Journal of Computational Mathematics
Vol.33, No.6, 2015, 557–575.
http://www.global-sci.org/jcm
doi:10.4208/jcm.1502-m4539

INEXACT TWO-GRID METHODS FOR EIGENVALUE PROBLEMS*

Qun Gu and Weiguo Gao[1)]
MOE Key Laboratory of Computational Physical Sciences and School of Mathematical Sciences, Fudan University, Shanghai 200433, China
Email: 081018018@fudan.edu.cn, wggao@fudan.edu.cn

Abstract

We discuss the inexact two-grid methods for solving eigenvalue problems, including both partial differential and integral equations. Instead of solving the linear system exactly in both traditional two-grid and accelerated two-grid method, we point out that it is enough to apply an inexact solver to the fine grid problems, which will cut down the computational cost. Different stopping criteria for both methods are developed for keeping the optimality of the resulting solution. Numerical examples are provided to verify our theoretical analyses.

Mathematics subject classification: 65N25, 65N30, 65B99.
Key words: Inexact, Two-grid, Eigenvalue, Eigenvector, Finite element method, Convergence rate.

1. Introduction

The purpose of this paper is to present inexact two-grid methods for solving eigenvalue problems, including both partial differential and integral equations.

The research on two-grid method was initialized by Xu [1–3] for nonselfadjoint and indefinite problems, and has been extensively developed. Then, it was applied to other problems by many other authors, for instance, Axelsson and Layton [4] for nonlinear elliptic problems, Dawson and Wheeler [5] for nonlinear parabolic equations, Layton and Lenferink [6], Utnes [7], and Layton and Tobiska [8] for Navier-Stokes problems, Marion and Xu [9] for evolution problems. This method is also used as part of the finite difference scheme (see also Dawson, Wheeler and Woodward [10] for parabolic equations). Recently, Chien and Jeng [11] used this method along with the continuation method for solving semilinear elliptic eigenvalue problems; Jin, Shu and Xu [12] employed it for decoupling systems of partial differential equations; Xu and Zhou [13,14] developed localized and parallelized algorithms based on two-grid discretizations for linear and nonlinear elliptic boundary problems as well as eigenvalue problems.

Two-grid method is first applied to eigenvalue problems by Xu and Zhou [15]. The correction step is similar to the early work in 1981 by Lin and Xie [16] in which both nonlinear and eigenvalue problems were discussed. Later, they also developed the localized and parallelized version [17]. Based on this two-grid idea, Dai and Zhou [18] presented a three scale methods for the eigenvalue problems in quantum mechanics. Hu and Cheng [19] raised an acceleration technique for two-grid method. Xie and Lin [20] generalized the two-grid where the sizes of h and H are independent with each other.

* Received June 11, 2014 / Revised version received November 19, 2014 / Accepted February 4, 2015 / Published online November 17, 2015 /
1) Corresponding author

In this paper, we present an inexact two-grid algorithms for eigenvalue problems: Find $\lambda_h \in \mathbb{R}$ and $u_h \in \mathcal{S}_h \setminus \{0\}$ such that

$$a(u_h, v) = \lambda_h b(u_h, v), \qquad \forall v \in \mathcal{S}_h.$$

Here $\mathcal{S}_h$ is a finite element space defined on a quasi-uniform grid of size h. Our inexact two-grid algorithms for eigenvalue problem reads

1. Solve a standard finite element discretization on a coarse space $\mathcal{S}_H$: Find $\lambda_H \in \mathbb{R}$ and $u_H \in \mathcal{S}_H \setminus \{0\}$ such that

$$a(u_H, v) = \lambda_H b(u_H, v), \qquad \forall v \in \mathcal{S}_H. \tag{1.1}$$

   obtaining an initial guess for the eigenpair on fine grid space $\mathcal{S}_h$.

2. Choose and solve a linear system inexactly on fine grid space depending on the mesh parameters.

   - If $h \geq H^2$, we adopt the *inexact two-grid scheme* (ITG). Find $u^h \in \mathcal{S}_h$ such that

$$a(u^h, v) = b(u_H + \xi d, v) \tag{1.2}$$

   - If $h < H^2$, we employ the *inexact accelerated two-grid scheme* (IATG). Find $u^h \in \mathcal{S}_h$ such that

$$a(u^h, v) - \lambda_H b(u^h, v) = b(u_H + \xi d, v), \tag{1.3}$$

   Here d is the b-normalized function that is proportional to the residual, and ξ is the b-norm of the residual. Note that in the second strategy, a shifted linear system is solved, which is also discussed in [21].

3. Recover the eigenvalue by

$$\lambda^h = \frac{a(u^h, u^h)}{b(u^h, u^h)}. \tag{1.4}$$

Our main contribution is in the second step, *i.e.*, the fine grid correction. Specifically, instead of solving the linear system exactly, which might be expensive to compute or even not available, we resort to inexact solvers to get a decent solution for the current discretization level. The accuracy of the solution and the computational cost of the total algorithm depends on the stopping criteria of the inexact solver. In comparison with the result in Xu and Zhou [15] where the optimal grid parameters are set as $h = H^2$, by the analysis of Section 3, we obtain that

- ITG with the stopping criteria $\tau = \mathcal{O}(H^2)$ on relative residual norm,
- IATG with the stopping criteria $\tau = \mathcal{O}(1) < 1$ on relative residual norm,

we both have

$$\|u^h - u\|_a \leq \mathcal{O}(h^r + H^{r+1}) \quad \text{and} \quad \left|\lambda^h - \lambda\right| \leq \mathcal{O}(h^{2r} + H^{2r+2}). \tag{1.5}$$

Both methods reduce the computational cost comparing to Xu and Zhou's method while ITG is slightly cheaper than IATG. In comparison with the result of Hu and Cheng [19] in which

$h = H^4$, although ITG doesn't work in this case, IATG with the stopping criteria $\tau = \mathcal{O}(H^2)$ still preserves the optimal accuracy

$$\|u^h - u\|_a \le \mathcal{O}(h^r + H^{3r+1}) \quad \text{and} \quad \left|\lambda^h - \lambda\right| \le \mathcal{O}(h^{2r} + H^{6r+2}), \tag{1.6}$$

yet with less computational cost.

The rest of the paper is organized as follows. In Section 2, we describe the standard finite element method for eigenvalue problems. Section 3 contains the two new algorithms of the paper, as well as error analysis and extensions. In Section 4, we give some numerical examples to show the efficiency of our new methods.

2. Preliminaries

We start from the basic notation and properties of the finite element analysis of the self-adjoint eigenvalue problems. Through out this paper, we use the letter C to denote a generic positive constant which may stand for different values at its different occurrences. Let $\mathcal{H}$ be a real Hilbert space equipped with inner product $(\cdot,\cdot)$ and norm $\|\cdot\|$. Let $a(\cdot,\cdot)$ and $b(\cdot,\cdot)$ be two symmetric bilinear forms on $\mathcal{H}\times\mathcal{H}$ with the assumptions that

$$|a(u,v)| \le C\|u\|\|v\|, \qquad \forall u,v \in \mathcal{H}, \tag{2.1}$$

and

$$C\|u\|^2 \le a(u,u), \qquad \forall u \in H. \tag{2.2}$$

Thus, we can introduce the so-called a-norm $\|u\|_a := \sqrt{a(u,u)}$ on Hilbert space $\mathcal{H}$. We say that u is an a-unit vector if $\|u\|_a = 1$. Obviously, norm $\|u\|_a$ and norm $\|u\|$ are equivalent. Denote the Hilbert space equipped with the inner product $a(\cdot,\cdot)$ and the norm $\|\cdot\|_a$ as $\mathcal{H}_a$. Furthermore, we assume that

$$b(u,u) > 0, \qquad \forall u \in \mathcal{H}, u \ne 0. \tag{2.3}$$

and assume that norm $\|\cdot\|$ is relatively compact with respect to the norm

$$\|u\|_b := \sqrt{b(u,u)}. \tag{2.4}$$

Let $\mathcal{H}_b$ be the completion of $\mathcal{H}_a$ with respect to $\|\cdot\|_b$. Thus, the space $\mathcal{H}_b$ is a Hilbert space equipped with inner produce $b(\cdot,\cdot)$ and norm $\|\cdot\|_b$ and is compact with respect to $\|\cdot\|_a$, and $\mathcal{H}_a \hookrightarrow\hookrightarrow \mathcal{H}_b$. Construct the negative space $\mathcal{H}_{-a} :=$ the dual of $\mathcal{H}_a$ with norm

$$\|u\|_{-a} = \sup_{0\ne v\in\mathcal{H}_a} \frac{b(u,v)}{\|v\|_a}. \tag{2.5}$$

Then $\mathcal{H}_b \subset \mathcal{H}_{-a}$ compactly, and for $v \in \mathcal{H}_a$, $b(u,v)$ has a continuous extension to $u \in \mathcal{H}_{-a}$ so that $b(u,v)$ is continuous on $\mathcal{H}_{-a}\times\mathcal{H}_a$.

