Journal of Computational Mathematics
Vol.35, No.1, 2017, 74–90.

http://www.global-sci.org/jcm
doi:10.4208/jcm.1608-m2014-0135

A CASCADIC MULTIGRID METHOD FOR EIGENVALUE PROBLEM*

Xiaole Han
IAPCM, Institute of Applied Physics and Computational Mathematics, Beijing 100094, China
Email: hanxiaole@lsec.cc.ac.cn
Hehu Xie
LSEC, NCMIS, Institute of Computational Mathematics, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing 100190, China
Email: hhxie@lsec.cc.ac.cn
Fei Xu
Beijing Institute for Scientific and Engineering Computing, Beijing University of Technology, Beijing 100124, China
Email: xufei@lsec.cc.ac.cn

Abstract

A cascadic multigrid method is proposed for eigenvalue problems based on the multilevel correction scheme. With this new scheme, an eigenvalue problem on the finest space can be solved by linear smoothing steps on a series of multilevel finite element spaces and nonlinear correcting steps on special coarsest spaces. Once the sequence of finite element spaces and the number of smoothing steps are appropriately chosen, the optimal convergence rate with the optimal computational work can be obtained. Some numerical experiments are presented to validate our theoretical analysis.

Mathematics subject classification: 65N30, 65N25, 65L15, 65B99.
Key words: Eigenvalue problem, Cascadic multigrid, Multilevel correction scheme, Finite element method.

1. Introduction

The cascadic multigrid method proposed by [4,6] and analyzed by [17] is based on a hierarchy of nested meshes. Going from the coarsest level to the finest one, in each level, the discrete approximation obtained from the previous level acts as the starting value of a simple iterative solver (a smoother) like conjugate gradient. It is well known that for some certain linear systems (e.g., descretized by finite element method), a smoother can not eliminate the error effectively, and the part of error hard to be reduced is called *algebraic error*, which has been motivating the research on multigrid method. Therefore, to achieve the desired accuracy, the algebraic error on each level must be small enough. In cascadic multigrid method, this is achieved by increasing the number of smoothing iteration steps on coarser levels. Fortunately, the smaller dimensions of the problems on the coarser levels lead to the optimality of the complete algorithm. Requiring the number of operations which is proportional to the number of unknowns on the finest level, the algebraic error of the final approximation solution is of the same order as the discretization error of the finite element method. For more information about the cascadic multigrid method, please refer to [4, 6, 11, 17, 18, 20] and the references cited therein.

* Received November 18, 2014 / Revised version received August 12, 2016 / Accepted August 16, 2016 / Published online January 18, 2017 /

In modern science and engineer, eigenvalue problems appear in many fields such as Physics, Chemistry, mechanics and material sciences. Recently, a type of multilevel correction method is proposed to solve eigenvalue problems in [13, 22]. In this multilevel correction scheme, the solution of eigenvalue problem on the final level mesh can be reduced to a series of solutions of boundary value problems on the multilevel meshes and a series of solutions of the eigenvalue problem on the coarsest mesh. Then it is natural to use the efficient linear solvers such as multigrid method and algebraic multigrid method to design the corresponding efficient eigenvalue solvers. It is well known that the cascadic multigrid method is simple and easy to be implemented. Therefore, the aim of this paper is to construct a cascadic multigrid method to solve the eigenvalue problem by transforming the eigenvalue problem solving to a series of smoothing iteration steps on the sequence of meshes and eigenvalue problem solving on the coarsest mesh by the multilevel correction method. Similarly to the cascadic multigrid for the boundary value problem, we also only do the smoothing steps for a boundary value problem by using the previous eigenpair approximation as the start value. As same as the cascadic multigrid method for boundary value problems, the numbers of smoothing iteration steps need to be increased in the coarse levels. The final eigenpair approximation has the same order algebraic error as the discretization error of the finite element method by organizing the suitable number of smoothing iteration steps on different levels. The original version of this paper is [9]. After that, there also have appeared a different cascadic multigrid method in [19] which is based on the shifted-inverse power iteration [8, 10, 16].

