J. Numer. Math., Vol. 0, No. 0, pp. 1–23 (2011)
DOI 10.1515/ JNUM.2011.000

Adaptive finite element solution of eigenvalue problems: Balancing of discretization and iteration error

Rolf Rannacher[*], Antje Westenberger[*], Winnifried Wollner[*]

Received

Abstract — This paper develops a combined a posteriori analysis for the discretization and iteration errors in the solution of elliptic eigenvalue problems by the finite element method. The emphasis is on the iterative solution of the discretized eigenvalue problem by a Krylov-space method. The underlying theoretical framework is that of the Dual Weighted Residual (DWR) method for goal-oriented error estimation. On the basis of computable a posteriori error estimates the algebraic iteration can be adjusted to the discretization within a successive mesh adaptation process. The functionality of the proposed method is demonstrated by numerical examples.

Keywords: eigenvalue problems, finite element method, mesh adaptation, DWR method, iteration error, stopping criteria

1. Introduction

In this paper, we consider the approximation of the eigenvalue problem of a linear elliptic partial differential operator by an adaptive finite element method, see, e.g., Heuveline & Rannacher [15]. For the solution of such large-scale algebraic eigenvalue problems Krylov space methods are well accepted, see, e.g., Sorensen [24]. We develop an adaptive Arnoldi method for the efficient solution of the algebraic eigenvalue problem resulting from the proposed finite element discretization. It seems natural to stop the iteration of the eigenvalue solver when the error due to the approximate solution of the discrete eigenvalue problem is comparable to the error due to the finite element discretization itself. To this purpose, we derive an a posteriori error estimator which assesses the influences of the discretization and the inexact solution of the discrete eigenvalue problem. This allows us to balance both sources of errors. These errors may be measured in natural norms, e.g., the discretization error for the eigenfunctions in H^1 or L^2. However, measuring the error in global norms may not provide useful bounds for the error in terms of a given functional, a so called *quantity of interest*, e.g., the eigenvalues themselves or certain point values or moments of the corresponding eigenfunctions. In this work, we propose to control both discretization and iteration errors with respect to a general output functional.

[*] Institut für Angewandte Mathematik, Universität Heidelberg, Im Neuenheimer Feld 293/294, D-69120 Heidelberg, Germany.

This approach is based on a posteriori error estimation by dual-weighted residuals as presented in Becker & Rannacher [6] in the context of the so-called “Dual Weighted Residual (DWR)” method. We incorporate the adaptive iteration method into the solution process of a given problem. The estimator derived for the discretization error on the one hand is used as stopping criterion for the algebraic iteration and on the other hand provides the necessary information for the construction of locally refined meshes in order to improve the accuracy of the discretization.

The use of adaptive techniques based on a posteriori estimation of the discretization error is well accepted in the context of finite element discretization of partial differential equations; see, e.g., Verfürth [25] and Bangerth & Rannacher [2]. On the other hand, the idea of balancing the iteration error with the discretization error for the solution of linear solvers, e.g., multigrid methods, has been pursued in only a few publications yet. In the case of the Poisson equation first results have been obtained in Becker, Johnson & Rannacher [4], which was extended to the Stokes equations in Becker [3]. There, the automatic control of the discretization and multigrid errors has been developed with respect to L^2- and energy norms, but the reliability of the proposed adaptive algorithm is verified on uniformly refined meshes only. In Meidner, Rannacher & Vihharev [20] this work was extended to balance the discretization error and the error from the linear solver with respect to an arbitrary target quantity on locally refined meshes.

In the case of eigenvalue and eigenvector computation such an a posteriori error analysis is in principle straightforward. However, due to the inherent nonlinearity of the problem special care has to be taken. For the symmetric case a priori as well as a posteriori estimates for the error in the eigenvalue where obtained by Babuska & Osborn [1], Verfürth [25], and Larson [18]. The non-symmetric case and arbitrary functionals where considered in Heuveline & Rannacher [15]. In all these studies it was assumed that the discretized eigenvalue problems are solved exactly.

On the other hand, when concerned with the computation of the discrete eigenvalues, the iteration is usually stopped once the residual is below a given tolerance, see, e.g., Simonici & Elden [23], Golub, Zhang & Zha [13], Saad [21], Lundström & Elden [19], Jia [16], and Greenbaum [14]. An alternative approach is to stop the iteration once the backward error is below a given tolerance, see, e.g., Chatelin & Fraysse [11] and Scott [22]. In either case it is not clear a priori how to choose the tolerance in the context of eigenvalue problems associated with differential operators. This is because the discretization error is usually the dominant error contribution. However, in order to estimate this error an exact solution of the discrete eigenvalue problem is assumed. In this paper, we will consider the simultaneous estimation of both iteration and discretization error, such that the tolerance for stopping the solution of the discretized eigenvalue problem may be adapted to the size of the discretization error. In this respect, the present analysis provides the natural extension of the results in Heuveline & Rannacher [15] and Meidner, Rannacher & Vihharev [20], respectively, to control the iteration error together with the discretization error in the Galerkin finite element approximation of (linear) elliptic eigenvalue problems. A further issue of this paper is the numerical realization of the adaptive method. We explain implementation details and verify the reliability and the efficiency of the proposed algorithm on locally refined meshes.

We consider the Galerkin finite element approximations of symmetric or non-symmetric elliptic eigenvalue problems. As a prototypical problem, we consider the convection-diffusion problem

$$\mathscr{A}v := -\nabla\cdot(a\nabla v)+b\cdot\nabla v+cv=\lambda v \quad \text{in } \Omega, \quad u=0 \text{ on } \partial\Omega. \tag{1.1}$$

For $b\not\equiv 0$, the governing operator is non-symmetric and may possess non-real eigenvalues. This problem represents only a model case. The results of this paper easily carry over to more complicated situations involving higher-order elliptic operators or systems and other kinds of boundary conditions.

The Galerkin finite element approximation of (1.1) is based on its variational formulation. We seek non-trivial pairs $\{v_h,\lambda_h\}\in V_h\times\mathbb{C}$, $v_h\neq 0$, satisfying

$$a(v_h,\varphi_h)=\lambda_h(v_h,\varphi_h) \quad \forall\varphi_h\in V_h, \tag{1.2}$$

where $a(\cdot,\cdot)$ is the sesquilinear form associated to the differential operator $\mathscr{A}$, and $(\cdot,\cdot)$ is the usual L^2 scalar product. The finite element subspace $V_h\subset V:=H_0^1(\Omega)$ consists of piecewise linear or bilinear functions on decompositions $\mathscr{T}_h$ of the domain $\overline{\Omega}$ into cells T (triangles or quadrilaterals in 2D and tetrahedra or hexahedra in 3D) with diameters $h_T:=\text{diam}\{T\}$. For details on the finite element discretization, we refer to the standard literature, e.g., Carey & Oden [9], Ciarlet [12], and Brenner & Scott [8].

In the following analysis a “dual” eigenvalue problem $\mathscr{A}^*v^*=\lambda^*v^*$ is associated to the general non-symmetric “primal” eigenvalue problem (1.1), together with its discrete counterpart

$$a(\psi_h,v_h^*)=\overline{\lambda}_h^*(\psi_h,v_h^*) \quad \forall\psi_h\in V_h. \tag{1.3}$$

With this, we can recall the discretization error estimates from Heuveline & Rannacher [15] and extend these results to incorporate the iteration error of the Arnoldi method.

The outline of the article is as follows. In Section 2, we will describe the problem class under consideration. We then introduce the corresponding finite element discretization, which is followed by a detailed discussion of the iterative solution of the discretized eigenvalue problem. Then, in Section 3, we describe the a posteriori error estimation. We begin by considering the overall error due to discretization and inexact solution of the discrete eigenvalue problem. After that, we develop the splitting of the derived error estimator into a discretization and an iteration component. Finally, in Section 4, we discuss how the estimates of Section 3 can be used for an adaptive computation of eigenvalues and demonstrate the behavior of the proposed algorithm by some numerical examples of moderate complexity.