We assume that $\mathcal{S}_h \subset \mathcal{H}_a$ is a family of finite dimensional spaces that satisfy the following assumption: For any $u \in \mathcal{H}_a$, we have

$$\inf_{v\in\mathcal{S}_h} \|u - v\|_a \to 0, \qquad \text{as } h \to 0. \tag{2.6}$$

Introduce the orthogonal projection operator $P_h : \mathcal{H}_a \to \mathcal{S}_h$ defined by

$$a(u - P_h u, v) = 0 \qquad \forall u \in \mathcal{H}_a, v \in \mathcal{S}_h. \tag{2.7}$$

Obviously

$$\|P_h u\|_a \le \|u\|_a, \qquad \forall u \in \mathcal{H}_a. \tag{2.8}$$

Let $\eta_a(h)$ be defined by

$$\eta_a(h) = \sup_{0\neq f\in\mathcal{H}_a} \inf_{v\in\mathcal{S}_h} \frac{\|Tfv\|_a}{\|f\|_a} \tag{2.9}$$

where $T : \mathcal{H}_a \to \mathcal{H}_{-a}$ satisfies

$$a(Tf, v) = b(f, v), \qquad \forall f \in \mathcal{H}_{-a}, v \in \mathcal{H}_a. \tag{2.10}$$

We have the following results (see Lemma 3.3 and Lemma 3.4 in [22])

Lemma 2.1. *Let $\eta_a(h)$ be defined in* (2.9), *we have*

$$\eta_a(h) \to 0 \qquad \text{as } h \to 0,$$

and

$$\|u - P_h u\|_{-a} \le C\eta_a(h)\|u - P_h u\|_a, \qquad \forall u \in \mathcal{H}_a. \tag{2.11}$$

Now we consider the standard variational self-adjoint eigenvalue problem. A number λ is called an eigenvalue of the form $a(\cdot,\cdot)$ and $b(\cdot,\cdot)$ if there is a nonzero vector $u \in \mathcal{H}_a$, called an associated eigenvector, such that

$$a(u, v) = \lambda b(u, v) \qquad \forall v \in \mathcal{H}_a. \tag{2.12}$$

It is well known that (2.12) has a countable sequence of real eigenvalues

$$0 < \lambda_1 \le \lambda_2 \le \lambda_3 \le \cdots,$$

and corresponding eigenvectors

$$u_1 \le u_2 \le u_3 \le \cdots,$$

which can be assumed to satisfy

$$a(u_i, u_j) = \lambda_i b(u_i, u_j) = \delta_{ij}.$$

A standard finite element formulation for (2.12) is: Solve $(\lambda_h, u_h) \in \mathbb{C} \times (\mathcal{S}_h \setminus \{0\})$, such that

$$a(u_h, v) = \lambda_h b(u_h, v), \qquad \forall v \in \mathcal{S}_h. \tag{2.13}$$

One has that (2.13) has a finite sequence of eigenvalues

$$0 < \lambda_{1,h} \le \lambda_{2,h} \le \lambda_{3,h} \le \cdots, \lambda_{n_h,h}, \qquad n_h = \dim(\mathcal{S}_h),$$

and corresponding eigenvectors

$$u_{1,h} \le u_{2,h} \le u_{3,h} \le \cdots, \le u_{n_h,h},$$

which can be assumed to satisfy

$$a(u_{i,h}, u_{j,h}) = \lambda_{i,h} b(u_{i,h}, u_{j,h}) = \delta_{ij}.$$

By minimum-maximum principle (see [23, 24]), we have

$$\lambda_i \le \lambda_{i,h} \qquad i = 1, \ldots, n_h.$$

Set $M(\lambda_i) = \{u \in \mathcal{H}_a : u \text{ is an eigenvector of (2.12) corresponding to } \lambda_i\}$ and let $M_h(\lambda_i)$ be the direct sum of eigenspaces corresponding to all eigenvalues that converge to λ_i. Let $\hat{M}(\lambda_i) = \{u \in M(\lambda_i) : \|u\|_a = 1\}$ and similarly, $\hat{M}_h(\lambda_i) = \{u \in M_h(\lambda_i) : \|u\|_a = 1\}$.

Let

$$\delta_h(\lambda_i) = \sup_{u \in \hat{M}(\lambda_i)} \inf_{v \in \mathcal{S}_h} \|u - v\|_a.$$

The following results (see p.699 of [23] and Lemma 3.7 and (3.29b) of [22], or cf. [24]) will be employed.

Proposition 2.1. *(i) For any $u_{i,h}$ of (2.13), $i = 1, 2, \ldots, n_h$, there is an a-unit eigenvector u_i of (2.12) corresponding to λ_i such that*

$$\|u_i - u_{i,h}\|_a \le C_i \delta_h(\lambda_i). \tag{2.14}$$

Moreover,

$$\|u_i - u_{i,h}\|_{-a} \le C_i \eta_a(h) \|u_i - u_{i,h}\|_a. \tag{2.15}$$

(ii) For eigenvalues,

$$\lambda_i \le \lambda_{i,h} \le \lambda_i + C_i \delta_h^2(\lambda_i), \tag{2.16}$$

where C_i is some constant that depending on i but not on the mesh parameter h.

We close this section by introducing a simple but crucial property of eigenvalue and eigenvector approximation (see *e.g.* Lemma 3.1 of [22] or Lemma 9.1 of [23]).

Proposition 2.2. *Let (λ, u) be an eigenpair of (2.12). For any b-unit vector $w \in \mathcal{H}_a \setminus \{0\}$, we have*

$$a(w, w) - \lambda = a(w - u, w - u) - \lambda b(w - u, w - u). \tag{2.17}$$

In our inexact two-grid algorithms, we need to determine the stopping criteria for the inexact solver. For easier description, we denote the stillness matrix as A_h and the mass matrix B_h. A function $f \in \mathcal{S}_h$ defined by

$$f = \sum_{j=1}^{n_h} v_j \phi_j = \mathbf{f}^\top \mathbf{\Phi}$$

is uniquely determined by the vector $\mathbf{f}$. Here $\mathbf{\Phi} = \{\phi_j\}$ is the basis of $\mathcal{S}_h$. To rule out the confusion, we denote the corresponding linear combination coefficient vector of $u_H \in \mathcal{S}_h$ as $\widetilde{\mathbf{u}}_H$, instead of $\mathbf{u}_H$.

3. Inexact Two-Grid Algorithms and Analysis

In this section, we discuss two type of inexact two-grid algorithm that expand the idea of Xu and Zhou [15] (Section 3.1) and Hu and Cheng [19] (Section 3.2), respectively. Examples of second order elliptic operator and Fredholm integral operator are discussed in Section 3.3.

3.1. Inexact Two-Grid Algorithm

In this subsection, we consider the generalization of the original two-grid method, shown in Algorithm 3.1.

