u) \geq c \|u\|_V^2 \qquad \text{for all } u \in V. \tag{2.6}$$

We can define the norm of the bilinear $a(u, v)$ as follows

$$\|a(u, v)\|_V := \sup_{u, v \in V} \frac{|a(u, v)|}{\|u\|_V \|v\|_V}, \quad u, v \neq 0. \tag{2.7}$$

2.1.4 Sobolev spaces

Definition 2.14. Consider real-(complex-) valued function u on a given domain Ω, that are Lebesgue measurable. We denote by $\int_\Omega u(x)dx$ the Lebesgue integral of u. Let

$$\|u\|_{L_p(\Omega)} = \left(\int_\Omega |u|^p \right)^{\frac{1}{p}} \quad \text{for} \quad 1 \leq p < \infty.$$

Then the Lebesgue space $L_p(\Omega)$ is defined as

$$L_p(\Omega) := \{u : \|u\|_{L_p(\Omega)} < \infty\}.$$

Remark 2.15. $L_2(\Omega)$ is a space of real-(complex-) valued square integrable functions on Ω with inner product

$$(u, v) = (u, v)_{L_2(\Omega)} = \int_\Omega u\bar{v}dx$$

and norm

$$\|u\| = \|u\|_{L_2(\Omega)} = \left(\int_\Omega |u|^2 dx\right)^{\frac{1}{2}}.$$

Definition 2.16. Let $m \in \mathbb{N}$ and $p \in [1, \infty)$, then the *Sobolev space* $\mathcal{W}^{m,p}(\Omega)$, its inner product, induced norm and seminorm are given by

$$\mathcal{W}^{m,p}(\Omega) := \{u \in L_p(\Omega) : \partial^\alpha u \in L_p(\Omega) \text{ for all } |\alpha| \leq m\},$$

$$\|u\|_{m,p} := \Big\{ \sum_{|\alpha|\leq m} \|\partial^\alpha u\|^p_{L_p(\Omega)} \Big\}^{\frac{1}{p}},$$

$$|u|_{m,p} := \Big\{ \sum_{|\alpha|= m} \|\partial^\alpha u\|^p_{L_p(\Omega)} \Big\}^{\frac{1}{p}},$$

respectively.

Remark 2.17. The Sobolev space based on $L_2(\Omega)$ is denoted by $H^m(\Omega)$ instead of $\mathcal{W}^{m,2}(\Omega)$.

$$H^m(\Omega) := \{u \in L_2(\Omega) : \partial^\alpha u \in L_2(\Omega) \text{ for all } |\alpha| \leq m\},$$

with corresponding norm

$$\|u\|_m := \|u\|_{H^m(\Omega)} = \Big\{ \sum_{|\alpha|\leq m} \|\partial^\alpha u\|^2_{L_2(\Omega)} \Big\}^{\frac{1}{2}}$$

and seminorm

$$|u|_m := |u|_{H^m(\Omega)} = \Big\{ \sum_{|\alpha|= m} \|\partial^\alpha u\|^2_{L_2(\Omega)} \Big\}^{\frac{1}{2}}.$$

Example 2.18. For $m = 0, H^m(\Omega) = H^0(\Omega) = L_2(\Omega)$. Following Remark 2.17 the corresponding norm

$$\|u\|_{H^0(\Omega)} = \|u\|_{L_2(\Omega)},$$

and seminorm

$$|u|_0 = \|u\|_{H^0(\Omega)}$$

are defined. For the case of $m = 1, H^m(\Omega) = H^1(\Omega)$ we have corresponding norm and seminorm definitions

$$\|u\|^2_{H^1(\Omega)} = \|u\|^2_{L_2(\Omega)} + \|\nabla u\|^2_{L_2(\Omega)},$$
$$|u|^2_{H^1(\Omega)} = \|\nabla u\|^2_{L_2(\Omega)},$$

respectively.

Definition 2.19. We denote by $\mathcal{W}_0^{m,p}(\Omega)$ the closure of $C_0^\infty(\Omega)$ with respect to the Sobolev norm $\|\cdot\|_{m,p}$.

Remark 2.20. We denote by $H_0^m(\Omega)$ the closure of $C_0^\infty(\Omega)$ with respect to the Sobolev norm $\|\cdot\|_m$. Therefore, $H_0^m(\Omega)$ is the space of functions in $H^m(\Omega)$ that vanish at the boundary, in the sense of traces, i.e., if $u \in H^m(\Omega)$ belongs to $H_0^m(\Omega)$, then $\partial^\alpha u = 0$ on $\partial\Omega, |\alpha| \leq m-1$ [99].

Remark 2.21. The dual space of $H_0^m(\Omega)$ is often denoted by $H^{-m}(\Omega)$ or H_0^{-m}:

$$H^{-m}(\Omega) = H_0^{-m}(\Omega) := (H_0^m)^*.$$

The corresponding norm and seminorm are defined as

$$\|u\|_{H^{-m}(\Omega)} := \sup_{\substack{v \in H_0^m(\Omega), \\ v \neq 0}} \frac{|(u,v)_{L_2(\Omega)}|}{\|v\|_m} \quad \text{for} \quad m \geq 0,$$

$$|u|_{H^{-m}(\Omega)} = \sup_{\substack{v \in H_0^m(\Omega), \\ v \neq 0}} \frac{|(u,v)_{L_2(\Omega)}|}{|v|_m} \quad \text{for} \quad m \geq 0,$$

respectively.

2.1.5 Matrix theory and eigenvalue problems

A matrix $A \in \mathbb{F}^{n\times n}$ is
Hermitian (or *symmetric* if $\mathbb{F} = \mathbb{R}$) if and only if

$$A^* = A \qquad (\text{ or } A^T = A),$$

where A^* is the conjugate transpose of A, i.e., $A^* = (\overline{A})^{\mathrm{T}} = \overline{A^{\mathrm{T}}}$,
normal if and only if

$$A^*A = AA^*,$$

positive definite if and only if

$$A \text{ is Hermitian (or symmetric) and } x^*Ax > 0 \text{ for all } x \in \mathbb{F}^n, x \neq 0.$$

The *standard eigenvalue problem* states as follow.
Given $A \in \mathbb{F}^{n\times n}$. Find scalars $\lambda \in \mathbb{F}$ and non-zero vectors $x \in \mathbb{F}^n$ such that

$$Ax = \lambda x.$$

λ is an *eigenvalue* of A and x is an *eigenvector* corresponding to λ. The eigenvalues of a Hermitian (symmetric) matrix are real and the eigenvectors corresponding to distinct eigenvalues are orthogonal. Moreover, if matrix is positive definite all its eigenvalues are

positive. In general a non-symmetric real matrix A may have complex eigenvalues which appear as complex conjugate pairs and a non-orthogonal set of eigenvectors. An eigenvalue λ is *simple* if its algebraic multiplicity is equal to one. The set of all eigenvalues of A is a *spectrum of* A denoted by $sp(A)$.

A non-zero vector $y \in \mathbb{F}^n$ such that

$$y^*A = \lambda y^*,$$

is a *left eigenvector* corresponding to eigenvalue λ, where y^* denotes the complex conjugate transpose of y.

Definition 2.22. The *eigenvalue condition number* of the simple eigenvalue λ is defined as

$$Cond(\lambda) = \frac{\|x\|\|y\|}{|y^*x|} = \frac{1}{\cos \angle(x, y)}, \tag{2.8}$$

where x, y are the right and the left eigenvector associated with λ, respectively and $\cos \angle(x, y)$ is defined as in Definition 2.9.

Definition 2.23. Let λ be a simple eigenvalue of a symmetric matrix $A \in \mathbb{R}^{n\times n}$ closest to $\widetilde{\lambda} \neq sp(A)$ and

$$0 < \lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_n$$

be the spectrum of A. Then

$$\delta = \min_{\lambda_i \neq \lambda} |\lambda_i - \widetilde{\lambda}| \tag{2.9}$$

is the *gap* of the eigenvalue λ and

$$spr(A) = \lambda_n - \lambda_1 \tag{2.10}$$

the *spread* of the spectrum of A.

Theorem 2.24. *The Singular Value Decomposition (SVD)[61]*
Let A be a real m by n matrix, then there exist orthogonal matrices $U \in \mathbb{R}^{m\times m}$ and $V \in \mathbb{R}^{n\times n}$ such that

$$U^TAV = diag(\sigma_1, \ldots, \sigma_p) \in \mathbb{R}^{m\times n} \qquad p = \min\{m, n\}$$

where $\sigma_1 \geq \sigma_2 \geq \ldots \sigma_p \geq 0$ are called the singular values.

Definition 2.25. Let A be a n by n matrix, then the quantity $\kappa(A)$ defined as

$$\kappa(A) = \frac{\sigma_{max}(A)}{\sigma_{min}(A)},$$

is a *matrix condition number*, where $\sigma_{max}(A)$ and $\sigma_{min}(A)$ are the maximal and the minimal singular values of A, respectively.

2.1.6 Backward error analysis and condition numbers

The definitions and notation we adopt here are taken from a very nice review on finite precision computations [36].

Consider the computed solution $\widetilde{x}$ as the exact solution of a nearby problem. [36]

Definition 2.26. Let φ be a function that maps

$$d \longmapsto x = \varphi(d),$$

where d represents some initial data and $\widetilde{x}$ is a computed solution of the problem. Then the backward error at $\widetilde{x}$ is defined by

$$\mathcal{BE}(\widetilde{x}) = \inf\left(\|\delta d\| : \varphi(d + \delta d) = \widetilde{x}\right).$$

At this point we should also introduce the notion of the so-called *condition number*. Let us first start with a formal definition.

Definition 2.27. The condition number of φ as in Definition 2.26 is

$$\mathcal{C} = \limsup_{\delta d \to 0} \frac{\|\varphi(d + \delta d) - \varphi(d)\|}{\|\delta d\|},$$

where d represents some initial data of the problem and δd is a perturbation of d.

In simple words, the condition number $\mathcal{C}$ describes the sensitivity of the solution x to a perturbation δd in the data d.

The backward error analysis itself does not deliver any information about the differences between a true solution x and the computed value $\widetilde{x}$. However, it allows to formulate a so-called *forward error*, i.e.,

$$forward\ error \lesssim condition\ number \times backward\ error, \tag{2.11}$$

which means that

$$\|\widetilde{x} - x\| \lesssim \mathcal{C} \times \mathcal{BE}(\widetilde{x}).$$

All quantities in (2.11) have a completely different nature. The forward error determines the accuracy of the obtained approximation without specifying any possible sources of errors explicitly. The condition number delivers some a priori information which are imposed by the problem, e.g., it may reflect properties like nonlinear character of function φ etc., where the backward error depends on the choice of the algorithm and the computer arithmetic which makes it purely a posteriori information [36].

2.2 PDE eigenvalue problems

The study of eigenvalue problems for partial differential operators is a broad area of research including different types of linear and nonlinear operators and boundary conditions [12, 112]. In its most compact definition, one can describe a general PDE eigenvalue problem as follows. Let $\mathcal{L}$ be an arbitrary differential operator over functions in $\mathbb{F}$ ($\mathbb{R}$ or $\mathbb{C}$), e.g., linear, nonlinear, with constant or variable coefficients, of stationary or instationary type and let Ω be a bounded Lipschitz domain in $\mathbb{R}^d, d = 1, 2, \ldots$.
Then, a PDE eigenvalue problem is the problem of finding a non-zero function u, called an *eigenfunction* and a scalar λ, called an *eigenvalue*, such that

$$\mathcal{L}(u) = \lambda u \quad \text{in } \Omega.$$

If $\mathcal{L}^\star$ denotes an adjoint operator of $\mathcal{L}$, then the PDE eigenvalue problem

$$\mathcal{L}^\star(u^\star) = \lambda u^\star \ \text{ in } \Omega,$$

is an *adjoint* or *dual* problem and $u^\star$ is a *dual eigenfunction*. $u^\star$ is either the transpose u^T or the complex conjugate transpose u^* of u. A triple $(\lambda, u, u^\star)$ is called an *eigentriple*.

Remark 2.28. Obviously, PDE eigenvalue problems may be associated with any type of boundary conditions, i.e., Dirichlet, Neumann, or Newton [12], but for the simplicity, here we consider only homogenous Dirichlet conditions, i.e.,

$$u = 0 \text{ on } \partial\Omega.$$

Further, from now one we restrict our attention to linear second-order elliptic PDE eigenvalue problems defined on two-dimensional, polygonal domains $\Omega \subset \mathbb{R}^2$.

2.2.1 Classical and variational formulation of elliptic PDE eigenvalue problems

Let Ω be a bounded Lipschitz domain in $\mathbb{R}^2$ and $\mathcal{A}, \mathcal{B}$ a (non-)selfadjoint second-order elliptic operator and a selfadjoint, positive definite operator, respectively. The *classical* (or strong) formulation of the eigenvalue problem is given as

$$\begin{aligned} \mathcal{A}u &= \lambda \mathcal{B}u \quad \text{in } \Omega, \\ u &= 0 \qquad \text{on } \partial\Omega. \end{aligned}$$

Then, for given spaces V and H, the eigenpair $(\lambda, u) \in \mathbb{F} \times V$ satisfies the *variational formulation*

$$a(u, v) = \lambda b(u, v) \quad \text{for all } v \in V, \tag{2.12}$$

where $a : V \times V \to \mathbb{F}$, $b : H \times H \to \mathbb{F}$ are bilinear (or sesquilinear) forms, generated by $\mathcal{A}$ and $\mathcal{B}$, respectively, and $V \subset H \subset V^*$.

If $a(\cdot,\cdot)$ is V-elliptic and symmetric it defines an alternative inner product on V. The associated norm is called *energy norm* and defined as

$$|||u||| := a(u,u)^{\frac{1}{2}} \quad \text{for } u \in V.$$

Furthermore, from the coercivity and continuity of the bilinear form $a(\cdot,\cdot)$, the energy norm is equivalent to the standard norm on V, i.e.,

$$c\|u\|_V^2 \leq |||u|||^2 \leq C\|u\|_V^2,$$

where c, C are the coercivity and the continuity constant, respectively.

Remark 2.29. If $a(\cdot,\cdot)$ is symmetric, coercive and continuous then $(V, a(\cdot,\cdot))$ is a Hilbert space.

Since the operator $\mathcal{B}$ is assumed to be selfadjoint and positive definite, the corresponding bilinear form $b(\cdot,\cdot)$ defines a scalar product on H, which in many cases is a standard inner product of H.

The existence and uniqueness of a weak eigenpair (λ, u) is a simple consequence of the *Lax-Milgram Lemma* [99, §5.5, Theorem 13]. For the self-adjoint eigenvalue problem the existence of a unique eigenpair (λ, u) can also be proved using the *Riesz Representation Theorem* [99, §5.4, Theorem 11].

2.2.2 The Galerkin method and the Finite Element Method (FEM)

For a finite-dimensional subspace $V_h \subseteq V$, the eigenvalue problem:
Determine a non-trivial eigenpair $(\lambda, u) \in \mathbb{F} \times V$ with $\|u\|_H = 1$ such that

$$a(u,v) = \lambda b(u,v) \qquad \text{for all } v \in V,$$

is approximated by the *discrete eigenvalue problem*:
Determine a non-trivial eigenpair $(\lambda_h, u_h) \in \mathbb{F} \times V_h$ with $\|u_h\|_H = 1$ such that

$$a(u_h, v_h) = \lambda b(u_h, v_h) \quad \text{for all } v_h \in V_h.$$

The idea of approximating the exact solution of a variational formulation by an element from a given finite dimensional subspace V_h is known as the *Galerkin* method (or the *Ritz-Galerkin* method in the self-adjoint case) [41]. Since $V_h \subseteq V$, the bilinear form $a(\cdot,\cdot)$ is also bounded and coercive on V_h. Therefore, the existence of a unique Galerkin solution (λ_h, u_h) is inherited from the well-posedness of the original problem [99].
One should mention that there exist a number of other approximate methods, i.e., the Petrov-Galerkin method, the generalized Galerkin method, the method of weighted residuals, collocation methods etc., see [96, 99]. The main difference in all these approaches, is the choice of the solution space U_h where u_h lives and the trial space V_h of test functions v_h.

The Galerkin method is simple to analyze, since both spaces are taken to be the same, i.e., $U_h = V_h$, however, in many cases one should consider them to be distinct.

At this point, let us discuss some of the possible choices for space V_h, namely, basics of the Finite Element Method (FEM). For simplicity, we restrict ourself to consider only polygonal domains in $\mathbb{R}^2$.
Let $\mathcal{T}_h$ be a partition (or *triangulation*) of a domain Ω into triangles (elements) T, such that

$$\mathcal{T}_h := \bigcup_{T \in \mathcal{T}_h} = \overline{\Omega},$$

and any two distinct elements in $\mathcal{T}_h$ share at most a common edge E or a common vertex. For each element $T \in \mathcal{T}_h$ by $\mathcal{E}(T)$ and $\mathcal{N}(T)$ we denote the set of corresponding edges and vertices, respectively, where $\mathcal{E}_h$ ($\mathcal{N}_h$) denote all edges (vertices) in $\mathcal{T}_h$. Likewise, we define a diameter (longest edge) of an element as h_T. For each edge E we denote its length by h_E and the unit normal vector by n_E. The label h associated with the triangulation $\mathcal{T}_h$ denotes its mesh size and is given as $h := \max_{T \in \mathcal{T}} h_T$. We say that the triangulation is *regular* in the sense of Ciarlet [41] if there exist a positive constant ρ such that

$$\frac{h_T}{d_T} < \rho,$$

where d_T is the diameter of the largest ball that may be inscribed in element T, i.e., the minimal angle of all triangles in $\mathcal{T}_h$ is bounded away from zero. Of course the choice of triangle elements is not a restriction of the finite element method and is made only in order to clarify the notation.
Consider a regular triangulation $\mathcal{T}_h$ of Ω and the set polynomials $\mathbb{P}_p$ of total degree $p \geq 1$ on $\mathcal{T}_h$, which vanish on the boundary of Ω, see, e.g., [24]. Then the *Galerkin* discretization of (2.12) given by

$$a(u_h, v_h) = \lambda_h b(u_h, v_h) \quad \text{for all } v_h \in V_h^p, \tag{2.13}$$

with $V_h^p \subset V$, $dim\ V_h^p = n_h$, taken as

$$V_h^p(\Omega) := \left\{ v_h \in C^0(\overline{\Omega}) : v_h|_T \in \mathbb{P}_p \text{ for all } T \in \mathcal{T}_h \text{ and } v_h = 0 \text{ on } \partial\Omega \right\},$$

is called a *finite element discretization*. The Finite Element Method (FEM) [41] is a Galerkin method with a special choice of the approximating subspace, namely, the subspace of piecewise polynomial functions, i.e., continuous in Ω and polynomials on each $T \in \mathcal{T}_h$. For the sake of simplicity we consider only $\mathbb{P}_1$ finite elements, i.e., $p = 1$, and use $V_h := V_h^1$. So far our space V_h satisfies two, out of three, basic requirements on the right finite element space formulated in [41]. The last condition, which is crucial from the practical point of view, states that the space V_h should have a canonical basis of functions with small supports over $\mathcal{T}_h$. It is easily seen that the simplest set satisfying this condition is the set $\left\{\varphi_1^h, \ldots, \varphi_{n_h}^h\right\}$ of the *Lagrange* basis (also known as nodal or hat functions) [41]. Since, with this special

choice of basis, the solution u_h is determined by its values at the n_h grid points of $\mathcal{T}_h$, it can be written as

$$u_h = \sum_{i=1}^{n_h} u_{h,i}\varphi_i^h.$$

Then the discretized problem (2.13) can be written as a *generalized algebraic eigenvalue problem* of the form

$$A_h \mathbf{u}_h = \lambda_h B_h \mathbf{u}_h, \tag{2.14}$$

where the matrices

$$A_h := [a(\varphi_i^h, \varphi_j^h)]_{1\leq i,j\leq n_h}, \quad B_h := [b(\varphi_i^h, \varphi_j^h)]_{1\leq i,j\leq n_h}$$

are called *stiffness* and *mass* matrix, respectively. The representation vector $\mathbf{u}_h$ is defined as

$$\mathbf{u}_h := [u_{h,i}]_{1\leq i\leq n_h}.$$

Notice that, because $\left\{\varphi_1^h, \ldots, \varphi_{n_h}^h\right\}$ are chosen to have a small support over $\mathcal{T}_h$, the resulting matrices A_h, B_h are sparse.

2.2.3 Error analysis

After reading the previous section, several questions immediately arise: What is the best choice of the finite element space V_h? How accurate is the approximate solution u_h? How can we improve the quality of the approximation during the process of solving the eigenvalue problem? All these questions share a common denominator: they ask about the size of the error $e = u - u_h$. Here, we estimate the error e in some specified norm $\|\cdot\|$ or $|||\cdot|||$, however, there exist a number of results on estimating the error using other quality measures, see [14, 94].
Let us first introduce a notion of the so-called *error estimator*.

Definition 2.30. Given a norm $|||\cdot|||$, an approximation η to an error $|||e||| = |||u - u_h|||$ is called an *error estimator*.

There are two types of error estimators: *a priori* error estimators depending on the continuous solution u and *a posteriori* error estimators involving the approximation u_h.

A priori error estimators

In general, a priori error estimators give information about the asymptotic behavior of the error or the stability of the applied solvers. Likewise, they require particular regularity conditions of the solution, the stability properties of the discrete operator or the continuous solution u itself [2, 111]. Except of some simple one-dimensional boundary value problems, where an optimal finite element space can be constructed based on a priori estimates, see [93, Lecture 1] for details, all these conditions make a priori error estimators not computable and hardly applicable in practice. Of course if some a priori information about the solution is

known it can be relevant for the construction of efficient numerical algorithms, unfortunately, this is often not the case.
One of the first a priori error results obtained in [98] gives estimates to the eigenvalue/eigenfunction error in $L_2(\Omega)$ and energy norm depending on the regularity of the solution space, i.e.,

$$\|\!|u - u_h|\!\| \leq Ch^r, \quad \|u - u_h\| \leq Ch^r \|\!|u - u_h|\!\|, \quad |\lambda - \lambda_h| \leq C \|\!|u - u_h|\!\|^2 \leq Ch^{2r},$$

for $u \in H^{1+r}(\Omega)$, where $r \in (0, 1]$ is a regularity constant and C is a constant depending on the eigenvalue λ and the triangulation $\mathcal{T}_h$. Note that if $r = 1$ (convex domain), the solution u has to fulfill a $H^2(\Omega)$-regularity condition, which is very restrictive and excludes a large class of problems, e.g., the Laplace eigenvalue problem on the L-shape domain. Another key observation, which we know from the standard perturbation results for the matrix eigenvalue problems, is the fact that the eigenvalue error is proportional to the square of the eigenfunction error in the energy norm [107]. Therefore, the eigenvalues are usually much more accurate than the corresponding eigenfunctions. Furthermore, in [12] this result was generalized for the case of multiple eigenvalues. Also, in this case, the error behaves like $\mathcal{O}(h)$, in the mesh size h, for the energy norm of the eigenfunction and $\mathcal{O}(h^2)$ for the eigenvalue. To conclude, we present a recent a priori error estimator based on the angles between the eigenspaces introduced in [77], i.e.,

$$0 \leq \frac{\lambda_{h,k} - \lambda_k}{\lambda_{h,k}} \leq \left(1 + \max_{j=1,\dots,k-1} \frac{\lambda_{h,j}^2 \lambda_k^2}{|\lambda_{h,j} - \lambda_k|^2} \|\!|(I - Q)TP_{1,\dots,k-1}|\!\|^2\right) \sin^2_{\|\!|\cdot|\!\|} \angle(u_k, V_h),$$

where Q is an orthogonal projection onto V_h, T an inverse Laplace operator, $P_{1,\dots,k-1}$ an orthogonal projection onto $\operatorname{span}\{u_{h,1}, \dots, u_{h,k-1}\}$ and $\min\limits_{1,\dots,k-1} |\lambda_{h,j} - \lambda_k| \neq 0$. A presence of the exact eigenvalue λ_k in the upper bound explains, empirically well-known, poor approximation of the large eigenvalues [31].

A posteriori error estimators

Although, in general, a priori error estimates are not available, we still would like to control the quality of our numerical approximation and be able to terminate the algorithm as soon as a prescribed tolerance ε is reached, i.e., $\|\!|e|\!\| \leq \varepsilon$. Therefore, we need some computable quantity η which we can use to replace $\|\!|e|\!\|$, precisely the *a posteriori* error estimator which will approximate an actual error. The formal definition, see [30], states as follows.

Definition 2.31 (A posteriori error estimator)**.** A computable quantity η is called *a posteriori error estimator* if it can be extracted from the computed numerical solution u_h and the given data of the problem, i.e., the known domain Ω and its boundary $\partial\Omega$.

There are several important practical requirements on a posteriori error estimators. First, as the definition states, they should be computable. In contrast to a priori error estimators they depend on the stability properties of the continuous operator which are known. Moreover, they should use the approximate solution itself to check its quality. Secondly, calculating the estimator should be cheaper than computing the new numerical approximation (e.g.,

assembling the matrices). A global upper bound is sufficient to assure the accuracy of the solution, while a posteriori error estimator should provide local upper and lower bounds for the true error in some quality measure, e.g., energy norm $\|\cdot\|$ [2, 30, 111]. The latter properties of a posteriori error estimators η are called *reliability*

$$\|e\| \leq C_{rel}\eta + h.o.t_{rel}$$

and *efficiency*

$$\eta \leq C_{eff}\|e\| + h.o.t_{.eff}$$

with constants $C_{rel}, C_{eff} > 0$ independent of the mesh size h or polynomial degree p and higher-order terms $h.o.t_{rel}$, $h.o.t_{eff}$. We will call an a posteriori estimator a "good" a posteriori error estimator if it is reliable and efficient, i.e.,

$$C_1\|e\| \leq \eta \leq C_2\|e\|,$$

with $C_1 = \frac{1}{C_{rel}}$ and $C_2 = C_{eff}$. Basically, we want to obtain an accurate solution with an optimal use of resources and guarantee that the a posteriori error estimator captures the behavior of the actual error as $h \to 0$. In practice, we are often interested in the asymptotical exactness or efficiency of the a posteriori error estimator. Following [2], we call the error estimator η *asymptotically exact* if

$$\lim_{h\to 0} \theta = 1,$$

where $\theta := \frac{\eta}{\|e\|}$ is called a global *efficiency index*. An error estimator is called *efficient* if its efficiency index θ and its inverse are bounded independent on the mesh size [30].
A posteriori error estimators, following [38, 111], can be classified as follows:

Residual estimators (explicit error estimators) The explicit error estimators bound the global error in the finite element solution using the residual in the current approximation. Therefore, the main goal is to estimate the residual in a sufficiently accurate and efficient way. The standard residual type a posteriori error estimator consist of two main terms: the interior (volumetric) residual R_T determing how well the finite element approximation satisfies the PDE in its strong form on the interior of the domain and the edge residual R_E (jumps), the jumps of the normal derivative of the approximation u_h across the face E, reflecting the accuracy of the approximation [26, 111]. Thus, in general, the residual error estimator is defined as

$$\eta^2 := \sum_{T\in\mathcal{T}_h} h_T^2\|R_T\|^2_{L_2(T)} + \sum_{E\in\mathcal{E}_h} h_E\|R_E\|^2_{L_2(E)}, \tag{2.15}$$

with mesh-depending weights h_T, h_E and problem dependend, local norms $\|\cdot\|_{L_2(T)}$, $\|\cdot\|_{L_2(E)}$. There is a large amount of literature on residual type error estimators for eigenvalue problems, e.g., [2, 50, 111], to mention only a few among them. The crucial relation in the residual type error analysis is the equivalence between the approximation error $\|u - u_h\|$ and the norm of the residual in the dual space V^* of V, i.e., $\|R(v)\|_{V^*} = \|a(u_h, v) - \lambda_h b(u_h, v)\|_{V^*}$, which, up to the higher-order terms was also proved for the selfadjoint eigenvalue problems in [31]. Nevertheless, it is still a challenge to compute the negative norm of $R(v)$.

Local problem-based estimators (implicit estimators) Here, instead of considering the original discrete problem, local analogues of the residual equations are solved and suitable norms of the local solutions are used for the error estimation [26]. In order to capture the local behavior of the solution and to get accurate information about the error, the local problems usually involve only small subdomains of Ω and more accurate finite element spaces. In terms of complexity the solution of all local problems should cost less than assembling the stiffness matrix of the original discrete problem. Partition of unity or equilibration estimators are discussed in [1, 26, 34]. We do not know any local problem-based error estimators designed specifically for eigenvalue problems.

Averaging estimators (recovery-based estimators) These error estimators use a local extrapolation or averaging technique, see [30, 117]. The error of the approximation can be controlled by a difference of a low-order approximation (e.g., a piecewise constant function) and a finite element solution obtained in the space of higher-order elements (e.g., globally continuous piecewise linear functions) which additionally satisfy more restrictive continuity conditions than the approximation itself [30]. Reference [30] gives a nice overview of averaging techniques in a posteriori finite element error analysis in general. Particular error estimators dedicated to the eigenvalue problems are discussed in [31, 88].

Hierarchical estimators (multilevel estimators) The main idea of a hierarchical error estimator is to evaluate the residual for the finite element solution $u_H \in V_H$ with respect to another finite element space V_h that satisfies $V_H \subset V_h \subset V$. Then the error $\|\!|u - u_H|\!\|$ can be bounded by

$$\eta := \|\!|u_h - u_H|\!\|,$$

where $u_h \in V_h$ [38, 55]. Usually V_h corresponds to a refinement $\mathcal{T}_h$ of $\mathcal{T}_H$ or consists of higher-order finite elements. This approach takes advantage of a so-called *saturation assumption* [16]. The error of a fine discrete solution u_h is supposed to be smaller than the error of the coarse solution u_H in the sense of an error reduction property, i.e.,

$$\|\!|u_h - u|\!\| \leq \beta \|\!|u_H - u|\!\|,$$

where $\beta \in (0, 1)$. Good general references concerning hierarchical estimators are [16, 15, 47, 55]. Some aspects of the hierarchical error estimators for eigenvalue problems are discussed in [91, 92].

Goal-oriented estimators The objective in goal-oriented error estimation is to determine the accuracy of the finite element solution with respect to some physically relevant quantity of the solution, the so-called quantity of interest, e.g. velocity or flow rates. The error in the quantity of interest is then related to the residual. One of the well-known techniques of goal-oriented error estimation is the *Dual Weighted Residual* method (or shortly DWR method) introduced in [97]. The theory of the DWR method was successfully applied to determine a goal-oriented a posteriori error estimator for eigenvalue problems based on the solution of the adjoint (dual problem), see [67]. For more details, see [14, 94].

Let us now give a brief historical overview of the evolution of a posteriori error estimators for eigenvalue problems. A first approach on a posteriori error analysis for symmetric second-order elliptic eigenvalue problems can be found in [111, §3.4]. A combination of a posteriori and a priori analysis was used in [79] to prove a posteriori estimates of optimal order. Under the assumption that the mesh is fine enough and that the eigenfunction possesses an appropriate regularity, it was proved that for smooth eigenvectors the error is bounded in terms of the mesh size, a stability factor, and the residual. However, this result require a H^2-regularity assumption of the eigenfunction, which excludes problems with local singularities. In [50] based on the residual equation, the approximation error $|||u - u_h|||$ was shown to be equivalent, up to higher-order terms, to the residual type estimator. Secondly, since the edge residual is an upper bound for the volumetric part of the residual, a new simpler error estimator was presented.
In [88] similar results are achieved by applying a local averaging technique. Based on the analysis of the residual equations, in [92] an a posteriori error estimator was presented that works on a subspace of eigenvector approximations obtained by the preconditioned inverse iteration.
Recently in [31] the results of [50, 88] were improved by showing that for all eigenvalues the refinement is possible without the element contributions in the estimator and that the higher order terms can in fact be neglected.
An approach for non-symmetric elliptic eigenvalue problems was presented in [67]. Using the general optimal control framework of Galerkin approximations of nonlinear variational equations of [18] a residual-based estimator with explicitly given remainder terms was presented. Unfortunately also this result, as it needs the knowledge of the exact solution of the adjoint problem and provides only upper bounds of the error, is not a true a posteriori error estimate. Surveys about a posteriori error estimation can be found in [2, 111].
Some of above mentioned a posteriori error estimators are presented in Table 2.1.

Residual a posteriori error estimator	
$\eta^2_{DPR} := \sum_{T\in\mathcal{T}_h} h_T^2\lambda_h^2\|u_h\|^2_{L_2(T)} + \sum_{E\in\mathcal{E}_h} h_E\left\|[\nabla u_h]\cdot n_E\right\|^2_{L_2(E)}$	[50]
$\eta^2_{CG} := \sum_{E\in\mathcal{E}_h} h_E\left\|[\nabla u_h]\cdot n_E\right\|^2_{L_2(E)}$	[31]
Averaging a posteriori error estimator with a local averaging operator $A(\cdot)$	
$\mu^2 := \sum_{T\in\mathcal{T}_h}\left(h_T^2\|\lambda_h u_h + \operatorname{div}(A(u_h))\|^2_{L_2(T)} + \|A(u_h) - \nabla u_h\|^2_{L_2(T)}\right)$	[88]
$\mu^2 := \sum_{T\in\mathcal{T}_h}\|A(u_h) - \nabla u_h\|^2_{L_2(T)}$	[31]
Goal-oriented a posteriori error estimator	
$\eta^2 := \sum_{T\in\mathcal{T}_h} h_T^2\|\mathcal{A}u_h + \lambda_h u_h\|^2_{L_2(T)} + \sum_{E\in\mathcal{E}_h} h_E\left\|[\nabla u_h]\cdot n_E\right\|^2_{L_2(E)}$ $+ \sum_{T\in\mathcal{T}_h} h_T^2\|\mathcal{A}^\star u_h^\star + \lambda_h^\star u_h^\star\|^2_{L_2(T)} + \sum_{E\in\mathcal{E}_h} h_E\left\|[\nabla u_h^\star]\cdot n_E\right\|^2_{L_2(E)}$	[67]

Table 2.1: Sample a posteriori error estimators.

2.2.4 The Adaptive Finite Element Method (AFEM)

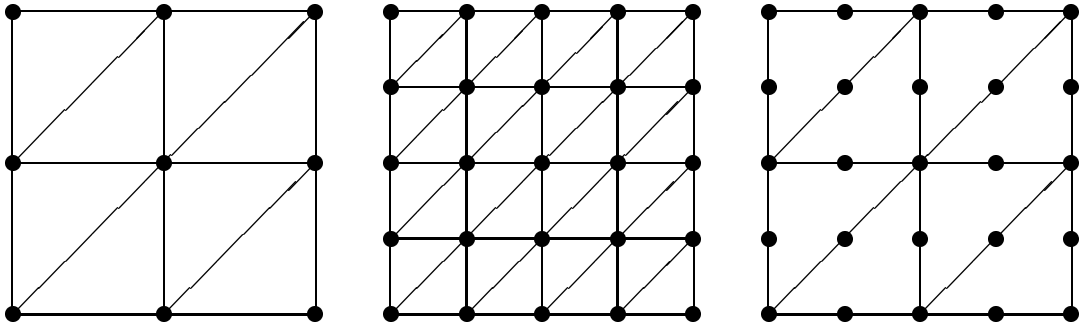
It is well-known that the accuracy of the numerical approximation is determined by the regularity of the solution. The most efficient approximations of smooth functions can be obtained using large higher-order finite elements (p-FEM), where local singularities arising from re-entrant corners, interior or boundary layers can be captured by small low-order elements (h-FEM) [38]. Unfortunately, in real-world applications, none of those phenomena are known a priori, which makes it impossible to construct an optimal finite dimensional space from scratch.
The *Adaptive Finite Element Method* (AFEM) is a numerical scheme which, based on the quality of the numerical approximation (a posteriori error estimator) automatically adapts the finite element space in order to determine a sufficiently accurate final solution. Moreover, this adjustment of the solution is accomplished during the process of solving the eigenvalue or the boundary value problem. Let us briefly introduce AFEM following [2, 26, 38, 60, 93, 103].
A typical loop of the AFEM consists of the four steps

$$\textsc{Solve} \longrightarrow \textsc{Estimate} \longrightarrow \textsc{Mark} \longrightarrow \textsc{Refine}.$$

As first step, the eigenvalue problem is solved on initial mesh $\mathcal{T}_0$ to produce a finite element approximation (λ_h, u_h) of the continuous eigenpair (λ, u). In the second step the total error in the computed solution is estimated by an a posteriori error estimator η. If the global error is sufficiently small, the adaptive algorithm terminates and returns (λ_h, u_h) as a final approximation, otherwise the local contributions of the error are estimated on each element by the so-called *refinement indicators*. Based on those, the elements for refinement are selected and form the set $\mathcal{M} \subset \mathcal{T}_h$ of marked elements. The final step involves refinement of marked elements. Since the resulting mesh may possess hanging nodes an additional closure procedure is applied is order to obtain a new conforming mesh. As a consequence, the adaptive finite element method (AFEM) generates a sequence of nested triangulations $\mathcal{T}_0, \mathcal{T}_1, \ldots$ with corresponding nested spaces

$$V_0 \subseteq V_1 \subseteq \ldots \subseteq V_h \subset V.$$

The two components for the adaptive finite element method which are of particular interest are steps Estimate and Refine. The refinement of the finite element solution can be performed using various techniques like moving grid points (r-refinement), subdividing elements of a fixed grid (h-refinement), applying locally higher-order basis functions (p-refinement) or any combinations of those [38], see Figure 2.1 for illustration.
To make the discussion as simple as possible, here, we consider only the h-refinement of the triangle elements. The most common refinement techniques based on edge marking are presented in Figure 2.2.
As we have mentioned before, applying these refinement procedures may lead to nonconforming meshes. Therefore, the so-called *closure algorithm* [31] is applied to overcome this drawback and get a regular triangulation. Summarizing, if the edge E of the element T is marked for refinement, the reference edge (e.g., longest edge) of T is marked as well. Since



Original mesh *A uniform h-refinement* *A uniform p-refinement*

Figure 2.1

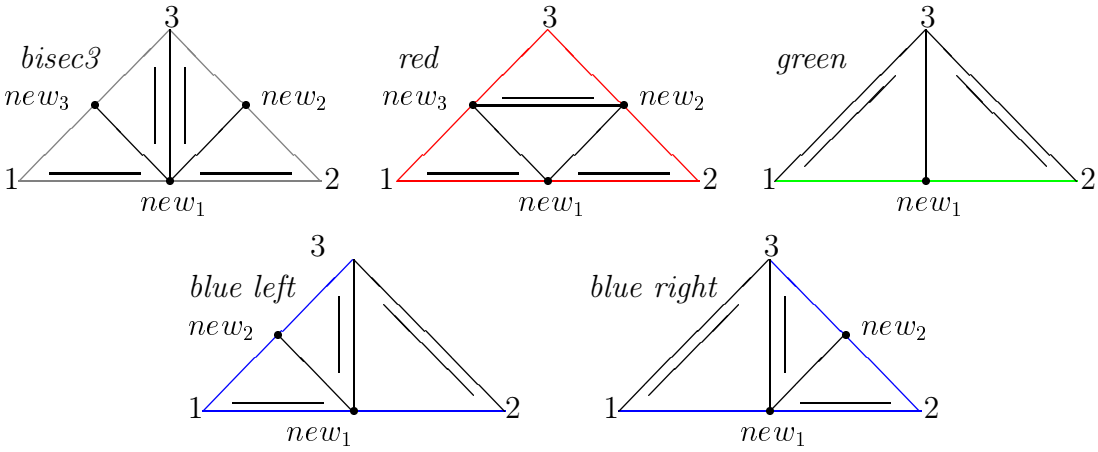


Figure 2.2: *Bisec3*, *red*, *green* and *blue* refinement. The edges marked by MARK step are colored. The new reference edge is marked through a second line in parallel opposite the new vertices new_1, new_2 or new_3 [31].

each element has $k = 0, 1, 2,$ or 3 edges marked for the refinement, if $k \geq 1$, the reference edge belongs to it. Morover, the choice of a refinement method, see Figure 2.2, depends on k, for instance, if $k = 1$ the *green* refinement is used etc.. For more details about adaptive refinement strategies see, e.g., [2, 38, 111].

To assure a good balance between the refined and the un-refined regions such that the overall accuracy is optimal, reliable estimates of the accuracy of the computed approximations are required. Moreover, since the automatic adaptation is steered by the size of the error indicators which measure the local quality of the solution, local upper and lower bounds for the true error are necessary to ensure the optimal use of resources, i.e., obtaining a numerical solution with a prescribed tolerance using the minimal number of degrees of freedom (DOFs) or achieve the maximal accuracy with a fixed finite element space [14]. Unfortunately, even the most accurate global error estimators itself do not guarantee the efficiency of the adaptive algorithm, however, their local counterparts do. A local error indicator (refinement indicator)

for an element $T \in \mathcal{T}_h$ is usually denoted by η_T and related to a global error estimator η through

$$\eta^2 = \sum_{T \in \mathcal{T}} \eta_T^2.$$

For example the refinement indicator extracted from the standard residual-type a posteriori error estimator (2.15) is given as

$$\eta_T^2 = h_T^2 \left\| R_T \right\|_{L_2(T)} + \sum_{E \in \mathcal{E}(T)} h_E \left\| R_E \right\|_{L_2(E)}.$$

Our next task is to give more insight into the MARK procedure. We already know that the set $\mathcal{M}$ of elements to refine, is determined based on the sizes of the refinement indicators. Now, a question arises: How do we decide which elements T should be added to the set $\mathcal{M}$? The definitions and notation we adopt here are taken from [26]. The process of selecting the elements of $\mathcal{M}$ with respect to a threshold $L \in \mathbb{R}, L > 0$ is called the *marking criterion* or the *marking strategy* and the set of marked elements is defined as

$$\mathcal{M} := \{T \in \mathcal{T}_h : L \leq \eta_T\}.$$

Notice that the choice of a threshold L is essential for the efficiency of the adaptive algorithm since it directly determines the size of the set $\mathcal{M}$. The simplest marking strategy takes fixed rate of elements or elements for which refinement indicators are larger than some fixed tolerance. A more sophisticated strategy is the *maximum criterion*, where the elements selection is based on a fixed fraction Θ of the maximal refinement indicator in $\mathcal{T}_h$, i.e.,

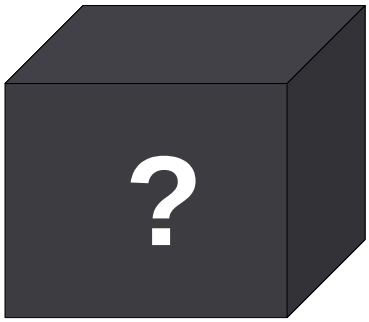
$$L := \Theta \max_{T \in \mathcal{T}_h} \eta_T,$$

with $0 \leq \Theta \leq 1$. The most interesting, especially in the context of optimality of a standard adaptive finite element method, is a *Dörfler marking* strategy [48]. The threshold L is then taken as the largest value such that

$$(1 - \Theta)^2 \sum_{T \in \mathcal{T}_h} \eta_T^2 \leq \sum_{T \in \mathcal{M}} \eta_T^2, \tag{2.16}$$

where $0 \leq \Theta \leq 1$.

Last, but not least, is the step SOLVE. If we would like to keep the standards regarding a general treatment of this part of the adaptive finite element method we should put



and finish the discussion. Except for very few works, e.g., [8, 68] for boundary value problems or [32, 57, 91] for eigenvalue problems, this part of the adaptive algorithm is considered as a black box. The most common approach is to run the iterative eigensolver up to the final accuracy to obtain the exact *Galerkin solution*. Some interesting computational schemes for elliptic eigenvalue problems are the preconditioned eigensolvers PINVIT [92] or LOBPCG [73]. In the context of the adaptive finite element method a two-grid discretization scheme introduced in [115] may be of particular interest as well. Although, determining the Galerkin solution for boundary value problems, on currently available computers, is not regarded as a problem, the situation changes completely if we start to think about eigenvalue problems. Since this phenomenon is the main subject of this thesis we will not go into too many details here, since this will be discussed in detail later.
Finally let us briefly review some results addressing the convergence and the optimality of adaptive algorithms for self-adjoint eigenvalue problems. In [59] the convergence of an adaptive linear finite element method is proved with the refinement procedure that considers both a standard a posteriori error estimator and eigenfunction oscillations, while the global convergence result of [31] requires no additional mesh size assumptions and inner node properties. In [43], based on the relation between the eigenvalue problem and the associated boundary value problem, the uniform convergence and the optimal complexity of the adaptive finite element eigenvalue approximation scheme are proved. Nearly at the same time, in [56] the convergence of the AFEM for any reasonable marking strategy and any initial mesh was shown. Complexity estimates for adaptive eigenvalue computations were discussed in [42]. Further eigenvalue/eigenvector estimates are obtained in [62, 74, 75]. Recently the quasi-optimal convergence result for the inexact AFEM for self-adjoint eigenvalue problems was introduced in [32]. Morover, [116] show the quasi-optimality of the inexact inverse iteration coupled with adaptive finite element methods for the self-adjoint eigenvalue problem in its operator form. We do not know of any convergence result of the AFEM for non-self-adjoint eigenvalue problems.
For a general introduction to the subject of adaptive finite element methods see [38, 93] and for the practical implementation details see, e.g., [60].

2.3 The Generalized Algebraic Eigenvalue Problem

In the previous section we have discussed some basic facts about elliptic PDE eigenvalue problems and their discretization methods. We have shown that the FEM discretization leads to the generalized algebraic eigenvalue problem.

Given two matrices $A, B \in \mathbb{R}^{n\times n}$. Find scalars $\lambda \in \mathbb{F}$ and vectors $x \in \mathbb{F}^n, x \neq 0$, such that

$$Ax = \lambda Bx. \tag{2.17}$$

Then λ is an eigenvalue of the pair (A, B), and x is an eigenvector corresponding to λ, the pair $(\lambda, x) \in \mathbb{F} \times \mathbb{F}^n$ is called an eigenpair.

We consider the matrix B to be symmetric positive definite, whereas A can be both symmetric or non-symmetric depending on the underlying differential operator $\mathcal{A}$. Due to the special choice of nodal basis functions in finite element discretization, matrices A, B are large and sparse with only a few nonzero entries per row, see Section 2.2.2, which excludes the use of direct eigensolvers such as the QR algorithm [61]. Therefore, *iterative methods*, e.g., Krylov subspace methods [101], using matrix-vector multiplications instead of matrix factorizations, are of particular interest.

In this section we recall the main idea of the Arnoldi/Lanczos method and its modification for the generalized eigenvalue problem following [81, 104]. We introduce the theoretical framework of homotopy methods and its application in solving non-symmetric generalized eigenvalue problems. Finally, since iterative methods deliver only approximate solutions, we recall some of the main results from the perturbation theory for eigenvalue problems.

2.3.1 The Arnoldi/Lanczos method

Let us consider for a moment the standard eigenvalue problem.
Determine $(\nu, x) \in \mathbb{F} \times \mathbb{F}^n$ *such that*

$$Sx = \nu x, \tag{2.18}$$

with a sparse matrix $S \in \mathbb{R}^{n\times n}$. The Arnoldi method is an iterative method based on the orthogonal projection onto the so-called *Krylov subspace*

$$\mathcal{K}_m \equiv \mathcal{K}_m(S, v_1) := span\{v_1, Sv_1, \ldots, S^{m-1}v_1\},$$

where $v_1 \in \mathbb{R}^n$ is a starting vector. The exact eigenpair (ν, x) of (2.18) is approximated by the pair $(\widetilde{\nu}, \widetilde{x})$, with vector $\widetilde{x}$ belonging to $\mathcal{K}_m$. As for the standard orthogonal projection method, the eigenpair approximation is found by imposing the *Galerkin condition*, i.e.,

$$S\widetilde{x} - \widetilde{\nu}\widetilde{x} \perp \mathcal{K}_m,$$

or, equivalently,

$$(S\widetilde{x} - \widetilde{\nu}\widetilde{x}, y) = 0, \text{ for all } y \in \mathcal{K}_m.$$

In simple words, one is interested in the best possible approximation $\widetilde{x}$ to the eigenvector x from the subspace $\mathcal{K}_m$.

The Arnoldi method constructs ideally an orthonormal basis $v_1, \ldots, v_k$ for the Krylov subspace

$$\mathcal{K}_m(S, v_1) = span\{v_1, Sv_1, \ldots, S^{m-1}v_1\}.$$

With this orthonormal basis one obtains the *m-step Arnoldi factorization* of S of the form

$$SV_m = V_m H_m + f_m e_m^T,$$

where $V_m = [v_1, \ldots, v_m] \in \mathbb{F}^{n\times m}$, $V_m^H f_m = 0$ and $H_m = V_m A V_m \in \mathbb{F}^{m\times m}$ is an upper Hessenberg matrix with nonnegative subdiagonal elements. If S is Hermitian then H_m is real, symmetric, and tridiagonal, and the method is called the *Lanczos method*.

An alternative way to write this factorization is

$$SV_m = V_m H_m + f_m e_m^T = V_m H_m + \beta_m v_{m+1} e_m^T = [V_m, v_{m+1}] \begin{bmatrix} H_m \\ \beta_m e_m^T \end{bmatrix}, \tag{2.19}$$

where $\beta_m = \|f_m\|$, $v_{m+1} = \frac{1}{\beta_m} f_m$.
Approximate eigenpairs $(\widetilde{\nu}, \widetilde{x})$, i.e., *Ritz pairs*, are determined from eigenpairs (θ, s) of the small matrix H_m, i.e.,

$$\widetilde{\nu} = \theta, \quad \widetilde{x} = V_m s, \quad \text{where} \quad H_m s = \theta s.$$

Furthermore, the norm of the corresponding residual vector yields

$$\|r\| = \|S\widetilde{x} - \theta\widetilde{x}\| = \|(SV_m - V_m H_m)s\| = |\beta_m e_m^T s| = \|f_m\||e_m^T s|. \tag{2.20}$$

The number $\|f_m\||e_m^T s|$ is called the *Ritz estimate*. In the ideal situation $f_m = 0$ and Ritz pairs are exact eigenpairs of S. However, in general the quality of the approximation is measured by the size of the Ritz estimate relative to $\|S\|$ [81]. Using the backward error analysis introduced in Section 2.1.6 we can write

$$(S + E)\widetilde{x} = \widetilde{\nu}\widetilde{x}, \text{ with } E = -(e_m^T s) f_m \widetilde{x}^*.$$

Therefore, the size of the backward error $\|E\| = \|f_m\||e_m^T s|$ is exactly the Ritz estimate. In the symmetric case, a small Ritz estimate implies accurate approximation, while in the non-symmetric case the conditioning of the eigenvalue has to be considered additionally, see Section 2.3.3 for details.
We consider now the case of the generalized eigenvalue problem. The most common technique for converting the generalized problem to the standard problem is a *spectral transformation* suggested in [53]. With a (real or complex) shift σ the generalized eigenvalue problem (2.17) is transformed to the standard formulation, i.e.,

$$Ax = \lambda Bx \quad \Leftrightarrow \quad (A - \sigma B)^{-1} Bx = \nu x, \text{ where } \nu = \frac{1}{\lambda - \sigma}.$$

The eigenvector x of the transformed problem is also the eigenvector of the original problem corresponding to the eigenvalue $\lambda = \sigma + \frac{1}{\nu}$.
Then the Arnoldi process is applied to the matrix $S = (A - \sigma B)^{-1} B$. In order to keep the symmetry (if A, B are symmetric and B is positive definite) the standard inner product is replaced by the B-inner product, i.e.,

$$(x, y)_B = y^* B x.$$

This version of the Arnoldi/Lanczos method is often called the B-Arnoldi method since it constructs a B-orthonormal basis for the subspace $\mathcal{K}_m(S, v_1)$.
The Ritz pair $(\widetilde{\lambda}, \widetilde{x})$ is then given as

$$\widetilde{\lambda} = \sigma + \frac{1}{\nu}, \quad \widetilde{x} = V_m s,$$

where (ν, s) is an eigenpair of H_m. The corresponding Ritz estimate is given by

$$\|A\widetilde{x} - \widetilde{\lambda}B\widetilde{x}\| = \frac{1}{|\nu|^2}\|Bf_m\||e_m^T s|.$$

In order to keep the discussion simple, we do not consider many details like implicit restarts, stopping criteria and choices of parameters m, σ etc.. For more information on the Arnoldi method see, e.g., [81, Chapter 4], [104]. In our computations we use the ARPACK implementation of the implicitly restarted Arnoldi method, i.e., the shift-invert mode with real shift σ for symmetric and non-symmetric problems, see [81, pp. 31–36], or its MATLAB interface `eigs`. In Section 4.4 our own implementation of the $(A + B)$-Arnoldi method is used, i.e., the Arnoldi/Lanczos method with the $(A + B)$-inner product.

2.3.2 Homotopy methods

The history of homotopy methods, also known as continuation, embedding or successive loading methods, is very hard to track back due to an immense variety of applications. See, e.g., the nice survey on continuation and path following methods [5]. Here we are discussing the application of the homotopy method to linear eigenvalue problems as proposed in [39] and developed further in [82, 83, 85, 86]. We introduce the simple idea of the method given in [87].

Let A be a real $n \times n$ matrix for which eigenpairs are to be computed and let A_0 be a matrix with a known set of eigenpairs. In order to determine the eigenpairs of A, the appropriate linear *homotopy equation* is defined, i.e.,

$$\mathcal{H}(t) = (1 - t)A_0 + tA, \qquad 0 \leq t \leq 1. \tag{2.21}$$

At $t = 0$, eigenpairs of $\mathcal{H}(t)$ are exactly given by eigenpairs of the initial matrix A_0. Successive change of the parameter t, with some time step τ, transforms the starting set of eigenpairs to desired eigenpairs of the matrix A obtained at $t = 1$. The process of following eigenpairs through intermediate steps $t \in (0, 1)$ is known as an eigenpath continuation, where following [87], as the eigenpath we denote the evolution of the eigenvalue and associated eigenvector as a function of time, i.e., $(\lambda(t), x(t))$.

The homotopy method can be interpreted as a globally convergent Newton method. Unlike the classical Newton method the set of good starting values is known and given by eigenpairs of the initial matrix A_0. Additionally, the well-known drawback of only local convergence can be removed. However, the continuation method suffers from some other problems.

In [71] it was proved that, in general, eigenvalues and eigenvectors of $\mathcal{H}(t)$ are analytic functions of t. For the symmetric matrix A, the only questionable situation may appear when, during the continuation process, we switch (jump) from one eigenpath to another, e.g., from $(\lambda_1(t), x_1(t))$ to $(\lambda_2(t), x_2(t))$. This so-called *path-jumping* may happen if, because of too large selection of the step size τ, the iterative solver starts to converge to the wrong eigenvector, i.e., $x_2(t)$ instead of $x_1(t)$.

In the non-symmetric case the situation is even more complicated. On top of the path jumping phenomenon some additional difficulties arise. In particular, non-symmetric problems

have in general non-orthogonal eigenvectors and therefore may be highly ill-conditioned. Moreover, it may happen that at some point t two eigenpaths achieve the same value, i.e., $(\lambda_1(t), x_1(t)) = (\lambda_2(t), x_2(t))$. Such *bifurcation point* appears, for example, when a real eigenvalue splits into complex conjugate pair. For more details about different types of bifurcation points, see, e.g., [87].

2.3.3 Perturbation results for the generalized eigenvalue problem

In Section 2.2 we have introduced a priori and a posteriori bounds to control the error between continuous and discrete eigenpairs. However, as mentioned before, FEM discretizations lead to large and sparse generalized eigenvalue problems which have to be solved using iterative methods. Therefore, quantities obtained by iterative eigensolvers, i.e, $(\widetilde{\lambda}, \widetilde{x})$, are only some approximations of exact discrete eigenpairs (λ, x). In order to determine the accuracy of these approximations we are interested in establishing error bounds which depend on some computable quantities, i.e., residual vectors etc.. As we will see, the perturbation theory for eigenvalue problems, although studied by many authors, i.e., [22, 69, 95, 105, 113], is far from complete, especially in the non-symmetric case.

The first error bound which we consider is based on the backward error analysis introduced in Section 2.1.6.

Theorem 2.32. *[46] Let λ be a simple eigenvalue of a matrix A with right eigenvector x and left eigenvector y, normalized so that $\|x\|_2 = \|y\|_2 = 1$. Let $\widetilde{\lambda} = \lambda + \delta\lambda$ be the corresponding eigenvalue of $A + E$. Then*

$$\widetilde{\lambda} - \lambda = \frac{y^*Ex}{y^*x} + O(\|E\|^2),$$

and

$$|\widetilde{\lambda} - \lambda| \leq \frac{\|E\|}{|y^*x|} + O(\|E\|^2).$$

Proof. See [46, Theorem 4.4]. □

The quantity $Cond(\lambda) = \frac{\|x\|\|y\|}{|y^*x|} = \frac{1}{\cos\angle(x,y)}$ is the condition number of the simple eigenvalue λ, see Definition 2.22, while $\|E\|_2$ determines the size of the perturbation of the original matrix A. We now introduce an analogue of Theorem 2.32 for the generalized eigenvalue problem with the positive definite matrix B.

Theorem 2.33. *Consider a pair (A, B) of real $n \times n$ matrices and assume that B is positive definite. Let λ be a simple eigenvalue of the pair (A, B) with right eigenvector x and left eigenvector y, normalized so that $\|x\|_2 = \|y\|_2 = 1$. Let $\widetilde{\lambda} = \lambda + \delta\lambda$ be the corresponding eigenvalue of the pair $(A + E, B)$ with eigenvector $\widetilde{x} = x + \delta x$. Then*

$$\widetilde{\lambda} - \lambda = \frac{y^*Ex}{y^*Bx} + O(\|E\|^2),$$

and

$$|\widetilde{\lambda} - \lambda| \leq \frac{\|E\|}{|y^*Bx|} + O(\|E\|^2).$$

The quantity $\frac{1}{|y^*Bx|}$ *is again the condition number of the simple eigenvalue* λ.

Proof. Since the matrix B is assumed to be positive definite, we can put the perturbation matrix E only in A such that $(\widetilde{\lambda}, \widetilde{x})$ is the exact eigenpair of the matrix pair $(A+E, B)$, see [13] for more details. Therefore, the proof follows directly from the proof of Theorem 2.32. □

Proposition 2.34. *Let* A, B *be* $n \times n$ *matrices and let* B *be invertible. Let* $\widetilde{\lambda}$ *be a computed eigenvalue of the matrix pair* (A, B). *Let* $r = A\widetilde{x} - \widetilde{\lambda}B\widetilde{x}$. *Then* $\widetilde{\lambda}$ *is an exact eigenvalue with associated eigenvector* $\widetilde{x}$ *of the pair* $(A - E, B)$, *where* $E\widetilde{x} = r$.

Proof. For the computed eigenpair $(\widetilde{\lambda}, \widetilde{x})$ of $Ax = \lambda Bx$ and the residual vector $r = A\widetilde{x} - \widetilde{\lambda}B\widetilde{x}$ we have

$$(A - E)\widetilde{x} = \widetilde{\lambda}B\widetilde{x}.$$

Thus, we obtain

$$E\widetilde{x} = A\widetilde{x} - \widetilde{\lambda}B\widetilde{x},$$

and hence $(\widetilde{\lambda}, \widetilde{x})$ is an exact eigenpair of $(A - E, B)$, where $E\widetilde{x} = r$. □

We now introduce the result which allows to determine the smallest perturbation E.

Theorem 2.35 (Kahan-Parlett-Jiang [69])**.** *Let* $A \in \mathbb{R}^{n\times n}$ *and two unit vectors* $\widetilde{x}, \widetilde{y}^* \in \mathbb{R}^n$, *with* $\widetilde{y}^*\widetilde{x} \neq 0$, *be given. For a scalar* $\widetilde{\lambda}$, *define residual vectors*

$$r = A\widetilde{x} - \widetilde{x}\widetilde{\lambda}, \quad s^* = \widetilde{y}^*A - \widetilde{\lambda}\widetilde{y}^*.$$

Let $\mathcal{E} = \left\{E : (A - E)\widetilde{x} = \widetilde{x}\widetilde{\lambda},\ \widetilde{y}^*(A - E) = \widetilde{\lambda}\widetilde{y}^*\right\}$. *Then*

$$\min_{E\in\mathcal{E}} \|E\| = \max\{\|r\|, \|s^*\|\},$$

If $\widetilde{\lambda}$ *is chosen to be a Rayleigh quotient of* $\widetilde{x}$ *and* $\widetilde{y}$, *i.e.,* $\rho(\widetilde{x}, \widetilde{y}^*) = \frac{\widetilde{y}^*A\widetilde{x}}{\widetilde{y}^*\widetilde{x}}$, *then the minimum is achieved by the perturbation*

$$E = r\widetilde{x}^* + \widetilde{y}^*s^*.$$

Proof. See [69]. □

Since the bound in Theorem 2.32 holds for any perturbation E, we can choose

$$E = arg\min_{E\in\mathcal{E}} \|E\|,$$

which means that

$$|\widetilde{\lambda} - \lambda| = \frac{\|x\|\|y\|}{|y^*x|}\|E\| + O(\|E\|^2) = Cond(\lambda)\max\{\|r\|, \|s^*\|\} + O(\|E\|^2). \tag{2.22}$$

It is easy to notice that $\|E\|$ is nothing else but the backward error, i.e., the size of the smallest perturbation for which the approximate solution is the exact solution of the perturbed problem. The above result, known as the eigenvalue *forward error*, yields that the eigenvalue

error is bounded by the eigenvalue condition number $Cond(\lambda)$, which can be viewed as an a priori information of the problem, since it is independent of the iterative solver, and the corresponding backward error.
The bound introduced in Theorem 2.35, though very important, is hard to use in practice. As we have mentioned in Section 2.3.1, there is no problem in determining the size of the backward error provided by the algorithm as a by product of each iteration, however, the eigenvalue condition number is an a priori property of the eigenvalue which, apart from some special cases, is not known. Let us now consider the simple case of the generalized symmetric eigenvalue problem, where the eigenvalue condition number is known.

2.3.3.1 The Symmetric Case

Let us first consider the eigenvalue problem with the symmetric matrix A. By the Courant-Fisher minimax [46] theorem we know that the eigenvalue problem $Ax = \lambda x$ has a finite number of real eigenvalues and a complete set of orthonormal eigenvectors. The symmetry of the matrix A has a very important consequence, namely, the left eigenvector is simply a transpose of the right eigenvector, and the condition number of the eigenvalue is equal to 1 [101]. The following theorem may be considered as a special case of Theorem 2.32.

Theorem 2.36 (Weyl)**.** *Let A and E be $n \times n$ symmetric matrices. Let $\lambda_1 \leq \ldots \leq \lambda_n$ be eigenvalues of A and let $\widetilde{\lambda}_1 \leq \ldots \leq \widetilde{\lambda}_n$ be eigenvalues of $A + E$. Then*

$$|\widetilde{\lambda}_i - \lambda_i| \leq \|E\|_2,$$

for each of the eigenvalues individually.

Proof. See [46, p.201]. □

Theorem 2.37 (Krylov-Weinstein Inequality)**.** *For any nonzero vector $\widetilde{x}$ and any scalar value $\widetilde{\lambda}$ there is an eigenvalue λ of A such that*

$$|\lambda - \widetilde{\lambda}| \leq \frac{\|A\widetilde{x} - \widetilde{\lambda}\widetilde{x}\|}{\|\widetilde{x}\|} = \frac{\|r\|}{\|\widetilde{x}\|}.$$

Proof. See [95, §4.5]. □
Another way to prove Theorem 2.37, proposed in [69], is by considering the simple version of Theorem 2.35.

Theorem 2.38. *Let $A \in \mathbb{R}^{n\times n}$ be a symmetric matrix and let $\widetilde{x} \in \mathbb{R}^n$ be any unit vector, i.e., $\|\widetilde{x}\| = 1$. For any scalar $\widetilde{\lambda} \in \mathbb{R}$, define $r = A\widetilde{x} - \widetilde{\lambda}\widetilde{x}$ and $\mathcal{E} = \left\{E : (A + E)\widetilde{x} = \widetilde{\lambda}\widetilde{x}\right\}$. Then*

$$\min_{\mathcal{E}} \|E\| = \|r\|.$$

Proof. See [101, §2.3]. □

Remark 2.39. The value of $\widetilde{\lambda}$ which minimizes $\|r\|$, and $\|E\|$, is given by the Rayleigh quotient $\rho(\widetilde{x}) = \widetilde{x}^T A\widetilde{x}$ and the matrix E which achieves the minimum in Theorem 2.38 is

$$E = r\widetilde{x}^T + \widetilde{x}r^T,$$

where $r = A\widetilde{x} - \rho(\widetilde{x})\widetilde{x}$. See [69].

Next we introduce several perturbation bounds, see, e.g., [95].

Theorem 2.40. *Let $A \in \mathbb{R}^{n\times n}$ be a symmetric matrix, $\widetilde{x} \in \mathbb{R}^n$ a nonzero vector, $\widetilde{\lambda}$ any scalar and r the residual vector*

$$r = A\widetilde{x} - \widetilde{\lambda}\widetilde{x}.$$

Let λ be the eigenvalue of A closest to $\widetilde{\lambda}$ and x its normalized eigenvector, i.e., $\|x\|_2 = 1$. Moreover, following Definition 2.23, let the gap δ and the spread $spr(A)$ be given as

$$\delta = \min_{\lambda_i \neq \lambda} |\lambda_i - \widetilde{\lambda}| \quad \text{and} \quad spr(A) = \lambda_n - \lambda_1.$$

Then

$$|\widetilde{\lambda} - \lambda| \leq \frac{\|r\|_2}{\|\widetilde{x}\|_2},$$

$$\sin\angle(\widetilde{x}, x) \leq \frac{1}{\delta}\frac{\|r\|_2}{\|\widetilde{x}\|_2}.$$

Furthermore, if $\widetilde{\lambda} = \rho(\widetilde{x})$ then

$$0 \leq |\widetilde{\lambda} - \lambda| \leq \min\left(\frac{\|r\|_2}{\|\widetilde{x}\|_2}, \frac{1}{\delta}\frac{\|r\|_2^2}{\|\widetilde{x}\|_2^2}\right)$$

$$\frac{1}{spr(A)}\frac{\|r\|_2}{\|\widetilde{x}\|_2} \leq |\sin\angle(\widetilde{x}, x)| \leq \frac{1}{\delta}\frac{\|r\|_2}{\|\widetilde{x}\|_2}.$$

Proof. See Parlett [95, §11.7]. □

Apart from previously introduced eigenvalue error bounds, Theorem 2.40 contains error bounds for approximated eigenvectors, known as the *Davis and Kahan* $\sin\Theta$*-Theorem.*

Theorem 2.41 (The $\sin\Theta$-Theorem)**.** *Let $A \in \mathbb{R}^{n\times n}$ be a symmetric matrix, $\widetilde{x} \in \mathbb{R}^n$ a nonzero vector, $\widetilde{\lambda}$ any scalar and r the residual vector*

$$r = A\widetilde{x} - \widetilde{\lambda}\widetilde{x}.$$

Let λ be the eigenvalue of A closest to $\widetilde{\lambda}$ and x its normalized eigenvector, i.e., $\|x\|_2 = 1$. Moreover, let δ denote the gap, i.e., $\delta = \min_{\lambda_i \neq \lambda} |\lambda_i - \widetilde{\lambda}|$. Then

$$\sin\angle(\widetilde{x}, x) \leq \frac{\|r\|_2}{\delta}.$$

Proof. See [44, §6], [45, §6]. □

The eigenvector error bound depends on the size of the gap in the spectrum, i.e., the actual separation of eigenvalues, which is not known neither a priori nor a posteriori. However, in [101, Chapter III] Saad proposes a way to estimate δ, namely

$$\delta = |\lambda_i - \widetilde{\lambda}| \geq |\widetilde{\lambda} - \widetilde{\lambda}_i| - \|r_i\|_2. \tag{2.23}$$

Since we are expecting that, as the iteration proceeds, we get more accurate approximations of the eigenvalues, the gap estimate should also tend to the real value of δ.
In particular, Theorem 2.40 applied to the generalized eigenvalue problem, is one of the main tools used in forthcoming sections.