2. Preliminaries

2.1. Eigenvalue problem and its discretization

Let $\mathscr{A}$ be a second-order uniformly elliptic operator defined on a bounded domain $\Omega \subset \mathbb{R}^d$. The classical formulation of the eigenvalue problem for this operator is

$$\mathscr{A} v = \lambda v \quad \text{in } \Omega, \quad v = 0 \text{ on } \partial\Omega. \tag{2.1}$$

The variational formulation of (2.1) seeks pairs $\{v, \lambda\} \in V \times \mathbb{C}$, satisfying

$$a(v, \psi) = \lambda\,(v, \psi) \quad \forall \psi \in V, \quad \|v\| = 1, \tag{2.2}$$

where $a(\cdot,\cdot)$ is the sesquilinear form generated by the operator $\mathscr{A}$, $(\cdot,\cdot)$ is the (complex) L^2-scalar product on Ω with corresponding norm $\|\cdot\|$, and $V := H^1_0(\Omega)$ is the usual first-order (complex) Sobolev space with norm $\|\cdot\|_V$. The sesquilinear form $a(\cdot,\cdot)$ is assumed to be bounded and V-elliptic, that is

$$|a(\varphi, \psi)| \leqslant \alpha \|\varphi\|_V \,\|\psi\|_V, \quad \varphi, \psi \in V, \tag{2.3}$$

$$\gamma \|\varphi\|_V^2 \leqslant |a(\varphi, \varphi)| + \beta \|\varphi\|^2, \quad \varphi \in V, \tag{2.4}$$

with certain constants $\alpha, \gamma > 0$, and $\beta \geqslant 0$. For simplicity, we assume in the following that (2.3) holds with $\beta = 0$, such that zero is not an eigenvalue of $\mathscr{A}$. Since the embedding $V \hookrightarrow L^2(\Omega)$ is compact, the classical Riesz-Schauder theory applies (see, e.g., Kato [17]). Hence, the primal as well as the dual eigenvalue problem possess countable infinite spectral sets $\Sigma(\mathscr{A}) := \{\lambda_i\}_{i=1}^\infty \subset \mathbb{C}$ and $\Sigma(\mathscr{A}^*) := \{\lambda_i^*\}_{i=1}^\infty \subset \mathbb{C}$ of isolated eigenvalues with finite (algebraic) multiplicities which have no finite accumulation point. Further, one easily sees that $\lambda_i^* = \overline{\lambda}_i$. The "algebraic" and "geometric" multiplicities of an eigenvalue λ are denoted by σ_λ and ρ_λ, respectively. Further, its "ascent" α_λ is the smallest integer such that $\mathrm{Ker}\{(\mathscr{A} - \lambda I)^{\alpha_\lambda + 1}\} = \mathrm{Ker}\{(\mathscr{A} - \lambda I)^{\alpha_\lambda}\}$. The eigenvalue λ is called "non-defective" if $\alpha_\lambda = 1$, which is related to $\sigma_\lambda = \rho_\lambda$.

For simplicity, we will assume throughout this paper that all considered eigenvalues are simple (and therefore "non-defective"). Then, the "adjoint" (or "dual") eigenvalue problem associated to (2.2) seeks pairs $\{v^*, \lambda^*\} \in V \times \mathbb{C}$, satisfying

$$a(\psi, v^*) = \overline{\lambda}^*\,(\psi, v^*) \quad \forall \psi \in V, \quad (v, v^*) = 1. \tag{2.5}$$

The eigenvalue problem (2.2) and (2.5) are approximated by a standard Galerkin finite element method. Let $V_h \subset V$ be a (finite dimensional) finite element space where $h \in \mathbb{R}_+$ is a discretization parameter. For our implementation, we used H^1-conforming Q_1 finite elements. In order to ease the mesh refinement, we allow the cells to have nodes which lie on midpoints of edges or faces of neighboring cells. But at most one of such *hanging nodes* is permitted per edge or face. Consideration of meshes with hanging nodes requires additional care. These irregular nodes do not carry degrees of freedom as the corresponding values of the finite element function are determined by linear interpolation of its values at neighboring regular nodes. We refer, e.g., to Carey & Oden [9] for implementation details.

Then, the approximate eigenvalue problems read as follows. Find non-trivial pairs $\{v_h, \lambda_h\}, \{v_h^*, \lambda_h^*\} \in V_h \times \mathbb{C}$ satisfying

$$a(v_h, \varphi_h) = \lambda_h\,(v_h, \varphi_h) \quad \forall \varphi_h \in V_h, \quad \|v_h\| = 1, \tag{2.6}$$

$$a(\varphi_h, v_h^*) = \bar{\lambda}_h^*\,(\varphi_h, v_h^*) \quad \forall \varphi_h \in V_h, \quad (v_h, v_h^*) = 1. \tag{2.7}$$

The normalization condition $(v_h, v_h^*) = 1$ in the adjoint eigenvalue problem (2.7) is admissible, if the discrete eigenvalue λ_h is also simple (and therefore non-defective). In algebraic notation (2.6) and (2.7) take the form of generalized eigenvalue problems

$$\mathbf{A}_h v_h = \lambda_h \mathbf{M}_h v_h, \qquad \mathbf{A}_h^H v_h^* = \lambda_h^* \mathbf{M}_h v_h^*, \tag{2.8}$$

where $\mathbf{A}_h$ is the "stiffness matrix" (generally non-hermitian) and $\mathbf{M}_h$ is the hermitian and positive-definite "mass matrix". Again, the approximate primal and dual eigenvalues are related to each other by $\lambda_h^* = \bar{\lambda}_h$. Here, we do not differentiate between the finite element function v_h and its coordinate vector in our notation.

2.2. Solution of the discrete eigenvalue problem

For solving (2.8), one could consider any suitable solution method for this problem class. However, as we are especially interested in the computation of the smallest eigenvalues of the real matrix $\mathbf{A}_h$, we consider the following reformulation of (2.8). Determine v_h, λ_h such that

$$\mathbf{A}_h^{-1} \mathbf{M}_h v_h = \lambda_h^{-1} v_h. \tag{2.9}$$

We consider an Arnoldi algorithm, which is restated here in order to introduce some of our notation.

Algorithm 2.1 Arnoldi-algorithm

```
Choose $\tilde{v}_1 \neq 0$ and set $v_1 := \|\tilde{v}_1\|^{-1}\tilde{v}_1,\ \ h_{i,j} := 0\ \ (i,j = 1,\dots,k)$.
for $j = 1$ to $k$ do
    $w_j := \mathbf{A}_h^{-1}\mathbf{M}_h v_j$,
    for $i = 1$ to $j$ do
        $h_{ij} := (v_i, w_j)$,
        $w_j := w_j - h_{ij} v_i$,

    $h_{j+1,j} := \|w_j\|$,
    if $h_{j+1,j} < TOL$ then
        break

    Set $v_{j+1} := h_{j+1,j}^{-1} w_j$.
```

After k iterations of the algorithm, we obtain an upper Hessenberg matrix $\mathbf{H}_k = (h_{i,j})_{i,j=1,\dots,k} \in \mathbb{R}^{k\times k}$ together with a matrix $\mathbf{V}_k := (v_1,\dots,v_k) \in \mathbb{R}^{n\times k}$ such that

$$\mathbf{H}_k = \mathbf{V}_k^T \mathbf{A}_h^{-1} \mathbf{M}_h \mathbf{V}_k.$$

Then, the approximate eigenvalues $\lambda_{h,k}$ are given by the k-dimensional eigenvalue problem

$$\mathbf{H}_k \tilde{v}_{h,k} = \lambda_{h,k}^{-1} \tilde{v}_{h,k},$$

which can be solved by a QR-method in $O(k^2)$ operations. In order to obtain an eigenvector approximation, we consider

$$v_{h,k} := \mathbf{V}_k \tilde{v}_{h,k}.$$

In order to assert that the effort of computing the eigenvalues and eigenvectors of $\mathbf{H}_k$ does not exceed the effort of applying $\mathbf{A}_h^{-1}\mathbf{M}_h$, we have to ensure that $k^2 \ll n$. This is because we can solve $w_j := \mathbf{A}_h^{-1}\mathbf{M}_h v_j$ in $O(n)$ operations using a multigrid method. This can be achieved using restarted versions of Algorithm 2.1, see, e.g., Sorensen [24].

3. A posteriori error estimation

For developing an a posteriori error analysis for the approximation scheme described above, we follow the DWR approach described in Becker & Rannacher [6] and Heuveline & Rannacher [15]. The first step is the reformulation of the continuous and discrete eigenvalue problems as nonlinear variational equations. To this end, we introduce the semi-linear form $A\colon V\times\mathbb{C}\times V\times\mathbb{C}\to\mathbb{C}$ given by

$$A(v,\lambda)(\delta v,\chi) = -a(v,\delta v) + \lambda(v,\delta v) + \overline{\chi}\left(\|v\|^2 - 1\right).$$

We see that A is Fréchet differentiable and its derivative A' with respect to v and λ in the point (v,λ,w,μ) in direction $(\delta v,\chi)$ takes the form

$$A'(v,\lambda)(\delta v,\chi,w,\mu) = -a(\delta v,w) + \lambda(\delta v,w) + \chi(v,w) + 2\overline{\mu}\,\mathrm{Re}\big((\delta v,v)\big).$$

Obviously, (2.2) is equivalent to

$$A(v,\lambda)(\delta v,\chi) = 0 \quad \forall(\delta v,\chi)\in V\times\mathbb{C}, \tag{3.1}$$

and its discrete analog (2.6) is equivalent to

$$A(v_h,\lambda_h)(\delta v,\chi) = 0 \quad \forall(\delta v,\chi)\in V_h\times\mathbb{C}. \tag{3.2}$$

Next, in order to control the error in this approximation scheme, we choose an arbitrary three times directionally differentiable functional $J\colon V\times\mathbb{C}\to\mathbb{C}$. The associated “dual problem” is formulated by

$$A'(v,\lambda)(\delta v,\chi,w,\mu) = J'(v,\lambda)(\delta v,\chi) \quad \forall(\delta v,\chi)\in V\times\mathbb{C}, \tag{3.3}$$

and its discrete analog by

$$A'(v_h,\lambda_h)(\delta v,\chi,w_h,\mu_h) = J'(v_h,\lambda_h)(\delta v,\chi) \quad \forall(\delta v,\chi) \in V_h \times \mathbb{C}. \tag{3.4}$$

The solvability of the dual equations obviously depends on the choice of J. We will consider here the case $J(v,\lambda) = \lambda \|v\|^2$, which leads to estimates for the eigenvalue error. In this particular case (3.3) is equivalent to (2.5) and (3.4) is equivalent to (2.7).

