Sonderforschungsbereich 382
Universität Tübingen

Multilevel Method for Mixed Eigenproblems

R. Hiptmair and **K. Neymeyr**

Report #159

January 2001

MULTILEVEL METHOD FOR MIXED EIGENPROBLEMS

R. HIPTMAIR* AND K. NEYMEYR†

Abstract. For a Lipschitz-polyhedron $\Omega \subset \mathbb{R}^3$ we consider eigenvalue problems $\mathbf{curl}\,\alpha\,\mathbf{curl}\,\mathbf{u} = \lambda\mathbf{u}$ and $\mathbf{grad}\,\alpha\,\mathrm{div}\,\mathbf{u} = \lambda\mathbf{u}$, $\lambda > 0$, set in $\boldsymbol{H}(\mathbf{curl};\Omega)$ and $\boldsymbol{H}(\mathrm{div};\Omega)$. They are discretized by means of the conforming finite elements introduced by Nédélec. The preconditioned inverse iteration in its subspace variant is adapted to these problems. A standard multigrid scheme serves as preconditioner. The main challenge arises from the large kernels of the operators **curl** and div. However, thanks to the choice of finite element spaces these kernels have a direct representation through the gradients/rotations of discrete potentials. This makes it possible to use a multigrid iteration in potential space to obtain approximate projections onto the orthogonal complements of the kernels. There is ample evidence that this will lead to an asymptotically optimal method. Numerical experiments confirm the excellent performance of the method even on very fine grids.

Key words. Mixed eigenvalue problems, edge elements, Raviart-Thomas elements, mixed finite elements, preconditioned inverse iteration, multigrid methods

AMS subject classifications.

1. Introduction. Let $\Omega \subset \mathbb{R}^3$ be a Lipschitz-polyhedron [33], whose boundary is partitioned into Γ_D and Γ_N. Our focus is on the vector-valued eigenvalue problems

$$\begin{aligned} \mathbf{curl}\,\alpha\,\mathbf{curl}\,\mathbf{u} &= \lambda\mathbf{u} \quad \text{in } \Omega\,, & \mathbf{u}\times\mathbf{n} &= 0 \quad \text{on } \Gamma_D\,,\\ \mathrm{div}\,\mathbf{u} &= 0 \quad \text{in } \Omega\,, & \alpha\,\mathbf{curl}\,\mathbf{u}\times\mathbf{n} &= 0 \quad \text{on } \Gamma_N\,, \end{aligned} \tag{1.1}$$

and

$$\begin{aligned} \mathbf{grad}\,\alpha\,\mathrm{div}\,\mathbf{u} &= \lambda\mathbf{u} \quad \text{in } \Omega\,, & \mathbf{u}\cdot\mathbf{n} &= 0 \quad \text{on } \Gamma_D\,,\\ \mathbf{curl}\,\mathbf{u} &= 0 \quad \text{in } \Omega\,, & \alpha\,\mathrm{div}\,\mathbf{u}\cdot\mathbf{n} &= 0 \quad \text{on } \Gamma_N\,. \end{aligned} \tag{1.2}$$

Here, the vectorfields $\mathbf{u}$ is an eigenfunction, $\lambda \geq 0$ stands for the eigenvalue and $\alpha \in L^\infty(\Omega)$ is a uniformly positive coefficient.

We seek approximations of a few of the smallest non-zero eigenvalues and corresponding eigenfunctions. This problem is of considerable relevance in several areas of scientific computing. For instance, (1.1) describes so-called electromagnetic resonators, if $\mathbf{u}$ is regarded as the (scaled) electric field. We refer to [1, Sect. 1] for more detailed explanations. When we want to determine a couple of the lowest resonant modes for a given cavity Ω, we encounter exactly the eigenvalue problem (1.1). Beyond the calculation of resonant modes, approximations of the lowest eigenmodes are the basis for modal approaches: A set of dominant modes is computed once, and the fields at other frequencies are then approximated by a superposition of these modes. This can be used to extract lumped parameters for electromagnetic devices in the frequency domain. A completely different application emerges in the study of coupled solid-fluid systems. When one tries to find their eigenmodes, the eigenvalue problem (1.2) pops up [7].

Of course, there is a close relationship between (1.1) and (1.2) and eigenvalue problems for second order elliptic differential operators. For the latter case, which amounts to a generalized eigenvalue problem for large sparse symmetric *positive definite* matrices, a huge body of work about numerical solution methods has been compiled over the years [4, 5, 20, 35, 44, 46]. The driving force was the sheer size of the eigenproblems

*SFB 382, Universität Tübingen

†Mathematisches Institut, Universität Tübingen

arising from discretized PDEs. Millions of unknowns rule out the use of methods that rely on dense matrices or factorizations. In addition, it is highly desirable to avoid a deterioration of the convergence of the iterative schemes for large problems. As far as the solution of discretized elliptic boundary value problems is concerned, multigrid methods meet this requirement. It turned out that the multigrid idea can be grafted onto solution methods for discrete elliptic eigenproblems in several ways resulting in eigensolvers with optimal or quasi–optimal computational complexity. For instance, Hackbusch [35, 36] applies multigrid principles directly to the nonlinear eigenvalue problem to compute eigenvalue/vector approximations on the final grid by combining a multigrid iteration and nested iteration. Let us also mention the multigrid minimization technique of Mandel and McCormick [46], its extension by Deuflhard et al. [25], as well as the class of methods which apply multigrid as a linear solver. Essentially, the idea underlying this last class is to linearize the discrete eigenvalue problem by methods like inverse iteration [54] and to solve the associated system of linear equations approximately by multigrid [5]. Representing the application of the multigrid procedure by a multigrid preconditioner and taking inverse iteration (without a shift) as an outer iteration defines *preconditioned inverse iteration* (PINVIT). Recently, a new convergence theory for preconditioned inverse iteration has been devised providing sharp convergence estimates and substantial insight into the underlying geometry [52, 53].

The scheme of preconditioned inverse iteration is also known in the literature as preconditioned gradient method for the eigenvalue problem. The idea behind this term is to compute a sequence of iterates with decreasing Rayleigh quotients by successively correcting the iterates in the direction of the negative preconditioned gradient of the Rayleigh quotient. By doing so, one expects that the sequence of iterates converges to an eigenvector while the Rayleigh quotients tend to the smallest eigenvalue. Preconditioned gradient methods have been studied predominantly by Russian authors, see for instance Samokish [56], Petryshyn [55], Godunov et al. [32], D'yakonov et al. [26, 28], Knyazev [43, 44] as well as the monograph of D'yakonov [27] including an extensive bibliography. Knyazev in [44] gives a survey on preconditioned eigensolvers.

Preconditioned inverse iteration has been generalized to a subspace algorithm for computing some of the smallest eigenvalues together with the eigenvectors by emulating the subspace variant of inverse iteration [54]. Once again, the associated matrix equation is solved approximately. After each subspace correction step the Rayleigh–Ritz procedure is applied. It provides the Ritz values and Ritz vectors spanning the approximating subspace. Convergence estimates have been presented in [18, 50]. In sum, the resulting preconditioned eigensolver inherits the typical asymptotic multigrid efficiency from the multigrid procedure used to solve the associated linear equations.

On a smaller scale, researchers have also investigated ways to compute solutions to (1.1) and (1.2) [1, 58]. It is obvious that the large kernels of the differential operators **curl** and div pose the main challenge: A straightforward application of iterative techniques developed for the symmetric positive definite case is doomed, because these methods single out the smallest eigenvalues and will invariable churn out kernel vectors in the end. However, as $\lambda > 0$ is requested, these are not the desired answer. We are left with the task of steering the iterations away from the kernels.

One option is regularization, i.e. adding a term corresponding to a weak version of $\mathbf{grad}\,\mathrm{div}\,\cdot$ for (1.1) and $\mathbf{curl}\,\mathbf{curl}\,\cdot$ for (1.2) to the differential operator (cf. [1, Sect. 4.1] and [8]). This will make the kernel "visible" and convert the problem into a standard positive definite one. Thus it becomes amenable to "shift-and-invert"

techniques combined with, e.g., an implicitly restarted Lanczos method. The resulting indefinite linear systems of equations can be solved by means of Krylov-subspace methods, whose convergence will degrade for very large problems, however.

An alternative option is projection of approximate eigenvectors onto a complement of the kernels. This is the gist of our method, which we call *projected preconditioned inverse iteration* (PPINVIT). The idea to forgo regularization in favor of projections is fairly natural. For instance, it is used in [57] for 2D problems arising in waveguide design. Yet, little is gained, unless a fast projections and good preconditioners are at our disposal.

Recently, multilevel methods for the solution of $\boldsymbol{H}(\mathbf{curl};\Omega)$- and $\boldsymbol{H}(\mathrm{div};\Omega)$-elliptic boundary value problems have become available [3, 38, 40], if discretization is based on special conforming finite elements. The goal of this paper is to demonstrate how they can be forged into eigenproblem solvers featuring multigrid efficiency. The key idea is to combine the subspace variant of preconditioned inverse iteration [50] with an inexact multigrid projection onto the orthogonal complements of the kernels.

As the approach crucially hinges on particular properties of the finite elements, those are reviewed in the next section. Then we give a detailed description of the algorithm, complete with projection control and termination criteria. The fourth section is dedicated to some theoretical investigations into the convergence of the method. Yet, we have not succeeded in providing a comprehensive theoretical analysis. To compensate for this, we report quite a few numerical experiments in the final section. They give evidence of the efficacy and satisfactory performance of the method for some typical large eigenvalue problems.

2. Discrete eigenvalue problems. The Galerkin-discretization starts from the weak form of the eigenvalue problems: In the case of (1.1) we seek $\mathbf{u} \in \boldsymbol{H}_{\Gamma_D}(\mathbf{curl};\Omega)$, $\lambda > 0$ such that

$$(\alpha\,\mathbf{curl}\,\mathbf{u},\mathbf{curl}\,\mathbf{v})_0 = \lambda\,(\mathbf{u},\mathbf{v})_0 \quad \forall \mathbf{v} \in \boldsymbol{H}_{\Gamma_D}(\mathbf{curl};\Omega)\;. \tag{2.1}$$

If (1.2) is of concern the weak form reads: Seek $\mathbf{u} \in \boldsymbol{H}_{\Gamma_D}(\mathrm{div};\Omega)$, $\lambda > 0$ such that

$$(\alpha\,\mathrm{div}\,\mathbf{u},\mathrm{div}\,\mathbf{v})_0 = \lambda\,(\mathbf{u},\mathbf{v})_0 \quad \forall \mathbf{v} \in \boldsymbol{H}_{\Gamma_D}(\mathrm{div};\Omega)\;. \tag{2.2}$$

As usual, we adopt the notation $(\cdot,\cdot)_0$ for the $\boldsymbol{L}^2(\Omega)$-inner product. By testing (2.1) with gradients and (2.2) with **curl**s we observe that solutions $\mathbf{u}$ are either weakly divergence-free or weakly **curl**-free. As $\boldsymbol{H}_{\Gamma_D}(\mathbf{curl};\Omega)\cap\boldsymbol{H}(\mathrm{div};\Omega)$ and $\boldsymbol{H}_{\Gamma_D}(\mathrm{div};\Omega)\cap\boldsymbol{H}(\mathbf{curl};\Omega)$ are both compactly embedded in $\boldsymbol{L}^2(\Omega)$ [42], the Riesz-Schauder theory guarantees the existence of increasing sequences of nonzero eigenvalues $\lambda_1 \le \lambda_2 \le \ldots$. Since the bilinear forms on the left hand sides of (2.1) and (2.2) are symmetric, we can also conclude that the corresponding eigenspaces are $\boldsymbol{L}^2(\Omega)$-orthogonal.

This carries over to the discrete eigenfunctions obtained through a Galerkin-discretization of (2.1) and (2.2). In particular, we use conforming finite elements based on a hexahedral or simplicial triangulation $\mathcal{T}_h = \{T_i\}_i$ of Ω. Its faces and edges have to be equipped with an interior orientation. Then, using the constructions proposed by Nédélec in [49], we obtain the finite element spaces $\boldsymbol{\mathcal{W}}^1_p(\mathcal{T}_h) \subset \boldsymbol{H}(\mathbf{curl};\Omega)$ and $\boldsymbol{\mathcal{W}}^2_p(\mathcal{T}_h) \subset \boldsymbol{H}(\mathrm{div};\Omega)$ of any polynomial order $p \in \mathbb{N}_0$. Details and descriptions of the degrees of freedom are given in, e.g. [19, 31, 47, 49]. Dirichlet boundary conditions can be enforced by setting the degrees of freedom (d.o.f.) on Γ_D to zero.

In the case of lowest polynomial order $p = 0$ the finite elements are either known as Whitney-forms [14] or, in the engineering literature, as *edge elements* ($\boldsymbol{H}(\mathbf{curl};\Omega)$-conforming scheme) and *face elements* ($\boldsymbol{H}(\mathrm{div};\Omega)$-conforming scheme), respectively. They owe these names to the definition of their d.o.f., which are given by path integrals along edges of the mesh and flux integrals over its faces, respectively

$$\left\{\begin{array}{rcl} \boldsymbol{\mathcal{W}}_0^1 & \mapsto & \mathbb{R} \\ \mathbf{u}_h & \mapsto & \int\limits_e \mathbf{u}_h \cdot \mathbf{t}_e \, d\Gamma, \ e \text{ edge} \end{array}\right. , \quad \left\{\begin{array}{rcl} \boldsymbol{\mathcal{W}}_0^2 & \mapsto & \mathbb{R} \\ \mathbf{u}_h & \mapsto & \int\limits_f \mathbf{u}_h \cdot \mathbf{n}_f \, dS, \ f \text{ face} . \end{array}\right.$$

The finite element spaces form affine equivalent families, if special transformations are used [39]. This makes it possible to show approximation properties (cf. [24]) and the inverse inequalities

$$\begin{aligned} \|\mathbf{curl}\, \mathbf{u}_h\|_0 &\leq C h^{-1} \|\mathbf{u}_h\|_0 \quad \forall \mathbf{u}_h \in \boldsymbol{\mathcal{W}}_p^1(\mathcal{T}_h) , \\ \|\mathrm{div}\, \mathbf{u}_h\|_0 &\leq C h^{-1} \|\mathbf{u}_h\|_0 \quad \forall \mathbf{u}_h \in \boldsymbol{\mathcal{W}}_p^2(\mathcal{T}_h) , \end{aligned}$$

where $h := \max\{\mathrm{diam}\, T,\ T \in \mathcal{T}_h\}$ is the meshwidth and $C > 0$ are generic constants. By this terminology we mean that C may **only** depend on $\Omega, \Gamma_D, \alpha, p$, and the shape-regularity of the finite element mesh. On the other hand, the value of generic constants C may change between different occurrences.

Despite the glaring differences in their definitions, the finite element spaces for $\boldsymbol{H}(\mathbf{curl};\Omega)$ and $\boldsymbol{H}(\mathrm{div};\Omega)$ introduced above are closely related. As discussed in [16, 17, 39], they all can be viewed as spaces of *discrete differential forms*. This is the rationale behind our decision to treat both (1.1) and (1.2) in a common framework. In a sense, we will adopt the common notation $\boldsymbol{\mathcal{V}}_h$ for both $\boldsymbol{\mathcal{W}}_p^1(\mathcal{T}_h)$ or $\boldsymbol{\mathcal{W}}_p^2(\mathcal{T}_h)$ with suitable Dirichlet boundary conditions imposed.

Hitherto, discrete differential forms supply the only conforming finite element discretization of (2.1) and (2.2) that can steer clear of so-called *spurious modes*. For instance, if one uses $H^1(\Omega)$-conforming finite elements to discretize the Cartesian components of the vectorfields $\mathbf{u}$, the discrete spectrum may feature eigenvalues that are not related to an eigenvalue of the continuous problem [12, 15, 30]. On the contrary, in recent years rigorous arguments have been found, why discrete differential forms ensure a correct approximation of the spectrum [10, 13, 21, 22, 30, 48]. For quasiuniform and shape-regular families of meshes convergence of the eigenvalues will be quadratic in the meshwidth [21] under mild assumptions on the smoothness of the eigenfunctions.