Theorem 2.42. *Let (λ, x) with $\|x\|_B = 1$ be an exact eigenpair of a real symmetric matrix pair (A, B) with positive definite matrix B, $(\widetilde{\lambda}, \widetilde{x})$ a corresponding computed eigenpair and r the residual vector*

$$r = A\widetilde{x} - \widetilde{\lambda}B\widetilde{x}.$$

Moreover, let the gap δ be given as in Definition 2.23. Then

$$|\widetilde{\lambda} - \lambda| \leq \frac{\|r\|_{B^{-1}}}{\|\widetilde{x}\|_B},$$

$$\sin_B \angle(\widetilde{x}, x) \leq \frac{1}{\delta}\frac{\|r\|_{B^{-1}}}{\|\widetilde{x}\|_B}.$$

Proof. Since the matrix B is assumed to be positive definite, we have the following relations

$$\begin{aligned} B^{-\frac{1}{2}}AB^{-\frac{1}{2}}z &= \lambda z, \\ \widetilde{r} &= B^{-\frac{1}{2}}AB^{-\frac{1}{2}}\widetilde{z} - \widetilde{\lambda}\widetilde{z}, \end{aligned} \tag{2.24}$$

where $z := B^{\frac{1}{2}}x$ and $\widetilde{z} := B^{\frac{1}{2}}\widetilde{x}$. Moreover,

$$\widetilde{r} = B^{-\frac{1}{2}}r, \tag{2.25}$$

$$\sin\angle(\widetilde{z}, z) = \sin\angle(B^{\frac{1}{2}}\widetilde{x}, B^{\frac{1}{2}}x) = \sin_B(\widetilde{x}, x). \tag{2.26}$$

Applying Theorem 2.40 to the standard eigenvalue problem (2.24) gives

$$|\widetilde{\lambda} - \lambda| \leq \frac{\|\widetilde{r}\|_2}{\|\widetilde{z}\|_2},$$

$$\sin\angle(\widetilde{z}, z) \leq \frac{1}{\delta}\frac{\|\widetilde{r}\|_2}{\|\widetilde{z}\|_2}.$$

Combining these inequalities with (2.25) and (2.26) completes the proof. □

2.3.3.2 The Non-symmetric Diagonalizable Case

Another step in the direction of error estimates for the general non-symmetric case is to first consider the non-symmetric, real diagonalizable case. The well-known *Bauer-Fike* Theorem states that the eigenvalue error can be bounded by the condition number of the eigenvector matrix and the norm of the perturbation matrix E. Obviously, this theorem can be viewed as a generalization of the already discussed Theorem 2.32.

Theorem 2.43 (Bauer-Fike)**.** *Let $A \in \mathbb{R}^{n\times n}$ be a diagonalizable matrix, and $Z \in R^{n\times n}$ the non-singular eigenvector matrix such that $A = ZDZ^{-1}$, where D is diagonal with eigenvalues of A on the diagonal. Moreover, assume that $\widetilde{\lambda}$ is a simple eigenvalue of the matrix $A + E$ and $\widetilde{z}$ the corresponding eigenvector, and $r = A\widetilde{z} - \widetilde{\lambda}\widetilde{z}$. Then the absolute distance between $\widetilde{\lambda}$ and a closest eigenvalue λ of A is bounded by*

$$|\lambda - \widetilde{\lambda}| \leq \kappa(Z)\|E\|_2,$$

or

$$|\lambda - \widetilde{\lambda}| \leq \kappa(Z)\frac{\|r\|_2}{\|\widetilde{z}\|_2}.$$

Proof. See [17, Theorem IIIa] or [101, Chapter III, §2]. □

Remark 2.44. If A is *normal*, i.e., $A^TA = AA^T$, then the Bauer-Fike Theorem reduces to

$$|\lambda - \widetilde{\lambda}| \leq \|E\|_2 \quad \text{or} \quad |\lambda - \widetilde{\lambda}| \leq \frac{\|r\|_2}{\|\widetilde{z}\|_2}.$$

In order to consider the perturbation bound for the eigenvector of the non-symmetric, real diagonalizable matrix we will use an extension of Davis and Kahan's $sin\Theta$-Theorem 2.41 introduced in [51, §3].

Theorem 2.45. *Let $(\widetilde{\lambda}, \widetilde{z})$ be an approximate eigenpair of a real diagonalizable matrix A, i.e., $A = ZDZ^{-1}$, where D is diagonal and $Z = [z\ Z_2]$, $Z^{-1} = \begin{bmatrix} w^T \\ W_2^T \end{bmatrix}$, are the right and the left eigenvector matrices, respectively. Moreover, assume that λ is the eigenvalue of A closest to $\widetilde{\lambda}$, and z, w are the corresponding right and left eigenvector, respectively. Furthermore, let $\|z\|_2 = 1$. Then for $r = A\widetilde{z} - \widetilde{\lambda}\widetilde{z}$ the bound for right eigenvectors is given by*

$$sin\angle(z, \widetilde{z}) \leq \kappa(W_2)\frac{\|r\|_2}{\delta},$$

where $\kappa(W_2) = \frac{\sigma_{max}(W_2)}{\sigma_{\min}(W_2)}$ and $\delta = \min_{\lambda_i \neq \lambda} |\lambda_i - \widetilde{\lambda}|$.

Proof. See Theorem 3.1 in [51, §3]. □

Theorem 2.46. *Let (A, B) be a matrix pair with $A \in \mathbb{R}^{n\times n}$ non-symmetric and $B \in \mathbb{R}^{n\times n}$ symmetric and positive definite. Let (λ, x, y) be the exact eigentriple of the generalized eigenvalue problem $Ax = \lambda Bx$ with $\|x\|_2 = 1$, $(\widetilde{\lambda}, \widetilde{x}, \widetilde{y})$ the eigentriple approximation and*

$r = A\widetilde{x} - \widetilde{\lambda}B\widetilde{x}$ the residual vector. Then if $B^{-1}A$ is a real diagonalizable matrix, i.e., $B^{-1}A = XDX^{-1}$, where D is diagonal and $X = [x\ X_2]$, $X^{-1} = \begin{bmatrix} y^TB \\ Y_2^TB \end{bmatrix}$, are the right and the left eigenvector matrices, respectively, the following bounds for the error in eigenvalues and eigenvectors hold

$$\begin{aligned} |\lambda - \widetilde{\lambda}| &\leq \kappa(B^{\frac{1}{2}}X)\frac{\|r\|_{B^{-1}}}{\|\widetilde{x}\|_B}, \\ \sin_B \angle(x, \widetilde{x}) &\leq \kappa(B^{\frac{1}{2}}Y_2)\frac{\|r\|_{B^{-1}}}{\delta}. \end{aligned}$$

where $\delta = \min_{\lambda_i \neq \lambda} |\lambda_i - \widetilde{\lambda}|$.

Proof. As $B^{-1}A$ is spectrally equivalent to $B^{-\frac{1}{2}}AB^{-\frac{1}{2}}$, the following eigenvalue problems are equivalent, i.e.,

$$\begin{aligned} B^{-1}Ax &= \lambda x, \\ B^{-\frac{1}{2}}AB^{-\frac{1}{2}}B^{\frac{1}{2}}x &= \lambda B^{\frac{1}{2}}x. \end{aligned} \tag{2.27}$$

Moreover, the matrix $B^{-\frac{1}{2}}AB^{-\frac{1}{2}}$ is real diagonalizable, i.e.,

$$B^{-\frac{1}{2}}AB^{-\frac{1}{2}} = ZDZ^{-1},$$

with $Z = [z\ Z_2] = B^{\frac{1}{2}}X = [B^{\frac{1}{2}}x, B^{\frac{1}{2}}X_2]$ and $Z^{-1} = \begin{bmatrix} w^T \\ W_2^T \end{bmatrix} = X^{-1}B^{-\frac{1}{2}} = \begin{bmatrix} y^TB^{\frac{1}{2}} \\ Y_2^TB^{\frac{1}{2}} \end{bmatrix}$.
By applying Theorem 2.43 and Theorem 2.45 to (2.27) with

$$\begin{aligned} \widetilde{A} &= B^{-\frac{1}{2}}AB^{-\frac{1}{2}}, \quad Z = B^{\frac{1}{2}}X, \quad Z^{-1} = X^{-1}B^{-\frac{1}{2}}, \quad W_2 = B^{\frac{1}{2}}Y_2, \quad z = B^{\frac{1}{2}}x, \quad w^T = y^TB^{\frac{1}{2}}, \\ \widetilde{r} &= \widetilde{A}\widetilde{z} - \widetilde{\lambda}\widetilde{z} = B^{-\frac{1}{2}}AB^{-\frac{1}{2}}B^{\frac{1}{2}}\widetilde{x} - \widetilde{\lambda}B^{\frac{1}{2}}\widetilde{x} = B^{-\frac{1}{2}}(A\widetilde{x} - \widetilde{\lambda}B\widetilde{x}) = B^{-\frac{1}{2}}r. \end{aligned}$$

it follows that

$$\begin{aligned} |\lambda - \widetilde{\lambda}| &\leq \kappa(Z)\frac{\|\widetilde{r}\|_2}{\|\widetilde{z}\|_2} = \kappa(B^{\frac{1}{2}}X)\frac{\|r\|_{B^{-1}}}{\|\widetilde{x}\|_B}, \\ \sin\angle(z, \widetilde{z}) &\leq \kappa(W_2)\frac{\|\widetilde{r}\|_2}{\delta} = \kappa(B^{\frac{1}{2}}Y_2)\frac{\|r\|_{B^{-1}}}{\delta}. \end{aligned}$$

Since $z = B^{\frac{1}{2}}x$ and $\widetilde{z} = B^{\frac{1}{2}}\widetilde{x}$

$$\sin\angle(z, \widetilde{z}) = \sin\angle(B^{\frac{1}{2}}x, B^{\frac{1}{2}}\widetilde{x}) = \sin\angle_B(x, \widetilde{x}),$$

which completes the proof. □

Although, at first glance, Theorem 2.46 looks very similar to the result for the symmetric case, it has much less impact in practice. In general, obtaining the condition number for the

eigenvector matrix is very hard, however, to get at least some information about the accuracy of the approximation, we use a by-product information obtained during the iteration process. As an example, let us consider the Arnoldi factorization (2.19).
The resulting Hessenberg (or tridiagonal) matrix can be real diagonalized with a matrix $U_m \in \mathbb{R}^{m\times m}$ such that

$$H_m = U_m D_m U_m^{-1},$$

where D_m is diagonal with eigenvalues of H_m on the diagonal. If the remainder term $f_m e_m^T$ in (2.19) is small (assuming that we are close to the invariant subspace), we can write that

$$SV_m \approx V_m U_m D_m U_m^{-1}.$$

Therefore, since

$$SV_m U_m \approx V_m U_m D_m,$$

the matrix $V_m U_m$ is a good approximation of the eigenvector matrix X_m of S. Using the *singular value decomposition* (SVD) of U_m, see Theorem 2.24, one can determine $\sigma_{max}(U_m), \sigma_{min}(U_m)$ the maximal and the minimal singular values of U_m, respectively, which can be used to approximate the eigenvector matrix condition number, see Definition 2.25, i.e.,

$$\kappa(X_m) \approx \frac{\sigma_{max}(U_m)}{\sigma_{min}(U_m)}. \tag{2.28}$$

This idea can be used to obtain condition numbers of both the left and the right eigenvector matrix.

Unfortunately there is no analogue of Theorem 2.46 for a general non-symmetric eigenvalue problem. Although, there exists a generalization of the Bauer-Fike Theorem, see [101, §2, Chapter III], it has no practical implications. Therefore, non-self-adjoint problems in general form are beyond the scope of this work.

2.4 Continuous-discrete inner product and norm relations

Moving from the infinite dimensional space V to some approximating subspace of finite dimension, i.e., V_h, means not only working with vectors and matrices, instead of functions and operators, but also creates the need for defining discrete equivalents for the underlying inner products and norms. As we see, appropriately defined inner products and norms reflect a natural correspondence between a function and its representation vector.
Since, as we mentioned in Section 2.2.2, $\mathbf{u}_h \in \mathbb{R}^{n_h}$, $\mathbf{v}_h \in \mathbb{R}^{n_h}$ represent functions u_h, v_h with respect to a basis $\left\{\varphi_1^h, \ldots . \varphi_{n_h}^h\right\}$, the following identities hold

$$a(u_h, v_h) = \mathbf{u}_h^T A_h \mathbf{v}_h = (\mathbf{u}_h, \mathbf{v}_h)_{A_h}, \quad \text{for all } u_h, v_h \in V_h, \tag{2.29}$$

$$b(u_h, v_h) = \mathbf{u}_h^T B_h \mathbf{v}_h = (\mathbf{u}_h, \mathbf{v}_h)_{B_h}, \quad \text{for all } u_h, v_h \in V_h. \tag{2.30}$$

Hence, if A_h, B_h are positive definite matrices, from (2.29), (2.30) the following relations follow.

$$\begin{aligned} |||u_h|||^2 &= a(u_h, u_h) = \mathbf{u}_h^T A_h \mathbf{u}_h = \|\mathbf{u}_h\|_{A_h}^2, \quad (2.31) \\ \|u_h\|^2 &= b(u_h, u_h) = \mathbf{u}_h^T B_h \mathbf{u}_h = \|\mathbf{u}_h\|_{B_h}^2. \quad (2.32) \end{aligned}$$

Let us define a discrete analogue of the energy and the $L_2(\Omega)$-norm in this setting via $\|\mathbf{u}_h\|_{A_h}$, $\|\mathbf{u}_h\|_{B_h}$, respectively. In practice we are interested in error estimates in the $H_0^1(\Omega)$-norm. Therefore, we identify $\|\mathbf{u}_h\|_{A_h+B_h}$ as a discrete equivalent of the $H_0^1(\Omega)$-norm, i.e.,

$$\begin{aligned} \|u_h\|_{H_0^1(\Omega)} &= \|u_h\|_{L_2(\Omega)} + \|\nabla u_h\|_{L_2(\Omega)} = \|u_h\|_{L_2(\Omega)} + |||u_h||| \\ &= b(u_h, u_h) + a(u_h, u_h) = \mathbf{u}_h^T B_h \mathbf{u}_h + \mathbf{u}_h^T A_h \mathbf{u}_h \\ &= \mathbf{u}_h^T (A_h + B_h) \mathbf{u}_h. \end{aligned}$$

As discussed in Section 2.2.3 the energy norm of the error is equivalent to the $H^{-1}(\Omega)$-norm of the corresponding residual vector. The following theorem follows from the definition of the dual norm (2.3).

Theorem 2.47. *Consider A_h, B_h to be symmetric, positive definite matrices. Let $u_h, v_h \in V_h$ with $\|v_h\|_{H^1(\Omega)} = 1$ and $\mathbf{u}_h, \mathbf{v}_h$ be their representation vectors with respect to a basis of V_h. Then*

$$\|u_h\|_{H^{-1}(\Omega)} = \|\mathbf{u}_h\|_{B_h(A_h+B_h)^{-1}B_h}.$$

Hence $B_h(A_h + B_h)^{-1}B_h$-norm can be identified as a discrete analogue of the $H^{-1}(\Omega)$-norm.

Proof. From the definition of the dual norm (2.3) we have

$$\|u_h\|_{H^{-1}(\Omega)} = \sup_{\substack{0 \neq v \in H^1(\Omega), \\ \|v_h\|_{H^1(\Omega)}=1}} |(u_h, v_h)_{L_2(\Omega)}| = \sup_{\mathbf{v}_h^T B_h \mathbf{v}_h + \mathbf{v}_h^T A_h \mathbf{v}_h = 1} \mathbf{v}_h^T B_h \mathbf{u}_h. \quad (2.33)$$

Therefore, (2.33) can be viewed as the corresponding optimization problem

$$\max \ \mathbf{v}_h^T B_h \mathbf{u}_h \quad \text{subject to} \quad \mathbf{v}_h^T B_h \mathbf{v}_h + \mathbf{v}_h^T A_h \mathbf{v}_h = 1.$$

With help of the method of *Lagrange multipliers* [40] the Lagrange function

$$\max \ \Lambda(\mathbf{v}_h, \xi) = \mathbf{v}_h^T B_h \mathbf{u}_h + \xi \mathbf{v}_h^T (A_h + B_h) \mathbf{v}_h$$

is studied. From the necessary optimality condition

$$\nabla_{\mathbf{v}_h} \Lambda(\mathbf{v}_h, \xi) = 0,$$

we get

$$B_h \mathbf{u}_h + \frac{1}{2} \xi (A_h + B_h) \mathbf{v}_h = 0,$$

and

$$\mathbf{v}_h = \frac{2}{\xi} (A_h + B_h)^{-1} B_h \mathbf{u}_h. \quad (2.34)$$

Since $\|\mathbf{v}_h\|_{H^1(\Omega)} = 1$ we obtain

$$\begin{aligned} 1 &= \mathbf{v}_h^T(A_h + B_h)\mathbf{v}_h = \frac{2}{\xi}\mathbf{u}_h^T B_h^T (A_h^T + B_h^T)^{-1}(A_h + B_h)\frac{2}{\xi}(A_h + B_h)^{-1}B_h\mathbf{u}_h \\ &= \frac{4}{\xi^2}\mathbf{u}_h^T B_h (A_h + B_h)^{-1}B_h\mathbf{u}_h, \end{aligned}$$

and finally

$$\begin{aligned} \xi &= 2\sqrt{\mathbf{u}_h^T B_h (A_h + B_h)^{-1}B_h\mathbf{u}_h} = 2\|B_h\mathbf{u}_h\|_{(A_h+B_h)^{-1}} \\ &= 2\|\mathbf{u}_h\|_{B_h(A_h+B_h)^{-1}B_h}. \end{aligned} \tag{2.35}$$

By inserting (2.34) and (2.35) into (2.33) we get

$$\begin{aligned} \|u_H\|_{H^{-1}} = \mathbf{v}_h^T B_h\mathbf{u}_h &= \frac{2}{\xi}\mathbf{u}_h^T(A_h + B_h)^{-1}B_h\mathbf{u}_h = \frac{\mathbf{u}_h^T B_h(A_h + B_h)^{-1}B_h\mathbf{u}_h}{\|\mathbf{u}_h\|_{B_h(A_h+B_h)^{-1}B_h}} \\ &= \|\mathbf{u}_h\|_{B_h(A_h+B_h)^{-1}B_h}. \end{aligned}$$

□

Theorem 2.48. *Consider A_h, B_h to be symmetric, positive definite stiffness and mass matrices obtained from the finite element discretization of an elliptic second-order self-adjoint operator on the Sobolev space $H^m(\Omega)$. Then a discrete analogue of the H^m-norm is given by*

$$\mathbf{H}^m := ((A_h + B_h)B_h^{-1})^m B_h.$$

Proof. See, e.g., [28]. □

Remark 2.49. By applying Theorem 2.48 for $m = 0, 1, -1$ we get

$m =$	0	$\mathbf{H}^0$	$=$	B_h	discrete $L_2(\Omega)$-norm,
$m =$	1	$\mathbf{H}$	$=$	$A_h + B_h$	discrete $H^1(\Omega)$-norm,
$m =$	-1	$\mathbf{H^{-1}}$	$=$	$B_h(A_h + B_h)^{-1}B_h$	discrete $H^{-1}(\Omega)$-norm.

Furthermore, one can easily exploit the correspondence between inner products (2.29) and (2.30) to obtain the relation for angles between functions and their representation vectors in V_h.

Theorem 2.50. *Let u, v be two functions in the finite-dimensional space V with the inner product $(\cdot,\cdot)_V$ and the norm $\|\cdot\|_V$. Moreover, we denote by $\boldsymbol{u} = [u_i]_{i=1,\dots,dimV}$, $\boldsymbol{v} = [v_j]_{j=1,\dots,dimV}$ their representation vectors with respect to a basis $\{\varphi_1, \dots, \varphi_{dimV}\}$ of V. Then*

$$\cos_V \angle(u, v) = \cos \angle_{\mathbf{V}}(\mathbf{u}, \mathbf{v}), \quad \sin_V \angle(u, v) = \sin \angle_{\mathbf{V}}(\mathbf{u}, \mathbf{v}),$$

where $\mathbf{V} = [v_{ij}]_{i,j=1,\dots,dimV}$, with $v_{ij} = (\varphi_i, \varphi_j)_V$ and $\cos \angle_{\mathbf{V}}(\mathbf{u}, \mathbf{v})$ is defined as

$$\cos \angle_{\mathbf{V}}(\mathbf{u}, \mathbf{v}) := \frac{(\mathbf{u}, \mathbf{v})_{\mathbf{V}}}{\|\mathbf{u}\|_{\mathbf{V}}\|\mathbf{v}\|_{\mathbf{V}}} = \frac{\mathbf{u}^T\mathbf{V}\mathbf{v}}{\sqrt{\mathbf{u}^T\mathbf{V}\mathbf{u}}\sqrt{\mathbf{v}^T\mathbf{V}\mathbf{v}}}.$$

Proof. Definition 2.9 of the angle between u and v states that

$$\cos_V \angle(u,v) = \frac{(u,v)_V}{\|u\|_V \|v\|_V}.$$

Since both u, v are elements of V they can be expressed as linear combinations of basis functions, i.e.,

$$u = \sum_{i=1}^{dimV} u_i \varphi_i, \quad v = \sum_{j=1}^{dimV} v_j \varphi_j.$$

This implies

$$\cos_V \angle(u,v) = \frac{(u,v)_V}{\|u\|_V \|v\|_V} = \frac{(\sum_{i=1}^{dimV} u_i\varphi_i, \sum_{j=1}^{dimV} v_j\varphi_j)}{\sqrt{(\sum_{i=1}^{dimV} u_i\varphi_i, \sum_{i=j}^{dimV} u_j\varphi_j)}\sqrt{(\sum_{i=1}^{dimV} v_i\varphi_i, \sum_{j=1}^{dimV} v_j\varphi_j)}},$$

hence

$$\begin{aligned}\cos_V \angle(u,v) &= \frac{\sum_{i=1}^{dimV}\sum_{j=1}^{dimV} u_i(\varphi_i,\varphi_j)_V v_j}{\sqrt{\sum_{i=1}^{dimV}\sum_{j=1}^{dimV} u_i(\varphi_i,\varphi_j)_V u_j}\sqrt{\sum_{i=1}^{dimV}\sum_{j=1}^{dimV} v_i(\varphi_i,\varphi_j)_V v_j}} \\ &= \frac{\mathbf{u}^T\mathbf{V}\mathbf{v}}{\sqrt{\mathbf{u}^T\mathbf{V}\mathbf{u}}\sqrt{\mathbf{v}^T\mathbf{V}\mathbf{v}}} = \frac{(\mathbf{u},\mathbf{v})_\mathbf{V}}{\|\mathbf{u}\|_\mathbf{V}\|\mathbf{v}\|_\mathbf{V}} = \cos_\mathbf{V} \angle(\mathbf{u},\mathbf{v}).\end{aligned}$$

Of course, relation

$$\sin_V \angle(u,v) = \sin \angle_\mathbf{V}(\mathbf{u},\mathbf{v})$$

follows immediately. □

Remark 2.51. From Theorem 2.50 it can be easily seen that

$$\cos_{L_2(\Omega)} \angle(u_h, v_h) = \cos_{B_h} \angle(\mathbf{u}_h, \mathbf{v}_h), \tag{2.36}$$

where

$$\cos \angle_{B_h}(\mathbf{u},\mathbf{v}) := \frac{(\mathbf{u}_h,\mathbf{v}_h)_{B_h}}{\|\mathbf{u}\|_{B_h}\|\mathbf{v}\|_{B_h}}.$$

Chapter 3

Model problems

In this chapter, we introduce two model problems which we use to demonstrate the essence of methods discussed in subsequent chapters. We apply the theory presented in Sections 2.2 and 2.3 to a *Laplace* and a *convection-diffusion* eigenvalue problem to understand some of the major properties and difficulties of self-adjoint and non-self-adjoint eigenvalue problems. Based on the variational formulation we investigate the corresponding sesquilinear (bilinear) forms and define appropriate norms. For the Laplace eigenvalue problem we present a general relation between continuous and discrete eigenpairs, which follows from the Courant-Fischer minimax theorem [46, 107]. In the case of non-self-adjoint problems, such as the *convection-diffusion* eigenvalue problem, very few information about a general spectrum are available. Therefore, we concentrate on examples with particularly chosen convection and diffusion coefficients or defined on the specific domain $\Omega \subset \mathbb{R}^2$.

3.1 A Laplace eigenvalue problem

As a simple example of a self-adjoint eigenvalue problem we consider the *Laplace eigenvalue problem*:
Determine a non-trivial eigenpair $(\lambda, u) \in \mathbb{R} \times H_0^1(\Omega)$ *with* $\|u\|_{L_2(\Omega)} = 1$ *such that*

$$\begin{aligned} -\Delta u &= \lambda u, \qquad \text{on } \Omega \\ u &= 0, \qquad \text{in } \partial\Omega \end{aligned} \tag{3.1}$$

and its variational formulation of the form:
Determine a non-trivial eigenpair $(\lambda, u) \in \mathbb{R} \times V$, *with* $b(u,u) = 1$ *such that*

$$a(u,v) = \lambda b(u,v) \quad \text{for all } v \in V, \tag{3.2}$$

with

$$a(u,v) := \int_\Omega \nabla u \nabla v \, dx, \qquad b(u,v) := \int_\Omega uv \, dx.$$

It is known, see, e.g., [12], that the Laplace problem has a countable set of real eigenvalues

$$0 < \lambda_1 \leq \lambda_2 \leq \dots$$

and corresponding eigenfunctions

$$u_1, u_2, \dots, \quad \text{such that} \quad b(u_i, u_j) = \delta_{i,j}, \quad i, j = 1, \dots.$$

Here $V := H_0^1(\Omega)$ and $H := L_2(\Omega)$, which has the following consequences. The bilinear form $a(\cdot,\cdot)$ is symmetric, continuous and coercive in $H_0^1(\Omega)$. Therefore, the energy norm $|||\cdot||| := a(\cdot,\cdot)^{\frac{1}{2}}$ is a semi-norm on $V := H_0^1(\Omega)$, i.e., $|||\cdot||| := |\cdot|_{H^1(\Omega,\mathbb{C})}$, and is equivalent to the standard $H^1(\Omega)$-norm, i.e.,

$$c\|u\|^2_{H^1(\Omega)} \leq a(u,u) \leq C\|u\|^2_{H^1(\Omega)},$$

where c, C are the coercivity and the continuity constant, respectively.
The bilinear form $b(\cdot,\cdot)$ is continuous, symmetric and positive definite and hence induces a norm on $L_2(\Omega)$, in this case, $b(\cdot,\cdot)$ turns out to be the standard inner product on $L_2(\Omega)$, i.e., $b(\cdot,\cdot) := (\cdot,\cdot)_{L_2(\Omega)}$ and $\|\cdot\| := \|\cdot\|_{L_2(\Omega)}$.
The *Ritz-Galerkin* discretization of the Laplace problem on the $\mathbb{P}_1$ finite element space V_h leads to the variational formulation

$$a(u_h, v_h) = \lambda_h(u_h, v_h) \quad \text{for all } v_h \in V_h. \tag{3.3}$$

and the associated generalized algebraic eigenvalue problem of the form

$$A_h \mathbf{u}_h = \lambda_h B_h \mathbf{u}_h, \tag{3.4}$$

where A_h and B_h are symmetric and positive definite matrices of dimension n_h.
This generalized symmetric eigenvalue problem has a finite set of eigenvalues

$$0 < \lambda_{1,h} \leq \lambda_{2,h} \leq \dots \leq \lambda_{n_h,h}$$

and corresponding eigenvectors

$$\mathbf{u}_{1,h}, \mathbf{u}_{2,h}, \dots, \mathbf{u}_{n_h,h} \quad \text{such that} \quad \mathbf{u}_{i,h}^T B_h \mathbf{u}_{j,h} = \delta_{i,j}, \quad i, j = 1, \dots, n_h.$$

It follows from the *Courant-Fischer minimax* theorem [46, 107] that

$$\lambda_i \leq \lambda_{i,h} \qquad \text{for all } i = 1, \dots, n_h,$$

and if $\mathcal{T}_h$ is any refinement of $\mathcal{T}_H$, i.e., $H > h$, then

$$0 \leq \lambda_{i,h} \leq \lambda_{i,H} \quad \text{for all } i = 1, \dots, n_H.$$

3.2 A convection-diffusion eigenvalue problem

Unfortunately, as we pointed out at the beginning, many practical problems go beyond the self-adjoint case [66]. From the spectral theory of non-normal operators we know that their eigenfunctions are in general non-orthogonal and the corresponding eigenvalues may be complex and badly conditioned [71, 109]. Moreover, in the non-self-adjoint case no analogue of the Courant-Fischer minimax theorem is known, which makes it impossible to uniquely assign the continuous eigenvalues to the discrete ones.
In order to make our investigations easier to follow we will consider a two-dimensional *convection-diffusion* eigenvalue problem of a general form:
Find a non-trivial pair (λ, u) *such that*

$$\begin{aligned} -\nu\Delta u + \beta \cdot \nabla u &= \lambda u \quad \text{in } \Omega \\ u &= 0 \quad \text{on } \partial\Omega, \end{aligned} \tag{3.5}$$

where $\nu \in \mathbb{R}$ *and* $\beta \in \mathbb{R}^2$ *are the diffusion and the convection coefficient, respectively.*
For $\beta \neq [0,0]^T$ the operator is non-self-adjoint and its departure from normality depends on the size of the diffusion coefficient ν relative to $\|\beta\|$ [66].
In Chapter 5 we investigate the model convection-diffusion problem, with constant convection coefficient β of the form:
Find a non-trivial eigenpair $(\lambda, u) \in \mathbb{C} \times H_0^1(\Omega) \cap H^2(\Omega)$ *with* $\|u\|_{L_2(\Omega)} = 1$ *such that*

$$\begin{aligned} -\Delta u + \beta \cdot \nabla u &= \lambda u \text{ in } \Omega \\ u &= 0 \text{ on } \partial\Omega, \end{aligned} \tag{3.6}$$

where $\beta \in \mathbb{R}^2$ *is divergence free, i.e.,*

$$\int_\Omega v\, div(\beta) dx = 0, \text{ for all } v \in H_0^1(\Omega).$$

Since the convection-diffusion operator is non-self-adjoint it is necessary to consider both, the *primal* and the *dual* eigenvalue problem. Therefore, we study the problem (3.6) in its weak formulation.
Determine a non-trivial primal and dual eigenpair $(\lambda, u) \in \mathbb{C} \times V$ *and* $(\lambda^\star, u^\star) \in \mathbb{C} \times V$*, with* $b(u,u) = 1$ *such that*

$$a(u,v) + c(u,v) = \lambda b(u,v) \quad \text{for all } \ v, \in V$$

and

$$a(w,u^\star) + c(w,u^\star) = \overline{\lambda^\star} b(w,u^\star) \quad \text{for all } \ w \in V,$$

where

$$a(u,v) := \int_\Omega \nabla u \nabla \overline{v} dx, \quad c(u,v) := \int_\Omega \beta \cdot \nabla u \overline{v} dx, \quad b(u,v) := \int_\Omega u \overline{v} dx,$$

and $\overline{(.)}$ *denotes the complex conjugate.*
Note that primal and dual eigenvalues are connected by $\lambda = \overline{\lambda^\star}$. The corresponding spaces are defined as follows: $V := H_0^1(\Omega,\ H := L_2(\Omega)$. Due to the choice of β to be divergence free the bilinear form $a(\cdot,\cdot) + c(\cdot,\cdot)$ is continuous and elliptic in V. The bilinear form $b(\cdot,\cdot)$ is continuous, symmetric and positive definite, and hence induces a norm $\|\cdot\| = \|\cdot\|_{L_2(\Omega)}$. Obviously $a(v,v) + c(v,v)$ is no longer symmetric and positive definite, therefore, for this example the energy norm is defined as $|||\cdot||| = a(\cdot,\cdot)^{\frac{1}{2}}$.
The *Ritz-Galerkin* discretization on $V_h \subset V$ leads to a variational formulation:
Determine the non-trivial primal and dual eigenpairs $(\lambda_h, u_h) \in \mathbb{C}\times V_h$ *and* $(\lambda_h^\star, u_h^\star) \in \mathbb{C}\times V_h$ *such that*

$$\begin{aligned} a(u_h, v_h) + c(u_h, v_h) &= \lambda_h b(u_h, v_h) \qquad \textit{for all } v_h \in V_h, \\ a(w_h, u_h^\star) + c(w_h, u_h^\star) &= \overline{\lambda_h^\star} b(w_h, u_h^\star) \qquad \textit{for all } w_h \in V_h. \end{aligned} \tag{3.7}$$

Since primal and dual eigenvalues are connected by $\lambda = \overline{\lambda^\star}$, we are interested in finding a non-trivial triple $(\lambda_h, u_h, u_h^\star)$, referred to as an *eigentriple*.
In summary, from (3.7) we get the two generalized algebraic eigenvalue problems, i.e.,

$$(A_h + C_h)\mathbf{u}_h = \lambda_h B_h \mathbf{u}_h, \qquad\qquad \mathbf{u}_h^\star(A_h + C_h) = \lambda_h^\star \mathbf{u}_h^\star B_h,$$

where A_h is the symmetric positive definite stiffness matrix, C_h the non-symmetric convection matrix and B_h the symmetric positive definite mass matrix. As a consequence, the eigentriple $(\lambda_h, u_h, u_h^\star)$ is determined based on the discrete eigentriple $(\lambda_h, \mathbf{u}_h, \mathbf{u}_h^\star)$, where $\mathbf{u}_h, \mathbf{u}_h^\star$ are the right and the left generalized eigenvectors of the matrix pair $(A_h + C_h, B_h)$, respectively. Generalized algebraic eigenvalue problems are discussed in details in Section 2.3, however, it is important to mention that the smallest eigenvalue (the eigenvalue with the smallest real part) of this problem is proved to be simple (real) and well separated [54].
Moreover, if the problem (3.5) is defined in the unit square, i.e., $\Omega := [0,1] \times [0,1]$ and the convection coefficient is constant, i.e., $\beta = [\beta_1, \beta_2]^T$, its eigenvalues and primal (dual) eigenfunctions are known explicitly, see [66],

$$\frac{\beta_1^2 + \beta_2^2}{4\nu} + \nu\pi^2(k_1^2 + k_2^2), \quad \pm \exp\frac{\beta_1 x + \beta_2 y}{2\nu} \sin(k_1\pi x)\sin(k_2\pi y), \quad k_1, k_2 \in \mathbb{N}\backslash\{0\}.$$

Chapter 4

Self-adjoint eigenvalue problem

In this chapter we describe new ideas of combining the information from the algebraic eigensolver with the standard ESTIMATE procedure of the adaptive finite element method. As an introduction, in Section 4.1.1, we show empirically how misleading the standard assumption about the exactness of the approximate solution is even in the case of a simple, one-dimensional Laplace eigenvalue problem. We present a comparison of the discretization and the iteration error for both eigenvalues and eigenfunctions. In Section 4.2 we introduce an extended approach, called AFEMLA, for the adaptive finite element solution of self-adjoint elliptic PDE eigenvalue problems that incorporates the solution (in finite precision arithmetic) of the algebraic problem into the adaptation process. An advantage of this new algorithm is that it can be used also when the problem comes in discretized form, e.g. directly from the finite element modeling. The robustness is confirmed both by experiments and theory. The results presented in this section are published in [91]. Throughout Section 4.3 the functional backward error and condition number for the self-adjoint eigenvalue problem are derived. A combined a posteriori error estimator for errors measured in the $H^1(\Omega)$-norm, which balances the influence of the discretization and the iteration error on the adaptive method, is introduced in Section 4.4. All these techniques are analyzed in terms of their possible extensions for more general problems.

4.1 A comparison of discretization and iteration errors for self-adjoint eigenvalue problems

In [110] the relationship between the discretization and the iteration error in the iterative solution was studied in the context of solving elliptic boundary value problems. Numerical examples confirmed that the influence of the error in the algebraic solver compared to the precision of the finite element approximation cannot be ignored and has to be considered. Here we derive a similar analysis for elliptic self-adjoint eigenvalue problems. For illustration we first investigate analytically the behavior of the discretization and iteration error for eigenvalues and eigenfunctions (eigenvectors) of the one-dimensional Laplace eigenvalue problem.

4.1.1 A model problem and error estimates

Let $d = 1$ and $\Omega = (0, 1)$. Then the problem (3.1) has the form

$$\begin{aligned} -u'' &= \lambda u \qquad \text{on } \Omega = (0,1) \\ u(0) &= u(1) = 0. \end{aligned} \tag{4.1}$$

The exact k-th eigenpair of (4.1) is given by

$$(\lambda_k, u_k(x)) = (k^2\pi^2, \sqrt{2}\sin(k\pi x)), \quad k = 1, \dots .$$

The corresponding variational formulation is given by

$$a(u, v) = \lambda b(u, v),$$

with

$$a(u,v) = \int_0^1 u'v'dx, \quad b(u,v) = \int_0^1 uvdx.$$

The problem is discretized with the *Ritz-Galerkin* method on the uniform mesh of size $h = \frac{1}{n+1}$ with $\mathbb{P}_1$ finite elements and inner nodes $x_i = ih$, for $i = 1, \dots, n$. The finite element approximation of (λ, u) is obtained as $(\lambda_h, u_h) \in \mathbb{R} \times V_h$ such that

$$a(u_h, v_h) = \lambda b(u_h, v_h) \text{ for all } v_h \in V_h. \tag{4.2}$$

Following [11, Chapter I, §5, pp. 677] the discrete problem can be written as the generalized matrix eigenvalue problem of the form

$$A_h \mathbf{u}_h = \lambda_h B_h \mathbf{u}_h, \tag{4.3}$$

where

$$A_h = \frac{1}{h}\begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & -1 & 2 & -1 & \\ & & & \ddots & -1 \\ & & & -1 & 2 \end{bmatrix} \in \mathbb{R}^{n\times n}, \quad B_h = \frac{1}{6}h\begin{bmatrix} 4 & 1 & & & \\ 1 & 4 & 1 & & \\ & 1 & 4 & 1 & \\ & & & \ddots & 1 \\ & & & 1 & 4 \end{bmatrix} \in \mathbb{R}^{n\times n}.$$

The k-th exact discrete eigenvalue of (4.3) is given by

$$\lambda_{k,h} = \frac{6}{h^2}\frac{1-\cos(k\pi h)}{2+\cos(k\pi h)}, \quad k = 1, \dots, n.$$

Since, just as for the boundary value problem [110], the exact discrete solution $u_{k,h}$ of (4.3) is equal to the exact solution u_k in mesh points, the following identity holds

$$\mathbf{u}_{k,h,i} = u_{k,h}(x_i) = u_k(x_i) = \sqrt{2}\sin(k\pi x_i) = \sqrt{2}\sin(k\pi ih), \quad k, i = 1, \dots, n. \tag{4.4}$$

In order to analyze the behavior of discretization and iteration errors we are interested in the following quantities:

	discretization error	**iteration error**
eigenvalue	$\lvert\lambda_k - \lambda_{k,h}\rvert$	$\lvert\lambda_{k,h} - \widetilde{\lambda}_{k,h}\rvert$
eigenfunction (eigenvector) $\lVert\!\lVert\cdot\rVert\!\rVert$-norm	$\lVert\!\lVert u_k - u_{k,h}\rVert\!\rVert$	$\lVert\mathbf{u}_{k,h} - \widetilde{\mathbf{u}}_{k,h}\rVert_{A_h}$
eigenfunction (eigenvector) $\sin\Theta$	$\sin\angle(u_k, u_{k,h})$	$\sin_{B_h}\angle(\mathbf{u}_{k,h}, \widetilde{\mathbf{u}}_{k,h})$

Table 4.1

From a priori error estimates for eigenvalues, i.e., [11, Chapter I, §5, pp. 678], it is well known that

$$|\lambda_k - \lambda_{k,h}| = |k^2\pi^2 - \frac{6}{h^2}\frac{1-\cos(k\pi h)}{2+\cos(k\pi h)}| = \mathcal{O}(h^2). \tag{4.5}$$

The eigenvalue iteration error $|\lambda_{k,h} - \widetilde{\lambda}_{k,h}|$ is computed based on the approximate eigenpair obtained from (4.3) by the Arnoldi/Lanczos method, i.e., the MATLAB function *eigs* [89] or ARPACK [81].

For simplicity, we consider in detail the case of $k = 1$ and we skip the index k in (λ_k, u_k), $(\lambda_{k,h}, u_{k,h})$ etc.. Let us analyze the discretization error for the eigenfunction using two different approaches. First we compute the discretization error using the energy norm $\lVert\!\lVert\cdot\rVert\!\rVert$.

Proposition 4.1. *Let $e = u - u_h$ define the eigenfunction discretization error with $u(x) = \sqrt{2}\sin(\pi x)$. Then*

$$\lVert\!\lVert e\rVert\!\rVert^2 = \pi^2 + \frac{2(\cos(\pi h) - 1)}{h^2}.$$

Moreover, $\lVert\!\lVert e\rVert\!\rVert = \mathcal{O}(h)$.

Proof. See Appendix 7.1.

Since for every eigenfunction u any nonzero multiple cu, $c \in \mathbb{R}$, of u is also an eigenfunction of (4.1), neither $\lVert\!\lVert u - u_h\rVert\!\rVert$ nor $\lVert\mathbf{u}_h - \widetilde{\mathbf{u}}_h\rVert_{A_h}$ are good measures of the error [61, 95]. In order to deal with this problem the idea of angles between subspaces spanned by vectors u and u_h (or $\widetilde{u}_h$) is exploited. These subspaces are uniquely determined, so the angle between them determines a good measure of the error.

Proposition 4.2. *Let u and u_h be exact eigenfunctions of problems (4.1), (4.2), respectively. Then*

$$\sin\angle(u, u_h) = \sqrt{\frac{\pi^4\cos(\pi h) + 2\pi^4 - 12|\frac{(\cos(\pi h)-1)^2}{h^4}|}{\pi^4(\cos(\pi h) + 2)}}.$$

Proof. See Appendix 7.1.

Now, based on (2.31) the energy norm of the iteration error $e_h = u_h - \widetilde{u}_h$ is computed by

$$\|\!|e_h|\!\|^2 = \|\!|u_h - \widetilde{u}_h|\!\|^2 = (\mathbf{u}_h - \widetilde{\mathbf{u}}_h)^T A_h (\mathbf{u}_h - \widetilde{\mathbf{u}}_h).$$

Obviously,

$$\sin \angle(u_h, \widetilde{u}_h) = \sin_{B_h} \angle(\mathbf{u}_h, \widetilde{\mathbf{u}}_h) = \sqrt{1 - \cos^2_{B_h} \angle(\mathbf{u}_h, \widetilde{\mathbf{u}}_h)} = \sqrt{1 - \frac{|(\mathbf{u}_h, \widetilde{\mathbf{u}}_h)_{B_h}|^2}{\|\mathbf{u}_h\|^2_{B_h} \|\widetilde{\mathbf{u}}_h\|^2_{B_h}}}.$$

4.1.2 Numerical examples - How exact the 'exact' really is?

Let us now illustrate previously derived results numerically. Figure 4.1 presents the discretization error based on equation (4.5). The discretization error decreases as the mesh size h is getting smaller and a correct order $\mathcal{O}(h^2)$ is recovered. The corresponding result with respect to the number of degrees of freedom is illustrated in Figure 4.2.

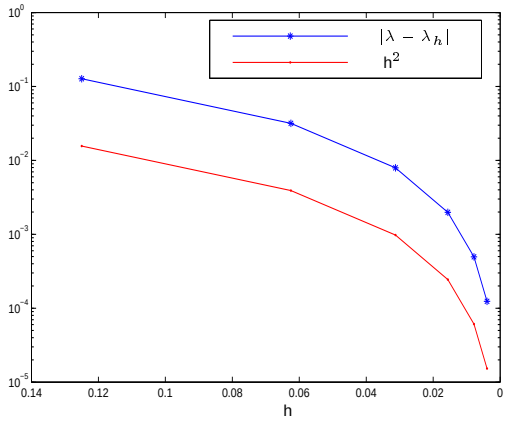


Figure 4.1: Eigenvalue discretization error versus mesh size h.

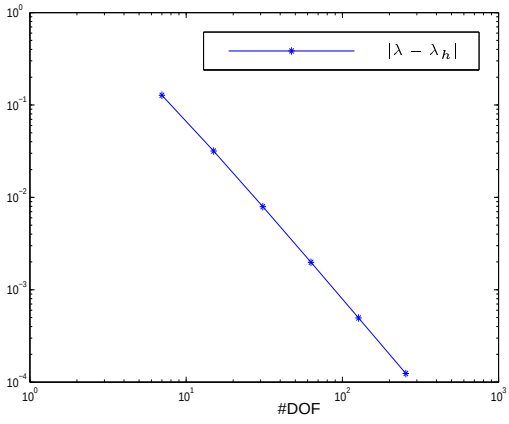


Figure 4.2: Eigenvalue discretization error versus #DOF.

In contrast to the discretization error, the eigenvalue iteration error increases as the mesh size h tends to zero as it is shown in Figures 4.3 and 4.4. At this point the question arises: Can it ever happen that the iteration error will dominate the discretization error. Unfortunately for the standard AFEM algorithms, the answer is "Yes". This phenomenon is illustrated in Figure 4.5. The experiments show that, even for the simple one-dimensional Laplace eigenvalue problem with symmetric positive definite, not badly conditioned matrices, the iteration error starts to dominate the discretization error already for a medium size problem of $\sim 10^4$ degrees of freedom. According to Proposition 4.1 the eigenfunction discretization error in the energy norm $\|\!|\cdot|\!\|$ is of order $\mathcal{O}(h)$ as shown in Figure 4.6.
Figure 4.8 indicates that the iteration error measured in the energy norm $\|\!|\cdot|\!\|$ is of order $\mathcal{O}(h^2)$. Looking at Figure 4.10 the results seem to be good, the iteration error is always below the discretization error.

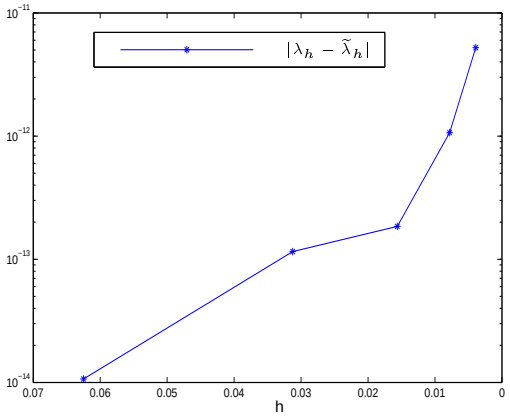


Figure 4.3: Eigenvalue iteration error versus mesh size h.

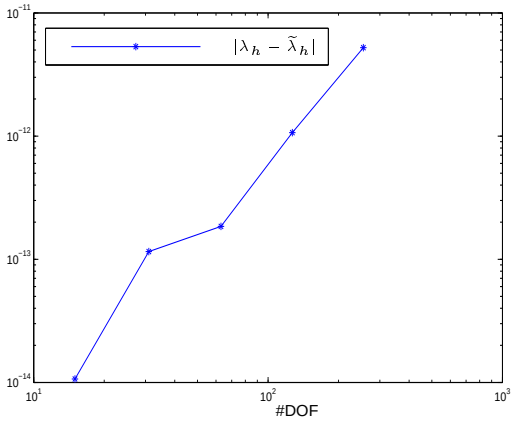


Figure 4.4: Eigenvalue iteration error versus #DOF.

However, as we have mentioned before, it is more reasonable to look at angles between subspaces in order to draw right conclusions. For the discretization error measured in terms of the angle between eigenspaces, see Figure 4.11 and 4.12, we recover the $\mathcal{O}(h^2)$ approximation which is better than the one obtained in the energy norm.
A slightly different behavior can be observed for the iteration error which increases while the mesh size h tends to zero, see Figure 4.13 and 4.14. All eigenfunction/eigenvector errors are compared in Figure 4.15. Although we do not observe the domination of the discretization error over the iteration error for the case of angles between subspaces (the results are just very good), it does not mean that a similar situation as for the eigenvalues may not occur as the mesh size h tends to zero.
The simple example presented above confirms that the typical assumption made for AFEM algorithms that matrix eigenvalue problems are solved exactly, so that the iteration error is always dominated by the discretization error, are not reasonable and may lead to incorrect results.

4.2 AFEMLA - two way adaptation based on the iteration error

In the previous section we have demonstrated that it is essential to consider the influence of iteration errors on AFEM algorithms. These errors have to be taken into account, not only in the process of designing reliable and efficient a posteriori error estimators, but also should be balanced with other errors, i.e., modeling or discretization errors, in order to define stopping criteria for algebraic eigensolvers. In this section we introduce an extended approach for the adaptive finite element solution of self-adjoint elliptic PDE eigenvalue problems that incorporates the solution (in finite precision arithmetic) of the algebraic problem into the adaptation process. In order to stress this fact we call the new approach AFEMLA, where

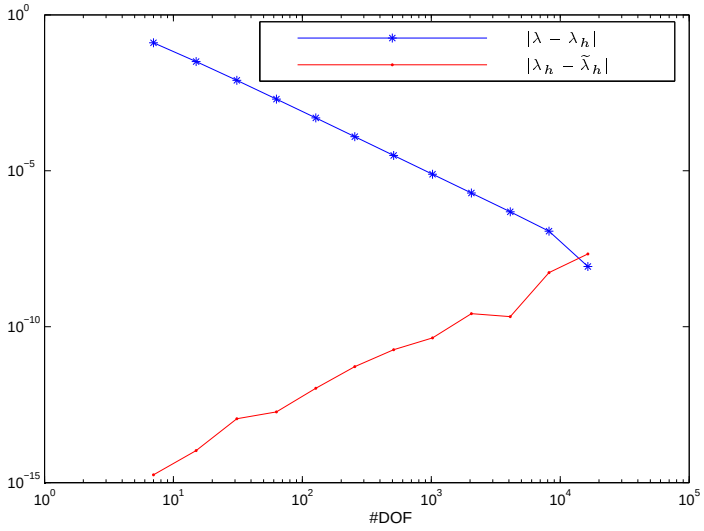


Figure 4.5: Eigenvalue discretization and iteration error versus #DOF.

letters 'LA' indicate that the adaptation also involves the numerical Linear Algebra part of the process. We will focus on the computation of the smallest real eigenvalues of the self-adjoint elliptic PDE eigenvalue problem.

4.2.1 Standard AFEM versus AFEMLA

The standard AFEM approach for eigenvalue problems is based on discretizing the variational formulation using the Ritz-Galerkin method on a given grid, i.e., Ω_{h_1} and solving the resulting generalized algebraic eigenvalue problem by an iterative solver. Based on this trial solution, elementwise a posteriori error estimates are determined and used to refine the grid. The same strategy proceeds for the next AFEM loop step, see Figure 4.16. This typical approach does not consider any influence of errors in the numerical linear algebra on the algorithm. Furthermore, since one may have to solve many algebraic eigenvalue problems related to finer and finer grids and information from the previous steps of the adaptive procedure, like previously well approximated eigenvalues, is not used on the next level, computational costs for the algebraic eigenvalue problem can dominate the total computational cost.

We introduce an adaptive finite element algorithm called AFEMLA which incorporates the information obtained during the iterative solving of algebraic eigenvalue problems into the error estimation and the refinement process. Since the accuracy of the computed eigenvalue approximation cannot be better than the quality of the discretization, there is no need to solve the algebraic eigenvalue problem up to very high precision if the discretization scheme guarantees only small precision. Also nested iterations, i.e., using actual eigenvector approximation as a starting vector for the eigensolver on the refined grid, reduce the total cost significantly. The idea of adaptive methods is to achieve a desired accuracy with the

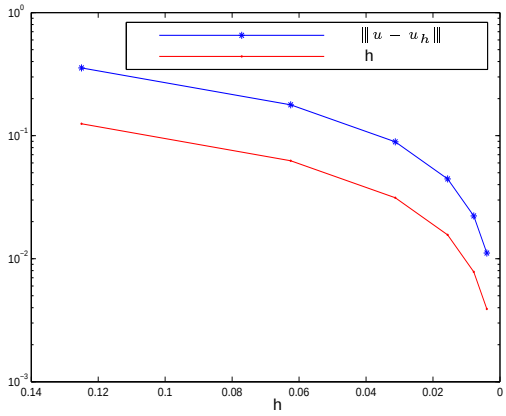


Figure 4.6: Eigenfunction discretization error versus mesh size h.

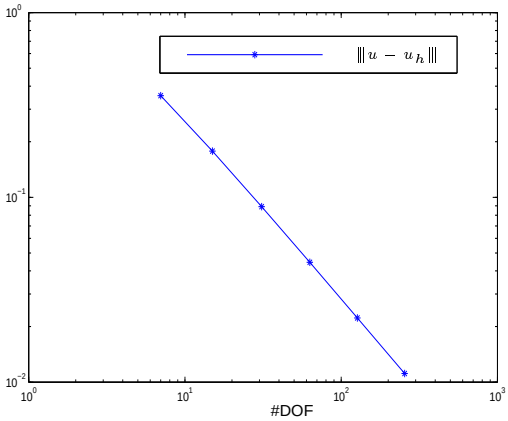


Figure 4.7: Eigenfunction discretization error versus #DOF.

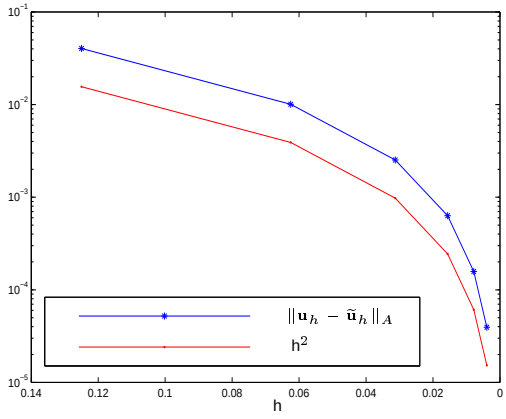


Figure 4.8: Eigenvector iteration error versus mesh size h.

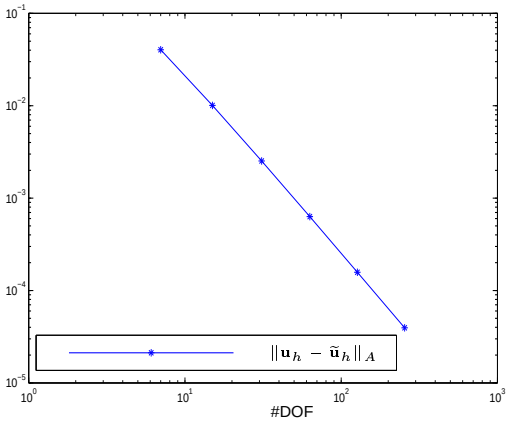


Figure 4.9: Eigenvector iteration error versus #DOF.

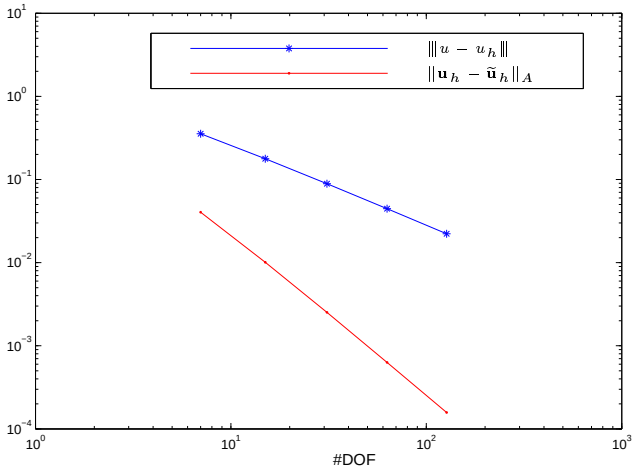


Figure 4.10: Eigenfunction/eigenvector discretization and iteration error versus #DOF.

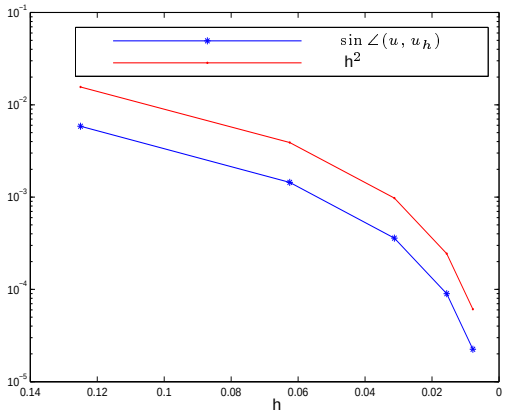


Figure 4.11: Eigenfunction discretization error versus mesh size h.

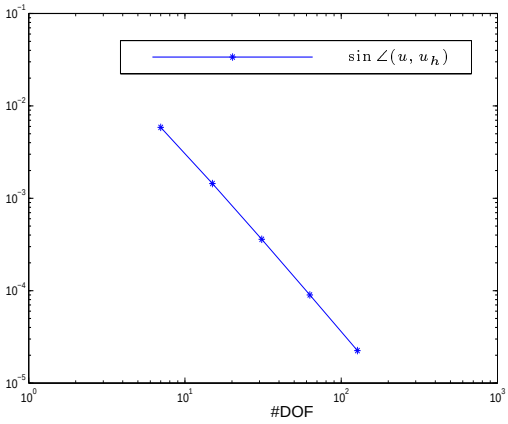


Figure 4.12: Eigenfunction discretization error versus #DOF.

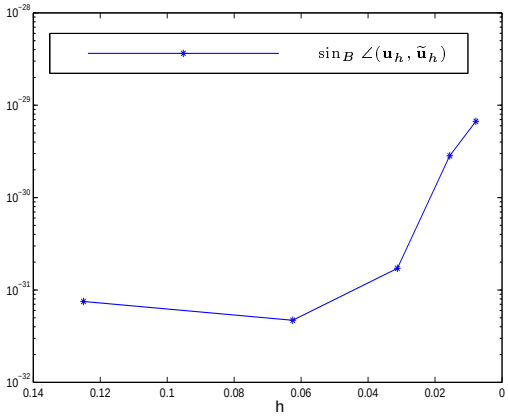


Figure 4.13: Eigenvector iteration error versus mesh size h.

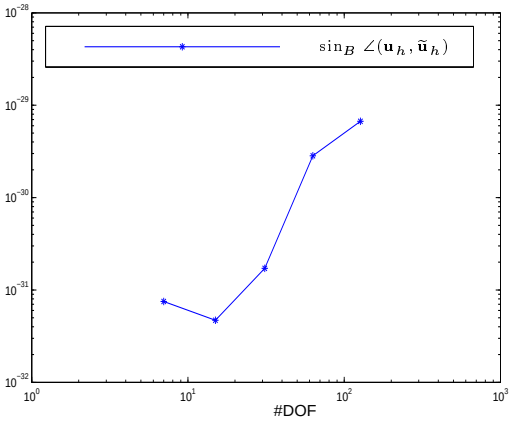


Figure 4.14: Eigenvector iteration error versus #DOF.

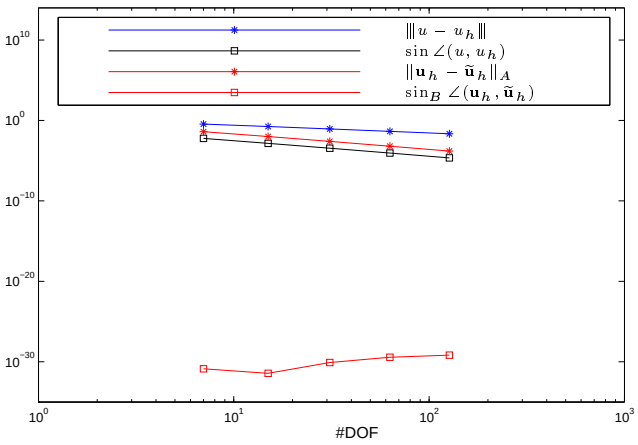


Figure 4.15: Eigenfunction/eigenvector discretization and iteration error versus #DOF.

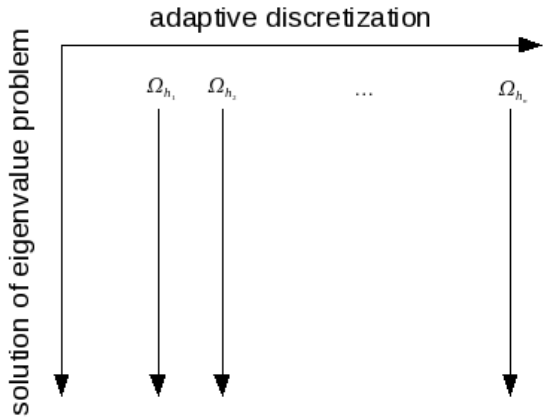


Figure 4.16: The standard AFEM idea.

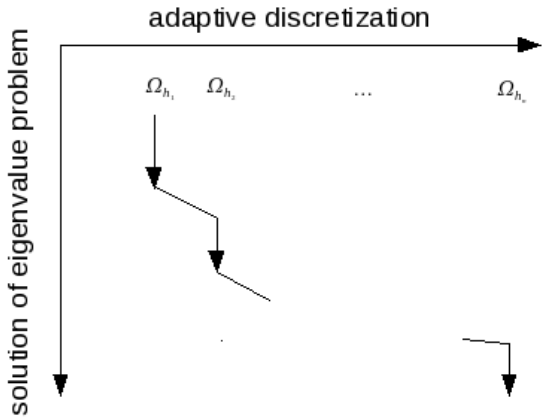


Figure 4.17: The AFEMLA idea.

minimal computational effort and this is exactly our main goal. The idea of the AFEMLA algorithm is illustrated in Figure 4.17.

An additional advantage of the proposed algorithm over the standard AFEM is that this adaptivity technique can be applied even without too much knowledge of the underlying PDE. Let us assume that we would like to obtain the solution of some physical problem, e.g., to compute the noise level inside the car, but the only available information is the matrix representation of the problem and the corresponding finite element grid. Although, we do not have a PDE equation describing the problem and we are not able to construct an appropriate a posteriori error estimator, we still can construct an adaptive algorithm which allows us to obtain a good approximation of the exact solution at a reasonable cost. Since, in the AFEMLA algorithm adaptivity is possible also based on the algebraic residual, it can be used also when the problem comes in a discretized form, e.g. from the finite element modeling [21]. We are not addressing this problem here, however, also in this case, performing the subspace adaptation requires information about underlying meshes and matrices obtained at different discretization levels.

Throughout this and subsequent sections, we use the following notation to distinguish continuous, discrete and approximated eigenpairs. We will denote by (λ_i, u_i) the exact pair of an eigenvalue and its corresponding eigenfunction of (3.1), by $(\lambda_{i,H}, u_{i,H})$ (or $(\lambda_{i,h}, u_{i,h})$) the exact eigenpair for the discrete formulation with respect to the space V_H (or V_h), respectively, and by $(\widetilde{\lambda}_{i,H}, \widetilde{u}_{i,H})$ (or $(\widetilde{\lambda}_{i,h}, \widetilde{u}_{i,h})$) its approximation obtained by an iterative eigenvalue solver in finite precision arithmetic. In order to distinguish the representation vector of the eigenfunction with respect to some basis of V_H (or V_h) from the eigenfunction itself, bold letters will be used, i.e., $\mathbf{u}_{i,H}$ (or $\mathbf{u}_{i,h}$), $\widetilde{\mathbf{u}}_{i,H}$ (or $\widetilde{\mathbf{u}}_{i,h}$). Note that these representation vectors are the eigenvectors of (2.14). In the following, when no index i is given in (λ_i, u_i), $(\lambda_{i,H}, u_{i,H})$, $(\lambda_{i,H}, u_{i,H})$, $(\widetilde{\lambda}_{i,H}, \widetilde{u}_{i,H})$ $(\widetilde{\lambda}_{i,H}, \widetilde{\mathbf{u}}_{i,H})$ etc., then we mean (λ_1, u_1), $(\lambda_{1,H}, u_{1,H})$, $(\widetilde{\lambda}_{1,H}, \widetilde{u}_{1,H})$, and $(\widetilde{\lambda}_{1,H}, \widetilde{\mathbf{u}}_{1,H})$, respectively.

4.2.2 The AFEMLA algorithm

As in the standard case of AFEM, our adaptive finite element method consists of the typical loop

$$\text{SOLVE} \rightarrow \text{ESTIMATE} \rightarrow \text{MARK} \rightarrow \text{REFINE}.$$

After discretizing, we *solve* the algebraic eigenvalue problem using a Krylov subspace method on the coarse mesh, but we stop the iteration early, when sufficient information is available. As a stopping criterion in the iterative procedure we can either use the maximal number p of Arnoldi/Lanczos steps or a desired tolerance; we will discuss these options below. This significantly reduces the cost in the algebraic eigenvalue solver. It would also be possible to use other iterative solvers, like Rayleigh quotient iterations or Newton iterations. In view of the goal to extend these approaches to non-self-adjoint problems and in order to be able to use standardized eigenvalue methods like those implemented in ARPACK [81] or MATLAB [89], we concentrate on the Arnoldi/Lanczos method [81].
For a given matrix $S \in \mathbb{R}^{n_H \times n_H}$ (which in our case corresponds to $B_H^{-1} A_H$, and a nonzero starting vector $w_1 \in \mathbb{R}^{n_H}$, the Arnoldi/Lanczos process generates the *Krylov subspace* $\mathcal{K}_m(S, w_1) = span\{w_1, Sw_1, S^2 w_1, \ldots, S^{m-1} w_1\}$ and determines an orthogonal basis for this subspace spanned by the columns of a matrix W_j.

The approximations to the eigenvalues of the matrix S are then determined via the eigenvalues (called *Ritz values*) of the Hessenberg matrix $H_m = W_m^T S W_m$ which represents an orthogonal projection of the matrix S onto $\mathcal{K}_m(S, w_1)$. Prolongating eigenvectors $\mathbf{y}_1, \ldots, \mathbf{y}_m$ associated with the eigenvalues $\mu_1, \ldots, \mu_m$ of H_m, by setting $\mathbf{u}_{k,H} = W_m \mathbf{y}_k$, $k = 1, 2, \ldots, m$, then yields eigenvector approximations for the given matrix S, i.e., for the generalized eigenvalue problem (3.4) on V_H.

The Arnoldi/Lanczos process is usually terminated at step m, when $\mathcal{K}_m(M, w_1)$ is approximately invariant under S or when a desired tolerance *tol* is reached, i.e., $\|S\mathbf{u}_{k,H} - \mathbf{u}_{k,H}\mu_m\| \leq \max\{\epsilon_M \|H_m\|, |tol||\mu_m|\}$, where ϵ_M is the machine precision [53, 81]. This stopping criterion assures a small backward error, which in the case of symmetric eigenvalue problems corresponds to a small residual vector. As a result of the Arnoldi/Lanczos process we get approximations $\widetilde{\lambda}_H$ to exact eigenvalues λ_H of the generalized eigenvalue problem (3.4) on V_H.