3.1. Estimation of the error

We will now come to the derivation of an error representation between a solution of (3.1) and an inexact solution of the discrete problem (3.2) with respect to the given functional J. To this end, we extend the result of Becker & Rannacher [6] to the case of an inexact discrete solution. Let $u = (v,\lambda)$, $z = (w,\mu)$, $\varphi = (\delta v,\chi) \in V \times \mathbb{C}$ be given. We define the corresponding primal and dual residuals associated to the variational problems (3.1) and (3.3) by

$$\begin{aligned} \rho(u)(\varphi) &:= -A(v,\lambda)(\delta v,\chi), \\ \rho^*(u,z)(\varphi) &:= J'(v,\lambda)(\delta v,\chi) - A'(v,\lambda)(\delta v,\chi,w,\mu). \end{aligned}$$

We begin our discussion with the following perturbation result.

Proposition 3.1. *Let* $u = (v,\lambda) \in V \times \mathbb{C}$ *be a solution to* (3.1) *and let* $z = (w,\mu) \in V \times \mathbb{C}$ *be the corresponding dual solution* (3.3)*. Then, for any* $\tilde{u} = (\tilde{v},\tilde{\lambda})$, $\tilde{z} = (\tilde{w},\tilde{\mu}) \in V \times \mathbb{C}$ *the following error representation holds:*

$$\begin{aligned} J(u) - J(\tilde{u}) = \tfrac{1}{2}\rho(\tilde{u})(z-\tilde{z}) &+ \tfrac{1}{2}\rho^*(\tilde{u},\tilde{z})(u-\tilde{u}) \\ &+ \rho(\tilde{u})(\tilde{z}) + R(u,z,\tilde{u},\tilde{z}). \end{aligned} \tag{3.5}$$

Here, the remainder term R *is given by*

$$\begin{aligned} R(u,z,\tilde{u},\tilde{z}) := \tfrac{1}{2}\int_0^1 \big\{ & J'''(\tilde{u}+se;e,e,e) - A'''(\tilde{u}+se;\tilde{z}+se^*;e,e,e) \\ & - 3A''(\tilde{u}+se;e,e,e^*)\big\}\, s(s-1)\,ds, \end{aligned} \tag{3.6}$$

where $e := u - \tilde{u}$ *and* $e^* := z - \tilde{z}$.

Proof. The proof only uses elementary arguments. Noting that u solves (3.1), we obtain

$$\begin{aligned} J(u) - J(\tilde{u}) &= J(u) - A(u)(z) - J(\tilde{u}) + A(\tilde{u})(\tilde{z}) - A(\tilde{u})(\tilde{z}) \\ &= J(u) - A(u)(z) - J(\tilde{u}) + A(\tilde{u})(\tilde{z}) + \rho(\tilde{u})(\tilde{z}). \end{aligned}$$

In order to estimate the first four terms on the right-hand side, we employ the main theorem of calculus and evaluate the integral using the trapezoidal rule to obtain

$$\begin{aligned}
&J(u)-J(\tilde{u})-\rho(\tilde{u})(\tilde{z})\\
&\quad= \int_0^1 J'(\tilde{u}+se)(e)-A'(\tilde{u}+se)(e,\tilde{z}+se^*)-A(\tilde{u}+se)(e^*)\,ds\\
&\quad= \tfrac{1}{2}\rho(u)(z-\tilde{z})+\tfrac{1}{2}\rho^*(u,z)(u-\tilde{u})+\tfrac{1}{2}\rho(\tilde{u})(z-\tilde{z})+\tfrac{1}{2}\rho^*(\tilde{u},\tilde{z})(u-\tilde{u})+R.
\end{aligned}$$

Then, since u and z solve (3.1) and (3.3) the first two terms on the right-hand side vanish. This proves the assertion. □

From this, one immediately derives the results of Heuveline & Rannacher [15] by considering $\tilde{u}$ to be the solution of the discrete problem (3.2).

Corollary 3.1. *Let $u=(v,\lambda)\in V\times\mathbb{C}$ be a solution to* (3.1) *and let $z=(w,\mu)\in V\times\mathbb{C}$ be the corresponding dual solution* (3.3). *Let further $u_h=(v_h,\lambda_h)\in V_h\times\mathbb{C}$ and $z_h=(w_h,\mu_h)\in V_h\times\mathbb{C}$ be the corresponding Galerkin approximations* (3.2) *and* (3.4). *Then, for any $\tilde{u}_h=(\tilde{v}_h,\tilde{\lambda}_h)$, $\tilde{z}_h=(\tilde{w}_h,\tilde{\mu}_h)\in V_h\times\mathbb{C}$ the following error representation holds:*

$$J(u)-J(u_h)=\tfrac{1}{2}\rho(u_h)(z-\tilde{z}_h)+\tfrac{1}{2}\rho^*(u_h,z_h)(u-\tilde{u}_h)+R(u,z,u_h,z_h). \tag{3.7}$$

Here, the remainder term R is given by

$$\begin{aligned}
R(u,z,u_h,z_h):=\tfrac{1}{2}\int_0^1 \big\{&J'''(u_h+se_h;e_h,e_h,e_h)\\
&-A'''(u_h+se_h;z_h+se_h^*;e_h,e_h,e_h)\\
&-3A''(u_h+se_h;e_h,e_h,e_h^*)\big\}\,s(s-1)\,ds\,,
\end{aligned}\tag{3.8}$$

where $e_h:=u-u_h$ and $e_h^*:=z-z_h$.

Proof. The result directly follows from Proposition 3.1 by noting that $V_h\subset V$ and that

$$\begin{aligned}
\rho(u_h)(\varphi_h)&=0 \quad \forall\,\varphi_h\in V_h\times\mathbb{C},\\
\rho^*(u_h,z_h)(\varphi_h)&=0 \quad \forall\,\varphi_h\in V_h\times\mathbb{C}.
\end{aligned}$$

□

Before we continue our discussion, we will restate Proposition 3.1 and Corollary 3.1 for the case of the error in the eigenvalue.

Corollary 3.2. *Let* $u = (v,\lambda) \in V \times \mathbb{C}$ *be a solution to* (2.2) *and let* $z = (w,\mu) \in V \times \mathbb{C}$ *be the corresponding dual solution* (2.5). *Then, for any* $\tilde{u} = (\tilde{v},\tilde{\lambda})$, $\tilde{z} = (\tilde{w},\tilde{\mu}) \in V \times \mathbb{C}$, *with* $\|\tilde{v}\| = (\tilde{v},\tilde{w}) = 1$, *the following error representation holds:*

$$\begin{aligned}\lambda - \tilde{\lambda} = {} & \tfrac{1}{2}\left\{a(\tilde{v}, w-\tilde{w}) - \tilde{\lambda}(\tilde{v}, w-\tilde{w})\right\} + \tfrac{1}{2}\left\{(a(v-\tilde{v},\tilde{w}) - \tilde{\mu}(v-\tilde{v},\tilde{w})\right\} \\ & + a(\tilde{v},\tilde{w}) - \tilde{\lambda}(\tilde{v},\tilde{w}) + \tfrac{1}{2}(\lambda-\tilde{\lambda})(v-\tilde{v}, w-\tilde{w}).\end{aligned} \tag{3.9}$$

Proof. The corollary directly follows from Proposition 3.1 for the choice of $J(v,\lambda) := \lambda\|v\|^2$. □

Then, by choosing $\tilde{u}$ as the solution to the discrete eigenvalue problem, we obtain the following special form of Corollary 3.2.

Corollary 3.3. *Let* $u = (v,\lambda) \in V \times \mathbb{C}$ *be a solution to* (2.2) *and let* $z = (v^*,\lambda^*) \in V \times \mathbb{C}$ *be the corresponding dual solution* (2.5). *Let further* $u_h = (v_h,\lambda_h) \in V_h \times \mathbb{C}$ *and* $z_h = (v_h^*,\lambda_h^*) \in V_h \times \mathbb{C}$ *be the corresponding Galerkin approximations* (2.6) *and* (2.7). *Then, for any* $\tilde{u}_h = (\tilde{v}_h,\tilde{\lambda}_h)$, $\tilde{z}_h = (\tilde{v}_h^*,\tilde{\lambda}_h^*) \in V_h \times \mathbb{C}$ *the following error representation holds:*

$$\begin{aligned}\lambda - \lambda_h = {} & \tfrac{1}{2}\left\{a(v_h, v^*-\tilde{v}_h^*) - \lambda_h(v_h, v^*-\tilde{v}_h^*)\right\} \\ & + \tfrac{1}{2}\left\{a(v-\tilde{v}_h, v_h^*) - \overline{\lambda}_h^*(v-\tilde{v}, v_h^*)\right\} + \tfrac{1}{2}(\lambda-\lambda_h)(v-v_h, v^*-v_h^*).\end{aligned} \tag{3.10}$$

Proof. The corollary directly follows from Corollary 3.1 for the choice of $J(v,\lambda) := \lambda\|v\|^2$. □

Remark 3.1. We remark, that the results of Corollary 3.2 and Corollary 3.3, once their structure is known, could also be obtained by direct calculation without the use of the abstract results from Proposition 3.1 and Corollary 3.1. However, this is no longer the case for an arbitrary functional $J(v,\lambda)$.