Algorithm 3.1 Inexact Two-Grid Algorithm for Eigenvalue Problem (ITG(τ))

1: Solve the coarse grid eigenvalue problem

$$a(u_{i,H}, v) = \lambda_{i,H} b(u_{i,H}, v), \qquad \forall v \in \mathcal{S}_H. \tag{3.1}$$

2: Solve the shifted linear system

$$a(u^{i,h}, v) = \lambda_{i,H} b(u_{i,H}, v) + \xi a(d, v), \qquad \forall v \in \mathcal{S}_h, \tag{3.2}$$

on fine grid inexactly with relative residual

$$\tau = \left(\lambda_{i,H} \|\mathsf{B}_h\|_2 \cdot \|\mathsf{A}_h^{-1/2}\|_2^2\right)^{-1} \cdot \delta_H^2(\lambda_i). \tag{3.3}$$

3: Compute the corresponding eigenvalue

$$\lambda^{i,h} = \frac{a(u^{i,h}, u^{i,h})}{b(u^{i,h}, u^{i,h})}. \tag{3.4}$$

Note that the inexact correction step is (3.2). Here ξ is controlled by the stopping criteria τ. In practice, inexact solver for this step in matrix form reads:

$$\mathsf{A}_h \mathbf{u}^{i,h} = \lambda_{i,H} \mathsf{B}_h \widetilde{\mathbf{u}}_{i,H} + \xi \mathsf{A}_h \mathbf{d}. \tag{3.5}$$

For zero initial guess of iterative solver, the initial residual reads

$$\mathbf{r}_0 = \lambda_{i,H} \mathsf{B}_h \widetilde{\mathbf{u}}_{i,H} \tag{3.6}$$

And the final residual takes the form

$$\mathbf{r} = \xi \mathsf{A}_h \mathbf{d}. \tag{3.7}$$

We assume that

$$\|d\|_a = 1, \tag{3.8}$$

or in matrix form,

$$\|\mathbf{d}\|_{\mathsf{A}_h} = 1. \tag{3.9}$$

The stopping criteria of an inexact solver requires

$$\frac{\|\mathbf{r}\|_2}{\|\mathbf{r}_0\|_2} \leq \tau, \tag{3.10}$$

which implies that

$$\begin{aligned}
\xi &= \xi \|\mathbf{d}\|_{\mathsf{A}_h} \\
&\leq \xi \cdot \|\mathsf{A}_h \mathbf{d}\|_2 \cdot \|\mathsf{A}_h^{-1/2}\|_2 \\
&\leq \tau \cdot \|\lambda_{i,H} \mathsf{B}_h \widetilde{\mathbf{u}}_{i,H}\|_2 \cdot \|\mathsf{A}_h^{-1/2}\|_2 \\
&\leq \tau \cdot \lambda_{i,H} \|\mathsf{B}_h\|_2 \cdot \|\mathsf{A}_h^{-1/2}\|_2^2,
\end{aligned} \tag{3.11}$$

assuming that $\|u_{i,H}\|_a = \|\widetilde{\mathbf{u}}_{i,H}\|_{\mathsf{A}_h} = 1$.

Theorem 3.1. *Assume that* $(\lambda^{i,h}, u^{i,h})$ *are obtained by Algorithm* 3.1. *If* $\mathcal{S}_H \subset \mathcal{S}_h$, *then*

$$\|u_i - u^{i,h}\|_a \le C_i(\lambda_{i,H} - \lambda_i + \lambda_i\|u_i - u_{i,H}\|_{-a} + \|u_i - P_h u_i\|_a + \xi) \tag{3.12}$$

and

$$0 \le \lambda^{i,h} - \lambda_i \le C_i\Big((\lambda_{i,H} - \lambda_i)^2 + \|u_i - u_{i,H}\|_{-a}^2 + \|u_i - P_h u_i\|_a^2 + \xi^2\Big). \tag{3.13}$$

If, in addition, the relative residual threshold τ *is set as* $\Big(\lambda_{i,H}\|\mathsf{B}_h\|_2 \cdot \|\mathsf{A}_h^{-1/2}\|_2^2\Big)^{-1} \cdot \delta_H^2(\lambda_i)$, *we have,*

$$\|u_i - u^{i,h}\|_a \le C_i\Big(\delta_H^2(\lambda_i) + \eta_a(H)\delta_H(\lambda_i) + \delta_h(\lambda_i)\Big) \tag{3.14}$$

and

$$0 \le \lambda^{i,h} - \lambda_i \le C_i\Big(\delta_H^4(\lambda_i) + \eta_a^2(H)\delta_H^2(\lambda_i) + \delta_h^2(\lambda_i)\Big). \tag{3.15}$$

Proof. Notice the identity that

$$a(P_h u_i - u^{i,h}, v) = (\lambda_i - \lambda_{i,H})b(u_{i,H}, v) + \lambda_i b(u_i - u_{i,H}, v) - \xi a(d, v), \qquad \forall v \in \mathcal{S}_h, \tag{3.16}$$

together with (2.7), (2.5) and (3.8), we immediately obtain (3.12). With the help of Proposition 2.2 and (3.12), we get (3.13). Finally, (3.14) and (3.15) are obtained from Proposition 2.1 and (3.11). □

3.2. Inexact Accelerated Two-grid Algorithm

We apply the inexact solver idea to the accelerated two-grid method proposed by Hu and Cheng [19] and Yang and Bi [25] and get the following algorithm:

Algorithm 3.2 Inexact Accelerated Two-Grid Algorithm for Eigenvalue Problem (IATG(τ))

1: Solve the coarse grid eigenvalue problem

$$a(u_{i,H}, v) = \lambda_{i,H} b(u_{i,H}, v), \qquad \forall v \in \mathcal{S}_H. \tag{3.17}$$

2: Solve the shifted linear system

$$a_{\lambda_{i,H}}(\widetilde{u}^{i,h}, v) = b(u_{i,H}, v) + \xi a(d, v), \qquad \forall v \in \mathcal{S}_h, \tag{3.18}$$

on fine grid inexactly with stopping criteria set as $\frac{1}{2} \cdot \Big(\lambda_{i,H}\|\mathsf{B}_h\|_2 \cdot \|\mathsf{A}_h^{-1/2}\|_2^2\Big)^{-1}$. The eigenvector is normalized into $u^{i,h} = \widetilde{u}^{i,h}/\|\widetilde{u}^{i,h}\|_a$ afterwards.

3: Compute the corresponding eigenvalue

$$\lambda^{i,h} = \frac{a(u^{i,h}, u^{i,h})}{b(u^{i,h}, u^{i,h})}. \tag{3.19}$$

Notice that the exactly solving (3.18) is time consuming, since the right hand side operator is nearly singular [21]. Hence, the inexact solver idea is applied here. From now on, we consider the multiplicity of λ_i by assuming its geometric multiplicity as q. The following theorem gives the error estimations for our inexact scheme based on the choice stopping criteria.

Theorem 3.2. *Assume that $(\lambda^{i,h}, u^{i,h})$ are obtained by Algorithm 3.2 and let H be properly small. Moreover, assuming the stopping criteria chosen as*

$$\tau \le \frac{1}{2} \cdot \left(\lambda_{i,H} \|\mathsf{B}_h\|_2 \cdot \|\mathsf{A}_h^{-1/2}\|_2^2 \right)^{-1}.$$

Then there exists $u_i \in M(\lambda_i)$ such that

$$\left\|u^{i,h} - u_i\right\|_a \lesssim \delta_H^2(\lambda_i)\Big(\delta_H^2(\lambda_i) + \eta_a(H)\delta_H(\lambda_i) + \xi\Big) + \delta_h(\lambda_i), \tag{3.20}$$

$$\lambda^{i,h} - \lambda_i \lesssim \delta_H^4(\lambda_i)\Big(\delta_H^4(\lambda_i) + \eta_a^2(H)\delta_H^2(\lambda_i) + \xi^2\Big) + \delta_h^2(\lambda_i). \tag{3.21}$$

Proof. The proof of this theorem is based on the idea of [25]. Let $u_i \in M(\lambda_i)$ be defined as

$$u_i = \sum_{j=i}^{i+q-1} a(u^{i,h}, u_{j,h}) u_j^0 \tag{3.22}$$

where $u_j^0 \in M(\lambda_i)$ satisfying $\|u_0^j - u_{j,h}\|_a \le C_\delta \delta_h(\lambda_j)$. Existence of such u_j^0 is ensured by Proposition 2.1. In the following, we prove (3.20). First, we split the left hand side into the following two pieces by using triangular inequality

$$\begin{aligned} \|u^{i,h} - u_i\|_a &\le \|u^* - u_i\|_a + \|u^{i,h} - u^*\|_a \\ &= \|u^* - u_i\|_a + \mathrm{dist}(u^{i,h}, M_h(\lambda_i)) \end{aligned} \tag{3.23}$$

in which we define

$$u^* = \sum_{j=i}^{i+q-1} a(u^{i,h}, u_{j,h}) u_{j,h}, \tag{3.24}$$

where $\{u_{j,h}\}_{j=i}^{i+q-1}$ is the orthonormal basis for $M_h(\lambda_i)$. It is easy to see that

$$\|u^* - u_i\|_a = \left\| \sum_{j=i}^{i+q-1} a(u^{i,h}, u_{j,h})(u_{j,h} - u_j^0) \right\|_a \lesssim \delta_h(\lambda_i). \tag{3.25}$$