A key role in the convergence theory is played by *discrete potentials*. They refer to an exceptional property of discrete differential forms, namely that they give rise to analogues to de Rham's exact sequences in a purely discrete setting [11, 17]. In particular, for contractible Ω, $\Gamma_D = \partial\Omega$ or $\Gamma_N = \partial\Omega$,

$$\{\mathbf{u}_h \in \boldsymbol{\mathcal{W}}_p^1(\mathcal{T}_h),\ \mathbf{curl}\, \mathbf{u}_h = 0\} = \mathbf{grad}\, \mathcal{W}_p^0(\mathcal{T}_h) , \tag{2.3}$$

$$\{\mathbf{u}_h \in \boldsymbol{\mathcal{W}}_p^2(\mathcal{T}_h),\ \mathrm{div}\, \mathbf{u}_h = 0\} = \mathbf{curl}\, \boldsymbol{\mathcal{W}}_p^1(\mathcal{T}_h) , \tag{2.4}$$

where $\mathcal{W}_p^0(\mathcal{T}_h)$ stands for the space of continuous finite element functions, piecewise polynomial of degree $p+1$ over $\mathcal{T}_h$, the conventional Lagrangian finite elements (see [23]). A proof of these identities can be found in [39]. Now it is clear, why $\mathcal{W}_p^0(\mathcal{T}_h)$ and $\boldsymbol{\mathcal{W}}_p^1(\mathcal{T}_h)$ have been dubbed spaces of discrete potentials. Those will be denoted by $\mathcal{S}_h$ and $G_h : \mathcal{S}_h \mapsto \boldsymbol{\mathcal{V}}_h$ is the related differential operator mapping into the kernel of A_h, that is, $G_h := \mathbf{grad}$ or $G_h := \mathbf{curl}$.

In the case of complex topologies and Dirichlet boundary conditions on parts of $\partial\Omega$, the kernels of the differential operators are no longer completely given by suitable discrete potentials. What is still missing are low-dimensional spaces of harmonic vectorfields, $\boldsymbol{\mathcal{H}}^1(\mathcal{T}_h) \subset \boldsymbol{\mathcal{W}}_0^1(\mathcal{T}_h)$ and $\boldsymbol{\mathcal{H}}_2(\mathcal{T}_h) \subset \boldsymbol{\mathcal{W}}_0^2(\mathcal{T}_h)$, whose dimensions depend on the topology of Ω and the arrangements of the connected components of Γ_D. For instance, if $\Gamma_D = \partial\Omega$ the dimension of $\boldsymbol{\mathcal{H}}^1(\mathcal{T}_h)$ is equal to the number of connected components of $\partial\Omega$. A basis for $\boldsymbol{\mathcal{H}}^1(\mathcal{T}_h)$ is given by the gradients of piecewise linear continuous functions that assume the value 1 on one connected component of Γ_D and vanish on the other. Evidently, this basis can be constructed with little effort. In the case of Neumann boundary conditions throughout, $\dim \boldsymbol{\mathcal{H}}^1(\mathcal{T}_h)$ is equal to the number of homology classes of boundary cycles that are bounding relative to Ω. To find a basis, we associate a cutting surface to each homology class and compute the gradient of a piecewise linear function that is continuous except for a jump of height 1 across the cutting surface [2]. The surfaces can be determined by means of graph-theoretic algorithms [34]. In case of mixed boundary conditions the situation is more involved [29], but for concrete geometries the harmonic vectorfields can usually be found easily. In the sequel we will write $\boldsymbol{\mathcal{H}}_h$ for a space of harmonic vectorfields and will take for granted that a basis $\{\mathbf{h}_1, \ldots, \mathbf{h}_q\}$ of $\boldsymbol{\mathcal{H}}_h$ has been computed.

In sum, we face the abstract discrete eigenvalue problem: Seek $\mathbf{u}_h \in \boldsymbol{\mathcal{V}}_h$ such that

$$a(\mathbf{u}_h, \mathbf{v}_h) = \lambda \left(\mathbf{u}_h, \mathbf{v}_h\right)_0 \quad \forall \mathbf{v}_h \in \boldsymbol{\mathcal{V}}_h \ , \tag{2.5}$$

where $a(\cdot,\cdot)$ stands for the positive semidefinite bilinear form from (2.1) or (2.2). We associate operators $A_h : \boldsymbol{\mathcal{V}}_h \mapsto \boldsymbol{\mathcal{V}}_h'$ and $M_h : \boldsymbol{\mathcal{V}}_h \mapsto \boldsymbol{\mathcal{V}}_h'$ with the bilinear forms in (2.5), which converts it into an operator equation

$$A_h \mathbf{u}_h = \lambda M_h \mathbf{u}_h \ . \tag{2.6}$$

The basis of $\boldsymbol{\mathcal{V}}_h$ dual to the set of degrees of freedom is called the nodal basis $\{\mathbf{b}_\iota\}_{\iota \in J}$, with J a suitable index set. The basis functions are locally supported and satisfy

$$\|\mathbf{b}_\iota\|_0 \leq C \operatorname{diam}\operatorname{supp}(\mathbf{b}_\iota) \|\mathbf{b}_\iota\|_A \quad \iota \in J \ , \tag{2.7}$$

with $\|\cdot\|_A$ the *energy-seminorm* induced by $a(\cdot,\cdot)$. Given the nodal basis, (2.6) can also be read as a matrix equation, A_h being the stiffness matrix and M_h the mass matrix, which are both large and sparse.

We follow the convention that functions will be given Roman symbols, whereas Greek letters are used for functionals. Those related to the base space $\boldsymbol{\mathcal{V}}_h$ will be given bold tokens, whereas entities from the potential space $\mathcal{S}_h$ are printed in plain style.

3. Projected preconditioned inverse iteration (PPINVIT). Standard inverse iteration (without shift) for an eigenvalue problem $A_h \mathbf{u}_h = \lambda M_h \mathbf{u}_h$ with symmetric *positive definite* operators $A_h : \boldsymbol{\mathcal{V}}_h \mapsto \boldsymbol{\mathcal{V}}_h'$, $M_h := \boldsymbol{\mathcal{V}}_h \mapsto \boldsymbol{\mathcal{V}}_h'$ computes a new iterate $\mathbf{x}_h^{\text{new}} \in \boldsymbol{\mathcal{V}}_h$ from the old $\mathbf{x}_h \in \boldsymbol{\mathcal{V}}_h$ through

$$\mathbf{y}_h = \kappa A_h^{-1} M_h \mathbf{x}_h^{\text{old}} \quad , \quad \mathbf{x}_h^{\text{new}} := \mathbf{y}_h / \|\mathbf{y}_h\|_0 \ ,$$

for some $\kappa \neq 0$. First, observe that the choice of κ is immaterial. Therefore, we may set $\kappa = r(\mathbf{x}_h)$, where

$$r(\mathbf{x}_h) = \frac{\langle A_h \mathbf{x}_h, \mathbf{x}_h \rangle}{\langle M_h \mathbf{x}_h, \mathbf{x}_h \rangle} \tag{3.1}$$

denotes the Rayleigh quotient and $\langle \cdot, \cdot \rangle$ the duality pairing. This choice of κ has the effect that $\mathbf{y}_h - \mathbf{x}_h$ converges to zero when $r(\mathbf{x}_h)$ approaches the smallest eigenvalue. Thus we recover the typical situation, where a correction is determined by solving a linear system with a small residual as right hand side. This paves the way for the application of a *preconditioner* $B_h : \mathcal{V}_h' \mapsto \mathcal{V}_h$, an approximate inverse of A_h, to compute $\mathbf{y}_h$. We arrive at the update formula

$$\mathbf{y}_h = \mathbf{x}_h - B_h(A_h\mathbf{x}_h - r(\mathbf{x}_h)M_h\mathbf{x}_h) \quad , \quad \mathbf{x}_h^{\text{new}} := \mathbf{y}_h / \left\|\mathbf{y}_h\right\|_0 \ , \tag{3.2}$$

which is the basic building block for the algorithm of the preconditioned inverse iteration [52]. The iterates will converge linearly to an eigenvector belonging to the smallest eigenvalue. The theoretically possible but unlikely case that preconditioned inverse iteration gets stuck in a higher eigenvalue does not take place in practice thanks to rounding errors. If an invariant subspace corresponding to the s smallest eigenvalues is desired, we can resort to the subspace variant. After a Rayleigh–Ritz projection, it updates each of the s Ritz vectors $\mathbf{x}_h^1, \ldots, \mathbf{x}_h^s$ according to (3.2) with $r(\mathbf{x}_h)$ replaced by the Ritz-values [18, 50].

Let us return to the actual setting, in which A_h is only *positive semidefinite*. Then, it is natural to demand that $\mathbf{y}_h$ is contained in the $\boldsymbol{L}^2(\Omega)$-orthogonal complement of $\mathrm{Ker}(A_h)$, as this is satisfied for any eigenvector belonging to a nonzero eigenvalue. In other words, the (exact) inverse iteration should be based on the pseudo-inverse $A_h^\dagger : \mathcal{V}_h' \mapsto \mathcal{V}_h$. Then $\mathbf{x}_h$ will converge to an eigenvector corresponding to λ_1 as long as the starting vector (for the case of exact arithmetic) is not orthogonal to that eigenvector.

Well, the pseudo-inverse $A^\dagger$ is elusive and has to be approximated. We suggest to do so by means of a plain multigrid method. It relies on a hierarchy of nested meshes $\mathcal{T}_0 \prec \mathcal{T}_1 \prec \ldots \prec \mathcal{T}_L := \mathcal{T}_h$ and the corresponding finite element spaces $\mathcal{V}_0 \subset \mathcal{V}_1 \subset \ldots \subset \mathcal{V}_L := \mathcal{V}_h$. The natural way to create such meshes is through successive refinement of an initial rather coarse mesh $\mathcal{T}_0$, as described in [6, 9] for tetrahedral meshes. The refinement strategies make sure that the shape regularity of $\mathcal{T}_0$ is almost preserved for all finer meshes.

We instantly get a sequence of operators $A_l : \mathcal{V}_l \mapsto \mathcal{V}_l'$ generated by the bilinear form $a(\cdot, \cdot)$ on $\mathcal{V}_l$. The embedding of the spaces $\mathcal{V}_{l-1} \subset \mathcal{V}_l$ spawns the canonical prolongation operators $I_l : \mathcal{V}_{l-1} \mapsto \mathcal{V}_l$, $l = 1, \ldots, L$. Their adjoints $I_l^* : \mathcal{V}_l' \mapsto \mathcal{V}_{l-1}'$ are known as restrictions [37, Sect. 3.6]. These operators are purely local and cheaply implemented [40].

The definition of the symmetric multigrid preconditioner is based on the recursive algorithm sketched in figure 3.1. There R_l^T is defined by $\left\langle \boldsymbol{\rho}_l, R_l^T \boldsymbol{\phi}_l \right\rangle = \langle \boldsymbol{\phi}_l, R_l \boldsymbol{\rho}_l \rangle$, $\boldsymbol{\rho}_l, \boldsymbol{\phi}_l \in \mathcal{V}_l'$. Then the application of the multigrid preconditioner $B_h : \mathcal{V}_h \mapsto \mathcal{V}_h'$ can be realized as follows

$$\mathbf{c}_h := B_h \boldsymbol{\rho}_h \quad \Longleftrightarrow \quad \mathbf{c}_h := 0; \quad \mathsf{mgcycle}(L, \mathbf{c}_h, \boldsymbol{\rho}_h) \ . \tag{3.3}$$

The operators $R_l : \mathcal{V}_l' \mapsto \mathcal{V}_l$ occurring in the algorithm are conventional smoothing operators on level l, $l = 1, \ldots, L$. We will only consider point smoothers of Jacobi- or Gauß-Seidel-type. For the latter, one sweep on level l, $l = 1, \ldots, L$, with initial guess $u_l \in \mathcal{V}_l$ and right hand side $\boldsymbol{\rho}_l \in \mathcal{V}_l'$ reads

$$\textbf{foreach}(\iota \in J_l) \quad \{ \quad \mathbf{u}_l \leftarrow \mathbf{u}_l + \frac{\langle \boldsymbol{\rho}_l, \mathbf{u}_l \rangle}{a(\mathbf{b}_\iota, \mathbf{b}_\iota)} \cdot \mathbf{b}_\iota \quad \} \ .$$

```
mgcycle< A > (int l,reference u_l ∈ V_l, const ρ_l ∈ V_l')
{
  if(l == 0) {   u_0 = A_0^† ρ_0   }
  else   {
    // Pre-smoothing
    for(int i = 0 ; i < μ_1 ; ++i) {   u_l   ←   u_l + R_l(ρ_l − A_l u_l)   }
    // Coarse grid correction
    σ_l := ρ_l − A_h u_l
    ρ_{l−1} := I_l^* σ_l
    c_{l−1} := 0 ∈ V_{l−1}
    for(int i = 0 ; i < ν ; ++i) mgcycle< A >(l − 1,c_{l−1},ρ_{l−1})
    u_l ← u_l + I_l c_{l−1}
    // Post-smoothing
    for(int i = 0 ; i < μ_2 ; ++i) {   u_l   ←   u_l + R_l^T(ρ_l − A_l u_l)   }
  }
}
```

FIG. 3.1. *Multigrid algorithm defining the preconditioner B_h. The parameters $\mu_1, \mu_2, \nu \in \mathbb{N}$ define the type of the cycle. For $\nu = 1$ we get a $V(\mu_1, \mu_2)$-cycle, for $\nu = 2$ a $W(\mu_1, \mu_2)$-cycle.*

Though A_h is singular, relaxation will go smoothly, as (2.7) guarantees $a(\mathbf{b}_\iota, \mathbf{b}_\iota) > 0$. However, this innocent looking procedure disrupts everything, because $\mathbf{b}_\iota$ does not exactly belong to $\mathrm{Ker}(A_l)^\perp$. Thus, the action of B_h will invariably introduce components in $\mathrm{Ker}(A_h)$ into the iterates. Eventually the iterates might tumble into the kernel.

To prevent this, we have to weed out the kernel contributions as soon as they are introduced. Formally, this can be done by projecting $\mathbf{y}_h$ from (3.2) onto $\mathrm{Ker}(A_h)^\perp$. Fortunately, if Ω is contractible, the representation of $\mathrm{Ker}(A_h)$ through discrete potentials according to $\mathrm{Ker}(A_h) = G_h \mathcal{S}_h$ enables us to express the $\boldsymbol{L}^2(\Omega)$-orthogonal projection $P_h : \boldsymbol{\mathcal{V}} \mapsto \mathrm{Ker}(A)^\perp$ through

$$P_h := Id - G_h T_h^\dagger G_h^* M_h \ , \tag{3.4}$$

where $T_h : \mathcal{S}_h \mapsto \mathcal{S}_h'$ is the operator associated with the bilinear form

$$d : \mathcal{S} \times \mathcal{S} \mapsto \mathbb{R} \quad , \quad d(u_h, v_h) = (G_h u_h, G_h v_h)_0 \ , \ u_h, v_h \in \mathcal{S}_h \ . \tag{3.5}$$

Yet, the exact computation of $T_h^\dagger \rho_h$ for some $\rho_h \in \mathcal{S}_h'$ is all but impossible. Just recall that in the case of the eigenvalue problem in $\boldsymbol{H}(\mathbf{curl}; \Omega)$ the operator T_h is the discrete Laplacian, i.e. in general described by a huge sparse stiffness matrix. Therefore, we cannot help using an approximate pseudo-inverse also in this case. A multigrid scheme analogous to the one outlined in figure 3.1 comes handy, this time to be conducted in the potential space with the operators A_l replaced by their counterparts $T_l : \mathcal{S}_l \mapsto \mathcal{S}_l'$. This will yield an approximate projection $\widetilde{P}_h$

$$\widetilde{P}_h := Id - G_h C_h G_h^* M_h \ , \tag{3.6}$$

where C_h stands for the approximate (pseudo-)inverse of T_h furnished by the multigrid cycle. Reassuringly, we do not have to worry about pollution in $\mathrm{Ker}(G_h)$ this time,

because in (3.4) the operator G_h is applied to the result, suppressing any kernel component.

If we have to take into account harmonic vectorfields in $\mathcal{H}_h := \mathrm{Span}\{\mathbf{h}_1, \ldots, \mathbf{h}_q\}$, their basis should be approximately orthogonalized to $G_h\mathcal{S}_h$. This can be done once and for all before the actual eigenvalue computations utilizing a few steps of the approximate multigrid projection $\widetilde{P}_h$. For the sake of efficiency, a nested iteration approach should be employed. Eventually, the basis functions should be $\boldsymbol{L}^2(\Omega)$-orthonormalized to each other by solving a small linear system of equations. If $\widetilde{\mathbf{h}}_1, \ldots, \widetilde{\mathbf{h}}_q$ are the functions thus obtained, $\widetilde{\mathcal{H}}_h := \mathrm{Span}\left\{\widetilde{\mathbf{h}}_1, \ldots, \widetilde{\mathbf{h}}_q\right\}$ will be another suitable space of harmonic vectorfields. Given this preprocessing, orthogonality to $\widetilde{\mathcal{H}}_h$ can be easily enforced.