With the approximation $\widetilde{\mathbf{u}}_H$ to the corresponding eigenvector $\mathbf{u}_H$, it follows that the corresponding approximate eigenfunction is given by

$$\widetilde{u}_H = \sum_{i=1}^{n_H} [\widetilde{\mathbf{u}}_H]_i \varphi_i^H = \sum_{i=1}^{n_H} \widetilde{u}_{H,i} \varphi_i^H .$$

We then can check the quality of this solution and use this information for adaptation. From a geometric point of view, it is our goal to enrich the space V_H corresponding to the coarse mesh $\mathcal{T}_H$ by further functions. Since V_H is a subspace of V_h corresponding to the mesh $\mathcal{T}_h$, which is obtained by a uniform refinement of $\mathcal{T}_H$, every function from V_H can be expressed

as a linear combination of functions from V_h. A uniform refinement of every single triangle, also called the *red refinement*, is usually realized by joining the midpoints of the edges [111]. If $\{\varphi_1^h, \ldots, \varphi_{n_h}^h\}$ is a finite element basis for V_h, then we have the identity

$$\widetilde{u}_H = \sum_{i=1}^{n_H} \widetilde{u}_{H,i} \varphi_i^H = \sum_{i=1}^{n_h} \widehat{u}_{h,i} \varphi_i^h,$$

with an appropriate coefficient vector $\widehat{\mathbf{u}}_h$. The relationship between coefficient vectors $\widehat{\mathbf{u}}_h$ and $\widetilde{\mathbf{u}}_H$ can be described by multiplication with a prolongation matrix P that is easily constructed [24, 27].
Therefore, the corresponding prolongated coordinate vector in the fine space associated with the computed eigenvector $\widetilde{\mathbf{u}}_H$ is

$$\widehat{\mathbf{u}}_h = P\widetilde{\mathbf{u}}_H. \tag{4.6}$$

We denote by $(\widehat{\lambda}_h, \widehat{\mathbf{u}}_h)$ an approximate eigenpair obtained from the prolongation of the eigenvector $\widetilde{\mathbf{u}}_H$ on the finite space V_h, where $\widehat{\lambda}_h$ is a generalized Rayleigh quotient corresponding to $\widehat{\mathbf{u}}_h$.

Remark 4.3. Note that if the algebraic eigenvalue problem could be solved exactly, then $\widetilde{\lambda}_H$ and $\widehat{\lambda}_h$ would be equal. But, since eigenvalues usually cannot be computed exactly (even in infinite precision arithmetic) and since we work in finite precision arithmetic, roundoff errors, although not discussed here, have to be taken into account and therefore it is important to distinguish these values.

Based on $(\widehat{\lambda}_h, \widehat{\mathbf{u}}_h)$ we can compute the corresponding residual

$$\widehat{\mathbf{r}}_h = A_h\widehat{\mathbf{u}}_h - \widehat{\lambda}_h B_h \widehat{\mathbf{u}}_h. \tag{4.7}$$

This gives a natural way of *estimating* the error in the computed eigenfunction using the coarse grid solution combined with the fine grid information, namely we can prolongate the already computed approximation $\widetilde{\mathbf{u}}_H$ from V_H to V_h. Then every entry in the residual vector $\widehat{\mathbf{r}}_h$ in (4.7) corresponds to the appropriate basis function from the fine space. Furthermore, we know that if the i-th entry in the vector $\widehat{\mathbf{r}}_h$ is large, then the i-th basis function has a large influence on the solution, namely its support should be further investigated [70]. All these basis functions with large entries in the vector $\widehat{\mathbf{r}}_h$ together with all basis functions from the coarse space form a basis for the new refined space. The decision on whether an entry in the residual vector is small or large is based on different criteria, e.g., a prescribed tolerance or a bulk strategy [48].

When we have identified the basis functions that should be added to enrich our trial space, we start the *marking* procedure. Since every FEM basis function is associated with a specific node in the mesh, enriching the space by new basis function means marking the edge corresponding to its node. In order to avoid hanging nodes or irregular triangulations, we mark some additional edges using a closure algorithm, i.e., if edge E is marked and is not a reference edge (the longest edge) of the element, then we add the reference edge to the set

of marked edges. If an element $T \in \mathcal{T}_H$ has one, two, or three edges marked, we *refine* it by the *green*, *blue*, or *red refinement*, respectively, see, e.g., [4, 26, 111] for more details.

After finishing the refinement procedure, we have a new mesh which will be an initial mesh for the next loop of our adaptive algorithm. Listing **AFEMLA** presents a pseudo-code of this algorithm. In this algorithm, we either use a fixed number of steps p for the iterative method to solve the algebraic eigenvalue problem or we stop the iterative procedure based on a tolerance that is related to the discretization error. It is clear that the optimal choice of parameters p and *tol* with respect to the number of desired eigenvalues is problem dependent and we do not have a global analysis how to achieve an optimal choice. However, one of the main goals of adaptive methods is the optimal use of resources, i.e., achieving a prescribed tolerance with minimal work or a maximal tolerance with given restrictions on the amount of work [14]. Due to the discretization process the considered algebraic eigenvalue problem is already contaminated with a discretization error, which (hopefully) decreases when the mesh size is getting smaller. The discretization error, therefore, is the first natural choice of the tolerance parameter *tol* for the iterative solver. For this reason for the algebraic eigenvalue problem we use a tolerance *tol* that is slightly less than the discretization error.

On the other hand if some storage restrictions are present, i.e., a limit on the maximal number p of Arnoldi/Lanczos vectors, then the question arises how accurate solution we can get. In the case of well separated eigenvalues, as a rule of thumb, the choice of $p = 2 *$ number of desired eigenvalues is acceptable, see [81], but it will not guarantee a high accuracy of the solution. However, we are usually only interested in a good accuracy of the final eigenvalue approximation (i.e., the approximation computed in the last step of the adaptive algorithm), the intermediate values only need to be accurate enough to capture the special behavior of the solution and to serve as initial guess for the nested iteration process. The convergence of the Arnoldi/Lanczos method depends on the gap in the spectrum, not on the quality of the mesh itself, see, e.g., [101]. Good meshes are required in order to assure the accuracy of the computed approximation. Since we are interested in simple and well separated eigenvalues, we expect the method to have a monotone convergence. Although, in the case of the symmetric eigenvalue problem, the convergence rate of the Arnoldi/Lanczos method is known [95, 101] and can be estimated, in general the parameter p is chosen based on heuristics.

AFEMLA for the smallest eigenvalue of the Laplace eigenvalue problem (3.1)

Input: An initial regular triangulation $\mathcal{T}_H^i$, a maximal number p of Arnoldi/Lanczos steps or a tolerance *tol* and a desired accuracy ε.

Output: Approximation $\widehat{\lambda}_1$ to the smallest eigenvalue λ_1 of (3.1) together with the corresponding approximate eigenfunction $\widehat{u}_1$.

Solve: Compute the smallest eigenvalue $\widetilde{\lambda}_H$ and associated eigenvector $\widetilde{\mathbf{u}}_H$ for the algebraic eigenvalue problem (3.4) associated with the coarse mesh $\mathcal{T}_H^i$. The Arnoldi/Lanczos method will be terminated after p steps or when a desired tolerance *tol* is reached.

Express $\widetilde{\mathbf{u}}_H$ using basis functions from the mesh $\mathcal{T}_h^i$ that is obtained by uniformly refining $\mathcal{T}_H^i$. With the prolongation matrix P from the coarse mesh $\mathcal{T}_H^i$ on the fine mesh $\mathcal{T}_h^i$ compute $\widehat{\mathbf{u}}_h = P\widetilde{\mathbf{u}}_H$.

Estimate: Determine the residual $\widehat{\mathbf{r}}_h = A_h\widehat{\mathbf{u}}_h - \widehat{\lambda}_h B_h\widehat{\mathbf{u}}_h$ for the associated eigenvector $\widehat{\mathbf{u}}_h$ and identify all large coefficients in $\widehat{\mathbf{r}}_h$ and corresponding basis functions (nodes).

if $\|\widehat{\mathbf{r}}_h\| < \varepsilon$ **then**
  return $(\widehat{\lambda}_h, \widehat{\mathbf{u}}_h)$
else
  Mark: Mark all edges that contain identified nodes and apply the closure algorithm.
  Refine: Refine the coarse mesh $\mathcal{T}_H^i$ using the red, green or blue refinement to get $\mathcal{T}_H^{i+1}$.

  Start Algorithm with $\mathcal{T}_H^{i+1}$.
end if

4.2.3 Error estimates involving the algebraic error for elliptic self-adjoint eigenvalue problems

In this section we analyze the AFEMLA algorithm theoretically. We discuss the question whether the residual vector provides sufficient information for the refinement procedure and what quality of eigenpairs we may expect. We note that if we have computed a FEM approximation to an eigenvalue and eigenfunction then it is possible, at least in the interior of the elements, to compute the analytic residual for (3.1) given by

$$\widetilde{r}_H = |\Delta\widetilde{u}_H + \widetilde{\lambda}_H\widetilde{u}_H|,$$

which together with the edge residual, that describes the size of discontinuities of normal derivatives across edges of the elements [79], forms the exact residual. In this way we can control errors caused due to the discretization of the infinite dimensional problem. Since, however, we are interested in combining the discretization errors together with the errors in the iterative eigensolver, we analyze the relation between the approximate solution of the algebraic eigenvalue problem on the coarse and the fine grid.

In the next section, we first establish estimates for the eigenvalue error. Using classical perturbation results introduced in Section 2.3.3.1 and the *saturation assumption* we determine bounds on errors between the exact eigenvalue and its approximations. In the second step

we analyze the quality of the corresponding eigenfunctions. Here, except of the theoretical framework from Section 2.3.3.1, we recall norm relations introduced in Section 2.4.

A priori and a posteriori error bounds for eigenvalues

Let us first denote residual vectors corresponding to eigenvectors $\widetilde{\mathbf{u}}_H$, $\widetilde{\mathbf{u}}_h$ and $\widehat{\mathbf{u}}_h$ by

$$\begin{aligned} \mathbf{r}_H &= A_H\widetilde{\mathbf{u}}_H - \widetilde{\lambda}_H B_H\widetilde{\mathbf{u}}_H, && (4.8)\\ \mathbf{r}_h &= A_h\widetilde{\mathbf{u}}_h - \widetilde{\lambda}_h B_h\widetilde{\mathbf{u}}_h, && (4.9)\\ \widehat{\mathbf{r}}_h &= A_h\widehat{\mathbf{u}}_h - \widehat{\lambda}_h B_h\widehat{\mathbf{u}}_h, && (4.10) \end{aligned}$$

respectively. Then applying Theorem 2.40 we obtain the following error bounds.

Corollary 4.4. *Let $(\lambda_H, \mathbf{u}_H)$, $(\lambda_h, \mathbf{u}_h)$ be exact eigenvalues and associated eigenvectors of the matrix pairs $(\mathbf{A}_H, \mathbf{B}_H)$, $(\mathbf{A}_h, \mathbf{B}_h)$, respectively, and let $(\widetilde{\lambda}_H, \widetilde{\mathbf{u}}_H)$, $(\widetilde{\lambda}_h, \widetilde{\mathbf{u}}_h)$ be corresponding computed eigenpairs. Let eigenvectors $\mathbf{u}_H$, $\mathbf{u}_h$ be normalized, i.e., $\|\mathbf{u}_H\|_{B_H} = \|\mathbf{u}_h\|_{B_h} = 1$. Then the following estimates hold.*

$$\begin{aligned} |\widetilde{\lambda}_H - \lambda_H| &\leq \frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}}, && (4.11)\\ |\widetilde{\lambda}_h - \lambda_h| &\leq \frac{\|\mathbf{r}_h\|_{B_h^{-1}}}{\|\widetilde{\mathbf{u}}_h\|_{B_h}}, && (4.12)\\ |\widehat{\lambda}_h - \lambda_h| &\leq \frac{\|\widehat{\mathbf{r}}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}}. && (4.13) \end{aligned}$$

It should be noted that in our algorithm we do not want to compute the fine grid eigenpair $(\widetilde{\lambda}_h, \widetilde{\mathbf{u}}_h)$. For this reason, the fine grid residual $\|\mathbf{r}_h\|_{B_h^{-1}}$ is usually not available. Instead, we use $(\widehat{\lambda}_h, \widehat{\mathbf{u}}_h)$ as its approximation and have the following bounds.

Theorem 4.5. *Let $(\widetilde{\lambda}_H, \widetilde{\mathbf{u}}_H)$ be the computed eigenpair of the matrix pair $(\mathbf{A}_H, \mathbf{B}_H)$ and let $(\widehat{\lambda}_h, \widehat{\mathbf{u}}_h)$ be the computed eigenpair obtained by the prolongation (with the prolongation matrix P) of $\widetilde{\mathbf{u}}_H$ on the fine space V_h as defined in (4.6). Assume that these eigenvectors are normalized, i.e., $\|\widetilde{\mathbf{u}}_H\|_{B_H} = \|\widehat{\mathbf{u}}_h\|_{B_h} = 1$. Then*

$$|\widetilde{\lambda}_H - \widehat{\lambda}_h| \leq \frac{\|\mathbf{r}_H\|_{B_H} + \|P^T\widehat{\mathbf{r}}_h\|_{B_H}}{\|P^T\mathbf{B}_h\widehat{\mathbf{u}}_h\|_{B_H}}.$$

Proof. Following Theorem 2.33, eigenpairs $(\widetilde{\lambda}_H, \widetilde{\mathbf{u}}_h)$, $(\widehat{\lambda}_h, \widehat{\mathbf{u}}_h)$ are exact eigenpairs of the eigenvalue problem

$$\begin{aligned} (A_H + E_H)\widetilde{\mathbf{u}}_H &= \widetilde{\lambda}_H B_H\widetilde{\mathbf{u}}_H, && (4.14)\\ (A_h + \widehat{E}_h)\widehat{\mathbf{u}}_h &= \widehat{\lambda}_h B_h\widehat{\mathbf{u}}_h, && (4.15) \end{aligned}$$

respectively, with $E_H = -\mathbf{r}_H \widetilde{\mathbf{u}}^T B_H$ and $\widehat{E}_h = -\widehat{\mathbf{r}}_h \widehat{\mathbf{u}}^T B_h$.
Using the relation between the coarse and the fine space, i.e.,

$$P^T A_h P = A_H, \; P^T B_h P = B_H,$$

it follows that (4.14) is equivalent to

$$(P^T A_h P + E_H)\widetilde{\mathbf{u}}_H = \widetilde{\lambda}_H (P^T B_h P)\widetilde{\mathbf{u}}_H. \tag{4.16}$$

Multiplying (4.15) from the left by P^T gives

$$P^T A_h \widehat{\mathbf{u}}_h + P^T \widehat{E}_h \widehat{\mathbf{u}}_h = \widehat{\lambda}_h P^T B_h \widehat{\mathbf{u}}_h. \tag{4.17}$$

Using the fact that $P\widetilde{\mathbf{u}}_H = \widehat{\mathbf{u}}_h$, we can rewrite (4.16) as

$$P^T A_h P \widetilde{\mathbf{u}}_H + E_H \widetilde{\mathbf{u}}_H = \widetilde{\lambda}_H P^T B_h P \widetilde{\mathbf{u}}_H. \tag{4.18}$$

By subtracting (4.17) from (4.18) we then get

$$(\widetilde{\lambda}_H - \widehat{\lambda}_h)(P^T B_h \widehat{\mathbf{u}}_h) = P^T A_h \widehat{\mathbf{u}}_h + E_H \widetilde{\mathbf{u}}_H - P^T A_h \widehat{\mathbf{u}}_h - P^T \widehat{E}_h \widehat{\mathbf{u}}_h$$

and, by applying the triangle inequality, finally

$$\begin{aligned} |\widetilde{\lambda}_H - \widehat{\lambda}_h| &= \frac{\|E_H \widetilde{\mathbf{u}}_H - P^T \widehat{E}_h \widehat{\mathbf{u}}_h\|_{B_H}}{\|P^T B_h \widehat{\mathbf{u}}_h\|_{B_H}} \leq \frac{\|E_H \widetilde{\mathbf{u}}_H\|_{B_H} + \|P^T \widehat{E}_h \widehat{\mathbf{u}}_h\|_{B_H}}{\|P^T B_h \widehat{\mathbf{u}}_h\|_{B_H}} \\ &= \frac{\| - \mathbf{r}_H \widetilde{\mathbf{u}}_H^T B_H \widetilde{\mathbf{u}}_H\|_{B_H} + \| - P^T \widehat{\mathbf{r}}_h \widehat{\mathbf{u}}_h^T B_h \widehat{\mathbf{u}}_h\|_{B_H}}{\|P^T B_h \widehat{\mathbf{u}}_h\|_{B_H}} \\ &= \frac{\|\mathbf{r}_H\|_{B_H} + \|P^T \widehat{\mathbf{r}}_h\|_{B_H}}{\|P^T B_h \widehat{\mathbf{u}}_h\|_{B_H}}. \quad \square \end{aligned}$$

Having obtained the estimate of the error between the computed eigenvalue on the coarse grid and the prolongated eigenvalue on the fine grid we next obtain the similar relation between the exact eigenvalue on the coarse grid and the prolongated eigenvalue.

Corollary 4.6. *Let $(\lambda_H, \mathbf{u}_H)$ be the exact eigenpair of the matrix pair (A_H, B_H) and let $(\widehat{\lambda}_h, \widehat{\mathbf{u}}_h)$ be the eigenpair obtained by the prolongation of the computed pair $(\widetilde{\lambda}_H, \widetilde{\mathbf{u}}_H)$ on the fine space V_h. Then, the following bound holds.*

$$|\lambda_H - \widehat{\lambda}_h| \leq \frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}} + \frac{\|\mathbf{r}_H\|_{B_H} + \|P^T \widehat{\mathbf{r}}_h\|_{B_H}}{\|P^T B_h \widehat{\mathbf{u}}_h\|_{B_H}}.$$

Proof. Using the triangle inequality we get

$$|\lambda_H - \widehat{\lambda}_h| = |\lambda_H - \widetilde{\lambda}_H + \widetilde{\lambda}_H - \widehat{\lambda}_h| \leq |\lambda_H - \widetilde{\lambda}_H| + |\widetilde{\lambda}_H - \widehat{\lambda}_h|.$$

Inserting bound (4.11) and the result of Theorem 4.5 we get

$$|\lambda_H - \widehat{\lambda}_h| \leq \frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}} + \frac{\|\mathbf{r}_H\|_{B_H} + \|P^T\widehat{\mathbf{r}}_h\|_{B_H}}{\|P^T B_h \widehat{\mathbf{u}}_h\|_{B_H}}. \quad \square$$

Applying triangle inequalities

$$|\lambda_H - \lambda_h| = |\lambda_H - \widehat{\lambda}_h + \widehat{\lambda}_h - \lambda_h| \leq |\lambda_H - \widehat{\lambda}_h| + |\widehat{\lambda}_h - \lambda_h|,$$

$$|\lambda_h - \widetilde{\lambda}_H| = |\lambda_h - \widehat{\lambda}_h + \widehat{\lambda}_h = \widetilde{\lambda}_H| \leq |\lambda_h - \widehat{\lambda}_h| + |\widehat{\lambda}_h - \widetilde{\lambda}_H|,$$

we get the error estimate between the exact fine and the coarse grid eigenvalue, i.e.,

$$|\lambda_H - \lambda_h| \leq \frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}} + \frac{\|\mathbf{r}_H\|_{B_H} + \|P^T\widehat{\mathbf{r}}_h\|_{B_H}}{\|P^T B_h \widehat{\mathbf{u}}_h\|_{B_H}} + \frac{\|\widehat{\mathbf{r}}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}}, \tag{4.19}$$

and between the exact fine grid and the computed coarse grid eigenvalue, i.e.,

$$|\lambda_h - \widetilde{\lambda}_H| \leq \frac{\|\widehat{\mathbf{r}}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}} + \frac{\|\mathbf{r}_H\|_{B_H} + \|P^T\widehat{\mathbf{r}}_h\|_{B_H}}{\|P^T B_h \widehat{\mathbf{u}}_h\|_{B_H}}. \tag{4.20}$$

Combining these bounds and using the triangle inequality in different ways we obtain the following further bounds.

Corollary 4.7. *Let $(\lambda_H, \mathbf{u}_H)$, $(\lambda_h, \mathbf{u}_h)$ be the exact and let $(\widetilde{\lambda}_H, \widetilde{\mathbf{u}}_H)$, $(\widetilde{\lambda}_h, \widetilde{\mathbf{u}}_h)$ be the computed eigenpairs of the matrix pair (A_H, B_H), (A_h, B_h), respectively. Furthermore, let $(\widehat{\lambda}_h, \widehat{\mathbf{u}}_h)$ be the eigenpair obtained by the prolongation of $\widetilde{\mathbf{u}}_H$ on the fine space V_h defined as in (4.6). Then the following bounds hold*

$$\begin{aligned}
|\widetilde{\lambda}_h - \widehat{\lambda}_h| &\leq \frac{\|\mathbf{r}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}} + \frac{\|\widehat{\mathbf{r}}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}}, \\
|\widetilde{\lambda}_H - \widetilde{\lambda}_h| &\leq \frac{\|\mathbf{r}_H\|_{B_H} + \|P^T\widehat{\mathbf{r}}_h\|_{B_H}}{\|P^T B_h \widehat{\mathbf{u}}_h\|_{B_H}} + \frac{\|\mathbf{r}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}} + \frac{\|\widehat{\mathbf{r}}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}}, \\
|\lambda_H - \widetilde{\lambda}_h| &\leq \frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}} + \frac{\|\mathbf{r}_H\|_{B_H} + \|P^T\widehat{\mathbf{r}}_h\|_{B_H}}{\|P^T B_h \widehat{\mathbf{u}}_h\|_{B_H}} + \frac{\|\mathbf{r}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}} + \frac{\|\widehat{\mathbf{r}}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}}.
\end{aligned}$$

Proof. The proof follows by combing the previous bounds and using the triangle inequality. $\square$

Theorem 4.5 and Corollaries 4.6 and 4.7 present error bounds with respect to the exact and computed eigenvalues for discretized algebraic eigenvalue problems. Some of these bounds are computable, while some have only theoretical meaning. In fact, we are interested in errors between eigenvalues of the continuous problem and those of the discrete problem, therefore, we would like to find computable bounds based on residual vectors. Since we were able to transform residual errors to backward errors, it follows that for well-conditioned eigenvalues we can expect that small residuals will imply a good accuracy of their approximations obtained by iterative solvers.

In order to relate the continuous and discrete eigenpairs we will use the so-called *saturation assumption*, namely we will assume that the approximation of the eigenpair computed on the fine space V_h is better than the approximation on the coarse space V_H. In practice this assumption is equivalent to small data oscillations and the convergence of the AFEM procedure [49].

Theorem 4.8 (Neymeyr [92])**.** *Let λ be an exact eigenvalue of (3.1). Let λ_H, λ_h be the corresponding exact eigenvalues of discretized problems on spaces V_H, V_h, respectively. Then the saturation assumption states that*

$$\lambda_H - \lambda_h \geq (1-\beta)(\lambda_H - \lambda), \tag{4.21}$$

with a positive $\beta < 1$, and it is equivalent to

$$\lambda_h - \lambda \leq \beta(\lambda_H - \lambda). \tag{4.22}$$

Remark 4.9. Inequality (4.22) is also equivalent to

$$\lambda_H - \lambda \leq \frac{1}{1-\beta}(\lambda_H - \lambda_h). \tag{4.23}$$

Therefore, later in this section, we will work with inequality (4.23). It is important to mention that (4.22) is actually stating the convergence of the adaptive method. Note also, that typically β is not known, but in our case, for a uniform subdivision and a sufficiently small mesh size h, is it known that $\beta \approx \frac{1}{2}$, see [2] for details.

Remark 4.10. Since for the symmetric eigenvalue problem the Courant-Fischer minimax theorem holds [46, 107], the exact eigenvalue λ of the PDE eigenvalue problem and the eigenvalues λ_H, λ_h of discretized problems satisfy the inequality

$$\lambda \leq \lambda_h \leq \lambda_H .$$

Thus, inequalities (4.22) and (4.23) are equivalent to

$$|\lambda_h - \lambda| \leq \beta|\lambda_H - \lambda| \quad \text{and} \quad |\lambda_H - \lambda| \leq \frac{1}{1-\beta}|\lambda_H - \lambda_h|,$$

respectively.

Based on the saturation assumption stated in Theorem 4.8 and estimates for errors between the exact and computed eigenvalues for discretized eigenvalue problems we obtain the following bounds.

Corollary 4.11. *Let λ be the exact eigenvalue of (3.1) and let u be the corresponding eigenfunction. Let λ_H be the corresponding exact eigenvalue of the discretized generalized eigenvalue problem with the matrix pair (A_H, B_H), and let $\widehat{\mathbf{u}}_H$ be defined as in (4.6). Then with residual vectors $\mathbf{r}_H$, $\widehat{\mathbf{r}}_h$ defined as in (4.8), (4.10) we have*

$$|\lambda_H - \lambda| \leq \frac{1}{1-\beta}\left(\frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}} + \frac{\|\mathbf{r}_H\|_{B_H} + \|P^T\widehat{\mathbf{r}}_h\|_{B_H}}{\|P^T B_h \widehat{\mathbf{u}}_h\|_{B_H}} + \frac{\|\widehat{\mathbf{r}}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}}\right). \tag{4.24}$$

Proof. Under the saturation assumption from Theorem 4.8 it follows that

$$|\lambda_H - \lambda| \leq \frac{1}{1-\beta}|\lambda_H - \lambda_h|.$$

By applying bound (4.19) we get

$$|\lambda_H - \lambda_h| \leq \frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}} + \frac{\|\mathbf{r}_H\|_{B_H} + \|P^T\widehat{\mathbf{r}}_h\|_{B_H}}{\|P^T B_h \widehat{\mathbf{u}}_h\|_{B_H}} + \frac{\|\widehat{\mathbf{r}}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}}$$

and thus

$$|\lambda_H - \lambda| \leq \frac{1}{1-\beta}\left(\frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}} + \frac{\|\mathbf{r}_H\|_{B_H} + \|P^T\widehat{\mathbf{r}}_h\|_{B_H}}{\|P^T B_h \widehat{\mathbf{u}}_h\|_{B_H}} + \frac{\|\widehat{\mathbf{r}}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}}\right). \quad \square$$

Corollary 4.12. *Let λ be the exact eigenvalue of (3.1), let λ_h be the corresponding exact fine grid eigenvalue and let $\widehat{\mathbf{u}}_h$ be defined as in (4.6). Then with residual vectors $\mathbf{r}_H$, $\widehat{\mathbf{r}}_h$ defined as in (4.8), (4.10) we have*

$$|\lambda_h - \lambda| \leq (1 + \frac{1}{1-\beta})\left(\frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}} + \frac{\|\mathbf{r}_H\|_{B_H} + \|P^T\widehat{\mathbf{r}}_h\|_{B_H}}{\|P^T B_h \widehat{\mathbf{u}}_h\|_{B_H}} + \frac{\|\widehat{\mathbf{r}}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}}\right).$$

Proof. Under the saturation assumption from Theorem 4.8 it follows that

$$|\lambda_h - \lambda| \leq |\lambda_h - \lambda_H| + |\lambda_H - \lambda| \leq |\lambda_h - \lambda_H| + \frac{1}{1-\beta}|\lambda_h - \lambda_H| = (1 + \frac{1}{1-\beta})|\lambda_h - \lambda_H|.$$

From (4.19) we have that

$$|\lambda_H - \lambda_h| \leq \frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}} + \frac{\|\mathbf{r}_H\|_{B_H} + \|P^T\widehat{\mathbf{r}}_h\|_{B_H}}{\|P^T B_h \widehat{\mathbf{u}}_h\|_{B_H}} + \frac{\|\widehat{\mathbf{r}}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}}$$

and hence

$$|\lambda_h - \lambda| \leq (1 + \frac{1}{1-\beta})\left(\frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}} + \frac{\|\mathbf{r}_H\|_{B_H} + \|P^T\widehat{\mathbf{r}}_h\|_{B_H}}{\|P^T B_h \widehat{\mathbf{u}}_h\|_{B_H}} + \frac{\|\widehat{\mathbf{r}}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}}\right). \quad \square$$

Using previous estimates we can also obtain the bound on the error between the computed eigenvalue on the coarse mesh and the corresponding eigenvalue of the original PDE eigenvalue problem.

Theorem 4.13. *Let λ be the exact eigenvalue of (3.1), $\widetilde{\lambda}_H$ the computed coarse grid eigenvalue and let $\widehat{\mathbf{u}}_h$ be defined as in (4.6). Then with residual vectors $\mathbf{r}_H$, $\widehat{\mathbf{r}}_h$ defined as in (4.8), (4.10), we have*

$$|\widetilde{\lambda}_H - \lambda| \leq \frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}} + \frac{1}{1-\beta}\left(\frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}} + \frac{\|\mathbf{r}_H\|_{B_H} + \|P^T\widehat{\mathbf{r}}_h\|_{B_H}}{\|P^T B_h \widehat{\mathbf{u}}_h\|_{B_H}} + \frac{\|\widehat{\mathbf{r}}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}}\right).$$

Proof. From (4.11) we know that

$$|\widetilde{\lambda}_H - \lambda_H| \leq \frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}}$$

and from Corollary 4.11 we have that

$$|\lambda_H - \lambda| \leq \frac{1}{1-\beta}\left(\frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}} + \frac{\|\mathbf{r}_H\|_{B_H} + \|P^T\widehat{\mathbf{r}}_h\|_{B_H}}{\|P^T B_h \widehat{\mathbf{u}}_h\|_{B_H}} + \frac{\|\widehat{\mathbf{r}}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}}\right).$$

Hence,

$$|\widetilde{\lambda}_H - \lambda| \leq \frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}} + \frac{1}{1-\beta}\left(\frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}} + \frac{\|\mathbf{r}_H\|_{B_H} + \|P^T\widehat{\mathbf{r}}_h\|_{B_H}}{\|P^T B_h \widehat{\mathbf{u}}_h\|_{B_H}} + \frac{\|\widehat{\mathbf{r}}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}}\right). \quad \square$$

Theorem 4.14. *Let λ be the exact eigenvalue of (3.1) and let $\widetilde{\mathbf{u}}_H$ be the computed coarse grid eigenvector. Furthermore, let $(\widehat{\lambda}_h, \widehat{\mathbf{u}}_h)$ be the corresponding eigenpair obtained by the prolongation of $\widetilde{\mathbf{u}}_H$ on the fine space V_h defined as in (4.6). Then with residual vectors $\mathbf{r}_H$, $\widehat{\mathbf{r}}_h$ as defined in (4.8), (4.10) we have*

$$\begin{aligned}|\widehat{\lambda}_h - \lambda| &\leq \frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}} + \frac{\|\mathbf{r}_H\|_{B_H} + \|P^T\widehat{\mathbf{r}}_h\|_{B_H}}{\|P^T B_h \widehat{\mathbf{u}}_h\|_{B_H}} \\ &+ \frac{1}{1-\beta}\left(\frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}} + \frac{\|\mathbf{r}_H\|_{B_H} + \|P^T\widehat{\mathbf{r}}_h\|_{B_H}}{\|P^T B_h \widehat{\mathbf{u}}_h\|_{B_H}} + \frac{\|\widehat{\mathbf{r}}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}}\right).\end{aligned}$$

Proof. From Corollary 4.6 it follows that

$$|\lambda_H - \widehat{\lambda}_h| \leq \frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}} + \frac{\|\mathbf{r}_H\|_{B_H} + \|P^T\widehat{\mathbf{r}}_h\|_{B_H}}{\|P^T B_h \widehat{\mathbf{u}}_h\|_{B_H}}$$

and from Corollary 4.11 we have that

$$|\lambda_H - \lambda| \leq \frac{1}{1-\beta}\left(\frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}} + \frac{\|\mathbf{r}_H\|_{B_H} + \|P^T\widehat{\mathbf{r}}_h\|_{B_H}}{\|P^T B_h \widehat{\mathbf{u}}_h\|_{B_H}} + \frac{\|\widehat{\mathbf{r}}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}}\right)$$

and hence,

$$\begin{aligned}|\widehat{\lambda}_h - \lambda| &\leq \frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}} + \frac{\|\mathbf{r}_H\|_{B_H} + \|P^T\widehat{\mathbf{r}}_h\|_{B_H}}{\|P^T B_h \widehat{\mathbf{u}}_h\|_{B_H}} \\ &+ \frac{1}{1-\beta}\left(\frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}} + \frac{\|\mathbf{r}_H\|_{B_H} + \|P^T\widehat{\mathbf{r}}_h\|_{B_H}}{\|P^T B_h \widehat{\mathbf{u}}_h\|_{B_H}} + \frac{\|\widehat{\mathbf{r}}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}}\right). \quad \square\end{aligned}$$

Theorem 4.15. *Let λ be the exact eigenvalue of (3.1) and $\widetilde{\lambda}_h$ the computed fine grid eigenvalue. Then with residual vectors $\mathbf{r}_H$, $\mathbf{r}_h$, $\widehat{\mathbf{r}}_h$ defined as in (4.8), (4.9), (4.10) we have*

$$\begin{aligned} |\widetilde{\lambda}_h - \lambda| &\leq (1 + \frac{1}{1-\beta}) \left(\frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}} + \frac{\|\mathbf{r}_H\|_{B_H} + \|P^T \widehat{\mathbf{r}}_h\|_{B_H}}{\|P^T B_h \widehat{\mathbf{u}}_h\|_{B_H}} + \frac{\|\widehat{\mathbf{r}}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}} \right) \\ &+ \frac{\|\mathbf{r}_h\|_{B_h^{-1}}}{\|\widetilde{\mathbf{u}}_h\|_{B_h}}. \end{aligned}$$

Proof. From the triangle inequality it follows that

$$|\widetilde{\lambda}_h - \lambda| \leq |\widetilde{\lambda}_h - \lambda_h| + |\lambda_h - \lambda|.$$

By applying bound (4.13) together with Corollary 4.12 we have that

$$\begin{aligned} |\widetilde{\lambda}_h - \lambda| &\leq (1 + \frac{1}{1-\beta}) \left(\frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}} + \frac{\|\mathbf{r}_H\|_{B_H} + \|P^T \widehat{\mathbf{r}}_h\|_{B_H}}{\|P^T B_h \widehat{\mathbf{u}}_h\|_{B_H}} + \frac{\|\widehat{\mathbf{r}}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}} \right) \\ &+ \frac{\|\mathbf{r}_h\|_{B_h^{-1}}}{\|\widetilde{\mathbf{u}}_h\|_{B_h}}. \end{aligned}$$

□

A priori and a posteriori error bounds for the eigenfunctions

Since a priori error estimates, discussed in Section 2.2.3, clearly indicate that the efficiency of the FEM algorithm and its convergence depend strongly on the eigenfunction approximation, here we investigate the quality of the (prolongated) eigenfunction (eigenvector) approximation.
At first we introduce the saturation assumption for eigenfunctions in terms of angles between them.

Theorem 4.16. *Let u be the exact eigenfunction of (3.1) and let u_H, u_h be its coarse and fine grid approximations defined as the exact solution of variational equations (3.2), (3.3), respectively. Then the saturation assumption states that the following inequality holds*

$$\sin \angle(u, u_h) \leq \gamma \sin \angle(u, u_H),$$

with a positive $\gamma < 1$.

Proof. See, e.g., [16, 15]. □
Similarly to the eigenvalue case, we assume that the angle between the exact eigenfunction u and its fine grid approximation u_h is smaller than the angle with respect to a coarse grid approximation u_H.

Corollary 4.17. *Let u be the exact eigenfunction of (3.1). Let u_H, u_h be the corresponding exact eigenfunctions of the discretized problem on spaces V_H, V_h, respectively. Then the saturation assumption from Theorem 4.16 is equivalent to*

$$\sin \angle(u, u_H) \leq \frac{1}{1-\gamma} \sin \angle(u_H, u_h). \tag{4.25}$$

Proof. $\Rightarrow$ The triangle inequality for angles gives

$$|\sin \angle(u, u_H) - \sin \angle(u_H, u_h)| \leq \sin \angle(u, u_h),$$

hence

$$-\sin \angle(u, u_h) \leq \sin \angle(u, u_H) - \sin \angle(u_H, u_h) \leq \sin \angle(u, u_h).$$

Applying the *saturation assumption* from Theorem 4.16 yields the upper bound

$$\sin \angle(u, u_H) - \sin \angle(u_H, u_h) \leq \sin \angle(u, u_h) \leq \gamma \sin \angle(u, u_H),$$

which leads to

$$(1-\gamma) \sin \angle(u, u_H) \leq \sin \angle(u_H, u_h).$$

Since $\gamma \in (0, 1)$, the inequality

$$\sin \angle(u, u_H) \leq \frac{1}{1-\gamma} \sin \angle(u_H, u_h)$$

holds.
$\Leftarrow$ Given (4.25) we have

$$-\gamma \sin \angle(u, u_H) \leq \sin \angle(u_H, u_h) - \sin \angle(u, u_H).$$

The triangle inequality for angles states that

$$|\sin \angle(u, u_H) - \sin \angle(u_H, u_h)| \leq \sin \angle(u, u_h).$$

Therefore,

$$-\gamma \sin \angle(u, u_H) \leq \sin \angle(u, u_h),$$

and finally

$$\sin \angle(u, u_h) \leq \gamma \sin \angle(u, u_H).$$

□

Corollary 4.18. *Let u be the exact eigenfunction of (3.1). Let u_H, u_h define the corresponding exact eigenfunctions of the discretized problem on spaces V_H, V_h, respectively. Then*

$$\sin \angle(u, u_h) \leq \frac{\gamma}{1-\gamma} \sin \angle(u_H, u_h).$$

Proof. The *saturation assumption* from Theorem 4.16 gives

$$\sin \angle(u, u_h) \leq \gamma \sin \angle(u, u_H),$$

which together with Corollary 4.17 yields

$$\sin \angle(u, u_h) \leq \gamma \sin \angle(u, u_H) \leq \frac{\gamma}{1-\gamma} \sin \angle(u_H, u_h).$$

□

Corollary 4.19. *Let u be the exact eigenfunction of (3.1). Let u_H, u_h define the corresponding exact eigenfunctions of the discretized problem on spaces V_H, V_h, respectively. If $\widetilde{u}_H$ is the approximation of the exact discrete eigenfunction u_H then*

$$\sin \angle(u, \widetilde{u}_H) \leq \sin \angle(\widetilde{u}_H, u_H) + \frac{1}{1-\gamma} \sin \angle(u_H, u_h).$$

Proof. The triangle inequality for angles

$$|\sin \angle(u, \widetilde{u}_H) - \sin \angle(\widetilde{u}_H, u_H)| \leq \sin \angle(u, u_H),$$

yields that

$$\sin \angle(u, \widetilde{u}_H) \leq \sin \angle(\widetilde{u}_H, u_H) + \sin \angle(u, u_H),$$

which together with Corollary 4.17 leads to

$$\sin \angle(u, \widetilde{u}_H) \leq \sin \angle(\widetilde{u}_H, u_H) + \frac{1}{1-\gamma} \sin \angle(u_H, u_h).$$

□

Corollary 4.20. *Let u be the exact eigenfunction of (3.1). Let u_H, u_h define the corresponding exact eigenfunctions of the discretized problem on spaces V_H, V_h, respectively. If $\widetilde{u}_H$ is the approximation of the exact discrete eigenfunction u_H, then for $\widehat{u}_h$ which is a prolongation of $\widetilde{u}_H$ on the space V_h the following inequality holds*

$$\sin \angle(u, \widehat{u}_h) \leq \sin \angle(\widehat{u}_h, u_h) + \frac{\gamma}{1-\gamma} \sin \angle(u_H, u_h)$$

Proof. The proof follows from the triangle inequality $|\sin \angle(u, \widehat{u}_h) - \sin \angle(\widehat{u}_h, u_h)| \leq \sin \angle(u, u_h)$ and Corollary 4.18. □

Following the previous propositions the quantities $\sin \angle(\widetilde{u}_H, u_H)$, $\sin \angle(u_h, \widehat{u}_h)$, $\sin \angle(u_H, u_h)$ are of particular interest. In order to get a posteriori error estimates for angles between eigenfunctions of interest and their approximations, upper bounds for angles should involve computable quantities, i.e., approximate eigenfunctions (eigenvectors), residuals (residual vectors) etc.. Since, as before, we would like to use perturbation results obtained for algebraic eigenvalue problems, which involve vectors instead of functions, the following result is the foundation for further estimates. With the relation between the eigenfunction and the eigenvector angle from Theorem 2.50 we can complete our estimates.

Corollary 4.21. *Let u_H, u_h be the exact eigenfunction of the discretized problem on spaces V_H, V_h, respectively and let $\widehat{u}_h$ be the prolongation of $\widetilde{u}_H$ on the space V_h, where $\widetilde{u}_H$ is an iterative approximation of u_H. The corresponding representation vectors are denoted by $\mathbf{u}_H, \mathbf{u}_h$, $\widehat{\mathbf{u}}_h$ and $\widetilde{\mathbf{u}}_H$, respectively. Then*

$$\sin \angle(\widetilde{u}_H, u_H) = \sin \angle_{B_H}(\widetilde{\mathbf{u}}_H, \mathbf{u}_H),$$

and

$$\sin \angle(\widehat{u}_h, u_h) = \sin \angle_{B_h}(\widehat{\mathbf{u}}_h, \mathbf{u}_h).$$

Proof. The results are straightforward consequences of applying Theorem 2.50 with a discrete equivalence of the $L_2(\Omega)$-norm on V_H and V_h, i.e., $\|\cdot\|_{B_H}$ and $\|\cdot\|_{B_h}$-norm. □

Corollary 4.22. *Let u_H, u_h be the exact eigenfunction of the discretized problem on spaces V_H, V_h, respectively. Furthermore, let the corresponding representation vectors be $\mathbf{u}_H, \mathbf{u}_h$, respectively. Then*

$$\sin \angle(u_H, u_h) = \sin \angle(u_H^h, u_h) = \sin \angle_{B_h}(\mathbf{u}_H^h, \mathbf{u}_h),$$

where u_H^h is the prolongation of the function u_H on the space V_h and $\mathbf{u}_H^h$ denotes the corresponding representation vector.

Proof. Since $V_H \subset V_h$, the function u_H is an element of the space V_h. Thus

$$\sin \angle(u_H, u_h) = \sin \angle(u_H^h, u_h).$$

The final equality follows from Theorem 2.50. □

Estimating $\sin \angle_{B_h}(\mathbf{u}_H^h, \mathbf{u}_h)$ involves one additional relation.

Corollary 4.23. *Let $\mathbf{u}_H^h, \widehat{\mathbf{u}}_h, \widetilde{\mathbf{u}}_H, \mathbf{u}_H$ be defined as in preceding corollaries. Then*

$$\sin \angle_{B_h}(\mathbf{u}_H^h, \widehat{\mathbf{u}}_h) = \sin \angle_{B_H}(\widetilde{\mathbf{u}}_H, \mathbf{u}_H)$$

Proof. It follows from Definition 2.9 that

$$\cos \angle_{B_h}(\mathbf{u}_H^h, \widehat{\mathbf{u}}_h) = \frac{(\mathbf{u}_H^h, \widehat{\mathbf{u}}_h)_{B_h}}{\|\mathbf{u}_H^h\|_{B_h} \|\widehat{\mathbf{u}}_h\|_{B_h}}.$$

Since

$$\mathbf{u}_H^h = P\mathbf{u}_H, \quad \widehat{\mathbf{u}}_h = P\widetilde{\mathbf{u}}_H \quad \text{and} \quad P^T B_h P = B_h,$$

thus

$$\begin{aligned}
\cos \angle_{B_h}(\mathbf{u}_H^h, \widehat{\mathbf{u}}_h) &= \frac{(\mathbf{u}_H^h, \widehat{\mathbf{u}}_h)_{B_h}}{\|\mathbf{u}_H^h\|_{B_h}\|\widehat{\mathbf{u}}_h\|_{B_h}} = \frac{(P\mathbf{u}_H, P\widetilde{\mathbf{u}}_H)_{B_h}}{\|P\mathbf{u}_H\|_{B_h}\|P\widetilde{\mathbf{u}}_h\|_{B_h}} \\
&= \frac{\widetilde{\mathbf{u}}_H^T P^T B_h P \mathbf{u}_H}{\sqrt{\mathbf{u}_H^T P^T B_h P \mathbf{u}_H}\sqrt{\widetilde{\mathbf{u}}_H^T P^T B_h P \widetilde{\mathbf{u}}_H}} \\
&= \frac{\widetilde{\mathbf{u}}_H^T B_H \mathbf{u}_H}{\sqrt{\mathbf{u}_H^T B_H \mathbf{u}_H}\sqrt{\widetilde{\mathbf{u}}_H^T B_H \widetilde{\mathbf{u}}_H}} \\
&= \frac{(\widetilde{\mathbf{u}}_H, \mathbf{u}_H)_{B_H}}{\|\mathbf{u}_H\|_{B_H}\|\widetilde{\mathbf{u}}_H\|_{B_H}} = \cos \angle_{B_H}(\widetilde{\mathbf{u}}_H, \mathbf{u}_H).
\end{aligned}$$

The Pythagorean identity leads to

$$\sin \angle_{B_h}(\mathbf{u}_H^h, \widehat{\mathbf{u}}_h) = \sin \angle_{B_H}(\widetilde{\mathbf{u}}_H, \mathbf{u}_H). \square$$

Now, we can obtain the final estimate.

Corollary 4.24. *Let $\mathbf{u}_H^h, \widehat{\mathbf{u}}_h, \widetilde{\mathbf{u}}_H, \mathbf{u}_H$ be defined as in preceding corollaries. Then*

$$\sin \angle(u_H, u_h) = \sin \angle_{B_h}(\mathbf{u}_H^h, \mathbf{u}_h) \leq \sin \angle_{B_H}(\widetilde{\mathbf{u}}_H, \mathbf{u}_H) + \sin \angle_{B_h}(\widehat{\mathbf{u}}_h, \mathbf{u}_h).$$

Proof. The triangle inequality

$$|\sin \angle_{B_h}(\widehat{\mathbf{u}}_h, \mathbf{u}_H^h) - \sin \angle_{B_h}(\mathbf{u}_H^h, \mathbf{u}_h)| \leq \sin \angle_{B_h}(\widehat{\mathbf{u}}_h, \mathbf{u}_h)$$

gives

$$-\sin \angle_{B_h}(\widehat{\mathbf{u}}_h, \mathbf{u}_h) \leq \sin \angle_{B_h}(\widehat{\mathbf{u}}_h, \mathbf{u}_H^h) - \sin \angle_{B_h}(\mathbf{u}_H^h, \mathbf{u}_h),$$

which can be written as

$$\sin \angle_{B_h}(\mathbf{u}_H^h, \mathbf{u}_h) \leq \sin \angle_{B_h}(\widehat{\mathbf{u}}_h, \mathbf{u}_H^h) + \sin \angle_{B_h}(\widehat{\mathbf{u}}_h, \mathbf{u}_h).$$

Therefore, from Corollary 4.22 and 4.23 we get

$$\sin \angle(u_H, u_h) = \sin \angle_{B_h}(\mathbf{u}_H^h, \mathbf{u}_h) \leq \sin \angle_{B_H}(\widetilde{\mathbf{u}}_H, \mathbf{u}_H) + \sin \angle_{B_h}(\widehat{\mathbf{u}}_h, \mathbf{u}_h).$$

$\square$

Corollary 4.25. *Let $\mathbf{u}_H$, $\widetilde{\mathbf{u}}_H$, $\mathbf{u}_h$, $\widetilde{\mathbf{u}}_h$, $\widehat{\mathbf{u}}_h$ be representation vectors of functions u_H, $\widetilde{u}_H$, u_h, $\widetilde{u}_h$, $\widehat{u}_h$, respectively, and let $\mathbf{r}_H$, $\mathbf{r}_h$, $\widehat{\mathbf{r}}_h$ be the corresponding residual vectors defined as in (4.8), (4.9) and (4.10). Then,*

$$\begin{aligned}
0 \leq |\sin \angle_{B_H}(\widetilde{\mathbf{u}}_H, \mathbf{u}_H)| &\leq \frac{1}{\delta_H}\frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}}, \\
0 \leq |\sin \angle_{B_h}(\widetilde{\mathbf{u}}_h, \mathbf{u}_h)| &\leq \frac{1}{\delta_h}\frac{\|\mathbf{r}_h\|_{B_h^{-1}}}{\|\widetilde{\mathbf{u}}_h\|_{B_h}}, \\
0 \leq |\sin \angle_{B_h}(\widehat{\mathbf{u}}_h, \mathbf{u}_h)| &\leq \frac{1}{\widehat{\delta}_h}\frac{\|\widehat{\mathbf{r}}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}}.
\end{aligned}$$

Proof. All bounds follow from Theorem 2.42. $\square$

In order to get a posteriori error estimates for angles between the eigenfunctions of interest, the previous results are combined.

Theorem 4.26. *Let u be the exact eigenfunction of (3.1) and let u_H (u_h) be the corresponding exact eigenfunction of the discretized problem on the space V_H (V_h), $\widetilde{u}_H$ its approximation and $\widehat{u}_h$ the prolongation of $\widetilde{u}_H$ on space V_h. If $\widetilde{\mathbf{u}}_H, \widehat{\mathbf{u}}_h$, $\mathbf{r}_H, \widehat{\mathbf{r}}_h$ are the corresponding representation vectors and residual vectors, respectively, then*

$$\sin\angle(u,u_H) \le \frac{1}{1-\gamma}\Big(\frac{1}{\delta_H}\frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}} + \frac{1}{\widehat{\delta}_h}\frac{\|\widehat{\mathbf{r}}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}}\Big),$$

$$\sin\angle(u,u_h) \le \frac{\gamma}{1-\gamma}\Big(\frac{1}{\delta_H}\frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}} + \frac{1}{\widehat{\delta}_h}\frac{\|\widehat{\mathbf{r}}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}}\Big).$$

Proof. From Corollary 4.17 and 4.22 it follows that

$$\sin\angle(u,u_H) \le \frac{1}{1-\gamma}\sin\angle(u_H,u_h) = \frac{1}{1-\gamma}\sin\angle(u_H^h,u_h) = \frac{1}{1-\gamma}\sin\angle_{B_h}(\mathbf{u}_H^h,\mathbf{u}_h).$$

By applying Corollary 4.24 we get

$$\sin\angle(u,u_H) \le \frac{1}{1-\gamma}\Big(\sin\angle_{B_H}(\widetilde{\mathbf{u}}_H,\mathbf{u}_H) + \sin\angle_{B_h}(\widehat{\mathbf{u}}_h,\mathbf{u}_h)\Big).$$

Finally, using Corollary 4.25 we obtain

$$\sin\angle(u,u_H) \le \frac{1}{1-\gamma}\Big(\frac{1}{\delta_H}\frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}} + \frac{1}{\widehat{\delta}_h}\frac{\|\widehat{\mathbf{r}}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}}\Big).$$

The proof for the second inequality can be obtained in a similar fashion. □

Theorem 4.27. *Let u be the exact eigenfunction of (3.1) and $\widetilde{u}_H$ its iterative approximation obtained in space V_H. For $\widehat{u}_h$ being a prolongation of $\widetilde{u}_H$ on the space V_h, representation vectors $\widetilde{\mathbf{u}}_H, \widehat{\mathbf{u}}_h$ and the corresponding residual vectors $\mathbf{r}_H, \widehat{\mathbf{r}}_h$, the following inequalities hold.*

$$\begin{aligned}
\sin\angle(u,\widetilde{u}_H) &\le \frac{2-\gamma}{1-\gamma}\frac{1}{\delta_H}\frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}} + \frac{1}{1-\gamma}\frac{1}{\widehat{\delta}_h}\frac{\|\widehat{\mathbf{r}}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}},\\
\sin\angle(u,\widehat{u}_h) &\le \frac{1}{1-\gamma}\frac{1}{\widehat{\delta}_h}\frac{\|\widehat{\mathbf{r}}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}} + \frac{\gamma}{1-\gamma}\frac{1}{\delta_H}\frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}}.
\end{aligned}$$

Proof. Corollary 4.19 implies that

$$\sin\angle(u,\widetilde{u}_H) \le \sin\angle(\widetilde{u}_H,u_H) + \frac{1}{1-\gamma}\sin\angle(u_H,u_h).$$

Hence, using Corollary 4.22 and 4.24, we have

$$\begin{aligned}
\sin\angle(u,\widetilde{u}_H) &\leq \sin\angle(\widetilde{u}_H,u_H)+\frac{1}{1-\gamma}\sin\angle(u_H,u_h)\\
&= \sin\angle_{B_H}(\widetilde{\mathbf{u}}_H,\mathbf{u}_H)+\frac{1}{1-\gamma}\sin\angle(u_H^h,u_h)\\
&= \sin\angle_{B_H}(\widetilde{\mathbf{u}}_H,\mathbf{u}_H)+\frac{1}{1-\gamma}\sin\angle_{B_h}(\mathbf{u}_H^h,\mathbf{u}_h)\\
&\leq \sin\angle_{B_H}(\widetilde{\mathbf{u}}_H,\mathbf{u}_H)+\frac{1}{1-\gamma}\Big(\sin\angle_{B_H}(\widetilde{\mathbf{u}}_H,\mathbf{u}_H)+\sin\angle_{B_h}(\widehat{\mathbf{u}}_h,\mathbf{u}_h)\Big)\\
&= \frac{2-\gamma}{1-\gamma}\sin\angle_{B_H}(\widetilde{\mathbf{u}}_H,\mathbf{u}_H)+\frac{1}{1-\gamma}\sin\angle_{B_h}(\widehat{\mathbf{u}}_h,\mathbf{u}_h).
\end{aligned}$$

The final result follows from Corollary 4.25

$$\sin\angle(u,\widetilde{u}_H)\leq\frac{2-\gamma}{1-\gamma}\frac{1}{\delta_H}\frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}}+\frac{1}{1-\gamma}\frac{1}{\widehat{\delta}_h}\frac{\|\widehat{\mathbf{r}}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}}.$$

For the second inequality the proof proceeds analogously. □

In summary, we have obtained bounds on the errors between the exact eigenpairs and their approximations considering both eigenvalue and eigenfunction errors. We immediately notice that all bounds except the one from Theorem 4.15 are computable from the coarse mesh information, and, under the saturation assumption, for problems with well-conditioned eigenvalues, a small residual vector is equivalent to a good accuracy of computed eigenpairs. Thus, in the discussed self-adjoint elliptic problem, these bounds and the corresponding residuals can be used to control the adaptation process via computed coarse mesh eigenvalues and eigenvectors.
In (2.32) the relation between the B_H-norm and the $L_2(\Omega)$-norm was shown, which allows us to get equivalent bounds using the $L_2(\Omega)$-norm. The only question may arise when the B_H^{-1}-norm of the residual has to be obtained. It this case the following simple relation holds

$$\|\cdot\|_{B_H^{-1}}=\|B_H^{-1}\cdot\|_{B_H},$$

which basically means solving a linear system with B_H.
In particular, the inclusion of the adaptation in the algebraic approach helps to avoid the usual assumption that the solution of the discrete problem is exact, to assure a good approximation via AFEM. However, the algebraic adaptation alone cannot assure that the refinement procedure will lead to convergence. If the analytic approach with the standard assumption of solving the algebraic problems exactly does not converge, then neither does our extended algorithm. However, the AFEMLA approach makes the adaptation process much more efficient with guaranteed computable bounds in the algebraic part. This will be demonstrated in the next section.

4.2.4 Numerical experiments

In this section, we present some numerical results that illustrate our algorithm. The numerical tests were realized with help of the finite element framework OPENFFW [29]. We consider the model eigenvalue problem (3.1) on different domains Ω.
As described in Section 4.2.2, the mesh refinement will be based on the entries of the residual vector $\widehat{\mathbf{r}}_h$, which can capture the behavior of the solution only when the (prolongated) eigenvector approximation is sufficiently accurate. Therefore, error bounds for eigenfunctions (eigenvectors) derived in Section 4.2.3 have crucial importance since they guarantee the quality of approximate eigenfunctions (eigenvectors).

L-shape domain

Let us consider the eigenvalue problem (3.1) with the L-shape domain $\Omega = [-1,1] \times [0,1] \cup [-1,0] \times [-1,0]$. An approximation of the smallest eigenvalue was given in [108], where the authors obtained that

$$\lambda_1 \approx 9.639723844.$$

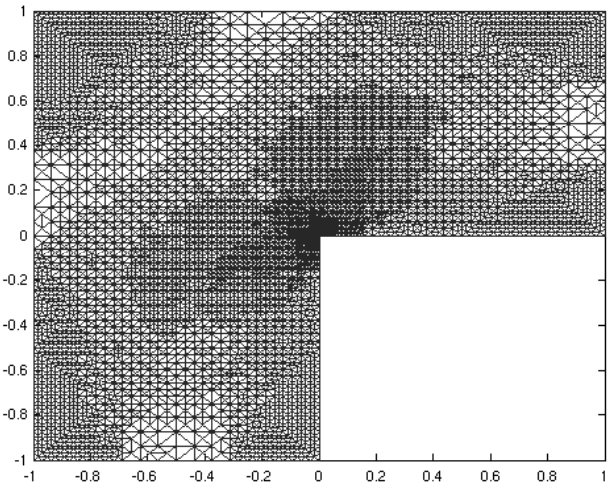


Figure 4.18: The AFEMLA mesh on the 8th refinement level of the L-shape domain.

Figure 4.18 shows the adaptively refined mesh on the 8th level of refinement. We note that the mesh constructed by our algorithm contains more elements around the singularity. In Figure 4.19, a log-log plot of the approximation error $|\lambda_1 - \widetilde{\lambda}_1|$ versus the number of degrees of freedom is presented. Here the discrete eigenvalue problem is solved using the MATLAB function *eigs* [89]. The squares show the approximation error based on the solution obtained on the uniformly refined grid, while the triangles illustrate the approximation on the residual based refined grid. This shows that, with the adaptive algorithm AFEMLA, we may reach the same accuracy of the computed eigenvalue with much fewer degrees of freedom. Tables 4.2 and 4.3 present the convergence history data for both strategies. Comparing the last columns

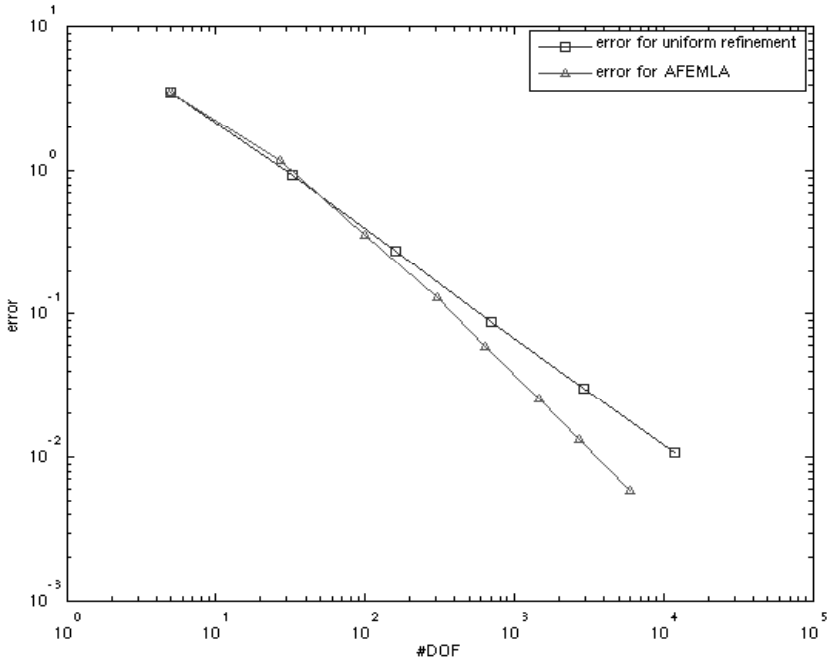


Figure 4.19: Convergence history for the uniform refinement and the AFEMLA for (3.1) on the L-shape domain.

of both tables, where we present the MATLAB CPU times of running the iterative eigenvalue solver for both algorithms, we notice that to reach the accuracy 10^{-2} for the first algorithm we have to work with 12033 degrees of freedom and spend 7.6 s in solving the algebraic eigenvalue problem, while we need only 2745 degrees of freedom and 0.7 s in the case of our algorithm. Note that in the following plots and tables we always use $\widetilde{\lambda}$ to denote the approximation of the eigenvalue, we do not distinguish between $\widetilde{\lambda}$ and $\widehat{\lambda}$, since the difference would be invisible in these examples.

Table 4.2: Approximations of the smallest eigenvalue of (3.1) on the uniformly refined L-shape domain.

ref. level	#DOF	$\widetilde{\lambda}_1$	$\lvert\lambda_1 - \widetilde{\lambda}_1\rvert$	CPU time (s)
1	5	13.1992	3.5595	0.01
2	33	10.5740	0.9342	0.03
3	161	9.9165	0.2768	0.04
4	705	9.7284	0.0886	0.16
5	2945	9.6698	0.0301	0.90
6	12033	9.6504	0.0107	7.60

The second goal of our algorithm is to reduce the computational costs of solving the algebraic eigenvalue problem. This can be done by restricting the number p of basis vectors for the Krylov space using the corresponding parameter in the MATLAB function *eigs* [89].

Table 4.3: Approximations of the smallest eigenvalue determined by the AFEMLA for (3.1) on the L-shape domain.

ref. level	#DOF	$\widetilde{\lambda}_1$	$\lvert\lambda_1 - \widetilde{\lambda}_1\rvert$	CPU time (s)
1	5	13.1992	3.5595	0.02
2	27	10.8173	1.1775	0.02
3	99	9.9982	0.3584	0.03
4	306	9.7721	0.1323	0.07
5	641	9.6982	0.0585	0.14
6	1461	9.6652	0.0255	0.33
7	2745	9.6528	0.0131	0.70
8	5961	9.6455	0.0058	2.14

Figures 4.20 and 4.21 present numerical examples in this direction.

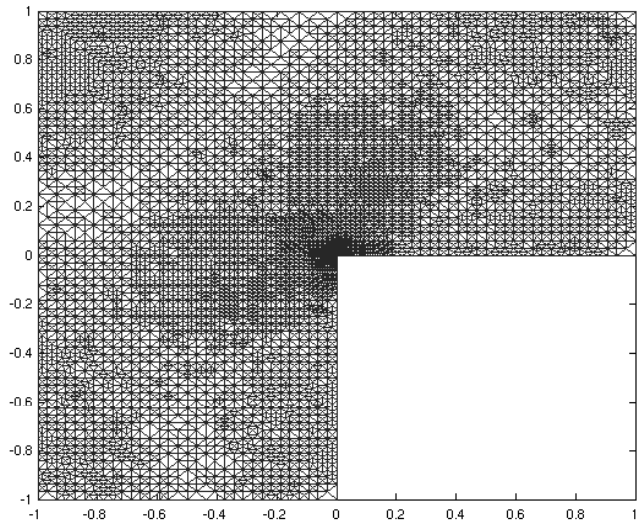


Figure 4.20: The AFEMLA mesh on the 8th refinement level of the L-shape domain with $p = 2$ basis vectors in the Krylov space.

Table 4.4 presents results for our algorithm using only $p = 2$ basis vectors for the Krylov space. In this case the required number of degrees of freedom has slightly increased. Moreover we observe a slower decrease of the error, see Figure 4.21. However, if we run the standard algorithm on the uniformly refined mesh using only $p = 2$ basis vectors, then we do not get any reasonable approximation of the eigenvalue, we get zero on each level. Of course (hopefully) at some point entries in the residual vector will be very small indicating that we have refined the grid sufficiently. This is the moment when the Krylov method should be run up to full desired accuracy.

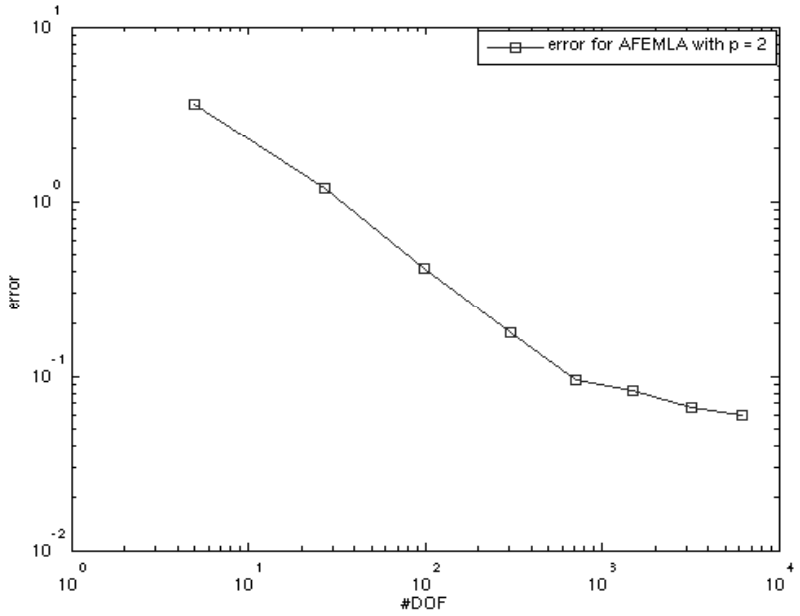


Figure 4.21: Convergence history of the AFEMLA for (3.1) on the L-shape domain with $p = 2$ basis vectors in the Krylov space.

Table 4.4: Approximations of the smallest eigenvalue of (3.1) determined by the AFEMLA on the L-shape domain with $p = 2$ basis vectors in the Krylov space.

ref. level	#DOF	$\widetilde{\lambda}_1$	$\lvert\lambda_1 - \widetilde{\lambda}_1\rvert$	CPU time (s)
1	5	13.2156	3.5758	0.01
2	27	10.8390	1.1993	0.01
3	98	10.0582	0.4184	0.02
4	305	9.8166	0.1769	0.06
5	712	9.7346	0.0949	0.14
6	1491	9.7221	0.0824	0.33
7	3186	9.7055	0.0657	0.82
8	6167	9.6991	0.0593	2.14

Dependence on the number of prescribed Krylov basis vectors

A natural question is, whether we can choose the number of basis vectors for the Krylov space in an optimal way during the adaptive refinement process. In Figure 4.22 we present the convergence history of the AFEMLA scheme using different numbers of basis vectors. We see that there is almost no difference in accuracy for $p = 2, 3, 4, 5$, so there is no reason to take more than $p = 2$ basis vectors. Figures 4.22 and 4.23 present the convergence behavior of the AFEMLA with respect to the mesh size and the number of prescribed Krylov basis vectors p. It is important to notice that the accuracy of approximation is clearly determined by the mesh properties and it is influenced only slightly by the choice of p. Of course the optimal value of p could be determined, but as we observe, choosing a large p for each refinement step is not necessary. Figure 4.23 shows that with a significant increase of the number of vectors, i.e., $p = 5, 10, 15, 20$, for the initial AFEMLA steps the accuracy will be affected only slightly.
In Figure 4.24 the accuracy of the eigenvalue for a fixed mesh (fixed #DOF) is considered in comparison to the number of prescribed basis vectors. Notice that for a given mesh, increasing the number of Krylov basis vectors does not necessarily imply better approximation, which means that for most (in particular the early) AFEMLA steps we can choose a small number of Krylov basis vectors p and at the end of the adaptation process, when the approximate finite dimensional eigenvalue problem yields a very good approximation, the number of basis vectors should be increased in order to reach the final desired accuracy. More specifically, if the adaptively constructed grid guarantees a certain size of the discretization error or if a maximal number of allowed degrees of freedom is reached, the final accuracy of the solution is obtained by performing additional iterations as long as the forward error is larger than a desired accuracy ε.

More complicated domains

Let us consider the eigenvalue problem (3.1) with the domain Ω as in Figure 4.25. Since in this case we do not know a priori any good approximation of the smallest eigenvalue, for comparison we will use the values obtained by $eigs$ [89], with default values, on the uniformly refined grid.
Comparing the values listed in Table 4.5 and 4.6 we see that to obtain the approximation $\widetilde{\lambda}_1 \approx 11.97$ using uniformly refined grids we need around 69825 degrees of freedom, while in our algorithm we work with 9584 degrees of freedom. Moreover, when we look at the CPU time of the iterative procedure, we need around 788 s on the uniform grid, while in the AFEMLA case only 4.6 s. We also see from the adaptively refined grid in Figure 4.25 that our algorithm clearly recognizes the critical regions of the domain.