We cannot directly evaluate the residual terms in the error representation 3.10 since the errors $u-\tilde{u}_h$ and $z-\tilde{z}_h$ are unknown. However, these quantities are local, for instance $\tilde{u}_h = i_h(u)$ the nodal interpolation is a feasible test function. There are several heuristic techniques to estimate this difference, see for instance Becker & Rannacher [6]. Here, we construct an approximation using a biquadratic interpolation on a mesh that has local mesh size $2h$, e.g., we reconstruct $v - \tilde{v}_h \approx I_{2h}^{(2)} v_h - v_h$, which has generally been observed to work well in the context of "goal-oriented" a posteriori error estimation, see Bangerth & Rannacher [2].

In order to use this error estimator for guiding mesh refinement, we have to localize it to cell-wise or node-wise contributions. A direct splitting of a term like $\rho(u_h)(I_{2h}^{(2)} w_h - w_h)$ into its cell contributions, in general, leads to local contributions of wrong order (overestimation) due to oscillatory behavior of the residual terms, see Carstensen & Verfürth [10]. To overcome this problem, one may integrate the residual terms by parts, see Becker & Rannacher [6] or Heuveline & Rannacher [15], or use a filtering operator, see Braack& Ern [7] for details. A more concrete form of the resulting a posteriori error bounds will be stated in Section 4.1, below.

3.2. Separating discretization and iteration errors

In order to split the influence of the iteration error and the discretization error, we denote the k-th iterate of the eigenvalue computation by $u_{h,k}$ and the k-th iterate of the dual problem by $z_{h,k}$, see Section 2.2. Then, $u_{h,k} \to u_h$ and $z_{h,k} \to z_h$ as $k \to \infty$, and hence

$$\rho(u_{h,k})(z - z_{h,k}) \to \rho(u_h)(z - z_h) \qquad (k \to \infty), \tag{3.11a}$$

$$\rho^*(u_{h,k}, z_{h,k})(u - u_{h,k}) \to \rho^*(u_h, z_h)(u - u_h) \qquad (k \to \infty), \tag{3.11b}$$

$$R(u, z, u_{h,k}, z_{h,k}) \to R(u, z, u_h, z_h) \qquad (k \to \infty). \tag{3.11c}$$

Unfortunately, for reasonable iteration numbers k the differences may be rather large and therefore, one cannot simply replace the limit by these approximations. This means that we may not consider $\rho(u_{h,k})(z_{h,k})$ as an indicator for the iteration error as might be suspected from the difference of (3.5) and (3.7). Instead one has to carefully monitor the differences in (3.11). For clarity of the presentation, we will focus our attention on the case of estimating the error in a given eigenvalue λ.

For the convenience of the reader, we will now develop the precise form of the residual terms derived in the previous section. Let $u = (v, \lambda)$ and $z = (v^*, \lambda^*)$ be given with $\|v\| = (v, v^*) = 1$ and let $\varphi = (\delta v, \delta\lambda)$ and $\psi = (\delta v^*, \delta\lambda^*)$ be arbitrary. Using the notation of Proposition 3.5 and Corollary 3.1, we have

$$\begin{aligned} \rho(u)(\varphi) &= \rho(v, \lambda)(\delta v) = a(v, \delta v) - \lambda(v, \delta v), \\ \rho^*(u, z)(\varphi) &= \rho^*(v, \lambda, v^*, \lambda^*)(\delta v) = a(\delta v, v^*) - \overline{\lambda^*}(\delta v, v^*), \\ R(u, z, \varphi, \psi) &= R(\delta v, \delta\lambda, \delta v^*) = \tfrac{1}{2}\delta\lambda(\delta v, \delta v^*). \end{aligned}$$

Let now $u = (v, \lambda)$ be a solution to (2.2) and $z = (v^*, \lambda^*)$ be the corresponding dual solution (2.5), e.g., $\lambda = \overline{\lambda^*}$. Let $u_h = (v_h, \lambda_h)$ be a Galerkin approximation given by (2.6) and $z_h = (v_h^*, \lambda_h^*)$ be the corresponding discrete dual solution given by (2.7). Further, let $u_{h,k} = (v_{h,k}, \lambda_{h,k})$ and $z_{h,k} = (v_{h,k}^*, \lambda_{h,k}^*)$ be the approximations to u_h and z_h at the k-th iteration as defined in Section 2.2.

In order to balance the error contributions, we will consider the discretization error indicator. We define

$$\eta := \tfrac{1}{2}\rho(v_h, \lambda_h)(v^* - v_h^*), \tag{3.12a}$$

$$\eta^* := \tfrac{1}{2}\rho^*(v_h, \lambda_h, v_h^*, \lambda_h^*)(v - v_h), \tag{3.12b}$$

$$\sigma := \tfrac{1}{2}(v - v_h, v^* - v_h^*). \tag{3.12c}$$

With these quantities the discretization error given by Corollary 3.3 reads as follows:

$$(\lambda - \lambda_h)(1 - \sigma) = \eta + \eta^*,$$

provided $\sigma \ll 1$. As observed in prior work (see, e.g., Becker & Rannacher [6]) approximation of the errors $v - v_h$ and $v^* - v_h^*$ using a higher order interpolation on a coarser grid leads to rather good guesses. Hence, we use

$$\begin{aligned}
\eta_h &= \tfrac{1}{2}\rho(v_h, \lambda_h)(I_{2h}^{(2)} v_h^* - v_h^*),\\
\eta_h^* &= \tfrac{1}{2}\rho^*(v_h, \lambda_h, v_h^*, \lambda_h^*)(I_{2h}^{(2)} v_h - v_h),\\
\sigma_h &= \tfrac{1}{2}(I_{2h}^{(2)} v_h - v_h, I_{2h}^{(2)} v_h^* - v_h^*),
\end{aligned}$$

as an approximation to these quantities. This means, we consider the following approximate representation of the error in the eigenvalues

$$(\lambda - \lambda_h)(1 - \sigma_h) \approx \eta_h + \eta_h^*, \tag{3.13}$$

as long as $\sigma_h \ll 1$. However, we cannot directly evaluate these quantities as we will only have the iterates $u_{h,k}$ and $z_{h,k}$ at our disposal, and not the exact discrete solutions u_h and z_h. Hence, instead of (3.12), we have to consider the following quantities:

$$\begin{aligned}
\eta_k &= \tfrac{1}{2}\rho(v_{h,k}, \lambda_{h,k})(v^* - v_{h,k}^*), && \text{(3.14a)}\\
\eta_k^* &= \tfrac{1}{2}\rho^*(v_{h,k}, \lambda_h, v_{h,k}^*, \lambda_{h,k}^*)(v - v_{h,k}), && \text{(3.14b)}\\
\sigma_k &= \tfrac{1}{2}(v - v_{h,k}, v^* - v_{h,k}^*). && \text{(3.14c)}
\end{aligned}$$

However, these quantities contain both discretization and iteration errors. Therefore, we have to monitor the errors $\eta - \eta_k$, $\eta^* - \eta_k^*$, and $\sigma - \sigma_k$. Only once they are small, compared to the values of η_k, η_k^*, and σ_k, they may be used as discretization error indicators. By a simple computation, we obtain

$$\begin{aligned}
\sigma - \sigma_k &= -\tfrac{1}{2}\big\{(v, v_h^* - v_{h,k}^*) + (v_h - v_{h,k}, v^*)\big\}, && \text{(3.15a)}\\
\eta - \eta_k &= \tfrac{1}{2}\big\{(a(v_h - v_{h,k}, v^* - v_{h,k}^*) - (\lambda_h v_h - \lambda_{h,k} v_{h,k}, v^* - v_{h,k}^*)\big\}, && \text{(3.15b)}\\
\eta^* - \eta_k^* &= \tfrac{1}{2}\big\{a(v - v_{h,k}, v_h^* - v_{h,h}^*) - (v - v_{h,k}, \lambda_h^* v_h^* - \lambda_{h,k}^* v_{h,k}^*)\big\}. && \text{(3.15c)}
\end{aligned}$$