The final algorithm implementing the inexact projection is given in figure 3.2 (right). We point out that G_h is a local operator, too, whose matrix representation can be derived from the embedding $G_h\mathcal{S}_h \subset \mathcal{V}_h$ [40, Sect. 6]. Let us elucidate this for edge elements: Assuming nodal bases of $\mathcal{V}_h$ and $\mathcal{S}_h$ the evaluation of G_h boils down to simply distributing the nodal values from vertices (to which d.o.f. of $\mathcal{S}_h$ are associated) to edges, taking into account their orientations by means of weights +1 or -1.

In the end, incorporating the total action of project into $\tilde{P}_h$, we get the following update formula for an approximate eigenvector

$$\mathbf{y}_h = \tilde{P}_h(Id - B_h(A_h - \kappa M_h))\tilde{P}_h\mathbf{x}_h \quad , \quad \mathbf{x}_h^{\mathrm{new}} = \mathbf{y}_h / \left\|\mathbf{y}_h\right\|_0 \ . \tag{3.7}$$

Cast into an algorithm, this yields the procedure update displayed in figure 3.2 (left).

```
update(reference x_h ∈ V_h, κ ∈ ℝ)
{
  project(x_h)
  φ_h := A_h x_h ; ψ_h := M_h x_h
  ρ_h := φ_h − κ · ψ_h
  c_h := 0 ∈ V_h
  mgcycle< A >(L, c_h, ρ_h)
  x_h ← x_h − c_h
  project(x_h)
  x_h ← x_h / |x_h|
}
```

```
project(reference x_h ∈ V_h)
{
  // Treat harmonic vectorfields
  for(int i = 1; i ≤ q; ++i)
  { x_h ← x_h − (h̃_i, x_h)_0 · h̃_i }
  η_h := M_h x_h
  φ_h := G_h^* η_h
  c_h = 0 ∈ S_h
  mgcycle< T >(L, c_h, φ_h)
  x_h ← x_h − G_h c_h
}
```

FIG. 3.2. *Update procedure for the projected preconditioned inverse interation*

It is hazardous to replace κ in (3.7) by the plain Rayleigh quotient (3.1), because significant kernel components might remain after the inexact projection. If we set $\kappa = r(\mathbf{x}_h)$, we might encounter $\kappa \ll \lambda$ though $A_h\mathbf{x}_h = \lambda M_h P_h\mathbf{x}_h$, i.e. the components of $\mathbf{x}_h$ in $\mathrm{Ker}(A_h)^\perp$ already provide the desired eigenvector. Guided by the idea that the scheme should come close to inverse iteration in the complement $\mathrm{Ker}(A_h)^\perp$ we should choose $\kappa = r_\perp(\mathbf{x}_h) = \langle A_h\mathbf{x}_h, \mathbf{x}_h\rangle \,/\, (P_h\mathbf{x}_h, P_h\mathbf{x}_h)_{\boldsymbol{L}^2(\Omega)}$. In practice, we are denied this option as $P_h\mathbf{x}_h$ is not available. However, we still want a replacement for $r_\perp$ that is insensitive to kernel components. A promising candidate is the "two-step Rayleigh

quotient"

$$r_Q(\mathbf{x}) = \frac{\left\langle A_h M_h^{-1} A_h \mathbf{x}_h, \mathbf{x}_h \right\rangle}{\left\langle A_h \mathbf{x}_h, \mathbf{x}_h \right\rangle} , \tag{3.8}$$

with $r_Q(\mathbf{x}) \geq r_\perp(\mathbf{x}_h) \geq r(\mathbf{x}_h)$. Obviously, it yields an eigenvalue, if we have already hit an eigenvector in $\mathrm{Ker}(A_h)^\perp$. Two issues arise, nevertheless: First, there is a risk of breakdown, if $A_h \mathbf{x}_h = 0$. This means that the current approximate eigenvector lies in $\mathrm{Ker}(A_h)$, which hints at inadequate approximate projections. A way to detect and cure this condition will be discussed in section 5. The second problem is that the evaluation of (3.8) entails the solution of a linear system $M_h \mathbf{z}_h = A_h \mathbf{x}_h$ for the mass matrix M_h. As M_h is well conditioned, a few steps of an iterative method (CG,Gauß-Seidel) will give a reasonable approximate solution. Moreover, if $(\lambda, \mathbf{x}_h)$ is already close to an eigenvalue/eigenvector pair, $\lambda \mathbf{x}_h$ is an excellent initial guess.

With all building blocks in place, we can now state the crucial update step of the algorithm for the computation of the s, $s \in \mathbb{N}$, smallest nonzero eigenvalues and corresponding eigenvectors of (2.6). Its details are given in figure 3.3. The procedure ppinvit_step is meant to improve on approximations θ_i and $\mathbf{x}_h^i$, $i = 1, \ldots, s$, for eigenvalues and eigenvectors.

ppinvit_step (`reference` $(\theta_1, \ldots, \theta_s)^T \in \mathbb{R}^s$, `reference` $(\mathbf{x}_h^1, \ldots, \mathbf{x}_h^s) \in (\boldsymbol{\mathcal{V}}_h)^s$)
{
  // Ritz-projection
  for$(i = 1 \,;\, i \leq s \,;\, ++i)$ {
    $\boldsymbol{\phi}_h^i := A_h \mathbf{x}_h^i$; $\mathbf{z}_h^i = \lambda_i \mathbf{x}_h^i$; $\mathsf{cg}^m < M_h > (\mathbf{z}_h^i, \boldsymbol{\phi}_h^i)$
    $a_{ii} := \left\langle \boldsymbol{\phi}_h^i, \mathbf{z}_h^i \right\rangle$; $m_{ii} := \left\langle \boldsymbol{\phi}_h^i, \mathbf{x}_h^i \right\rangle$
    for$(j = 1 \,;\, j < i \,;\, ++i)$ { $a_{ij} = a_{ji} := \left\langle \boldsymbol{\phi}_h^j, \mathbf{z}_h^i \right\rangle$; $m_{ij} = m_{ji} := \left\langle \boldsymbol{\phi}_h^j, \mathbf{x}_h^i \right\rangle$ }
  }
  // Rayleigh–Ritz procedure
  $\mathsf{A}_s := (a_{ij}) \in \mathbb{R}^{s,s}$; $\mathsf{M}_s := (m_{ij}) \in \mathbb{R}^{s,s}$;
  Find $\mathsf{Y} \in \mathbb{R}^{s,s}$ and Ritz values $\Theta = \mathrm{diag}(\theta_1, \ldots, \theta_s)$ such that $\mathsf{A}_s \mathsf{Y} = \mathsf{Y} \mathsf{M}_s \Theta$
  // Ritz vectors
  $(\mathbf{x}_h^1, \ldots, \mathbf{x}_h^s) \leftarrow (\mathbf{x}_h^1, \ldots, \mathbf{x}_h^s) \cdot \mathsf{Y}$
  // Approximate projected inverse iteration
  for$(i = 1 \,;\, i \leq s \,;\, ++i)$ { update$(\mathbf{x}_h^i, \theta_i)$ }
}

FIG. 3.3. *One step of the subspace variant of the algorithm for projected preconditioned inverse iteration.* $\mathsf{cg}^m < M_h > (\mathbf{z}_h, \boldsymbol{\phi}_h)$ *refers to* $m \in \mathbb{N}$ *CG-steps for the solution of* $M_h \mathbf{z}_h = \boldsymbol{\phi}_h$.

The discussion of termination criteria is postponed to section 5. Initial guesses for the eigenvectors can easily obtained through nested iteration by prolongating approximate eigenfunction from coarser grids.

Remark. For positive definite operators the Rayleigh–Ritz method is often applied to a modified/enlarged subspace (consisting of the actual subspace, the actual search directions and possibly the old iterates). This is known to improve convergence [44, 45] if $A_h > 0$. Yet, this trick is not advisable for the semidefinite problem, because a massive amplification of kernel components might occur.

4. Convergence. The theoretical examination of the algorithm starts with the $\boldsymbol{L}^2(\Omega)$-orthogonal decomposition and dual polar decomposition

$$\boldsymbol{\mathcal{V}}_h = \boldsymbol{\mathcal{X}}_h \otimes \boldsymbol{\mathcal{Z}}_h \,,\ \boldsymbol{\mathcal{Z}}_h := \mathrm{Ker}(A_h) \quad , \quad \boldsymbol{\mathcal{V}}_h' = \boldsymbol{\mathcal{X}}_h' \otimes \boldsymbol{\mathcal{Z}}_h' \,. \tag{4.1}$$

With respect to the splittings (4.1) the operators can be written in block form. For a symmetric preconditioner it reads

$$B_h = \begin{pmatrix} B_{\perp\!\!\perp} & B_{0\perp} \\ B_{0\perp}^T & B_{00} \end{pmatrix} : \boldsymbol{\mathcal{X}}_h' \otimes \boldsymbol{\mathcal{Z}}_h' \mapsto \boldsymbol{\mathcal{X}}_h \otimes \boldsymbol{\mathcal{Z}}_h \,, \tag{4.2}$$

and for the other operators

$$A_h = \begin{pmatrix} A_\perp & 0 \\ 0 & 0 \end{pmatrix} \,,\ M_h = \begin{pmatrix} M_\perp & 0 \\ 0 & M_0 \end{pmatrix} \,,\ A_h^\dagger = \begin{pmatrix} A_\perp^{-1} & 0 \\ 0 & 0 \end{pmatrix} \,,\ \widetilde{P}_h = \begin{pmatrix} Id_\perp & 0 \\ 0 & P_0 \end{pmatrix}$$

These formulas are immediate from the definition of the operators and the properties of the splittings. Be aware that $B_{0\perp} \neq 0$ causes the pollution by kernel components, and $P_0 \neq 0$ hints at an inexact projection.

Using $A_h A_h^\dagger + Q_h^* = Id_h^*$, where $Q_h : \boldsymbol{\mathcal{V}}_h \mapsto \boldsymbol{\mathcal{Z}}_h$ is the $\boldsymbol{L}^2(\Omega)$-orthogonal projection, we obtain from (3.7) with $\kappa = r_Q(\mathbf{x}_h)$

$$\mathbf{y}_h = \widetilde{P}_h \left((I_h - B_h A_h)(I_h - \kappa A_h^\dagger M_h) + \kappa B_h Q_h^* M_h + \kappa A_h^\dagger M_h \right) \widetilde{P}_h \mathbf{x}_h \,.$$

Splitting $\mathbf{y}_h = \mathbf{y}^0 + \mathbf{y}^\perp$, $\mathbf{x}_h = \mathbf{x}^0 + \mathbf{x}^\perp$, $\mathbf{x}^0, \mathbf{y}^0 \in \boldsymbol{\mathcal{Z}}_h$, $\mathbf{y}^\perp, \mathbf{x}^\perp \in \boldsymbol{\mathcal{X}}_h$, and plugging in the block forms of the operators leads to

$$\begin{pmatrix} \mathbf{y}^\perp - \mathbf{z}_h \\ \mathbf{y}^0 \end{pmatrix} = \begin{pmatrix} Id_\perp & 0 \\ 0 & P_0 \end{pmatrix} \left(\begin{pmatrix} Id_\perp - B_{\perp\!\!\perp} A_\perp & 0 \\ -B_{0\perp}^T A_\perp & Id_0 \end{pmatrix} \begin{pmatrix} Id_\perp - \kappa A_\perp^{-1} M_\perp & 0 \\ 0 & Id_0 \end{pmatrix} + \right.$$
$$\left. + \begin{pmatrix} 0 & \kappa B_{0\perp} M_0 \\ 0 & \kappa B_{00} M_0 \end{pmatrix} \right) \begin{pmatrix} Id_\perp & 0 \\ 0 & P_0 \end{pmatrix} \begin{pmatrix} \mathbf{x}^\perp \\ \mathbf{x}^0 \end{pmatrix}$$

with $\mathbf{z}_h := \kappa A_\perp^{-1} M_\perp \mathbf{x}^\perp$. This results in a kind of *error propagation equation*

$$\begin{pmatrix} \mathbf{y}^\perp - \mathbf{z}_h \\ \mathbf{y}^0 \end{pmatrix} = \begin{pmatrix} Id_\perp - B_{\perp\!\!\perp} A_\perp & \kappa B_{0\perp} M_0 P_0 \\ -P_0 B_{0\perp}^T A_\perp & P_0 (Id_0 + \kappa B_{00} M_0) P_0 \end{pmatrix} \begin{pmatrix} \mathbf{x}^\perp - \mathbf{z}_h \\ \mathbf{x}^0 \end{pmatrix} \,. \tag{4.3}$$

Note that $(\mathbf{z}_h, 0)^T$ is what an exact inverse iteration for the pseudo-inverse would give us before scaling. Thus (4.3) reflects how much the projected preconditioned inverse iteration differs from an exact inverse iteration. Next, we aim at quantitative estimates of this deviation. To this end we seek bounds for norms of the block-operators in (4.3).

Various norms need to be considered for operators $X_h : \boldsymbol{\mathcal{V}}_h \mapsto \boldsymbol{\mathcal{V}}_h$

$$\|X_h\|_{0\to 0} := \sup_{\mathbf{v}_h \in \boldsymbol{\mathcal{V}}_h} \frac{\|X_h \mathbf{v}_h\|_0}{\|\mathbf{v}_h\|_0} \quad , \quad \|X_h\|_{Z\to 0} := \sup_{\mathbf{v}_h \in \boldsymbol{\mathcal{Z}}_h} \frac{\|X_h \mathbf{v}_h\|_0}{\|\mathbf{v}_h\|_0} \,,$$
$$\|X_h\|_{A\to A} := \sup_{\mathbf{v}_h \in \boldsymbol{\mathcal{V}}_h} \frac{\|X_h \mathbf{v}_h\|_A}{\|\mathbf{v}_h\|_A} \quad , \quad \|X_h\|_{A\to 0} := \sup_{\mathbf{v}_h \in \boldsymbol{\mathcal{V}}_h} \frac{\|X_h \mathbf{v}_h\|_0}{\|\mathbf{v}_h\|_A} \,,$$
$$\|X_h\|_{Z\to A} := \sup_{\mathbf{v}_h \in \boldsymbol{\mathcal{Z}}_h} \frac{\|X_h \mathbf{v}_h\|_A}{\|\mathbf{v}_h\|_0} \,.$$

In order to bound the operator norm $\|I - B_{\perp\!\!\!\perp} A_\perp\|_{A\to A}$ we remember that $\|\cdot\|_A$ is the energy-seminorm in $\boldsymbol{H}(\mathbf{curl};\Omega)$ and $\boldsymbol{H}(\mathrm{div};\Omega)$, respectively. In other words, this norm agrees with the convergence rate of the multigrid method in the energy-seminorm. In [38, 40, 41] it was shown that this convergence rate is bounded away from 1 independently of the number L of grid levels involved in the multigrid scheme. This justifies the assumption

$$\|Id_\perp - B_{\perp\!\!\!\perp} A_\perp\|_{A\to A} \le \gamma < 1 \; . \tag{\textbf{A1}}$$

In fact numerical experiments give evidence that we can expect γ to be smaller than 0.5, at worst.

Next, we have to gauge the impact of the inexact projection. Again, we can rely on theoretical results and practical experience with multigrid methods to justify

$$\|G_h(Id_h - C_h T_h)u_h\|_0 \le \beta \|G_h u_h\|_0 \quad \forall u_h \in \mathcal{S}_h \tag{\textbf{A2}}$$

for $\beta < 1$ uniformly in L. Note that β is the convergence rate of the iterative solver in potential space. The practical range for β will be the same as for γ. From (**A2**) and $T_h = G_h^* M_h G_h$ we conclude

$$\lambda_{\max}(G_h(T_h^\dagger - C_h)G_h^* M_h) = \lambda_{\max}(G_h^* M_h G_h(T_h^\dagger - C_h)) = \lambda_{\max}(Id_h - T_h C_h) = \beta \; .$$

Because of $(Id_h - G_h T_h^\dagger G_h^* M_h)\mathbf{x}_h^0 = 0$ for $\mathbf{x}_h^0 \in \boldsymbol{\mathcal{Z}}_h$, this teaches us that

$$\begin{aligned} \left\|P_0 \mathbf{x}_h^0\right\|_0 &= \left\|\mathbf{x}_h^0 - G_h C_h G_h^* M_h \mathbf{x}_h^0\right\|_0 \\ &\le \left\|(Id_h - G_h T_h^\dagger G_h^* M_h + G_h(T_h^\dagger - C_h)G_h^* M_h)\mathbf{x}_h^0\right\|_0 \\ &\le \left\|G_h(T_h^\dagger - C_h)G_h^* M_h \mathbf{x}_h^0\right\|_0 \le \beta \left\|\mathbf{x}_h^0\right\|_0 \; . \end{aligned} \tag{4.4}$$

The remaining terms involving the multigrid preconditioner will be tackled under the restrictive assumption of uniform refinement. That is, we take for granted a geometric decrease of the meshwidths according to $h_l \approx 2^{-l}$.