More eigenvalues - refinement based on all residual vectors

Often one is interested in more than one eigenvalue. But usually eigenfunctions associated with different eigenvalues behave quite different analytically. For this reason, when more

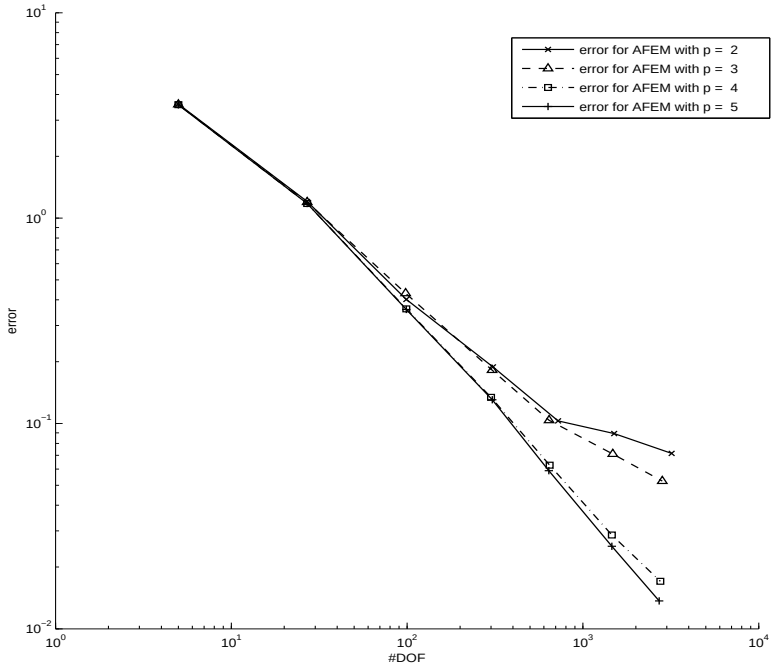


Figure 4.22: Convergence history for the AFEMLA for (3.1) on the L-shape domain for $p = 2, 3, 4, 5$.

Table 4.5: Approximations of the smallest eigenvalue of (3.1) obtained by the standard FEM on the uniformly refined domain with holes.

ref. level	#DOF	$\widetilde{\lambda}_1$	CPU time (s)
1	44	13.6075	0.26
2	225	12.5102	0.07
3	1001	12.1579	0.23
4	4209	12.0352	1.43
5	17249	11.9905	15.45
6	69825	11.9737	788.02

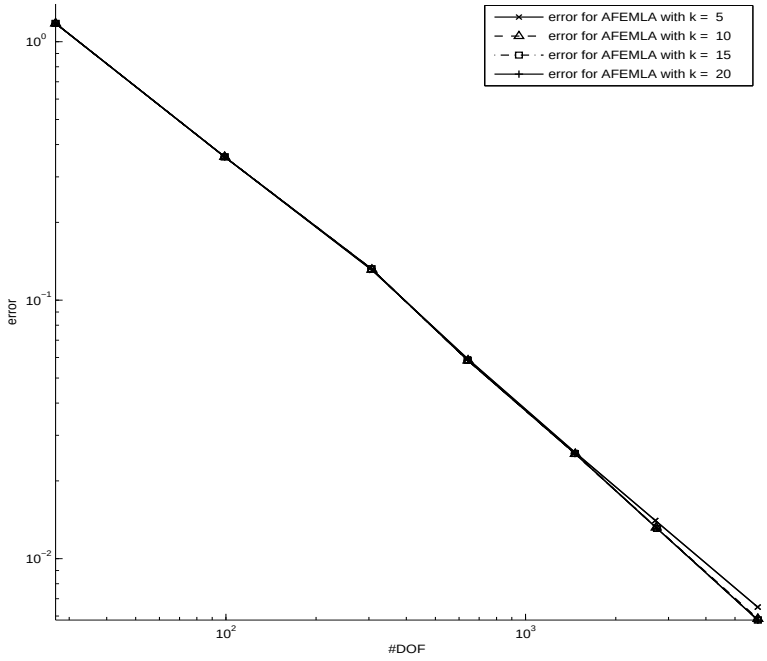


Figure 4.23: Convergence history for the AFEMLA for (3.1) on the L-shape domain for $p = 5, 10, 15, 20$.

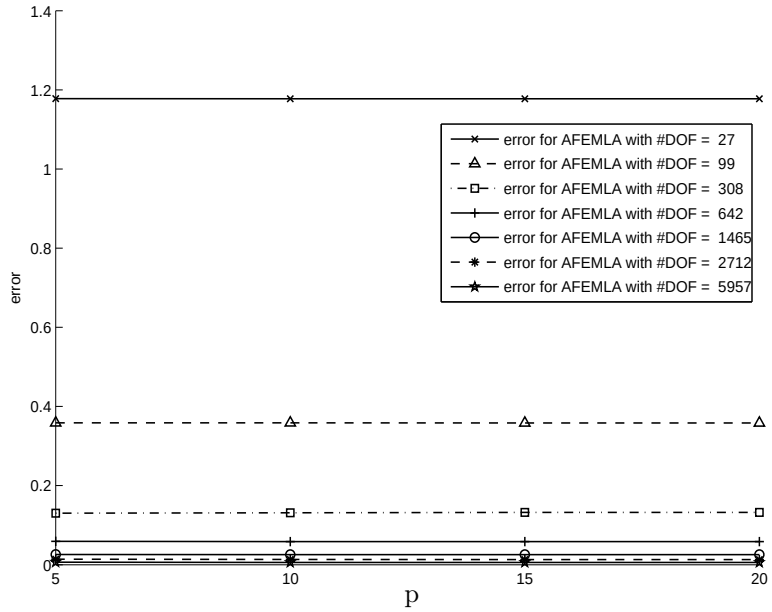


Figure 4.24: Convergence history for the AFEMLA for (3.1) on the L-shape domain with different number of Krylov basis vectors, $p = 5, 10, 15, 20$, on a fixed mesh.

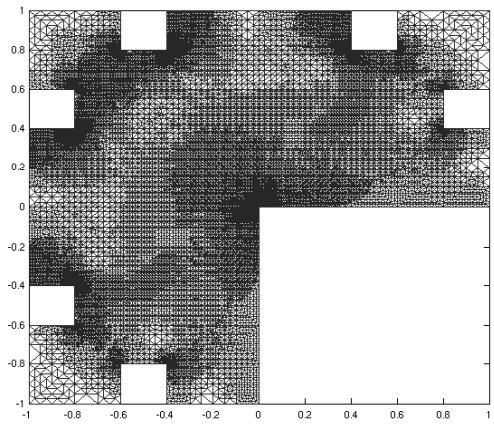


Figure 4.25: The AFEMLA mesh on the 8th refinement level.

Table 4.6: Approximations of the smallest eigenvalue of (3.1) obtained by the AFEMLA on the domain with holes.

ref. level	#DOF	$\widetilde{\lambda}_1$	CPU time (s)
1	44	13.6075	0.03
2	110	12.6818	0.04
3	262	12.2930	0.06
4	552	12.1340	0.12
5	1112	12.0474	0.25
6	2348	12.0037	0.59
7	4622	11.9834	1.49
8	9584	11.9728	4.60

Table 4.7: Approximations of the three smallest eigenvalues of (3.1) on the L-shape domain obtained by the AFEMLA.

ref. level	1	2	3	4	5	6	7	8
#DOF	5	33	133	465	1306	2770	4997	11499
$\widetilde{\lambda}_1$	13.1992	10.5542	9.9192	9.7376	9.6817	9.6591	9.6496	9.6440
$\widetilde{\lambda}_2$	22.0215	16.9097	15.6315	15.3211	15.2421	15.2184	15.2085	15.2024
$\widetilde{\lambda}_3$	32.0000	22.9075	20.5262	19.9515	19.8089	19.7760	19.7569	19.7482

Table 4.8: Errors corresponding to the three smallest eigenvalues of (3.1) on the L-shape domain obtained by the AFEMLA.

ref. level	1	2	3	4	5	6	7	8
$\lvert\lambda_1 - \widetilde{\lambda}_1\rvert$	3.5595	0.9144	0.2795	0.0979	0.0420	0.0194	0.0099	0.0043
$\lvert\lambda_2 - \widetilde{\lambda}_2\rvert$	6.8242	1.7125	0.4342	0.1239	0.0448	0.0211	0.0112	0.0051
$\lvert\lambda_3 - \widetilde{\lambda}_3\rvert$	12.2608	3.1683	0.7870	0.2123	0.0697	0.0367	0.0177	0.0090

than one eigenvalue is desired, typically the adaptation process for each eigenvalue is different. This is another place where the algebraic adaptation process has many advantages, since the marking strategy used in the AFEMLA algorithm, based on entries of the residual vector, can be easily extended to use several residual vectors corresponding to different eigenvalues. The marking procedure will identify large entries in all residual vectors and take the union of the corresponding basis functions. Table 4.7 contains approximations of the three smallest eigenvalues of the problem (3.1) obtained by the AFEMLA, additionally in Table 4.8 the errors $|\lambda_i - \widetilde{\lambda}_i|$, for $i = 1, 2, 3$ are presented. The corresponding adaptively refined grid is depicted in Figure 4.26, while all three eigenfunctions are presented in Figure 4.27. The convergence history is presented in Figure 4.28. Approximations of the three smallest eigenvalues of (3.1) are given in [108] as

$$\lambda_1 \approx 9.639723844, \quad \lambda_2 \approx 15.197252, \quad \lambda_3 \approx 19.739209.$$

A natural question arises whether in the case of approximating several eigenvalues at once, adaptive methods have at all an advantage over the standard FEM, since they may lead to almost uniformly refined grids. Of course choosing a very fine mesh at the beginning typically leads to very good approximations of some eigenvalues. However, we do not know a priori how fine the grid should be and if the actually chosen mesh is appropriate for the problem, since this depends strongly on the support of each eigenfunction. Secondly, the adaptive algorithm constructs a given mesh with fewer degrees of freedom, which will still guarantee the desired accuracy.

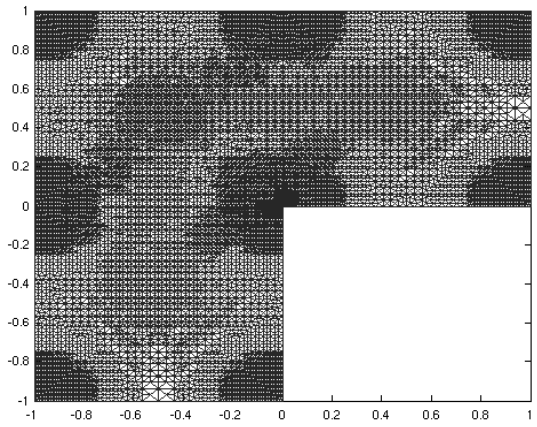


Figure 4.26: The AFEMLA mesh for the 8th level of refinement in computing the three smallest eigenvalues on the L-shape domain.

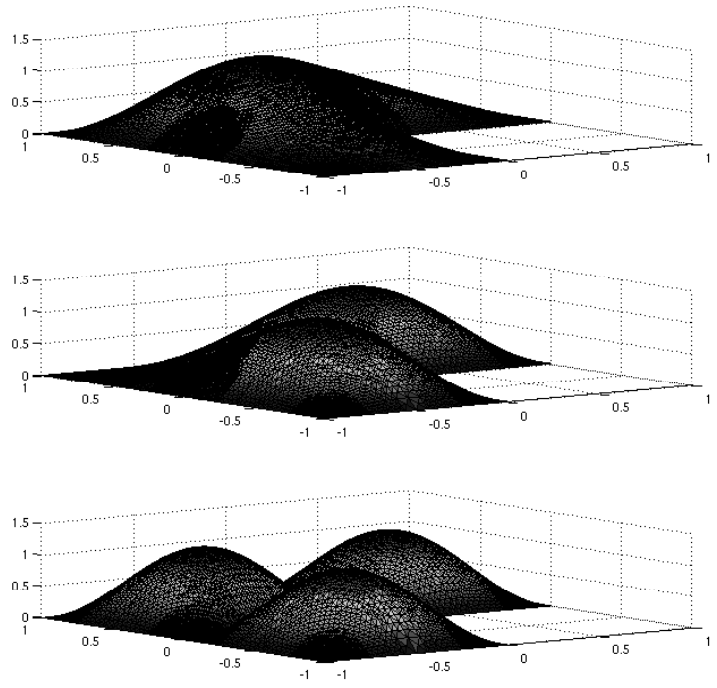


Figure 4.27: Computed eigenfunctions corresponding to the eigenvalue $\widetilde{\lambda}_1 = 9.644$, $\widetilde{\lambda}_2 = 15.2024$, $\widetilde{\lambda}_3 = 19.7482$, respectively.

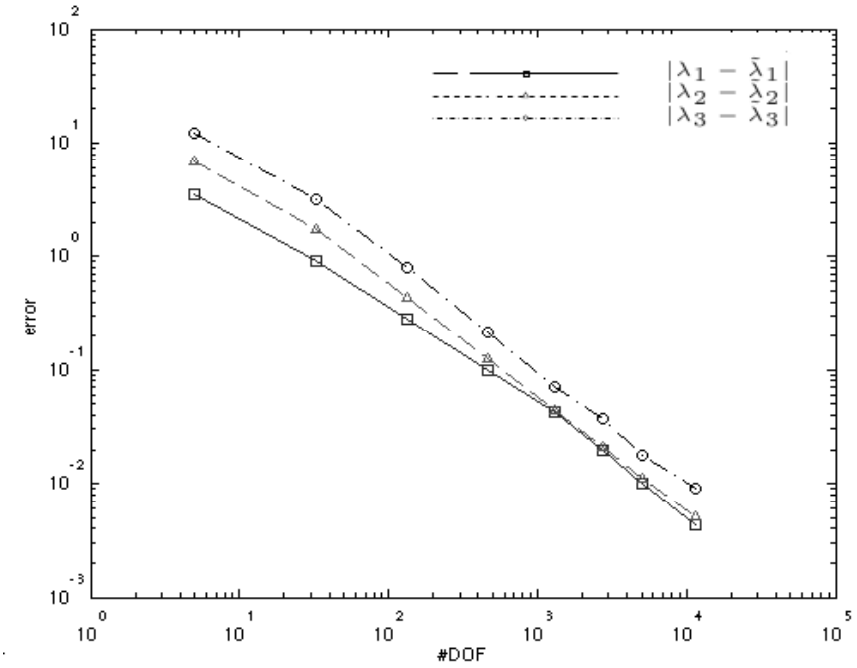


Figure 4.28: Convergence history for the AFEMLA for computing the three smallest eigenvalues of (3.1) on the L-shape domain.

Results for the L-shape with large number of DOF and estimator evaluation

In the case of a large number of degrees of freedom ($10^5 - 10^6$), the optimal linear convergence rate of the AFEMLA algorithm can be observed as shown in Figure 4.29. Corresponding eigenvalue approximations, grids and CPU time information are presented in Table 4.9. Figure 4.30 indicates that the error estimator η_{AFEMLA}, based on the bound from Theorem 4.14, is indeed an upper bound for the error and preserves the optimal convergence rate. Note that the estimated error decreases in each step of the adaptive procedure.

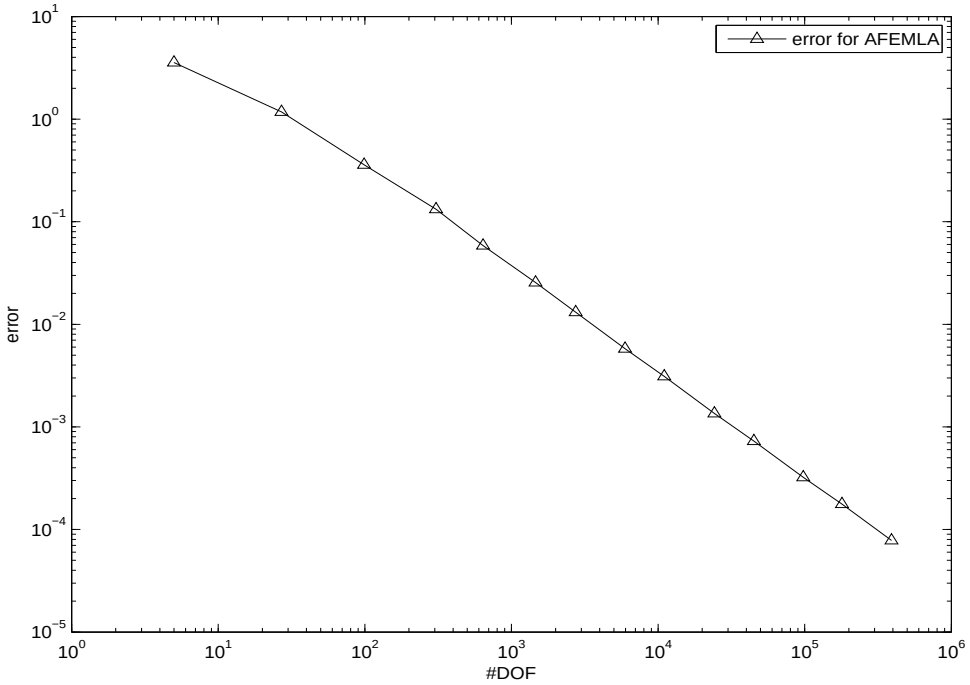


Figure 4.29: Convergence history of the AFEMLA for (3.1) on the L-shape domain with #DOF up to 10^6.

Table 4.9: Approximations of the smallest eigenvalue determined by the AFEMLA for (3.1) on the L-shape domain with #DOF up to 10^6.

ref. level	#DOF	$\widetilde{\lambda}_1$	$\lvert\lambda_1 - \widetilde{\lambda}_1\rvert$	CPU time (s)
1	5	13.199179221542078	3.5595	0.4699
2	27	10.817272738817195	1.1775	0.0280
3	99	9.998164924361316	0.3584	0.0170
4	306	9.772068757969148	0.1323	0.0204
5	641	9.698244102424695	0.0585	0.0253
6	1461	9.665242314196133	0.0255	0.0469
7	2745	9.652824991492565	0.0131	0.0928
8	5961	9.645514134388604	0.0058	0.2163
9	11013	9.642828562681894	0.0031	0.4729
10	24202	9.641077917919095	0.0014	1.1375
11	45062	9.640451575223047	7.2773e-04	2.4259
12	97698	9.640045863592908	3.2202e-04	5.7857
13	179461	9.639900769884797	1.7693e-04	11.2883
14	391319	9.639801977462790	7.8133e-05	34.0399

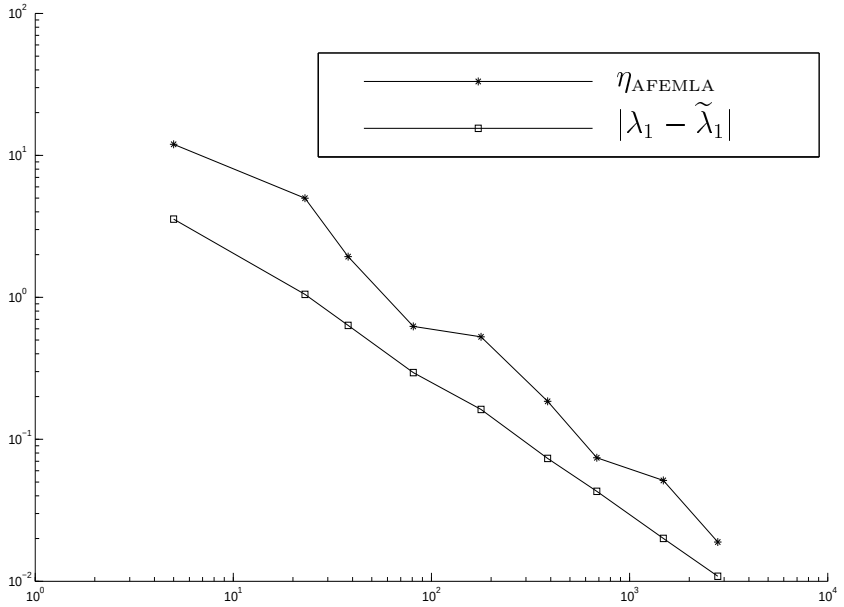


Figure 4.30: Eigenvalue error $|\lambda_1 - \widetilde{\lambda}_1|$ and estimated eigenvalue error η_{AFEMLA} of the AFEMLA for (3.1) on the L-shape domain.

4.3 Functional perturbation results for PDE eigenvalue problems

In Section 2.3.3 the backward error and the condition number were introduced in the context of analyzing the approximation error and the backward stability of the algebraic eigenvalue problem. Here, we are interested in applying this theoretical framework in oder to analyze the discretization and the iteration error at the same time. In [9], Arioli et. al. introduce *functional backward errors* and the so-called *Compatibility Theorem* for boundary value problems. This section describes the extension of this result to eigenvalue problems. Functional backward errors and condition numbers are used to analyze the continuous dependence of the inexact solution on the data. Although, all the proofs do not require any special assumptions about the inexact eigenpair $(\widetilde{\lambda}, \widetilde{u})$, we are particularly interested in the eigenpair approximation resulting from the iterative eigensolver. The theoretical framework we adopt here is taken from [9].

4.3.1 The functional backward error and condition number

Let H be a Hilbert space and V its closed subspace. i.e, $V \subset H$. Let $\mathcal{BL}(V)$, $\mathcal{BL}(H)$ define the space of continuous bilinear forms $V \times V \to \mathbb{R}$ and $H \times H \to \mathbb{R}$, respectively. We first describe the functional backward error analysis and the functional condition number for the self-adjoint PDE eigenvalue problem in its variational formulation:

Find $\lambda \in \mathbb{R}$ *such that there exist* $u \in V$, $u \neq 0$ *satisfying*

$$a(u, v) = \lambda b(u, v), \text{ for all } v \in V, \tag{4.26}$$

where $a(\cdot, \cdot) \in \mathcal{BL}(V)$ *and* $b(\cdot, \cdot) \in \mathcal{BL}(H)$.
For the sake of simplicity, since $V \subset H$ and $b(\cdot, \cdot)$ is also a bilinear form on V, we restrict ourself to the space V. We suppose that $a(\cdot, \cdot)$, $b(\cdot, \cdot)$ are bounded on V and $a(\cdot, \cdot)$ is V-elliptic, namely

$$\begin{aligned} a(u, v) &\leq C_1 \|u\|_V \|v\|_V, && \text{for all } u, v \in V, \text{ with some constant } C_1 > 0, \\ b(u, v) &\leq C_2 \|u\|_V \|v\|_V, && \text{for all } u, v \in V, \text{ with some constant } C_2 > 0, \\ a(v, v) &\geq C_3 \|v\|_V^2, && \text{for all } v \in V, \text{ with some constant } C_3 > 0. \end{aligned} \tag{4.27}$$

The underlying theoretical framework of the backward error analysis, introduced in Section 2.1.6, in functional spaces has the following form.

Definition 4.28 (The functional condition number and backward error)**.** Let a, b be bilinear forms on V. Consider a simple eigenpair (λ, u) of the problem (4.26) and let φ be the mapping

$$\varphi : \quad (a, b) \quad \to \quad (\lambda, u).$$

Furthermore, let $(\widetilde{\lambda}, \widetilde{u}) \in \mathbb{R} \times V$ be an approximation of the eigenpair (λ, u). Then the functional condition number of φ is

$$\mathcal{C} = \limsup_{\substack{\delta a \to 0 \\ \delta b \to 0}} \frac{\|(\widetilde{\lambda}, \widetilde{u}) - (\lambda, u)\|_V}{\|(\delta a, \delta b)\|_{\mathcal{BL}(V)}},$$

and the normwise functional backward error associated with $(\widetilde{\lambda}, \widetilde{u})$ is

$$\eta = \min\{\varepsilon > 0; \|\delta a\|_{\mathcal{BL}(V)} \leq \varepsilon \|a\|_{\mathcal{BL}(V)}, \ \|\delta b\|_{\mathcal{BL}(V)} \leq \varepsilon \|b\|_{\mathcal{BL}(V)}$$

$$\text{such that} \quad (a + \delta a)(\widetilde{u}, v) = \widetilde{\lambda}(b + \delta b)(\widetilde{u}, v), \text{ for all } v \in V\},$$

where $\delta a, \delta b \in \mathcal{BL}(V)$ are perturbations of the bilinear form a and b, respectively, and $\varepsilon \in \mathbb{R}^+$.

The mapping φ describes how the eigenpair (λ, u) is determined by bilinear forms a, b. Similarly, as in the discrete case, the condition number is the smallest upper bound for the ratio between the error in the output, i.e., eigenpair, and the error in the input, i.e., perturbations in both bilinear forms [36]. The normwise functional backward error determines the size of the smallest perturbation in bilinear forms, such that the approximate eigenpair is the exact eigenpair of the perturbed variational equation. However, here the situation is much more complicated. Since, usually, the approximate eigenpair $(\widetilde{\lambda}, \widetilde{u})$ is a finite element solution obtained after discretizing the PDE problem and applying the iterative eigensolver to the finite dimensional problem, the mapping φ in fact is a composite of two maps, i.e.,

$$\varphi((a,b)) = \varphi_{iter} \circ \varphi_{disc}((a,b)) = (\lambda, u).$$

Applying perturbation arguments leads to

$$\varphi((a+\delta a, b+\delta b)) = \varphi_{iter} \circ \varphi_{disc}((a+\delta a, b+\delta b)) = (\widetilde{\lambda}, \widetilde{u}).$$

Therefore, the error between the exact continuous eigenpair (λ, u) and its finite element approximation $(\widetilde{\lambda}, \widetilde{u})$ can be written as

$$\|(\widetilde{\lambda}, \widetilde{u}) - (\lambda, u)\|_V \approx \mathcal{C}_{iter}\varphi_{iter}\Big(\mathcal{C}_{disc}\varphi_{disc}((a+\delta a, b+\delta b)) + error\Big).$$

In the ideal situation, when the map φ is linear, obtaining the condition number and the backward error which combine information from the discretization and the iteration process would be relatively easy. However, eigenvalue problems are nonlinear, i.e., φ is a nonlinear map and the condition number of the problem does not depend linearly on two easily determined condition numbers $\mathcal{C}_{disc}$ and $\mathcal{C}_{iter}$. Therefore, it seems natural to analyze the variational formulation without any assumptions about the exactness of the finite element solution.
As a starting point let us determine the normwise functional backward error as a generalization of [20, Theorem 3.1].

Theorem 4.29. *Let a, b be bilinear forms on V, i.e., $a, b \in \mathcal{BL}(\mathcal{V})$. Furthermore, let $\delta a \in \mathcal{BL}(\mathcal{V})$ and $\delta b \in \mathcal{BL}(\mathcal{V})$ be perturbations of a and b, respectively, such that*

$$\|\delta a\|_{\mathcal{BL}(\mathcal{V})} \leq \varepsilon \|a\|_{\mathcal{BL}(\mathcal{V})}, \quad \|\delta b\|_{\mathcal{BL}(\mathcal{V})} \leq \varepsilon \|b\|_{\mathcal{BL}(\mathcal{V})},$$

with $\varepsilon \in \mathbb{R}^+$. Then, the normwise functional backward error associated with the approximate eigenpair $(\widetilde{\lambda}, \widetilde{u})$ of (4.26) is

$$\eta = \frac{\|R(v)\|_{V'}}{(|\widetilde{\lambda}|\|b\|_{\mathcal{BL}(V)} + \|a\|_{\mathcal{BL}(V)})\|\widetilde{u}\|_V}, \tag{4.28}$$

with the residual $R(v) = a(\widetilde{u}, v) - \widetilde{\lambda} b(\widetilde{u}, v) \in V'$.

Proof. $\Rightarrow$ From the backward error analysis we have the following identity

$$(a+\delta a)(\widetilde{u}, v) = \widetilde{\lambda}(b+\delta b)(\widetilde{u}, v),$$

which corresponds to

$$R(v) = -\delta a(\widetilde{u}, v) + \widetilde{\lambda}\delta b(\widetilde{u}, v).$$

Moreover, for all $v \in V$, the following relation holds

$$|R(v)| = |-\delta a(\widetilde{u}, v) + \widetilde{\lambda}\delta b(\widetilde{u}, v)| \leq \|\delta a\|_{\mathcal{BL}(V)}\|\widetilde{u}\|_V\|v\|_V + |\widetilde{\lambda}|\|\delta b\|_{\mathcal{BL}(V)}\|\widetilde{u}\|_V\|v\|_V.$$

Hence, we have for all $v \in V$

$$\frac{|R(v)|}{\|v\|_V} \leq \|\delta a\|_{\mathcal{BL}(V)}\|\widetilde{u}\|_V + |\widetilde{\lambda}|\|\delta b\|_{\mathcal{BL}(V)}\|\widetilde{u}\|_V.$$

Therefore, it is obvious that

$$\sup_{v\in V\setminus\{0\}} \frac{|R(v)|}{\|v\|_V} \leq \|\delta a\|_{\mathcal{BL}(V)}\|\widetilde{u}\|_V + |\widetilde{\lambda}|\|\delta b\|_{\mathcal{BL}(V)}\|\widetilde{u}\|_V.$$

We note that from Definition 2.3 of the dual norm it follows that

$$\|R(v)\|_{V'} \leq \|\delta a\|_{\mathcal{BL}(V)}\|\widetilde{u}\|_V + |\widetilde{\lambda}|\|\delta b\|_{\mathcal{BL}(V)}\|\widetilde{u}\|_V.$$

With assumptions on $\|\delta a\|_{\mathcal{BL}(V)}$, $\|\delta b\|_{\mathcal{BL}(V)}$ we have that

$$\|R(v)\|_{V'} \leq \varepsilon\|a\|_{\mathcal{BL}(V)}\|\widetilde{u}\|_V + \varepsilon|\widetilde{\lambda}|\|b\|_{\mathcal{BL}(V)}\|\widetilde{u}\|_V.$$

Therefore,

$$\frac{\|R(v)\|_{V'}}{\big(|\widetilde{\lambda}|\|b\|_{\mathcal{BL}(V)} + \|a\|_{\mathcal{BL}(V)}\big)\|\widetilde{u}\|_V} \leq \varepsilon$$

Since this inequality holds for an arbitrarily chosen ε, it holds also for η, i.e.,

$$\frac{\|R(v)\|_{V'}}{\big(|\widetilde{\lambda}|\|b\|_{\mathcal{BL}(V)} + \|a\|_{\mathcal{BL}(V)}\big)\|\widetilde{u}\|_V} \leq \eta.$$

$\Leftarrow$

Let δa, δb be defined as follows

$$\delta a(u,v) = -\frac{\|a\|_{\mathcal{BL}(V)}}{\|a\|_{\mathcal{BL}(V)}\|\widetilde{u}\|_V + \|b\|_{\mathcal{BL}(V)}\widetilde{\lambda}\|\widetilde{u}\|_V}\|u\|_V R(v)$$

and

$$\delta b(u,v) = \frac{\|b\|_{\mathcal{BL}(V)}}{\|a\|_{\mathcal{BL}(V)}\|\widetilde{u}\|_V + \|b\|_{\mathcal{BL}(V)}\widetilde{\lambda}\|\widetilde{u}\|_V}\|u\|_V R(v).$$

Since $\widetilde{\lambda} > 0$, we have from the definition of the norm of the bilinear form (2.7) and the dual norm (2.3) that

$$\begin{aligned}\|\delta a\|_{\mathcal{BL}(V)} &= \sup_{u,v\in V}\frac{|\delta a(u,v)|}{\|u\|_V\|v\|_V} = \frac{\|a\|_{\mathcal{BL}(V)}}{\|a\|_{\mathcal{BL}(V)}\|\widetilde{u}\|_V + \|b\|_{\mathcal{BL}(V)}|\widetilde{\lambda}|\|\widetilde{u}\|_V}\sup_{v\in V}\frac{|R(v)|}{\|v\|_V}\\ &= \frac{\|a\|_{\mathcal{BL}(V)}}{\|a\|_{\mathcal{BL}(V)}\|\widetilde{u}\|_V + \|a\|_{\mathcal{BL}(V)}\widetilde{\lambda}\|\widetilde{u}\|_V}\|R(v)\|_{V'}\end{aligned}$$

and

$$\begin{aligned}\|\delta b\|_{\mathcal{BL}(V)} &= \sup_{u,v\in V}\frac{|\delta b(u,v)|}{\|u\|_V\|v\|_V} = \frac{\|b\|_{\mathcal{BL}(V)}}{\|a\|_{\mathcal{BL}(V)}\|\widetilde{u}\|_V + \|b\|_{\mathcal{BL}(V)}|\widetilde{\lambda}|\|\widetilde{u}\|_V}\sup_{v\in V}\frac{|R(v)|}{\|v\|_V}\\ &= \frac{\|b\|_{\mathcal{BL}(V)}}{\|a\|_{\mathcal{BL}(V)}\|\widetilde{u}\|_V + \|b\|_{\mathcal{BL}(V)}\widetilde{\lambda}\|\widetilde{u}\|_V}\|R(v)\|_{V'}.\end{aligned}$$

Therefore, the following inequalities hold

$$\|\delta a\|_{\mathcal{BL}(V)} \le \varepsilon\|a\|_{\mathcal{BL}(V)} \quad \text{and} \quad \|\delta b\|_{\mathcal{BL}(V)} \le \varepsilon\|b\|_{\mathcal{BL}(V)},$$

with

$$\varepsilon = \frac{\|R(v)\|_{V'}}{\big(|\widetilde{\lambda}|\|b\|_{\mathcal{BL}(V)} + \|a\|_{\mathcal{BL}(V)}\big)\|\widetilde{u}\|_V}.$$

Moreover,

$$\begin{aligned}(a+\delta a)(\widetilde{u},v) &= a(\widetilde{u},v) + \delta a(\widetilde{u},v)\\ &= a(\widetilde{u},v) - \frac{\|a\|_{\mathcal{BL}(V)}}{\|a\|_{\mathcal{BL}(V)}\|\widetilde{u}\|_V + \widetilde{\lambda}\|b\|_{\mathcal{BL}(V)}\|\widetilde{u}\|_V}\|\widetilde{u}\|_V R(v)\\ &= R(v) + \widetilde{\lambda}b(\widetilde{u},v) - \frac{\|a\|_{\mathcal{BL}(V)}}{\|a\|_{\mathcal{BL}(V)}\| + \widetilde{\lambda}\|b\|_{\mathcal{BL}(V)}}R(v)\\ &= \widetilde{\lambda}b(\widetilde{u},v) + \frac{R(v)\|a\|_{\mathcal{BL}(V)}\| + R(v)\widetilde{\lambda}\|b\|_{\mathcal{BL}(V)}\| - R(v)\|a\|_{\mathcal{BL}(V)}}{\|a\|_{\mathcal{BL}(V)}\| + \widetilde{\lambda}\|b\|_{\mathcal{BL}(V)}}\\ &= \widetilde{\lambda}b(\widetilde{u},v) + \frac{R(v)\widetilde{\lambda}\|b\|_{\mathcal{BL}(V)}}{\|a\|_{\mathcal{BL}(V)}\| + \widetilde{\lambda}\|b\|_{\mathcal{BL}(V)}}\\ &= \widetilde{\lambda}b(\widetilde{u},v) + \widetilde{\lambda}\delta b(\widetilde{u},v).\end{aligned}$$

Thus

$$(a+\delta a)(\widetilde{u},v) = \widetilde{\lambda}(b+\delta b)(\widetilde{u},v)$$

and

$$\eta \le \varepsilon = \frac{\|R(v)\|_{V'}}{\big(|\widetilde{\lambda}|\|b\|_{\mathcal{BL}(V)} + \|a\|_{\mathcal{BL}(V)}\big)\|\widetilde{u}\|_V}. \quad \square$$

Following [9] with the functional backward error analysis we prove the existence of the perturbations δa and δb such that the approximate eigenpair $(\widetilde{\lambda},\widetilde{u})$ of (4.26) is the exact solution of the same variational equation perturbed with δa and δb. This result can be stated as the *Eigenvalue Functional Compatibility Theorem.*

Theorem 4.30 (Eigenvalue Functional Compatibility Theorem)**.** *Let $(\widetilde{\lambda},\widetilde{u})$ be an approximate eigenpair of the variational formulation (4.26) and $R(v)$ the corresponding residual, i.e., $R(v) = a(\widetilde{u},v) - \widetilde{\lambda}b(\widetilde{u},v)$. There exist $\delta a \in \mathcal{BL}(\mathcal{V})$ and $\delta b \in \mathcal{BL}(\mathcal{V})$ such that*

$$(a+\delta a)(\widetilde{u},v) = \widetilde{\lambda}(b+\delta b)(\widetilde{u},v), \ \textit{for all } v \in V,$$

with

$$\|\delta a\|_{\mathcal{BL}(\mathcal{V})} \leq \alpha, \quad \|\delta b\|_{\mathcal{BL}(\mathcal{V})} \leq \beta,$$

is equivalent to

$$\|R(v)\|_{V'} \leq \alpha\|\widetilde{u}\|_V + \beta|\widetilde{\lambda}|\|\widetilde{u}\|_V, \tag{4.29}$$

with $\alpha, \beta \in \mathbb{R}^+$.

Proof. $\Rightarrow$ Following the proof of Theorem 4.29 we get

$$\|R(v)\|_{V'} \leq \|\delta a\|_{\mathcal{BL}(V)}\|\widetilde{u}\|_V + |\widetilde{\lambda}|\|\delta b\|_{\mathcal{BL}(V)}\|\widetilde{u}\|_V,$$

which together with assumptions on $\|\delta a\|_{\mathcal{BL}(V)}$, $\|\delta b\|_{\mathcal{BL}(V)}$ yields (4.29).
$\Leftarrow$
We set

$$\delta a(u,v) = -\frac{\alpha}{\alpha\|\widetilde{u}\|_V + \beta\widetilde{\lambda}\|\widetilde{u}\|_V}\|u\|_V R(v)$$

and

$$\delta b(u,v) = \frac{\beta}{\alpha\|\widetilde{u}\|_V + \beta\widetilde{\lambda}\|\widetilde{u}\|_V}\|u\|_V R(v).$$

First, we have

$$\begin{aligned} -\delta a(\widetilde{u},v) + \widetilde{\lambda}\delta b(\widetilde{u},v) &= \frac{\alpha}{\alpha\|\widetilde{u}\|_V + \beta\widetilde{\lambda}\|\widetilde{u}\|_V}\|\widetilde{u}\|_V R(v) \\ &\quad + \widetilde{\lambda}\frac{\beta}{\alpha\|\widetilde{u}\|_V + \beta\widetilde{\lambda}\|\widetilde{u}\|_V}\|\widetilde{u}\|_V R(v) \\ &= \Big(\frac{\alpha\|\widetilde{u}\|}{\alpha\|\widetilde{u}\|_V + \beta\widetilde{\lambda}\|\widetilde{u}\|_V} + \frac{\beta\widetilde{\lambda}\|\widetilde{u}\|}{\alpha\|\widetilde{u}\|_V + \beta\widetilde{\lambda}\|\widetilde{u}\|_V}\Big)R(v), \end{aligned}$$

and therefore

$$-\delta a(\widetilde{u},v) + \widetilde{\lambda}\delta b(\widetilde{u},v) = R(v).$$

Assuming $\|R(v)\|_{V'} \leq \alpha\|\widetilde{u}\|_V + \beta|\widetilde{\lambda}|\|\widetilde{u}\|_V$ and employing the definition of the norm of the bilinear form (2.7) and of the dual norm (2.3) we get

$$\begin{aligned} \|\delta a\|_{\mathcal{BL}(V)} &= \sup_{u,v\in V}\frac{|\delta a(u,v)|}{\|u\|_V\|v\|_V} \leq \frac{\alpha}{\alpha\|\widetilde{u}\|_V + \beta|\widetilde{\lambda}|\|\widetilde{u}\|_V}\sup_{v\in V}\frac{|R(v)|}{\|v\|_V} \\ &= \frac{\alpha}{\alpha\|\widetilde{u}\|_V + \beta|\widetilde{\lambda}|\|\widetilde{u}\|_V}\|R(v)\|_{V'} \\ &\leq \frac{\alpha}{\alpha\|\widetilde{u}\|_V + \beta|\widetilde{\lambda}|\|\widetilde{u}\|_V}\big(\alpha\|\widetilde{u}\|_V + \beta|\widetilde{\lambda}|\|\widetilde{u}\|_V\big) \end{aligned}$$

and

$$\begin{aligned}\|\delta b\|_{\mathcal{BL}(V)} &= \sup_{u,v\in V}\frac{|\delta b(u,v)|}{\|u\|_V\|v\|_V} \le \frac{\beta}{\alpha\|\widetilde{u}\|_V+\beta|\widetilde{\lambda}|\|\widetilde{u}\|_V}\sup_{v\in V}\frac{|R(v)|}{\|v\|_V}\\ &= \frac{\beta}{\alpha\|\widetilde{u}\|_V+\beta|\widetilde{\lambda}|\|\widetilde{u}\|_V}\|R(v)\|_{V'}\\ &\le \frac{\beta}{\alpha\|\widetilde{u}\|_V+\beta|\widetilde{\lambda}|\|\widetilde{u}\|_V}\big(\alpha\|\widetilde{u}\|_V+\beta|\widetilde{\lambda}|\|\widetilde{u}\|_V\big).\end{aligned}$$

Therefore,

$$\|\delta a\|_{\mathcal{BL}(V)}\le\alpha \quad \text{and} \quad \|\delta b\|_{\mathcal{BL}(V)}\le\beta,$$

which completes the proof. □

The normwise functional backward error determines the size of the smallest perturbation such that the approximate eigenpair is an exact solution of the perturbed variational equation. It provides a posteriori information which is dependent on the approximate solution and thus on the choice of the solution method. On the other hand, the condition number allows us to say something about the sensitivity of the eigenpairs to the particular perturbations in the input data. Together, they give us a possibility to estimate the total error, i.e., the forward error, see Definition 2.11, of the problem. In the context of finite element methods the crucial element is to reduce the number of controlled parameters. Since, the discretization process and the iteration process interact, we would like to eliminate their influence whenever it is possible. The condition number can be viewed as an a priori information about the problem which describes a property of the problem and we would like it to be independent of the applied solution method and in particular independent of the discretization parameters [9].

Definition 4.31 (Condition number of the elliptic eigenvalue problem)**.** Let $\delta a\in\mathcal{BL}(V)$, $\delta b\in\mathcal{BL}(V)$ be perturbations of bilinear forms a and b defined as in the variational formulation (4.26), respectively, such that

$$\|\delta a\|_{\mathcal{BL}(V)}\le\varepsilon\alpha, \qquad \|\delta b\|_{\mathcal{BL}(V)}\le\varepsilon\beta, \quad \text{with } \ \varepsilon,\alpha,\beta\in\mathbb{R}^+.$$

The relative eigenvalue and eigenfunction functional condition number, $\mathcal{C}(\mathcal{EVP}_\lambda)$, $\mathcal{C}(\mathcal{EVP}_u)$, for the variational problem (4.26), are the smallest constants C_λ, C_u for which the inequalities

$$|\lambda-\widetilde{\lambda}|\le\varepsilon C_\lambda|\lambda|,$$

$$\|u-\widetilde{u}\|_{V'}\le\varepsilon C_u\|u\|_V,$$

are satisfied.

Moreover, following Definition 2.9

$$\sin_V\angle(\widetilde{u},u)\le\varepsilon C_u.$$

Theorem 4.32 (The eigenvalue functional condition number $\mathcal{C}(\mathcal{EVP}_\lambda)$)**.** *Let $\delta a \in \mathcal{BL}(V)$, $\delta b \in \mathcal{BL}(V)$ be perturbations of bilinear forms a and b defined as in the variational formulation (4.26), respectively, such that*

$$\|\delta a\|_{\mathcal{BL}(V)} \leq \varepsilon\alpha, \qquad \|\delta b\|_{\mathcal{BL}(V)} \leq \varepsilon\beta,$$

with $\varepsilon, \alpha, \beta \in \mathbb{R}^+$.
Then,

$$|\lambda - \widetilde{\lambda}| \leq \varepsilon(|\lambda|\beta + \alpha) + \ h.o.t..$$

Moreover, the eigenvalue functional condition number $\mathcal{C}(\mathcal{EVP}_\lambda)$ satisfies the bound

$$\mathcal{C}(\mathcal{EVP}_\lambda) \leq \frac{|\lambda|\beta + \alpha}{|\lambda|} + h.o.t..$$

Proof. The main idea of the proof is to estimate the eigenvalue error without assuming that the approximate eigenpair $(\widetilde{\lambda}, \widetilde{u})$ fulfills the weak formulation (4.26).
First, we assume that the approximate eigenpair $(\widetilde{\lambda}, \widetilde{u})$ satisfy the equation

$$(a + \delta a)(\widetilde{u}, v) = \widetilde{\lambda}(b + \delta b)(\widetilde{u}, v),$$

which means that it fulfills the modified variational equation

$$\widetilde{a}(\widetilde{u}, v) = \widetilde{\lambda}\widetilde{b}(\widetilde{u}, v).$$

Now, due to the linearity of both bilinear forms we get

$$a(\widetilde{u}, v) + \delta a(\widetilde{u}, v) = \widetilde{\lambda}b(\widetilde{u}, v) + \widetilde{\lambda}\delta b(\widetilde{u}, v).$$

The essential observation is that v can be chosen arbitrarily in V, so let $v = u$ then

$$a(\widetilde{u}, u) + \delta a(\widetilde{u}, u) = \widetilde{\lambda}b(\widetilde{u}, u) + \widetilde{\lambda}\delta b(\widetilde{u}, u).$$

With $a(\widetilde{u}, u) = \lambda b(\widetilde{u}, u)$ and writing $\widetilde{\lambda}$ as $(\lambda - (\lambda - \widetilde{\lambda}))$ we obtain that

$$(\lambda - \widetilde{\lambda})b(\widetilde{u}, u) = (\lambda - (\lambda - \widetilde{\lambda}))\delta b(\widetilde{u}, u) - \delta a(\widetilde{u}, u)$$

and

$$|\lambda - \widetilde{\lambda}||b(\widetilde{u}, u)| = |(\lambda - (\lambda - \widetilde{\lambda}))\delta b(\widetilde{u}, u) - \delta a(\widetilde{u}, u)|.$$

After applying the triangle inequality we get

$$|\lambda - \widetilde{\lambda}||b(\widetilde{u}, u)| \leq |\lambda|\|\delta b\|_{\mathcal{BL}(V)}\|\widetilde{u}\|_V\|u\|_V + |\lambda - \widetilde{\lambda}|\|\delta b\|_{\mathcal{BL}(V)}\|\widetilde{u}\|_V\|u\|_V + \|\delta a\|_{\mathcal{BL}(V)}\|\widetilde{u}\|_V\|u\|_V.$$

As, $\|\delta a\|_{\mathcal{BL}(V)} \leq \varepsilon\alpha$ and $\|\delta b\|_{\mathcal{BL}(V)} \leq \varepsilon\beta$, we have

$$|\lambda - \widetilde{\lambda}||b(\widetilde{u}, u)| \leq \varepsilon\beta|\lambda|\|\widetilde{u}\|_V\|u\|_V + \varepsilon\beta|\lambda - \widetilde{\lambda}|\|\widetilde{u}\|_V\|u\|_V + \varepsilon\alpha\|\widetilde{u}\|_V\|u\|_V.$$

Since $|b(\widetilde{u},u)| \neq 0$, the last inequality yields

$$|\lambda - \widetilde{\lambda}| \leq (\varepsilon\beta|\lambda| + \varepsilon\beta|\lambda - \widetilde{\lambda}| + \varepsilon\alpha)\frac{\|\widetilde{u}\|_V \|u|_V}{|b(\widetilde{u},u)|}.$$

Following Definition 2.9 of the angle between functions, it is straightforward that

$$|\lambda - \widetilde{\lambda}| \leq (\varepsilon\beta|\lambda| + \varepsilon\beta|\lambda - \widetilde{\lambda}| + \varepsilon\alpha)\frac{1}{\cos\angle(\widetilde{u},u)}.$$

We are now in the position to apply Wilkinson's first order perturbation result [113], namely

$$|\lambda - \widetilde{\lambda}| \leq (|\lambda|\varepsilon\beta + |\lambda - \widetilde{\lambda}|\varepsilon\beta + \varepsilon\alpha) + \mathcal{O}(\angle(\widetilde{u},u)^2).$$

If $\varepsilon\beta < 1$ then

$$|\lambda - \widetilde{\lambda}| \leq \varepsilon(1-\varepsilon\beta)^{-1}(|\lambda|\beta + \alpha) + \mathcal{O}(\angle(\widetilde{u},u)^2).$$

After neglecting higher order terms in the Taylor expansion of $(1-\varepsilon\beta)^{-1}$, we finally get

$$|\lambda - \widetilde{\lambda}| \leq \varepsilon(|\lambda|\beta + \alpha) + \mathcal{O}((\varepsilon\beta)^2) + \mathcal{O}(\angle(\widetilde{u},u)^2). \quad \square$$

Remark 4.33. We note that if $\delta b = 0$ then

$$|\lambda - \widetilde{\lambda}| \leq \frac{\|\delta a\|_{\mathcal{BL}(V)}\|\widetilde{u}\|_V\|u\|_V}{|b(\widetilde{u},u)|} = \frac{\|\delta a\|_{\mathcal{BL}(V)}}{\cos\angle(\widetilde{u},u)}$$

and the relative eigenvalue error is given by

$$\begin{aligned}\frac{|\lambda - \widetilde{\lambda}|}{|\lambda|} &\leq \frac{\|\delta a\|_{\mathcal{BL}(V)}}{|\lambda|\cos\angle(\widetilde{u},u)} \leq \frac{\|\delta a\|_{\mathcal{BL}(V)}}{|\lambda|}\sec\angle(\widetilde{u},u) = \frac{\|\delta a\|_{\mathcal{BL}(V)}}{|\lambda|} + \mathcal{O}(\angle(\widetilde{u},u)^2) \\ &\leq \underbrace{\frac{\|a\|_{\mathcal{BL}(V)}}{|\lambda|}}_{\substack{\text{relative eigenvalue}\\ \text{condition number}}} \underbrace{\frac{\|\delta a\|_{\mathcal{BL}(V)}}{\|a\|_{\mathcal{BL}(V)}}}_{\text{backward error}} + \mathcal{O}(\angle(\widetilde{u},u)^2).\end{aligned}$$

The essential observation is that the last inequality is a direct infinite dimensional analogue of the well-known result [20, Section 3.4] or [35].

Theorem 4.34 (The eigenvector functional condition number $\mathcal{C}(\mathcal{EVP}_u)$)**.** *Let (λ,u) be the exact eigenpair of the variational formulation (4.26) and let $(\widetilde{\lambda},\widetilde{u})$ be its approximation. Let bilinear forms a, $b \in \mathcal{BL}(V)$ satisfy (4.27) and let $\delta a, \delta b \in \mathcal{BL}(V)$ be perturbations of a, b, respectively, such that*

$$\|\delta a\|_{\mathcal{BL}(V)} \leq \varepsilon\alpha, \qquad \|\delta b\|_{\mathcal{BL}(V)} \leq \varepsilon\beta,$$

with $\varepsilon, \alpha, \delta \in \mathbb{R}^+$.
Let C_2, C_3 be the continuity and the coercivity constant as in (4.27) and $C = (|\lambda|\beta + \alpha)(C_2 + \varepsilon\beta + 1)$. If $C_3 > \varepsilon C + C_2|\lambda|$ then

$$\|u-\widetilde{u}\|_V \le \varepsilon\Big(1-\frac{\varepsilon C+|\lambda|C_2}{C_3}\Big)^{-1}\frac{1}{C_3}C\|u\|_V.$$

The eigenfunction functional condition number $\mathcal{C}(\mathcal{EVP}_u)$ *satisfies the bound*

$$\mathcal{C}(\mathcal{EVP}_u) \le \frac{C}{C_3} = \frac{(|\lambda|\beta+\alpha)(C_2+\varepsilon\beta+1)}{C_3}.$$

Proof. Let $(\widetilde{\lambda},\widetilde{u})$ be the approximate solution of the variational equation (4.26). The inexact eigenpair satisfies a perturbed variational equation

$$(a+\delta a)(\widetilde{u},v) = \widetilde{\lambda}(b+\delta b)(\widetilde{u},v),$$

which can be written as

$$a(\widetilde{u},v)+\delta a(\widetilde{u},v) = \widetilde{\lambda}b(\widetilde{u},v)+\widetilde{\lambda}\delta b(\widetilde{u},v).$$

Setting $\widetilde{u} = u-(u-\widetilde{u})$ implies that

$$a(u-(u-\widetilde{u}),v)+\delta a(u-(u-\widetilde{u}),v) = \widetilde{\lambda}b(u-(u-\widetilde{u}),v)+\widetilde{\lambda}\delta b(u-(u-\widetilde{u}),v).$$

Then

$$a(u,v)-a(u-\widetilde{u},v)+\delta a(u,v)-\delta a(u-\widetilde{u},v) = \widetilde{\lambda}b(u,v)-\widetilde{\lambda}b(u-\widetilde{u},v)+\widetilde{\lambda}\delta b(u,v)-\widetilde{\lambda}\delta b(u-\widetilde{u},v).$$

Since $a(u,v)=\lambda b(u,v)$, we have

$$\lambda b(u,v)-a(u-\widetilde{u},v)+\delta a(u,v)-\delta a(u-\widetilde{u},v) = \widetilde{\lambda}b(u,v)-\widetilde{\lambda}b(u-\widetilde{u},v)+\widetilde{\lambda}\delta b(u,v)-\widetilde{\lambda}\delta b(u-\widetilde{u},v).$$

With the choice $v=u-\widetilde{u}$, we get

$$\begin{aligned}-a(u-\widetilde{u},u-\widetilde{u}) &= \widetilde{\lambda}b(u,u-\widetilde{u})-\lambda b(u,u-\widetilde{u})-\widetilde{\lambda}b(u-\widetilde{u},u-\widetilde{u})+\widetilde{\lambda}\delta b(u,u-\widetilde{u})\\ &\quad -\widetilde{\lambda}\delta b(u-\widetilde{u},u-\widetilde{u})-\delta a(u,u-\widetilde{u})+\delta a(u-\widetilde{u},u-\widetilde{u}).\end{aligned}$$

With the coercivity condition (4.27), we can show that

$$\begin{aligned}C_3\|u-\widetilde{u}\|_V^2 &\le |\widetilde{\lambda}-\lambda|\|b\|_{\mathcal{BL}(V)}\|u\|_V\|u-\widetilde{u}\|_V+|\widetilde{\lambda}|_V\|b\|_{\mathcal{BL}(V)}\|u-\widetilde{u}\|_V^2\\ &\quad +|\widetilde{\lambda}|\|\delta b\|_{\mathcal{BL}(V)}\|u\|_V\|u-\widetilde{u}\|_V+|\widetilde{\lambda}|\|\delta b\|_{\mathcal{BL}(V)}\|u-\widetilde{u}\|_V^2\\ &\quad +\|\delta a\|_{\mathcal{BL}(V)}\|u\|_V\|u-\widetilde{u}\|_V+\|\delta a\|_{\mathcal{BL}(V)}\|u-\widetilde{u}\|_V^2.\end{aligned}$$

The boundness of the bilinear form b (4.27) and the assumptions on the size of perturbations $\|\delta a\|_{\mathcal{BL}(V)}$, $\|\delta b\|_{\mathcal{BL}(V)}$, yield

$$
\begin{aligned}
C_3\|u-\widetilde{u}\|_V^2 &\leq |\widetilde{\lambda}-\lambda|C_2\|u\|_V\|u-\widetilde{u}\|_V + |\widetilde{\lambda}|C_2\|u-\widetilde{u}\|_V^2 \\
&\quad + |\widetilde{\lambda}|\varepsilon\beta\|u\|_V\|u-\widetilde{u}\|_V + |\widetilde{\lambda}|\varepsilon\beta\|u-\widetilde{u}\|_V^2 \\
&\quad + \varepsilon\alpha\|u\|_V\|u-\widetilde{u}\|_V + \varepsilon\alpha\|u-\widetilde{u}\|_V^2.
\end{aligned}
$$

Since $\|u-\widetilde{u}\|_V \neq 0$, we get

$$
\begin{aligned}
C_3\|u-\widetilde{u}\|_V &\leq |\widetilde{\lambda}-\lambda|C_2\|u\|_V + |\widetilde{\lambda}|C_2\|u-\widetilde{u}\|_V \\
&\quad + |\widetilde{\lambda}|\varepsilon\beta\|u\|_V + |\widetilde{\lambda}|\varepsilon\beta\|u-\widetilde{u}\|_V + \varepsilon\alpha\|u\|_V + \varepsilon\alpha\|u-\widetilde{u}\|_V.
\end{aligned}
$$

By setting $\widetilde{\lambda} = \lambda - (\lambda - \widetilde{\lambda})$, we obtain

$$
\begin{aligned}
C_3\|u-\widetilde{u}\|_V &\leq |\widetilde{\lambda}-\lambda|C_2\|u\|_V + |\lambda-(\lambda-\widetilde{\lambda})|C_2\|u-\widetilde{u}\|_V \\
&\quad + |\lambda-(\lambda-\widetilde{\lambda})|\varepsilon\beta\|u\|_V + |\lambda-(\lambda-\widetilde{\lambda})|\varepsilon\beta\|u-\widetilde{u}\|_V \\
&\quad + \varepsilon\alpha\|u\|_V + \varepsilon\alpha\|u-\widetilde{u}\|_V,
\end{aligned}
$$

and

$$
\begin{aligned}
C_3\|u-\widetilde{u}\|_V &\leq |\widetilde{\lambda}-\lambda|(C_2\|u\|_V + C_2\|u-\widetilde{u}\|_V + \varepsilon\beta\|u\| + \varepsilon\beta\|u-\widetilde{u}\|_V) \\
&\quad + |\lambda|\|C_2\|u-\widetilde{u}\|_V + |\lambda|\varepsilon\beta\|u\|_V + |\lambda|\varepsilon\beta\|u-\widetilde{u}\|_V \\
&\quad + \varepsilon\alpha\|u\| + \varepsilon\alpha\|u-\widetilde{u}\|_V.
\end{aligned}
$$

Using the estimate of the eigenvalue error from Theorem 4.32, we arrive at

$$
\begin{aligned}
C_3\|u-\widetilde{u}\|_V &\leq \Big(\varepsilon(|\lambda|\beta+\alpha)\Big)\Big(C_2\|u\|_V + C_2\|u-\widetilde{u}\|_V + \varepsilon\beta\|u\|_V + \varepsilon\beta\|u-\widetilde{u}\|_V\Big) \\
&\quad + |\lambda|\|C_2\|u-\widetilde{u}\|_V + |\lambda|\varepsilon\beta\|u\|_V + |\lambda|\varepsilon\beta\|u-\widetilde{u}\|_V \\
&\quad + \varepsilon\alpha\|u\|_V + \varepsilon\alpha\|u-\widetilde{u}\|_V,
\end{aligned}
$$

which implies that

$$
\begin{aligned}
C_3\|u-\widetilde{u}\|_V &\leq \varepsilon\Big(|\lambda|\beta+\alpha\Big)\Big(C_2+\varepsilon\beta+1\Big)\|u\|_V \\
&+ \varepsilon\Big(|\lambda|\beta+\alpha\Big)\Big(C_2+\varepsilon\beta+1\Big)\|u-\widetilde{u}\|_V + |\lambda|\|C_2\|u-\widetilde{u}\|_V. \qquad (4.30)
\end{aligned}
$$

Let now $C = \Big(|\lambda|\beta+\alpha\Big)\Big(C_2+\varepsilon\beta+1\Big)$, then (4.30) reads

$$
C_3\|u-\widetilde{u}\|_V \leq \varepsilon C\|u\|_V + \varepsilon C\|u-\widetilde{u}\|_V + |\lambda|\|C_2\|u-\widetilde{u}\|_V
$$

and the inequality

$$\|u - \widetilde{u}\|_V (C_3 - \varepsilon C - |\lambda| C_2) \leq \varepsilon C_3 \|u\|_V,$$

holds.
If $(C_3 > \varepsilon C + |\lambda| C_2)$, then we have that

$$\|u - \widetilde{u}\|_V \leq \frac{1}{(C_3 - \varepsilon C - |\lambda| C_2)} \varepsilon C \|u\|_V$$

and readily

$$\|u - \widetilde{u}\|_V \leq \varepsilon \Big(1 - \frac{\varepsilon C + |\lambda| C_2}{C_3}\Big)^{-1} \frac{C}{C_3} \|u\|_V.$$

Therefore, the eigenfunction functional condition number $\mathcal{C}(\mathcal{EVP}_u)$ satisfies

$$\mathcal{C}(\mathcal{EVP}_u) \leq \frac{C}{C_3} = \frac{(|\lambda|\beta + \alpha)(C_2 + \varepsilon\beta + 1)}{C_3}.$$

□

4.4 A combined a posteriori error estimator for self-adjoint eigenvalue problems

Let us consider the Laplace eigenvalue problem (3.1) in its variational form.
Determine a non-trivial eigenpair $(\lambda, u) \in \mathbb{R} \times H_0^1(\Omega)$*, with* $b(u,u) = 1$ *such that*

$$a(u,v) = \lambda b(u,v) \quad \textit{for all} \;\; v \in V = H_0^1(\Omega),$$

with λ *being a simple eigenvalue.*
Let $(\lambda_h, u_h) \in \mathbb{R} \times V_h$ be the exact discrete eigenpair (the Galerkin solution), such that

$$a(u_h, v_h) = \lambda_h b(u_h, v_h) \quad \text{for all} \quad v_h \in V_h \subset V.$$

For the case of the Galerkin solution $(\lambda_h, u_h) \in \mathbb{R} \times V_h$ the equivalence between the discretization error $\|u - u_h\|_{H_0^1(\Omega)}$ and the dual norm of the residual, i.e., $\|R(v)\|_{H^{-1}(\Omega)} = \|a(u_h, v) - \lambda_h b(u_h, v)\|_{H^{-1}(\Omega)}$, up to the higher-order terms, is a well-known relation [31, 50], which allows to derive the standard residual error estimator $\eta(\lambda_h, u_h)$, i.e.,

$$\eta(\lambda_h, u_h) \lesssim \|u - u_h\|_{H_0^1(\Omega)} \lesssim \eta(\lambda_h, u_h),$$

with the norm equivalence

$$c\|u - u_h\|_{H^1(\Omega)} \leq \|u - u_h\|_{H_0^1(\Omega)} \leq C\|u - u_h\|_{H^1(\Omega)}.$$

Note that $\|\cdot\|_{H_0^1(\Omega)} = |||\cdot|||$.

However, if we consider the inexact finite element solution $(\widetilde{\lambda}_h, \widetilde{u}_h) \in \mathbb{R} \times V_h$, where the representation vector $\widetilde{\mathbf{u}}_h$ is calculated as an approximate solution of the generalized eigenvalue problem

$$A_h \mathbf{u}_h = \lambda_h B_h \mathbf{u}_h,$$

with the matrices, A_h and B_h being symmetric and symmetric positive definite, respectively, the corresponding equivalence relation between the complete error $\|u - \widetilde{u}_h\|_{H_0^1(\Omega)}$ and the dual norm of the residual, i.e., $\|R(v)\|_{H^{-1}(\Omega)} = \|a(\widetilde{u}_h, v) - \widetilde{\lambda}_h b(\widetilde{u}_h, v)\|_{H^{-1}(\Omega)}$, is not proved. Nevertheless, it is still possible to control the adaptive finite element method with the standard residual error estimator calculated with a non-Galerkin solution $(\widetilde{\lambda}_h, \widetilde{u}_h)$ providing that the influence of the approximation error $\|u_h - \widetilde{u}_h\|_{H_0^1(\Omega)}$ will be analyzed and combined into error bounds. In [64] the combined a posteriori error estimator for a non-Galerkin solution of the boundary value problem was designed where the $H^{-1}(\Omega)$-norm of the residual is split into the discrete part, the BPX preconditioner [25] of the algebraic residual and the continuous part, i.e., the standard residual error estimator $\eta(\widetilde{\lambda}_h, \widetilde{u}_h)$. The first part of the estimator controls the iteration error and reflects the influence of the quality of the approximate solution on the estimator, where the latter term measures the size of the discretization error.

4.4.1 A combined residual error estimator

Let us consider the following bound of the global eigenfunction error.

$$\|u - \widetilde{u}_h\|_{H_0^1(\Omega)} \leq \|u - u_h\|_{H_0^1(\Omega)} + \|u_h - \widetilde{u}_h\|_{H_0^1(\Omega)},$$

where $\|u - u_h\|_{H_0^1(\Omega)}$ is the discretization error and $\|u_h - \widetilde{u}_h\|_{H_0^1(\Omega)}$ the algebraic error. We now exploit the idea of the combined a posteriori error estimator introduced in [64] and bound each term independently. In order to control the discretization error we can use the standard residual error estimator $\eta(\lambda_h, u_h)$, i.e.,

$$\eta(\lambda_h, u_h) \lesssim \|u - u_h\|_{H_0^1(\Omega)} \lesssim \eta(\lambda_h, u_h).$$

Let $\mathbf{u}_h$, $\widetilde{\mathbf{u}}_h$ be the representation vector for u_h and $\widetilde{u}_h$, respectively, and $\|\cdot\|_{\mathbf{H}}$ be a discrete H_0^1-norm. Then, from Theorem 2.50

$$\|u_h - \widetilde{u}_h\|_{H_0^1(\Omega)} = \|\mathbf{u}_h - \widetilde{\mathbf{u}}_h\|_{\mathbf{H}},$$

and the following upper bound holds

$$\|u - \widetilde{u}_h\|_{H^1(\Omega)} \leq \|u - u_h\|_{H^1(\Omega)} + \|u_h - \widetilde{u}_h\|_{H^1(\Omega)} \lesssim \eta(\lambda_h, u_h) + \|\mathbf{u}_h - \widetilde{\mathbf{u}}_h\|_{\mathbf{H}}. \tag{4.31}$$

The correspondence between this upper bound and the upper bound derived in [64] is straightforward. The discretization error is controlled by the standard residual a posteriori error estimator, while the approximation error is controlled by the difference between underlying representation vectors measured in the discrete $H_0^1(\Omega)$-norm. In contrast to [64], where all the bounds are derived directly with the inexact eigenpair $(\widetilde{\lambda}_h, \widetilde{u}_h)$, here we first use

the residual a posteriori error estimator derived with the Galerkin solution u_h, i.e., $\eta(\lambda_h, u_h)$ and afterwards we apply the perturbation argument where we assume that the difference between $\eta(\lambda_h, u_h)$ and $\eta(\widetilde{\lambda}_h, \widetilde{u}_h)$ can be measured by $\|\mathbf{u}_h - \widetilde{\mathbf{u}}_h\|_{\mathbf{H}}$, which makes this result slightly weaker, i.e., the lack of the lower bound prevents to prove the efficiency of the estimator.
The major questions are now: How are the discrete $H_0^1(\Omega)$-norm and its dual norm defined, How can we estimate the size of the approximation error in the discrete $H_0^1(\Omega)$-norm and Where is the dual norm $H^{-1}(\Omega)$ of the discrete residual taken into account.

In [65] a shift-invert Lanczos method with a specially defined inner product was introduced. The authors showed that applying the $\mathbf{H} := (A+B)$-inner product in the shift-invert Lanczos method for the matrix pencil (A, B) enables to measure the size of the residual vector in the discrete $H^{-1}(\Omega)$-norm and that the $\mathbf{H}$-norm itself can be identified as a discrete $H_0^1(\Omega)$-norm. Let us recall the crucial perturbation result stated in [65].