The rest of this paper is organized as follows. In the next section, we introduce the finite element method for the eigenvalue problem and the corresponding error estimates. A cascadic multigrid method for eigenvalue problem based on the multilevel correction scheme is presented and analyzed in Section 3. In Section 4, three numerical examples are presented to validate our theoretical analysis. Some concluding remarks are given in the last section.

2. Finite Element Method for Eigenvalue Problem

This section is devoted to introducing some notation and the finite element method for the eigenvalue problem. In this paper, we shall use the standard notation for Sobolev spaces $W^{s,p}(\Omega)$ and their associated norms and semi-norms ([1]). For $p = 2$, we denote $H^s(\Omega) = W^{s,2}(\Omega)$ and $H_0^1(\Omega) = \{v \in H^1(\Omega) : \ v|_{\partial\Omega} = 0\}$, where $v|_{\Omega} = 0$ is in the sense of trace, $\|\cdot\|_{s,\Omega} = \|\cdot\|_{s,2,\Omega}$. In some places, $\|\cdot\|_{s,2,\Omega}$ should be viewed as piecewise defined if it is necessary. The letter C (with or without subscripts) denotes a generic positive constant independent of mesh size which may be different at its different occurrences through the paper.

For simplicity, we consider the following model problem to illustrate the main idea: Find (λ, u) such that

$$\begin{cases} -\nabla\cdot(\mathcal{A}\nabla u) = \lambda u, & \text{in } \Omega, \\ u = 0, & \text{on } \partial\Omega, \end{cases} \tag{2.1}$$

where $\mathcal{A}$ is a symmetric and positive definite matrix with suitable regularity, $\Omega \subset \mathcal{R}^d (d = 2, 3)$ is a bounded domain with Lipschitz boundary $\partial\Omega$ and ∇, $\nabla\cdot$ denote the gradient, divergence operators, respectively.

In order to use the finite element method to solve the eigenvalue problem (2.1), we need to define the corresponding variational form as follows: Find $(\lambda, u) \in \mathcal{R}\times V$ such that $b(u, u) = 1$ and

$$a(u, v) = \lambda b(u, v), \quad \forall v \in V, \tag{2.2}$$

where $V := H_0^1(\Omega)$ and

$$a(u,v) = \int_\Omega \nabla u \cdot \mathcal{A}\nabla v d\Omega, \qquad b(u,v) = \int_\Omega uv d\Omega. \tag{2.3}$$

The norms $\|\cdot\|_a$ and $\|\cdot\|_b$ are defined by

$$\|v\|_a = a(v,v)^{1/2} \quad \text{and} \quad \|v\|_b = b(v,v)^{1/2}.$$

It is well known that the eigenvalue problem (2.2) has an eigenvalue sequence $\{\lambda_j\}$ (cf. [3,7]):

$$0 < \lambda_1 \le \lambda_2 \le \cdots \le \lambda_k \le \cdots, \quad \lim_{k\to\infty} \lambda_k = \infty,$$

and associated eigenfunctions

$$u_1, u_2, \cdots, u_k, \cdots,$$

where $b(u_i, u_j) = \delta_{ij}$ (δ_{ij} denotes the Kronecker function). In the sequence $\{\lambda_j\}$, the λ_j are repeated according to their geometric multiplicity.

Now, let us define the finite element approximations of the problem (2.2). First we generate a shape-regular triangulation $\mathcal{T}_h$ of the computing domain $\Omega \subset \mathcal{R}^d$ ($d = 2,3$) into triangles or rectangles for $d = 2$ (tetrahedrons or hexahedrons for $d = 3$). The diameter of a cell $K \in \mathcal{T}_h$ is denoted by h_K and the mesh size h describes the maximum diameter of all cells $K \in \mathcal{T}_h$. Based on the mesh $\mathcal{T}_h$, we can construct a finite element space denoted by $V_h \subset V$. For simplicity, we set V_h as the linear finite element space which is defined as follows

$$V_h = \Big\{ v_h \in C(\Omega) \ \Big| \ v_h|_K \in \mathcal{P}_1, \ \ \forall K \in \mathcal{T}_h \Big\} \cap H_0^1(\Omega), \tag{2.4}$$

where $\mathcal{P}_1$ denotes the linear function space.