In order to evaluate these quantities, we approximate the difference $v_h - v_{h,k}$ using two consecutive iterates, $v_h - v_{h,k} \approx v_{h,k+1} - v_{h,k}$, and analogously for the other discrete iteration errors. In order to estimate the error $v - v_{h,k}$, we may not consider the approximation $v - v_{h,k} \approx I_{2h}^{(2)} v_{h,k} - v_{h,k}$ because this will not work for $v_{h,k}$ far away from v_h. Instead we consider

$$v - v_{h,k} \approx I_{2h}^{(2)} v_{h,k+1} - v_{h,k},$$

in order to account for the convergence with respect to k. Hence, we define the following approximations to the error in the discretization error indicators

$$\sigma-\sigma_k \approx \delta^{\sigma}_{h,k} := \tfrac{1}{2}\big\{(v_{h,k+1},v^*_{h,k+1}-v^*_{h,k})+(v_{h,k+1}-v_{h,k},v^*_{h,k+1})\big\}, \tag{3.16a}$$

$$\begin{aligned}\eta-\eta_k \approx \delta^{\eta}_{h,k} := \tfrac{1}{2}\big\{&a(v_{h,k+1}-v_{h,k},I^{(2)}_{2h}v^*_{h,k+1}-v^*_{h,k})\\ &-(\lambda_{h,k+1}v_{h,k+1}-\lambda_{h,k}v_{h,k},I^{(2)}_{2h}v^*_{h,k+1}-v^*_{h,k})\big\},\end{aligned} \tag{3.16b}$$

$$\begin{aligned}\eta^*-\eta^*_k \approx \delta^{\eta^*}_{h,k} := \tfrac{1}{2}\big\{&a(I^{(2)}_{2h}v_{h,k+1}-v_{h,k},v^*_{h,k+1}-v^*_{h,k})\\ &-(I^{(2)}_{2h}v_{h,k+1}-v_{h,k},\lambda^*_{h,k+1}v^*_{h,k+1}-\lambda^*_{h,k}v^*_{h,k})\big\}.\end{aligned} \tag{3.16c}$$

Finally, we define approximations for the quantities (3.14) by

$$\begin{aligned}\eta_{h,k} &= \tfrac{1}{2}\rho(v_{h,k},\lambda_{h,k})(I^{(2)}_{2h}v^*_{h,k+1}-v^*_{h,k}),\\ \eta^*_{h,k} &= \tfrac{1}{2}\rho^*(v_{h,k},\lambda_h,v^*_{h,k},\lambda^*_{h,k})(I^{(2)}_{2h}v_{h,k+1}-v_{h,k}),\\ \sigma_{h,k} &= \tfrac{1}{2}(I^{(2)}_{2h}v_{h,k+1}-v_{h,k},I^{(2)}_{2h}v^*_{h,k+1}-v^*_{h,k}).\end{aligned}$$

Hence, using additionally the notion

$$\eta^{\text{it}}_k := \rho(v_{h,k},\lambda_{h,k})(v^*_{h,k}),$$

for the estimator of the iteration error estimator, we obtain the following computable error estimator for the full eigenvalue error

$$\lambda-\lambda_{h,k} \approx \frac{1}{1-\sigma_{h,k}}\big\{\eta_{h,k}+\eta^*_{h,k}+\eta^{\text{it}}_k\big\}, \tag{3.17}$$

provided that $|\sigma_{h,k}| \ll 1$ and $\delta^{\sigma}_{h,k}$, $\delta^{\eta}_{h,k}$, and $\delta^{\eta^*}_{h,k}$ are sufficiently small with respect to $\sigma_{h,k}$, $\eta_{h,k}$, and $\eta^*_{h,k}$. The overall computation including the monitoring of the error quantities $\delta^{\sigma}_{h,k}$, $\delta^{\eta}_{h,k}$, and $\delta^{\eta^*}_{h,k}$ is now rather complex and will be described in detail in the next section, see Algorithm 4.1.

Numerical Example. We will now consider the effectivity of our estimates for the various error terms for a simple model problem with known eigenvalues and functions. Thereto let $\Omega=(0,1)\times(0,1/2)$. We consider the following eigenvalue problem of finding $v\in H^1_0(\Omega)$ and $\lambda\in\mathbb{C}$ such that, for given $\mathbf{b}=(b_1,b_2)^T$,

$$-\nu\Delta v+\mathbf{b}\cdot\nabla v=\lambda v. \tag{3.18}$$

The solutions are given by $\lambda=|\mathbf{b}|^{\mathbf{2}}(\mathbf{4}\nu)^{-\mathbf{1}}+\nu\pi^{\mathbf{2}}(\gamma_{\mathbf{1}}^{\mathbf{2}}+\mathbf{4}\gamma_{\mathbf{2}}^{\mathbf{2}})$ with

$$v(x_1,x_2)=\exp\big(\tfrac{\mathbf{b}\cdot\mathbf{x}}{2\nu}\big)\sin(\gamma_1\pi x_1)\sin(2\gamma_2\pi x_2),\quad v^*(x_1,x_2)=\exp\big(-\tfrac{\mathbf{b}\cdot\mathbf{x}}{\nu}\big)v(x_1,x_2),$$

with arbitrary $\gamma_i \in \mathbb{N}$. In particular by the choice of the domain the four eigenvalues with smallest real part are simple. We will now consider the behavior of (3.15) and see how well it is recovered by (3.16). For this, we choose $b_1 = b_2 = 1$ and $\nu = 0.1$. Then, we consider the approximation to the smallest eigenvalue on a triangulation obtained by five global refinements of the domain. The obtained results are shown in Table 1. First, we see, that the errors between σ_h and $\sigma_{h,k}$ and also the estimate for $\eta_h + \eta_h^*$ is captured very well by our estimate, which is reflected by the fact that

$$I_{\text{eff}}^{\sigma} := \frac{|\delta_{h,k}^{\sigma}|}{|\sigma - \sigma_k|} \approx 1, \qquad I_{\text{eff}}^{\eta} := \frac{|\delta_{h,k}^{\eta} + \delta_{h,k}^{\eta^*}|}{|\eta - \eta_k + \eta^* - \eta_k^*|} \approx 1.$$

Table 1: Efficency of the estimate (3.17) for the 1st eigenvalue of (3.18) for $\nu = 0.1$

k	$\lvert\sigma_k\rvert$	$\lvert\delta_{h,k}^{\sigma}\rvert$	I_{eff}^{σ}	$\lvert\eta_k + \eta_k^*\rvert$	$\lvert\delta_{h,k}^{\eta} + \delta_{h,k}^{\eta^*}\rvert$	I_{eff}^{η}	$\lvert\eta_k^{\text{it}}\rvert$	$\lvert\lambda - \lambda_{h,k}\rvert$
4	$4.0e-1$	$4.0e-1$	0.9	$7.0e-1$	$2.0e+1$	34	$5.0e-1$	$2.0e+0$
6	$7.0e-3$	$9.0e-3$	1.4	$1.0e-0$	$1.0e+0$	1.1	$1.0e+0$	$4.0e-1$
8	$1.0e-4$	$2.0e-4$	1.5	$5.0e-4$	$2.0e-2$	1.4	$5.0e-2$	$5.0e-2$
10	$3.0e-7$	$2.0e-6$	0.5	$1.0e-2$	$8.0e-8$	0.4	$2.0e-3$	$1.0e-2$

We remark that the large overestimation indicated by the large value of I_{eff}^{η} for $k \leqslant 4$ is of no concern, as in this case both σ_k and $\delta_{h,k}^{\sigma}$ are still close to one, and, in view of (3.13), the quantity $\eta + \eta^*$ does not provide any useful information. Finally, we remark, that after only ten steps of the algorithm (initialized by a random vector), we obtain the required eigenvalue up to discretization accuracy, whereas the discrete residual η_k^{it} is still of size 10^{-3}. This demonstrates the potential advantage of balancing discretization and iteration error.

Next, for comparison, we consider the approximation of the 4th eigenvalue where we chose $\nu = 1$ in order to have a comparable size of the discretization error on the same grid. The results are shown in Table 2. Once again, we see that the estimate for the error in σ is of good quality. Further the estimates for $\eta + \eta^*$ are also of good quality once $|\sigma_k| + |\delta_{h,k}^{\sigma}| \ll 1$. In addition, we see again that the discretization error is dominant while the discrete residual η_k^{it} is still of the size 10^{-3}. In these simple cases it would require about twice the number of iterations to reach the machine accuracy $\eta_k^{\text{it}} \approx 10^{-12}$.

Table 2: Efficency of the estimate (3.17) for the 4th eigenvalue of (3.18) for $\nu = 1.0$

k	$\lvert\sigma_k\rvert$	$\lvert\delta_{h,k}^{\sigma}\rvert$	I_{eff}^{σ}	$\lvert\eta_k + \eta_k^*\rvert$	$\lvert\delta_{h,k}^{\eta} + \delta_{h,k}^{\eta^*}\rvert$	I_{eff}^{η}	$\lvert\eta_k^{\text{it}}\rvert$	$\lvert\lambda - \lambda_{h,k}\rvert$
5	$1.0e+1$	$1.0e+0$	1.4	$9.0e+2$	$2.0e+2$	0.2	$9.0e+2$	$8.0e+2$
7	$1.0e-1$	$5.0e-2$	0.4	$4.0e+0$	$5.0e+1$	1.6	$3.0e+1$	$9.0e+0$
9	$1.0e-2$	$4.0e-3$	0.4	$2.0e+0$	$4.0e+0$	1.8	$1.0e+0$	$1.0e+0$
11	$1.0e-4$	$9.0e-5$	0.7	$6.0e-1$	$1.0e-2$	1.5	$5.0e-2$	$5.0e-1$
13	$1.0e-6$	$4.0e-7$	0.3	$5.0e-1$	$9.0e-4$	1.3	$3.0e-3$	$5.0e-1$

4. Adaptive eigenvalue computation

In contrast to the test computation in the previous section, in practice, it is not necessary to evaluate all indicators at every iteration. For example as long as $\delta^{\sigma}_{h,k} \gg \sigma_{h,k}$ or $|\delta^{\sigma}_{h,k} + \sigma_{h,k}| \not\ll 1$, we need not evaluate any other term than $\delta^{\sigma}_{h,k}$ and $\sigma_{h,k}$. Accordingly, we propose the following adaptive procedure, which combines mesh adaptation for reducing the discretization error with an automatic stopping criterion for the iteration.