Under these circumstances, the bilinear form $s_l : \boldsymbol{\mathcal{V}}_l \times \boldsymbol{\mathcal{V}}_l \mapsto \mathbb{R}$ that defines the smoother R_l via

$$s_l(R_l \boldsymbol{\phi}_l, \mathbf{v}_l) = \boldsymbol{\phi}_l(\mathbf{v}_l) \quad \forall \mathbf{v}_l \in \boldsymbol{\mathcal{V}}_l,\ \boldsymbol{\phi}_l \in \boldsymbol{\mathcal{V}}_l'$$

fulfills

$$C h_l^{-2} \left(\mathbf{u}_l, \mathbf{u}_l\right)_0 \le s(\mathbf{u}_l, \mathbf{u}_l) \le C h_l^{-2} \left(\mathbf{u}_l, \mathbf{u}_l\right)_0 \quad \forall \mathbf{u}_l \in \boldsymbol{\mathcal{V}}_l \; .$$

For the point smoothers that we have in mind, this is a consequence of (2.7). In particular, $s(\cdot,\cdot)$ turns out to be positive definite. Then the Cauchy-Schwarz inequality gives for $\mathbf{x}_l^0 \in \boldsymbol{\mathcal{Z}}_l$

$$\begin{aligned} \left\|R_l M_l \mathbf{x}_l^0\right\|_0^2 &\le C h_l^2 s(R_l M_l \mathbf{x}_l^0, R_l M_l \mathbf{x}_l^0) = C h_l^2 \sup_{\mathbf{w}_l \in \boldsymbol{\mathcal{V}}_l} \frac{s(R_l M_l \mathbf{x}_l^0, \mathbf{w}_l)^2}{s(\mathbf{w}_l, \mathbf{w}_l)} \\ &\le C h_l^4 \sup_{\mathbf{w}_l \in \boldsymbol{\mathcal{V}}_l} \frac{\left\langle M_l \mathbf{x}_l^0, \mathbf{w}_l \right\rangle^2}{\left\|\mathbf{w}_l\right\|_0^2} \le C h_l^4 \left\|\mathbf{x}_l^0\right\|_0^2 \; , \end{aligned}$$

from which we infer

$$\|R_l M_l\|_{Z\to 0} \le C h_l^2 \quad \text{and} \quad \|R_l M_l\|_{Z\to A} \le C h_l \ . \tag{4.5}$$

The latter estimate is a consequence of the inverse inequalities (2) that involve

$$a(\mathbf{u}_l, \mathbf{u}_l) \le C h_l^{-2} \|\mathbf{u}_l\|_0^2 \ . \tag{4.6}$$

The same arguments reveal

$$\begin{aligned} \|R_l A_l \mathbf{x}_l\|_0^2 &\le C h_l^2 \, s(R_l A_l \mathbf{x}_l, R_l A_l \mathbf{x}_l) = C h_l^2 \sup_{\mathbf{w}_l \in \boldsymbol{\mathcal{V}}_l} \frac{s(R_l A_l \mathbf{x}_l, \mathbf{w}_l)^2}{s(\mathbf{w}_l, \mathbf{w}_l)} \\ &\le C h_l^4 \sup_{\mathbf{w}_l \in \boldsymbol{\mathcal{X}}_l} \frac{\langle A_l \mathbf{x}_l, \mathbf{w}_l \rangle^2}{\|\mathbf{w}_l\|_0^2} \le C h_l^2 \|\mathbf{x}_l\|_A^2 \ . \end{aligned}$$

The inverse inequality in $\boldsymbol{\mathcal{V}}_l$ is concealed in the final estimate. Eventually,

$$\|R_l A_l\|_{A\to 0} \le C h_l \quad \text{and} \quad \|R_l A_l\|_{A\to A} \le C \ . \tag{4.7}$$

The estimates carry over to R_l^T, of course. Now we are in a position to examine the full multigrid cycle. For the sake of simplicity we confine ourselves to a V(1,1)-cycle:

LEMMA 4.1. *Assume that the smoother alone provides a convergent iteration in the $\|\cdot\|_A$-seminorm. Then*

$$\|B_l A_l\|_{A\to 0} \le K_A \quad , \quad \|B_l M_l\|_{Z\to 0} \le K_0 \quad , \quad \|B_l M_l\|_{Z\to A} \le K_\perp \ ,$$

with constants $K_A > 0$, $K_0 > 0$ and $K_\perp > 0$ that depend on the shape-regularity of the meshes $\mathcal{T}_0, \ldots, \mathcal{T}_L$, but not on l.

Proof. The recursive nature of the multigrid algorithm suggests that we study two subsequent levels l and $l-1$. For ease of notation, we will use a subscript h to refer to level l (fine grid), and H will tag entities associated with level $l-1$ (coarse grid).

We retrace the single steps of the algorithm of figure 3.1 and start with $\boldsymbol{\rho}_h := A_h \mathbf{x}_h$ for some $\mathbf{x}_h \in \boldsymbol{\mathcal{V}}_h$. Presmoothing takes it to $\mathbf{w}_h := R_h A_h \mathbf{x}_h$ since a zero initial guess has to be used. Afterwards, the coarse grid correction will result in $\mathbf{c}_H := B_H I_h^* A_h (\mathbf{x}_h - \mathbf{w}_h)$. Then, with $P_h^H : \boldsymbol{\mathcal{V}}_h \mapsto \boldsymbol{\mathcal{X}}_H$ denoting the $a(\cdot,\cdot)$-orthogonal projection, we infer from $I_h^* A_h = A_H P_h^H$

$$\mathbf{c}_H = B_H A_H P_h^H (Id_h - R_h A_h) \mathbf{x}_h \ .$$

As the multigrid method is supposed to converge in the $\|\cdot\|_A$-seminorm, $\|Id_H - B_H A_H\|_{A\to A} < 1$ is guaranteed, so that

$$\|B_H A_H\|_{A\to A} \le 2 \ . \tag{4.8}$$

The smoother alone also provides a convergent iteration, i.e $\|Id_h - R_h A_h\|_{A\to A} < 1$, such that

$$\|\mathbf{c}_H\|_0 \le \|B_H A_H\|_{A\to 0} \|\mathbf{x}_h\|_A \quad , \quad \|\mathbf{c}_H\|_A \le 2 \|\mathbf{x}_h\|_A \ .$$

With $\mathbf{u}_h := \mathbf{w}_h + I_h \mathbf{c}_H$, which fulfills due to (4.7)

$$\|\mathbf{u}_h\|_0 \le (\|B_H A_H\|_{A\to 0} + Ch) \|\mathbf{x}\|_A \quad , \quad \|\mathbf{u}_h\|_A \le 4 \|\mathbf{x}_h\|_A \ ,$$

we can express the result of post-smoothing as

$$B_h A_h \mathbf{x}_h = \mathbf{u}_h + R_h^T (A_h \mathbf{x}_h - A_h \mathbf{u}_h) = \mathbf{w}_h + \mathbf{u}_h - R_h^T A_h \mathbf{u}_h \ .$$

Again, we invoke (4.7) and see

$$\|B_h A_h \mathbf{x}_h\|_0 \leq (\|B_H A_H\|_{A\to 0} + Ch) \|\mathbf{x}\|_A + Ch \|\mathbf{u}_h\|_A \leq (\|B_H A_H\|_{A\to 0} + Ch) \|\mathbf{x}\|_A \ .$$

Consequently, $\|B_h A_h\|_{A\to 0} \leq \|B_H A_H\|_{A\to 0} + Ch$. Taking into account that $B_0 = A_0^\dagger$, i.e. $\|B_h A_h\|_{A\to 0} = 0$, and the geometric decrease of the meshwidth, this ensures $\|B_h A_h\|_{A\to 0} \leq K_A$, for $K_A > 0$ independent of the level.

Analogous considerations can be carried out with $\boldsymbol{\rho}_h := M_h \mathbf{x}_h^0$ for some $\mathbf{x}_h^0 \in \boldsymbol{\mathcal{Z}}_h$. Presmoothing yields $\mathbf{w}_h := R_h \boldsymbol{\rho}_h$, and after the cycle on the coarse grid we end up with

$$\mathbf{c}_H = \mathbf{c}_1 + \mathbf{c}_2 := B_H I_h^* M_h \mathbf{x}_h^0 + B_H I_h^* A_h \mathbf{w}_h \ .$$

As $I_h^* M_h = M_H Q_h^H$, where $Q_h^H : \boldsymbol{\mathcal{V}}_h \mapsto \boldsymbol{\mathcal{V}}_H$ is the $\boldsymbol{L}^2(\Omega)$-orthogonal projection, we get for the first contribution to $\mathbf{c}_H$

$$\|\mathbf{c}_1\|_0 \leq \|B_H M_H\|_{Z\to 0} \left\|\mathbf{x}_h^0\right\|_0 \quad , \quad \|\mathbf{c}_1\|_A \leq \|B_H M_H\|_{Z\to A} \left\|\mathbf{x}_h^0\right\|_0 \ .$$

Similarly, from $I_h^* A_h = A_H P_h^H$, (4.5) and (4.8) follows that

$$\begin{aligned} \|\mathbf{c}_2\|_0 &\leq \left\|B_H A_H P_h^H \mathbf{w}_h\right\|_0 \leq \|B_H A_H\|_{A\to 0} \|\mathbf{w}_h\|_A \leq Ch \|B_H A_H\|_{A\to 0} \left\|\mathbf{x}_h^0\right\|_0 \ , \\ \|\mathbf{c}_2\|_A &\leq \left\|B_H A_H P_h^H \mathbf{w}_h\right\|_A \leq \|B_H A_H\|_{A\to A} \|\mathbf{w}_h\|_A \leq Ch \left\|\mathbf{x}_h^0\right\|_0 \ . \end{aligned}$$

In sum, based on earlier estimates,

$$\begin{aligned} \|\mathbf{c}_H\|_0 &\leq (\|B_H M_H\|_{Z\to 0} + ChK_A) \left\|\mathbf{x}_h^0\right\|_0 \ , \\ \|\mathbf{c}_H\|_A &\leq (\|B_H M_H\|_{Z\to A} + Ch) \left\|\mathbf{x}_h^0\right\|_0 \ . \end{aligned}$$

Next we consider the coarse grid correction $\mathbf{u}_h = \mathbf{w}_h + I_h \mathbf{c}_H$. As the prolongation is an identity mapping in disguise, the following estimates are straightforward:

$$\begin{aligned} \|\mathbf{u}_h\|_0 &\leq \left(\|B_H M_H\|_{Z\to 0} + ChK_A + Ch^2\right) \left\|\mathbf{x}_h^0\right\|_0 \ , \\ \|\mathbf{u}_h\|_A &\leq (\|B_H M_H\|_{Z\to A} + Ch) \left\|\mathbf{x}_h^0\right\|_0 \ . \end{aligned}$$

The postsmoothing results in

$$B_h M_h \mathbf{x}_h^0 = \mathbf{u}_h + \mathbf{w}_h - R_h A_h \mathbf{u}_h \ .$$

By (4.7) and (4.8) we know

$$\|R_h A_h \mathbf{u}_h\|_0 \leq Ch \|\mathbf{u}_h\|_A \quad , \quad \|R_h A_h \mathbf{u}_h\|_A \leq C \|\mathbf{u}_h\|_A \ .$$

We end up with the estimates

$$\begin{aligned} \left\|B_h M_h \mathbf{x}_h^0\right\|_0 &\leq \left((1+Ch) \|B_H M_H\|_{Z\to 0} + Ch((1+Ch)K_A + Ch^2 + Ch^3\right) \left\|\mathbf{x}_h^0\right\|_0 \ , \\ \left\|B_h M_h \mathbf{x}_h^0\right\|_A &\leq (\|B_H A_H\|_{Z\to A} + Ch) \left\|\mathbf{x}_h^0\right\|_0 \ . \end{aligned}$$

On the coarsest grid $l = 0$, we have $B_{00}M_0 = 0$ and $B_{0\perp}M_0 = 0$. Then the geometric decrease of the meshwidth leads to the assertion of the lemma for $\|B_l M_l\|_{Z\to A}$. For the other norm, we observe

$$\|B_h M_h\|_{Z\to 0} \le (1 + Ch)\,\|B_H M_H\|_{Z\to 0} + Ch\ .$$

In closed form this amounts to

$$\|B_l M_l\|_{Z\to 0} \le C \sum_{i=0}^{l} \prod_{j=i}^{l} (1 + C2^{-j}) \cdot 2^{-i}\ .$$

As $\prod_{j=i}^{l}(1+C2^{-j}) \le \exp(2C)$, the bottom line is that $\|B_l M_l\|_{Z\to 0}$ is uniformly bounded. □

Now, we can convert (4.3) into the estimates

$$\begin{pmatrix} \left\|\mathbf{y}_h^{\perp} - \kappa A_{\perp}^{-1} M_{\perp} \mathbf{x}_h^{\perp}\right\|_A \\ \left\|\mathbf{y}_h^0\right\|_0 \end{pmatrix} \le \begin{pmatrix} \gamma & \kappa K_{\perp}\beta \\ K_A\beta & \beta^2(1+\kappa K_0) \end{pmatrix} \begin{pmatrix} \left\|\mathbf{x}_h^{\perp} - \kappa A_{\perp}^{-1} M_{\perp} \mathbf{x}_h^{\perp}\right\|_A \\ \left\|\mathbf{x}_h^0\right\|_0 \end{pmatrix}\ . \tag{4.9}$$

All the constants are basically independent of the meshwidth and the number L of levels involved in the multigrid solvers. Heuristic insights into the significance of (4.9) can be gained from the theory of preconditioned inverse iteration in the positive definite case [52, 53]. If, with $\kappa = r_{\perp}$, for some positive $\Gamma < 1$

$$\left\|\mathbf{y}_h^{\perp} - \kappa A_{\perp}^{-1} M_{\perp} \mathbf{x}_h^{\perp}\right\|_A \le \Gamma \left\|\mathbf{x}_h^{\perp} - \kappa A_{\perp}^{-1} M_{\perp} \mathbf{x}_h^{\perp}\right\|_A \tag{4.10}$$

the PINVIT convergence theory gives some lengthy, sharp estimate for the Rayleigh quotient $r_{\perp}$ of the new iterate $\mathbf{y}_h^{\perp}$ demonstrating that PINVIT converges at least linearly to the eigenvalue λ_i. Here, we cite only the asymptotically sharp estimate from [51]. In the case of a subspace iteration let θ_j (θ_j') be the j–th Ritz value (ordered by magnitude) of a given (and the next) subspace and let λ_i and λ_{i+1} be the nearest eigenvalues enclosing θ_j. Then

$$\Delta_{i,i+1}(\theta_j') \le \left(\Gamma + (1-\Gamma)\frac{\lambda_i}{\lambda_{i+1}}\right)^2 \Delta_{i,i+1}(\theta_j)\,, \quad \Delta_{i,i+1}(\zeta) := \frac{\zeta - \lambda_i}{\lambda_{i+1} - \zeta}\ . \tag{4.11}$$

In the PINVIT theory, Γ is the spectral radius of its error propagation matrix. For the best multigrid or domain decomposition preconditioners Γ is bounded away from 1 independently on the meshwidth. Having in mind the convergence factor presented in Eq. (4.11), we conclude that PINVIT converges mesh-independently. To illustrate these results consider the discrete Laplacian on $[0,\pi]^2$ whose smallest eigenvalues (with multiplicity) tend to $2, 5, 5, 8, 10, 10, 13, 13, \ldots$. Figure 4.1 displays upper bounds $\Theta(\lambda, \Gamma)$, cf. Theorem 2.1 in [50], for the relative decrease of θ_j' towards the next smaller eigenvalue λ_i, i.e.

$$\frac{\theta_j' - \lambda_i}{\theta_j - \lambda_i} \le \Theta(\lambda, \Gamma).$$

In the semidefinite case Eq. (4.9) applies to PINVIT in $\mathrm{Ker}(A)^{\perp}$ only for the unrealistic choice of a perfect projection, i.e. $\beta = 0$. For a small $\beta > 0$ the iterates

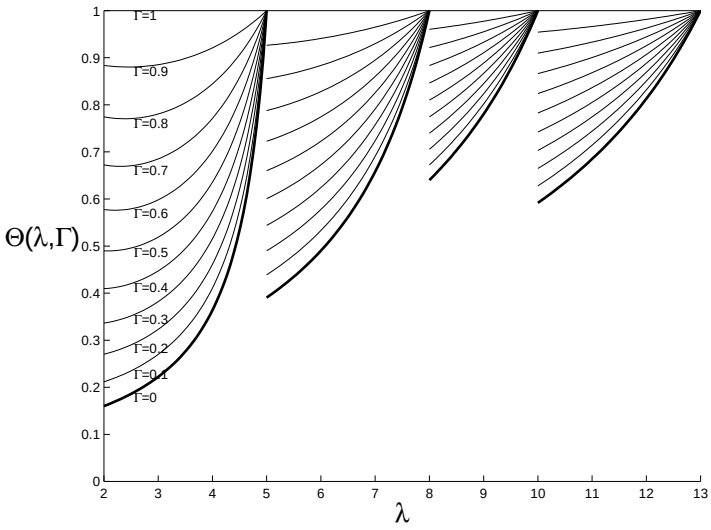

FIG. 4.1. *Estimates* $\Theta(\lambda, \Gamma)$ *for* Δ_h *on* $[0, \pi]^2$.

in $\mathrm{Ker}(A)^\perp$ will inevitably be perturbed by the term $\kappa B_{0\perp} M_0 P_0 \mathbf{x}^0$ in Eq. (4.3). Nevertheless, (4.10) may hold with $\Gamma < 1$ throughout the iteration. Then convergence according to the PINVIT theory is guaranteed (apart from the minor modification of replacing $r_\perp$ by r_Q). Unfortunately, if $\left\|\mathbf{x}_h^\perp - \kappa A_\perp^{-1} M_\perp \mathbf{x}_h^\perp\right\|_A \ll \left\|\mathbf{x}_h^0\right\|_0$ the constant in (4.10) may blow up. However, (4.9) teaches that in this case a significant reduction of the kernel component will be achieved, provided that β is sufficiently small. Hence, $\Gamma > 1$ might happen in a single step, but in the next step (4.10) is likely to hold with a rather small Γ. In other words, for a $\beta \ll 1$ the kernel components are damped out in the course of the iteration. Therefore, the perturbations become more and more insignificant. This effect is elusive and we have not succeeded in giving a rigorous analysis.