Next we estimate the second term. Notice that Step 2 in Algorithm 3.2 is equivalent to

$$a(\widetilde{u}^{i,h}, v) - \lambda_{i,H} b(\widetilde{u}^{i,h}, v) = a(\bar{u}^{i,h} + \xi d, v), \qquad \forall v \in \mathcal{S}_h, \tag{3.26}$$

where $\bar{u}^{i,h}$ is the eigenvector computed by original two-grid algorithm. Moreover, one may assume that $\|\bar{u}^{i,h}\|_a = 1$. Plug the expansions

$$\widetilde{u}^{i,h} = \sum_{j=1}^{N_h} \widetilde{\alpha}_j u_{j,h}, \qquad \bar{u}^{i,h} = \sum_{j=1}^{N_h} \bar{\alpha}_j u_{j,h}, \qquad \xi d = \sum_{j=1}^{N_h} \beta_j u_{j,h}. \tag{3.27}$$

into (3.27), we have

$$a\left(\sum_{j=1}^{N_h} u_{j,h}\widetilde{\alpha}_j\left(1-\lambda_{i,H}/\lambda_{j,h}\right), v\right) = a\left(\sum_{j=1}^{N_h} u_{j,h}(\bar{\alpha}_j+\beta_j), v\right), \qquad \forall v \in \mathcal{S}_h. \tag{3.28}$$

Hence

$$\widetilde{\alpha}_j = \frac{\lambda_{j,h}(\bar{\alpha}_j+\beta_j)}{\lambda_{j,h}-\lambda_{i,H}} = \begin{cases} \frac{\lambda_{i,h}(\bar{\alpha}_j+\beta_j)}{\lambda_{i,h}-\lambda_{i,H}} & j = i_1,\ldots,i_q; \\ \frac{\lambda_{j,h}(\bar{\alpha}_j+\beta_j)}{\lambda_{j,h}-\lambda_{i,H}} & j \neq i_1,\ldots,i_q. \end{cases} \tag{3.29}$$

Therefore,

$$\begin{aligned}
&\tan\angle(\widetilde{u}^{i,h}, M_h(\lambda_i)) \\
&= \left(\sum_{j\neq i_1,\ldots,i_q}\left(\frac{\lambda_{j,h}(\bar{\alpha}_j+\beta_j)}{\lambda_{j,h}-\lambda_{i,H}}\right)^2\right)^{1/2}\cdot\left(\sum_{j=i_1,\ldots,i_q}\left(\frac{\lambda_{i,h}(\bar{\alpha}_j+\beta_j)}{\lambda_{i,h}-\lambda_{i,H}}\right)^2\right)^{-1/2} \\
&= \left(\sum_{j\neq i_1,\ldots,i_q}\left(\frac{\lambda_{i,h}-\lambda_{i,H}}{\lambda_{j,h}-\lambda_{i,H}}\cdot\frac{\lambda_{j,h}}{\lambda_{i,h}}\cdot(\bar{\alpha}_j+\beta_j)\right)^2\right)^{1/2}\cdot\left(\sum_{j=i_1,\ldots,i_q}(\bar{\alpha}_j+\beta_j)^2\right)^{-1/2} \\
&\leq \frac{|\lambda_{i,h}-\lambda_{i,H}|}{\rho}\left(\frac{\sum_{j\neq i_1,\ldots,i_q}(\bar{\alpha}_j+\beta_j)^2}{\sum_{j=i_1,\ldots,i_q}(\bar{\alpha}_j+\beta_j)^2}\right)^{1/2}.
\end{aligned} \tag{3.30}$$

where $\rho = \sup_{j\neq i}\frac{\lambda_{i,h}}{\lambda_{j,h}}(\lambda_{j,h}-\lambda_{i,H})$. From estimation (3.14), we know that

$$\begin{aligned}
&\sqrt{\sum_{j\neq i_1,\ldots,i_q}\bar{\alpha}_j^2} \leq C_i\Big(\delta_H^2(\lambda_i)+\eta_a(H)\delta_H(\lambda_i)+\delta_h(\lambda_i)\Big), \\
&\sqrt{\sum_{j=i_1,\ldots,i_q}\bar{\alpha}_j^2} \geq 1 - C_i\Big(\delta_H^2(\lambda_i)+\eta_a(H)\delta_H(\lambda_i)+\delta_h(\lambda_i)\Big).
\end{aligned} \tag{3.31}$$

Hence, when $\|\beta\|_2 \leq \frac{1}{2}$, we have

$$\tan\angle\Big(\widetilde{u}^{i,h}, M_h(\lambda_i)\Big) \lesssim |\lambda_{i,h}-\lambda_{i,H}|\Big(\delta_H^2(\lambda_i)+\eta_a(H)\delta_H(\lambda_i)+\delta_h(\lambda_i)+\|\beta\|_2\Big). \tag{3.32}$$

Combining (3.15), (3.25) and the fact that $\|\beta\|_2 = \xi$ leads to estimation (3.20). Estimation (3.21) follows directly from (3.20) and Proposition 2.2. □

Remark 3.1. In theory, this inexact step is one inexact Rayleigh quotient iteration (RQI) step. Similar analysis could be found in Jia [26], which is purely from the matrix analysis point of view. In order to get the same estimation, the uniform positiveness condition should be assumed. However, there is not such restriction in finite element setting. Moreover, recently, [27] provided a similiar analysis that is based on multigrid setting and uses multi-shifted inverse iteration. Furthermore, the above results do not specify the choice of inexact solver. One may resort to fast algorithms, such as, the multigrid algorithm [21] to provide an approximate solution.

Remark 3.2. Our inexact two-grid scheme can also be extended to nonselfadjoint eigenvalue problems. Assume that $a(\cdot,\cdot)$ may not be symmetric, but $b(\cdot,\cdot)$ is symmetric. Note that for $\|w\|_b = 1$, we have

$$a(w,w)-\lambda = a(w-u,w-u)-\lambda b(w-u,w-u)+a(w-u,u)-a(u,w-u). \tag{3.33}$$

For instance, when $r>1$ and bilinear form $a(\cdot,\cdot)$ corresponds to a general elliptic operator of second order, then

$$|a(w-u,u)-a(u,w-u)| \le C\|w-u\|_{1-r}$$

and

$$\|w-u\|_{1-r} \ll \|w-u\|_1$$

Therefore, in this situation, our two-grid scheme can also be applied to nonselfadjoint case.

3.3. Examples

Our inexact two-grid discretization scheme can be applied to a large class of eigenvalue problems. In this subsection, we give two examples. One example is a partial differential operator and another for an Fredholm integral operator. Define $\Omega = \mathbb{R}^d$ (d = 1, 2, ...) be a bounded polygonal and convex domain and $\mathcal{T}^h(\Omega)$, consisting of shape- regular simplices, be a mesh of Ω of size h.

3.3.1. Second Order Elliptic Operators

Let $a(\cdot,\cdot)$ be defined as

$$a(u,v) = \int_\Omega \sum_{i,j=1}^d a_{ij}\frac{\partial u}{\partial x_i}\frac{\partial v}{\partial x_j}, \tag{3.34}$$

where $a_{i,j} \in W^{1,\infty}(\Omega)$ and (a_{ij}) is uniformly positive definite on Ω. Let $b(\cdot,\cdot)$ be the standard L^2 inner product. Set $\mathcal{H}_a = H_0^1(\Omega)$ and $\mathcal{H}_b = L^2(\Omega)$. Define the finite element space

$$\mathcal{S}_h(\Omega) = \{v \in C(\overline{\Omega}) \cup H_0^1(\Omega) : v|_\tau \in P_\tau^r, \forall\, \tau \in \mathcal{T}^h(\Omega)\},$$

where P_τ^r is the space of polynomials whose degree is not greater than $r \in \mathbb{Z}^+$.