The standard finite element scheme for eigenvalue problem (2.2) is: Find $(\bar\lambda_h, \bar u_h) \in \mathcal{R} \times V_h$ such that $b(\bar u_h, \bar u_h) = 1$ and

$$a(\bar u_h, v_h) = \bar\lambda_h b(\bar u_h, v_h), \qquad \forall v_h \in V_h. \tag{2.5}$$

From [2,3], we have the following Rayleigh quotient expression for $\bar\lambda_h$:

$$\bar\lambda_h = \frac{a(\bar u_h, \bar u_h)}{b(\bar u_h, \bar u_h)}, \tag{2.6}$$

and the discrete eigenvalue problem (2.5) has eigenvalues:

$$0 < \bar\lambda_{1,h} \le \bar\lambda_{2,h} \le \cdots \le \bar\lambda_{k,h} \le \cdots \le \bar\lambda_{N_h,h},$$

and corresponding eigenfunctions

$$\bar u_{1,h}, \bar u_{2,h}, \cdots, \bar u_{k,h}, \cdots, \bar u_{N_h,h},$$

where $b(\bar u_{i,h}, \bar u_{j,h}) = \delta_{ij}, 1 \le i,j \le N_h$ (N_h is the dimension of the finite element space V_h).

Let $M(\lambda_i)$ denote the eigenspace corresponding to the eigenvalue λ_i which is defined by

$$M(\lambda_i) = \Big\{ w \in H_0^1(\Omega) : w \text{ is an eigenfunction of (2.2) corresponding to } \lambda_i \text{ and } b(w,w) = 1 \Big\}, \tag{2.7}$$

and define

$$\delta_h(\lambda_i) = \sup_{w\in M(\lambda_i)} \inf_{v\in V_h} \|w-v\|_a. \tag{2.8}$$

Let us define the following quantity:

$$\eta_a(h) = \sup_{f\in L^2(\Omega),\|f\|_b=1} \inf_{v\in V_h} \|Tf-v\|_a, \tag{2.9}$$

where $T: L^2(\Omega)\to V$ is defined as

$$a(Tf,v) = b(f,v), \qquad \forall f\in L^2(\Omega) \quad \text{and} \quad \forall v\in V. \tag{2.10}$$

Then the error estimates for the eigenpair approximations by the finite element method can be described as follows.

Lemma 2.1. ([2 Lemma 3.7], [3,7]) *For any eigenpair approximation* $(\bar{\lambda}_{i,h},\bar{u}_{i,h})$ $(i=1,\cdots,N_h)$ *of* (2.5)*, there exists an exact eigenpair* (λ_i,u_i) *of* (2.2) *such that* $b(u_i,u_i)=1$ *and*

$$\|u_i-\bar{u}_{i,h}\|_a \le C_i\delta_h(\lambda_i), \tag{2.11}$$

$$\|u_i-\bar{u}_{i,h}\|_b \le C_i\eta_a(h)\|u_i-\bar{u}_{i,h}\|_a, \tag{2.12}$$

$$|\lambda_i-\bar{\lambda}_{i,h}| \le C_i\delta_h^2(\lambda_i). \tag{2.13}$$

Here and hereafter C_i *is some constant depending on* i *but independent of the mesh size* h.

The following Rayleigh quotient expansion of the eigenvalue error is the tool to obtain the error estimates of the eigenvalue approximations.

Lemma 2.2. ([2]) *Assume* (λ,u) *is an eigenpair of the eigenvalue problem* (2.2)*. Then for any* $w\in H_0^1(\Omega)\backslash\{0\}$*, the following expansion holds:*

$$\frac{a(w,w)}{b(w,w)}-\lambda = \frac{a(w-u,w-u)}{b(w,w)}-\lambda\frac{(w-u,w-u)}{b(w,w)}. \tag{2.14}$$

3. Cascadic Multi-level Correction Scheme for Eigenvalue Problem

In this section, we propose a type of cascadic multigrid method for eigenvalue problems. The main idea is to approximate the underlying boundary value problems on each level by some simple smoothing iteration steps. In order to describe the cascadic multigrid method, we first introduce the sequence of finite element spaces and the smoothing properties of appropriate smoothers.