Proposition 4.35. *[65, Proposition 3.2]*
Let (A, B) be a symmetric definite pencil, $\mathbf{H} = A + B$ a symmetric positive definite matrix, and σ a real number such that $A_\sigma := A - \sigma B$ is invertible. Let $\widetilde{x}$ be a nonzero vector in $\mathbb{R}^n$, $\widetilde{\lambda}$ a real number such that $\widetilde{\lambda} \neq \sigma$ and the residual vector

$$r = A\widetilde{x} - \widetilde{\lambda} B\widetilde{x}.$$

If

$$\Big|\frac{1}{\lambda - \sigma} - \frac{1}{\widetilde{\lambda} - \sigma}\Big| = \min_{\lambda_i} \Big|\frac{1}{\lambda_i - \sigma} - \frac{1}{\widetilde{\lambda} - \sigma}\Big|, \tag{4.32}$$

and $Ax = \lambda Bx$, where $\|x\|_{\mathbf{H}} = 1$, then

$$\Big|\frac{\lambda - \widetilde{\lambda}}{\lambda - \sigma}\Big| \leq \frac{\|r\|_{A_\sigma^{-1}\mathbf{H}A_\sigma^{-1}}}{\|\widetilde{x}\|_{\mathbf{H}}}, \tag{4.33}$$

$$0 \leq |\sin_{\mathbf{H}} \angle(\widetilde{x}, x)| \leq \Big|\frac{\lambda_\gamma - \sigma}{\lambda_\gamma - \widetilde{\lambda}}\Big| \frac{\|r\|_{A_\sigma^{-1}\mathbf{H}A_\sigma^{-1}}}{\|\widetilde{x}\|_{\mathbf{H}}}, \tag{4.34}$$

where

$$\Big|\frac{1}{\lambda_\gamma - \sigma} - \frac{1}{\widetilde{\lambda} - \sigma}\Big| = \min_{\lambda_i \neq \lambda} \Big|\frac{1}{\lambda_i - \sigma} - \frac{1}{\widetilde{\lambda} - \sigma}\Big|. \tag{4.35}$$

Proof. See [65]. □
Following Proposition 4.35, the $\mathbf{H}$-norm can be identified as a discrete $H_0^1(\Omega)$-norm, whereas the $A_\sigma^{-1}\mathbf{H}A_\sigma^{-1}$ as a discrete $H^{-1}(\Omega)$-norm.
Our next task is to approximate the eigenvector error in a discrete $H_0^1(\Omega)$-norm. With the appropriate normalization, $\|u_h\|_{H_0^1(\Omega)} = 1$, the difference between the representation vectors is given by

$$\|\mathbf{u}_h - \widetilde{\mathbf{u}}_h\|_{\mathbf{H}} = \sin_{\mathbf{H}} \angle(\mathbf{u}_h, \widetilde{\mathbf{u}}_h).$$

Now, let us apply Proposition 4.35 to the pencil (A_h, B_h). With $\mathbf{H} = A_h + B_h$ we have

$$\sin_{\mathbf{H}} \angle(\mathbf{u}_h, \widetilde{\mathbf{u}}_h) \lesssim \frac{\|\mathbf{r}\|_{A_\sigma^{-1}\mathbf{H}A_\sigma^{-1}}}{\|\widetilde{\mathbf{u}}_h\|_{\mathbf{H}}}.$$

which together with (4.31) gives a final upper bound

$$\|u - \widetilde{u}_h\|_{H^1(\Omega)} \lesssim \eta(\lambda_h, u_h) + \frac{\|\mathbf{r}\|_{A_\sigma^{-1}\mathbf{H}A_\sigma^{-1}}}{\|\widetilde{\mathbf{u}}_h\|_{\mathbf{H}}} := \eta_{new}(\lambda_h, u_h). \tag{4.36}$$

This theoretical upper bound has to be slightly modified to be used in practice. At first we are not able to compute $\eta(\lambda_h, u_h)$, however, since we know that the second term measures the approximation error, we can use $\eta(\widetilde{\lambda}_h, \widetilde{u}_h)$ instead. On the other hand, we want to get the second term as a by product of the shift-invert Lanczos method. Therefore, performing m iterations of the shift-inverse Lanczos algorithm with the pencil (A_h, B_h) and $\mathbf{H} = A_h + B_h$ inner product, gives an approximate eigenpair $(\widetilde{\lambda}_h, \widetilde{\mathbf{u}}_h)$ such that

$$\frac{\|\mathbf{r}\|_{A_\sigma^{-1}\mathbf{H}A_\sigma^{-1}}}{\|\widetilde{\mathbf{u}}_h\|_{\mathbf{H}}} = \frac{\beta_m |e_m^T s|}{\theta},$$

where (θ, s) is an approximate eigenpair of the tridiagonal Lanczos matrix H_m from the Arnoldi/Lanczos factorization (2.19), see Section 2.3.1, $\widetilde{\lambda}_h = \frac{1}{\theta}$, $\widetilde{\mathbf{u}}_h = V_m s$ and β_m, e_m^T, V_m are defined as in (2.20). For the complete analysis we refer to [65].
Consequently, after m iterations of the shift-invert Lanczos method we have the following global upper bound for the eigenfunction error

$$\|u - \widetilde{u}_h\|_{H^1(\Omega)} \lesssim \eta(\widetilde{\lambda}_h, \widetilde{u}_h) + \frac{\beta_m |e_m^T s|}{\theta} := \eta_{new}(\widetilde{\lambda}_h, \widetilde{u}_h).$$

4.4.2 The balanced AFEM algorithm

With an appropriate norm and the new combined a posteriori error estimator, we can try to reduce the number of Arnoldi/Lanczos iterations performed during the AFEM algorithm for solving the self-adjoint eigenvalue problem, e.g., the Laplace eigenvalue problem. We consider stopping criteria for the iterative eigensolver and the adaptive mesh refinement based on the new estimator $\eta_{new}(\widetilde{\lambda}_h, \widetilde{u}_h)$. As we have mentioned before, the upper bound (4.36) is a global upper bound. The only part of the new estimator which can be used to control the error locally, is a standard residual type error estimator, i.e., $\eta(\widetilde{\lambda}_h, \widetilde{u}_h)$ or precisely elementwise refinement indicators $\eta_T(\widetilde{\lambda}_h, \widetilde{u}_h)$ extracted from it. Although, the discrete $H^{-1}(\Omega)$-norm of the algebraic residual contains the global information about the discrete problem, we can still define an appropriate stopping criterion and perform the adaptive mesh refinement.

Let us for a moment consider the ideal situation when after each Arnoldi/Lanczos iteration step we check if the $H^{-1}(\Omega)$-norm of the algebraic residual, i.e., $\frac{\beta_m |e_m^T s|}{\theta}$, is smaller than a certain fraction ω of the corresponding continuous part of the error, i.e., $\eta(\widetilde{\lambda}_h, \widetilde{u}_h)$. Obviously, we can stop the iteration if this condition is satisfied or continue in the other case.

This balancing strategy assures that the iteration stops when the approximation error is of order of the discretization error. Therefore, the set of marked elements, determined by selecting only those for which local indicators $\eta_T(\widetilde{\lambda}_h, \widetilde{u}_h)$ are satisfying a certain marking criterion, will guarantee a certain accuracy of the solution. Of course calculating all indicators $\eta_T(\widetilde{\lambda}_h, \widetilde{u}_h)$ at every iteration of the eigensolver is not optimal. Therefore, the number m of the Arnoldi/Lanczos iterations performed before calculating $\eta_T(\widetilde{\lambda}_h, \widetilde{u}_h)$ and a proper parameter ω for equilibration have to be analyzed. In order to have an efficient estimator it is necessary to obtain also a lower bound, i.e.,

$$\eta_{new}(\widetilde{\lambda}_h, \widetilde{u}_h) \lesssim \|u - \widetilde{u}_h\|_{H^1(\Omega)}.$$

Unfortunately, the proof for the efficiency of this new estimator is not complete and is still under investigations. Instead, we will analyze the behavior of the estimator numerically.

The pseudo code of the balanced AFEM algorithm described in this section is presented below. In [65] the shift-invert Lanczos method is discussed, however, since we are interested is the smallest eigenvalue, we analyze the simplest version of the algorithm, i.e., with shift $\sigma = 0$. Of course calculating the error estimator $\eta(\widetilde{\lambda}_h, \widetilde{u}_h)$ at each iteration step is expensive and not possible in practice, nevertheless, in order to formulate some thesis about the minimal number of required iterations we will make this effort here.

The balanced AFEM algorithm for the smallest eigenvalue of the Laplace eigenvalue problem (3.1)

Input: An initial regular triangulation $\mathcal{T}_h^i$, a balancing parameter ω, a maximal number of degrees of freedom `maxDOF`

Output: Approximation $\widetilde{\lambda}_h$ to the smallest eigenvalue λ of (3.1) together with the corresponding approximate eigenfunction $\widetilde{u}_h$.

Solve: Discretize problem (3.1) on $\mathcal{T}_h^i$ and obtain the matrix pencil (A_h, B_h)

Select shift $\sigma = 0$

while $\frac{\|\mathbf{r}\|_{A_h^{-1}\mathbf{H}A_h^{-1}}}{\|\widetilde{\mathbf{u}}_h\|_{\mathbf{H}}} \geq \omega\eta(\widetilde{\lambda}_h, \widetilde{u}_h)$ **do**

Perform one iteration of the **H**-Arnoldi/Lanczos method with (A_h, B_h)

Compute the approximate eigenpair $(\widetilde{\lambda}_h, \widetilde{u}_h)$, the continuous and the discrete error estimators $\eta(\widetilde{\lambda}_h, \widetilde{u}_h)$ and $\frac{\|\mathbf{r}\|_{A_h^{-1}\mathbf{H}A_h^{-1}}}{\|\widetilde{\mathbf{u}}_h\|_{\mathbf{H}}}$

end while

Estimate: Calculate the combined error estimator $\eta_{new}(\widetilde{\lambda}_h, \widetilde{u}_h)$ and the standard error estimator $\eta(\widetilde{\lambda}_h, \widetilde{u}_h)$

Mark: Mark the elements based on $\eta(\widetilde{\lambda}_h, \widetilde{u}_h)$ using the bulk criterion

Refine: Refine the coarse mesh $\mathcal{T}_h^i$ using the green, blue or red refinement to get $\mathcal{T}_h^{i+1}$

if #DOF $<$ `maxDOF` **then**

return $(\widetilde{\lambda}_H, \widetilde{\mathbf{u}}_H)$

else

Start the algorithm with $\mathcal{T}_h^{i+1}$

end if

4.4.3 Numerical experiments

Throughout this section we investigate several aspects of our new balanced AFEM algorithm. As a model example, we consider the Laplace eigenvalue problem 3.1 on the L-shape domain, i.e., $\Omega = [-1, 1] \times [0, 1] \cup [-1, 0] \times [-1, 0]$. We dedicate all numerical experiments to approximating only the smallest eigenvalue which we compare with a reference value obtained in [108], i.e.,

$$\lambda_1 \approx 9.639723844.$$

At first we analyze the cost and the accuracy dependence of our new algorithm on the choice of the balancing parameter ω. Next we give some empirical information about the minimal number of Arnoldi/Lanczos iterations required at each step of the balanced AFEM algorithm to determine an accurate final approximation. Since our combined error estimator $\eta_{new}(\widetilde{\lambda}_h, \widetilde{u}_h)$, contains a residual type a posteriori error estimator $\eta(\widetilde{\lambda}_h, \widetilde{u}_h)$, the last part of this section is dedicated to compare the behavior of the combined error estimator $\eta_{new}(\widetilde{\lambda}_h, \widetilde{u}_h)$ where $\eta(\widetilde{\lambda}_h, \widetilde{u}_h)$ is chosen to be a standard residual type estimator $\eta_{DPR}(\widetilde{\lambda}_h, \widetilde{u}_h)$ [50] or an edge residual error estimator $\eta_{CG}(\widetilde{\lambda}_h, \widetilde{u}_h)$ [31], see Section 2.2.3 for details.

All the experiments were realized with the OPENFFW [29] finite element framework, which contains implementations of both η_{DPR} and η_{CG} error estimators. For our new balanced AFEM algorithm we use our own implementation of the Arnoldi/Lanczos method with the **H**-inner product, while standard MATLAB *eigs* function is used for the comparison.

Balancing with different values of parameter ω Let us first examine the behavior of our new balanced AFEM algorithm with respect to the choice of the parameter ω. We consider here the ideal situation when after each Lanczos iteration step we can check whether the $H^{-1}(\Omega)$-norm of the algebraic residual, i.e., $\frac{\beta_m |e_m^T s|}{\theta}$, is smaller than a certain fraction ω of the corresponding $\eta(\widetilde{\lambda}_h, \widetilde{u}_h)$. Here we restrict ourselves to $\eta(\widetilde{\lambda}_h, \widetilde{u}_h) = \eta_{DPR}(\widetilde{\lambda}_h, \widetilde{u}_h)$ and we assume that the minimal number of required Arnoldi/Lanczos iterations is set to $2k+1$, where k is the number of eigenvalues of interest, here $k = 1$.
Tables 4.10 – 4.12 present the numerical results for $\omega = 0.5, 0.1, 0.9$, respectively. In addition to the eigenvalue approximation and the number of degrees of freedom, information about the size of the continuous $\eta(\widetilde{\lambda}_h, \widetilde{u}_h)$ and the discrete $\frac{\|\mathbf{r}\|_{A_\sigma^{-1}\mathbf{H}A_\sigma^{-1}}}{\|\widetilde{\mathbf{u}}_h\|_{\mathbf{H}}}$ part of the estimator is given. We notice that for $\omega = 0.5$ three Arnoldi/Lanczos iterations are enough to obtain a good error estimator to steer the adaptive mesh refinement.
Reducing ω to $\omega = 0.1$ forces the discrete residual to be smaller which leads to slightly more Arnoldi/Lanczos iterations, however, the final accuracy is reached with smaller amount of degrees of freedom. Obviously choosing $\omega = 0.9$ leads to meshes with more degrees of freedom but reduces the number of Arnoldi/Lanczos iterations performed on each step of the adaptive algorithm. The corresponding final meshes and convergence history plots are depicted in Figures 4.31 – 4.33.
These few examples show that determining the approximate solution of the same accuracy with a different balancing parameter is possible. Particularly chosen ω may lead to more Arnoldi/Lanczos iterations or more degrees of freedom. Of course, performing one more Arnoldi/Lanczos iteration on the coarse mesh is cheaper than dealing with finer problems, therefore, this decision has to be made by the user, depending on existing limitations.

Table 4.10: Approximations of the smallest eigenvalue for (3.1) on the L-shape domain with $\omega = 0.5$.

ref. level	#DOF	# iteration	$\widetilde{\lambda}_1$	$\lvert\lambda_1 - \widetilde{\lambda}_1\rvert$	$\eta(\widetilde{\lambda}_h, \widetilde{u}_h)$	$\frac{\|\mathbf{r}\|_{A_\sigma^{-1}\mathbf{H}A_\sigma^{-1}}}{\|\widetilde{\mathbf{u}}_h\|_{\mathbf{H}}}$	$\eta_{new}(\widetilde{\lambda}_h, \widetilde{u}_h)$
1	33	3	10.6008	0.9610	2.2454	0.0401	2.2853
2	78	3	10.2025	0.5628	1.0976	0.0349	1.1325
3	166	3	9.8854	0.2457	0.5476	0.0430	0.5906
4	341	3	9.7941	0.1544	0.2890	0.0373	0.3262
5	622	3	9.7234	0.0837	0.1540	0.0384	0.1924
6	1203	4	9.6805	0.0407	0.0866	0.0085	0.0950
7	2096	4	9.6606	0.0208	0.0472	0.0091	0.0562
8	3946	4	9.6518	0.0121	0.0273	0.0088	0.0361
9	6737	5	9.6464	0.0066	0.0154	0.0012	0.0167

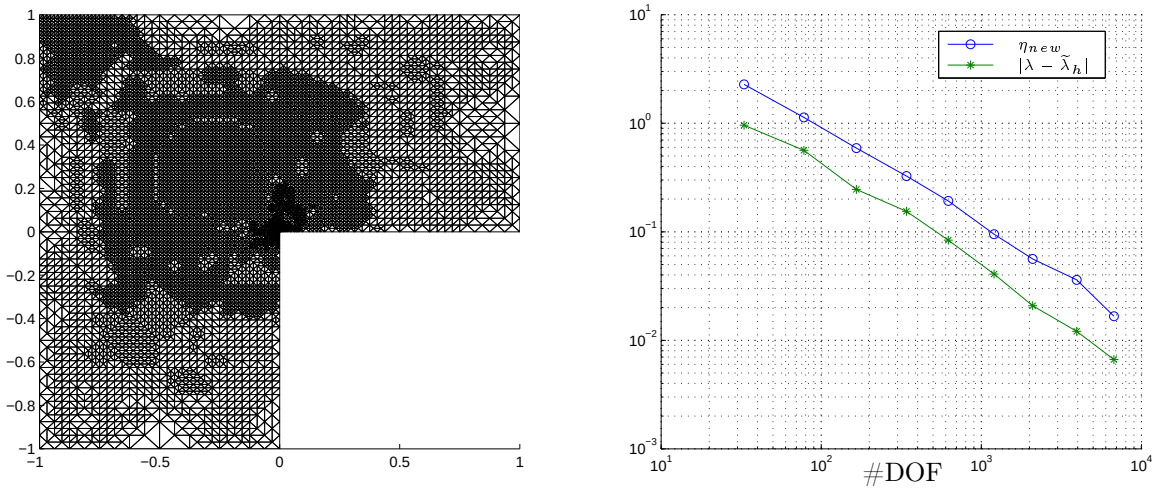


Figure 4.31: The final mesh with 6737 degrees of freedom and the convergence history for the smallest eigenvalue for (3.1) on the L-shape domain with $\omega = 0.5$.

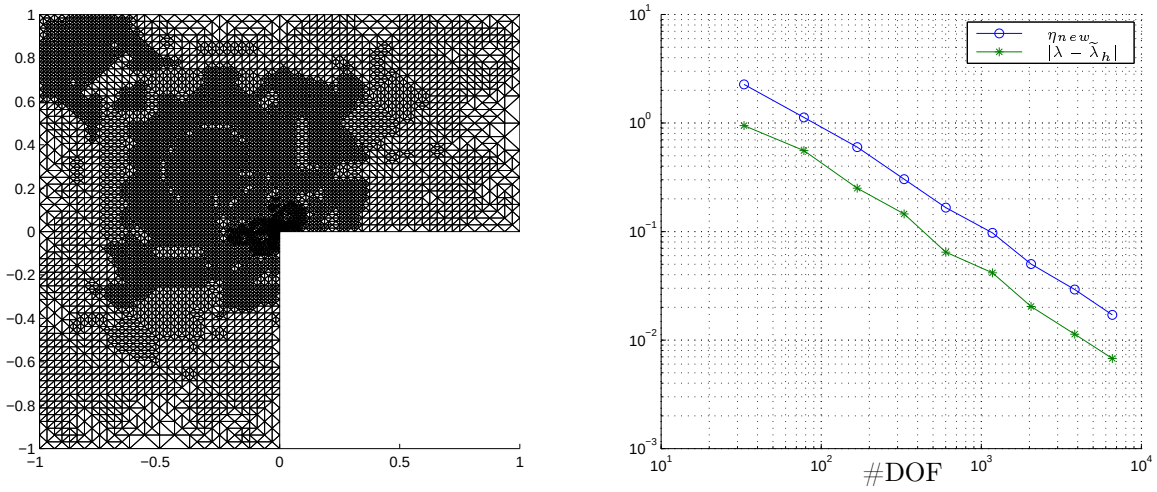


Figure 4.32: The final mesh with 6585 degrees of freedom and the convergence history for the smallest eigenvalue for (3.1) on the L-shape domain with $\omega = 0.1$.

Table 4.11: Approximations of the smallest eigenvalue for (3.1) on the L-shape domain with $\omega = 0.1$.

ref. level	#DOF	# iteration	$\widetilde{\lambda}_1$	$\lvert\lambda_1 - \widetilde{\lambda}_1\rvert$	$\eta(\widetilde{\lambda}_h, \widetilde{u}_h)$	$\frac{\lVert \mathbf{r} \rVert_{A_\sigma^{-1}\mathbf{H}A_\sigma^{-1}}}{\lVert \widetilde{\mathbf{u}}_h \rVert_{\mathbf{H}}}$	$\eta_{new}(\widetilde{\lambda}_h, \widetilde{u}_h)$
1	33	3	10.5862	0.9465	2.2422	0.0284	2.2706
2	78	3	10.1976	0.5578	1.0908	0.0370	1.1278
3	168	3	9.8906	0.2509	0.5550	0.0448	0.5998
4	330	4	9.7850	0.1452	0.2964	0.0080	0.3044
5	601	4	9.7043	0.0646	0.1588	0.0075	0.1663
6	1174	4	9.6814	0.0416	0.0886	0.0087	0.0973
7	2048	5	9.6601	0.0204	0.0489	0.0013	0.0502
8	3839	5	9.6510	0.0113	0.0280	0.0012	0.0292
9	6585	5	9.6465	0.0068	0.0158	0.0013	0.0171

Table 4.12: Approximations of the smallest eigenvalue for (3.1) on the L-shape domain with $\omega = 0.9$.

ref. level	#DOF	# iteration	$\widetilde{\lambda}_1$	$\lvert\lambda_1 - \widetilde{\lambda}_1\rvert$	$\eta(\widetilde{\lambda}_h, \widetilde{u}_h)$	$\frac{\lVert \mathbf{r} \rVert_{A_\sigma^{-1}\mathbf{H}A_\sigma^{-1}}}{\lVert \widetilde{\mathbf{u}}_h \rVert_{\mathbf{H}}}$	$\eta_{new}(\widetilde{\lambda}_h, \widetilde{u}_h)$
1	33	3	10.5838	0.9441	2.2443	0.0244	2.2688
2	78	3	10.2022	0.5625	1.0891	0.0398	1.1289
3	169	3	9.8829	0.2431	0.5535	0.0385	0.5920
4	342	3	9.8110	0.1713	0.2888	0.0507	0.3395
5	638	3	9.7329	0.0932	0.1552	0.0469	0.2021
6	1209	3	9.7104	0.0707	0.0874	0.0457	0.1331
7	2200	4	9.6597	0.0200	0.0462	0.0088	0.0550
8	4028	4	9.6518	0.0121	0.0266	0.0088	0.0355
9	6916	4	9.6473	0.0075	0.0151	0.0088	0.0239

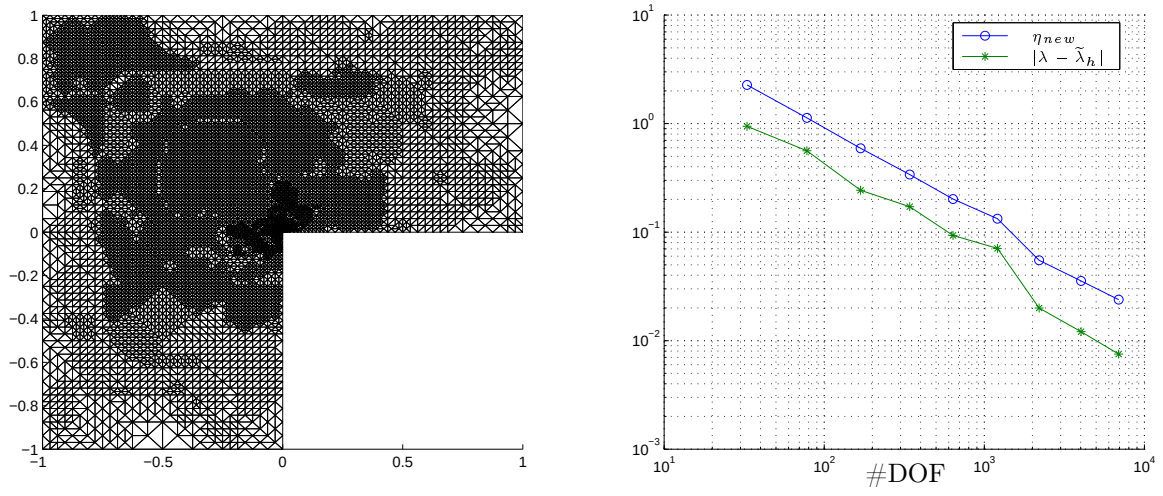


Figure 4.33: The final mesh with 6916 degrees of freedom and the convergence history for the smallest eigenvalue for (3.1) on the L-shape domain with $\omega = 0.9$.

Balancing with $\omega = 0.5$ and different minimal number of required Arnoldi/Lanczos iteration In [81] the minimal number of required Arnoldi/Lanczos iterations, as the well-known rule of thumb, is set to $2k+1$, where k is the number of eigenvalues of interest, here we analyze the actual restrictions in this respect. Table 4.13 contains numerical results for the case where no restrictions are given. At fist refinement steps we see that only one iteration was enough to assure that the iteration error is smaller than the discretization error, however, this one iteration was of course not enough to obtain a good approximation of the eigenvalue. Surprisingly, after these few adaptive steps, the approximate solution has the same accuracy as the corresponding approximation (approximation at the same adaptive step) obtained by the AFEM algorithm starting with a much better approximation, see, e.g., ref. level 5 in Table 4.10 and 4.13. Of course the corresponding grid is much finer in the latter case. The resulting final mesh and the convergence history are presented in Figure 4.34.

Table 4.13: Approximations of the smallest eigenvalue for (3.1) on the L-shape domain with $\omega = 0.5$ and no restriction on the minimal number of iterations.

ref. level	#DOF	# iteration	$\widetilde{\lambda}_1$	$\lvert\lambda_1 - \widetilde{\lambda}_1\rvert$	$\eta(\widetilde{\lambda}_h, \widetilde{u}_h)$	$\frac{\lVert\mathbf{r}\rVert_{A_\sigma^{-1}\mathbf{H}A_\sigma^{-1}}}{\lVert\widetilde{\mathbf{u}}_h\rVert_{\mathbf{H}}}$	$\eta_{new}(\widetilde{\lambda}_h, \widetilde{u}_h)$
1	33	1	37.9495	28.3098	9.3925	1.4206	10.8131
2	103	1	71.6865	62.0467	8.9268	2.1554	11.0823
3	230	1	124.0862	114.4465	8.9370	2.8001	11.7371
4	533	1	253.2587	243.6190	9.1481	4.0423	13.1904
5	1193	3	9.7568	0.1171	0.1237	0.0508	0.1745
6	1533	4	9.6856	0.0459	0.0769	0.0095	0.0865
7	2310	4	9.6642	0.0245	0.0450	0.0089	0.0539
8	4102	4	9.6536	0.0138	0.0257	0.0088	0.0346
9	7313	5	9.6466	0.0069	0.0145	0.0013	0.0157

We have already noticed, that performing only one Arnoldi/Lanczos iteration may not be enough to assure the accuracy and the optimal complexity of our new balanced AFEM algorithm. Nevertheless, our next example shows that performing at least two Arnoldi/Lanczos iterations at every step of our AFEM algorithm seems to be enough. The corresponding numerical results and the convergence history are given in Table 4.14 and Figure 4.35.
As the comparison Table 4.15 and Figure 4.36 present results obtained on the uniformly refined grid. The information about the size of the continuous $\eta(\widetilde{\lambda}_h, \widetilde{u}_h)$ and the discrete $\frac{\lVert\mathbf{r}\rVert_{A_\sigma^{-1}\mathbf{H}A_\sigma^{-1}}}{\lVert\widetilde{\mathbf{u}}_h\rVert_{\mathbf{H}}}$ part of the estimator is given only to illustrate the stopping criterion used for the Arnoldi/Lanczos process. None of this information is used during the grid generation process. To conclude, we point out that obtaining the solution with an accuracy of 10^{-3} for our balanced AFEM algorithm requires two times less degrees of freedom than determining the similar solution on the uniformly refined grid, with the same number of Arnoldi/Lanczos iterations.

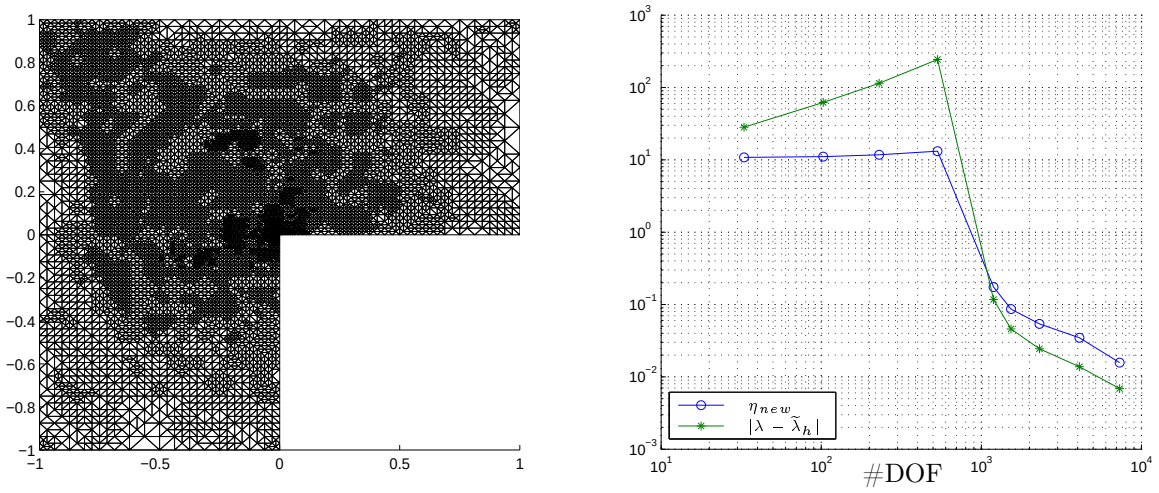


Figure 4.34: The final mesh with 7313 degrees of freedom and the convergence history for the smallest eigenvalue for (3.1) on the L-shape domain with $\omega = 0.5$ and no restriction on the minimal number of iterations.

Comparison with different error estimators As we mentioned at the beginning of this section our new combined error estimator consists of two parts: the continuous part $\eta(\widetilde{\lambda}_h, \widetilde{u}_h)$ and the discrete part $\frac{\|\mathbf{r}\|_{A_\sigma^{-1}\mathbf{H}A_\sigma^{-1}}}{\|\widetilde{\mathbf{u}}_h\|_{\mathbf{H}}}$. So far we have analyzed the behavior of our new balanced AFEM algorithm where the continuous estimator was chosen as $\eta_{DPR}(\widetilde{\lambda}_h, \widetilde{u}_h)$ [50]. Let us now first compare the performance of our new balanced AFEM algorithm with the standard AFEM algorithm based on the $\eta_{DPR}(\widetilde{\lambda}_h, \widetilde{u}_h)$ estimator, where the underlying algebraic eigenvalue problem will be solved with MATLAB function *eigs* up to final accuracy. The resulting meshes are presented in Figure 4.37, while Figure 4.38 shows the convergence history. Both algorithms result in the optimal convergence with respect to the number of degrees of freedom. The new combined error estimator, due to the right choice of the discrete norm, estimates the real error much better than the standard residual type estimator $\eta_{DPR}(\widetilde{\lambda}_h, \widetilde{u}_h)$. Furthermore, we compare our balancing algorithm with the standard AFEM algorithm where $\eta(\widetilde{\lambda}_h, \widetilde{u}_h)$ is chosen as the edge residual error estimator $\eta_{CG}(\widetilde{\lambda}_h, \widetilde{u}_h)$ [31]. We observe that the original $\eta_{CG}(\widetilde{\lambda}_h, \widetilde{u}_h)$ guarantees the optimal convergence rate with respect to the number of degrees of freedom. Although, our balanced AFEM algorithm deviate slightly from the optimal convergence, it almost perfectly captures the behavior of the real error, see Figures 4.39 – 4.40.

Conclusions We have analyzed the behavior of the balanced AFEM algorithm based on the combined a posteriori error estimator $\eta_{new}(\widetilde{\lambda}_h, \widetilde{u}_h)$. Some aspects of reducing the number of the Arnoldi/Lanczos iterations in the algebraic eigensolver and the influence of the parameter ω on the performance of the algorithm were discussed. The results show potential for further investigations of the algorithm, e.g., defining different stopping criteria, i.e., guaranteeing certain reduction of the continuous or the discrete error in the next iteration

Table 4.14: Approximations of the smallest eigenvalue for (3.1) on the L-shape domain with $\omega = 0.5$ and the minimal number of required iterations equal 2.

ref. level	#DOF	# iteration	$\widetilde{\lambda}_1$	$\lvert\lambda_1 - \widetilde{\lambda}_1\rvert$	$\eta(\widetilde{\lambda}_h, \widetilde{u}_h)$	$\frac{\lVert \mathbf{r} \rVert_{A_\sigma^{-1}\mathbf{H}A_\sigma^{-1}}}{\lVert \widetilde{\mathbf{u}}_h \rVert_{\mathbf{H}}}$	$\eta_{new}(\widetilde{\lambda}_h, \widetilde{u}_h)$
1	33	2	10.8075	1.1678	2.3114	0.1185	2.4299
2	86	2	10.5196	0.8799	1.1524	0.1641	1.3165
3	157	2	10.2919	0.6522	0.6536	0.1726	0.8262
4	339	3	9.7909	0.1512	0.2915	0.0415	0.3330
5	610	3	9.7367	0.0970	0.1552	0.0498	0.2049
6	1159	4	9.6802	0.0404	0.0852	0.0089	0.0941
7	2103	4	9.6592	0.0195	0.0466	0.0085	0.0550
8	3856	4	9.6517	0.0119	0.0265	0.0086	0.0351
9	6736	5	9.6459	0.0062	0.0150	0.0013	0.0163

Table 4.15: Approximations of the smallest eigenvalue for (3.1) on the L-shape domain refined uniformly with $\omega = 0.5$ and the minimal number of required iterations equal 2.

ref. level	#DOF	# iteration	$\widetilde{\lambda}_1$	$\lvert\lambda_1 - \widetilde{\lambda}_1\rvert$	$\eta(\widetilde{\lambda}_h, \widetilde{u}_h)$	$\frac{\lVert \mathbf{r} \rVert_{A_\sigma^{-1}\mathbf{H}A_\sigma^{-1}}}{\lVert \widetilde{\mathbf{u}}_h \rVert_{\mathbf{H}}}$	$\eta_{new}(\widetilde{\lambda}_h, \widetilde{u}_h)$
1	33	2	11.2225	1.5828	2.4189	0.1851	2.6040
2	161	2	10.4686	0.8289	0.6589	0.1867	0.8455
3	705	3	9.7616	0.1219	0.1711	0.0476	0.2186
4	2945	4	9.6710	0.0312	0.0520	0.0092	0.0612
5	12033	5	9.6504	0.0107	0.0162	0.0012	0.0174

or incorporating the discrete residual into the marking process in the similar fashion as in the AFEMLA algorithm. Also the efficiency of the estimator is of a particular interest.

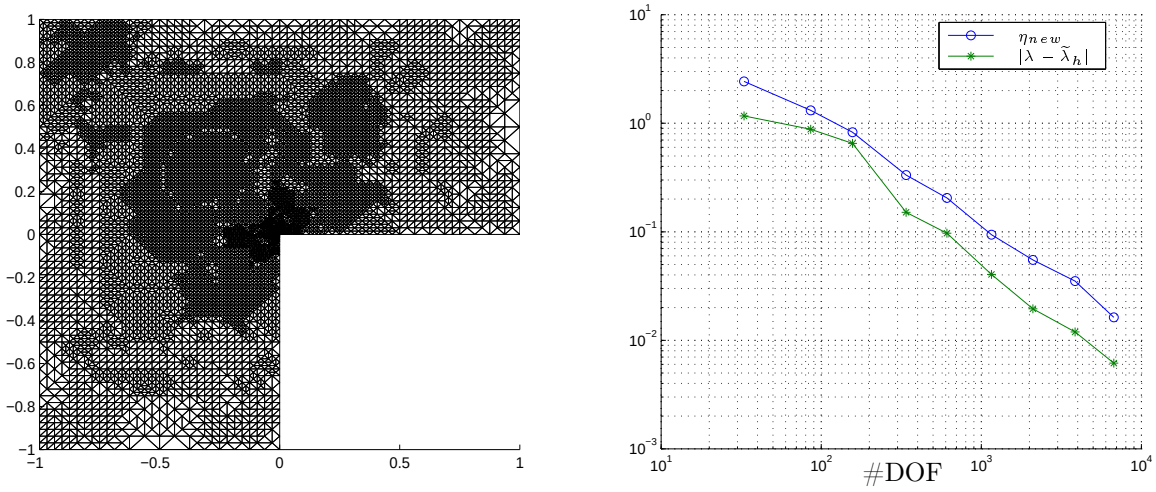


Figure 4.35: The final mesh with 6736 degrees of freedom and the convergence history for the smallest eigenvalue for (3.1) on the L-shape domain with $\omega = 0.5$ and the minimal number of required iterations equal 2.

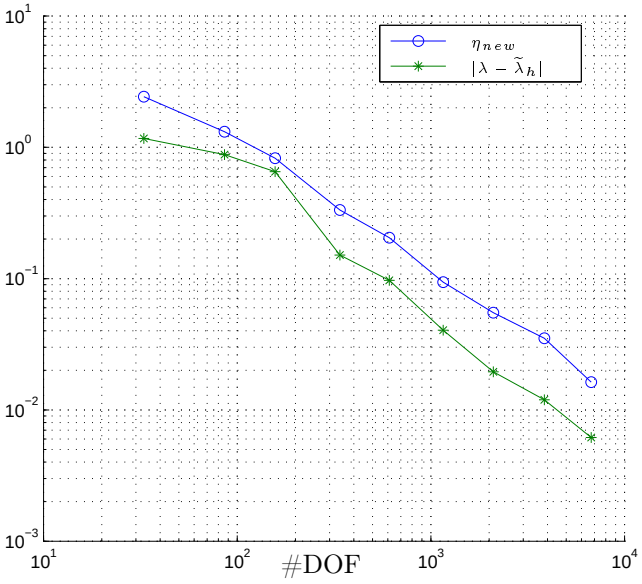


Figure 4.36: The convergence history for the smallest eigenvalue for (3.1) on the uniformly refined L-shape domain with 12033 degrees of freedom, $\omega = 0.5$ and the minimal number of required iterations equal 2.

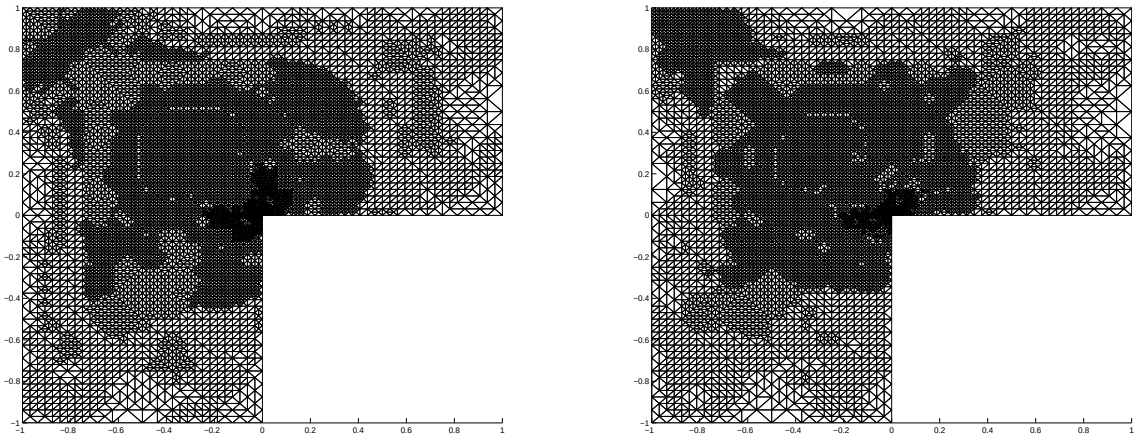


Figure 4.37: The final mesh for the AFEM with η_{DPR} with 6990 degrees of freedom (left) and with η_{new} with 6610 degrees of freedom (right).

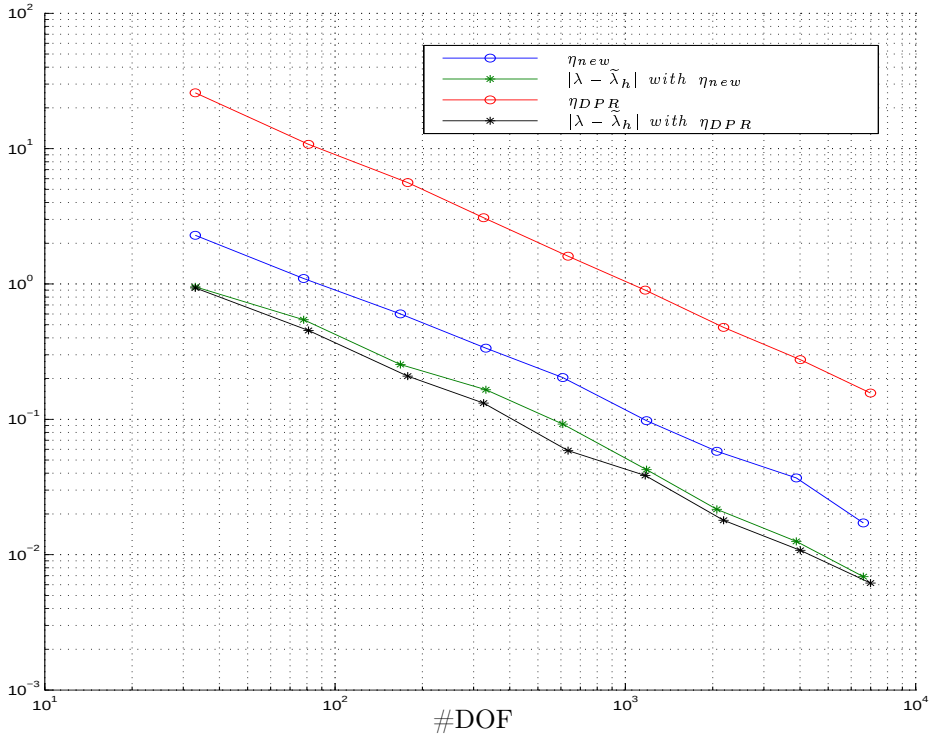


Figure 4.38: The convergence history for the smallest eigenvalue for (3.1) on the L-shape domain.

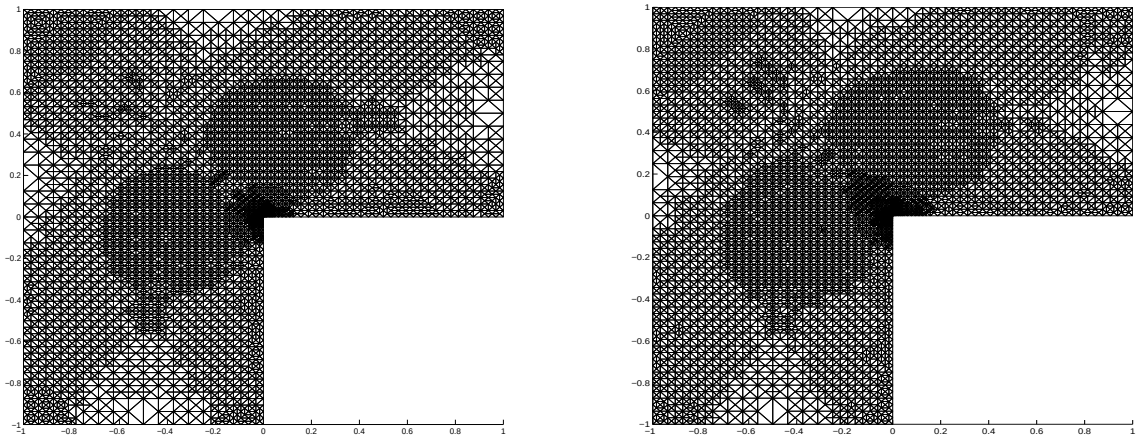


Figure 4.39: The final mesh for the AFEM with η_{CG} with 5577 degrees of freedom (left) and the final mesh for AFEM with η_{new} with 5995 degrees of freedom (right).

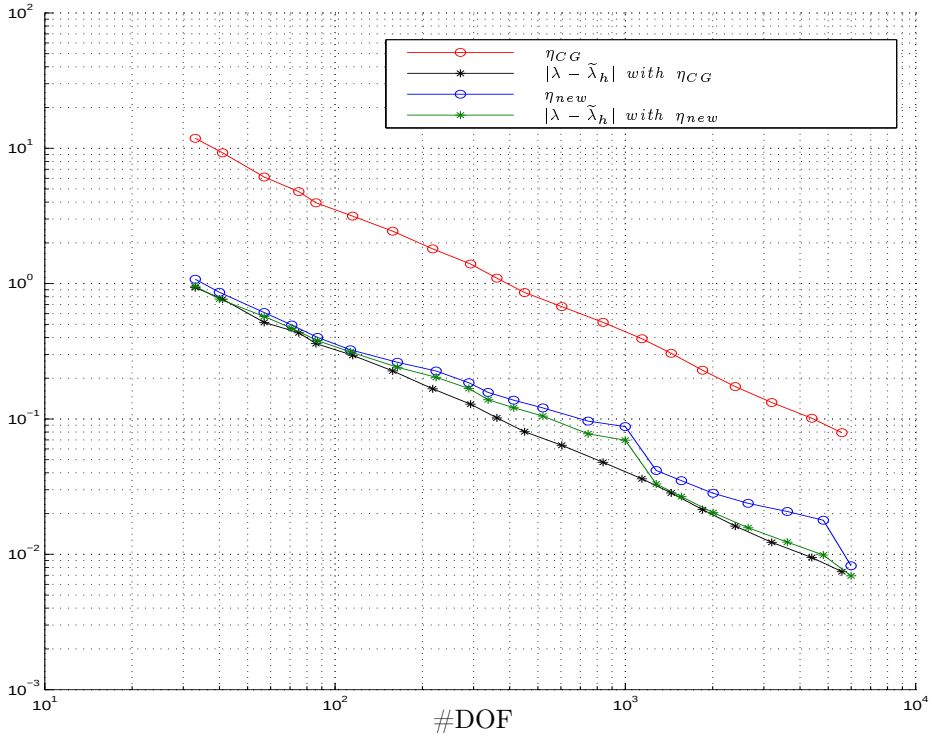


Figure 4.40: The convergence history for the smallest eigenvalue for (3.1) on the L-shape domain.

Chapter 5

Non-self-adjoint eigenvalue problem

The difficulty with non-self-adjoint PDE eigenvalue problems is multifold, eigenvalues may be complex or may have linearly dependent eigenvectors. The basic scope of this chapter is to design adaptive finite element methods which can be used to determine eigenvalues and eigenfunctions of non-self-adjoint problems. We restrict our investigation to the class of convection-diffusion problems, however, all discussed approaches can be further generalized. As an introduction we look at the extended version of the AFEMLA algorithm introduced in Section 4.2, and discuss the adaptation based on both, the right and the left residual vector, see Section 5.1. Under the saturation assumption, we obtain error estimates for the simple eigenvalue and the corresponding eigenfunction for the problems with a small convection. In Section 5.2 we look at those non-self-adjoint problems in terms of homotopy methods. We discuss the adaptive homotopy approach which combines a continuation method with the mesh adaptivity and algebraic eigensolvers to fully utilize an idea of a multi-way adaptivity. The algorithm combines three different kinds of errors: the homotopy error, the discretization error and the algebraic error. Our numerical experiments confirm that in order to derive an efficient and accurate algorithm neither of these errors can be ignored. The results presented in Section 5.2 are a joint work with C. Carstensen, J. Gedicke and V. Mehrmann and are published in [33].

5.1 The Non-self-adjoint AFEMLA

In Section 4.2 we have introduced the AFEMLA algorithm for the self-adjoint eigenvalue problem as an example of the adaptive finite element method based on the iteration error. This section is devoted to extend this idea to the non-self-adjoint eigenvalue problem with an elliptic partial differential operator which is non-self-adjoint but normal or whose departure from non-normality is not large. We derive the non-self-adjoint AFEMLA algorithm to determine an approximation of some particularly chosen eigentriple $(\lambda_h, u_h, u_h^\star)$, e.g., the eigenvalue with the smallest real part and its corresponding eigenvectors. For simplicity we concentrate on the convection-diffusion model problem (3.6). We use here the notation from Section 4.2.

The $\mathbb{P}_1$ finite element discretization of the primal and the dual problem (3.7) on space V_h leads to two generalized eigenvalue problems

$$(A_h + C_h)\mathbf{u}_h = \lambda_h B_h \mathbf{u}_h, \qquad \mathbf{u}_h^\star(A_h + C_h) = \lambda_h^\star \mathbf{u}_h^\star B_h, \tag{5.1}$$

where matrices A_h, B_h are symmetric and positive definite, matrix C_h is non-symmetric, see Section 3.2 for details. Therefore, in order to determine the eigentriple $(\lambda_h, u_h, u_h^\star)$, we need to compute the approximate eigenvalue $\widetilde{\lambda}_h$ together with the corresponding right and left eigenvectors $\widetilde{\mathbf{u}}_h, \widetilde{\mathbf{u}}_h^\star$. Since matrix $A_h + C_h$ is not symmetric we have to solve the generalized eigenvalue problem twice for matrix $A_h + C_h$ and its transpose.

5.1.1 The Non-self-adjoint AFEMLA algorithm

Similarly as in the symmetric case we start the algorithm from computing the coarse grid approximation $(\lambda_H, \widetilde{\mathbf{u}}_H, \widetilde{\mathbf{u}}_H^\star)$ on the initial grid $\mathcal{T}_H$ using the Arnoldi method for matrices $A_H + C_H$ and $(A_H + C_H)^T$. As an alternative the two-sided Arnoldi algorithm proposed in [100] can be applied to further reduce the computation complexity. Next we prolongate the computed eigenvectors $\widetilde{\mathbf{u}}_H, \widetilde{\mathbf{u}}_H^\star$ on the uniformly refined grid $\mathcal{T}_h$ to obtain the fine space approximations $(\widehat{\lambda}_h, \widehat{\mathbf{u}}_h, \widehat{\mathbf{u}}_h^\star)$. Let us recall that, as in the original AFEMLA algorithm, we distinguish the data corresponding to the coarse and fine space with subscript h and H, respectively. Accordingly to Theorem 2.35 we compute the residual vectors $\widehat{\mathbf{r}}_h$, $\widehat{\mathbf{r}}_h^\star$ associated with the right and the left eigenvector, respectively, i.e.,

$$\widehat{\mathbf{r}}_h = (A_h + C_h)\widehat{\mathbf{u}}_h - \widehat{\lambda}_h B_h \widehat{\mathbf{u}}_h, \quad \widehat{\mathbf{r}}_h^\star = \widehat{\mathbf{u}}_h^\star(A_h + C_h) - \widehat{\lambda}_h \widehat{\mathbf{u}}_h^\star B_h.$$

The crucial element of this approach is that the set of marked elements (edges) $\mathcal{M}_H \subset \mathcal{T}_H$ is determined based on the entries in both residual vectors $\widehat{\mathbf{r}}_h$, $\widehat{\mathbf{r}}_h^\star$. Whenever the i-th entry in the vector $\widehat{\mathbf{r}}_h$ or $\widehat{\mathbf{r}}_h^\star$ is large, then the corresponding basis function should be added to enrich the finite element space. Adapting the grid based on both the right and the left eigenvectors, as the numerical examples indicate, is fully justified as the primal and dual eigenfunction usually has their support on a different parts of the domain. After completing the closure algorithm the green, blue or red refinement is performed. The resulting mesh is taken as a new coarse mesh $\mathcal{T}_H$ for the next loop of the algorithm. The pseudo-code of the algorithm is presented in listing **Non-self-adjoint AFEMLA**.

Non-self-adjoint AFEMLA for the eigenvalue of interest of the convection-diffusion eigenvalue problem (3.6)

Input: An initial regular triangulation $\mathcal{T}_H^i$, a maximal number p of Arnoldi steps or a tolerance *tol* and a desired accuracy ε.

Output: Approximation $\widehat{\lambda}$ to the eigenvalue of interest λ (3.1) together with the corresponding approximate primal and dual eigenfunction $\widehat{u}$, $\widehat{u}^\star$, respectively.

Solve: Compute the eigenvalue of interest $\widetilde{\lambda}_H$ and the associated right and left eigenvector $\widetilde{\mathbf{u}}_H$, $\widetilde{\mathbf{u}}_H^\star$ for the algebraic eigenvalue problem (5.1) associated with the coarse mesh $\mathcal{T}_H^i$. The Arnoldi method will be terminated after p steps or when a desired tolerance *tol* is reached.

Express $\widetilde{\mathbf{u}}_H$, $\widetilde{\mathbf{u}}_H^\star$ using basis functions from the mesh $\mathcal{T}_h^i$ that is obtained by uniformly refining $\mathcal{T}_H^i$. With the prolongation matrix P from the coarse mesh $\mathcal{T}_H^i$ on the fine mesh $\mathcal{T}_h^i$ compute $\widehat{\mathbf{u}}_h = P\widetilde{\mathbf{u}}_H$ and $\widehat{\mathbf{u}}_h^\star = P\widetilde{\mathbf{u}}_H^\star$

Estimate: Determine residual vectors $\widehat{\mathbf{r}}_h$, $\widehat{\mathbf{r}}_h^\star$ associated with eigenvectors $\widehat{\mathbf{u}}_h$, $\widehat{\mathbf{u}}_h^\star$, respectively. Identify all large coefficients in $\widehat{\mathbf{r}}_h$, $\widehat{\mathbf{r}}_h^\star$ and corresponding basis functions (nodes).

if $\|\widehat{\mathbf{r}}_h\| < \varepsilon$ and $\|\widehat{\mathbf{r}}_h^\star\| < \varepsilon$ **then**
  return $(\widehat{\lambda}_h, \widehat{\mathbf{u}}_h, \widehat{\mathbf{u}}_h^\star)$
else
  Mark: Mark all edges that contain identified nodes and apply the closure algorithm.
  Refine: Refine the coarse mesh $\mathcal{T}_H^i$ using the green, blue or red refinement to get $\mathcal{T}_H^{i+1}$.

  Start Algorithm with $\mathcal{T}_H^{i+1}$.
end if

5.1.2 Numerical experiments

The goal of this chapter is to demonsrate the performance of the non-self-adjoint AFEMLA approach on the model convection-diffusion problem (3.6).

$$-\Delta u + \beta \cdot \nabla u = \lambda u \quad \text{in} \quad \Omega \qquad \text{and} \qquad u = 0 \quad \text{on} \quad \partial\Omega.$$

We consider this non-self-adjoint eigenvalue problem with the constant convection coefficient $\beta = [\beta_1, \beta_2]^T \neq [0,0]^T$, and domain Ω being either the unit square, i.e., $\Omega = [0,1] \times [0,1]$, the slit domain or the L-shape domain, i.e., $\Omega = [-1,1] \times [0,1] \cup [-1,0] \times [-1,0]$.

Unit square domain As first example we consider the model convection-diffusion problem on the unit square domain $\Omega = [0,1] \times [0,1]$ with different convection coefficients, i.e., $\beta = [1,0]^T, \beta = [5,0]^T$ etc.. Since the problem is defined on the unit square domain, all eigenvalues are real and given explicitly, see [66], by

$$\frac{\beta_1^2 + \beta_2^2}{4} + \pi^2(k_1^2 + k_2^2), \quad k_1, k_2 \in \mathbb{N}\backslash\{0\}$$

Table 5.1: Approximations of the eigenvalue with the smallest real part of (3.6) obtained by the non-self-adjoint AFEMLA on the square domain.

β	λ_1	$\widetilde{\lambda}_1$	$\lvert\lambda_1 - \widetilde{\lambda}_1\rvert$	#DOF
$\beta = [1,0]^T$	19.9892	19.9993	0.0101	3205
$\beta = [5,0]^T$	25.9892	25.9884	0.0008	5075
$\beta = [10,0]^T$	44.7392	44.7312	0.0080	11245
$\beta = [20,0]^T$	119.7386	119.7908	0.0522	23769

Table 5.2: Approximation of the eigenvalue with the smallest real part of (3.6) on the uniformly refined square domain.

β	λ_1	$\widetilde{\lambda}_1$	$\lvert\lambda - \widetilde{\lambda}_1\rvert$	#DOF
$\beta = [10,0]^T$	44.7392	44.7328	0.0064	16129

with the corresponding primal and dual eigenfunction

$$\pm \exp \frac{\beta_1 x + \beta_2 y}{2} \sin(k_1 \pi x) \sin(k_2 \pi y), \quad k_1, k_2 \in \mathbb{N} \backslash \{0\}.$$

Here, for simplicity, we calculate only the eigentriple corresponding to the smallest eigenvalue, i.e.,

$$\lambda_1 = \frac{\beta_1^2 + \beta_2^2}{4} + 2\pi^2,$$

$$u_1 = \exp \frac{\beta_1 x + \beta_2 y}{2} \sin(\pi x) \sin(\pi y), \quad u_1^\star = -\exp \frac{\beta_1 x + \beta_2 y}{2} \sin(\pi x) \sin(\pi y).$$

Table 5.1 presents eigenvalue approximations obtained with the non-self-adjoint AFEMLA algorithm with different parameters β. The exact reference eigenvalues and errors are given to present the quality of approximations. The corresponding primal and dual eigenfunctions with associated adaptive meshes are depicted in Figures 5.1– 5.4. For this particular problem we were able to compute an accurate approximation also for a quite large convection coefficient, i.e., $\beta = [20,0]^T$. Comparatively, Table 5.2 and Figure 5.5 show the results calculated on the uniformly refined grid. Surprisingly, approximations obtained for the convection coefficient $\beta = [5,0]^T$ are better than those for $\beta = [1,0]^T$.

Slit domain As a second example we consider the convection-diffusion problem on the slit domain. In contrast to the square domain the primal and dual eigenfunctions on the slit, do not have almost symmetric supports. Table 5.3 contains some numerical approximations of the eigenvalue with the smallest real part. The reference eigenvalues were calculated in the finite element framework OPENFFW [29]. Figures 5.6– 5.8 present the corresponding

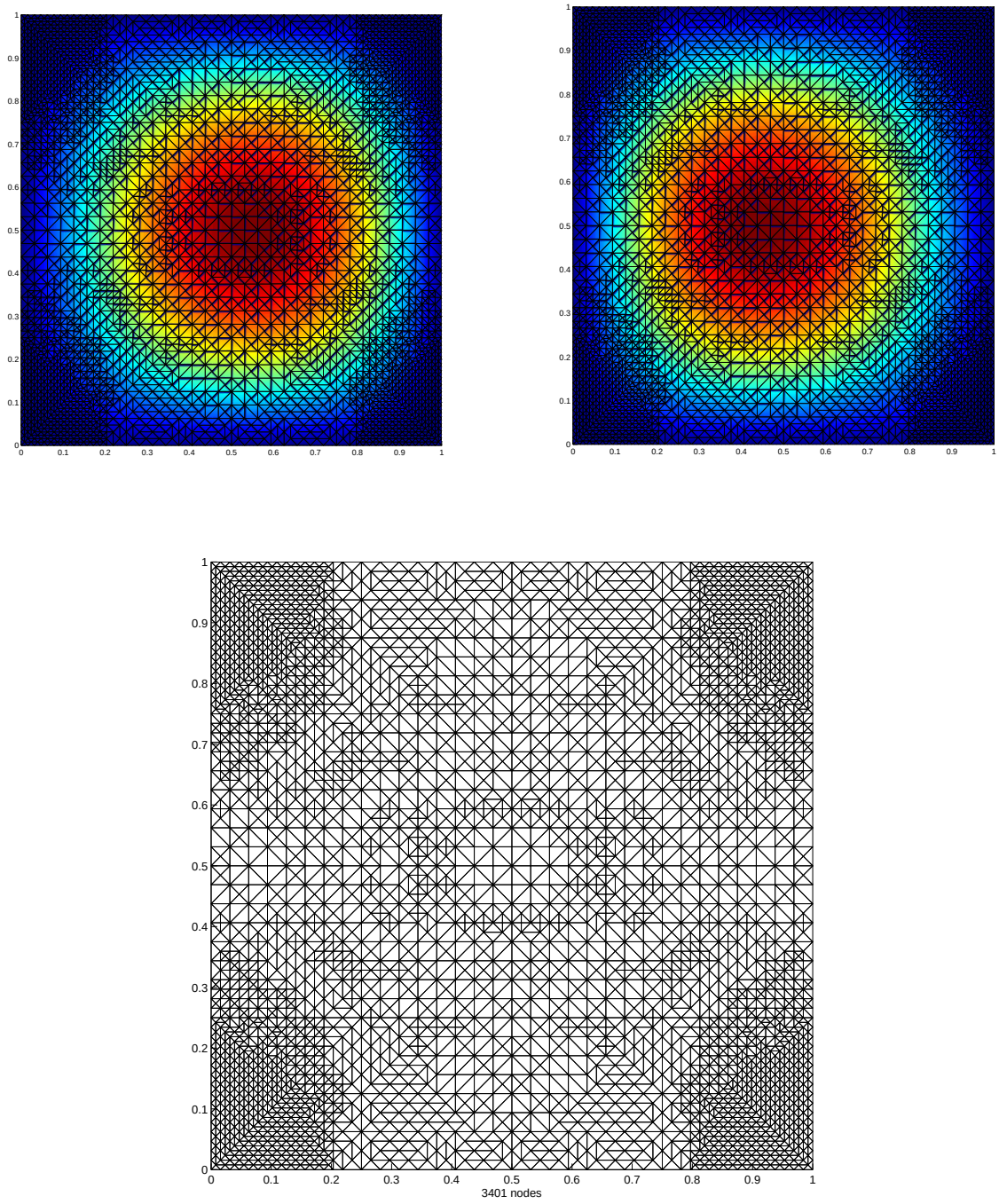


Figure 5.1: Primal (top left) and dual (top right) eigenfunction approximation for the final mesh (bottom) with 3205 degrees of freedom and $\beta = [1, 0]^T$.

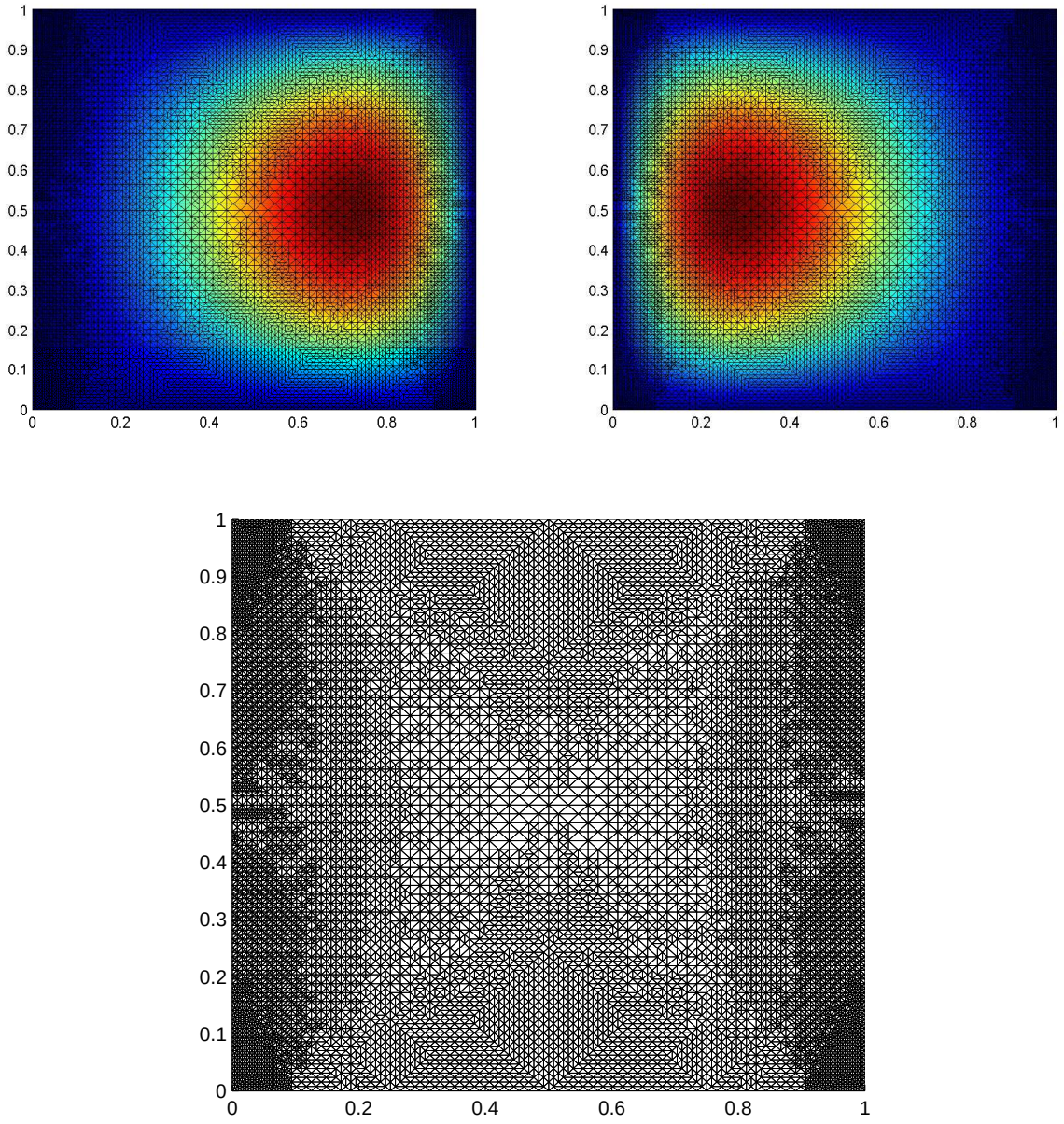


Figure 5.2: Primal (top left) and dual (top right) eigenfunction approximation for the final mesh (bottom) with 5075 degrees of freedom and $\beta = [5, 0]^T$.

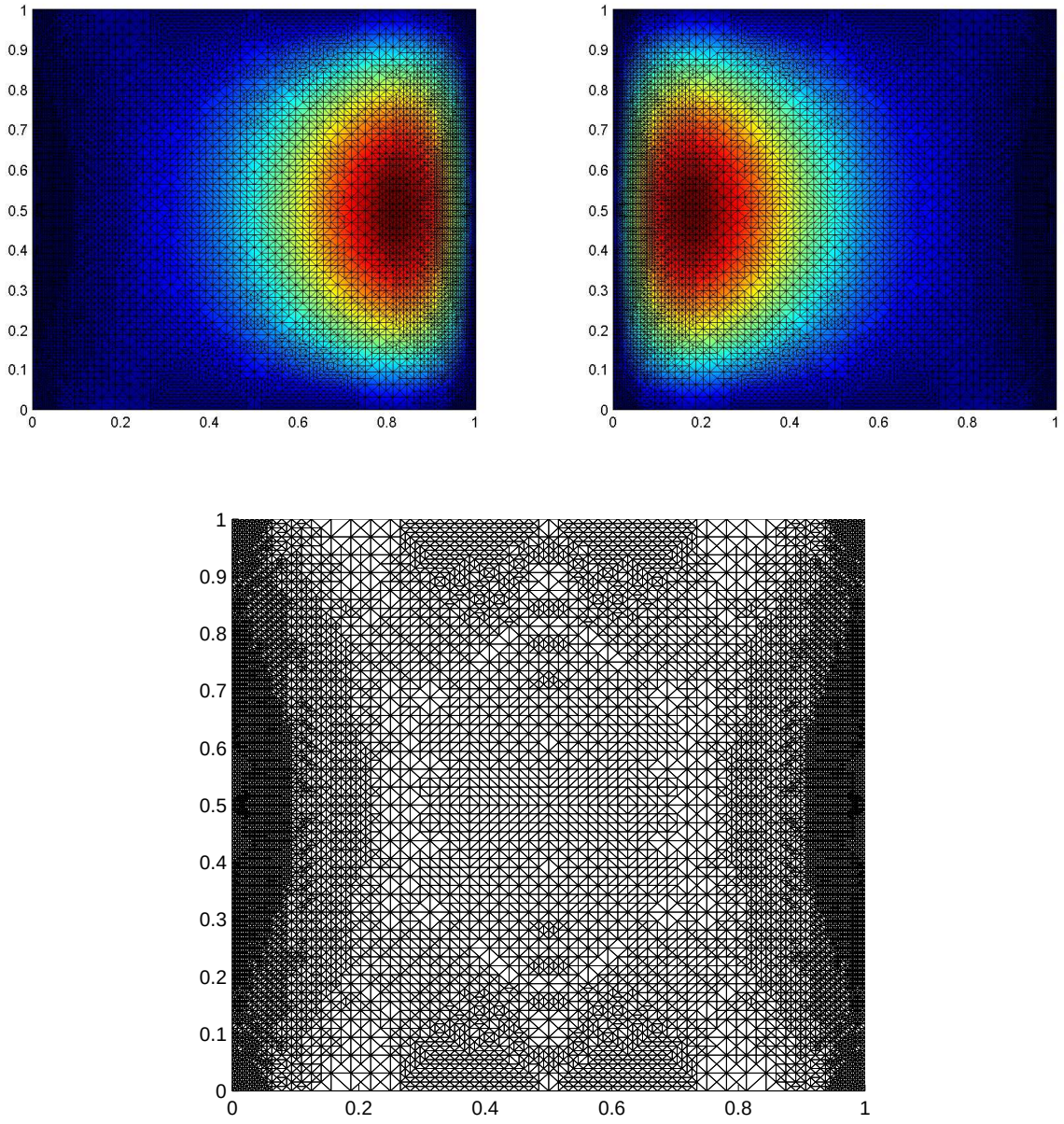


Figure 5.3: Primal (top left) and dual (top right) eigenfunction approximation for the final mesh (bottom) with 11245 degrees of freedom and $\beta = [10, 0]^T$.