Let us study as a model system the eigenvalue problem for $A_h = \mathrm{diag}(2, 5, 8, 0)$ with $\gamma = 0.5$. Then the preconditioner is a 4×4 matrix. We take $b_{\max}$ as the bound for the absolute value of B_{00} and for the A–norm of $B_{0\perp}$. Note that the low dimension of the model problem is motivated by the fact that preconditioned inverse iteration takes its extremal convergence in a 2D space which is spanned by those eigenvectors whose corresponding eigenvalues enclosing the Rayleigh quotient of the actual iterate. Moreover, as a result of [52] the assumption that all eigenvalues are of the algebraic multiplicity 1 is non-restrictive. In figure 4.2

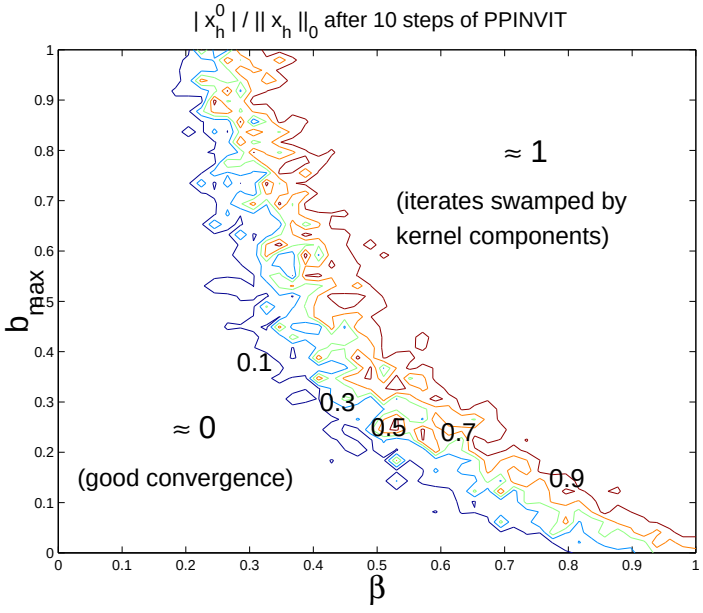

FIG. 4.2. *Relative damping of the kernel component for the model problem.*

the maximal ratio $|x_h^0|/\|x_h\|_0$ after 10 steps of PPINVIT is displayed in a contour plot for $\beta \in [0,1]$ and $b_{\max} \in [0,1]$. For each point of the underlying 50×50 mesh 150000 combinations of random preconditioners and random start vectors with a fixed initial kernel component have been tested. For $\beta < 0.2$ the kernel is damped out very well independently on the choice of $b_{\max}$.

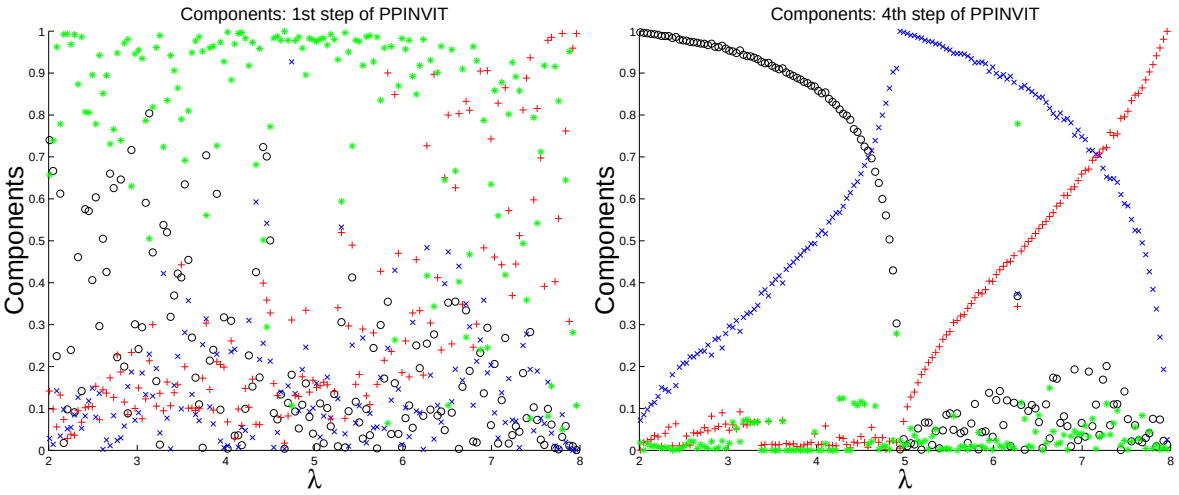

FIG. 4.3. *Components of the iterate x_h for the model problem. Components to $(\lambda_1, \lambda_2, \lambda_3, 0)$ correspond to symbols $(o, x, +, *)$. Left: 1st step of PPINVIT Right: 4th step of PPINVIT.*

The components of the iterates within the 1st and 4th step of PPINVIT in the case of poorest convergence can be looked up in figure 4.3. For 10^7 combinations of random preconditioners ($\gamma = 0.5$, $\beta = 0.25$, $b_{\max} = 0.2$) the components of those vectors which are responsible for the poorest decrease of the Rayleigh quotient are displayed against their two-step Rayleigh quotient. While in the first step of PPINVIT the kernel components appear as the dominating part, we identify in the 4th step within the interval $[\lambda_i, \lambda_{i+1}]$ the ith and $i+1$th components as the prevailing ones. Such a result does not come as a surprise, and can be understood by the distinctive feature of preconditioned inverse iteration to take its extremal convergence in exactly this 2D space. The kernel has all but disappeared. Thus, figure 4.3 highlights a key trait of PPINVIT: Convergence is brought about by the subtle interaction of multiple steps.

Some quantitative conclusions can also be drawn from (4.9).

THEOREM 4.2. *If $\kappa := \kappa_Q < \kappa^*$ for all steps of the iteration and the initial iterate $\mathbf{x}_h$ satisfies*

$$\frac{\left\|\mathbf{x}_h^0\right\|_0}{\left\|\mathbf{x}_h\right\|_A} \leq \frac{(1-\gamma)}{2\kappa^* K_\perp}$$

then this will hold for all other iterates, provided that the approximate projection is sufficiently accurate.

Proof. Given $\mathbf{x}_h = \mathbf{x}_h^0 + \mathbf{x}_h^\perp \in \boldsymbol{\mathcal{V}}_h$, $\mathbf{x}_h^0 \in \boldsymbol{\mathcal{Z}}_h$, $\mathbf{x}_h^\perp \in \boldsymbol{\mathcal{X}}_h$, computing $\mathbf{y}_h = \mathbf{y}_h^0 + \mathbf{y}_h^\perp$ according to (4.3), we obtain from (4.9)

$$\begin{aligned}
\left\|\mathbf{y}_h^\perp\right\|_A &\geq \left\|\mathbf{z}_h\right\|_A - \gamma \left\|\mathbf{x}_h^\perp - \mathbf{z}_h\right\|_A - \kappa^* K_\perp \beta \left\|\mathbf{x}_h^0\right\|_0 \ , \\
\left\|\mathbf{y}_h^0\right\|_0 &\leq K_A \beta \left\|\mathbf{x}_h^\perp - \mathbf{z}_h\right\|_A + \beta^2 (1 + \kappa^* K_0) \left\|\mathbf{x}_h^0\right\|_0 \ ,
\end{aligned}$$

where $\mathbf{z}_h := r_Q(\mathbf{x}_h) A_\perp^{-1} M_\perp \mathbf{x}_h^\perp$. This definition immediately implies

$$\begin{aligned}\left\|\mathbf{z}_h - \mathbf{x}_h^\perp\right\|_A^2 &= \|\mathbf{z}_h\|_A^2 - 2\left\langle A_\perp \mathbf{z}_h^\perp, \mathbf{x}_h^\perp\right\rangle + \left\|\mathbf{x}_h^\perp\right\|_A^2 \\ &= \|\mathbf{z}_h\|_A^2 - (2\frac{r_Q(\mathbf{x}_h)}{r_\perp(\mathbf{x}_h)} - 1)\left\|\mathbf{x}_h^\perp\right\|_A^2 \ .\end{aligned}$$

Owing to the Cauchy-Schwarz-inequality, we have $r_Q(\mathbf{x}_h) \geq r_\perp(\mathbf{x}_h)$ and thus

$$\left\|\mathbf{z}_h - \mathbf{x}_h^\perp\right\|_A \leq \|\mathbf{z}_h\|_A \quad , \quad \|\mathbf{z}_h\|_A \geq \left\|\mathbf{x}_h^\perp\right\|_A \tag{4.12}$$

The above estimates can be blended into

$$\begin{aligned}\frac{\left\|\mathbf{y}_h^0\right\|_0}{\left\|\mathbf{y}_h^\perp\right\|_A} &\leq \frac{K_A\beta\left\|\mathbf{x}_h^\perp - \mathbf{z}_h\right\|_A + \beta^2(1+\kappa^* K_0)\left\|\mathbf{x}_h^0\right\|_0}{\|\mathbf{z}_h\|_A - \gamma\left\|\mathbf{x}_h^\perp - \mathbf{z}_h\right\|_A - \kappa^* K_\perp \beta\left\|\mathbf{x}_h^0\right\|_0} \\ &\leq \frac{K_A\beta + \beta^2(1+\kappa^* K_0)\left\|\mathbf{x}_h^0\right\|_0/\|\mathbf{z}_h\|_A}{(1-\gamma) - \kappa^* K_\perp \beta\|\mathbf{x}_h^0\|_0/\|\mathbf{z}_h\|_A}\end{aligned}$$

From (4.12) and the assumptions of the theorem, we conclude

$$\frac{\left\|\mathbf{x}_h^0\right\|_0}{\|\mathbf{z}_h\|_A} \leq \frac{(1-\gamma)}{2\kappa^* K_\perp \beta} \quad \Longrightarrow \quad (1-\gamma) - \kappa^* K_\perp \beta \frac{\left\|\mathbf{x}_h^0\right\|_0}{\|\mathbf{z}_h\|_A} \geq \tfrac{1}{2}(1-\gamma) \ ,$$

which implies

$$\frac{\left\|\mathbf{y}_h^0\right\|_0}{\left\|\mathbf{y}_h^\perp\right\|_A} \leq \beta\left(\frac{2K_A}{1-\gamma} + \frac{1+\kappa^* K_0}{\kappa^* K_\perp}\right) \ .$$

The ratio $\left\|\mathbf{y}_h^0\right\|_0 / \left\|\mathbf{y}_h^\perp\right\|_A$ will remain below the threshold $(1-\gamma)/2\kappa^* K_\perp \beta$, if

$$\beta^2 \leq \frac{(1-\gamma)^2}{4\kappa^* K_\perp K_A + (1-\gamma)(1+\kappa^* K_0)} \ .$$

This states the condition on the accuracy of the approximate projection. As the ratio in the statement of the theorem is not affected by scaling, the proof is finished. □

The previous theorem guarantees that the iterates cannot plunge into the kernel, if a sufficient damping of kernel components is achieved by the projection.

5. Projection control and termination criteria. The theoretical considerations highlight the importance of a good projection: It goes without saying that the method will fail, if the projection is too weak to reign in kernel components. Taking the cue from theorem 4.2, we aim to force the ratio $\|\mathbf{x}_h^0\|_0 : \|\mathbf{x}_h\|_A$ below a threshold $\delta > 0$ for all iterates.

From the properties of the inexact projection $\widetilde{P}_h$ and (4.4) we learn that

$$\frac{\beta}{1-\beta} \cdot \frac{\|\mathbf{x}_h - \widetilde{P}\mathbf{x}_h\|_0}{\|\mathbf{x}_h\|_A} \leq \delta \quad \Longrightarrow \quad \frac{\|P_0 \mathbf{x}_h\|_0}{\|\mathbf{x}_h\|_A} \leq \delta \ . \tag{5.1}$$

Of course, good bounds for β are hard to get. We take a crude estimate based on the decrease of the $L^2(\Omega)$-norm of the residual during a multigrid sweep. It is computed whenever a projection is carried out, and β is chosen to be the maximum of all estimates thus obtained. The final adaptive projection is depicted in figure 5.1. There, $\sigma \in]0,1[$ is a safety factor intended to prevent gross underestimation of β.

```
project(reference x_h ∈ V_h, δ > 0, σ ∈ [0,1[)
{
  for(int i = 1; i ≤ q; ++i)   { x_h ← x_h − (h̃_i, x_h)_0 · h̃_i }
  ν := ⟨A_h x_h, x_h⟩
  do   {
    φ_h := G_h^* M_h x_h;   c_h = 0 ∈ S_h
    mgcycle< T >(L, c_h, φ_h);   z_h := G_h c_h;   x_h ← x_h − z_h
    ρ_h := φ_h − T_h c_h;   β := ⟨φ_h, φ_h⟩ / ⟨ρ_h, ρ_h⟩;   β̄ ← max{β̄, (1 − σ)β + σ}
    μ := β̄/(1 − β̄) · ⟨M_h z_h, z_h⟩/ν
  }
  while (μ > δ);
}
```

FIG. 5.1. *Enhanced projection with adaptive control. The global variable $\bar{\beta}$ is set to 0 initially.*

Our next concern is the termination of the iteration. After the completion of the Rayleigh Ritz procedure, cf. figure 3.3, there are on hand the Ritz values θ_i and the Ritz vectors $\mathbf{x}_h^i$ with $\left\|\mathbf{x}_h^i\right\|_0 = 1$, $i = 1, \ldots, s$. The M_h^{-1}–norm of the residual $\mathbf{r}_h^i = A_h\mathbf{x}_h^i - \theta_i M_h\mathbf{x}_h^i$ provides a simple residual bound [54] for the quality of the Ritz value θ_i. It is guaranteed that in each interval $[\theta_i - \left\|\mathbf{r}_h^i\right\|_{M_h^{-1}}, \theta_i + \left\|\mathbf{r}_h^i\right\|_{M_h^{-1}}]$ an eigenvalue of (A_h, M_h) is contained. For disjoint intervals the θ_i provide s approximations to s different eigenvalues of (A_h, M_h). In practice the inverse of the mass matrix may be approximated through one Gauß-Seidel step. This yields a quantity that is equivalent to the M_h^{-1}-norm independent of the meshwidth.

Beyond, we suggest that the ratio r/r_Q of Rayleigh-quotients of approximate eigenfunctions is used to judge whether the iteration has been successful. Only if it is very close to 1 the results can be trusted.