Algorithm 4.1 Adaptive eigenvalue computation on fixed mesh

```
Choose $c \in (0,1)$ and set $k := 0$.
Compute $u_{h,0}$, $z_{h,0}$, e.g., by Algorithm 2.1.
loop
    Compute $u_{h,k+1}$, $z_{h,k+1}$, e.g., by Algorithm 2.1.
    Compute $\sigma_{h,k}$, $\delta^{\sigma}_{h,k}$.
    if $|\delta^{\sigma}_{h,k}| \leqslant c|\sigma_{h,k}|$ or $|\delta^{\sigma}_{h,k}| + |\sigma_{h,k}| < c$ then
        if $|\sigma_{h,k}| > 1/2$ then
            return Refine mesh (by the strategies in [15]), as $h$ resolution is too
            coarse to compute the desired eigenvalue.

        else
            Compute $\eta_{h,k}$, $\eta^{*}_{h,k}$, $\delta^{\eta}_{h,k}$, $\delta^{\eta^{*}}_{h,k}$.
            if $|\delta^{\eta}_{h,k}| \leqslant c|\eta_{h,k}|$ and $|\delta^{\eta^{*}}_{h,k}| \leqslant c|\eta^{*}_{h,k}|$ then
                Compute $\eta_k$
                if $|\eta_k| \leqslant c|\eta_{h,k} + \eta^{*}_{h,k}|$ then
                    return Refine mesh, as iteration error is below discretization error.

    Set $k := k+1$.
```

In the case that Algorithm 4.1 terminates because $\sigma_{h,k}$ is too large, the mesh refinement has to be done by some generic refinement rule, as we are unable to extract any information from the error identity. One might for instance use global refinement. If on the other hand Algorithm 4.1 terminates after reaching the discretization accuracy, we may localize the information in $\eta_{h,k} + \eta^{*}_{h,k}$, compare our remark at the end of Section 3.1. We may then refine the mesh according to this local information until we have reached sufficient accuracy in the computed eigenvalue.

We consider three different possibilities: we may refine uniformly, non-uniformly by the localized error indicators from the DWR error indicator (3.17),

$$\lambda - \lambda_{h,k} \approx \frac{1}{1-\sigma_{h,k}} \{\eta_{h,k} + \eta^{*}_{h,k} + \eta^{\mathrm{it}}_{k}\}, \tag{4.1}$$

or non-uniformly by using a residual-based energy-norm error indicator for the primal and dual eigenvalue problem of the form

$$|\lambda - \lambda_{h,k_{\max}}| \leqslant c_\lambda \left(\eta_E^2 + \eta_E^{*2}\right)^{1/2}. \tag{4.2}$$

For the derivation of the detailed form of the localized error indicators obtained from the DWR error estimate (4.1) and the energy-norm error estimate (4.2) and corresponding strategies for mesh adaptation, we refer to Heuveline & Rannacher [15]. From [15], we recall that in the present situation

$$\begin{aligned}
\eta_{h,k} &= \tfrac{1}{2}\rho(v_{h,k},\lambda_{h,k})(\psi), \quad \psi := I_{2h}^{(2)} v_{h,k+1}^* - v_{h,k}^*,\\
&= \tfrac{1}{2}\sum_{T\in\mathbb{T}_h}\big\{(\mathscr{A}v_{h,k} - \lambda_{h,k}v_{h,k},\psi)_T - \tfrac{1}{2}(a[\partial_n v_{h,k}],\psi)_{\partial T}\big\},\\
\eta_{h,k}^* &= \tfrac{1}{2}\rho^*(v_{h,k},\lambda_{h,k},v_{h,k}^*,\lambda_{h,k}^*)(\varphi), \quad \varphi := I_{2h}^{(2)} v_{h,k+1} - v_{h,k},\\
&= \tfrac{1}{2}\sum_{T\in\mathbb{T}_h}\big\{(\varphi,\mathscr{A}^*v_{h,k}^* - \lambda_{h,k}^*v_{h,k}^*)_T - \tfrac{1}{2}(\varphi,a[\partial_n v_{h,k}^*])_{\partial T}\big\}.
\end{aligned}$$

where $[\cdot]$ indicates the jump across the common face of two neighboring cells. The mesh adaptation is then oriented by the size of the computable cell-error indicators

$$\begin{aligned}
\eta_T &:= |(\mathscr{A}v_{h,k} - \lambda_{h,k}v_{h,k}, I_{2h}^{(2)}v_{h,k+1}^* - v_{h,k}^*)_T - \tfrac{1}{2}(a[\partial_n v_{h,k}], I_{2h}^{(2)}v_{h,k+1}^* - v_{h,k}^*)_{\partial T}|,\\
\eta_T^* &:= |(I_{2h}^{(2)}v_{h,k+1} - v_{h,k}, \mathscr{A}^*v_{h,k}^* - \lambda_{h,k}^*v_{h,k}^*)_T - \tfrac{1}{2}(I_{2h}^{(2)}v_{h,k+1} - v_{h,k}, a[\partial_n v_{h,k}^*])_{\partial T}|.
\end{aligned}$$

Further, we have

$$\eta_k^{\mathrm{it}} = \rho(v_{h,k},\lambda_{h,k})(v_{h,k}^*) = a(v_{h,k},v_{h,k}^*) - \lambda_{h,k}(v_{h,k},v_{h,k}^*).$$

The standard energy-norm error estimator involves the term

$$\eta_E^2 + \eta_E^{*2} = \sum_{T\in\mathbb{T}_h} h_T^2\big\{\rho_T^2 + \rho_T^{*2}\big\},$$

with the cell residuals

$$\begin{aligned}
\rho_T^2 &:= \|\mathscr{A}v_h - \lambda_h v_h\|_T^2 + \tfrac{1}{2}h_T^{-1}\|a[\partial_n v_h]\|_{\partial T\setminus\partial\Omega}^2,\\
\rho_T^{*2} &:= \|\mathscr{A}^*v_h^* - \lambda_h^* v_h^*\|_T^2 + \tfrac{1}{2}h_T^{-1}\|a[\partial_n v_h^*]\|_{\partial T\setminus\partial\Omega}^2.
\end{aligned}$$

In order to avoid problems due to different norms of primal and dual eigenfunctions in the residual based error indicator for its evaluation both primal and dual eigenfunctions are scaled such that $\|v\| = \|v^*\| = 1$. We use an interpolation of the first two eigenvectors as initial vector on the next mesh and continue the computation.

4.1. Numerical examples

We will illustrate the performance of the adaptive algorithms proposed above at several examples. First, we compare duality-based versus energy-norm-based mesh adaptation with respect to discretization efficiency on a polygonal domain with reentrant corner (slit domain). Then, we use our adaptive algorithm for balancing discretization and iteration error and investigate its efficiency and reliability.

We consider the eigenvalue problem of finding $v \in H_0^1(\Omega)$ and $\lambda \in \mathbb{C}$ such that, for $\nu = 1$ and $\mathbf{b} = (0,3)^T$,

$$-\nu\Delta v + \mathbf{b}\cdot\nabla v = \lambda v, \tag{4.3}$$

on the rectangular domain $\Omega = (-1,1)\times(-1,3)$ with a vertical slit with tip at $(0,0)$ shown in Fig. 1. This example has been adopted from Heuveline & Rannacher [15]. As the exact eigenvalues are unknown, we computed reference values on a uniform refined mesh with roughly four million unknowns.

In the presence of a reentrant corner at $(0,0)$ with angle $\omega = 2\pi$, the "first" eigenfunction contains a "corner singularity", that is it can be written in the form $v = Ar^{1/2}\sin(\vartheta/2) + \tilde{v}$, with $\tilde{v} \in H^2(\Omega)$ and $\{r,\vartheta\}$ being polar coordinates. Further, for nontrivial transport, say $b_y > 0$, a boundary layer occurs at the upper boundary $\{y = 3\}$. Note that in this case, the corresponding dual eigenfunction v^* has a boundary layer at the lower boundary $\{y = -1\}$. This particular example has been chosen in order to demonstrate the capability of the duality-based error estimator to balance between the goal of accurately representing the eigenvalue error with the sufficient resolution of the corner singularity in the eigenfunction.