We assume that $M(\lambda_i) \subset H^{r+1}(\Omega)$, then we have

$$\eta_a(h) = \mathcal{O}(h), \qquad \delta_h(\lambda_i) \le C_i h^r. \tag{3.35}$$

Therefore, for ITG, we have

$$\|u_i - u^{i,h}\|_a \le C_i(H^{r+1}+h^r+\xi) \tag{3.36}$$

and

$$|\lambda_i - \lambda^{i,h}| \le C_i(H^{2r+2}+h^{2r}+\xi^2). \tag{3.37}$$

And for IATG, we have

$$\|u_i - u^{i,h}\|_a \le C_i(H^{3r+1}+h^r+\xi H^{2r}) \tag{3.38}$$

and

$$|\lambda_i - \lambda^{i,h}| \le C_i(H^{6r+2}+h^{2r}+\xi^2 H^{4r}). \tag{3.39}$$

3.3.2. Fredholm Integral Operators

Set

$$a(u,v) = (u,v)_{L^2}, \qquad b(u,v) = (ku,v)_{L^2},$$

where $(ku)(x) = \int_\Omega k(x,y)u(x)\,\mathrm{d}y$ is the symmetric and positive definite Fredholm integral operator on $L^2(\Omega)$ ($k(x,y) \in C^{r+1}(\Omega\times\Omega)$). Define $\mathcal{H}_a = L^2(\Omega)$ and $\mathcal{H}_b =$ the completion of $\mathcal{H}_a$ with respect to $\|\cdot\|_b$, and $\mathcal{S}_h(\Omega) = \{v \in C(\overline{\Omega}) \cup H_0^1(\Omega) : v|_\tau \in P_\tau^r, \forall \tau \in \mathcal{T}_h(\Omega)\}$.

In this case, we have

$$\eta_a(h) = \mathcal{O}(h^{r+1}), \qquad \delta_h(\lambda_i) \le C_i h^{r+1}. \tag{3.40}$$

Therefore, for ITG, we have

$$\|u_i - u^{i,h}\|_a \le C_i(H^{2r+2} + h^{r+1} + \xi) \tag{3.41}$$

and

$$|\lambda_i - \lambda^{i,h}| \le C_i(H^{4r+4} + h^{2r+2} + \xi^2); \tag{3.42}$$

and for IATG, we have

$$\|u_i - u^{i,h}\|_a \le C_i(H^{4r+4} + h^{r+1} + \xi H^{2r+2}) \tag{3.43}$$

and

$$|\lambda_i - \lambda^{i,h}| \le C_i(H^{8r+8} + h^{2r+2} + \xi^2 H^{4r+4}). \tag{3.44}$$

4. Numerical Experiments

We consider three model problems that had been utilized in [15,19] to verify our approaches. All the numerical experiments are generated by AFEM@matlab package [28]. Throughout the numerical test, the inexact solver is chosen as MINRES. To set up a fair environment in numerical test, the exact solver is chosen as MINRES with a tolerance set as 10^{-8} with maximum iteration number set as 1000. We also applied preconditioned version of MINRES (PMINRES) for the inexact step. The preconditioner is chosen as the incomplete Cholesky factorization of the stiffness matrix A_h.

Example 4.1. The first problem reads

$$\begin{cases} -\Delta u = \lambda u, & u \in \Omega, \\ u = 0, & u \in \partial\Omega, \end{cases} \tag{4.1}$$

where domain $\Omega = (0,1)\times(0,1)$ is a unit square.

The analytical form of the eigenvalues for this problem are

$$\lambda_{k,l} = (k^2 + l^2)\pi^2, \qquad k,l = 1,2,\ldots, \tag{4.2}$$

and their corresponding eigenvectors read

$$u_{k,l} = \sin(k\pi x)\sin(l\pi y), \qquad k,l = 1,2,\ldots. \tag{4.3}$$

We test the discretizations with both structured and unstructured 2D meshes. The unstructured meshes are shown in Fig. 4.1.

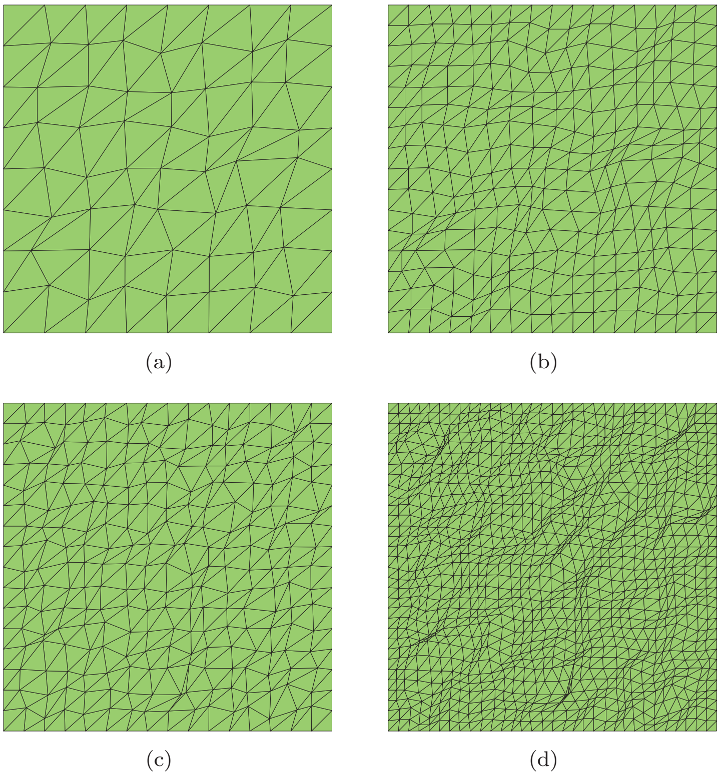

Fig. 4.1. Unstructured 2D meshes.

We use the piecewise linear finite element space in the following numerical experiments.

We first show that the convergence rate of our two inexact two-grid schemes. Setting the stopping criteria on the relative residual as $\tau = H^2$ for ITG and $\tau = 0.5$ for IATG, we observe the scaling behavior of error of eigenvalues, scaling of the error in eigenvectors with a-norm, and the CPU time consumed. Test data are arranged in Table 4.1.

For accuracy consideration, as we can see in Table 4.1, under the restriction that the coarse fine grids satisfy the relation $h = H^2$, the scaling behavior of Xu and Zhou's method (XZ), our ITG(H^2) and IATG(.5) schemes have achieved the optimal accuracy, *i.e.*, $\mathcal{O}(h)$ for the error of approximated eigenvector in a-norm and $\mathcal{O}(h^2)$ for approximated eigenvalue. For the accelerated two-grid method proposed by Hu and Cheng (HC), the scaling behavior of accuracy is of $\mathcal{O}(h^2)$ for eigenvector in a-norm, which is over-qualified for the current discretization with parameter h. On the CPU time aspect, we see that HC scheme consumes more time than XZ's scheme and both two inexact schemes requires less computational costs than XZ and HC scheme, while between ITG and IATG the CPU time consumptions are almost the same. According to Table 4.1, we can see that the iterations of MINRES for the original XZ's scheme is 462 steps for $H = \frac{1}{16}$, compared to the ITG scheme, requiring 130 steps to achieve $\mathcal{O}(H^2) \approx \mathcal{O}(10^{-2})$. Moreover, we observed that the preconditioned MINRES (PMINRES) will reduce the iteration number in general. However, due to the increasing cost of each step, the overall computational cost remains almost the same as the non-preconditioned one.

Secondly, in order to show that our inexact scheme can improve the performance on a large class of coarse and fine grids, we choose mesh sizes satisfying $h = H/2$, a common occurrence

Table 4.1: Results for Example 4.1 on a structured grid with $h = H^2$.