Remark. It is not a moot point that δ should be reduced during the iteration as the approximate eigenvectors get closer and closer to the exact eigenvectors. However, we failed to find a strategy with heuristic, let alone rigorous, underpinning.

6. Numerical experiments. For all numerical experiments covered in this section we relied on lowest order edge/face elements on uniform Cartesian grids. Ritz projections and eigenvalues/eigenfunctions on the coarsest grids were determined by means of suitable LAPACK routines. All computations were carried out in double precision arithmetic, whereas the matrices were stored in single precision format. Homogeneous Dirichlet boundary conditions were imposed throughout. We computed 7 eigenfunction/eigenvalue pairs in each case.

For the tests we resorted to three different settings: **Setting A** used the unit cube $\Omega =]0,1[^3$ and constant coefficient $\alpha \equiv 1$. The uniform grid on level l, $l = 0, \ldots, 5$ consisted of $27 \cdot 8^l$ equal cubes. This means that for $l = 5$ the discretized problems (1.1) and (1.2) feature 2599200 and 2626560 degrees of freedom, respectively.

Setting B sports the discontinuous coefficient

$$\alpha(\mathbf{x}) = \begin{cases} 100 & \text{if } |\mathbf{x} - (\frac{1}{3}, \frac{1}{3}, \frac{1}{3})^T| \leq \frac{1}{3}\ , \\ 1 & \text{elsewhere}\ , \end{cases}$$

and retains the unit cube as computational domain. The grids are the same as for setting A. We study this situation, because discontinuous coefficients are notorious for affecting multigrid convergence.

In **setting C** we set $\alpha \equiv 1$, but used the "L-shaped" computational domain $\Omega :=]0;1[^3 \backslash ([0,\frac{1}{2}] \times [0,\frac{1}{3}] \times [0,\frac{1}{2}])$ equipped with a coarsest Cartesian grid with meshwidth $\frac{1}{2}$ in x_1- and x_3-direction, and meshwidth $\frac{1}{3}$ in x_2-direction, leading to 1048800 edges that bear degrees of freedom on level 5.

Nested iteration with a tight termination threshold (see below) was used to obtain the "exact discrete eigenvalues" for each setting. Those for problem (1.1) are listed in table 6.1.

Setting	Level	#1	#2	#3	#4	#5	#6	#7
A	$l=3$	19.76	19.76	19.76	29.65	29.65	49.58	49.58
	$l=4$	19.74	19.74	19.74	29.61	29.61	49.40	49.40
	$l=5$	19.74	19.74	19.74	29.61	29.61	49.36	49.36
B	$l=3$	20.02	20.02	20.02	29.71	29.71	49.76	49.76
	$l=4$	20.00	20.00	20.00	29.68	29.68	49.58	49.58
	$l=5$	19.99	19.99	19.99	29.67	29.67	49.53	49.53
C	$l=3$	15.19	22.77	22.91	34.35	35.14	44.58	49.80
	$l=4$	15.17	22.71	22.85	34.25	35.01	44.21	49.35
	$l=5$	15.11	22.70	22.83	34.23	34.97	44.11	49.20

TABLE 6.1

The seven smallest nonzero "exact discrete eigenvalues" of problem (1.1) *computed by nested iteration (cf. experiment 2)*

Experiment 1. To begin with, we monitored the behavior of the "two-step" Rayleigh quotients r_Q from (3.8) and the M_l^{-1}-norms $\left\langle M_l^{-1}\boldsymbol{\rho}_l, \boldsymbol{\rho}_l \right\rangle^{\frac{1}{2}}$ of the residuals $\boldsymbol{\rho}_l := A_l \mathbf{x}_l - r_Q(\mathbf{x}_l) M_l \mathbf{x}_l$ of approximate eigenfunctions $\mathbf{x}_l \in \boldsymbol{\mathcal{V}}_l$, $l = 3, 4, 5$. Of course, $M_l^{-1}\boldsymbol{\rho}_l$ could not be computed exactly, but was realized by two Gauß-Seidel-sweeps (GS). Both quantities were tracked for 3 eigenfunctions (belonging to eigenvalues #1,#3, and #5) during 15 iterations of PPINVIT. Random grid functions with their degrees of freedoms uniformly distributed in $[0,1]$ served as initial guesses for the eigenfunctions. We observed no qualitative differences between the data recorded in different runs of the code.

Single symmetric multigrid V(1,1)-cycles with lexicographic GS-smoothers were used both in $\boldsymbol{\mathcal{V}}_l$ and potential space. The inverse mass matrix required for the calculation of the two-step Rayleigh quotient was approximated by three steps of the preconditioned conjugate gradient (PCG) method with a symmetric GS-preconditioner.

The values of r_Q were considered as useful approximations of eigenvalues and thus it makes sense to examine their relative errors with respect to the "exact discrete eigenvalues" from table 6.1. The results for problem (1.1) are plotted in figures 6.1, 6.2.

First of all, after the effect of the random initial guesses has abated, a rather uniform decrease of the errors/residual norms takes place. Next, we note that the M_l^{-1}-norm of the residual permits us to assess the accuracy of the approximate eigenvalue very well. As expected, the larger the eigenvalue the poorer the convergence (up to a total failure to converge for the 7th eigenvalue). One should follow the customary advice that dimension of the subspace should be chosen somewhat larger than the number of eigenvalue one is interested in.

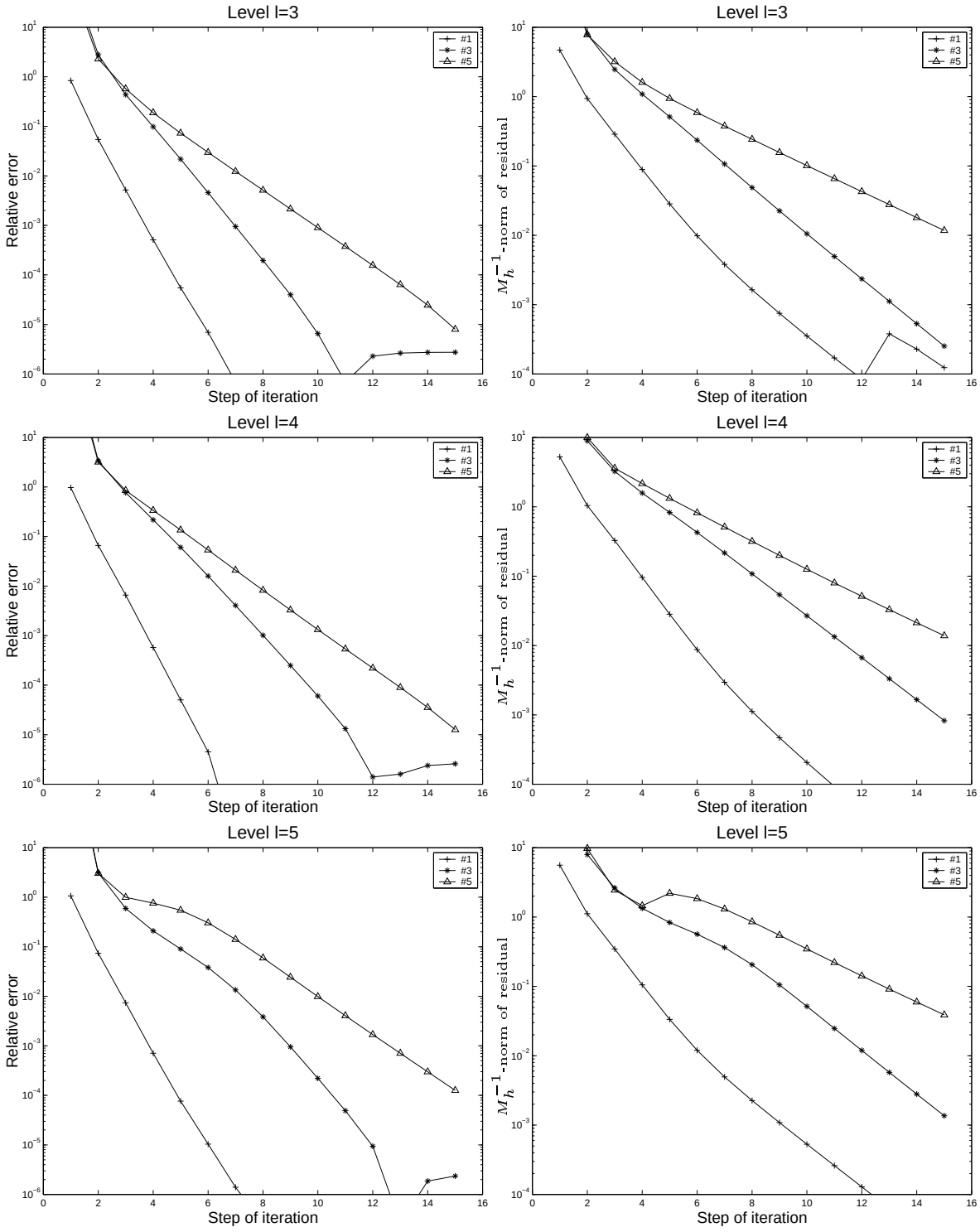

FIG. 6.1. *Experiment 1, setting A, problem* (1.1). *Left: Relative errors of two-step Rayleigh quotients when compared with "exact discrete eigenvalues". Right: Approximate* M_l^{-1}*-norms of residuals*

Experiment 2. Of course, choosing random initial guesses is foolish, in particular, as a nested iteration approach will do much better in a multilevel environment. The behavior of the approximate M_l^{-1}-norms of the eigenfunction residuals during nested iteration was recorded for the various settings. On each level l the iteration was terminated, if $\left\langle M_l^{-1}\boldsymbol{\rho}_l, \boldsymbol{\rho}_l \right\rangle \leq \tau$ for eigenfunction #1 through #5, where $\tau > 0$ is a prescribed threshold.

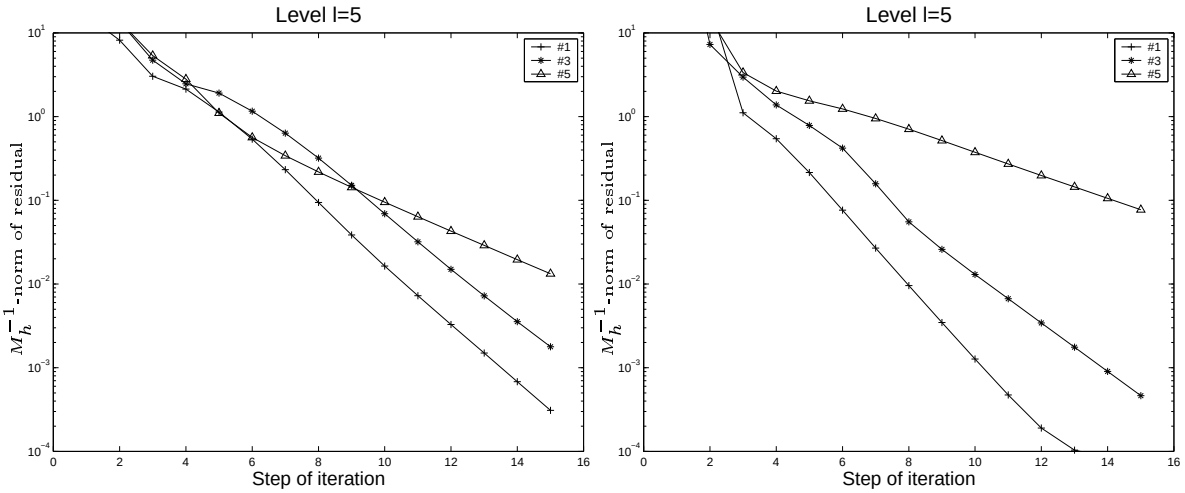

FIG. 6.2. *Experiment 1, settings B(left) and C(right), problem* (1.1)*: Approximate* M_l^{-1}*-norms of residuals*

The same multigrid cycles as before were employed. Moreover, we chose $\tau = 10^{-4}$ and $\tau = 10^{-3}$. In the latter case the evaluation of M_h^{-1} in the computation of r_Q was based on only one symmetric GS-sweep, which is much cheaper than the three PCG-steps used for the former case. The results can be looked up in figures 6.3-6.6.

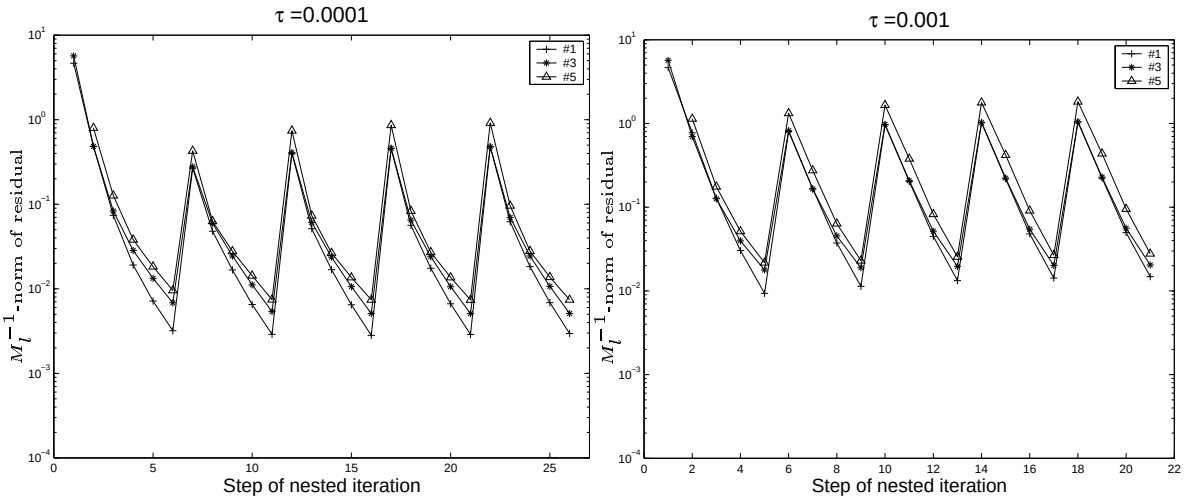

FIG. 6.3. *Experiment 2, setting A, edge elements: Nested iteration (Left:* M_h^{-1} *realized by 3 PCG-steps. Right:* M_h^{-1} *realized by 1 GS-sweep.)*

The data strikingly confirm that the convergence of multigrid-PPINVIT is independent of the depth of refinement: About the same number of iterations is required on each level to achieve the prescribed reduction of the norm of the residuals.