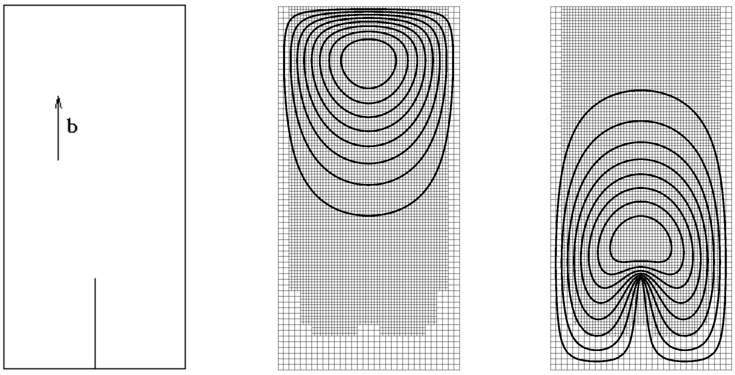

Figure 1: Configuration (left) of the test problem "rectangle with slit" and isolines of primal (middle) and dual (right) eigenfunction of the "first" eigenvalue.

4.1.1. Comparison duality-based versus energy-norm-based mesh adaptation.

First, we demonstrate the performance of the sensitivity-driven DWR error estimator for mesh adaptation against the traditional energy-error estimator. In this test the eigenvalue iteration is carried to its limit, i.e., until the iteration residual η_k^{it} is reduced by a factor of 10^{-12}. Table 3 displays the results obtained for the computation of the first and third eigenvalues $\lambda^{(1)}$ and $\lambda^{(3)}$, respectively. The corresponding primal and dual eigenfunctions and adapted meshes are shown in Figures 2 and 3. We clearly see that the meshes generated by the DWR approach are significantly more efficient than those obtained by energy-based adaptation.

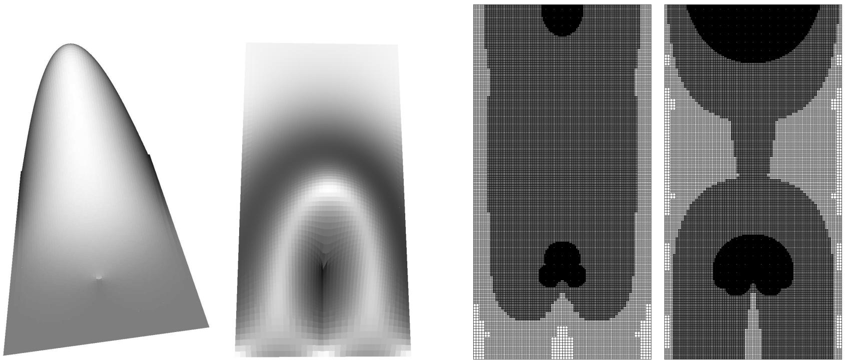

Figure 2: Primal and dual eigenfunction corresponding to the “first” eigenvalue and finest locally adapted mesh by the DWR (first) and the energy (second) approach.

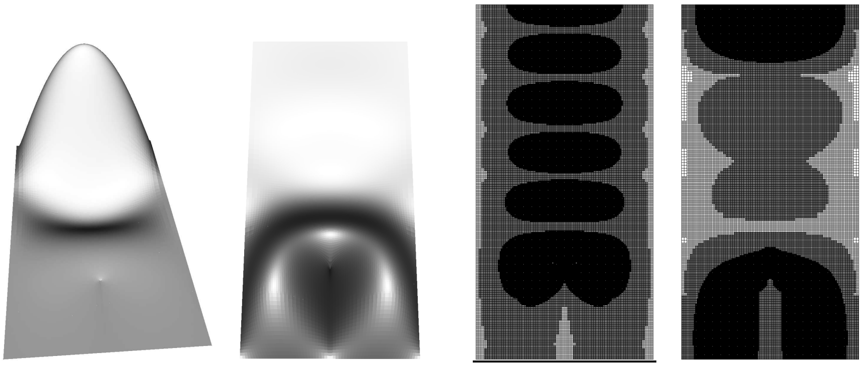

Figure 3: Primal and dual eigenfunction corresponding to the “third” eigenvalue and finest locally adapted mesh by the DWR (first) and the energy (second) approach.

4.1.2. Balancing of discretization and iteration error. Next, we test our adaptive algorithm. We start with a random vector on the coarse mesh. After that, we either use Algorithm 4.1, to stop the eigenvalue computation on the mesh once the discretization error is reached, or we stop the eigenvalue computation once the discrete eigenvalue residual η_k^{it} has been reduced by a factor of 10^{-12}. Once the eigenvalue computation has stopped, we refine the grid.

Table 3: Slit domain: Computation of the first and third eigenvalues $\lambda^{(1)} \approx 5.65740291$ and $\lambda^{(3)} \approx 12.70505782$, respectively, using the DWR (upper row) and the energy (lower row) method for error estimation

N	$k_{\max}$	$\lvert\lambda^{(1)} - \lambda^{(1)}_{h,k_{\max}}\rvert$
569	18	$1.53e-2$
2061	18	$7.51e-3$
6985	18	$3.71e-3$
12115	18	$1.97e-3$
26979	17	$9.49e-4$
34495	17	$5.22e-4$

N	$k_{\max}$	$\lvert\lambda^{(3)} - \lambda^{(3)}_{h,k_{\max}}\rvert$
569	30	$8.49e-2$
2161	31	$4.49e-2$
7453	32	$2.31e-2$
22553	31	$1.18e-2$
32113	31	$5.83e-3$
84597	32	$2.95e-3$

N	$k_{\max}$	$\lvert\lambda^{(1)} - \lambda^{(1)}_{h,k_{\max}}\rvert$
569	18	$1.53e-2$
1835	18	$1.09e-2$
5409	18	$4.89e-3$
12361	17	$2.39e-3$
26067	16	$1.23e-3$
49729	16	$6.07e-4$

N	$k_{\max}$	$\lvert\lambda^{(3)} - \lambda^{(3)}_{h,k_{\max}}\rvert$
569	30	$8.49e-2$
1739	30	$3.00e-2$
5347	32	$1.74e-2$
17519	31	$1.06e-2$
55561	33	$5.62e-3$
124205	32	$2.80e-3$

(i) Computation of “first” and “third” eigenvalue on the slit domain. We start our discussion with the “first” eigenvalue, i.e., that with smallest (positive) real part. As the results in Table 4 show, by comparing the left column of tables to the right ones, using the adaptive stopping criterion reduces the number of inner iterations by a factor of two while obtaining the same accuracy of the eigenvalue under consideration. On the other hand, we can save significantly in terms of degrees of freedom, N, when using locally refined meshes. The corresponding finest adapted mesh is shown in Fig. 2. We remark that, although the meshes obtained by local mesh refinement differ slightly when the eigenvalue computation is stopped early, the overall quality of the meshes for the approximation of the eigenvalue is not influenced by the adaptive stopping criterion.

Next, we consider the computation of the “third” eigenvalue, i.e., that with the third smallest real part. The results are shown in Table 5. Once again the use of the adaptive stopping criterion is visible, as we require only two thirds of the iterations used otherwise.

Table 4: Slit domain: Computation of the “first” eigenvalue $\lambda \approx 5.65740291$

(a) Global refinement, non-adaptive iteration

N	$k_{\max}$	$\lvert\lambda-\lambda_h^{k_{\max}}\rvert$	$I_{\text{eff}}^{k_{\max}}$
569	18	$1.53e-2$	3.0
2161	17	$7.51e-3$	3.0
8417	17	$3.71e-3$	3.1
33217	17	$1.84e-3$	3.1
131969	17	$9.16e-4$	3.1

(b) Global refinement, adaptive iteration

N	$k_{\max}$	$\lvert\lambda-\lambda_h^{k_{\max}}\rvert$	$I_{\text{eff}}^{k_{\max}}$
569	9	$1.54e-2$	2.9
2161	9	$7.40e-3$	2.9
8417	10	$3.73e-3$	3.1
33217	10	$1.84e-3$	3.1
131969	10	$8.97e-4$	3.1

(c) Local refinement using DWR indicators, non-adaptive iteration

N	$k_{\max}$	$\lvert\lambda-\lambda_h^{k_{\max}}\rvert$	$I_{\text{eff}}^{k_{\max}}$
569	18	$1.53e-2$	3.0
2061	18	$7.51e-3$	3.0
6985	18	$3.71e-3$	3.1
12115	18	$1.97e-3$	2.7
26979	17	$9.49e-4$	2.9

(d) Local refinement using DWR indicators, adaptive iteration

N	$k_{\max}$	$\lvert\lambda-\lambda_h^{k_{\max}}\rvert$	$I_{\text{eff}}^{k_{\max}}$
569	9	$1.54e-2$	2.9
2061	8	$7.31e-3$	2.9
6997	9	$3.70e-3$	3.0
11993	9	$2.01e-3$	2.7
26967	10	$9.49e-4$	2.9

Table 5: Slit domain: Computation of the third eigenvalue $\lambda \approx 12.70505782$

(a) Global refinement, non-adaptive iteration

N	$k_{\max}$	$\lvert\lambda-\lambda_h^{k_{\max}}\rvert$	$I_{\text{eff}}^{k_{\max}}$
569	30	$8.49e-2$	3.8
2161	31	$4.49e-2$	3.7
8417	32	$2.32e-2$	3.5
33217	30	$1.18e-2$	3.3
131969	32	$5.98e-3$	3.2