H	$h=H^2$	XZ	Ratio	HC	Ratio	ITG(H^2)	Ratio	IATG(.5)	Ratio
		$\|\lambda^h - \lambda\|$ by MINRES							
2^{-2}	2^{-4}	2.0313E-01	-	1.9105E-01	-	1.9104E-01	-	2.1452E-01	-
2^{-3}	2^{-6}	1.2795E-02	$2^{3.99}$	1.1894E-02	$2^{4.01}$	1.1894E-02	$2^{4.01}$	1.3931E-02	$2^{3.94}$
2^{-4}	2^{-8}	8.0313E-04	$2^{3.99}$	7.4319E-04	$2^{4.00}$	7.4319E-04	$2^{4.00}$	8.9412E-04	$2^{3.96}$
		$\\|u^h - u_h\\|_a$ by MINRES							
2^{-2}	2^{-4}	2.8099E-02	-	5.4870E-03	-	4.8006E-02	-	3.9502E-02	-
2^{-3}	2^{-6}	7.7001E-03	$2^{1.87}$	3.8883E-04	$2^{3.82}$	1.3064E-02	$2^{1.88}$	1.1669E-02	$2^{1.76}$
2^{-4}	2^{-8}	1.9918E-03	$2^{1.95}$	2.5223E-05	$2^{3.95}$	3.4121E-03	$2^{1.94}$	3.1794E-03	$2^{1.88}$
		MINRES CPU Time (sec)							
2^{-2}	2^{-4}	0.0043	-	0.0043	-	0.0020	-	0.0018	-
2^{-3}	2^{-6}	0.0470	-	0.0581	-	0.0133	-	0.0131	-
2^{-4}	2^{-8}	1.5387	-	2.1297	-	0.4408	-	0.4585	-
		MINRES Steps							
2^{-2}	2^{-4}	33	-	37	-	8	-	8	-
2^{-3}	2^{-6}	128	-	152	-	33	-	32	-
2^{-4}	2^{-8}	462	-	599[1]	-	130	-	124	-
H	$h=H^2$	XZ	Ratio	HC	Ratio	ITG(H^2)	Ratio	IATG(.5)	Ratio
		$\|\lambda^h - \lambda\|$ by PMINRES							
2^{-2}	2^{-4}	2.0313E-01	-	1.9105E-01	-	1.9115E-01	-	1.9708E-01	-
2^{-3}	2^{-6}	1.2795E-02	$2^{3.99}$	1.1894E-02	$2^{4.01}$	1.1895E-02	$2^{4.01}$	1.2578E-02	$2^{3.97}$
2^{-4}	2^{-8}	8.0313E-04	$2^{3.99}$	7.4319E-04	$2^{4.00}$	7.4319E-04	$2^{4.00}$	8.0090E-04	$2^{3.97}$
		$\\|u^h - u_h\\|_a$ by PMINRES							
2^{-2}	2^{-4}	2.8099E-02	-	5.4870E-03	-	3.6261E-02	-	2.0456E-02	-
2^{-3}	2^{-6}	7.7001E-03	$2^{1.87}$	3.8883E-04	$2^{3.82}$	1.1532E-02	$2^{1.65}$	6.7053E-03	$2^{1.61}$
2^{-4}	2^{-8}	1.9918E-03	$2^{1.95}$	2.5223E-05	$2^{3.95}$	2.9437E-03	$2^{1.97}$	1.9493E-03	$2^{1.78}$
		PMINRES CPU Time (sec)							
2^{-2}	2^{-4}	0.0054	-	0.0054	-	0.0017	-	0.0017	-
2^{-3}	2^{-6}	0.0465	-	0.0555	-	0.0121	-	0.0129	-
2^{-4}	2^{-8}	1.7407	-	2.4632	-	0.4701	-	0.5044	-
		PMINRES Steps							
2^{-2}	2^{-4}	21	-	24	-	8	-	5	-
2^{-3}	2^{-6}	74	-	89	-	32	-	19	-
2^{-4}	2^{-8}	260	-	350	-	138	-	70	-

[1] MINRES suggests to use a bigger tolerance, the actual relative residual norm is 5.1201×10^{-8}.

in the mesh refinement process (see Table 4.2). We also consider unstructured mesh case (see Table 4.3). We observe the absolute eigenvalue error and the a-norm eigenvector error.

From Table 4.2 and Table 4.3, we can see that in all cases, our inexact scheme still keeps the optimal accuracy while using less CPU time. For example, in the last case when $H = 2^{-7}$ and $h = 2^{-8}$, our methods have the same accuracy with XZ (both of them are of $\mathcal{O}(10^{-8})$), but is twice as faster as XZ's scheme. Although not as accurate as HC, but it is fourth times as faster (ITG's 1.3690 seconds and IATG's 0.6144 seconds compared to HC's 1.9028 seconds), while still keeping the optimal scaling behavior.

Table 4.2: Results for Example 4.1 on a structured grid with $h = \frac{H}{2}$.

H	$h=\frac{H}{2}$	XZ	HC	ITG(H^2)	IATG(.5)
		$\lvert\lambda^h-\lambda\rvert$			
2^{-2}	2^{-3}	7.7437E-01	7.6650E-01	7.6650E-01	7.7072E-01
2^{-3}	2^{-4}	1.9109E-01	1.9058E-01	1.9058E-01	1.9160E-01
2^{-4}	2^{-5}	4.7617E-02	4.7583E-02	4.7583E-02	4.7660E-02
2^{-5}	2^{-6}	1.1894E-02	1.1892E-02	1.1892E-02	1.1896E-02
2^{-6}	2^{-7}	2.9729E-03	2.9728E-03	2.9728E-03	2.9731E-03
2^{-7}	2^{-8}	7.4319E-04	7.4318E-04	7.4318E-04	7.4320E-04
		$\lVert u^h-u_h\rVert_a$			
2^{-2}	2^{-3}	2.1829E-02	3.1872E-03	3.1878E-03	1.5960E-02
2^{-3}	2^{-4}	5.7486E-03	2.1671E-04	2.2730E-04	8.0955E-03
2^{-4}	2^{-5}	1.4906E-03	1.4138E-05	1.5020E-05	2.2437E-03
2^{-5}	2^{-6}	3.7669E-04	8.9454E-07	9.1759E-07	5.3112E-04
2^{-6}	2^{-7}	9.4438E-05	5.6087E-08	5.7348E-08	1.4030E-04
2^{-7}	2^{-8}	2.3626E-05	3.5082E-09	3.5740E-09	3.5184E-05
		CPU Time (sec)			
2^{-2}	2^{-3}	0.0023	0.0023	0.0016	0.0014
2^{-3}	2^{-4}	0.0037	0.0041	0.0023	0.0018
2^{-4}	2^{-5}	0.0082	0.0101	0.0052	0.0033
2^{-5}	2^{-6}	0.0349	0.0538	0.0274	0.0148
2^{-6}	2^{-7}	0.1441	0.2684	0.1471	0.0690
2^{-7}	2^{-8}	1.2583	2.2126	1.3690	0.6114
		MINRES Steps			
2^{-2}	2^{-3}	13	15	4	5
2^{-3}	2^{-4}	31	37	9	9
2^{-4}	2^{-5}	58	75	17	18
2^{-5}	2^{-6}	102	147	34	37
2^{-6}	2^{-7}	171	285[1]	68	73
2^{-7}	2^{-8}	325	531[2]	136	146

[1] Actual relative residual norm is 1.0990×10^{-7}

[2] Actual relative residual norm is 4.9895×10^{-6}.

Table 4.3: Results for Example 4.1 on a unstructured grid.

Coarse	Fine	XZ	HC	ITG(H^2)	IATG(.5)
		$\lvert\lambda^h-\lambda\rvert$			
Fig. 4.1a	Fig. 4.1b	7.3364E-04	1.1856E-06	1.0020E-04	1.4545E-03
Fig. 4.1c	Fig. 4.1d	3.7122E-05	4.5947E-09	1.2714E-06	2.2122E-04
		$\lVert u^h-u_h\rVert_a$			
Fig. 4.1a	Fig. 4.1b	6.9025E-03	2.8037E-04	2.8321E-03	1.0478E-02
Fig. 4.1c	Fig. 4.1d	1.5485E-03	1.7350E-05	2.9371E-04	4.2423E-03
		CPU Time (sec)			
Fig. 4.1a	Fig. 4.1b	0.0069	0.0078	0.0030	0.0025
Fig. 4.1c	Fig. 4.1d	0.0195	0.0264	0.0112	0.0081

Third, in order to verify the different choices stopping criteria that affects the resulting solution, we choose different stopping criteria ($\tau = H, H^2$, and H^3) for ITG comparing with XZ's scheme in Table 4.4 and ($\tau = .5, H$ and H^2) for IATG comparing with HC's scheme in

Table 4.4: Results for Example 4.1 on a structured grid with $h = H^2$, with different stopping criteria for ITG.