Experiment 3. Next, we studied the impact of choices of different multigrid cycles for both the update and projection step in setting A. Everything else is like in the first experiment. In table 6.2 we report the rate of convergence of the eigenvalue approximations between the 3rd and 5th step of the iteration

$$\rho = \sqrt{\frac{r_Q(\mathbf{x}_h^{(5)}) - \lambda_{\text{exact}}}{r_Q(\mathbf{x}_h^{(3)}) - \lambda_{\text{exact}}}}\,. \tag{6.1}$$

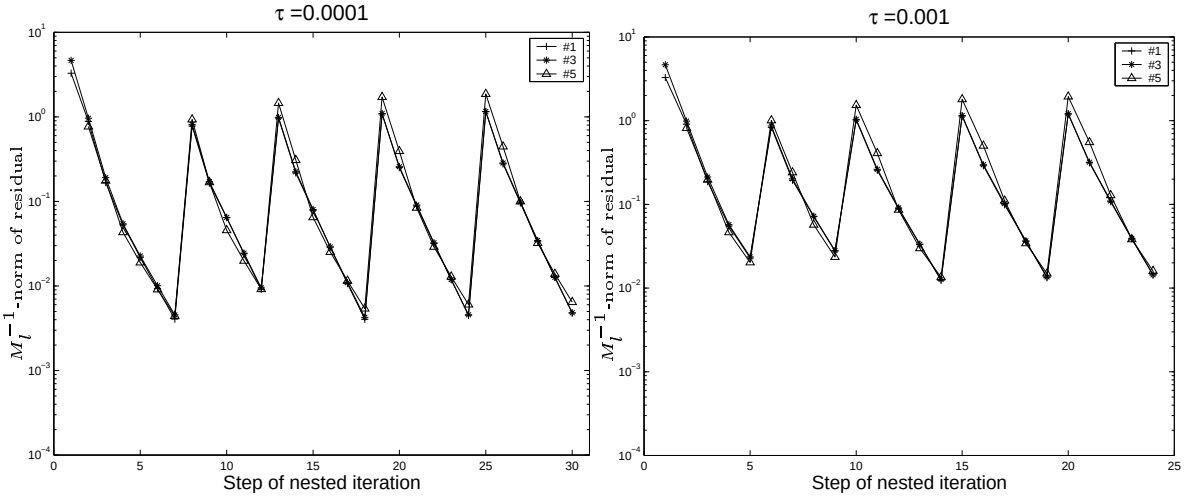

FIG. 6.4. *Experiment 2, setting A,* ***face*** *elements: Nested iteration (Left:* M_l^{-1} *realized by 3 PCG-steps. Right:* M_l^{-1} *realized by 1 GS-sweep.)*

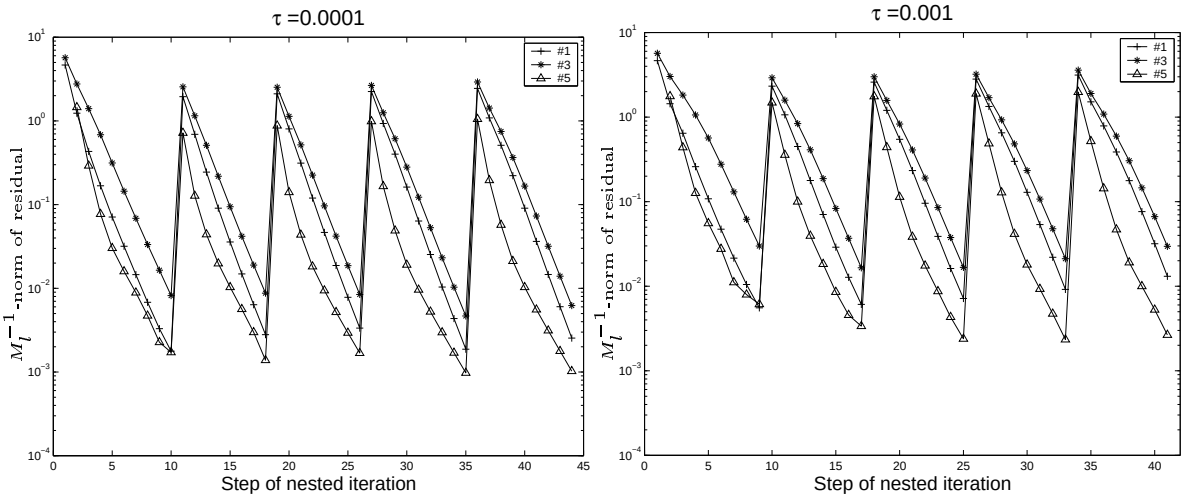

FIG. 6.5. *Experiment 2, setting B, edge elements: Nested iteration (Left:* M_l^{-1} *realized by 3 PCG-steps. Right:* M_l^{-1} *realized by 1 GS-sweep.)*

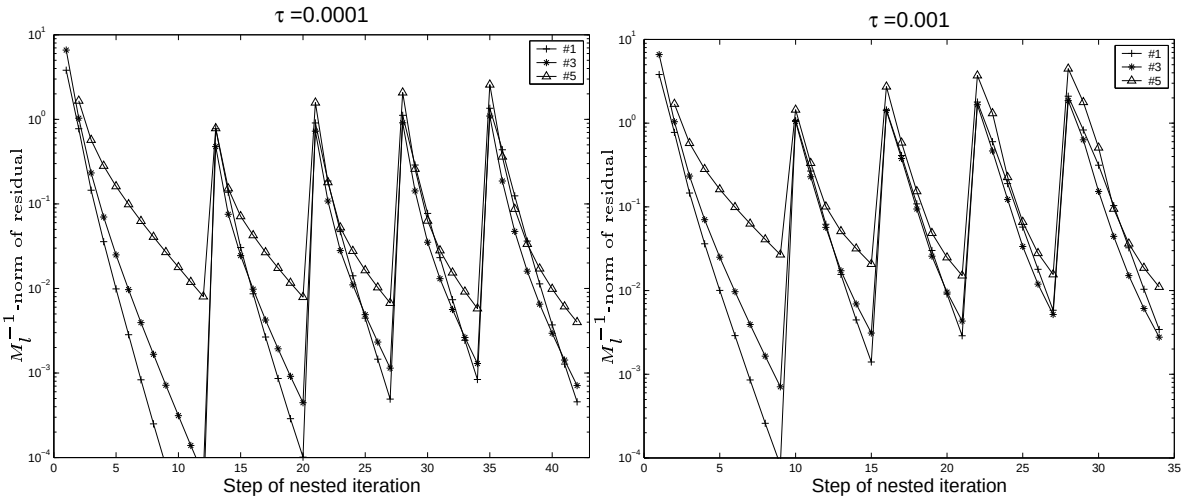

FIG. 6.6. *Experiment 2, setting C, edge elements: Nested iteration (Left:* M_l^{-1} *realized by 3 PCG-steps. Right:* M_l^{-1} *realized by 1 GS-sweep.)*

Level	B_h	C_h	#1	#2	#3	#4	#5	#6	#7
$l=3$	V(1,1)	V(1,1)	0.10	0.17	0.22	0.24	0.35	0.45	0.47
	W(2,2)	V(1,1)	0.03	0.12	0.16	0.21	0.26	0.41	0.61
	V(1,1)	W(2,2)^2	0.09	0.20	0.38	0.30	0.41	0.38	0.65
$l=4$	V(1,1)	V(1,1)	0.08	0.21	0.28	0.24	0.39	0.44	0.45
	W(2,2)	V(1,1)	0.03	0.09	0.12	0.26	0.30	0.46	0.57
	V(1,1)	W(2,2)^2	0.10	0.16	0.31	0.29	0.29	0.45	0.42
$l=5$	V(1,1)	V(1,1)	0.10	0.19	0.38	0.24	0.74	0.42	0.41
	W(2,2)	V(1,1)	0.03	0.11	0.26	0.22	0.45	0.51	0.56
	V(1,1)	W(2,2)^2	0.10	0.16	0.19	0.35	0.43	0.39	0.49

TABLE 6.2

Experiment 3, setting A, edge elements: Different rates of convergence ρ according to (6.1) *for eigenvalue approximations.*

The effect of very accurate preconditioners/projections seems to be limited, as predicted by the theory of PINVIT: As can be seen from figure 4.1 we cannot be better than exact inverse iteration and, for instance, decreasing γ from 0.3 to 0.1 has little impact.

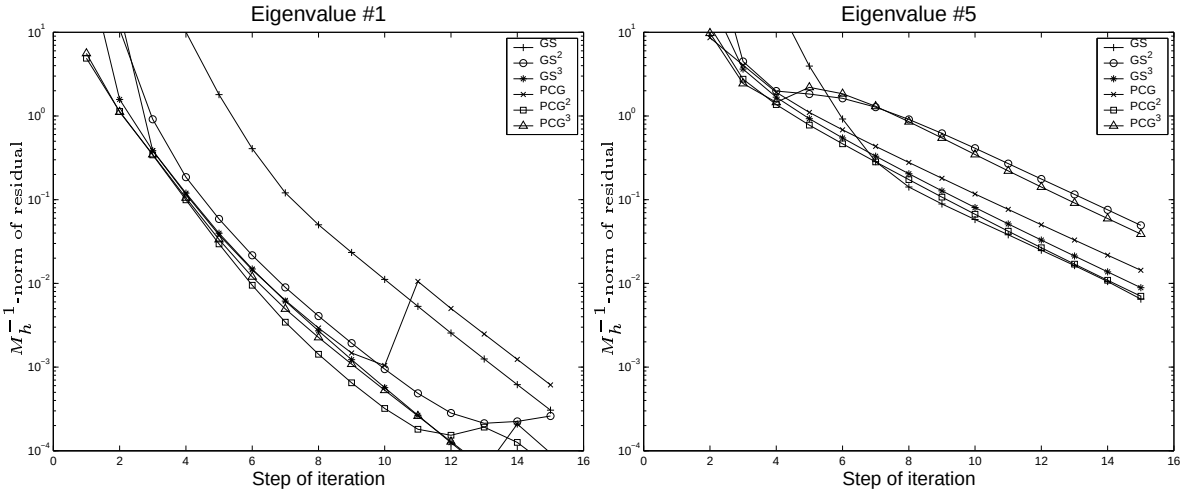

FIG. 6.7. *Experiment 4: Impact of approximation of* M_h^{-1}

Experiment 4. In experiment 2 we boldly relied on a single symmetric GS-sweep to get an approximation for r_Q. Now, we aim to investigate, how different approximations of M_h^{-1} perform in setting A. In particular, we used either one, two, or three steps of GS or PCG. The remainder of the algorithm is just borrowed from experiment 1. All computations were conducted on level 5 and for edge elements and the results are displayed in figure 6.7. The message is that spending much effort on M_h^{-1} does not pay off.

Experiment 5. The final experiment scrutinizes whether projection control as dicussed in section 5 can really offset poor projections. To that end we used a plain symmetric Gauß-Seidel sweep for C_h, which yields an outrageously bad $\widetilde{P}_h$ on fine grids. Otherwise, the algorithm of the first experiment was retained and we focused on level 4.

Projection control with $\delta = 0.05$, $\delta = 0.01$ and a safety factor $\sigma = \frac{1}{4}$ was enabled. In addition, as we observed wild fluctuation of the number of GS-steps suggested by

the projection control, we imposed that this number could not shrink by more than a factor of two between to subsequent projections (zig-zag-evasion). In figures 6.8, 6.8, and 6.10 the behavior of relative errors of eigenvalues and the norms of eigenfunction residuals were logged. Some ratios $r(\mathbf{x}_h) : r_Q(\mathbf{x}_h)$ are recorded in figure 6.11. Number of GS sweeps enforced by the projection control are plotted in figure 6.12.

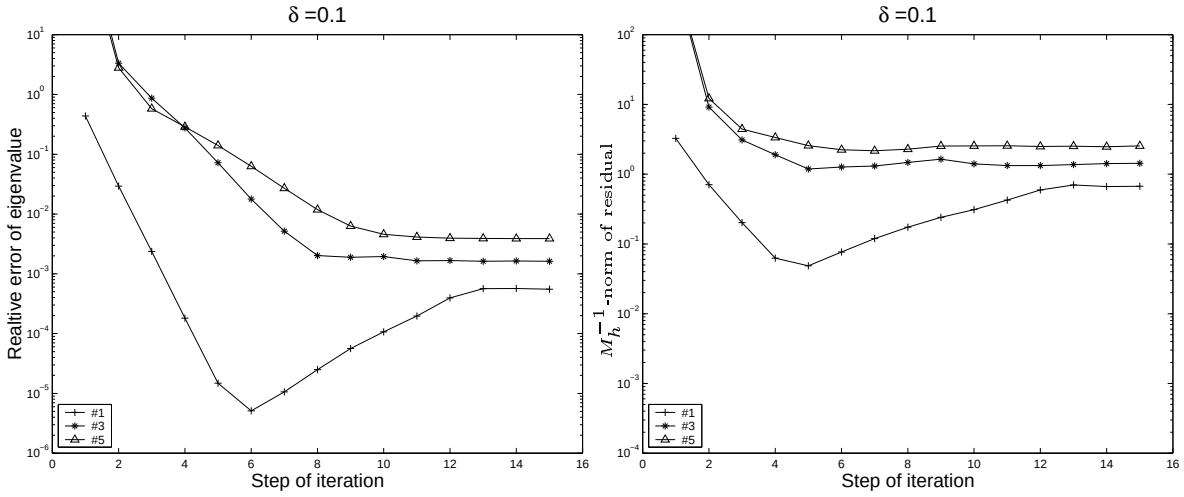

FIG. 6.8. *Experiment 5: Projection control with* $\delta = 0.1$

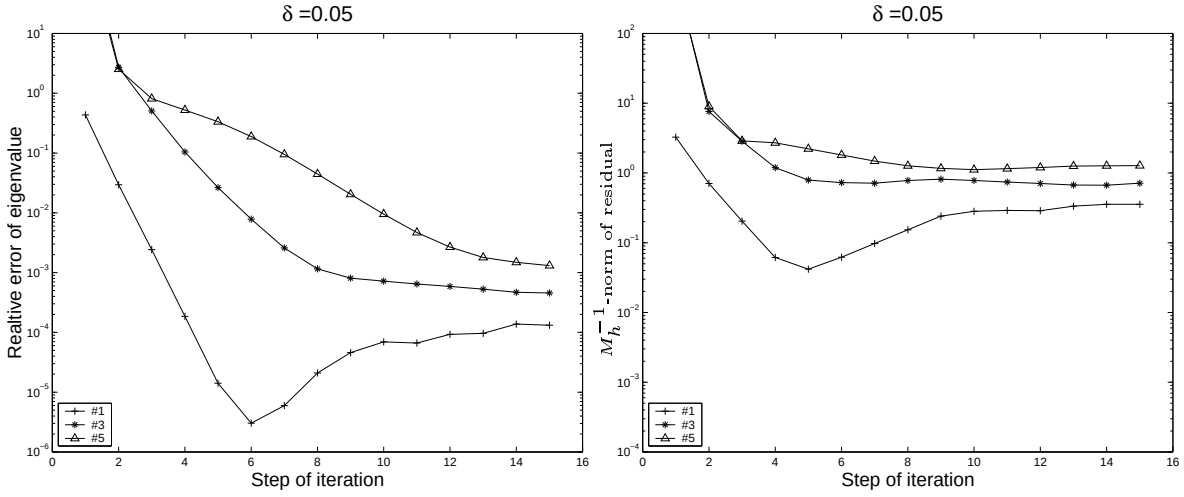

FIG. 6.9. *Experiment 5: Projection control with* $\delta = 0.05$

Sufficiently tight projection control ensures convergence. However, the results also highlight the need for an adaptive choice of δ, because it seems hard to determine in advance, when δ will be sufficiently small. This experiment also hints that the ratio of Rayleigh quotients, which is to tend to 1, can help detect ineffective projections.

Finally, the enhanced robustness of the method due to the choice of the two-step Rayleigh quotient is conveyed in figure 6.13. If we choose $\lambda = r(\mathbf{x}_h)$ with r from (3.1) the method will be way more sensitive to poor projections.

7. Conclusion. We presented a multigrid-preconditioned inverse iteration method for the solution of large discrete semidefinite eigenvalue problems in $\boldsymbol{H}(\mathbf{curl};\Omega)$ and $\boldsymbol{H}(\mathrm{div};\Omega)$. Though a complete theoretical analysis is still missing, there is strong numerical evidence that the method inherits the efficiency of multi-

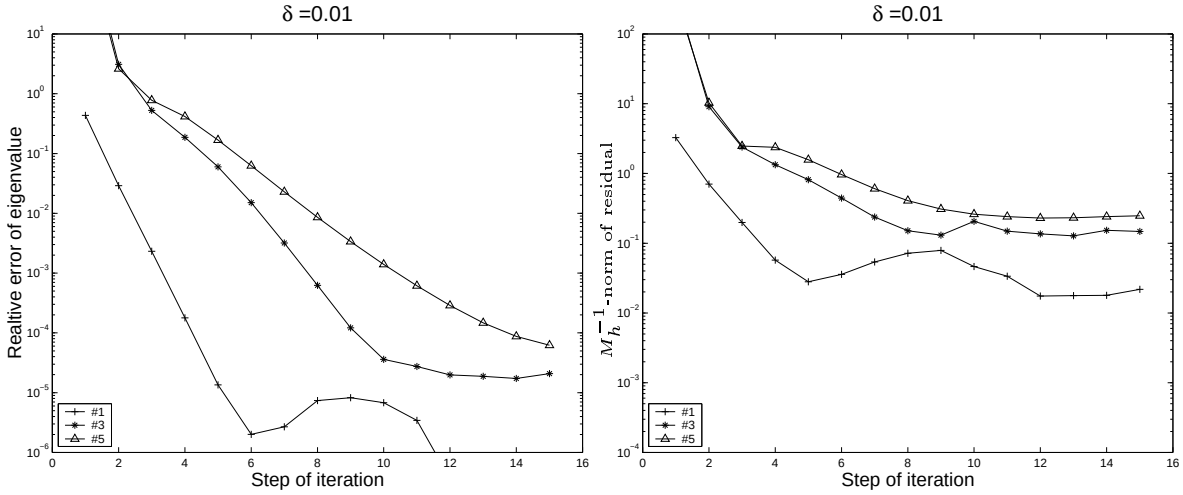

FIG. 6.10. *Experiment 5: Projection control with* $\delta = 0.01$

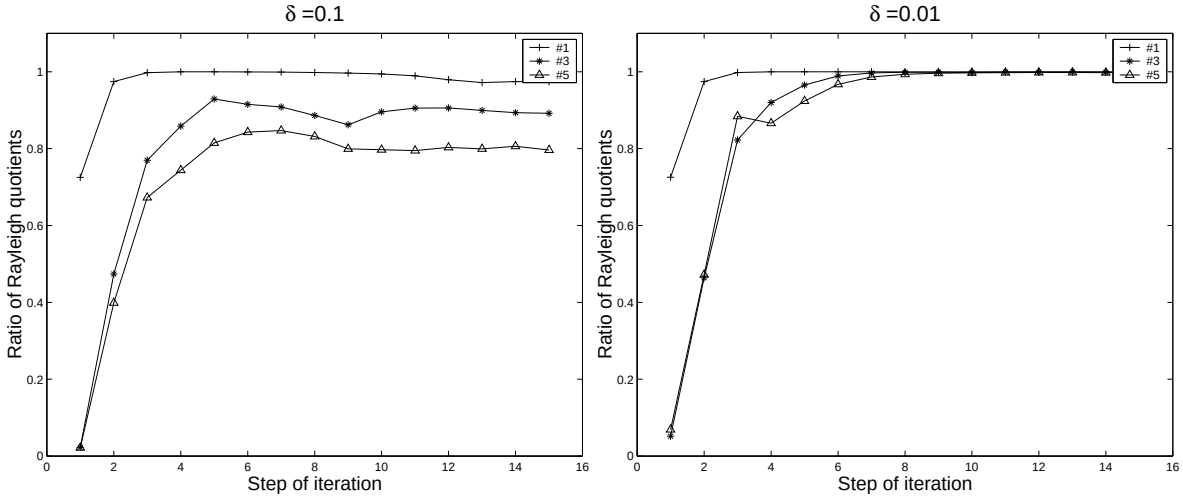

FIG. 6.11. *Experiment 5: Ratios of Rayleigh quotients in the case of loose* ($\delta = 0.1$) *and tight* ($\delta = 0.01$) *projection control.*