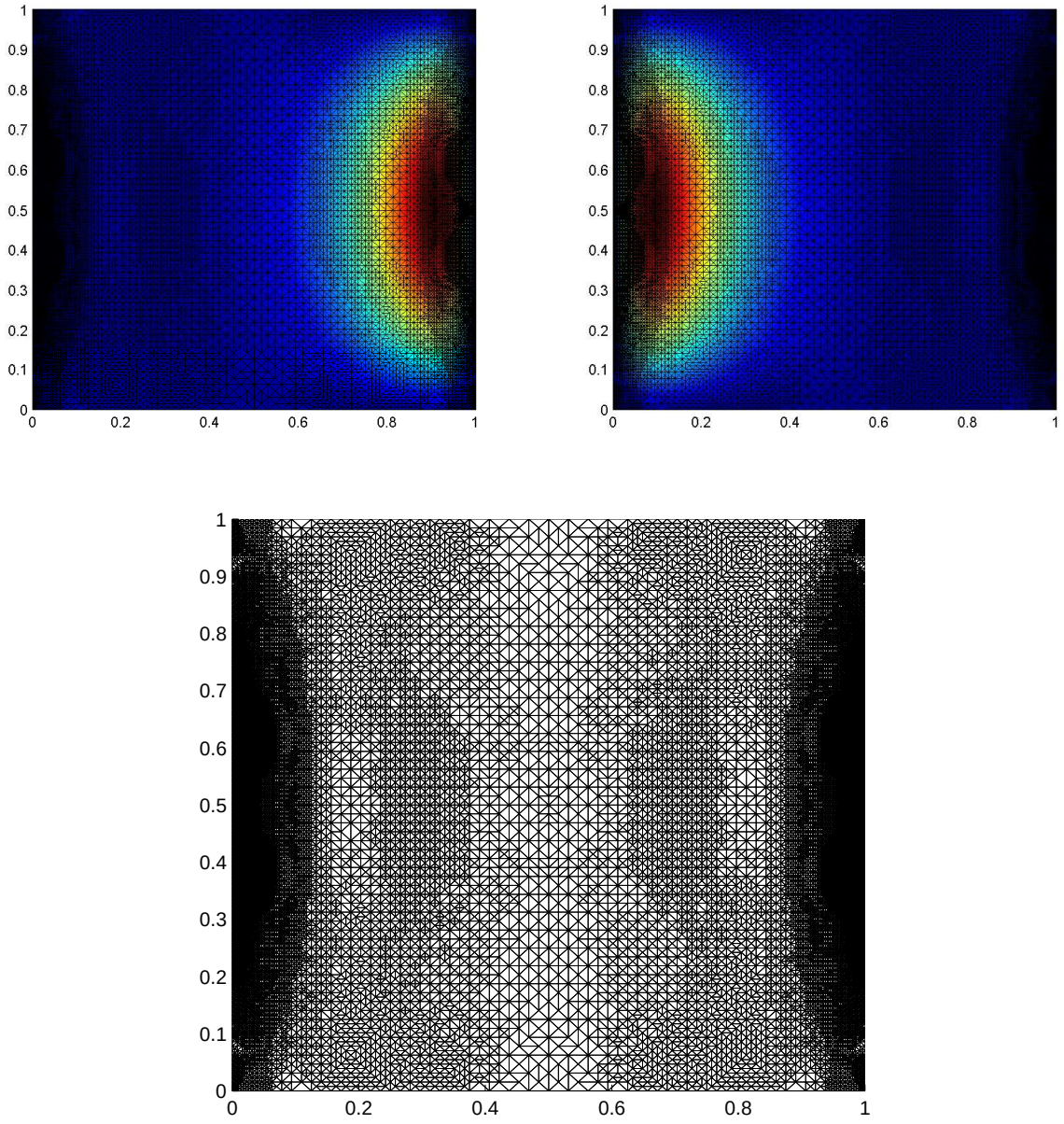


Figure 5.4: Primal (top left) and dual (top right) eigenfunction approximation for the final mesh (bottom) with 23769 degrees of freedom and $\beta = [20, 0]^T$.

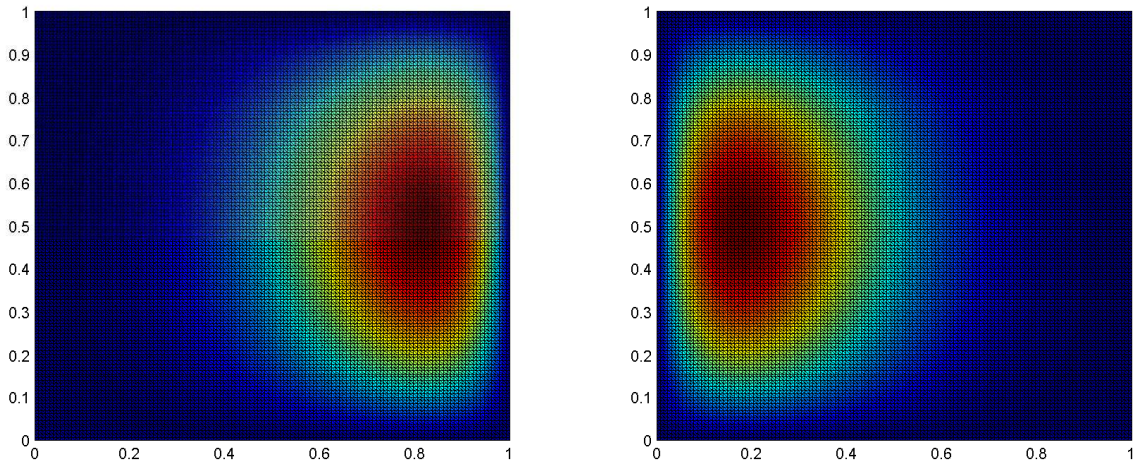


Figure 5.5: Primal (top left) and dual (top right) eigenfunction approximation on the uniformly refined mesh with 16129 degrees of freedom and $\beta = [10, 0]^T$.

Table 5.3: Approximations of the eigenvalue with the smallest real part of (3.6) obtained by the non-self-adjoint AFEMLA on the slit domain.

β	λ_1	$\widetilde{\lambda}_1$	$\lvert\lambda_1 - \widetilde{\lambda}_1\rvert$	#DOF
$\beta = [1, 0]^T$	8.6213	8.6274	0.0061	8338
$\beta = [5, 0]^T$	14.6211	14.6240	0.0029	6618
$\beta = [10, 0]^T$	33.3688	33.4044	0.0356	14159

primal and dual eigenfunction together with final meshes. Unfortunately, for $\beta = [20, 0]^T$ we were not able to calculate a good approximation of the eigenvalue of interest, which confirms that non-normal problems are hard to deal with. We believe that approximating the whole eigenspace corresponding to the close eigenvalues may give better results. For comparison Table 5.4 and Figure 5.9 show results obtained on the uniformly refined grid.

Table 5.4: Approximation of the eigenvalue with the smallest real part of (3.6) on the uniformly refined slit domain.

β	λ_1	$\widetilde{\lambda}_1$	$\lvert\lambda_1 - \widetilde{\lambda}_1\rvert$	#DOF
$\beta = [10, 0]^T$	33.3688	33.4067	0.0379	16065

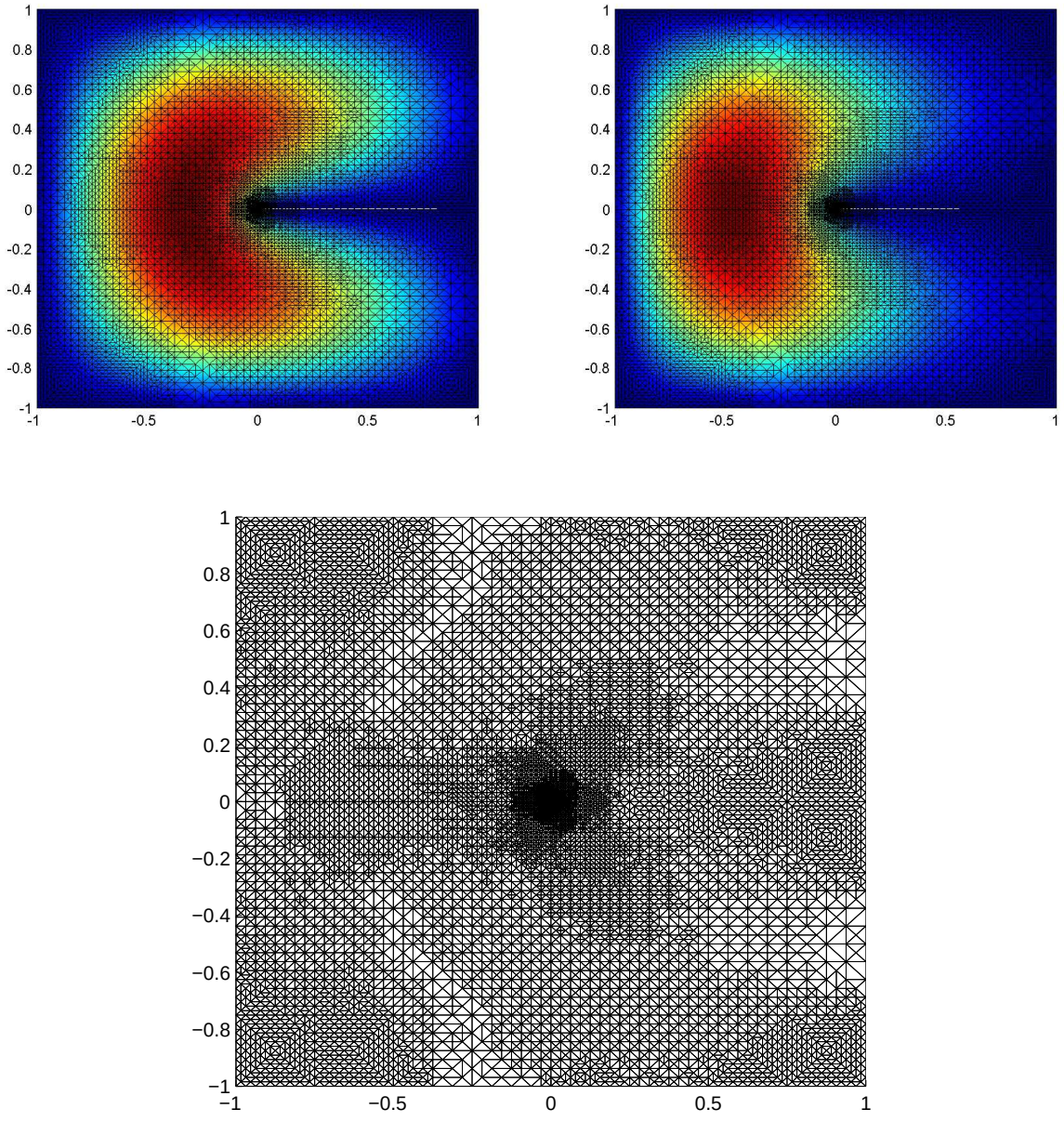


Figure 5.6: Primal (top left) and dual (top right) eigenfunction approximation for the final mesh (bottom) with 8338 degrees of freedom and $\beta = [1, 0]^T$ on the slit domain.

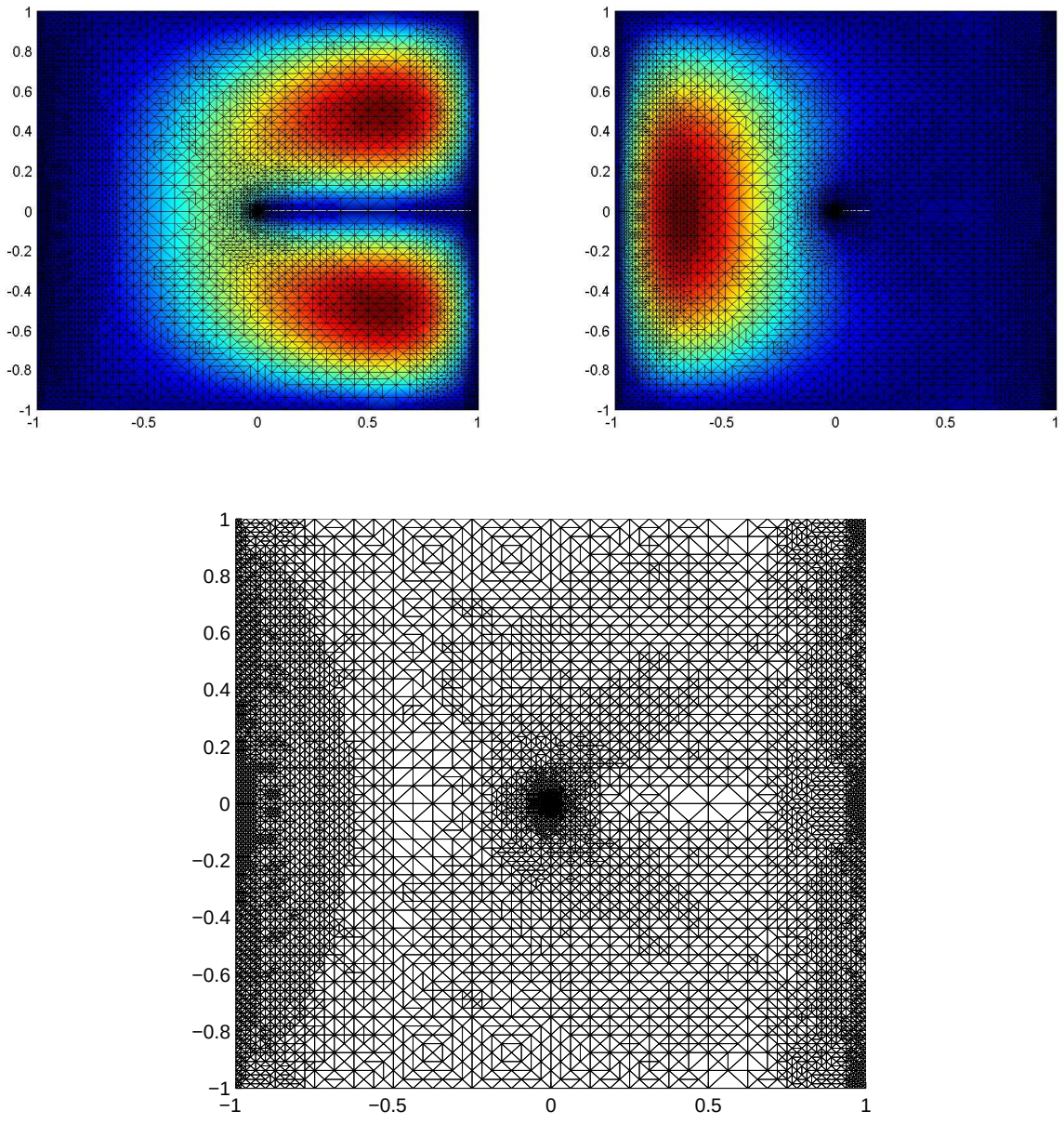


Figure 5.7: Primal (top left) and dual (top right) eigenfunction approximation for the final mesh (bottom) with 6618 degrees of freedom and $\beta = [5, 0]^T$ on the slit domain.

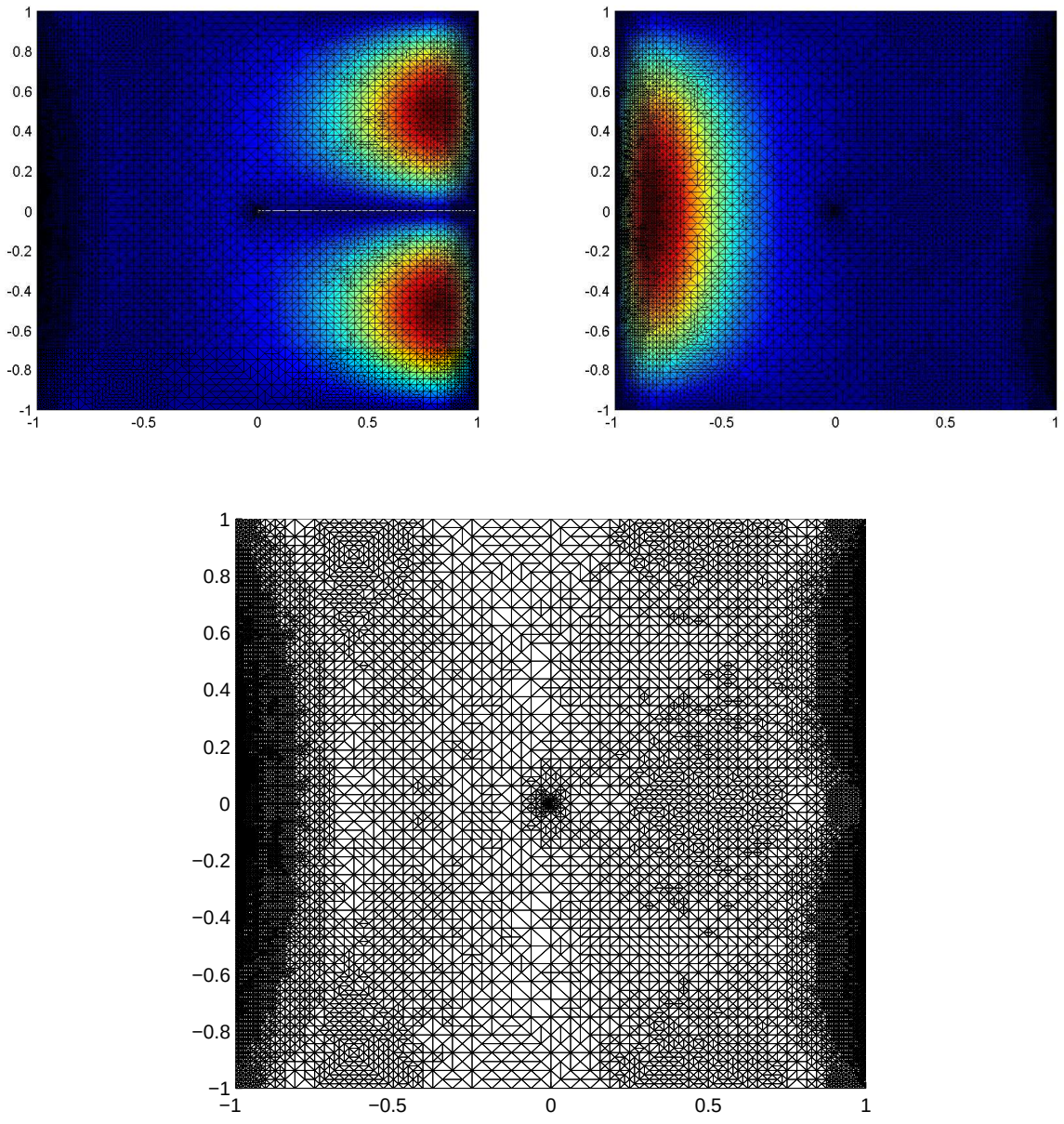


Figure 5.8: Primal (top left) and dual (top right) eigenfunction approximation for the final mesh (bottom) with 14159 degrees of freedom and $\beta = [10, 0]^T$ on the slit domain.

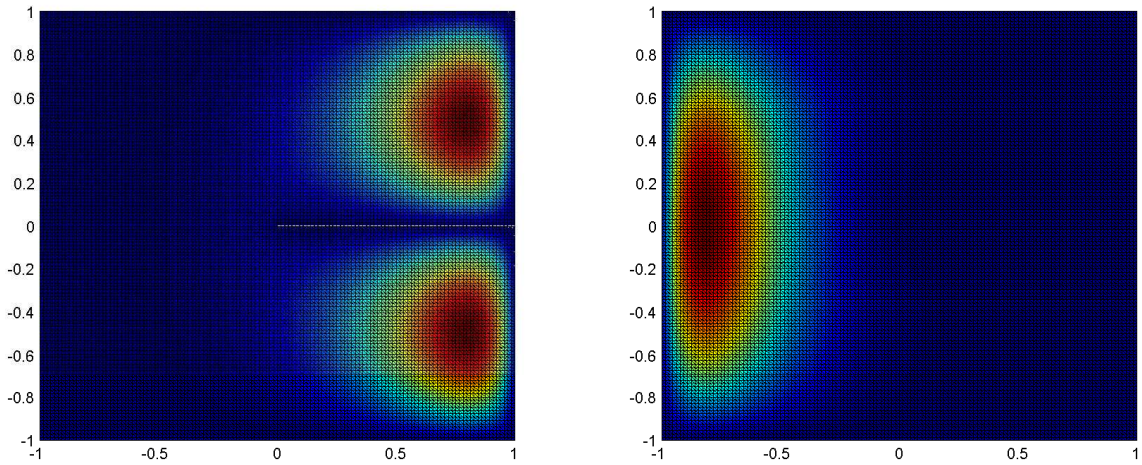


Figure 5.9: Primal (top left) and dual (top right) eigenfunction approximation on the uniformly refined mesh with 16065 degrees of freedom and $\beta = [10, 0]^T$ on the slit domain.

L-shape domain It is well-known that the presence of singularities often deteriorate the quality of the numerical approximation. We investigate the behavior of the non-self-adjoint AFEMLA algorithm for the convection-diffusion problem on the re-entrant corner domain, i.e., the L-shape domain $\Omega = [-1, 1] \times [0, 1] \cup [-1, 0] \times [-1, 0]$. Following previous examples we compute eigentriple approximations for different choices of the parameter β. In this case no explicit formula for exact eigentriples is known, which means that the spectrum of the operator may contain complex eigenvalues. Nevertheless, it is possible to approximate the eigenvalue with the smallest real part. As we mentioned in Section 3.2 this eigenvalue is simple, well-separated and real. Thus, in our numerical examples we will concentrate on determing an accurate approximation of this particular eigenvalue and its primal and dual eigenfunctions. All reference eigenvalues were calculated in the finite element framework OPENFFW [29]. For the small convection term eigenvalue approximations obtained with the non-self-adjoint AFEMLA algorithm are quite accurate, as shown in Table 5.5. The corresponding primal and dual eigenfunctions are depicted in Figures 5.10– 5.10, while the results obtained on the uniformly refined grid are presented in Table 5.6 and in Figure 5.13. These two last examples confirm that for the highly non-normal problem, any method involving only the information from the residual vector, will suffer from the same problems which we faced in previous examples. We also do not observe significant benefits in comparison to the uniformly refined meshes. Therefore, for the real non-symmetric problems completely new methods have to be developed.

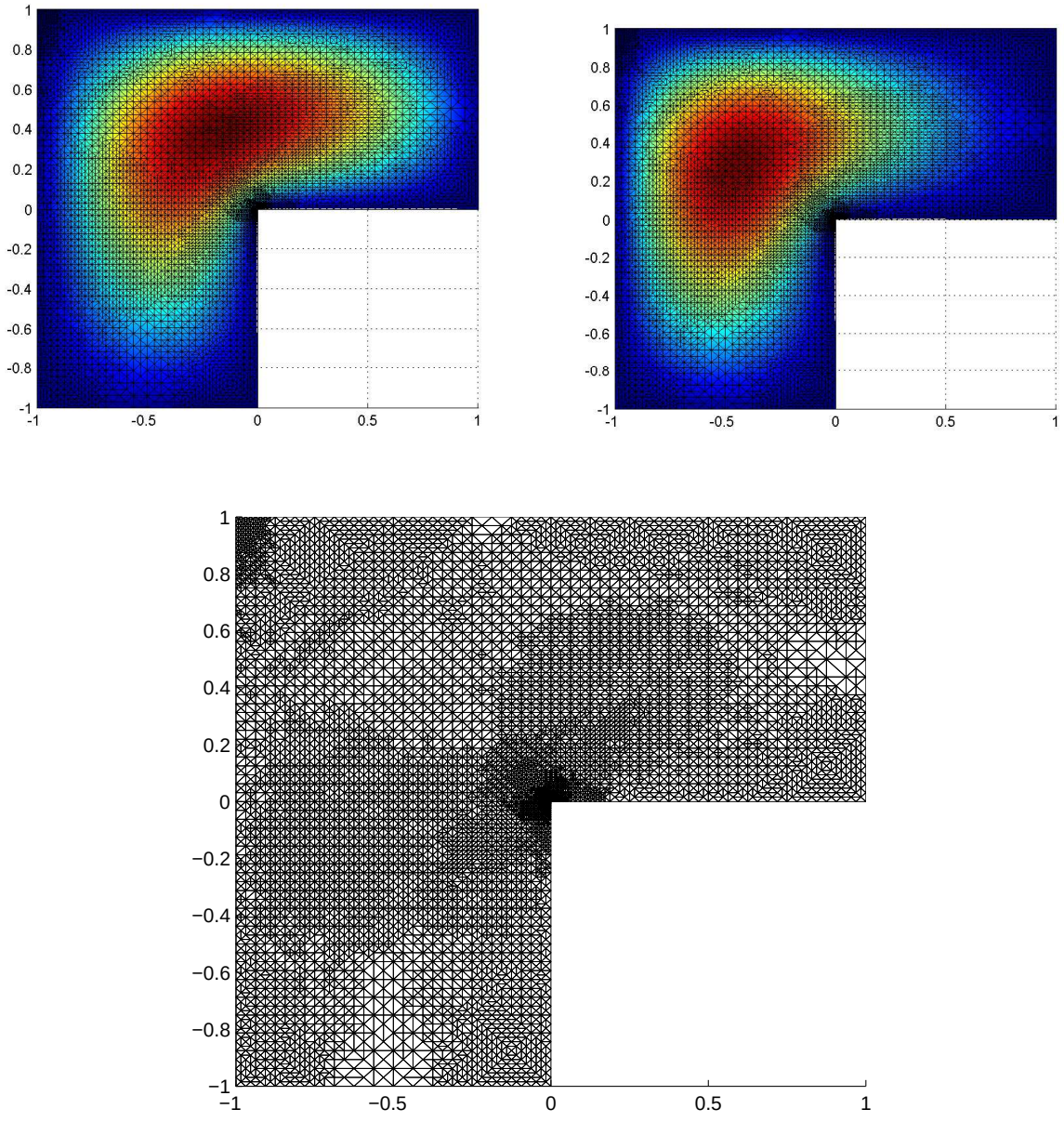


Figure 5.10: Primal (top left) and dual (top right) eigenfunction approximation for the final mesh (bottom) with degrees of freedom and $\beta = [1, 0]^T$ on the L-shape domain.

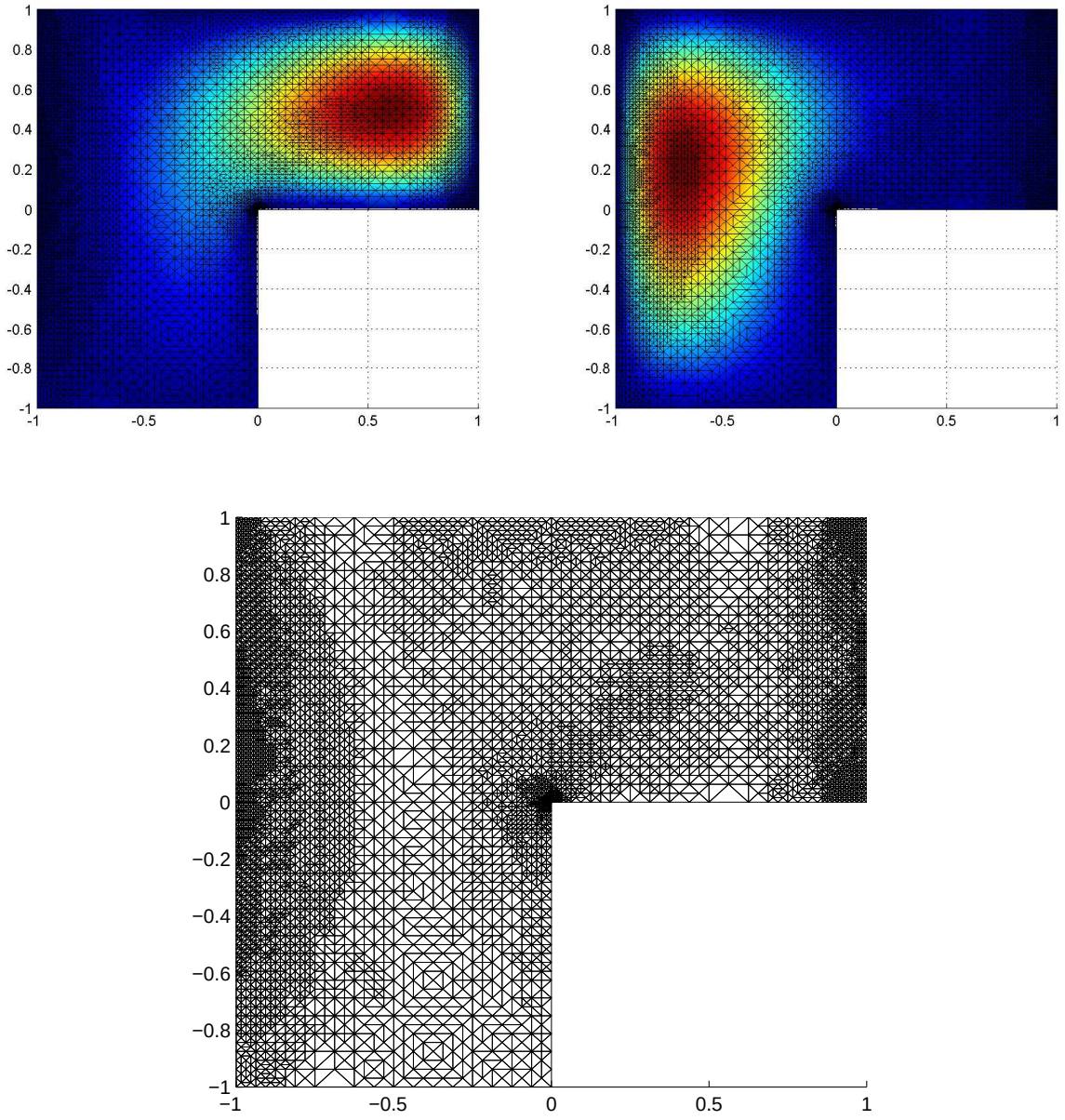


Figure 5.11: Primal (top left) and dual (top right) eigenfunction approximation for the final mesh (bottom) with degrees of freedom and $\beta = [5, 0]^T$ on the L-shape domain.

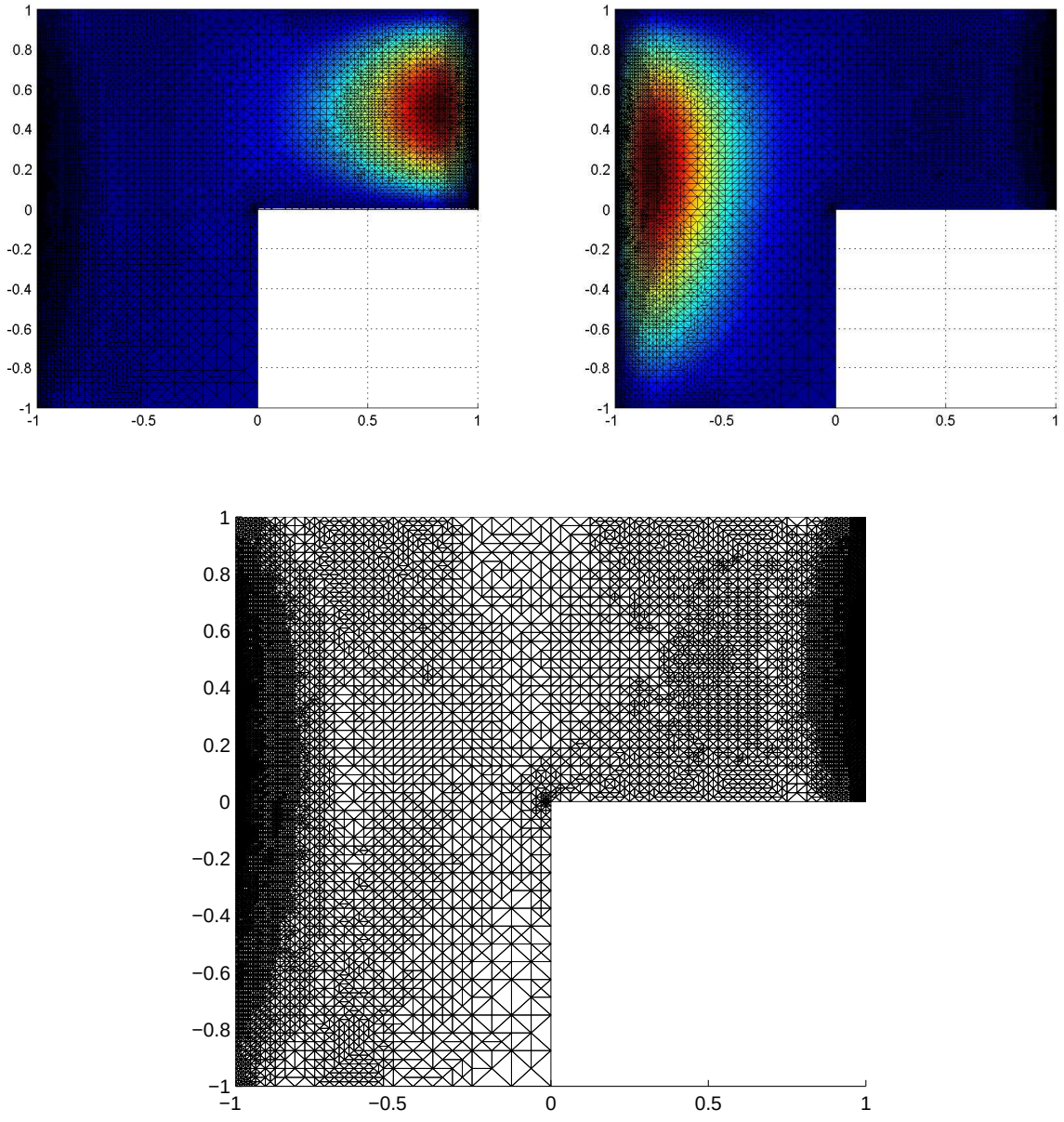


Figure 5.12: Primal (top left) and dual (top right) eigenfunction approximation for the final mesh (bottom) with degrees of freedom and $\beta = [10, 0]^T$ on the L-shape domain.

Table 5.5: Approximations of the eigenvalue with the smallest real part of (3.6) obtained by the non-self-adjoint AFEMLA on the L-shape domain.

β	λ_1	$\widetilde{\lambda}_1$	$\lvert\lambda_1 - \widetilde{\lambda}_1\rvert$	#DOF
$\beta = [1,0]^T$	9.8897	9.8935	0.0038	7318
$\beta = [5,0]^T$	15.8895	15.8886	0.0009	6305
$\beta = [10,0]^T$	34.6398	34.6422	0.0024	11883

Table 5.6: Approximation of the eigenvalue with the smallest real part of (3.6) on the uniformly refined L-shape domain.

β	λ	$\widetilde{\lambda}_1$	$\lvert\lambda_1 - \widetilde{\lambda}_1\rvert$	#DOF
$\beta = [10,0]^T$	34.6398	34.6417	0.0019	12033

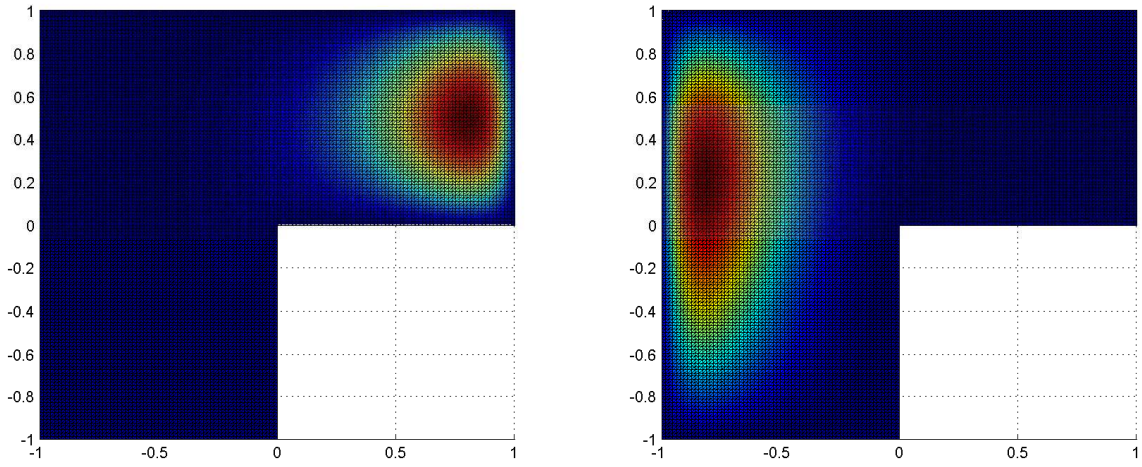


Figure 5.13: Primal (top left) and dual (top right) eigenfunction approximation on the uniformly refined L-shape domain with 12033 degrees of freedom and $\beta = [10,0]^T$.

5.1.3 Some error bounds for the eigenvalues and eigenfunctions of non-self-adjoint eigenvalue problems

In this section we derive error bounds for eigenpairs of non-self-adjoint problems which lead to generalized algebraic eigenvalue problems with non-symmetric but real diagonalizable matrices. For the model convection-diffusion problem (3.6) this will be the case when the convection coefficient β is small. In order to relate continuous and discrete eigentriples we will use the idea introduced in Section 4.2, namely, a combination of perturbation results for the algebraic eigenvalue problems for real diagonalizable matrices from Section 2.3.3.2 together with an appropriate saturation assumption. This analysis can be viewed as a theoretical justification of the non-self-adjoint AFEMLA algorithm introduced in Section 5.1.
Let λ be a simple eigenvalue of the convection-diffusion problem (3.6) and u, $u^\star$ be the corresponding primal and dual eigenfunctions, respectively. Let $(\lambda_H, \mathbf{u}_H, \mathbf{u}_H^\star)$, $(\lambda_h, \mathbf{u}_h, \mathbf{u}_h^\star)$ be exact eigenvalues and associated right and left eigenvectors of matrix pairs $((A_H + C_H), B_H)$, $((A_h + C_h, B_h))$, resulting from the finite element discretization of (3.6) on the coarse and the fine space V_H, V_h, respectively, i.e.,

$$(A_H + C_H)\mathbf{u}_H = \lambda_H \mathbf{u}_H \qquad \mathbf{u}_H^\star(A_H + C_H) = \lambda_H \mathbf{u}_H^\star B_H, \tag{5.2}$$

$$(A_h + C_h)\mathbf{u}_h = \lambda_h \mathbf{u}_h \qquad \mathbf{u}_h^\star(A_h + C_h) = \lambda_h \mathbf{u}_h^\star B_h. \tag{5.3}$$

Matrices B_H, B_h are positive definite, while matrices $(A_H + C_H), (A_h + C_h)$ are real but non-symmetric. However, throughout this section, we assume that the convection coefficient β is chosen such that the matrices $B_H^{-1}(A_H + C_H)$ and $B_h^{-1}(A_h + C_h)$ are real diagonalizable, i.e.,

$$B_H^{-1}(A_H + C_H) = U_H D_H U_H^{-1}, \qquad B_h^{-1}(A_h + C_h) = U_h D_h U_h^{-1}, \tag{5.4}$$

where D_H, D_h are diagonal matrices with eigenvalues of $B_H^{-1}(A_H + C_H)$ and $B_h^{-1}(A_h + C_h)$, while

$$U_H = [\mathbf{u}_H \; U_{H,2}] \qquad U_H^{-1} = \begin{bmatrix} \mathbf{u}_H^\star B_H \\ U_{H,2}^\star B_H \end{bmatrix},$$

$$U_h = [\mathbf{u}_h \; U_{h,2}] \qquad U_h^{-1} = \begin{bmatrix} \mathbf{u}_h^\star B_h \\ U_{h,2}^\star B_h \end{bmatrix},$$

are the right and the left eigenvector matrix in space V_H and V_h, respectively.
Let $(\widetilde{\lambda}_H, \widetilde{\mathbf{u}}_H, \widetilde{\mathbf{u}}_H^\star)$, $(\widetilde{\lambda}_h, \widetilde{\mathbf{u}}_h, \widetilde{\mathbf{u}}_h^\star)$ be approximations of $(\lambda_H, \mathbf{u}_H, \mathbf{u}_H^\star)$ and $(\lambda_h, \mathbf{u}_h, \mathbf{u}_h^\star)$, respectively. Since we do not compute the fine grid eigentriple $(\widetilde{\lambda}_h, \widetilde{\mathbf{u}}_h, \widetilde{\mathbf{u}}_h^\star)$, instead we use the prolongation of $(\widetilde{\lambda}_H, \widetilde{\mathbf{u}}_H, \widetilde{\mathbf{u}}_H^\star)$ on the fine space and denote it by $(\widehat{\lambda}_h, \widehat{\mathbf{u}}_h, \widehat{\mathbf{u}}_h^\star)$. Then, the corresponding residual vectors are given by

$$\mathbf{r}_H = (A_H + C_H)\widetilde{\mathbf{u}}_H - \widetilde{\lambda}_H B_H \widetilde{\mathbf{u}}_H, \tag{5.5}$$

$$\mathbf{r}_h = (A_h + C_h)\widetilde{\mathbf{u}}_h - \widetilde{\lambda}_h B_h \widetilde{\mathbf{u}}_h, \tag{5.6}$$

$$\widehat{\mathbf{r}}_h = (A_h + C_h)\widehat{\mathbf{u}}_h - \widehat{\lambda}_h B_h \widehat{\mathbf{u}}_h. \tag{5.7}$$

Assumption 5.1. *Let λ be a simple eigenvalue of (3.6). Let λ_H, λ_h be the corresponding exact eigenvalues of the discretized problem on spaces V_H, V_h, respectively. Moreover, let u be the exact eigenfunction corresponding to the eigenvalue λ and u_H, u_h the coarse and the fine space approximation, respectively. Then the error of the fine discrete solution $\lambda_h, (u_h)$ is smaller than the error of the coarse solution $\lambda_H, (u_H)$, i.e.,*

$$|\lambda_h - \lambda| \leq c|\lambda_H - \lambda|,$$
$$\sin \angle(u, u_h) \leq C \sin \angle(u, u_H).$$

with $c, C \in (0, 1)$.

Corollary 5.2. *Let (λ, u) be a simple eigenpair of (3.6). Let λ_H, λ_h be the corresponding exact eigenvalues of the discretized problem (3.6) on space V_H, V_h, respectively, and u_H, u_h the corresponding primal eigenfunctions. Then the saturation Assumption 5.1 is equivalent to*

$$|\lambda_H - \lambda| \leq \frac{1}{1-c}|\lambda_H - \lambda_h|.$$
$$\sin \angle(u, u_H) \leq \frac{1}{1-C} \sin \angle(u_H, u_h).$$

with positive $c, C < 1$.

Proof. See Theorem 4.8 and Corollary 4.17. □

Corollary 5.3. *Consider generalized eigenvalue problems (5.2), (5.3) with exact eigenpairs $(\lambda_H, \mathbf{u}_H)$, $(\lambda_h, \mathbf{u}_h)$ and corresponding approximations $(\widetilde{\lambda}_H, \widetilde{\mathbf{u}}_H)$, $(\widetilde{\lambda}_h, \widetilde{\mathbf{u}}_h)$, respectively. Let $(\widehat{\lambda}_h, \widehat{\mathbf{u}}_h)$ be obtained by the prolongation of $(\widetilde{\lambda}_H, \widetilde{\mathbf{u}}_H)$ on the fine space V_h. Furthermore, residual vectors $\mathbf{r}_H$, $\mathbf{r}_h$, $\widehat{\mathbf{r}}_h$ are defined as in (5.5), (5.6), (5.7). Then the following bounds for the distance between eigenvalues hold*

$$|\widetilde{\lambda}_H - \lambda_H| \leq \kappa(B_H^{\frac{1}{2}} U_H) \frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}}, \tag{5.8}$$
$$|\widetilde{\lambda}_h - \lambda_h| \leq \kappa(B_h^{\frac{1}{2}} U_h) \frac{\|\mathbf{r}_h\|_{B_h^{-1}}}{\|\widetilde{\mathbf{u}}_h\|_{B_h}}, \tag{5.9}$$
$$|\widehat{\lambda}_h - \lambda_h| \leq \kappa(B_h^{\frac{1}{2}} U_h) \frac{\|\widehat{\mathbf{r}}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}}, \tag{5.10}$$

with right eigenvector matrices $U_H = [\mathbf{u}_H \; U_{H,2}]$ and $U_h = [\mathbf{u}_h \; U_{h,2}]$.

Proof. The proof follows directly from applying Theorem 2.46. □

Corollary 5.4. *Let $(\widetilde{\lambda}_H, \widetilde{\mathbf{u}}_H)$ be a computed eigenpair of the generalized eigenvalue problem (5.2)and let $(\widehat{\lambda}_h, \widehat{\mathbf{u}}_h)$ be the prolongation (with the prolongation matrix P) of $(\widetilde{\lambda}_H, \widetilde{\mathbf{u}}_H)$ on the fine space V_h. Furthermore, let residual vectors $\mathbf{r}_H$, $\widehat{\mathbf{r}}_h$ be defined as in (5.5), (5.7). Then*

$$|\widetilde{\lambda}_H - \widehat{\lambda}_h| \leq \frac{\|\mathbf{r}_H\|_{B_H} + \|P^T\widehat{\mathbf{r}}_h\|_{B_H}}{\|P^T B_h \widehat{\mathbf{u}}_h\|_{B_H}}.$$

Proof. Following Proposition 2.34, the eigenpairs $(\widetilde{\lambda}_H, \widetilde{\mathbf{u}}_h)$, $(\widehat{\lambda}_h, \widehat{\mathbf{u}}_h)$ are exact eigenpairs of eigenvalue problems

$$((A_H + C_H) + E_H)\widetilde{\mathbf{u}}_H = \widetilde{\lambda}_H B_H \widetilde{\mathbf{u}}_H, \tag{5.11}$$

$$((A_h + C_h) + \widehat{E}_h)\widehat{\mathbf{u}}_h = \widehat{\lambda}_h B_h \widehat{\mathbf{u}}_h, \tag{5.12}$$

respectively, with $E_H\widetilde{\mathbf{u}} = \mathbf{r}_H$ and $\widehat{E}_h\widehat{\mathbf{u}} = \widehat{\mathbf{r}}_h$.
Using the relation between the coarse and the fine mesh, i.e.,

$$P^T(A_h + C_h)P = A_H + C_H, \quad P^T B_h P = B_H,$$

it follows that (5.11) is equivalent to

$$(P^T(A_h + C_H)P + E_H)\widetilde{\mathbf{u}}_H = \widetilde{\lambda}_H(P^T B_h P)\widetilde{\mathbf{u}}_H. \tag{5.13}$$

Multiplying (5.12) from the left by P^T gives

$$P^T(A_h + C_h)\widehat{\mathbf{u}}_h + P^T\widehat{E}_h\widehat{\mathbf{u}}_h = \widehat{\lambda}_h P^T B_h \widehat{\mathbf{u}}_h. \tag{5.14}$$

Using the fact that $P\widetilde{\mathbf{u}}_H = \widehat{\mathbf{u}}_h$, we can rewrite (5.13) as

$$P^T(A_h + C_h)P\widetilde{\mathbf{u}}_H + E_H\widetilde{\mathbf{u}}_H = \widetilde{\lambda}_H P^T B_h P\widetilde{\mathbf{u}}_H. \tag{5.15}$$

By subtracting (5.14) from (5.15) we get

$$(\widetilde{\lambda}_H - \widehat{\lambda}_h)(P^T B_h \widehat{\mathbf{u}}_h) = P^T(A_h + C_h)\widehat{\mathbf{u}}_h + E_H\widetilde{\mathbf{u}}_H - P^T(A_h + C_h)\widehat{\mathbf{u}}_h - P^T\widehat{E}_h\widehat{\mathbf{u}}_h.$$

Finally, applying the triangle inequality yields

$$\begin{aligned} |\widetilde{\lambda}_H - \widehat{\lambda}_h| &= \frac{\|E_H\widetilde{\mathbf{u}}_H - P^T\widehat{E}_h\widehat{\mathbf{u}}_h\|_{B_H}}{\|P^T B_h\widehat{\mathbf{u}}_h\|_{B_H}} \leq \frac{\|E_H\widetilde{\mathbf{u}}_H\|_{B_H} + \|P^T\widehat{E}_h\widehat{\mathbf{u}}_h\|_{B_H}}{\|P^T B_h\widehat{\mathbf{u}}_h\|_{B_H}} \\ &= \frac{\|\mathbf{r}_H\|_{B_H} + \|P^T\widehat{\mathbf{r}}_h\|_{B_H}}{\|P^T B_h\widehat{\mathbf{u}}_h\|_{B_H}}. \end{aligned}$$

□

Corollary 5.5. *Let $(\lambda_H, \mathbf{u}_H)$, $(\lambda_h, \mathbf{u}_h)$ be the exact eigenpair of (5.2), (5.3), respectively, and let $(\widehat{\lambda}_h, \widehat{\mathbf{u}}_h)$ be the eigenpair obtained by the prolongation of the computed pair $(\widetilde{\lambda}_H, \widetilde{\mathbf{u}}_H)$ on the fine space V_h. Furthermore, let residual vectors $\mathbf{r}_H$, $\widehat{\mathbf{r}}_h$ be defined as in (5.5), (5.7). Then*

$$\begin{aligned}
|\lambda_H - \widehat{\lambda}_h| &\leq \kappa(B_H^{\frac{1}{2}} U_H) \frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}} + \frac{\|\mathbf{r}_H\|_{B_H} + \|P^T \widehat{\mathbf{r}}_h\|_{B_H}}{\|P^T B_h \widehat{\mathbf{u}}_h\|_{B_H}}, \\
|\lambda_H - \lambda_h| &\leq \kappa(B_H^{\frac{1}{2}} U_H) \frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}} + \frac{\|\mathbf{r}_H\|_{B_H} + \|P^T \widehat{\mathbf{r}}_h\|_{B_H}}{\|P^T B_h \widehat{\mathbf{u}}_h\|_{B_H}} + \kappa(B_h^{\frac{1}{2}} U_h) \frac{\|\widehat{\mathbf{r}}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}}, \\
|\lambda_h - \widetilde{\lambda}_H| &\leq \kappa(B_h^{\frac{1}{2}} U_h) \frac{\|\widehat{\mathbf{r}}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}} + \frac{\|\mathbf{r}_H\|_{B_H} + \|P^T \widehat{\mathbf{r}}_h\|_{B_H}}{\|P^T B_h \widehat{\mathbf{u}}_h\|_{B_H}},
\end{aligned}$$

with right eigenvector matrices $U_H = [\mathbf{u}_H \ U_{H,2}]$ and $U_h = [\mathbf{u}_h \ U_{h,2}]$.

Proof. Using the triangle inequality we get

$$|\lambda_H - \widehat{\lambda}_h| = |\lambda_H - \widetilde{\lambda}_H + \widetilde{\lambda}_H - \widehat{\lambda}_h| \leq |\lambda_H - \widetilde{\lambda}_H| + |\widetilde{\lambda}_H - \widehat{\lambda}_h|.$$

From the bound (5.8) and Corollary 5.4 we get

$$|\lambda_H - \widehat{\lambda}_h| \leq \kappa(B_H^{\frac{1}{2}} U_H) \frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}} + \frac{\|\mathbf{r}_H\|_{B_H} + \|P^T \widehat{\mathbf{r}}_h\|_{B_H}}{\|P^T B_h \widehat{\mathbf{u}}_h\|_{B_H}}.$$

Applying again triangle inequalities

$$\begin{aligned}
|\lambda_H - \lambda_h| &= |\lambda_H - \widehat{\lambda}_h + \widehat{\lambda}_h - \lambda_h| \leq |\lambda_H - \widehat{\lambda}_h| + |\widehat{\lambda}_h - \lambda_h|, \\
|\lambda_h - \widetilde{\lambda}_H| &= |\lambda_h - \widehat{\lambda}_h + \widehat{\lambda}_h - \widetilde{\lambda}_H| \leq |\lambda_h - \widehat{\lambda}_h| + |\widehat{\lambda}_h - \widetilde{\lambda}_H|,
\end{aligned}$$

bound (5.10) and Corollary 5.4 completes the proof. □

Corollary 5.6. *Let $(\lambda_H, \mathbf{u}_H)$, $(\lambda_h, \mathbf{u}_h)$ be an exact eigenpair of (5.2), (5.3) and let $(\widetilde{\lambda}_H, \widetilde{\mathbf{u}}_H)$, $(\widetilde{\lambda}_h, \widetilde{\mathbf{u}}_h)$ be their approximations, respectively. Let furthermore $(\widehat{\lambda}_h, \widehat{\mathbf{u}}_h)$ be the eigenpair obtained by the prolongation of $\widetilde{\mathbf{u}}_H$ on the fine space V_h. Then residual vectors $\mathbf{r}_H$, $\mathbf{r}_h$, $\widehat{\mathbf{r}}_h$ defined as in (5.5), (5.6), (5.7) we have*

$$\begin{aligned}
|\widetilde{\lambda}_h - \widehat{\lambda}_h| &\leq \kappa(B_h^{\frac{1}{2}} U_h) \frac{\|\mathbf{r}_h\|_{B_h^{-1}}}{\|\widetilde{\mathbf{u}}_h\|_{B_h}} + \kappa(B_h^{\frac{1}{2}} U_h) \frac{\|\widehat{\mathbf{r}}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}}, \\
|\widetilde{\lambda}_H - \widetilde{\lambda}_h| &\leq \kappa(B_h^{\frac{1}{2}} U_h) \frac{\|\mathbf{r}_h\|_{B_h^{-1}}}{\|\widetilde{\mathbf{u}}_h\|_{B_h}} + \kappa(B_h^{\frac{1}{2}} U_h) \frac{\|\widehat{\mathbf{r}}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}} + \frac{\|\mathbf{r}_H\|_{B_H} + \|P^T \widehat{\mathbf{r}}_h\|_{B_H}}{\|P^T B_h \widehat{\mathbf{u}}_h\|_{B_H}}, \\
|\lambda_H - \widetilde{\lambda}_h| &\leq \kappa(B_H^{\frac{1}{2}} U_H) \frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}} + \kappa(B_h^{\frac{1}{2}} U_h) \frac{\|\mathbf{r}_h\|_{B_h^{-1}}}{\|\widetilde{\mathbf{u}}_h\|_{B_h}} + \kappa(B_h^{\frac{1}{2}} U_h) \frac{\|\widehat{\mathbf{r}}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}} \\
&\quad + \frac{\|\mathbf{r}_H\|_{B_H} + \|P^T \widehat{\mathbf{r}}_h\|_{B_H}}{\|P^T B_h \widehat{\mathbf{u}}_h\|_{B_H}},
\end{aligned}$$

with right eigenvector matrices $U_H = [\mathbf{u}_H \ U_{H,2}]$ and $U_h = [\mathbf{u}_h \ U_{h,2}]$.

Proof. The proof follows by combing the previous bounds and using triangle inequalities, i.e.,

$$|\widetilde{\lambda}_h - \widehat{\lambda}_h| \;\leq\; |\widetilde{\lambda}_h - \lambda_h| \;+\; |\lambda_h - \widehat{\lambda}_h|$$

which together with (5.9) and (5.10) gives the first bound. For the second bound we use the triangle inequality

$$|\widetilde{\lambda}_H - \widetilde{\lambda}_h| \;\leq\; |\widetilde{\lambda}_H - \widehat{\lambda}_h| \;+\; |\widehat{\lambda}_h - \lambda_h| \;+\; |\lambda_h - \widetilde{\lambda}_h|,$$

Corollary 5.4 and bounds (5.10), (5.9). The last inequality follows from

$$|\lambda_H - \widetilde{\lambda}_h| \;\leq\; |\lambda_H - \widetilde{\lambda}_H| \;+\; |\widetilde{\lambda}_H - \widetilde{\lambda}_h|$$

combined with previous bound and inequality (5.8).

□

Based on Assumption 5.1 and estimates of eigenvalue errors between the exact and approximate discrete eigenvalues on spaces V_H and V_h, we obtain the following bounds.

Corollary 5.7. *Let λ be an exact, simple eigenvalue of problem (3.6) and λ_H, λ_h the corresponding exact discrete eigenvalues on space V_H and V_h, respectively. Furthermore, let $\widehat{\mathbf{u}}_h$ and P be defined as in Proposition 5.6 and let Assumption 5.1 for eigenvalues hold with a positive constant $c < 1$. Then with residual vectors $\mathbf{r}_H$, $\widehat{\mathbf{r}}_h$ defined as in (5.5), (5.7) we have*

$$\begin{aligned}
|\lambda - \lambda_H| \;&\leq\; \frac{1}{1-c}\left(\kappa(B_H^{\frac{1}{2}}U_H)\frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}} + \frac{\|\mathbf{r}_H\|_{B_H} + \|P^T\widehat{\mathbf{r}}_h\|_{B_H}}{\|P^T B_h\widehat{\mathbf{u}}_h\|_{B_H}}\right.\\
&\quad\left. + \kappa(B_h^{\frac{1}{2}}U_h)\frac{\|\widehat{\mathbf{r}}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}}\right),\\
|\lambda - \lambda_h| \;&\leq\; \left(1 + \frac{1}{1-c}\right)\left(\kappa(B_H^{\frac{1}{2}}U_H)\frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}} + \frac{\|\mathbf{r}_H\|_{B_H} + \|P^T\widehat{\mathbf{r}}_h\|_{B_H}}{\|P^T B_h\widehat{\mathbf{u}}_h\|_{B_H}}\right.\\
&\quad\left. + \kappa(B_h^{\frac{1}{2}}U_h)\frac{\|\widehat{\mathbf{r}}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}}\right),
\end{aligned}$$

with right eigenvector matrices $U_H = [\mathbf{u}_H \; U_{H,2}]$ and $U_h = [\mathbf{u}_h \; U_{h,2}]$.

Proof. The proof follows directly from applying Corollaries 5.2 and 5.5. □

Using the previous estimates we obtain bounds for the errors between eigenvalues of the original PDE (3.6) and their discrete approximations.

Theorem 5.8. *Let λ be an exact eigenvalue of (3.6) and let $(\widetilde{\lambda}_H, \widetilde{\mathbf{u}}_H)$, $(\widetilde{\lambda}_h, \widetilde{\mathbf{u}}_h)$ be approximate eigenvalues and right eigenvectors of (5.2), (5.3), respectively. Furthermore, let $(\widehat{\lambda}_h, \widehat{\mathbf{u}}_h)$ be the prolongation of $\widetilde{\mathbf{u}}_H$ on the fine space V_h and let Assumption 5.1 for the eigenvalues hold with a positive constant $c < 1$. Then with residual vectors $\mathbf{r}_H$, $\mathbf{r}_h$, $\widehat{\mathbf{r}}_h$ defined as in (5.5), (5.6), (5.7) we have*

$$
\begin{aligned}
|\lambda - \widetilde{\lambda}_H| &\leq \frac{1}{1-c}\left(\kappa(B_H^{\frac{1}{2}}U_H)\frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}} + \frac{\|\mathbf{r}_H\|_{B_H} + \|P^T\widehat{\mathbf{r}}_h\|_{B_H}}{\|P^T B_h\widehat{\mathbf{u}}_h\|_{B_H}} + \kappa(B_h^{\frac{1}{2}}U_h)\frac{\|\widehat{\mathbf{r}}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}}\right) \\
&\quad + \kappa(B_H^{\frac{1}{2}}U_H)\frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}}, \\
|\lambda - \widetilde{\lambda}_h| &\leq \left(1+\frac{1}{1-c}\right)\left(\kappa(B_H^{\frac{1}{2}}U_H)\frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}} + \frac{\|\mathbf{r}_H\|_{B_H} + \|P^T\widehat{\mathbf{r}}_h\|_{B_H}}{\|P^T B_h\widehat{\mathbf{u}}_h\|_{B_H}}\kappa(B_h^{\frac{1}{2}}U_h)\frac{\|\widehat{\mathbf{r}}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}}\right) \\
&\quad + \kappa(B_h^{\frac{1}{2}}U_h)\frac{\|\mathbf{r}_h\|_{B_h^{-1}}}{\|\widetilde{\mathbf{u}}_h\|_{B_h}}, \\
|\lambda - \widehat{\lambda}_h| &\leq \frac{1}{1-c}\left(\kappa(B_H^{\frac{1}{2}}U_H)\frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}} + \frac{\|\mathbf{r}_H\|_{B_H} + \|P^T\widehat{\mathbf{r}}_h\|_{B_H}}{\|P^T B_h\widehat{\mathbf{u}}_h\|_{B_H}} + \kappa(B_h^{\frac{1}{2}}U_h)\frac{\|\widehat{\mathbf{r}}_h\|_{B_h^{-1}}}{\|\widehat{\mathbf{u}}_h\|_{B_h}}\right) \\
&\quad + \kappa(B_H^{\frac{1}{2}}U_H)\frac{\|\mathbf{r}_H\|_{B_H^{-1}}}{\|\widetilde{\mathbf{u}}_H\|_{B_H}} + \frac{\|\mathbf{r}_H\|_{B_H} + \|P^T\widehat{\mathbf{r}}_h\|_{B_H}}{\|P^T B_h\widehat{\mathbf{u}}_h\|_{B_H}},
\end{aligned}
$$

with right eigenvector matrices $U_H = [\mathbf{u}_H \; U_{H,2}]$ *and* $U_h = [\mathbf{u}_h \; U_{h,2}]$.

Proof. At first we use the triangle inequality

$$|\lambda - \widetilde{\lambda}_H| \;\leq\; |\lambda - \lambda_H| \;+\; |\lambda_H - \widetilde{\lambda}_H|$$

together with Corollary 5.7 and bound (5.8). For the second relation we exploit

$$|\lambda - \widetilde{\lambda}_h| \;\leq\; |\lambda - \lambda_h| \;+\; |\lambda_h - \widetilde{\lambda}_h|,$$

Corollary 5.7 and bound (5.9). The last bound follows from

$$|\lambda - \widehat{\lambda}_h| \;\leq\; |\lambda - \lambda_h| \;+\; |\lambda_h - \widehat{\lambda}_h|,$$

Corollary 5.7 and relation (5.8). □

Likewise, we obtain bounds for angles between the corresponding eigenfunctions.

Corollary 5.9. *Consider generalized eigenvalue problems (5.2), (5.3) with exact eigenpairs* $(\lambda_H, \mathbf{u}_H)$, $(\lambda_h, \mathbf{u}_h)$ *and corresponding approximations* $(\widetilde{\lambda}_H, \widetilde{\mathbf{u}}_H)$, $(\widetilde{\lambda}_h, \widetilde{\mathbf{u}}_h)$, *respectively. Let* $(\widehat{\lambda}_h, \widehat{\mathbf{u}}_h)$ *be the prolongation of* $(\widetilde{\lambda}_H, \widetilde{\mathbf{u}}_H)$ *on the fine space* V_h. *Furthermore, let residual vectors* $\mathbf{r}_H$, $\mathbf{r}_h$, $\widehat{\mathbf{r}}_h$ *be defined as in (5.5), (5.6), (5.7). Then*

$$\sin\angle_{\mathbf{B}_H}(\widetilde{\mathbf{u}}_H, \mathbf{u}_H) \;\leq\; \kappa(B_H^{\frac{1}{2}}U_{H,2}^{\star})\frac{\|\mathbf{r}_H\|_{\mathbf{B}_H^{-1}}}{\delta_H}, \tag{5.16}$$

$$\sin\angle_{\mathbf{B}_h}(\widetilde{\mathbf{u}}_h, \mathbf{u}_h) \;\leq\; \kappa(B_h^{\frac{1}{2}}U_{h,2}^{\star})\frac{\|\mathbf{r}_h\|_{\mathbf{B}_h^{-1}}}{\delta_h}, \tag{5.17}$$

$$\sin\angle_{\mathbf{B}_h}(\widehat{\mathbf{u}}_h, \mathbf{u}_h) \;\leq\; \kappa(B_h^{\frac{1}{2}}U_{h,2}^{\star})\frac{\|\widehat{\mathbf{r}}_h\|_{\mathbf{B}_h^{-1}}}{\widehat{\delta}_h}, \tag{5.18}$$

with left eigenvector matrices $U_H^{-1} = \begin{bmatrix} \mathbf{u}_H^\star B_H \\ U_{H,2}^\star B_H \end{bmatrix}$, $U_h^{-1} = \begin{bmatrix} \mathbf{u}_h^\star B_h \\ U_{h,2}^\star B_h \end{bmatrix}$ *and gaps*

$$\delta_H = \min_{\lambda_{i,H} \neq \lambda_H} |\lambda_{i,H} - \widetilde{\lambda}_H|, \quad \delta_h = \min_{\lambda_{i,h} \neq \lambda_h} |\lambda_{i,h} - \widetilde{\lambda}_h|, \quad \widehat{\delta}_h = \min_{\lambda_{i,h} \neq \lambda_h} |\lambda_{i,h} - \widehat{\lambda}_h|.$$

Proof. All bounds follow from applying Proposition 2.46. □

In order to derive error estimates for angles between eigenfunctions of interest we use here the theoretical framework from Section 4.2.3 and Assumption 5.1.