H	$h = \frac{H}{2}$	XZ	ITG(H)	ITG(H^2)	ITG(H^3)
		$\lvert\lambda^h - \lambda\rvert$			
2^{-2}	2^{-4}	2.0313E-01	3.4477E-01	2.2591E-01	2.0409E-01
2^{-3}	2^{-6}	1.2795E-02	3.3069E-02	1.4438E-02	1.2816E-02
2^{-4}	2^{-8}	8.0313E-04	3.0227E-03	9.1624E-04	8.0364E-04
		$\|u^h - u_h\|_a$			
2^{-2}	2^{-4}	2.8099E-02	9.8190E-02	4.8006E-02	2.9238E-02
2^{-3}	2^{-6}	7.7001E-03	3.6018E-02	1.3064E-02	7.7949E-03
2^{-4}	2^{-8}	1.9918E-03	1.1539E-02	3.4121E-03	2.0007E-03
		CPU Time (sec)			
2^{-2}	2^{-4}	0.0040	0.0013	0.0017	0.0021
2^{-3}	2^{-6}	0.0434	0.0052	0.0122	0.0200
2^{-4}	2^{-8}	1.7866	0.0931	0.5008	0.8213

Table 4.5: Results for Example 4.1 on a structured grid with $h = H^2$, with different stopping criteria for IATG.

H	$h = \frac{H}{2}$	HC	IATG(.5)	IATG(H)	IATG(H^2)
		$\lvert\lambda^h - \lambda\rvert$			
2^{-2}	2^{-4}	1.9105E-01	2.1452E-01	1.9242E-01	1.9104E-01
2^{-3}	2^{-6}	1.1894E-02	1.3931E-02	1.1914E-02	1.1894E-02
2^{-4}	2^{-8}	7.4319E-04	8.9412E-04	7.4345E-04	7.4319E-04
		$\|u^h - u_h\|_a$			
2^{-2}	2^{-4}	5.4870E-03	3.9502E-02	1.0357E-02	5.3730E-03
2^{-3}	2^{-6}	3.8883E-04	1.1669E-02	1.1380E-03	3.9184E-04
2^{-4}	2^{-8}	2.5223E-05	3.1794E-03	1.2333E-04	2.6062E-05
		CPU Time (sec)			
2^{-2}	2^{-4}	0.0048	0.0019	0.0024	0.0022
2^{-3}	2^{-6}	0.0575	0.0137	0.0185	0.0222
2^{-4}	2^{-8}	2.6344	0.5204	0.8347	1.1105

Table 4.5.

Table 4.4 shows that the correction step in original XZ's two-grid scheme allows to be solved inexactly at the stopping criteria $\mathcal{O}(H^2)$. The resulting solution is still asymptotically optimal. Even the tolerance is set at $\mathcal{O}(H^3)$, on accuracy hand, better approximation is observed. Yet the order is not raised. Even in this case, the computational cost is still almost as half as the original XZ's scheme. Similarly, in Table 4.5, we see that for IATG(.5), the scaling behavior is already optimal as is shown in 4.1. When the tolerance τ is getting smaller, IATG converges to HC's scheme. When $\tau = H^2$, the accuracy of IATG(H^2) is almost the same as HC, however, the cost in time is roughly half.

Example 4.2. The second model problem is the same with Example 4.1, but defined on an L-shaped domain $\Omega = (-1, 1) \times (-1, 1) \setminus (0, 1) \times (-1, 0)$.

Discretization is based on an adaptive local refinement procedure. The coarsest grid is based on a uniform grid as is shown in Fig. 4.2a. Note that the eigenvector has singularity near the

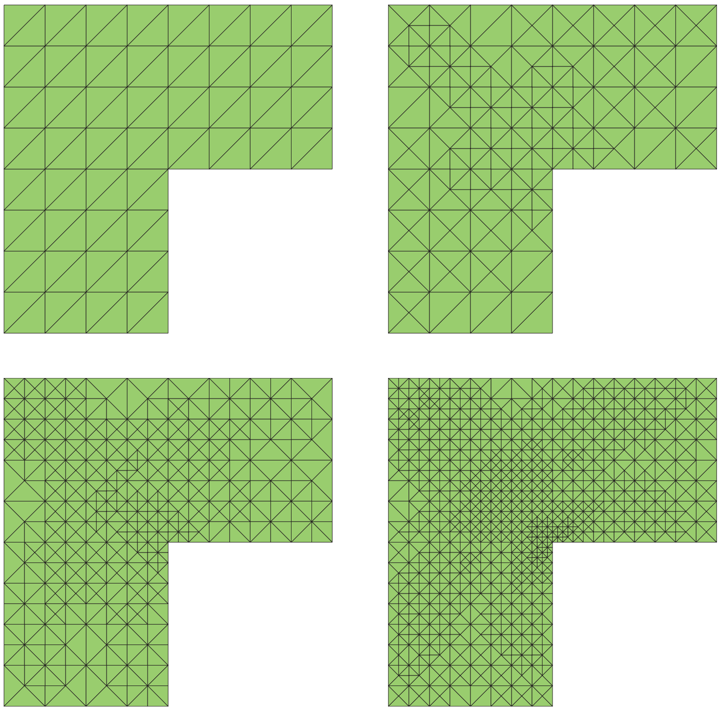

Fig. 4.2. Coarse and fine grids for an L-shaped domain. Fig. 4.2a is the coarse grid. Figs. 4.2b, 4.2c and 4.2d.

Table 4.6: Results for Example 4.2 on a unstructured grid.

Coarse	Fine	XZ	HC	ITG(H^2)	IATG(.5)
			$\|\lambda^h - \lambda_h\|$		
Fig. 4.2a	Fig. 4.2b	1.9294E-03	1.3119E-05	8.7101E-03	4.7381E-03
Fig. 4.2a	Fig. 4.2c	2.1057E-03	2.2708E-05	5.2978E-03	1.5317E-02
Fig. 4.2a	Fig. 4.2d	2.3274E-03	3.1546E-05	9.0203E-03	1.6623E-02
			$\|u^h - u_h\|_a$		
Fig. 4.2a	Fig. 4.2b	1.6635E-02	1.4853E-03	3.7904E-02	2.8101E-02
Fig. 4.2a	Fig. 4.2c	1.6492E-02	1.7554E-03	2.7425E-02	5.2682E-02
Fig. 4.2a	Fig. 4.2d	1.7334E-02	2.0458E-03	3.5005E-02	5.6284E-02
			CPU Time (sec)		
Fig. 4.2a	Fig. 4.2b	0.0041	0.0048	0.0016	0.0018
Fig. 4.2a	Fig. 4.2c	0.0061	0.0078	0.0022	0.0025
Fig. 4.2a	Fig. 4.2d	0.0115	0.0148	0.0031	0.0040

origin, hence the fine grids are generated by an adaptive refinement procedure (see Figs. 4.2b, 4.2c and 4.2d).

The numerical results are shown in Table 4.6.

Obviously, we can see that our inexact two-grid scheme has advantages in cutting computational cost while still keeping the precision of the solution.

Table 4.7: Results for Example 4.3 on a structured grid.