(b) Global refinement, adaptive iteration

N	$k_{\max}$	$\lvert\lambda-\lambda_h^{k_{\max}}\rvert$	$I_{\text{eff}}^{k_{\max}}$
569	19	$8.48e-2$	3.8
2161	20	$4.52e-2$	3.8
8417	20	$2.35e-2$	3.5
33217	21	$1.19e-2$	3.3
131969	21	$6.03e-3$	3.3

(c) Local refinement using DWR indicators, non-adaptive iteration

N	$k_{\max}$	$\lvert\lambda-\lambda_h^{k_{\max}}\rvert$	$I_{\text{eff}}^{k_{\max}}$
569	30	$8.49e-2$	3.8
2161	31	$4.49e-2$	3.7
7453	32	$2.31e-2$	3.5
22553	31	$1.18e-2$	3.4
32113	31	$5.83e-3$	3.4

(d) Local refinement using DWR indicators, adaptive iteration

N	$k_{\max}$	$\lvert\lambda-\lambda_h^{k_{\max}}\rvert$	$I_{\text{eff}}^{k_{\max}}$
569	19	$8.48e-2$	3.8
2161	20	$4.52e-2$	3.8
7453	21	$2.30e-2$	3.5
22527	21	$1.18e-2$	3.3
31961	21	$5.84e-3$	3.5

Table 5 shows that the effectivity of the eigenvalue-error estimation by the DWR approach is not perfect as the effectivity index

$$I_{\mathrm{eff}}^{k_{\max}} := \frac{|\lambda - \lambda_{h,k_{\max}}||1 - \sigma_{h,k_{\max}}|}{|\eta_{h,k_{\max}} + \eta^*_{h,k_{\max}}|}$$

is of size $I_{\mathrm{eff}}^{k_{\max}} > 2$ even on very much refined meshes. We suppose that this deficiency is due to the rather poor approximation of the slit singularity in the primal and dual eigenfunctions (occurring in the weights of the error estimator) on meshes, which are adapted only for computing the eigenvalues. In order to clarify this effect, we perform the following two tests: First, we repeat the above computations on a rectangular domain without a slit, in order to see whether the large effectivity index is caused by the corner singularity. Second, we again consider the slit domain but augment the DWR-driven mesh refinement by artificially imposed refinement around the tip of the slit in oder to better resolve the singularity.

(ii) Computation of “first” and “third” eigenvalue on the full rectangle. We repeat the computation for the same data but on the rectangular domain $\Omega = (-1,1) \times (-1,3)$ without a slit. In this case the eigenfunctions are smooth. The corresponding results in Table 6 show effectivity indices close to one. This indicates that the underestimation on the slit domain is due to the poor approximation of the exact eigenfunctions in the neighborhood of the slit singularity, which is not sufficiently compensated by local mesh refinement. As conjectured the results in Table 6 show effectivity indices, which are close to one.

Table 6: No-slit domain: Computation of the “first” and “third” eigenvalue using mesh adaptation by the DWR indicators and adaptive iteration.

N	$k_{\max}$	$\lvert\lambda^{(1)}-\lambda_h^{(1)}\rvert$	I_{eff}
153	8	$1.75e-2$	1.14
561	6	$4.21e-3$	1.03
2145	6	$1.00e-3$	0.97
7905	6	$2.49e-4$	0.97

N	$k_{\max}$	$\lvert\lambda^{(3)}-\lambda_h^{(3)}\rvert$	I_{eff}
153	10	$1.52e-1$	1.13
561	13	$4.19e-2$	1.04
2145	13	$9.99e-3$	0.95
8213	14	$2.74e-3$	1.03

(iii) Computation of “first” and “third” eigenvalue on the slit rectangle with additional refinement around the tip of the slit. Finally, in this test, we again consider the slit domain. The mesh refinement is driven by the DWR error estimator but in each refinement cycle additional mesh refinement is enforced in the neighborhood of the tip of the slit. The results in Table 7 show effectivity indices, which are close to one. This supports our conjecture about the cause of the deterioration of the effectivity indices observed on meshes which do not sufficiently resolve the corner singularity.

Table 7: Slit domain: Computation of the “first” and “third” eigenvalue using mesh adaptation by the DWR method and adaptive iteration with additional refinement around the tip of the slit.

N	$k_{\max}$	$\|\lambda^{(1)}-\lambda_h^{(1)}\|$	I_{eff}
1981	17	$9.14e-3$	1.1
2837	17	$2.11e-3$	1.0
5257	17	$5.70e-4$	1.0
11571	18	$2.11e-4$	1.0
34011	19	$5.44e-5$	1.0

N	$k_{\max}$	$\|\lambda^{(3)}-\lambda_h^{(3)}\|$	I_{eff}
1981	30	$5.79e-2$	6.2
2981	33	$1.43e-2$	1.3
6105	30	$3.45e-3$	1.1
16247	32	$6.65e-4$	1.0
47631	32	$1.10e-4$	0.9

Acknowledgment

For this work the second author has received a research student scholarship by the MAThematics Center Heidelberg (MATCH) at the University of Heidelberg. This support is gratefully acknowledged.

References

1. I. Babuška and J. Osborn, *A posteriori error estimates for the finite element method*. Int. J. Numer. Methods Engrg. 12, 1597–1615 (1978).

2. W. Bangerth and R. Rannacher, *Adaptive Finite Element Methods for Differential Equations*. Birkhäuser: Basel-Boston-Berlin, 2003.

3. R. Becker, *An adaptive finite element method for the Stokes equations including control of the iteration error*. In: Numerical Mathematics and Advanced Applications (H.G. Bock et al., eds), pp. 609–620, Proc. of ENUMATH 1997, World Scientific, London, 1998.

4. R. Becker, C. Johnson, and R. Rannacher, Adaptive error control for multigrid finite element methods. *Computing* 55 (1995), 271–288.

5. R. Becker and R. Rannacher, *A feed-back approach to error control in finite element methods: Basic analysis and examples*. East-West J. Numer. Math. 4, 237–264 (1996).

6. R. Becker and R. Rannacher, *An optimal control approach to a posteriori error estimation*. In Acta Numerica 2001 (A. Iserles, ed.), pp. 1-102, Cambridge University Press, Cambridge, 2001.

7. R. Becker and A. Ern, *A posteriori control of modeling errors and discretization errors*. Multiscale Model. Simul. 1, 221–238 (2003).

8. S. C. Brenner and L. R. Scott, *The Mathematical Theory of Finite Element Methods*. Springer, New York, 3rd Edition, 2008.

9. G. F. Carey and J. T. Oden, *Finite Elements. Computational Aspects*. Prentice-Hall, 1984.

10. C. Carstensen and R. Verfürth, *Edge residuals dominate a posteriori error estimates for low order finite element methods*. SIAM J. Numer. Anal. 36, 1571–1587 (1999).

11. F. Chaitin-Chatelin and V. Frayssé, *Lectures on Finite Precision Computations*. Series Software, Environments, and Tools, SIAM, Philadelphia, PA, 1996.

12. P. G. Ciarlet, *The Finite Element Method for Elliptic Problems*. Classics Appl. Math. 40, SIAM, Philadelphia, 2002.

13. G. H. Golub, Z. Zhang, and H. Zha, *Large sparse symmetric eigenvalue problems with homogeneous linear constraints: The Lanczos process with inner-outer iterations*. Linear Algebra Appl. 309, 289–306 (2000).

14. A. Greenbaum, *Iterative Methods for Solving Linear Systems*. Frontiers in Applied Mathematics, Vol. 17, SIAM, Philadelphia, PA, 1997.

15. V. Heuveline and R. Rannacher, *A Posteriori error control for finite element approximations of elliptic eigenvalue problems*, Adv. in Comput. Math. 15, 1–32 (2001).

16. Z. Jia, *Refined iterative algorithms based on Arnoldi's process for large unsymmetric eigenproblems*, Linear Algebra Appl. 259, 1–23 (1997).

17. T. Kato, *Perturbation Theory for Linear Operators*, Die Grundlehren der mathematischen Wissenschaft in EinzeldarstellungenSpringer, Vol. 132, 1966.

18. 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).

19. E. Lundström and L. Eldén, *Adaptive eigenvalue computations using Newton's method on the Grassmann manifold*. SIAM J. Matrix Anal. Appl. 23, 819–839 (2002).

20. D. Meidner, R. Rannacher, and J. Vihharev, *Goal-oriented error control of the iterative solution of finite element equations*. J. Numer. Math. 17, 143–172 (2009).

21. Y. Saad, *Variations on Arnoldi's method for computing eigenelements of large unsymmetric matrices*. Linear Algebra Appl. 34, 269–295 (1980).

22. J. A. Scott, *An Arnoldi code for computing selected eigenvalues of sparse, real, unsymmetric matrices*. ACM Trans. Math. Software 21, 432–475 (1995).

23. V. Simonici and L. Eldén, *Inexact Rayleight quotient-type methods for eigenvalue computations*. BIT 42, 159–182 (2002).

24. D. C. Sorensen, *Numerical Methods for Large Eigenvalue Problems*, Acta Numerica 2002 (A. Iserles, ed.). Cambridge University Press, 2002.

25. R. Verfürth, *A Review of A Posteriori Error Estimation and Adaptive Mesh-Refinement Techniques*. Wiley/Teubner, New York-Stuttgart, 1996.