Theorem 5.10. *Let u be an exact eigenfunction of (3.6), $u_H(u_H^\star)$ and $u_h(u_h^\star)$ the corresponding exact, discrete primal (dual) eigenfunction on the space V_H, V_h and $\mathbf{u}_H(\mathbf{u}_H^\star)$, $\mathbf{u}_h(\mathbf{u}_h^\star)$ their representation vectors, respectively. Moreover, let Assumption 5.1 for the eigenfunctions hold with a positive constant $C < 1$. If $\widetilde{\mathbf{u}}_H(\widetilde{\mathbf{u}}_H^\star), \widehat{\mathbf{u}}_h(\widetilde{\mathbf{u}}_h^\star)$ are approximate right (left) eigenvectors of (5.2), (5.3), respectively, and $\mathbf{r}_H, \widehat{\mathbf{r}}_h$ residual vectors defined as in (5.5), (5.7), then*

$$\sin \angle(u, u_H) \leq \frac{1}{1-C} \Big(\kappa(B_H^{\frac{1}{2}} U_{H,2}^\star) \frac{\|\mathbf{r}_H\|_{\mathbf{B}_H^{-1}}}{\delta_H} + \kappa(B_h^{\frac{1}{2}} U_{h,2}^\star) \frac{\|\widehat{\mathbf{r}}_h\|_{\mathbf{B}_h^{-1}}}{\widehat{\delta}_h} \Big), \tag{5.19}$$

$$\sin \angle(u, u_h) \leq \frac{C}{1-C} \Big(\kappa(B_H^{\frac{1}{2}} U_{H,2}^\star) \frac{\|\mathbf{r}_H\|_{\mathbf{B}_H^{-1}}}{\delta_H} + \kappa(B_h^{\frac{1}{2}} U_{h,2}^\star) \frac{\|\widehat{\mathbf{r}}_h\|_{\mathbf{B}_h^{-1}}}{\widehat{\delta}_h} \Big), \tag{5.20}$$

with left eigenvector matrices $U_H^{-1} = \begin{bmatrix} \mathbf{u}_H^\star B_H \\ U_{H,2}^\star B_H \end{bmatrix}$, $U_h^{-1} = \begin{bmatrix} \mathbf{u}_h^\star B_h \\ U_{h,2}^\star B_h \end{bmatrix}$ *and gaps*

$$\delta_H = \min_{\lambda_{i,H} \neq \lambda_H} |\lambda_{i,H} - \widetilde{\lambda}_H|, \quad \delta_h = \min_{\lambda_{i,h} \neq \lambda_h} |\lambda_{i,h} - \widetilde{\lambda}_h|, \quad \widehat{\delta}_h = \min_{\lambda_{i,h} \neq \lambda_h} |\lambda_{i,h} - \widehat{\lambda}_h|.$$

Proof. From Corollaries 5.2 and 4.22 we have

$$\sin \angle(u, u_H) \leq \frac{1}{1-C} \sin \angle(u_H, u_h) = \frac{1}{1-C} \sin \angle(u_H^h, u_h) = \frac{1}{1-C} \sin \angle_{\mathbf{B}_h}(\mathbf{u}_H^h, \mathbf{u}_h)$$

By combining last inequality with Corollary 4.24 we obtain

$$\sin \angle(u, u_H) \leq \frac{1}{1-C} \Big(\sin \angle_{\mathbf{B}_H}(\widetilde{\mathbf{u}}_H, \mathbf{u}_H) + \sin \angle_{\mathbf{B}_h}(\widehat{\mathbf{u}}_h, \mathbf{u}_h) \Big).$$

Finally, by using the first bound from Corollary 5.9 we get

$$\sin \angle(u, u_H) \leq \frac{1}{1-C} \Big(\kappa(B_H^{\frac{1}{2}} U_{H,2}^\star) \frac{\|\mathbf{r}_H\|_{\mathbf{B}_H^{-1}}}{\delta_H} + \kappa(B_h^{\frac{1}{2}} U_{h,2}^\star) \frac{\|\widehat{\mathbf{r}}_h\|_{\mathbf{B}_h^{-1}}}{\widehat{\delta}_h} \Big).$$

The proof for the second inequality can be obtained in a similar fashion. □

Summarizing, we present bounds for angles between exact primal eigenfunctions and their finite element approximations.

Theorem 5.11. *Let u be an exact eigenfunction of (3.6), $u_H(u_H^\star)$ and $u_h(u_h^\star)$ the corresponding exact, discrete primal (dual) eigenfunctions on the space V_H, V_h and $\mathbf{u}_H(\mathbf{u}_H^\star)$, $\mathbf{u}_h(\mathbf{u}_h^\star)$ their representation vectors, respectively. Moreover, let Assumption 5.1 for the eigenfunctions hold with a positive constant $C < 1$. If $\widetilde{\mathbf{u}}_H(\widetilde{\mathbf{u}}_H^\star), \widehat{\mathbf{u}}_h(\widetilde{\mathbf{u}}_h^\star)$ are approximate right (left) eigenvectors of (5.2), (5.3), respectively, and $\mathbf{r}_H, \widehat{\mathbf{r}}_h$ residual vectors defined as in (5.5), (5.7), then the following inequalities hold*

$$\begin{aligned}
\sin\angle(u,\widetilde{u}_H) &\leq \kappa(B_H^{\frac{1}{2}}U_{H,2}^\star)\frac{2-C}{1-C}\frac{\|\mathbf{r}_H\|_{\mathbf{B}_H^{-1}}}{\delta_H} + \kappa(B_h^{\frac{1}{2}}U_{h,2}^\star)\frac{1}{1-C}\frac{\|\widehat{\mathbf{r}}_h\|_{\mathbf{B}_h^{-1}}}{\widehat{\delta}_h},\\
\sin\angle(u,\widehat{u}_h) &\leq \kappa(B_H^{\frac{1}{2}}U_{H,2}^\star)\frac{C}{1-C}\frac{\|\mathbf{r}_H\|_{\mathbf{B}_H^{-1}}}{\delta_H} + \kappa(B_h^{\frac{1}{2}}U_{h,2}^\star)\frac{1}{1-C}\frac{\|\widehat{\mathbf{r}}_h\|_{\mathbf{B}_h^{-1}}}{\widehat{\delta}_h},
\end{aligned}$$

with left eigenvector matrices $U_H^{-1} = \begin{bmatrix} \mathbf{u}_H^\star B_H \\ U_{H,2}^\star B_H \end{bmatrix}$, $U_h^{-1} = \begin{bmatrix} \mathbf{u}_h^\star B_h \\ U_{h,2}^\star B_h \end{bmatrix}$ *and gaps*

$$\delta_H = \min_{\lambda_{i,H}\neq\lambda_H} |\lambda_{i,H} - \widetilde{\lambda}_H|, \quad \delta_h = \min_{\lambda_{i,h}\neq\lambda_h} |\lambda_{i,h} - \widetilde{\lambda}_h|, \quad \widehat{\delta}_h = \min_{\lambda_{i,h}\neq\lambda_h} |\lambda_{i,h} - \widehat{\lambda}_h|.$$

Proof. The proof follows directly from the proof of Theorem 4.27. The final step is an application of Corollary 5.9. □

Of course, information like the conditioning of the left eigenvector matrix or the size of the gap in the spectrum is, in general, not available. However, for practical applications one can use approximations discussed in Sections 2.3.3.1 and 2.3.3.2.

5.2 An adaptive homotopy approach for non-self-adjoint eigenvalue problems

Since, of today, the adaptive solution of a general non-self-adjoint eigenvalue problem remains a real challenge, we devote this section to study the class of convection-diffusion eigenvalue problems and to emphasize the usefulness of the homotopy method in solving non-self-adjoint PDE eigenvalue problems. As a model problem we consider again the convection-diffusion eigenvalue problem (3.6).

We design an adaptive homotopy method which combines a continuation method with the mesh adaptivity and matrix eigenvalue solvers. The solution of a simple, well-studied eigenvalue problem is continuously transformed to obtain the solution of the problem of interest. In order to assure the accuracy and efficiency of the algorithm, we have to balance three different types of errors. The *discretization error* η that arises when the infinite dimensional variational problems is considered in a finite dimensional subspace [57, 67], the *homotopy error* ν that arises when the diffusion problem is slowly transferred to the convection-diffusion problem [23] and the *approximation error* μ that arises from the iterative matrix eigensolver.

The underlying concept for all algorithms introduced in this section is quite general, however, in order to make the analysis simple we do not consider phenomena like bifurcation points or path-jumping, see, e.g., [87]. Therefore, as a test case, we consider only the eigenvalue with the smallest real part, which is known to be simple and well-separated [54] for all $0 \leq t \leq 1$. Thus it will not bifurcate and will follow an analytic path.

5.2.1 Homotopy method for an operator eigenvalue problem

The homotopy concept introduced in Section 2.3.2 can be easily extended to the convection-diffusion operator eigenvalue problem (3.6). Starting from the well-studied spectrum of a simple operator, here $\mathcal{L}_0 u := -\Delta u$, we use a continuation method to obtain the eigenpairs for the convection-diffusion operator $\mathcal{L}_1 u := -\Delta u + \beta \cdot \nabla u$.
Throughout this section, the following homotopy equation

$$\mathcal{H}(t) = (1-t)\mathcal{L}_0 + t\mathcal{L}_1 \quad \text{for} \quad 0 \leq t \leq 1, \tag{5.21}$$

for the model problem (3.6) is considered. Since for $t = 0$ we have

$$\mathcal{H}(0) = \mathcal{L}_0,$$

the eigenpairs of $\mathcal{H}(0)$ are the eigenpairs of the Laplace eigenvalue problem. The continuation method uses a 'time'-stepping procedure with nodes $t_0 = 0 < t_1 < \ldots < t_N = 1$ to compute eigenvalues and eigenvectors of

$$-\Delta u + t_i \beta \cdot \nabla u = \lambda u \quad \text{in} \quad \Omega. \tag{5.22}$$

Finally, when the homotopy reaches its final value $t = 1$, eigenpairs of $\mathcal{H}(1) = \mathcal{L}_1$ are the eigenpairs of the desired problem,

$$-\Delta u + \beta \cdot \nabla u = \lambda u \quad \text{in} \quad \Omega.$$

For each step t_i the corresponding weak finite dimensional primal and dual problems

$$\begin{aligned} a(u_\ell, v_\ell) + t_i\, c(u_\ell, v_\ell) &= \lambda_\ell b(u_\ell, v_\ell) \quad \text{for all} \quad v_\ell \in V_\ell, \\ a(w_\ell, u_\ell^\star) + t_i\, c(w_\ell, u_\ell^\star) &= \overline{\lambda_\ell^\star} b(w_\ell, u_\ell^\star) \quad \text{for all} \quad w_\ell \in V_\ell, \end{aligned}$$

are equivalent to the generalized primal and dual matrix eigenvalue problems

$$(A_\ell + t_i\, C_\ell)\mathbf{u}_\ell = \lambda_\ell B_\ell \mathbf{u}_\ell, \tag{5.23}$$

$$\mathbf{u}_\ell^\star (A_\ell + t_i\, C_\ell) = \lambda_\ell^\star \mathbf{u}_\ell^\star B_\ell, \tag{5.24}$$

corresponding to the discrete homotopy equation

$$\mathcal{H}_\ell(t) = (1-t)A_\ell + t(A_\ell + C_\ell) = A_\ell + tC_\ell.$$

For the case considered here, of simple and well-separated eigenvalues that do not bifurcate during the homotopy process, it is known [71] that every eigenvalue $\lambda_\ell(t)$ of the generalized eigenvalue problem (5.23) and (5.24) is an analytic function in t. Hence by choosing appropriate homotopy step sizes, eigenvalues can be continued on an analytic path towards the eigenvalues of $(A_\ell + C_\ell, B_\ell)$, see [84, 87]. The evolution of an eigenpair as a function of t is called an *eigenpath* and is denoted by $(\lambda_\ell(t), \mathbf{u}_\ell(t))$ and $(\lambda_\ell^\star(t), \mathbf{u}_\ell^\star(t))$, respectively.

5.2.2 A posteriori error estimates

An efficient algorithm, which uses the homotopy process in combination with the mesh adaptivity and the inexact algebraic eigenvalue solver, requires deriving a combined a posteriori analysis for the homotopy, discretization and iteration error. At the beginning of the homotopy process we solve the self-adjoint problem (the generalized symmetric eigenvalue problem on the matrix level). Real eigenvalues obtained at the homotopy step $t = 0$ move to (potentially complex conjugate) eigenvalues of the original problem. In order to understand the influence of the homotopy on the behavior of eigenvalues of interest, we analyze the so-called *homotopy error* which in another context is called *modeling error* [23]. In particular, we are interested in bounds between the exact eigenvalue of the original problem and the eigenvalue obtained at the intermediate step $t \leq 1$, i.e., we want to estimate

$$|\lambda(1) - \lambda(t)| \lesssim \nu(t) \quad \text{for} \quad 0 \leq t \leq 1,$$

where by $\nu(t)$ we denote the homotopy error estimator as stated in the following Lemma, [33].

Lemma 5.12. *For the model problem (3.6), the difference between the exact eigenvalues $\lambda(t)$ of the homotopy $\mathcal{H}(t)$ in (5.21) and $\lambda(1)$ can be estimated via*

$$|\lambda(1) - \lambda(t)| \lesssim \nu(t) := (1-t)|\beta|_\infty \left(|||u(t)||| + |||u^\star(t)|||\right) \quad \textit{for} \quad 0 \leq t \leq 1. \tag{5.25}$$

The constant in the inequality tends to $1/(2b(u(1), u^\star(1))$ as $t \to 1$.

Proof. See [33]. □

Since our objective is to balance errors arising at each step of the adaptive homotopy algorithm and the exact solution of the problem is unknown, a combined a posteriori bound should be based only on the available information, i.e., on computed eigentriple approximations.
At first we recall a posteriori error bounds for the discretization error introduced in [57, 67]

$$|||u(t) - u_\ell(t)|||^2 + |||u^\star(t) - u_\ell^\star(t)|||^2 + |\lambda(t) - \lambda_\ell(t)| \lesssim \eta^2(\lambda_\ell(t), u_\ell(t), u_\ell^\star(t)),$$

where

$$\eta^2(\lambda_\ell(t), u_\ell(t), u_\ell^\star(t)) := \sum_{T \in \mathcal{T}_\ell} \left(\eta^2(\lambda_\ell(t), u_\ell(t); T) + \eta^{\star 2}(\lambda_\ell(t), u_\ell^\star(t); T)\right)$$

and the residual error estimators $\eta^2(\lambda_\ell(t), u_\ell(t); T)$, $\eta^{\star 2}(\lambda_\ell(t), u_\ell^\star(t); T)$ are defined as follows.

$$\eta^2(\lambda_\ell(t), u_\ell(t); T) := h_T^2 \|\beta \cdot \nabla u_\ell(t) - \lambda_\ell(t) u_\ell(t)\|_{L^2(T)}^2 + \sum_{E \in \mathcal{E}_\ell(T)} h_E \|[\nabla u_\ell(t)] \cdot n_E\|_{L^2(E)}^2,$$

$$\eta^{\star 2}(\lambda_\ell(t), u_\ell^\star(t); T) := h_T^2 \|-\beta \cdot \nabla \overline{u_\ell^\star(t)} - \lambda_\ell(t) \overline{u_\ell^\star(t)}\|_{L^2(T)}^2 + \sum_{E \in \mathcal{E}_\ell(T)} h_E \|[\nabla \overline{u_\ell^\star(t)}] \cdot n_E\|_{L^2(E)}^2.$$

Moreover, the following estimate holds for the algebraic iteration errors [65, 95].

$$\|\!|u_\ell(t) - \widetilde{u}_\ell(t)|\!\|^2 + \|\!|u^\star_\ell(t) - \widetilde{u}^\star_\ell(t)|\!\|^2 + |\lambda_\ell(t) - \widetilde{\lambda}_\ell(t)| \lesssim \mu^2(\widetilde{\lambda}_\ell(t), \widetilde{u}_\ell(t), \widetilde{u}^\star_\ell(t)),$$

where

$$\mu^2(\widetilde{\lambda}_\ell(t), \widetilde{u}_\ell(t), \widetilde{u}^\star_\ell(t)) := \left(\frac{\|\mathbf{r}_\ell\|_{B_\ell^{-1}}}{\|\mathbf{u}_\ell\|_{B_\ell}}\right)^2 + \left(\frac{\|\mathbf{r}^\star_\ell\|_{B_\ell^{-1}}}{\|\mathbf{u}^\star_\ell\|_{B_\ell}}\right)^2,$$

and the algebraic residuals are given by

$$\mathbf{r}_\ell := (A_\ell + C_\ell)\mathbf{u}_\ell - \lambda_\ell B_\ell \mathbf{u}_\ell, \quad \mathbf{r}^\star_\ell := \mathbf{u}^\star_\ell(A_\ell + C_\ell) - \lambda^\star_\ell \mathbf{u}^\star_\ell B_\ell.$$

Constants for the algebraic error estimators depend on the condition number of the considered eigenvalue and the gap in the spectrum, see Section 2.3. However, in our numerical examples the eigenvalue of interest is well-conditioned and well-separated from the remaining spectrum.
Using perturbation results of the a posteriori error estimator for the discretization error, introduced in [32, 57], a combined a posteriori error estimator can be obtained for the adaptive homotopy algorithm as stated in the following Lemma [33].

Lemma 5.13. *For the model problem (3.6), the difference between the iterative eigenvalue* $\widetilde{\lambda}_\ell(t)$ *in the homotopy* $\mathcal{H}_\ell(t)$ *and the continuous eigenvalue* $\lambda(1)$ *of the original problem (3.6) can be estimated a posteriori via*

$$|\lambda(1) - \widetilde{\lambda}_\ell(t)| \lesssim \nu(\widetilde{\lambda}_\ell(t), \widetilde{u}_\ell(t), \widetilde{u}^\star_\ell(t)) + \eta^2(\widetilde{\lambda}_\ell(t), \widetilde{u}_\ell(t), \widetilde{u}^\star_\ell(t)) + \mu^2(\widetilde{\lambda}_\ell(t), \widetilde{u}_\ell(t), \widetilde{u}^\star_\ell(t))$$

in terms of

$$\begin{aligned}\nu(\widetilde{\lambda}_\ell(t), \widetilde{u}_\ell(t), \widetilde{u}^\star_\ell(t)) \;:=\; & (1-t)|\beta|_\infty \left(\|\!|\widetilde{u}_\ell(t)|\!\| + \|\!|\widetilde{u}^\star_\ell(t)|\!\|\right) \\ & + (1-t)|\beta|_\infty \left(\eta(\widetilde{\lambda}_\ell(t), \widetilde{u}_\ell(t), \widetilde{u}^\star_\ell(t)) + \mu(\widetilde{\lambda}_\ell(t), \widetilde{u}_\ell(t), \widetilde{u}^\star_\ell(t))\right).\end{aligned}$$

Proof. A complete proof can be found in [33]. □

5.2.3 Algorithms

In the following we present three different adaptive algorithms for the homotopy driven eigenvalue problem. We combine the homotopy method with the adaptive finite element method and provide multi-way adaptivity by a suitable balancing of the discretization error, the homotopy error and errors in the iterative solutions of the generalized algebraic eigenvalue problems (5.23)–(5.24).

One of a crucial factors in presented algorithms is the step size control for the homotopy steps. It is well-known that the convergence and the accuracy of the homotopy method strongly depends, not only on the choice of a good initial value, but also on the appropriate selection of step sizes τ. A very small τ will assure a good approximation of the desired eigenvalue and eigenvector, but unfortunately will lead to large computational costs. On the

other hand if the step size τ is too large, then the method may be unable to keep track of the eigenvalues. Thus, the goal is to choose τ in such a way that it will assure the accuracy of the approximation, minimize the computational effort and keep track of the eigenpath. At this stage, employing adaptive step size control techniques that are well established in the numerical solution of ordinary differential equations [63], e.g., predictor-corrector procedures as they are commonly used [84], seems to be an obvious choice. However, combining the homotopy approach with the adaptive finite element method, requires a modification of the adaptive step size control techniques. By multi-way adaptivity we mean not only the adaptive choice of parameters in each of the component processes, i.e., homotopy, discretization and iteration, separately, but we also have to take into account existing interdependences. The homotopy process directly affects the mesh adaptivity that modifies the space discretization which again influences the quality of the iteration etc.. Therefore, the following step size control seems to be natural.

Consider the eigenvalue problem (5.22) for two different homotopy parameters t_i and $t_i + \tau$. If problems for t_i and $t_i + \tau$ do not differ to much, the quantities obtained at step t_i should be appropriate initial values for the problem at $t_i + \tau$. Thus, a final grid obtained for the solution at step t_i can be taken as a initial grid for the problem at $t_i + \tau$. Morover, one expects that obtaining a solution of the same accuracy as for the previous step will require only a small number of additional mesh refinements. If the number of required refinement steps for the homotopy parameters t_i and $t_i + \tau$ differs significantly, the step size τ was to large. The homotopy step for $t_i + \tau$ is then rejected and $t_i + q\tau$ is used, where $0 < q < 1$, e.g., $q = \frac{1}{2}$. If the number of refinements is small, then the step size τ is preserved or even increased by choosing, e.g., $\tau = q^{-1}\tau$. This simple idea allows to describe the dependence of the homotopy step size τ not only on the solution but also on the mesh adaptation process. In Algorithm 1, a fixed step size τ for the homotopy is considered in order to analyze the influence of the homotopy error on the mesh adaptation process and the accuracy of the solution. Algorithm 2 considers an adaptive step size control for the homotopy, based on the number of refinements required to balance the discretization error η_ℓ and the final tolerance ε. Algorithm 3 then, finally combines two concepts from Algorithms 1 and 2.

In all three algorithms, ρ will denote an intermediate accuracy for the matrix eigensolver, $0 < \omega < 1$ is the parameter in the relative accuracy condition for the algebraic approximation error, $0 < \delta < 1$ is the parameter balancing the discretization and the homotopy error estimator, $0 < \theta < 1$ is the marking parameter for the bulk marking strategy and γ denotes the maximal number of refinement steps allowed in each homotopy step of Algorithms 2 and 3. In Algorithm 1, τ is the fixed step size, while in the other two algorithms it denotes the initial step size. For simplicity we will write ν, η, ν (or ν_ℓ, η_ℓ, ν_ℓ for particular level ℓ) meaning $\nu(\widetilde{\lambda}_\ell(t), \widetilde{\mathbf{u}}_\ell(t), \widetilde{\mathbf{u}}^\star_\ell(t))$, $\eta(\widetilde{\lambda}_\ell(t), \widetilde{\mathbf{u}}_\ell(t), \widetilde{\mathbf{u}}^\star_\ell(t))$ and $\mu(\widetilde{\lambda}_\ell(t), \widetilde{\mathbf{u}}_\ell(t), \widetilde{\mathbf{u}}^\star_\ell(t))$, respectively.

In order to illustrate differences between three algorithms their main ideas are depicted in Figure 5.14.

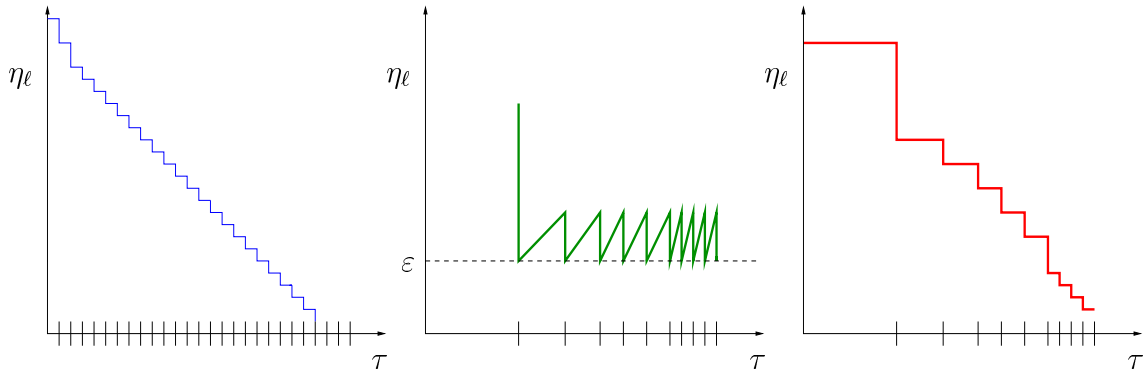


Figure 5.14: Schematic view of three homotopy-based algorithms.

In all three homotopy methods, the basic mesh adaptation method given by the procedures `Estimate & Solve`, `Mark` and `Refine` is used. In the `Estimate & Solve` function (see listing below), for the given mesh and parameters ρ and ω, the generalized algebraic eigenvalue problem (GAEVP) $((A_\ell + tC_\ell), B_\ell)$ has to be solved at each refinement step. The approximation of the eigentriple is considered to be accurate if the estimate for the complete algebraic approximation error μ, (both for the left and the right eigenvector), is smaller then the discretization error η, up to some fixed constant ω (see line 5). In fact, this step is a golden mean between the ideal, but impossible balancing based on a discrete H^{-1}-norm, see Section 4.4, and driving the approximation error to machine precision at each refinement step. To ensure that the algebraic approximation error itself is small, the intermediate tolerance parameter ρ for the iterative solver depends on the discretization error η and is also adapted (see lines 4–6). The algebraic eigenvalue problem is solved using the ARPACK [81] implementation of the implicitly restarted Arnoldi method for non-symmetric eigenvalue problems. Furthermore, the size of the constructed Krylov subspace is chosen to be as small as possible and the approximations of the right and the left eigenvector $\widetilde{\mathbf{u}}, \widetilde{\mathbf{u}}^\star$ from the previous iteration are taken as starting values for the new Arnoldi step. Note that here the final accuracy ε of the solution is not required at every step, only the relation between the discretization error and the algebraic approximation error is used to stop the procedure. In procedure `Mark` a bulk marking strategy is used to choose which triangles should be refined. To this purpose, we need some elementwise information about the error. The combined a posteriori error estimator introduced in Lemma 5.13 provides global estimate for the error and can not be used to obtain any local information. The only component which can be exploited locally is the discretization error estimator η. Due to the special balancing, we guarantee that for the accurate eigentriple $(\widetilde{\lambda}, \widetilde{\mathbf{u}}, \widetilde{\mathbf{u}}^\star)$ the approximation error estimator μ is of the same order as estimator η. Thus, the set of marked elements $\mathcal{M}$ can be identified based on the local contributions of the discretization error estimator η. Afterwards, the procedure `Refine` is realized to obtain a new mesh $\mathcal{T}$. Some additional information about procedures `Mark` and `Refine` can be found in Section 2.2.4.

Estimate & Solve

Input: $\mathcal{T}$, t, ρ, ω, $\widetilde{\mathbf{u}}, \widetilde{\mathbf{u}}^\star$

1: $[(A+C), B]$ = Create AEVP($\mathcal{T}, \beta, t$)
2: $[\mu, \widetilde{\mathbf{u}}, \widetilde{\mathbf{u}}^\star]$ = Solve AEVP($A+C$, B, ρ, $\widetilde{\mathbf{u}}$, $\widetilde{\mathbf{u}}^\star$)
3: Compute η
4: $\rho = 2\eta$
5: **while** $\mu > \omega\eta$ **do**
6: $\quad \rho = \frac{\rho}{2}$
7: $\quad [\mu, \widetilde{\mathbf{u}}, \widetilde{\mathbf{u}}^\star]$ = Solve AEVP($A+C$, B, ρ, $\widetilde{\mathbf{u}}$, $\widetilde{\mathbf{u}}^\star$)
8: $\quad$ Compute η
9: **end while**
10: Compute ν

Output: η, ν, μ, λ, $\widetilde{\mathbf{u}}, \widetilde{\mathbf{u}}^\star$

Algorithm 1

Algorithm 1

Input: $t_0 = 0, \tau, \mathcal{T}_0(t_0), \rho, \varepsilon, \omega, \delta$, $\widetilde{\mathbf{u}}_0(t_0)$, $\widetilde{\mathbf{u}}_0^\star(t_0)$

1: $\ell = 1,\ i = 0$
2: **while** $t_i \leq 1$ **do**
3: $\quad \mathcal{T}_0(t_i) = \mathcal{T}_\ell(t_{i-1})$
4: $\quad \rho(t_i) = \rho$
5: $\quad [\widetilde{\mathbf{u}}_0(t_i), \widetilde{\mathbf{u}}_0^\star(t_i)] = [\widetilde{\mathbf{u}}_\ell(t_{i-1}), \widetilde{\mathbf{u}}_\ell^\star(t_{i-1})]$
6: $\quad [\eta_0(t_i), \nu_0(t_i), \widetilde{\mathbf{u}}_0(t_i), \widetilde{\mathbf{u}}_0^\star(t_i)]$ = `Estimate & Solve`$(\mathcal{T}_0(t_i), \rho(t_i), \omega, \widetilde{\mathbf{u}}_0(t_i), \widetilde{\mathbf{u}}_0^\star(t_i))$
7: $\quad j = 0$
8: $\quad \rho(t_i) = \eta_\ell(t_i)$
9: $\quad$ **while** $\eta_j(t_i) > \max(\delta\nu_j(t_i), \varepsilon)$ **do**
10: $\quad\quad j = j + 1$
11: $\quad\quad \mathcal{M}_j(t_i)$ = `Mark`$(\eta_j(t_i), \theta)$
12: $\quad\quad \mathcal{T}_j(t_i)$ = `Refine`$(\mathcal{T}_{j-1}(t_i), \mathcal{M}_j(t_i))$
13: $\quad\quad [\widetilde{\mathbf{u}}_j(t_i), \widetilde{\mathbf{u}}_j^\star(t_i)] = [P_{j,j-1}\widetilde{\mathbf{u}}_{j-1}(t_i), P_{j,j-1}\widetilde{\mathbf{u}}_{j-1}^\star(t_i)]$
14: $\quad\quad [\eta_j(t_i), \nu_j(t_i), \widetilde{\mathbf{u}}_j(t_i), \widetilde{\mathbf{u}}_j^\star(t_i)]$ = `Estimate & Solve`$(\mathcal{T}_j(t_i), \rho(t_i), \omega, \widetilde{\mathbf{u}}_j(t_i), \widetilde{\mathbf{u}}_j^\star(t_i))$
15: $\quad$ **end while**
16: $\quad \ell = \ell + j$
17: $\quad t_{i+1} = t_i + \tau,\ i = i + 1$
18: **end while**

Output: $\widetilde{\lambda}(1), \widetilde{\mathbf{u}}(1), \widetilde{\mathbf{u}}^\star(1)$

The first algorithm introduces the homotopy method with a fixed step size τ. For the initial homotopy parameter $t_0 = 0$, the corresponding Laplace eigenvalue problem is solved on the initial mesh $\mathcal{T}_0(t_0)$ (see line 3), where the algebraic eigenvalue problem is solved up to

tolerance $\rho(t_0)$ (see line 4 of `Estimate & Solve` for details). This step is the same for all three algorithms. Based on the calculated initial approximation of the eigentriple at t_0 the corresponding discretization and homotopy error estimators $\eta_0(t_0)$, $\nu_0(t_0)$ are determined (see line 6).
In order to balance the discretization error, the homotopy error, and the final accuracy ε, the adaptive mesh refinement method is used (see lines 9–15). The mesh adaptation process is repeated as long as the discretization error dominates over the homotopy error multiplied by the balancing factor δ or is larger than the final accuracy ε (see line 9). Throughout the adaptive loop, sequences of meshes $\mathcal{T}_{\ell+j}(t_i)$, error estimators $\eta_j(t_i), \nu_j(t_i), \mu_j(t_i)$ and eigentriple approximations $(\widetilde{\lambda}_j(t_i),\widetilde{\mathbf{u}}_j(t_i),\ \widetilde{\mathbf{u}}_j^{\star}(t_i))$ are assembled. To avoid unnecessary computational work, at each step of the adaptive loop, the algebraic eigenvalue problem is solved only up to the accuracy $\rho(t_k)$, which depends on the discretization error $\eta_j(t_i)$ (see line 8 and the `Estimate & Solve` function for details). When the stopping condition (see line 9) does not hold, a new homotopy parameter $t_{i+1} = t_i + \tau$ is chosen and the new adaptation process starts with a previously obtained approximation taken as initial guess (see line 13). Here $P_{j,j-1}$ denotes the prolongation matrix from the last coarse mesh $\mathcal{T}_{j-1}(t_i)$ on the refined mesh $\mathcal{T}_j(t_i)$ (see lines 12–13). Note that the final mesh derived for the former homotopy parameter is taken as the initial mesh for new computations (see line 3). After a fixed number of homotopy steps, t_i reaches its final value one and the algorithm returns the approximated eigenvalues and eigenvectors of the model problem.
The final number of refinement levels reached up to the parameter t_i is denoted by ℓ, while j is a refinement index for the current parameter t_i. This distinction is made to separate a sequence of meshes for a single homotopy step from the final sequence obtained for the whole algorithm. It has particular importance for the next two algorithms.
Controlling the homotopy error is beneficial and an arbitrary fixed choice of the homotopy step size, in general, will not work, especially for more complicated problems. In the non-symmetric case eigenvalues move according to their condition numbers [101]. Ill-conditioned eigenvalues, as a function of t, may move very fast. The lack of an analogue of the minimax principle [61] for non-symmetric problems makes the localization of the eigenvalue very hard. In particular, it may be difficult to guarantee fast convergence of the iterative eigensolver to the eigenvalue with smallest real part at next homotopy step $t_i + \tau$. If the step size τ is chosen too large, even a very accurate approximation from step t_i may simply be not good starting value for the next homotopy step. On the other hand choosing τ very small leads to a large number of homotopy steps, and since for each step the whole adaptive mesh refinement loop has to be performed, this may lead to large computational effort.

Algorithm 2

In contrast to Algorithm 1, in Algorithm 2 an adaptive step size control for the homotopy is used. Starting with an initial step size τ the first approximation is computed to assure that the discretization error $\eta_j(t_i)$ is smaller than the fixed, desired accuracy ε (see line 9). No dependence on the homotopy error is considered here. Additionally, for each homotopy parameter only a fixed number of mesh refinement steps γ inside the adaptive loop is allowed

Algorithm 2

Input: $t_0 = 0, \tau, \beta, \mathcal{T}_0(t_0), \rho, \varepsilon, \omega,\ \gamma,\ \widetilde{\mathbf{u}}_0(t_0),\ \widetilde{\mathbf{u}}_0^\star(t_0)$

$\ell = 1,\ i = 0$
while $t_i < 1$ **do**
  $\mathcal{T}_0(t_i) = \mathcal{T}_{\ell-1}(t_{i-1})$
  $\rho(t_i) = \rho$
  $[\widetilde{\mathbf{u}}_0(t_i), \widetilde{\mathbf{u}}_0^\star(t_i)] = [\widetilde{\mathbf{u}}_{\ell-1}(t_{i-1}), \widetilde{\mathbf{u}}_{\ell-1}^\star(t_{i-1})]$
  $[\eta_0(t_i), \nu_0(t_i), \widetilde{\mathbf{u}}_0(t_i), \widetilde{\mathbf{u}}_0^\star(t_i)] =$ `Estimate & Solve`$(\mathcal{T}_0(t_i), \rho(t_i), \omega, \widetilde{\mathbf{u}}_0(t_i), \widetilde{\mathbf{u}}_0^\star(t_i))$
  $\rho(t_i) = \eta_\ell(t_i)$
  $j = 0$
  while $\eta_j(t_i) > \varepsilon$ **do**
    if $j > \gamma$ **then**
      $i = i - 1$
      $\tau = q\tau$
      $j = 0$
      break
    end if
    $j = j + 1$
    $\mathcal{M}_j(t_i) =$ `Mark`$(\eta_j(t_i), \theta)$
    $\mathcal{T}_j(t_i) =$ `Refine`$(\mathcal{T}_{j-1}(t_i), \mathcal{M}_j(t_i))$
    $[\widetilde{\mathbf{u}}_j(t_i), \widetilde{\mathbf{u}}_j^\star(t_i)] = [P_{j,j-1}\widetilde{\mathbf{u}}_{j-1}(t_i), P_{j,j-1}\widetilde{\mathbf{u}}_{j-1}^\star(t_i)]$
    $[\eta_j(t_i), \nu_j(t_i), \widetilde{\mathbf{u}}_j(t_i), \widetilde{\mathbf{u}}_j^\star(t_i)] =$ `Estimate & Solve`$(\mathcal{T}_j(t_i), \rho(t_i), \omega, \widetilde{\mathbf{u}}_j(t_i), \widetilde{\mathbf{u}}_j^\star(t_i))$
  end while
  $\ell = \ell + j$
  if $j < \gamma$ **then**
    $\tau = q^{-1}\tau$
  end if
  $t_{i+1} = \min(t_i + \tau, 1),\ i = i + 1$
end while

Output: $\widetilde{\lambda}_\ell(1), \widetilde{\mathbf{u}}_\ell(1), \widetilde{\mathbf{u}}^\star(1)$

(see line 10). If the adaptive loop needs more refinement steps than γ (see line 10), it means that eigenvalue problems considered for parameters t_i and $t_i + \tau$ differ too much and that the step size τ should be decreased. In that case, to ensure good approximations in the eigenvalue continuation, the algorithm rejects the current homotopy step (see lines 11–13), sets up a new $\tau = q\tau$ (see line 12), for some $0 < q < 1$, and starts the adaptive loop for the new homotopy parameter $t_i + \tau$. If the number of refinements is smaller than γ, then the algorithm attempts to increase the step size to $q^{-1}\tau$ (see line 24). Otherwise, τ is preserved at the next homotopy step. At this point, the previously introduced distinction between the global and the local refinement indices ℓ and j is used to carry out the rejection step, while keeping the right mesh hierarchy. Meshes obtained for the rejected homotopy parameter will not be considered in the final sequence of meshes denoted by ℓ.
Note, that here the initial mesh for the new homotopy parameter is taken as the last but one mesh obtained at the previous homotopy step (see line 3). If step sizes were chosen optimally and the consecutive problems do not differ too much, then the previous mesh should be a good starting mesh for the next step. In this way the continuation of meshes is also guaranteed. At the beginning it is reasonable to allow τ to be large and let the algorithm to adapt the step size by itself. However, it is obvious that if the total error is dominated by the homotopy error $\nu_\ell(t_i)$, then driving the discretization error $\eta_\ell(t_i)$ in each homotopy step below ε may lead to large computational effort.

Algorithm 3

The third algorithm combines both ideas of controlling the homotopy error and using the adaptive step size control. In this way the homotopy method accepts only approximations which are of a desired accuracy and whose computational cost is reasonable. Simultaneously, the adaptation in space, homotopy and in the iterative solver is applied. During the mesh adaptation the discretization error $\eta_j(t_i)$ is adapted to be smaller than the homotopy error $\nu_j(t_i)$. Also at each iteration step of the algebraic eigensolver, the approximation error $\mu_j(t_i)$ is adjusted, to avoid computing a solution that is too accurate in comparison to the discretization error $\eta_j(t_i)$. The adaptation of the homotopy parameter t is based on the maximal number of refinement levels γ. Currently, however, there is no theoretical justification for the optimal choice of γ that will lead to the minimal number of mesh refinement steps.

In summary, for Algorithm 1 fixed step sizes in t are considered together with the adaptivity in the mesh size assuring that the complete discretization error η is below the homotopy error ν for each homotopy parameter t. In Algorithm 2, the adaptivity in both the homotopy step size τ and the mesh size is achieved. Here, however, unlike in Algorithm 1 the discretization error is driven below the fixed tolerance ε, which is the same at each homotopy step and adaptation level. Algorithm 3 combines techniques of Algorithms 1 and 2. The homotopy error ν drives the mesh adaptivity and homotopy step sizes are adapted with respect to the parameter γ. The best choice of the maximal number of refinement steps is still an open question.
Future work will have to include the combination of presented concepts with methods that deal with multiple eigenvalues, bifurcation points, ill-conditioning or jumps in the eigenpaths.

Algorithm 3

Input: $t_0 = 0, \tau, q, \mathcal{T}_0(t_0), \varepsilon, \omega, \delta,\ \gamma,\ \widetilde{\mathbf{u}}_0(t_0),\ \widetilde{\mathbf{u}}_0^\star(t_0)$

$\ell = 1, i = 0$
while $t_i \leq 1$ & $t_{i-1} < 1$ **do**
  $\mathcal{T}_0(t_i) = \mathcal{T}_{\ell-1}(t_{i-1})$
  $\rho(t_i) = \rho$
  $[\widetilde{\mathbf{u}}_0(t_i), \widetilde{\mathbf{u}}_0^\star(t_i)] = [\widetilde{\mathbf{u}}_{\ell-1}(t_{i-1}), \widetilde{\mathbf{u}}_{\ell-1}^\star(t_{i-1})]$
  $[\eta_0(t_i), \nu_0(t_i), \widetilde{\mathbf{u}}_0(t_i), \widetilde{\mathbf{u}}_0^\star(t_i)] =$ `Estimate & Solve`$(\mathcal{T}_0(t_i), \rho(t_i), \omega, \widetilde{\mathbf{u}}_0(t_i), \widetilde{\mathbf{u}}_0^\star(t_i))$
  $\rho(t_i) = \eta_\ell(t_i)$
  $j = 0$
  while $\eta_j(t_i) > \max(\delta\nu_j(t_i), \varepsilon)$ **do**
    if $j > \gamma$ **then**
      $i = i - 1$
      $\tau = q\tau$
      $j = 0$
      break
    end if
    $j = j + 1$
    $\mathcal{M}_j(t_i) =$ `Mark`$(\eta_j(t_i), \theta)$
    $\mathcal{T}_j(t_i) =$ `Refine`$(\mathcal{T}_{j-1}(t_i), \mathcal{M}_j(t_i))$
    $[\widetilde{\mathbf{u}}_j(t_i), \widetilde{\mathbf{u}}_j^\star(t_i)] = [P_{j,j-1}\widetilde{\mathbf{u}}_{j-1}(t_i), P_{j,j-1}\widetilde{\mathbf{u}}_{j-1}^\star(t_i)]$
    $[\eta_j(t_i), \nu_j(t_i), \widetilde{\mathbf{u}}_j(t_i), \widetilde{\mathbf{u}}_j^\star(t_i)] =$ `Estimate & Solve`$(\mathcal{T}_j(t_i), \rho(t_i), \omega, \widetilde{\mathbf{u}}_j(t_i), \widetilde{\mathbf{u}}_j^\star(t_i))$
  end while
  $\ell = \ell + j$
  if $j < \gamma$ **then**
    $\tau = q^{-1}\tau$
  end if
  $t_{i+1} = \min(t_i + \tau, 1), i = i + 1$
end while

Output: $\widetilde{\lambda}_\ell(1), \widetilde{\mathbf{u}}_\ell(1), \widetilde{\mathbf{u}}^\star(1)$

5.2.4 Numerical experiments

This section presents some numerical results obtained with three adaptive homotopy Algorithms 1–3 introduced in Section 5.2.3. As a model problem we consider

$$-\Delta u + \beta \cdot \nabla u = \lambda u \quad \text{in} \quad \Omega \qquad \text{and} \qquad u = 0 \quad \text{on} \quad \partial\Omega,$$

with Ω being either the unit square or the L-shape domain. In order to calculate the eigenvalue errors we use the reference values obtained by Aitken extrapolation on uniform meshes [3].
Common to all experiments is that for ARPACK [81] the number p of Arnoldi vectors equals three and the maximal number `MXITER` of Arnoldi restarts is set to one [81]. All experiments were realized using MATLAB [90].
The homotopy starts with the simple symmetric eigenvalue problem with known smallest eigenvalue $\lambda(t_0) = 2\pi^2$ for the unit square and known approximation $\lambda(t_0) \approx 9.6397238440219$ [108] for the L-shape domain and then uses the homotopy to bring in the convection part. All experiments determine the eigenvalue with the smallest real part, since it is known to be simple and well-separated for any value of the convection parameter β [54], thus there are no bifurcation points and algorithms are following analytic eigenpaths.
To recall the motivation of the homotopy method, it is important to note that for general non-self-adjoint problems, there is no guarantee to achieve convergence to the eigenvalue of interest if standard methods are used. Experiments show that with a small number of Arnoldi vectors (i.e., a low dimensional Krylov subspace) and a random starting vector ARPACK does not find any good approximation to the eigenvalue for $t = 1$ even for very fine meshes. Thus, the stable adaptive mesh refinement is not possible with a low cost variation of the Arnoldi method as shown for self-adjoint problems in [91]. On the other hand the numerical experiments show that starting from the symmetric problem and following the eigenpath lead to sufficiently accurate approximations of the eigenpairs of the original non-self-adjoint problem. This shows that we can view our algorithms as a way of providing a starting vector for the non-self-adjoint problem which is sufficiently close to the eigenvector of interest. Therefore, most of the computational work is expected to occur in the last homotopy step $t = 1$ which is confirmed by the numerical experiments.

Example 1

For this example let Ω be the unit square $\Omega = [0, 1] \times [0, 1]$. We choose the convection parameter $\beta = [20, 0]^T$, the starting point of the homotopy $t_0 = 0$, the marking parameter $\theta = 0.3$, the balancing parameter of the discretization and the approximation error estimator $\omega = 0.1$, the step size update parameter $q = 1/2$, the number of refinement steps $\gamma = 2$, the overall accuracy $\varepsilon = 10^{-1}$, the initial tolerance for the iterative solver $\rho = 1$ and the balancing parameter of the homotopy and the discretization error estimator $\delta = 0.1$. A reference value for the eigenvalue with the smallest real part is given by

$$\lambda \approx 119.7392.$$

In general one can observe that all three algorithms lead to a finite sequence of homotopy steps and to the approximation of the eigenvalue of interest at the last step $t = 1$. Notice that for all algorithms, most of the computational work is done at the last step and therefore for the final problem. This can be seen in Tables 5.8, 5.10 and 5.12 when comparing the CPU time at the last step with the previous one. Note that here we only present the data for the best approximation of each homotopy step and not these for the intermediate approximations.
In Algorithm 1 the fixed homotopy step size $\tau = 0.1$ is chosen. Tables 5.7 and 5.8 for Algorithm 1 show that a small homotopy step size leads to a sequence where the second last homotopy step $t = 0.9$ does involve a small discrete problem, i.e., #DOF = 10370. Therefore, most of the refinement is done only in the last homotopy step $t = 1$, when the final accuracy is reached. The computational overhead introduced by the homotopy is minor for the right choice of homotopy step size τ. Since the best choice for τ is not known, it is necessary, and in practice reasonable, to introduce some extra computational overhead by using adaptive step size control. One may notice that the value obtained in the second last homotopy step has a large relative error and only the final approximation is good. As displayed in Figure 5.15 this effect leads to a nonlinear convergence rate and results in larger eigenvalue errors for $t_i < 1$ and accurate values only for $t = 1$.

t	$\eta_\ell(t)$	$\nu_\ell(t)$	$\mu_\ell(t)$	error estimator
0.0	18.7972	267.9989	0.0025677	286.7986
0.1	21.9037	250.3131	0.0003188	272.2171
0.2	17.6390	224.2302	0.0042579	241.8735
0.3	14.7243	204.8199	0.0066615	219.5508
0.4	12.0933	185.7716	0.0054502	197.8704
0.5	10.1746	167.8197	0.0560768	178.0503
0.6	7.8788	142.9867	0.0189887	150.8845
0.7	11.0907	121.0055	0.0577501	132.1540
0.8	8.4339	85.4466	0.0206147	93.9012
0.9	3.4934	44.0072	0.0025632	47.5031
1.0	0.0854	0.0000	0.0008344	0.0862

Table 5.7: The discretization $\eta_\ell(t)$, the homotopy $\nu_\ell(t)$, and the iteration $\mu_\ell(t)$ error estimator for all homotopy steps t in Algorithm 1 for Example 1.

t	$\widetilde{\lambda}_\ell(t)$	$\frac{\lvert\lambda_\ell(1)-\widetilde{\lambda}_\ell(t)\rvert}{\lvert\lambda_\ell(1)\rvert}$	#DOF	CPU time
0.0	20.31171	0.83037	65	0.04
0.1	21.19837	0.82296	65	0.05
0.2	23.76193	0.80155	114	0.09
0.3	28.68327	0.76045	222	0.13
0.4	35.57882	0.70286	436	0.17
0.5	44.58901	0.62762	838	0.24
0.6	55.71845	0.53467	1607	0.35
0.7	68.87482	0.42479	1607	0.41
0.8	83.83805	0.29983	3075	0.66
0.9	100.83461	0.15788	10370	1.86
1.0	119.74434	0.00004	587509	127.34

Table 5.8: The eigenvalue approximation $\widetilde{\lambda}_\ell(t)$, the relative eigenvalue error $\frac{\lvert\lambda_\ell(1)-\widetilde{\lambda}_\ell(t)\rvert}{\lvert\lambda_\ell(1)\rvert}$, the number of degrees of freedom (#DOF), and the CPU time for all homotopy steps t in Algorithm 1 applied to Example 1.

Algorithm 2 introduces the adaptive homotopy step size control. As initial step size $\tau = 1$ is chosen. Tables 5.9 and 5.10 show that the first homotopy step is rejected and a smaller step size τ is taken. In this example Algorithm 2 chooses less homotopy steps than the other two algorithms. Due to the fixed control of the discretization error by ε, the number of degrees of freedom is already high for the simple symmetric problem. This means that for $t_i < 1$ the error with respect to the DOFs is much larger than for the other algorithms as displayed in Figure 5.15. On the other hand for the last step $t = 1$ the result is very accurate.

t	$\eta_\ell(t)$	$\nu_\ell(t)$	$\mu_\ell(t)$	error estimator
0.00	0.0725	183.1140	0.0000000	183.1865
0.25	0.0649	156.7655	0.0000002	156.8303
0.50	0.0740	136.5043	0.0000012	136.5783
0.75	0.0640	88.4754	0.0000598	88.5395
1.00	0.0783	0.0000	0.0004680	0.0788

Table 5.9: The discretization $\eta_\ell(t)$, the homotopy $\nu_\ell(t)$, and the iteration $\mu_\ell(t)$ error estimator for all homotopy steps t in Algorithm 2 applied to Example 1.

To overcome the drawback of a fixed step size in Algorithm 1 and a fixed discretization error control in Algorithm 2, both techniques are combined in Algorithm 3. In Tables 5.11 and 5.12 we observe that the homotopy step size is decreased very much towards the end of the homotopy process. This effect is due to the fact, that the algorithm increases the number of DOF strongly only for t close to 1. This observation can be interpreted as that the algorithm computes a sufficiently accurate initial approximation to an eigenvector for $t = 1$. Note that most of the computational costs arise during the last three homotopy steps. Figure 5.15 shows that Algorithm 3 is a combination of the other two algorithms. The error

t	$\widetilde{\lambda}_\ell(t)$	$\frac{\lvert\lambda_\ell(1)-\widetilde{\lambda}_\ell(t)\rvert}{\lvert\lambda_\ell(1)\rvert}$	#DOF	CPU time
0.00	19.74139	0.83513	18420	2.62
0.25	25.98903	0.78295	48506	20.51
0.50	44.73837	0.62637	124817	40.28
0.75	75.98888	0.36538	366519	112.36
1.00	119.74216	0.00002	641569	278.09

Table 5.10: The eigenvalue approximation $\widetilde{\lambda}_\ell(t)$, the relative eigenvalue error $\frac{\lvert\lambda_\ell(1)-\widetilde{\lambda}_\ell(t)\rvert}{\lvert\lambda_\ell(1)\rvert}$, the number of degrees of freedom (#DOF), and the CPU time for all homotopy steps t in Algorithm 2 applied to Example 1.

t	$\eta_\ell(t)$	$\nu_\ell(t)$	$\mu_\ell(t)$	error estimator
0.0000	18.7972	267.9987	0.0025668	286.7984
0.2500	21.9560	224.1103	0.0070254	246.0733
0.5000	12.7398	173.0761	0.1539409	185.9698
0.7500	6.2305	99.7848	0.0008341	106.0161
0.8750	5.1172	54.7893	0.0003906	59.9069
0.9375	1.8715	27.6650	0.0001211	29.5367
0.9688	1.1430	14.0956	0.0271601	15.2658
0.9844	0.6630	7.0425	0.0141278	7.7196
0.9922	0.2189	3.4744	0.0006248	3.6940
1.0000	0.0745	0.0000	0.0020618	0.0765

Table 5.11: The discretization $\eta_\ell(t)$, the homotopy $\nu_\ell(t)$, and the iteration $\mu_\ell(t)$ error estimator for all homotopy steps t concerning Algorithm 3 applied to Example 1.

for approximations obtained at homotopy steps $t_i < 1$ is much smaller than for Algorithm 2 but similar to that of Algorithm 1. In contrast to Algorithm 1 homotopy step sizes are adapted, fewer homotopy steps are needed and they are more concentrated towards $t = 1$. In Figure 5.16 all three algorithms are compared with respect to the computational time. Obviously, Algorithm 2 and 3 need more time than Algorithm 1, since they reject some steps during their automatic step size control. For more complicated problems, going beyond this simple model example, it is expected that the adaptive step size control will lead to the faster computation than the method with a fixed step size. The homotopy procedure in Algorithm 1 introduces only a little computational overhead, with the possible drawback of a small (unknown) fixed step size while Algorithm 2 does adapt the step size automatically, but for the cost of larger computational overhead. In fact Table 5.10 shows that the overhead is less than $1/2$ of the overall CPU time, which is worthwhile. On the other hand Algorithm 3 needs even more computational time but combines the two advantages of Algorithm 1 and 2. The increase of the CPU time is due to the fact that Algorithm 3 rejects many steps during the homotopy process. Nevertheless, this moderate increase of the computational cost seems to be reasonable for more difficult situations, where without path following techniques no

t	$\widetilde{\lambda}_\ell(t)$	$\frac{\lvert\lambda_\ell(1)-\widetilde{\lambda}_\ell(t)\rvert}{\lvert\lambda_\ell(1)\rvert}$	#DOF	CPU time
0.0000	20.31171	0.83037	65	0.04
0.2500	25.86284	0.78401	112	0.25
0.5000	44.52525	0.62815	661	0.45
0.7500	75.97150	0.36553	3613	0.88
0.8750	96.37374	0.19514	6538	5.20
0.9375	107.66847	0.10081	21936	22.60
0.9688	113.63394	0.05099	40027	53.26
0.9844	116.67842	0.02556	71610	194.81
0.9922	118.19399	0.01290	226196	358.30
1.0000	119.76367	0.00020	685571	587.75

Table 5.12: The eigenvalue approximation $\widetilde{\lambda}_\ell(t)$, the relative error $\frac{\lvert\lambda_\ell(1)-\widetilde{\lambda}_\ell(t)\rvert}{\lvert\lambda_\ell(1)\rvert}$, the number of degrees of freedom (#DOF), and the CPU time for all homotopy steps t in Algorithm 3 for Example 1.

convergence to the desired eigenvalues can be guaranteed.
The final approximate primal and dual eigenfunctions for Algorithms 1, 2, and 3, together with the corresponding meshes, are depicted in Figures 5.17, 5.18 and 5.19. The final meshes for all problems look quite similar. Notice that, due to the adaptive refinement procedure for triangles, the symmetry of the mesh cannot be strictly preserved. For the square domain, primal and dual solutions of the problem have almost independent supports living on the opposite boundaries of the domain due to the convection in x direction. Therefore, all final meshes look quite symmetric. This observation shows that, in general, it is necessary to adapt the mesh for both the primal and dual eigenfunctions. Note that these meshes are more refined towards the strong boundary layers of both the primal and the dual solution.

Example 2

As in the first example, let Ω be the unit square $\Omega = [0,1] \times [0,1]$ and the convection parameter $\beta = [20,0]^T$. We choose the starting point of the homotopy $t_0 = 0$, the marking parameter $\theta = 0.3$, the balancing parameter of the discretization and the approximation error estimator $\omega = 0.1$, the step size update parameter $q = 1/3$, the number of refinement steps $\gamma = 2$, the overall accuracy $\varepsilon = 10^{-1}$, the initial tolerance for the iterative solver $\rho = 1$ and the balancing parameter of the homotopy and the discretization error estimator $\delta = 0.1$. Note that the only difference to Example 1 is the choice of the homotopy update parameter q. Here we demonstrate how a different choice of q influences the homotopy process for Algorithms 2 and 3. The results are presented in Tables 5.13– 5.16. Figures 5.20 and 5.21 compare the results obtained for Examples 1 and 2. In general we do not observe significant differences, which confirms that presented algorithms seem to be rather robust with respect to the adaptivity of the homotopy. Comparing the results with those of Example 1 shows that the choice $q = 1/3$ leads to similar relative eigenvalue errors for $t_i < 1$ but smaller relative eigenvalue error at the end. It is remarkable that Table 5.14 indicates that Algorithm 2

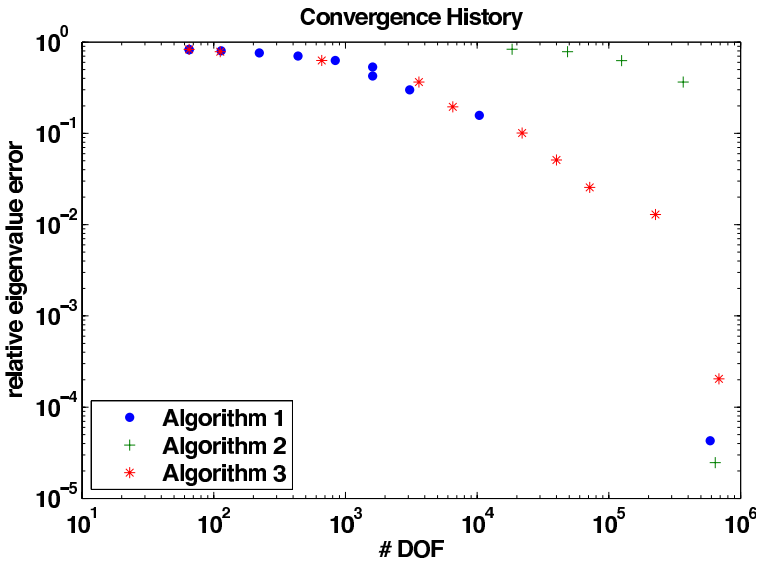


Figure 5.15: Convergence history of Algorithms 1, 2, and 3 with respect to #DOF for Example 1.

generates a sequence with almost fixed homotopy step size $\tau = 0.1$. For Algorithm 2 the choice of $q = 1/3$ leads to 10 homotopy steps compared to 5 steps in Example 1. Although this is an increase by a factor of two, the overall computational costs increase only slightly. This can be explained by the fact that at each homotopy step there are fewer refinements and overall fewer rejections of homotopy steps than in Example 1. For Algorithm 3 the choice of $q = 1/3$ leads to one additional homotopy step but computational costs moderately decrease. All these examples show that a proper choice of the parameter q is important for the overall performance of the algorithms.

Example 3

For this example let Ω be the L-shape domain $\Omega = [-1,1]\times[0,1]\cup[-1,0]\times[-1,0]$. We choose the convection parameter $\beta = [10,0]^T$, the starting point of the homotopy $t_0 = 0$, the marking parameter $\theta = 0.3$, the balancing parameter of the discretization and the approximation error estimator $\omega = 0.1$, the step size update parameter $q = 1/2$, the number of refinement steps $\gamma = 2$, the overall accuracy $\varepsilon = 10^{-1}$, the initial tolerance for the iterative solver $\rho = 1$ and the balancing parameter of the homotopy and the discretization error estimator $\delta = 0.1$. A reference value for the eigenvalue with the smallest real part is given by

$$\lambda \approx 34.6397.$$

Again for Algorithm 1 a fixed step size $\tau = 0.1$ is chosen. The results look similar to those of Examples 1 and 2. The eigenvalue errors for homotopy steps $t_i < 1$ are rather large and only the values for $t = 1$ are accurate. Table 5.18 shows that most of the CPU time is used on the last level.
Algorithm 2 starts with a step size $\tau = 1$ which is reduced by the adaptive procedure to $\tau = 0.25$ and afterwards not changed any more. Therefore, Algorithm 2 needs in total only

t	$\eta_\ell(t)$	$\nu_\ell(t)$	$\mu_\ell(t)$	error estimator
0.0000	0.0725	183.1140	0.0000000	183.1865
0.1111	0.0912	168.2191	0.0000003	168.3103
0.2222	0.0942	159.3306	0.0000002	159.4248
0.3333	0.0989	152.3163	0.0000008	152.4151
0.4444	0.0960	143.1067	0.0000018	143.2027
0.5556	0.0911	129.1835	0.0000177	129.2746
0.6667	0.0801	108.7764	0.0001128	108.8567
0.7778	0.0683	80.8181	0.0000674	80.8864
0.8889	0.0957	45.0185	0.0054764	45.1197
1.0000	0.0754	0.0000	0.0000176	0.0755

Table 5.13: The discretization $\eta_\ell(t)$, the homotopy $\nu_\ell(t)$, and the iteration $\mu_\ell(t)$ error estimator for all homotopy steps t in Algorithm 2 applied to Example 2.

t	$\widetilde{\lambda}_\ell(t)$	$\frac{\lvert\lambda_\ell(1)-\widetilde{\lambda}_\ell(t)\rvert}{\lvert\lambda_\ell(1)\rvert}$	#DOF	CPU time
0.0000	19.74139	0.83513	18420	2.49
0.1111	20.97550	0.82482	18790	16.60
0.2222	24.67750	0.79391	29056	25.07
0.3333	30.84926	0.74236	45356	30.59
0.4444	39.49122	0.67019	79339	40.55
0.5556	50.60291	0.57739	125471	57.97
0.6667	64.18232	0.46398	229212	94.00
0.7778	80.23295	0.32994	373527	163.85
0.8889	98.74642	0.17532	374404	223.33
1.0000	119.74011	0.00001	664996	347.61

Table 5.14: The eigenvalue approximation $\widetilde{\lambda}_\ell(t)$, the relative eigenvalue error $\frac{\lvert\lambda_\ell(1)-\widetilde{\lambda}_\ell(t)\rvert}{\lvert\lambda_\ell(1)\rvert}$, the number of degrees of freedom (#DOF), and the CPU time for all homotopy steps t in Algorithm 2 applied to Example 2.

t	$\eta_\ell(t)$	$\nu_\ell(t)$	$\mu_\ell(t)$	error estimator
0.0000	16.8811	263.0051	0.0021032	279.8883
0.3333	18.0718	206.6701	0.0269778	224.7690
0.6667	12.4096	131.4125	0.0267075	143.8488
0.7778	5.3468	93.0901	0.5150339	98.9520
0.8889	4.1960	49.5161	0.1155408	53.8276
0.9259	2.5740	33.2055	0.0664040	35.8459
0.9630	1.6085	16.6848	0.0016130	18.2949
0.9753	0.9158	11.0943	0.0048967	12.0149
0.9877	0.5368	5.5432	0.0030163	6.0829
0.9918	0.3062	3.6710	0.0001411	3.9773
1.0000	0.0585	0.0000	0.0001896	0.0586

Table 5.15: The discretization $\eta_\ell(t)$, the homotopy $\nu_\ell(t)$, and the iteration $\mu_\ell(t)$ error estimator for all homotopy steps t in Algorithm 3 applied to Example 2.

t	$\widetilde{\lambda}_\ell(t)$	$\frac{\lvert\lambda_\ell(1)-\widetilde{\lambda}_\ell(t)\rvert}{\lvert\lambda_\ell(1)\rvert}$	#DOF	CPU time
0.0000	20.23079	0.83104	67	0.05
0.3333	30.72139	0.74343	211	0.29
0.6667	64.20818	0.46377	1283	0.46
0.7778	80.23738	0.32990	4610	1.90
0.8889	98.94516	0.17366	8390	2.47
0.9259	105.61009	0.11800	15539	20.97
0.9630	112.53208	0.06019	27839	38.42
0.9753	114.91115	0.04032	50910	92.94
0.9877	117.31628	0.02023	90675	148.05
0.9918	118.11498	0.01356	162166	340.33
1.0000	119.74169	0.00002	874628	510.75

Table 5.16: The approximation $\widetilde{\lambda}_\ell(t)$, the relative error $\frac{\lvert\lambda_\ell(1)-\widetilde{\lambda}_\ell(t)\rvert}{\lvert\lambda_\ell(1)\rvert}$, the number of degrees of freedom (#DOF), and the CPU time for all homotopy steps t in Algorithm 3 applied Example 2.

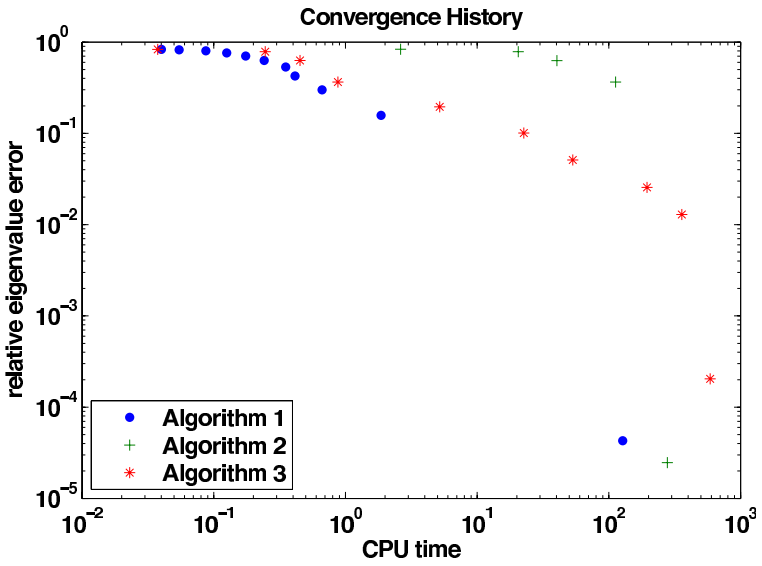


Figure 5.16: Convergence history of Algorithms 1, 2, and 3 with respect to CPU time for Example 1.

5 homotopy steps and not 11 as Algorithm 1. Since the discretization error estimator at each homotopy step is forced to be smaller than the fixed tolerance ε, the number of degrees of freedom is large already for the first homotopy step. Here, in contrast to the previous examples, the approximation for the last step $t = 1$ is less accurate than for the other two algorithms.

The results for Algorithm 3 show the nature of both other algorithms. The step size is chosen adaptively without the loss of accuracy compared to the eigenvalue error of Algorithm 1. Moreover, it needs only one more homotopy step than Algorithm 2 and meshes for steps $t_i < 1$ are much coarser than those of Algorithm 2. Again most of the time is spent to compute the final approximation on the last and the second last level. It is also interesting to see that the second last approximation of the eigenvalue obtained in Algorithm 3 is much better than the corresponding one for Algorithm 2, despite using four times fewer DOFs.

It is remarkable that for this more complicated example the fastest algorithm, with respect to computational time, is Algorithm 3, see Figure 5.23. This experiment strongly underlines the advantages of adaptivity in all three directions, namely in the homotopy, discretization and approximation process.