H	$h=H^2$	i	XZ	HC	ITG(H^2)	IATG(.5)
			$\lvert\lambda^{h,i}-\lambda_i\rvert$			
2^{-2}	2^{-4}	2	5.4324E-02	5.0914E-02	7.0535E-02	5.6216E-02
		3	8.2578E-02	7.8944E-02	9.5596E-02	9.1821E-02
2^{-3}	2^{-6}	2	3.2200E-03	2.9485E-03	4.3552E-03	3.4370E-03
		3	4.2698E-03	3.9940E-03	5.4253E-03	5.0670E-03
H	$h=\frac{H}{2}$	i	XZ	HC	ITG(H^2)	IATG(.5)
			$\lvert\lambda^{h,i}-\lambda_i\rvert$			
2^{-2}	2^{-3}	2	2.0649E-01	2.0436E-01	2.1847E-01	2.0684E-01
		3	3.2264E-01	3.2026E-01	3.3540E-01	3.2957E-01
2^{-3}	2^{-4}	2	5.1035E-02	5.0879E-02	5.1890E-02	5.1154E-02
		3	7.9008E-02	7.8848E-02	7.9880E-02	7.9525E-02
2^{-4}	2^{-5}	2	1.2544E-02	1.2534E-02	1.2607E-02	1.2549E-02
		3	1.8953E-02	1.8942E-02	1.9016E-02	1.8986E-02
2^{-5}	2^{-6}	2	2.9489E-03	2.9483E-03	2.9530E-03	2.9494E-03
		3	3.9942E-03	3.9935E-03	3.9982E-03	3.9961E-03
H	$h=H^2$	i	XZ	HC	ITG(H^2)	IATG(.5)
			CPU Times (sec)			
2^{-2}	2^{-4}	2	0.0083	0.0101	0.0019	0.0025
		3	0.0082	0.0105	0.0019	0.0021
2^{-3}	2^{-6}	2	0.1954	0.2694	0.0277	0.0480
		3	0.1982	0.2663	0.0268	0.0350
H	$h=\frac{H}{2}$	i	XZ	HC	ITG(H^2)	IATG(.5)
			CPU Times (sec)			
2^{-2}	2^{-3}	2	0.0037	0.0040	0.0012	0.0014
		3	0.0035	0.0041	0.0011	0.0012
2^{-3}	2^{-4}	2	0.0083	0.0100	0.0019	0.0032
		3	0.0079	0.0101	0.0020	0.0024
2^{-4}	2^{-5}	2	0.0386	0.0532	0.0071	0.0122
		3	0.0385	0.0533	0.0072	0.0096
2^{-5}	2^{-6}	2	0.1743	0.2884	0.0292	0.0553
		3	0.1740	0.2776	0.0298	0.0434

Example 4.3. *The third model problem takes the following form*

$$\begin{cases} -\partial_{xx}u-(1+\delta^2)\delta_{yy}u=\lambda u, & x\in\Omega, \\ u=0, & x\in\partial\Omega, \end{cases} \tag{4.4}$$

where $\Omega=(-1,1)\times(-1,1)$. *The second and third eigenvalue is*

$$\lambda_2=(5+\delta)\pi^2/4, \qquad \textit{and} \qquad \lambda_3=(5/4+\delta)\pi^2, \tag{4.5}$$

where $\delta=10^{-5}$.

In this example, we consider an eigenvalue problem with scaled Laplacian. We are interested in the second and the third eigen-pair. Note that the distance between these two eigenvalues is $\frac{3}{4}\pi^2\delta$. We tested both $h=H/2$, $h=H^2$ cases, for the four two-grid schemes. Numerical experiments in Table 4.7 also reflects that the inexact scheme is efficient and robust for eigenvalue problems.

Acknowledgements. The authors are grateful to Prof. Zhaojun Bai in University of California, Davis, and Prof. Carlos J. García-Cervera in University of California, Santa Barbara for their helpful discussions. The authors are grateful to the editor and the referees for their valuable comments, which improves the quality of the paper greatly. Weiguo Gao is supported by the National Natural Science Foundation of China under grants 91330202, Special Funds for Major State Basic Research Projects of China (2015CB858560003), and Shanghai Science and Technology Development Funds 13dz2260200 and 13511504300. Qun Gu acknowledges the financial support from China Scholarship Council (No. 2011610055).

References

[1] J. Xu, A new class of iterative methods for nonselfadjoint or indefinite problems, *SIAM J. Numer. Anal.*, **29**:2 (1992), 303–319.

[2] J. Xu, A novel two-grid method for semilinear elliptic equations, *SIAM J. Sci. Comput.*, **15**:1 (1994), 231–237.

[3] J. Xu, Two-grid discretization techniques for linear and nonlinear pdes, *SIAM J. Numer. Anal.*, **33**:5 (1996), 1759–1777.

[4] O. Axelsson and W. Layton, A two-level discretization of nonlinear boundary value problems, *SIAM J. Numer. Anal.*, **33**:6 (1996), 2359–2374.

[5] C.N. Dawson and M.F. Wheeler, Two-grid methods for mixed finite element approximations of nonlinear parabolic equations, *Contemp. Math.*, **180** (1994), 191–203.

[6] W. Layton and W. Lenferink, Two-level Picard and modified Picard methods for the Navier-Stokes equations, *Appl. Math. Comput.*, **69**:2 (1995), 263–274.

[7] T. Utnes, Two-grid finite element formulations of the incompressible Navier–Stokes equations, *Commun. Numer. Meth. Engng.*, **13**:8 (1997), 675–684.

[8] W. Layton and L. Tobiska, A two-level method with backtracking for the Navier–Stokes equations, *SIAM J. Numer. Anal.*, **35**:5 (1998), 2035–2054.

[9] M. Marion and J. Xu, Error estimates on a new nonlinear Galerkin method based on two-grid finite elements, *SIAM J. Numer. Anal.*, **32**:4 (1995), 1170–1184.

[10] C.N. Dawson, M.F. Wheeler and C.S. Woodward, A two-grid finite difference scheme for nonlinear parabolic equations, *SIAM J. Numer. Anal.*, **35**:2 (1998), 435–452.

[11] C. Chien and B. Jeng, A two-grid discretization scheme for semilinear elliptic eigenvalue problems, *SIAM J. Sci. Comput.*, **27**:4 (2006), 1287–1304.

[12] J. Jin, S. Shu and J. Xu, A two-grid discretization method for decoupling systems of partial differential equations, *Math. Comp.*, **75**:256 (2006), 1617–1626.

[13] J. Xu and A. Zhou, Local and parallel finite element algorithms based on two-grid discretizations, *Math. Comp.*, **69**:231 (2000), 881–909.

[14] J. Xu and A. Zhou, Local and parallel finite element algorithms based on two-grid discretizations for nonlinear problems, *Adv. Comput. Math.*, **14**:4 (2001), 293–327.

[15] J. Xu and A. Zhou, A two-grid discretization scheme for eigenvalue problems, *Math. Comp.*, **70**:233 (2001), 17–25.

[16] Q. Lin and G. Xie, Accelerating the finite element method in eigenvalue problems, *Chinese Sci. Bull.*, **26**:8 (1981).

[17] J. Xu and A. Zhou, Local and parallel finite element algorithms for eigenvalue problems, *Acta Math. Appl. Sin-E*, **18**:2 (2002), 185–200.

[18] X. Dai and A. Zhou, Three-scale finite element discretizations for quantum eigenvalue problems, *SIAM J. Numer. Anal.*, **46**:1 (2008), 295–324.

[19] X. Hu and X. Cheng, Acceleration of a two-grid method for eigenvalue problems, *Math. Comput.*, **80**:275 (2011), 1287–1301.

[20] Q. Lin and H. Xie, A multi-level correction scheme for eigenvalue problems, *Math. Comput.*, **84**:291 (2015), 71–88.

[21] W. Hackbusch, Multi-Grid Methods and Applications, vol. 4, Springer-Verlag, Berlin, 1985.

[22] I. Babuška and J. Osborn, Finite element-galerkin approximation of the eigenvalues and eigenvectors of selfadjoint problems, *Math. Comp.*, **52**:186 (1989), 275–297.

[23] I. Babuška and J. Osborn, Eigenvalue problems, *Handbook of Numerical Analysis II*, 641–784, 1991.

[24] F. Chatelin, Spectral Approximation of Linear Operators, SIAM, 1983.

[25] Y. Yang and H. Bi, Two-grid finite element discretization schemes based on shifted-inverse power method for elliptic eigenvalue problems, *SIAM J. Numer. Anal.*, **49**:4 (2011), 1602–1624.

[26] Z. Jia, On convergence of the inexact rayleigh quotient iteration with minres, *J. Comput. Appl. Math.*, **236**:17 (2012), 4276–4295.

[27] H. Chen, Y. He, Y. Li and H. Xie, A multigrid method for eigenvalue problems based on shifted-inverse power technique, *Eur. J. Math.*,**1**:207–228, 2015.

[28] L. Chen and C.S. Zhang, AFEM@MATLAB: A MATLAB package of adaptive finite element methods, Technical Report, Dept. Math., Univ. Maryland, 2006.