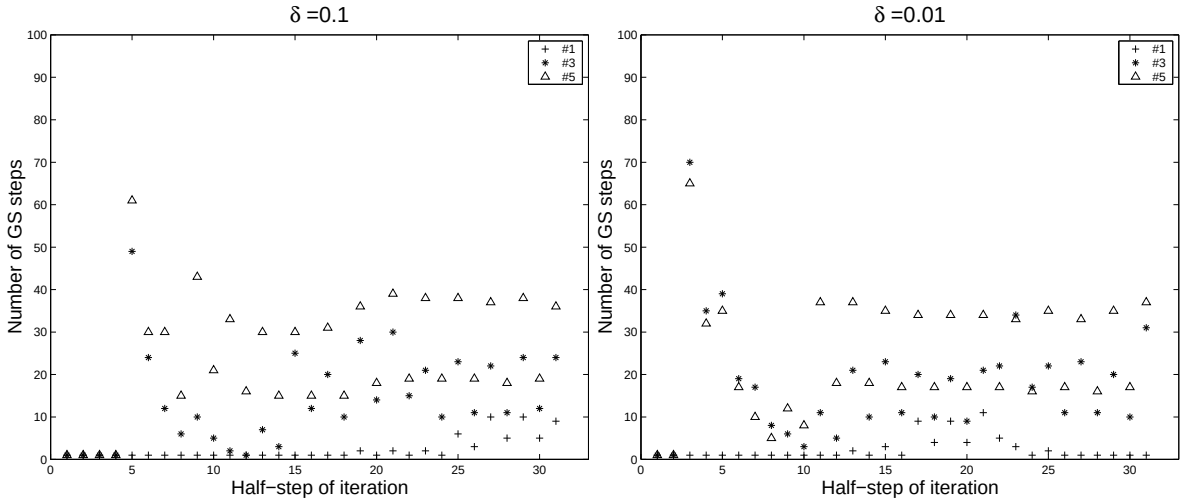

FIG. 6.12. *Experiment 5: Numbers of GS-sweeps for projection*

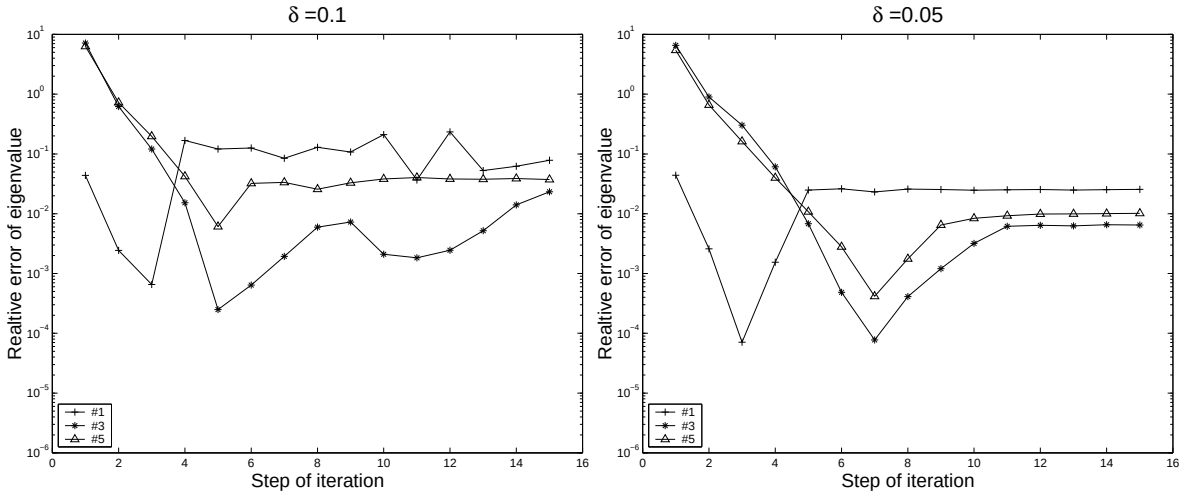

FIG. 6.13. *Experiment 5: Convergence when using standard Rayleigh quotient from* (3.1) *for* λ

grid based iterative solution procedures. Besides a thorough theoretical understanding many issues remain to be examined: Among others, improved adaptive projection control, detection of topological complications, and potential acceleration by means of preconditioned steepest descent or various kinds of subspace enlargements and, last but not least, the benefit of shift strategies.

REFERENCES

[1] S. ADAM, P. ARBENZ, AND R. GEUS, *Eigenvalue solvers for electromagnetic fields in cavities*, Tech. Rep. 275, Institute of Scientific Computing, ETH Zürich, Zürich, Switzerland, October 1997.

[2] C. AMROUCHE, C. BERNARDI, M. DAUGE, AND V. GIRAULT, *Vector potentials in three–dimensional nonsmooth domains*, Math. Methods Appl. Sci., 21 (1998), pp. 823–864.

[3] D. ARNOLD, R. FALK, AND R. WINTHER, *Multigrid in H*(div) *and H*(**curl**), Numer. Math., 85 (2000), pp. 175–195.

[4] Z. BAI, J. DEMMEL, J. DONGARRA, A. RUHE, AND H. VAN DER VORST, eds., *Templates for the solution of algebraic eigenvalue problems: a practical guide*, SIAM, Phildelphia, 2000.

[5] R. BANK, *Analysis of a multilevel inverse iteration procedure for eigenvalue problems*, SIAM J. Numer. Anal., 19 (1982), pp. 886–898.

[6] E. BÄNSCH, *Local mesh refinement in 2 and 3 dimensions*, IMPACT Comput. Sci. Engrg., 3 (1991), pp. 181–191.

[7] A. BERMÚDEZ, R. DURÁN, M. MUSCHIETTI, R. RODRÍGUEZ, AND J. SOLOMIN, *Finite element vibration analysis of fluid-solid systems without spurious modes*, SIAM J. Numer. Anal., 32 (1995), pp. 1280–1295.

[8] A. BESPALOV, *Finite element method for the eigenmode problem of a RF cavity resonator*, Soviet J. Numer. Anal. Math. Model., 3 (1988), pp. 163–178.

[9] J. BEY, *Tetrahedral grid refinement*, Computing, 55 (1995), pp. 355–378.

[10] D. BOFFI, *Discrete compactness and Fortin operator for edge elements*, Tech. Rep. AM187, Department of Mathematics, Pennsylvania State University, State College, USA, April 1999. Published online in Numerische Mathematik, 86, 2000.

[11] ———, *A note on the discrete compactness property and the de Rham complex*, Tech. Rep. AM188, Department of Mathematics, Pennsylvania State University, State College, PA, April 1999. To appear in Appl. Math. Letters.

[12] D. BOFFI, F. BREZZI, AND L. GASTALDI, *On the problem of spurious eigenvalues in the approximation of linear elliptic problems in mixed form*, Math. Comp., 69 (2000), pp. 121–140.

[13] D. BOFFI, P. FERNANDES, L. GASTALDI, AND I. PERUGIA, *Computational models of electromagnetic resonators: Analysis of edge element approximation*, SIAM J. Numer. Anal., 36 (1999), pp. 1264–1290.

[14] A. BOSSAVIT, *Mixed finite elements and the complex of Whitney forms*, in The Mathematics

of Finite Elements and Applications VI, J. Whiteman, ed., Academic Press, London, 1988, pp. 137–144.

[15] ———, *A rationale for edge elements in 3D field computations*, IEEE Trans. Mag., 24 (1988), pp. 74–79.

[16] ———, *Whitney forms: A class of finite elements for three–dimensional computations in electromagnetism*, IEE Proc. A, 135 (1988), pp. 493–500.

[17] ———, *A new viewpoint on mixed elements*, Meccanica, 27 (1992), pp. 3–11.

[18] J. Bramble, J. Pasciak, and A. Knyazev, *A subspace preconditioning algorithm for eigenvector/eigenvalue computation*, Adv. Comput. Math., 6 (1996), pp. 159–189.

[19] F. Brezzi and M. Fortin, *Mixed and hybrid finite element methods*, Springer–Verlag, 1991.

[20] Z. Cai, J. Mandel, and S. McCormick, *Multigrid methods for nearly singular linear equations and eigenvalue problems*, SIAM J. Numer. Anal., 34 (1997), pp. 178–200.

[21] S. Caorsi, P. Fernandes, and M. Raffetto, *Approximations of electromagnetic eigenproblems: A general proof of convergence for edge finite elements of any order of both Nédélec's families*, Tech. Rep. 16/99, CNR-IMA Genoa, Genoa, Italy, 1999.

[22] ———, *On the convergence of Galerkin finite element approximations of electromagnetic eigenproblems*, SIAM J. Numer. Anal., 38 (2000), pp. 580–607.

[23] P. Ciarlet, *The Finite Element Method for Elliptic Problems*, vol. 4 of Studies in Mathematics and its Applications, North-Holland, Amsterdam, 1978.

[24] P. Ciarlet, Jr. and J. Zou, *Fully discrete finite element approaches for time–dependent Maxwell equations*, Numer. Math., 82 (1999), pp. 193–219.

[25] P. Deuflhard, T. Friese, and F. Schmidt, *A nonlinear multigrid eigenproblem solver for the complex Helmholtz equation*, Tech. Rep. SC 97–55, ZIB–Berlin, 1997.

[26] E. D'yakonov, *Iteration methods in eigenvalue problems*, Math. Notes, 34 (1983), pp. 945–953.

[27] ———, *Optimization in solving elliptic problems*, CRC Press, Boca Raton, Florida, 1996.

[28] E. D'yakonov and M. Orekhov, *Minimization of the computational labor in determining the first eigenvalues of differential operators*, Math. Notes, 27 (1980), pp. 382–391.

[29] P. Fernandes and G. Gilardi, *Magnetostatic and electrostatic problems in inhomogeneous anisotropic media with irregular boundary and mixed boundary conditions*, M^3AS, 7 (1997), pp. 957–991.

[30] P. Fernandes and M. Raffetto, *Recent developments in the (spurious-free) approximation of electromagnetic eigenproblems by the finite element method*, Tech. Rep. 06/00, CNR-IMA Genoa, Genoa, Italy, 2000.

[31] V. Girault and P. Raviart, *Finite element methods for Navier–Stokes equations*, Springer–Verlag, Berlin, 1986.

[32] S. Godunov, V. Ogneva, and G. Prokopov, *On the convergence of the modified method of steepest descent in the calculation of eigenvalues*, Amer. Math. Soc. Transl. Ser. 2, 105 (1976), pp. 111–116.

[33] P. Grisvard, *Singularities in boundary value problems*, vol. 22 of Research Notes in Applied Mathematics, Springer–Verlag, New York, 1992.

[34] P. Gross, *Efficient finite element-based algorithms for topological aspects of 3-dimensional magnetoquasistatic problems*, PhD thesis, College of Engineering, Boston University, Boston, USA, 1998.

[35] W. Hackbusch, *On the computation of approximate eigenvalues and eigenfunctions of elliptic operators by means of a multi–grid method*, SIAM J. Numer. Anal., 16 (1979), pp. 201–215.

[36] ———, *Multi–Grid methods and applications*, Springer series in computational mathematics 4, Springer, Berlin, 1985.

[37] ———, *Theorie und Numerik elliptischer Differentialgleichungen*, B.G. Teubner–Verlag, Stuttgart, 1986.

[38] R. Hiptmair, *Multigrid method for $H(\mathrm{div})$ in three dimensions*, ETNA, 6 (1997), pp. 7–77.

[39] ———, *Canonical construction of finite elements*, Math. Comp., 68 (1999), pp. 1325–1346.

[40] ———, *Multigrid method for Maxwell's equations*, SIAM J. Numer. Anal., 36 (1999), pp. 204–225.

[41] ———, *Multigrid for eddy current computation*, Report 154, SFB 382, Universität Tübingen, Tübingen, Germany, March 2000. Submitted to Math. Comp.

[42] F. Jochmann, *A compactness result for vector fields with divergence and curl in $L^q(\Omega)$ involving mixed boundary conditions*, Appl. Anal., 66 (1997), pp. 189–203.

[43] A. Knyazev, *Convergence rate estimates for iterative methods for a mesh symmetric eigenvalue problem*, Russian J. Numer. Anal. Math. Modelling, 2 (1987), pp. 371–396.

[44] A. Knyazev, *Preconditioned eigensolvers -an oxymoron?* , Electron. Trans. Numer. Anal., 7 (1998), pp. 104–123.

[45] ———, *Toward the optimal preconditioned eigensolver: Locally optimal block preconditioned*

conjugate gradient method, Accepted to SIAM J. Sci. Comp., (2000).

[46] J. Mandel and S. McCormick, *A multilevel variational method for $Au = \lambda Bu$ on composite grids*, J. Comput. Phys., 80 (1989), pp. 442–452.

[47] P. Monk, *Analysis of a finite element method for Maxwell's equations*, SIAM J. Numer. Anal., 29 (1992), pp. 714–729.

[48] P. Monk and L. Demkowicz, *Discrete compactness and the approximation of Maxwell's equations in $\mathbb{R}^3$*, Math. Comp., (2000). Published online February 23, 2000.

[49] J. Nédélec, *Mixed finite elements in R^3*, Numer. Math., 35 (1980), pp. 315–341.

[50] K. Neymeyr, *A geometric theory for preconditioned inverse iteration applied to a subspace.* Sonderforschungsbereich 382, Universität Tübingen, Report 131, 1999, Accepted for Math. Comput., 1999.

[51] ———, *Why preconditioning gradient type eigensolvers?* Sonderforschungsbereich 382, Universität Tübingen, Report 146, revised version, 2000.

[52] ———, *A geometric theory for preconditioned inverse iteration. I: Extrema of the Rayleigh quotient*, Linear Algebra Appl., 322 (2001), pp. 61–85.

[53] ———, *A geometric theory for preconditioned inverse iteration. II: Convergence estimates*, Linear Algebra Appl., 322 (2001), pp. 87–104.

[54] B. Parlett, *The symmetric eigenvalue problem*, Prentice Hall, Englewood Cliffs New Jersey, 1980.

[55] W. Petryshyn, *On the eigenvalue problem $Tu - \lambda Su = 0$ with unbounded and non–symmetric operators T and S*, Philos. Trans. Roy. Soc. Math. Phys. Sci., 262 (1968), pp. 413–458.

[56] B. Samokish, *The steepest descent method for an eigenvalue problem with semi–bounded operators*, Izv. Vyssh. Uchebn. Zaved. Mat., 5 (1958), pp. 105–114.

[57] F. Schmidt, T. Friese, L. Zschiederich, and P. Deuflhard, *Adaptive multirgrid methods for the vectorial Maxwell eigenvalue problem for optical waveguide design*, Tech. Rep. 00-54, ZIB Berlin, Berlin, Germany, December 2000.

[58] D. White and J. Koning, *Efficient solution of large-scale electromagnetic eigenvalue problems using the implicitly restarted Arnoldi method*, Preprint UCRL-JC-129188, Lawrence Livermore National Laboratory, 1997.