Figures 5.24, 5.25 and 5.26 show adaptively refined meshes obtained by Algorithms 1, 2 and 3 for Example 3. Note that due to the re-entrant corner meshes show stronger refinement towards the origin. Since the solution for the self-adjoint problem is known to have a strong singularity at the origin, it is not clear whether this extra refinement results from the homotopy process or from the refinement on the last homotopy step $t = 1$. Indeed, looking at the approximated final primal and dual solution does not suggest extra refinement, since they have function values close to zero at the origin, but this may be misleading. The fact that the convection acts only along the x axis is clearly visible in the shape of the discrete primal and dual solution. Note that the primal and dual solution are not mirror images as in the

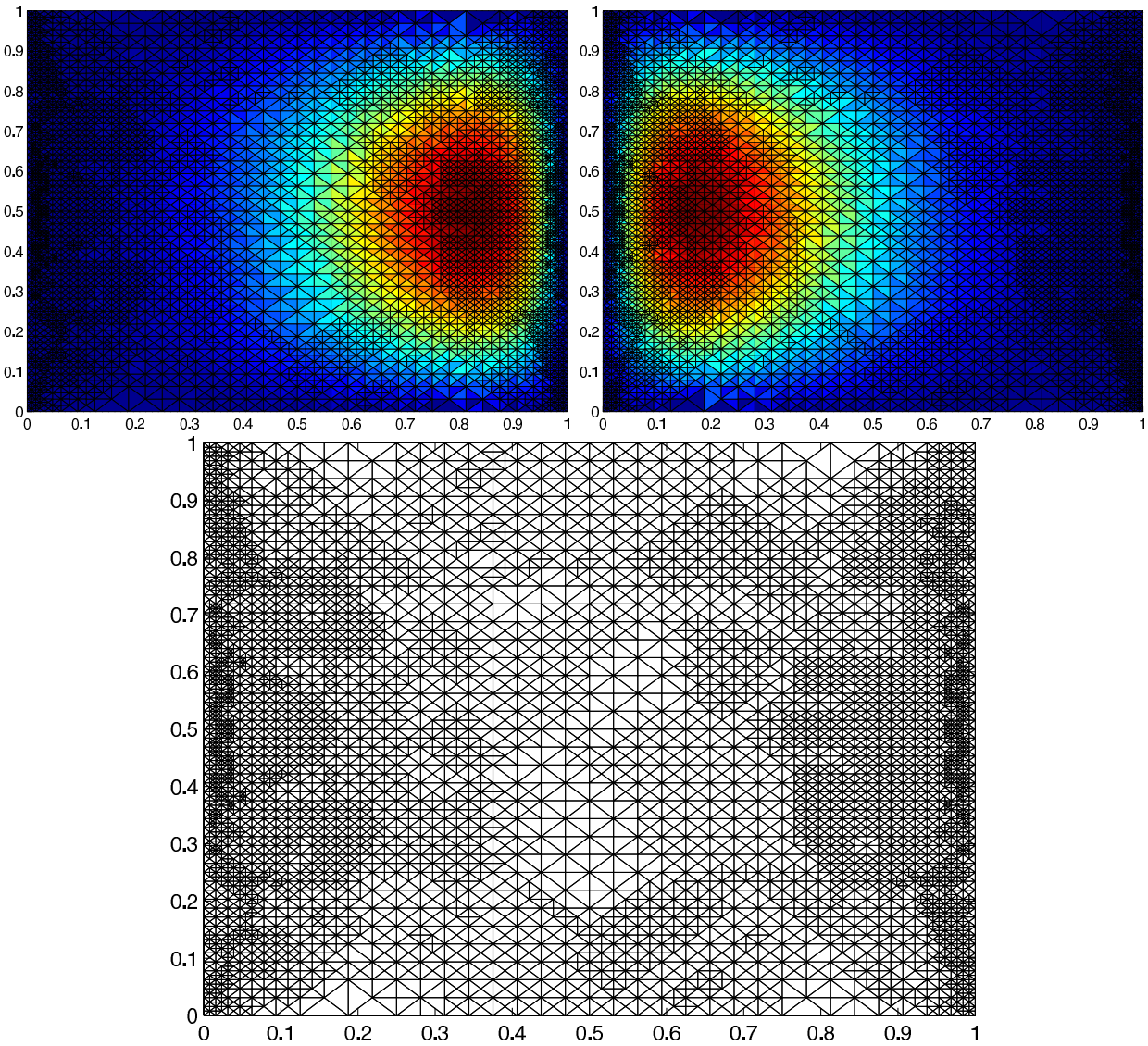


Figure 5.17: Primal (top left) and dual (top right) eigenfunction approximation for the final mesh (bottom) with 5130 nodes for Algorithm 1 applied to Example 1 with $\varepsilon = 2$.

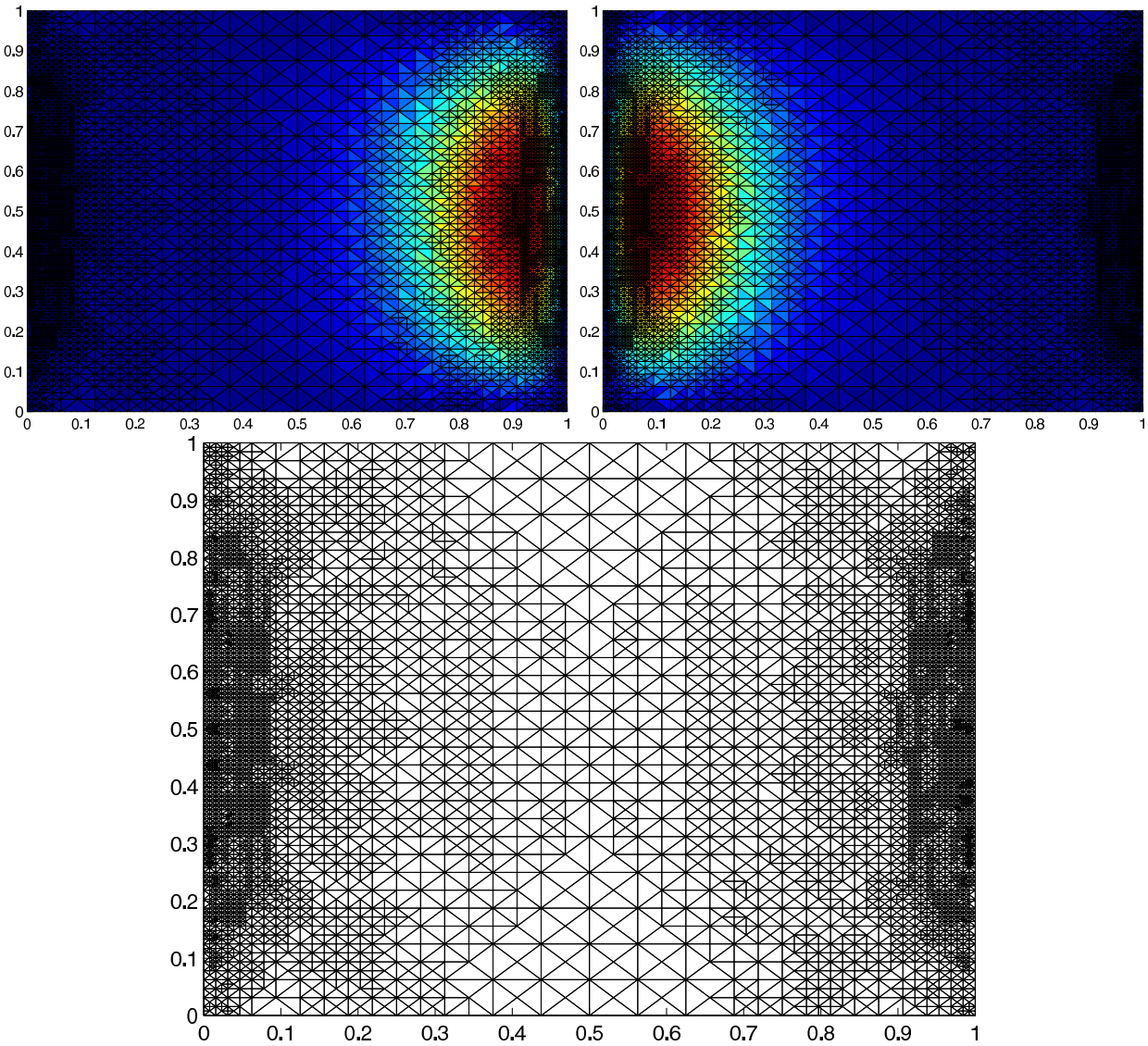


Figure 5.18: Primal (top left) and dual (top right) eigenfunction approximation for the final mesh (bottom) with 6225 nodes for Algorithm 2 applied to Example 1 with $\varepsilon = 10$.

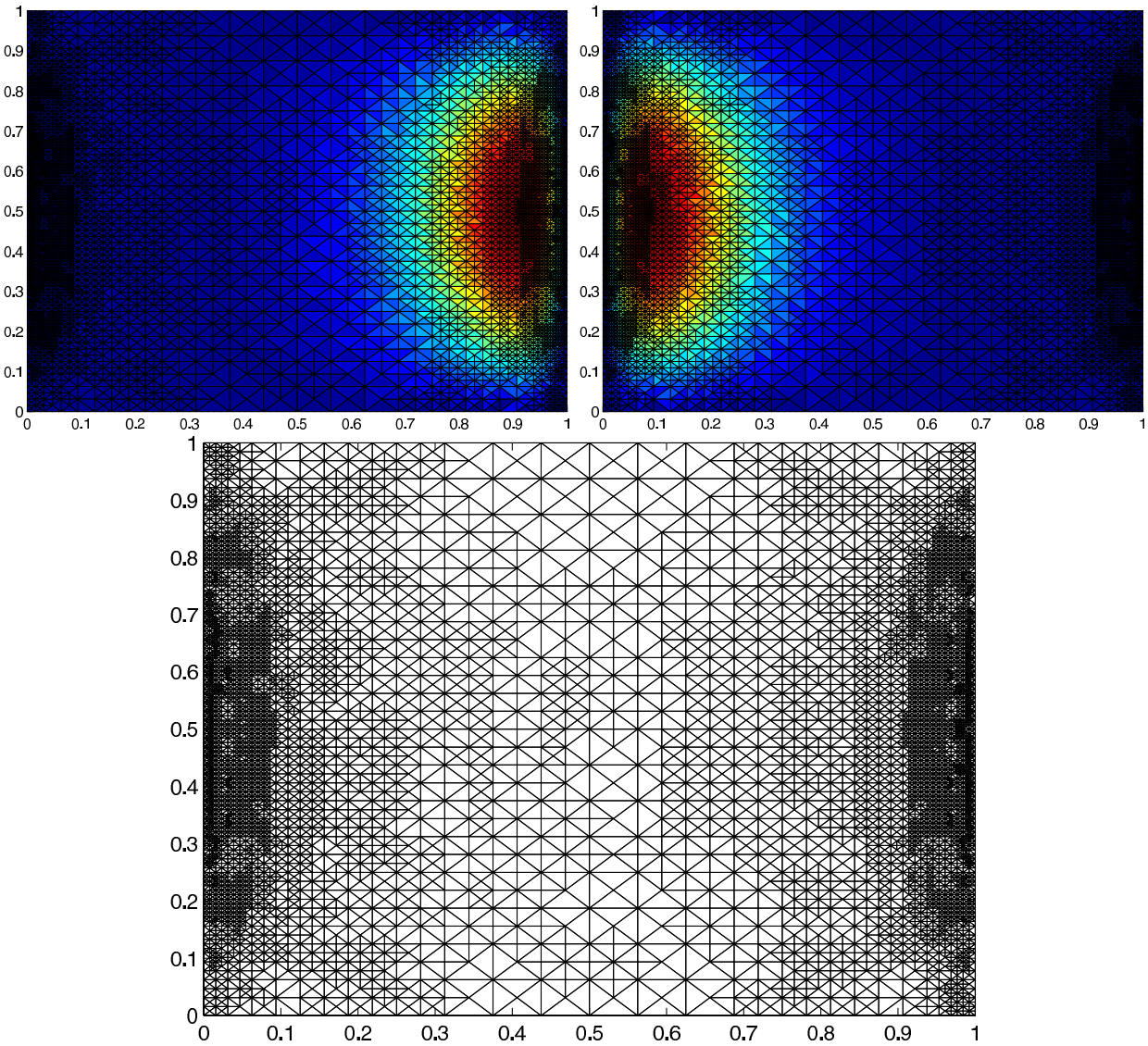


Figure 5.19: Primal (top left) and dual (top right) eigenfunction approximation for the final mesh (bottom) with 6663 nodes for Algorithm 3 applied to Example 1 with $\varepsilon = 10$.

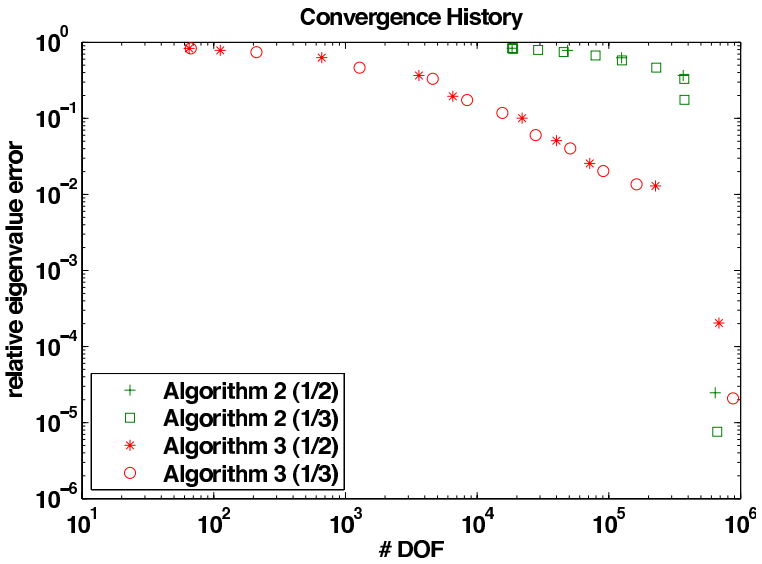


Figure 5.20: Comparison of the convergence history of Algorithms 2, and 3 with respect to #DOF for Example 1 and Example 2.

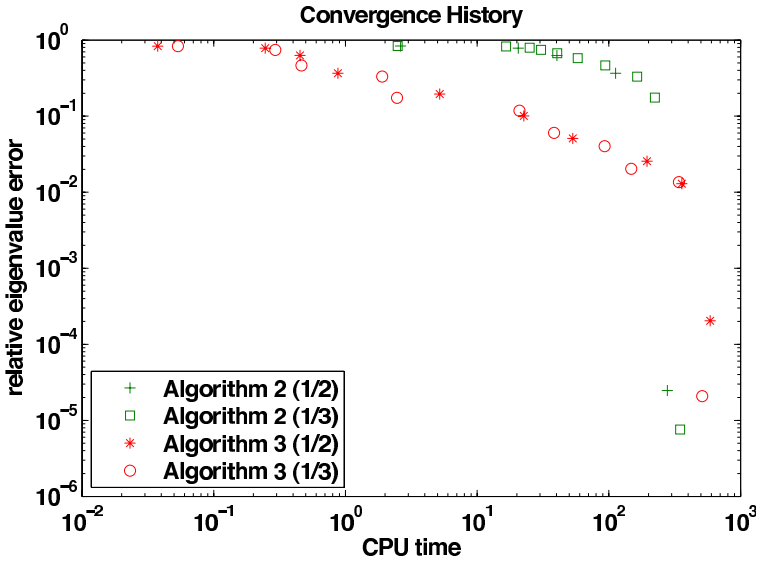


Figure 5.21: Comparison of the convergence history of Algorithms 2, and 3 with respect to CPU time for Example 1 and Example 2.

t	$\eta_\ell(t)$	$\nu_\ell(t)$	$\mu_\ell(t)$	error estimator
0.0	7.1409	90.0380	0.0020414	97.1809
0.1	7.6368	83.3621	0.0197150	91.0186
0.2	5.1146	71.0670	0.0044748	76.1861
0.3	6.3955	67.5948	0.0474799	74.0378
0.4	4.7441	58.7391	0.0509343	63.5341
0.5	3.5712	50.2084	0.0339932	53.8136
0.6	2.5295	42.2268	0.1135079	44.8698
0.7	3.2350	33.5816	0.0020547	36.8187
0.8	2.3482	23.5356	0.0127627	25.8966
0.9	0.9418	11.9678	0.0041016	12.9137
1.0	0.0721	0.0000	0.0068876	0.0790

Table 5.17: The discretization $\eta_\ell(t)$, the homotopy $\nu_\ell(t)$, and the iteration $\mu_\ell(t)$ error estimator for all homotopy steps t in Algorithm 1 applied to Example 3.

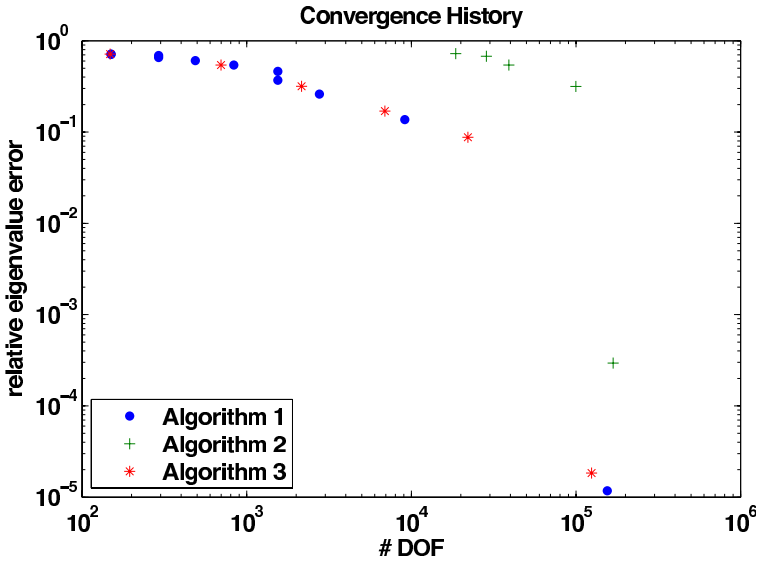


Figure 5.22: Convergence history of Algorithms 1, 2, and 3 with respect to #DOF for Example 3.

previous examples, but again show strong boundary layers on opposite boundary edges.

t	$\widetilde{\lambda}_\ell(t)$	$\frac{\lvert\lambda_\ell(1)-\widetilde{\lambda}_\ell(t)\rvert}{\lvert\lambda_\ell(1)\rvert}$	#DOF	CPU time
0.0	9.87965	0.71479	150	0.05
0.1	10.11007	0.70814	150	0.07
0.2	10.74190	0.68990	292	0.11
0.3	11.94127	0.65527	292	0.13
0.4	13.64386	0.60612	488	0.18
0.5	15.87295	0.54177	835	0.25
0.6	18.63379	0.46207	1546	0.38
0.7	21.85930	0.36895	1546	0.47
0.8	25.62643	0.26020	2769	0.69
0.9	29.89331	0.13702	9117	1.51
1.0	34.63932	0.00001	154994	79.15

Table 5.18: The eigenvalue approximation $\widetilde{\lambda}_\ell(t)$, the relative eigenvalue error $\frac{\lvert\lambda_\ell(1)-\widetilde{\lambda}_\ell(t)\rvert}{\lvert\lambda_\ell(1)\rvert}$, the number of degrees of freedom (#DOF), and the CPU time for all homotopy steps t in Algorithm 1 applied to Example 3.

t	$\eta_\ell(t)$	$\nu_\ell(t)$	$\mu_\ell(t)$	error estimator
0.00	0.0688	64.7314	0.0000002	64.8002
0.25	0.0669	52.1595	0.0000032	52.2264
0.50	0.0864	41.3648	0.0000454	41.4512
0.75	0.0612	24.9728	0.0000235	25.0340
1.00	0.0654	0.0000	0.0002845	0.0657

Table 5.19: The discretization $\eta_\ell(t)$, the homotopy $\nu_\ell(t)$, and the iteration $\mu_\ell(t)$ error estimator for all homotopy steps t in Algorithm 2 applied to Example 3.

t	$\widetilde{\lambda}_\ell(t)$	$\frac{\lvert\lambda_\ell(1)-\widetilde{\lambda}_\ell(t)\rvert}{\lvert\lambda_\ell(1)\rvert}$	#DOF	CPU time
0.00	9.64199	0.72165	18602	2.26
0.25	11.20316	0.67658	28573	15.85
0.50	15.88943	0.54129	39141	20.94
0.75	23.70187	0.31576	99976	37.13
1.00	34.62952	0.00029	168258	74.28

Table 5.20: The eigenvalue approximation $\widetilde{\lambda}_\ell(t)$, the relative eigenvalue error $\frac{\lvert\lambda_\ell(1)-\widetilde{\lambda}_\ell(t)\rvert}{\lvert\lambda_\ell(1)\rvert}$, the number of degrees of freedom (#DOF), and the CPU time for all homotopy steps t in Algorithm 2 applied to Example 3.

t	$\eta_\ell(t)$	$\nu_\ell(t)$	$\mu_\ell(t)$	error estimator
0.0000	7.1752	90.1329	0.0023012	97.3104
0.5000	4.3043	50.7307	0.0108970	55.0459
0.7500	2.7624	29.2814	0.1024431	32.1463
0.8750	1.1924	14.9202	0.0143657	16.1270
0.9375	0.4360	7.5295	0.0208416	7.9863
1.0000	0.0932	0.0000	0.0000282	0.0932

Table 5.21: The discretization $\eta_\ell(t)$, the homotopy $\nu_\ell(t)$, and the iteration $\mu_\ell(t)$ error estimator for all homotopy steps t in Algorithm 3 applied to Example 3.

t	$\widetilde{\lambda}_\ell(t)$	$\frac{\lvert\lambda_\ell(1)-\widetilde{\lambda}_\ell(t)\rvert}{\lvert\lambda_\ell(1)\rvert}$	#DOF	CPU time
0.0000	9.88054	0.71476	148	0.11
0.5000	15.87104	0.54183	698	0.34
0.7500	23.66888	0.31671	2156	0.98
0.8750	28.75123	0.16999	6912	3.10
0.9375	31.60501	0.08761	22058	11.88
1.0000	34.63909	0.00002	124469	32.37

Table 5.22: The approximation $\widetilde{\lambda}_\ell(t)$, the relative error $\frac{\lvert\lambda_\ell(1)-\widetilde{\lambda}_\ell(t)\rvert}{\lvert\lambda_\ell(1)\rvert}$, the number of degrees of freedom (#DOF), and the CPU time for all homotopy steps t in Algorithm 3 applied to Example 3.

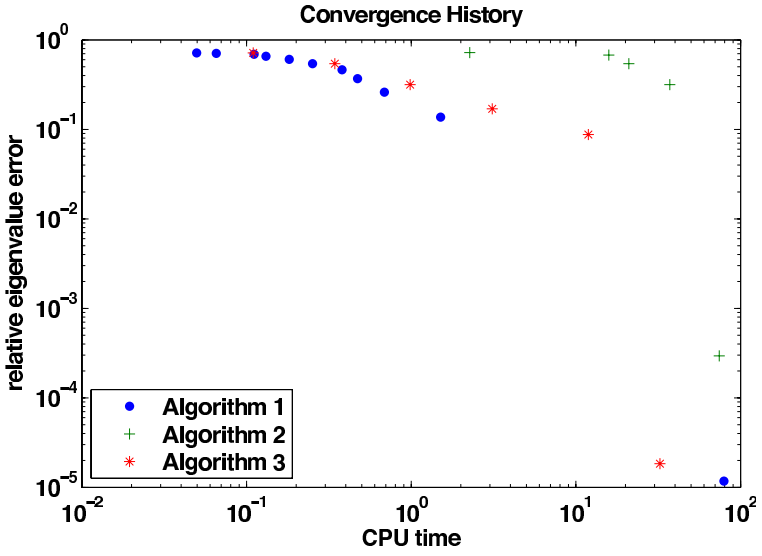


Figure 5.23: Convergence history of Algorithms 1, 2, and 3 with respect to CPU time for Example 3.

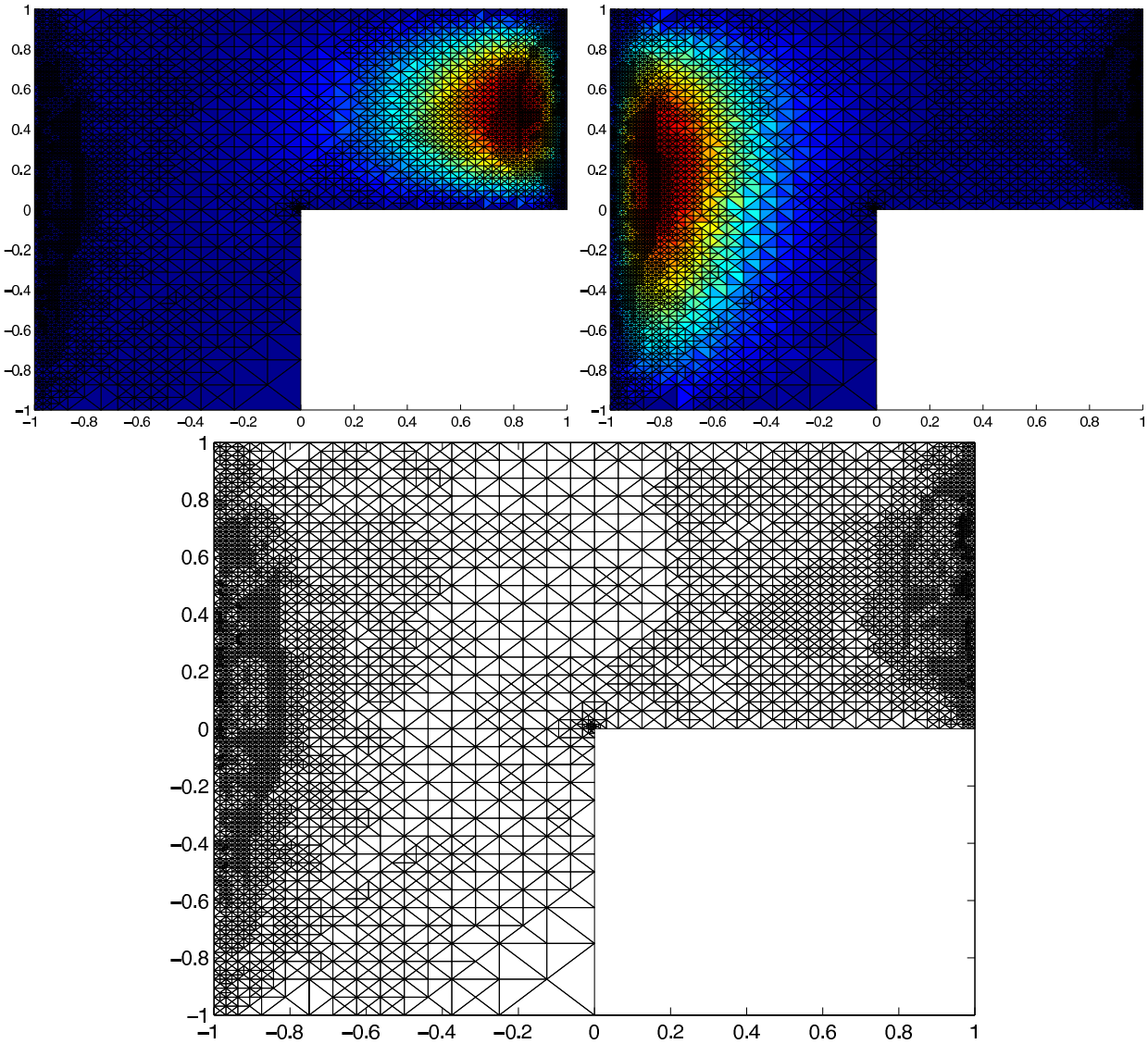


Figure 5.24: Primal (top left) and dual (top right) eigenfunction approximation for the final mesh (bottom) with 4926 nodes for Algorithm 1 for Example 3 with $\varepsilon = 3$.

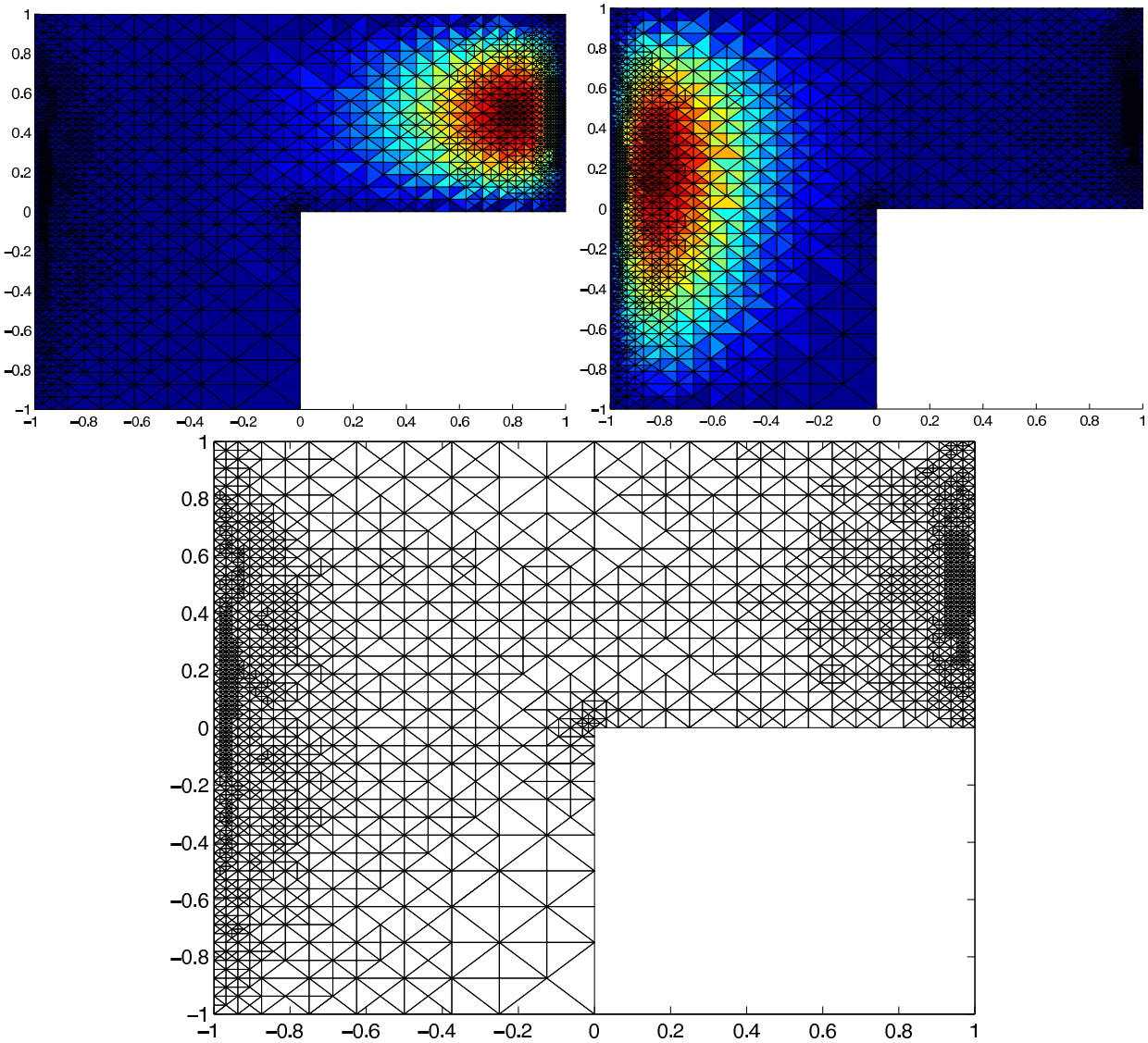


Figure 5.25: Primal (top left) and dual (top right) eigenfunction approximation for the final mesh (bottom) with 3694 nodes for Algorithm 2 for Example 3 with $\varepsilon = 3$.

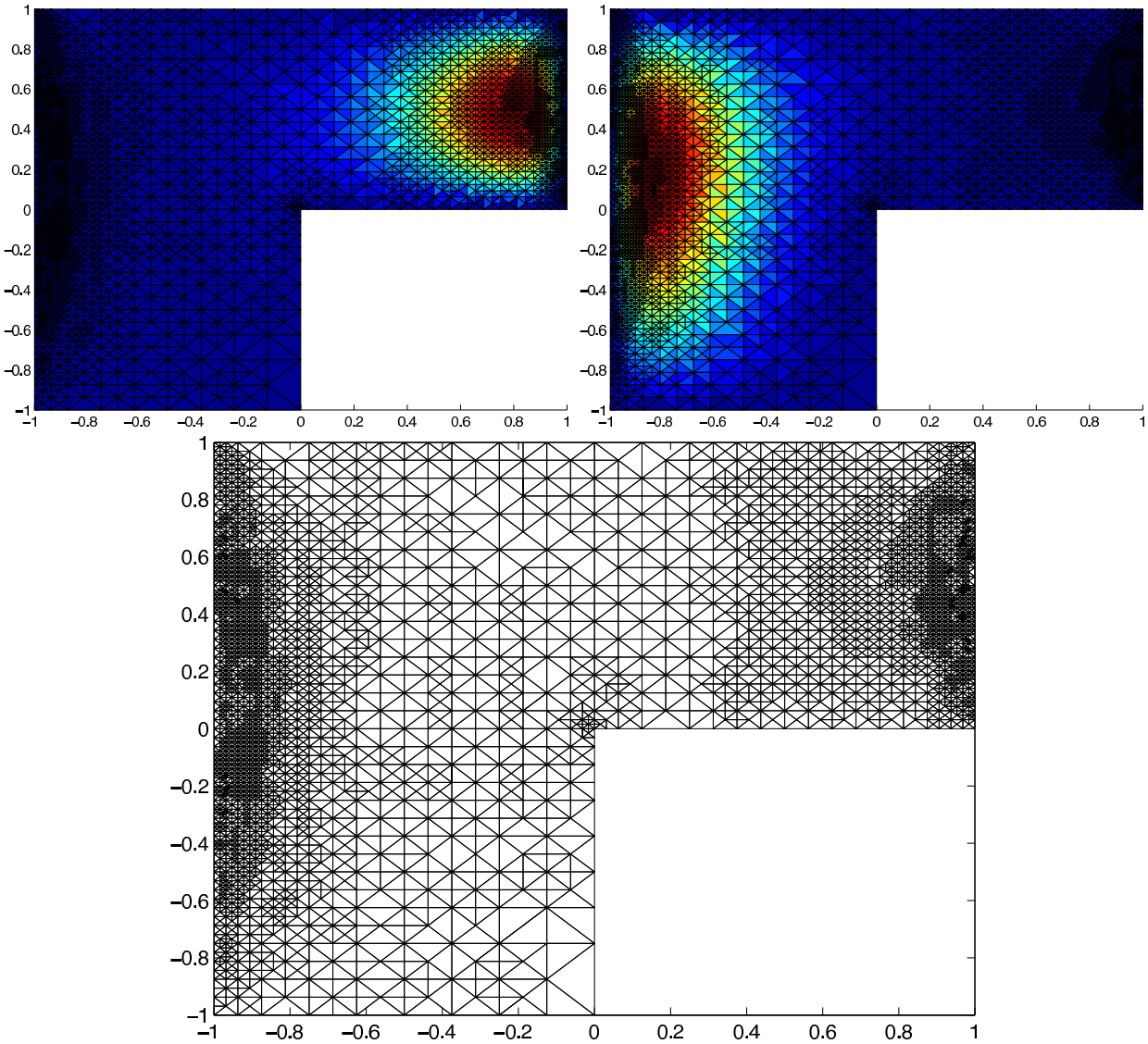


Figure 5.26: Primal (top left) and dual (top right) eigenfunction approximation for the final mesh (bottom) with 3745 nodes for Algorithm 3 for Example 3 with $\varepsilon = 3$.

Chapter 6

Conclusions

At the beginning of this work we wanted to answer all the questions and solve all the problems. Over time, we realized that each answered question generates several new and instead of being closer to our final goal we are actually moving away. Although, we did not invent the wheel, we think that this work has shed a light on some important aspects of adaptive finite element methods for the PDE eigenvalue problems.

A study of self-adjoint eigenvalue problems allowed us to investigate our new ideas, having the complete theoretical framework in hand. Based on a simple example we confirmed the danger of underestimating the influence of the algebraic components on adaptive finite element methods. With our new AFEMLA algorithm we extended the standard AFEM approaches to incorporate the approximation error into the adaptation process. Furthermore, we showed that the adaptive mesh refinement can be steered by the discrete residual vector, e.g., when the problem is stated in the discrete formulation where only the underlying matrices and meshes are available. With classical perturbation results we were able to prove upper bounds for the eigenvalue and the eigenfunction error. Under certain assumptions similar results were obtained for convection-diffusion problems. All our statements were illustrated with number of numerical experiments. The idea of the AFEMLA algorithm where the inexact solution is considered, has recently earned a large interest.

Following [9] we have formulated functional perturbation results for PDE eigenvalue problems including the functional backward error and the functional condition number. These results are used to relate the eigenvalue and the eigenvector error to the residual and furthemore to establish a combined a posteriori error estimator embodying the discretization and the approximation error. At the end of the chapter on self-adjoint problems the balanced AFEM algorithm was introduced. Exploiting the result of [65] on a discrete $H^1(\Omega)$- and $H^{-1}(\Omega)$-norm, we have determined a combined a posteriori error estimator and designed the balanced AFEM algorithm which significantly reduces the number of eigensolver iterations. The eigensolver stopping criterion is based on the equilibrating strategy, i.e., iterations proceed as long as the discrete part of the error estimator dominates the continuous part. Several numerical examples confirm the reliability of our estimator. A formal proof for the efficiency of the combined error estimator and the convergence of the balancing algorithm is still

an open question. Also there are several other choices of the eigensolver stopping criterion which are an interesting subject for further research. The convergence of the inexact adaptive algorithm based on a slightly different combined a posteriori error estimator is a subject of the ongoing work.

The second part of the thesis was dedicated to exploit experiences from the self-adjoint problems and apply them to much more complicated problems. We have investigated the possible extension of the AFEMLA algorithm to the class of real diagonalizable convection-diffusion problems. Unfortunately, since the eigenfunction condition numbers grow exponentially with respect to the size of the convection coefficient, we have reached the limits of our algorithm very fast. Therefore, we have continued our work analyzing so-called homotopy methods. A new adaptive homotopy approach was introduced to determine the particular eigenvalue of the convection-diffusion problem. The idea of multi-way adaptivity based on three different errors, the homotopy, the discretization and the iteration error, was discussed. Also here, several doors are still open for further research. The phenomena of path jumping or bifurcation points as well as the subspace version of the algorithm have to be taken into consideration.

Since decades practical applications lead to challenging PDE eigenvalue problems. Although adaptive methods have gained a recognition and are well-established, they are far away from needs in reality. Techniques like model reduction or reduced order modeling combined with adaptive algorithms are still an open question. Multi-way adaptive methods incorporating optimization parameters are of particular interest. The mathematical theory and algorithms for really nonlinear eigenvalue problems have to be developed. General non-self-adjoint eigenvalue problems are still a great challenge and investigating methods like the two-sided Arnoldi or the Jacobi-Davidson seems to be the right research direction. Moreover, implementation details like subspace recycling and deflation techniques in homotopy or Newton methods need to be improved and combined within the adaptive loop. Most of the commercially available codes are not satisfactory and need to be redesigned. Up to now, there are only few results going beyond the elliptic PDE eigenvalue problems, which keeps the whole research area of adaptive finite element methods for the parabolic and hyperbolic PDE eigenvalue problems still open. Furthermore, since research in quantum physic and chemistry requires efficient large scale computations in multi-dimensions, the field of adaptive methods for tensors is the future of multi-linear algebra.

Chapter 7

Appendix

7.1 Appendix A

Proof of Proposition 4.1

Proof. From (2.31) we see that

$$\|e\|^2 = \|u_k - u_{k,h}\|^2 = a(u_k, u_k) - 2a(u_k, u_{k,h}) + a(u_{k,h}, u_{k,h}).$$

Since

$$\nabla u(x) = \sqrt{2}\pi \cos(\pi x)$$

it follows that

$$\begin{aligned} a(u,u) &= \int_0^1 \nabla u(x)\nabla u(x)dx = \int_0^1 (\sqrt{2}\pi\cos(\pi x))^2 dx \\ &= \pi^2 \int_0^1 2\cos^2(\pi x)dx = \pi^2 \int_0^1 (1+\cos(2\pi x))dx \\ &= \pi^2 \int_0^1 1dx + \pi^2 \int_0^1 \cos(2\pi x)dx = \pi^2. \end{aligned} \tag{7.1}$$

From

$$\nabla u_h(x)_{|_{I_i}} = -\frac{1}{h}\sqrt{2}\sin(\pi i h) + \frac{1}{h}\sqrt{2}\sin(\pi(i+1)h)$$

and property (4.4) we get

$$\begin{aligned} a(u_h,u_h) &= \int_0^1 \nabla u_h(x)\nabla u_h(x)dx = \sum_{i=0}^{n} \int_{I_i} (\nabla u_h(x)_{|_{I_i}})^2 dx \\ &= \sum_{i=0}^{n} (\nabla u_h(x)_{|_{I_i}})^2 \int_{I_i} dx = h\sum_{i=0}^{n} (\nabla u_h(x)_{|_{I_i}})^2 \overset{\text{MAPLE}}{=} -\frac{2(\cos(\pi h)-1)}{h^2}. \end{aligned} \tag{7.2}$$

Additionally,

$$\begin{aligned} a(u,u_h) &= \int_0^1 \nabla u(x)\nabla u_h(x)dx = \int_0^1 \sqrt{2}\pi\cos(\pi x)\nabla u_h(x)dx \qquad (7.3)\\ &= \sum_{i=0}^{n} \int_{x_i=ih}^{x_{i+1}=(i+1)h} \sqrt{2}\pi\cos(\pi x)\nabla u_h(x)dx \\ &= \sum_{i=0}^{n} \nabla u_h(x)_{|_{I_i}} \int_{x_i=ih}^{x_{i+1}=(i+1)h} \sqrt{2}\pi\cos(\pi x)dx \\ &= \sum_{i=0}^{n} \nabla u_h(x)_{|_{I_i}} \sqrt{2}\pi(\frac{1}{\pi}\sin(\pi(i+1)h) - \frac{1}{\pi}\sin(\pi ih)) \\ &= \sum_{i=0}^{n} \nabla u_h(x)_{|_{I_i}} \sqrt{2}(\sin(\pi(i+1)h) - \sin(\pi ih)) \stackrel{\text{MAPLE}}{=} -\frac{2(\cos(\pi h)-1)}{h^2}. \end{aligned}$$

From (7.1), (7.2), and (7.3) we see that

$$\begin{aligned} \|e\|^2 &= a(e,e) = a(u-u_h, u-u_h) = a(u,u) - 2a(u,u_h) + a(u_h,u_h) \\ &= \pi^2 - 2(-\frac{2(\cos(\pi h)-1)}{h^2}) + (-\frac{2(\cos(\pi h)-1)}{h^2}) = \pi^2 + \frac{2(\cos(\pi h)-1)}{h^2}. \end{aligned}$$

Using *Taylor expansion* for $\cos(ih)$ we obtain

$$\begin{aligned} \|e\|^2 &= \pi^2 + \frac{2(\cos(\pi h)-1)}{h^2} = \pi^2 + \frac{2(1-\frac{\pi^2h^2}{2}+\frac{\pi^4h^4}{4}+\ldots+1)}{h^2} \\ &= \pi^2 - \pi^2 + \frac{\pi^4h^4}{h^2} + \ldots = O(h^2), \end{aligned}$$

which yields $\|e\| = O(h)$ □

Proof of Proposition 4.2
Since $u_k(x) = \sqrt{2}\sin(k\pi x)$, the $L_2(\Omega)$-norm of the eigenfunction u corresponding to the smallest eigenvalue is given by

$$\begin{aligned} \|u\|^2_{L_2(\Omega)} &= (u,u)_{L_2(\Omega)} = \int_0^1 u(x)u(x)dx = \int_0^1 (\sqrt{2}\sin(\pi x))^2 dx \qquad (7.4)\\ &= \int_0^1 2\sin^2(\pi x)dx = \int_0^1 (1-\cos(2\pi x))dx = \int_0^1 1dx - \int_0^1 \cos(2\pi x)dx \\ &= 1 - (\frac{1}{2\pi}\sin(2\pi) - \frac{1}{2\pi}\sin(0)) = 1. \end{aligned}$$

Using *Simpson's rule* the following integrals can be computed

$$\begin{aligned}
\int_{I_i} \varphi_i(x)\varphi_{i+1}(x)dx &= \int_{ih}^{(i+1)h} \varphi_i(x)\varphi_{i+1}(x)dx \qquad (7.5)\\
&= \frac{(i+1)h-ih}{6}\Big(\varphi_i(ih)\varphi_{i+1}(ih) + 4\varphi_i((i+\frac{1}{2})h)\varphi_{i+1}((i+\frac{1}{2})h)\\
&+ \varphi_i((i+1)h)\varphi_{i+1}((i+1)h)\Big)\\
&= \frac{h}{6}(1\cdot 0 + 4\cdot\frac{1}{2}\cdot\frac{1}{2} + 0\cdot 1) = \frac{h}{6},
\end{aligned}$$

$$\begin{aligned}
\int_{I_i\cup I_{i+1}} \varphi_i(x)\varphi_i(x)dx &= 2\int_{(i-1)h}^{ih} \varphi_i(x)\varphi_i(x)dx \qquad (7.6)\\
&= 2\frac{ih-(i-1)h}{6}\Big(\varphi_i((i-1)h)\varphi_i((i-1)h) + 4\varphi_i((i-\frac{1}{2})h)\varphi_i((i-\frac{1}{2})h)\\
&+ \varphi_i(ih)\varphi_i(ih)\Big)\\
&= 2\frac{h}{6}(0\cdot 0 + 4\cdot\frac{1}{2}\cdot\frac{1}{2} + 1\cdot 1) = 2\frac{2h}{6} = \frac{2h}{3}.
\end{aligned}$$

From (2.30), (7.5), and (7.6) we find that

$$\begin{aligned}
\|u_h\|^2_{L_2(\Omega)} &= (u_h,u_h)_{L_2(\Omega)} = \int_0^1 u_h(x)u_h(x)dx \qquad (7.7)\\
&= \int_0^1 \sum_{i=1}^n \varphi_i(x)\sqrt{2}\sin(\pi x)\sum_{i=j}^n \varphi_j(x_j)\sqrt{2}\sin(\pi x_j)dx\\
&= \sum_{i=1}^n\sum_{j=1}^n \int_0^1 \varphi_i(x)\sqrt{2}\sin(\pi x_i)\varphi_j(x)\sqrt{2}\sin(\pi x_j)dx\\
&= \sum_{i=1}^n\sum_{j=1}^n 2\sin(\pi x_i)\sin(\pi x_j)\int_{I_i\cup I_{i+1}} \varphi_i(x)\varphi_j(x)dx\\
&= \sum_{i=1}^n 2\sin(\pi x_i)\Big(\frac{h}{6}\sin(\pi x_{i-1} + \frac{2h}{3}\sin(\pi x_i) + \frac{h}{6}\sin(\pi x_{i+1})\Big)\\
&\overset{\text{MAPLE}}{=} \frac{1}{3}\cos(\pi h) + \frac{2}{3},
\end{aligned}$$

and

$$
\begin{aligned}
(u,u_h)_{L_2(\Omega)} &= \int_0^1 u(x)u_h(x)dx = \int_0^1 \sqrt{2}\sin(\pi x)\sum_{i=1}^n \varphi_i(x)\sqrt{2}\sin(\pi x)dx \qquad (7.8)\\
&= 2\int_0^1 \sin(\pi x)\sum_{i=1}^n \varphi_i(x)\sin(\pi x_i)dx\\
&= 2\sum_{i=1}^n \sin(\pi x_i)\int_0^1 \sin(\pi x)\varphi_i(x)dx\\
&= 2\sum_{i=1}^n \sin(\pi x_i)\int_{I_i\cup I_{i+1}} \sin(\pi x)\varphi_i(x)dx\\
&= 2\sum_{i=1}^n \sin(\pi x_i)\Big(\int_{(i-1)h}^{ih} \sin(\pi x)\frac{x-(i-1)h}{h}dx + \int_{ih}^{(i+1)h} \sin(\pi x)\frac{ih-x}{h}dx\Big)\\
&\overset{\text{MAPLE}}{=} -\frac{2(\cos(\pi h)-1)}{\pi^2 h^2}.
\end{aligned}
$$

(7.4), (7.7), and (7.8) together give

$$
\cos\angle(u,u_h) \overset{\text{MAPLE}}{=} 6\frac{|\frac{\cos(\pi h)-1}{h^2}|}{\pi^2\sqrt{3\cos(\pi h)+6}} \quad ,
$$

and

$$
\sin\angle(u,u_h) \overset{\text{MAPLE}}{=} \sqrt{\frac{\pi^4\cos(\pi h)+2\pi^4-12|\frac{(\cos(\pi h)-1)^2}{h^4}|}{\pi^4(\cos(\pi h)+2)}}.
$$

□

Bibliography

[1] M. Ainsworth and J. T. Oden. A posteriori error estimators for second order elliptic systems II. An optimal order process for calculating self-equilibrating fluxes. *Comput. Math. Appl.*, 26(9):75–87, 1993.

[2] M. Ainsworth and J.T. Oden. *A posteriori error estimation in finite element analysis.* John Wiley and Sons Inc., New York, 2000.

[3] A. C. Aitken. On Bernoulli's numerical solution of algebraic equations. *Proceedings Royal Soc. Edinburgh*, 46:289–305, 1926.

[4] J. Alberty, C. Carstensen, and S.A. Funken. Remarks around 50 lines of Matlab: short finite element implementation. *Numer. Algorithms*, 20:117–137, 1999.

[5] E. L. Allgower and K. Georg. Continuation and path following. *Acta Numerica*, pages 1–64, 1993.

[6] P. Arbenz and D. Kressner. Numerische Methoden für grosse Matrixeigenwertprobleme FS 2010. Lecture Notes. `http://people.inf.ethz.ch/arbenz/ewp/lnotes.html`, 2010.

[7] M. E. Argentati, A. V. Knyazev, C. C. Paige, and I. Panayotov. Bounds on changes in Ritz values for a perturbed invariant subspace of a Hermitian matrix. *SIAM J. Matrix Anal. Appl.*, 30:548–559, 2008.

[8] M. Arioli, E. H. Georgoulis, and D. Loghin. Convergence of inexact adaptive finite element solvers for elliptic problems. Technical Report RAL-TR-2009-21, Rutherford Appleton Laboratory, 2009.

[9] M. Arioli, E. Noulard, and A. Russo. Stopping criteria for iterative methods: applications to PDE's. *CALCOLO*, 38:97–112, 2001.

[10] I. Babuška, J. Chandra, and J. E. Flaherty. *Adaptive Computational Methods for Partial Differential Equations.* SIAM Publications, Philadelphia, PA, USA, 1983.

[11] I. Babuška and J. E. Osborn. Finite Element-Galerkin approximation of the eigenvalues and eigenvectors of selfadjoint problems. *Math. Comp.*, 52:275–297, 1989.

[12] I. Babuška and J. E. Osborn. *Eigenvalue problems. Handbook of Numerical Analysis Vol. II.* North-Holland, Amsterdam, 1991.

[13] Z. Bai, J. Demmel, J. Dongarra, A. Ruhe, and H. van der Vorst. *Templates for the Solution of Algebraic Eigenvalue Problem. A Practical Guide.* SIAM Publications, Philadelphia, 2000.

[14] W. Bangerth and R. Rannacher. *Adaptive Finite Element Methods for Differential Equations.* Birkhäuser, Basel, 2003.

[15] R. E. Bank and R. K. Smith. A posteriori error estimates based on hierarchical bases. *SIAM J. Numer. Anal.*, 30:921–935, 1993.

[16] R. E. Bank and A. Weiser. Some a posteriori error estimators for elliptic partial differential equations. *Math. Comp.*, 44:283–301, 1985.

[17] F. Bauer and C. Fike. Norms and exclusion theorems. *Numer. Math.*, 2:137–141, 1960.

[18] R. Becker and R. Rannacher. An optimal control approach to a posteriori error estimation in finite element methods. *Acta Numerica*, 10:1–102, 2001.

[19] M. Beerten. *Mayco Design and Engineering.* Interior design tutorial. `http://www.maycodesign.nl/_images/tutorials/Car_design_tutorial_car_interiour_design_Maarten_Beerten_Maycodesign.pdf`.

[20] M. Bennani and T. Braconnier. Stopping criteria for eigensolvers. Technical Report TR/PA/94/22, CERFACS, 1994.

[21] SFE GmbH Berlin. *SFE AKUSMOD.* `http://sfe1.extern.tu-berlin.de/sfe_first.html`.

[22] R. Bhatia. *Perturbation Bounds for Matrix Eigenvalues.* SIAM Publications, Philadelphia, 2007.

[23] M. Braack and A. Ern. A posteriori control of modeling errors and discretization errors. *Multiscale Model. Simul.*, 1(2):221–238, 2003.

[24] D. Braess. *Finite Elements.* Cambridge University Press, New York, 2008.

[25] J. Bramble, J. Pasciak, and J. Xu. Parallel multilevel preconditioners. *Math. Comp.*, 55:1–22, 1990.

[26] S. C. Brenner and C. Carstensen. Finite element methods. In E. Stein, R. de Borst, and T.J.R. Huges, editors, *Encyclopedia of Computational Mechanics, Vol. I*, pages 73–114. John Wiley and Sons Inc., New York, 2004.

[27] S.C. Brenner and L.R. Scott. *The Mathematical Theory of Finite Element Methods.* Springer-Verlag, Berlin, 2008.

[28] C. Burstedde. On the Numerical Evaluation of Fractional Sobolev Norms. *Communications on Pure and Applied Analysis*, 6(3):587–605, 2007.

[29] A. Byfut, J. Gedicke, D. Günther, J. Reininghaus, and S. Wiedemann. openFFW, The Finite Element Framework. GNU General Public License v.3.

[30] C. Carstensen. Some remarks on the history and future of averaging techniques in a posteriori finite element error analysis. *Z. Angew. Math. Mech.*, 84:3–21, 2004.

[31] C. Carstensen and J. Gedicke. An oscillation-free adaptive FEM for symmetric eigenvalue problems. Preprint 489, DFG Research Center Matheon, Strasse des 17. Juni 136, D-10623 Berlin, 2008.

[32] C. Carstensen and J. Gedicke. An adaptive finite element eigenvalue solver of quasi-optimal computational complexity. Preprint 662, DFG Research Center Matheon, Strasse des 17. Juni 136, D-10623 Berlin, 2009.

[33] C. Carstensen, J. Gedicke, V. Mehrmann, and A. Międlar. An adaptive homotopy approach for non-selfadjoint eigenvalue problems. Preprint 718, DFG Research Center Matheon, Strasse des 17. Juni 136, D-10623 Berlin, 2010.

[34] C. Carstensen and C. Merdon. Estimator competition for Poisson problems. *J. of Comp. Math.*, 28(3):309–330, 2010.

[35] F. Chaitin-Chatelin. The influence of nonnormality on matrix computations. In R. J. Plemmons and C. D. Meyer, editors, *Linear Algebra, Markov Chains and Queing Models*, pages 13–19. Springer-Verlag, New York, 1992.

[36] F. Chaitin-Chatelin and V. Fraysse. *Lectures on Finite Precision Computations - Software, Environments, and Tools*. SIAM Publications, 1996.

[37] F. Chatelin. *Spectral Approximation of Linear Operators*. Academic Press, New York, 1983.

[38] Z. Chen. *Finite element methods and their applications*. Springer-Verlag, Berlin, 2005.

[39] M. T. Chu. A simple application of the homotopy method to symmetric eigenvalue problems. *Linear Algebra Appl.*, 59:85–90, 1984.

[40] P. G. Ciarlet. *Introduction to Numerical Linear Algebra and Optimisation*. Cambridge University Press, Cambridge, UK, 1989. (with the assistance of B. Miara and J.-M. Thomas for the exercises).

[41] P. G. Ciarlet. *The Finite Element Method for Elliptic Problems*. SIAM Publications, Philadelphia, 2002.

[42] W. Dahmen, T. Rohwedder, R. Schneider, and A. Zeiser. Adaptive eigenvalue computation - complexity estimates. *Numer. Math.*, 110:277–312, 2008.

[43] X. Dai, J. Xu, and A. Zhou. Convergence and optimal complexity of adaptive finite element eigenvalue computations. *Numer. Math.*, 110:313–355, 2008.

[44] C. Davis and W. Kahan. Some new bounds on perturbation of subspaces. *Bull. Amer. Math. Soc*, 75:863–868, 1969.

[45] C. Davis and W. Kahan. The rotation of eigenvectors by a perturbation, III. *SIAM J. Numer. Anal.*, 7:1–46, 1970.

[46] J. W. Demmel. *Applied Numerical Linear Algebra*. SIAM Publications, Philadelphia, 1997.

[47] P. Deuflhard, P. Leinen, and H. Yserentant. Concepts of an adaptive hierarchical finite element code. *IMPACT Comput. in. Sci. and Eng.*, 1:3–35, 1989.

[48] W. Dörfler. A convergent adaptive algorithm for Poisson‘s equation. *SIAM J. Numer. Anal.*, 33:1106–1124, 1996.

[49] W. Dörfler and R. H. Nochetto. Small data oscillation implies the saturation assumption. *Numer. Math.*, 91:1–12, 2002.

[50] R. G. Durán, C. Padra, and R. Rodríguez. A posteriori error estimates for the finite element approximation of eigenvalue problems. *Math. Mod. Meth. Appl. Sci.*, 13:1219–1229, 2003.

[51] S. C. Eisenstat and I. C. F. Ipsen. Relative perturbation results for eigenvalues and eigenvectors of diagonalizable matrices. *BIT*, 38(3):502–509, 1998.

[52] H. Elman, D. Silvester, and A. Wathen. *Finite Elements and Fast Iterative Solvers with applications in incompressible fluid dynamics*. Oxford University Press, Oxford, 2005.

[53] T. Ericsson and A. Ruhe. The spectral transformation lanczos method for the numerical solution of large sparse generalized symmetric eigenvalue problems. *Math. Comp.*, 35(152):1251–1268, 1980.

[54] L. C. Evans. *Partial differential equations*. American Mathematical Society, 2000.

[55] S. Ferraz-Leite, C. Ortner, and D. Praetorius. Convergence of simple adaptive Galerkin schemes based on h - h/2 error estimators. *Numer. Math.*, 116:291–316, 2010.

[56] E. M. Garau and P. Morin and C. Zuppa. Convergence of adaptive finite element methods for eigenvalue problems. Preprint arXiv:0803.0365v1, 2008. `http://arxiv.org/abs/0803.0365v1`.

[57] J. Gedicke and C. Carstensen. A posteriori error estimators for non-symmetric eigenvalue problems. Preprint 659, DFG Research Center MATHEON, Straße des 17.Juni 136, D-10623 Berlin, 2009.

[58] S. Giani. *Convergence of Adaptive Finite Element Methods for Elliptic Eigenvalue Problems with Applications to Photonic Crystals*. PhD thesis, University of Bath, 2008.

[59] S. Giani and I. G. Graham. A convergent adaptive method for elliptic eigenvalue problems. *SIAM J. Numer. Anal.*, 47:1067–1091, 2009.

[60] M. S. Gockenbach. *Understanding and Implementing the Finite Element Method*. SIAM Publications, 2006.

[61] G. H. Golub and C. F. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, third edition, 1996.

[62] L. Grubišić and J. S. Ovall. On estimators for eigenvalue/eigenvector approximations. *Math. Comp.*, 78(266):739–770, 2009.

[63] E. Hairer, S. P. Nørsett, and G. Wanner. *Solving Ordinary Differential Equations I: Nonstiff Problems*. Springer, Berlin, Germany, 2nd edition, 1993.

[64] H. Harbrecht and R. Schneider. On error estimation in finite element methods without having Galerkin orthogonality. Preprint 457, Berichtsreihe des SFB 611, Universität Bonn, 2009.

[65] U. Hetmaniuk and R. Lehoucq. Uniform accuracy of eigenpairs from a shift-invert Lanczos method. *SIAM J. Matrix Anal. Appl.*, 28:927–948, 2006.

[66] V. Heuveline and C. Bertsch. On multigrid methods for the eigenvalue computation of nonselfadjoint elliptic operators. *East-West J. Numer. Math.*, 8:275–297, 2000.

[67] V. Heuveline and R. Rannacher. A posteriori error control for finite element approximations of elliptic eigenvalue problems. *Adv. Comp. Math.*, 15:107–138, 2001.

[68] P. Jiranek, Z. Strakoš, and M. Vohralik. A posteriori error estimates including algebraic error: computable upper bounds and stopping criteria for iterative solvers. *SIAM J. Sci. Comput.*, 32:1567–1590, 2010.

[69] W. Kahan, B. N. Parlett, and E. Jiang. Residual bounds on approximate eigensystems of nonnormal matrices. *SIAM J. Numer. Anal.*, 19(3):470–484, 1982.

[70] C. Kamm. *A posteriori error estimation in numerical methods for solving self-adjoint eigenvalue problems*. Diplomarbeit TU Berlin, Berlin, 2007.

[71] T. Kato. *A Short Introduction to Perturbation Theory for Linear Operators*. Springer, 1982.

[72] A. V. Knyazev. New estimates for Ritz vectors. *Math. Comp.*, 66:985–995, 1997.

[73] A. V. Knyazev. Toward the optimal preconditioned eigensolver: Locally optimal block preconditioned conjugate gradient method. *SIAM J. Sci. Comput.*, 23:517–541, 2001.

[74] A. V. Knyazev and M. E. Argentati. Principal angles between subspaces in an A-based scalar product: Algorithms and perturbation estimates. *SIAM J. Sci. Comput.*, 23:2009–2041, 2002.

[75] A. V. Knyazev and M. E. Argentati. Majorization for changes in angles between subspaces, Ritz values, and graph Laplacian spectra. *SIAM J. Matrix Anal. Appl.*, 29:15–32, 2006.

[76] A. V. Knyazev and M. E. Argentati. Rayleigh-Ritz majorization error bounds with applications to FEM and subspace iterations. Preprint arXiv:math/0701784v1, 2007. `http://arxiv.org/abs/math.NA/0701784v1`.

[77] A. V. Knyazev and J. E. Osborn. New a priori FEM error estimates for eigenvalues. *SIAM J. Numer. Anal.*, 43:2647–2667, 2006.

[78] L. Komzsik. *Lanczos Method: Evolution and Application.* SIAM Publications, Philadelphia, PA, USA, 2003.

[79] M. G. Larson. A posteriori and a priori error analysis for finite element approximations of self-adjoint elliptic eigenvalue problems. *SIAM J. Numer. Anal.*, 38:608–625, 2000.

[80] S. Larsson and V. Thomée. *Partial Differential Equations with Numerical Methods.* Springer-Verlag, Berlin, 2003.

[81] R. B. Lehoucq, D. C. Sorensen, and C. Yang. *ARPACK User's Guide: Solution of Large-Scale Eigenvalue Problems with Implicitly Restarted Arnoldi Methods.* SIAM Publications, Philadelphia, 1998.

[82] T. Y. Li and N. H. Rhee. Homotopy algorithm for symmetric eigenvalue problems. *Numer. Math.*, 55:265–280, 1989.

[83] T. Y. Li and Z. Zeng. Homotopy-determinant algorithm for solving non-symmetric eigenvalue problems. *Math. Comp.*, 59:483–502, 1992.

[84] T. Y. Li and Z. Zeng. The homotopy continuation algorithm for the real nonsymmetric eigenproblem: Further development and implementation. *SIAM J. Sci. Comp.*, 20:1627–1651, 1999.

[85] T. Y. Li, Z. Zeng, and L. Cong. Solving eigenvalue problems of real nonsymmetric matrices with real homotopies. *SIAM J. Numer. Anal.*, 29:229–248, 1992.

[86] W.-W. Lin and G. Lutzer. An application of the homotopy method to the generalized symmetric eigenvalue problem. *J. Austral. Math. Soc. Ser. B*, 30:230–249, 1988.

[87] S. H. Lui, H. B. Keller, and T. W. C. Kwok. Homotopy method for the large sparse real nonsymmetric eigenvalue problem. *SIAM J. Matrix Anal. Appl.*, 18:312–333, 1997.

[88] D. Mao, L. Shen, and A. Zhou. Adaptive finite element algorithms for eigenvalue problems based on local averaging type a posteriori error estimates. *Adv. Comp. Math.*, 25:135–160, 2006.

[89] MATLAB, Version 7.5.0.336 (R2007b). The MathWorks, inc., 24 Prime Park Way, Natick, MA 01760-1500, USA, 2007.

[90] MATLAB, Version 7.10.0.499 (R2010a). The MathWorks, inc., 24 Prime Park Way, Natick, MA 01760-1500, USA, 2010.

[91] V. Mehrmann and A. Międlar. Adaptive computation of smallest eigenvalues of elliptic partial differential equations. *Numer. Linear Algebra Appl.*, page electronically `DOI: 10.1002/nla.733`, 2010.

[92] K. Neymeyr. A posteriori error estimation for elliptic eigenproblems. *Numer. Alg. Appl.*, 9:263–279, 2002.

[93] R. H. Nochetto. *Adaptive Finite Element Methods for Elliptic PDE.* CNA Summer School, Probabilistic and Analytical Perspectives in Contemporary PDE, 2006. Lecture Notes `http://www-users.math.umd.edu/~rhn/lectures.html`.

[94] J. T. Oden and S. Prudhomme. Goal-oriented error estimation and adaptivity for the finite element method. *Comput. Math. Appl*, 41:735–756, 1999.

[95] B. N. Parlett. *The symmetric eigenvalue problem. Classics in Applied Mathematics.* SIAM Publications, Philadelphia, 1998.

[96] A. Quarteroni and A. Valli. *Numerical Approximation of Partial Differential Equations.* Springer Series in Computational Mathematics. Springer-Verlag, 2008.

[97] R. Rannacher. Error control in finite element computations. In H. Buglak and C. Zenger, editors, *Proc. NATO-Summer School, Error Control and Adaptivity in Scientific Computing*, NATO Science Series, pages 247–278. Kluwer Academic Publ. Dortrecht/Boston/London, 1998.

[98] P. A. Raviart and J. M. Thomas. *Introduction à l'Analyse Numérique des Équations aux Dérivées Partielles, Collection Mathématiques Appliqées pour la Maîtrise.* Masson, Paris, 1983.

[99] B. D. Reddy. *Introductory Functional Analysis.* Texts in Applied Mathematics 27. Springer-Verlag, 1998.

[100] A. Ruhe. The Two-sided Arnoldi Algorithm for Nonsymmetric Eigenvalue Problems. In B. Kågström and A. Ruhe, editors, *Matrix Pencils*, volume 973 of *Lecture Notes in Mathematics*, pages 104–120. Springer-Verlag, Berlin / Heidelberg, 1983.

[101] Y. Saad. *Numerical methods for large eigenvalue problems.* Manchester University Press, Oxford rd, Manchester, UK, 1992.

[102] S. Sauter. Finite elements for elliptic eigenvalue problems in the preasymptotic regime. Preprint 17-2007, Institut für Mathematik der Universität Zürich, Universität Zürich, Winterthurerstraße 190, CH-8057 Zürich, 2007.

[103] A. Schmidt and K. G. Siebert. *Design of Adaptive Finite Element Software: The Finite Element Toolbox ALBERTA.* Springer-Verlag, Berlin, Heidelberg, 2005.

[104] D. C. Sorensen. Numerical methods for large eigenvalue problems. *Acta Numerica*, pages 519–584, 2002. DOI:10.1017/S0962492902000089.

[105] G. W. Stewart and J.-G. Sun. *Matrix Perturbation Theory.* Academic Press, New York, 1990.

[106] Z. Strakoš and J. Liesen. On numerical stability in large scale linear algebraic computations. *Z. Angew. Math. Mech.*, 85(5):307–325, 2005.

[107] G. Strang and G.J. Fix. *An analysis of the finite element method.* Prentice-Hall, Englewood Cliffs, N.J., 1973.

[108] L. N. Trefethen and T. Betcke. Computed eigenmodes of planar regions. *Contemp. Math.*, 412:297–314, 2006.

[109] Ch. Tretter. *Spectral theory of block operator matrices and applications.* Imperial College Press, 2008.

[110] T. Vejchodsky. Computational comparison of the discretization and iteration errors. Preprint 2007-12-18, Institute of Mathematics, AS CR, Prague, Institute of Mathematics, Czech Academy of Sciences, Praha, 2007.

[111] R. Verfürth. *A Review of a Posteriori Error Estimation and Adaptive Mesh-Refinement Techniques.* Wiley and Teubner, New York, Stuttgart, 1996.

[112] H. F. Weinberger. *Variational methods for eigenvalue approximation.* National Science Foundation / Conference Board of the Mathematical Sciences: CBMS-NSF regional conference series in applied mathematics. SIAM Publications, Philadelphia, 1974.

[113] J. H. Wilkinson. *The Algebraic Eigenvalue Problem.* Oxford University Press, Oxford, 1965.

[114] Ch.-T. Wu and H. C. Elman. *Stopping criteria for iterative methods in solving convection-diffusion equations on adaptive meshes*, 2004. Presented in the third international congress of Chinese mathematicians, 2004.

[115] J. Xu and A. Zhou. A two-grid discretization scheme for eigenvalue problem. *Math. Comp.*, 70:17–25, 1999.

[116] A. Zeiser. On the Optimality of the Inexact Inverse Iteration Coupled with Adaptive Finite Element Methods. Preprint 57, DFG-SPP 1324, 2010.

[117] O. C. Zienkiewicz and J. Zhu. A simple error estimator and adaptive procedure for practical engineering analysis. *Int. J. Num. Meth. Eng.*, 24:337–357, 1987.