In order to do multigrid scheme, we first generate a coarse mesh $\mathcal{T}_H$ with the mesh size H and the coarse linear finite element space V_H is defined on the mesh $\mathcal{T}_H$. Then we define a sequence of triangulations $\mathcal{T}_{h_k}$ of $\Omega\subset\mathcal{R}^d$ determined as follows. Suppose $\mathcal{T}_{h_1}$ (produced from $\mathcal{T}_H$ by regular refinements) is given and let $\mathcal{T}_{h_k}$ be obtained from $\mathcal{T}_{h_{k-1}}$ via some regular refinements (produce β^d subelements) such that

$$h_k \approx \frac{1}{\beta}h_{k-1}, \tag{3.1}$$

where the positive number β denotes the refinement index and is larger than 1. Based on this sequence of meshes, we construct the corresponding nested linear finite element spaces such that

$$V_H \subseteq V_{h_1} \subset V_{h_2} \subset \cdots \subset V_{h_n}. \tag{3.2}$$

The sequence of finite element spaces $V_{h_1} \subset V_{h_2} \subset \cdots \subset V_{h_n}$ and the finite element space V_H have the following relations of approximation accuracy

$$\eta_a(H) \gtrsim \delta_{h_1}(\lambda_i), \quad \delta_{h_k}(\lambda_i) \approx \frac{1}{\beta}\delta_{h_{k-1}}(\lambda_i), \quad k = 2, \cdots, n. \tag{3.3}$$

Remark 3.1. The relation (3.3) is reasonable since we can choose $\delta_{h_k}(\lambda_i) \approx h_k$ $(k = 1, \cdots, n)$. Always the upper bound of the estimate $\delta_{h_k}(\lambda_i) \lesssim h_k$ holds. Recently, we also obtain the lower bound result $\delta_{h_k}(\lambda_i) \gtrsim h_k$ (cf. [15]).

For generality, we introduce a smoothing operator $S_h : V_h \to V_h$ which satisfies the following estimate

$$\begin{cases} \|S_h^m w_h\|_a \le \frac{C}{m^\alpha}\frac{1}{h}\|w_h\|_b, \\ \|S_h^m w_h\|_a \le \|w_h\|_a, \\ \|S_h^m (w_h + v_h)\|_a \le \|S_h^m w_h\|_a + \|S_h^m v_h\|_a, \end{cases} \tag{3.4}$$

where C is a constant independent of h and α is some positive number depending on the choice of smoother. It is proved in [8,16,20] that the symmetric Gauss-Seidel, the SSOR, the damped Jacobi and the Richardson iteration are smoothers in the sense of (3.4) with parameter $\alpha = 1/2$ and the conjugate-gradient iteration is the smoother with $\alpha = 1$ (cf. [17,18]).

Then we define the following notation

$$w_h = Smooth(V_h, f, \xi_h, m, S_h) \tag{3.5}$$

as the smoothing process for the following boundary value problem

$$a(u_h, v_h) = b(f, v_h), \quad \forall v_h \in V_h, \tag{3.6}$$

where ξ_h denote the initial value of the smoothing process, S_h denote the chosen smoothing operator, m the number of the iteration steps and w_h is the output of the smoothing process.

Now, we come to introduce the cascadic multigrid method for the eigenvalue problem (2.2). To state it more clearly, we assume the desired eigenvalue is simple and the computing domain is convex. Then we have the following estimates

$$\eta_a(H) \approx H, \quad \eta_a(h_k) \approx h_k \quad \text{and} \quad \delta_{h_k}(\lambda_i) \approx h_k, \quad k = 1, \cdots, n. \tag{3.7}$$

Assume we have obtained an eigenpair approximation $(\lambda^{h_k}, u^{h_k}) \in \mathcal{R} \times V_{h_k}$. Now we introduce a cascadic type one correction step to improve the accuracy of the current eigenpair approximation $(\lambda^{h_k}, u^{h_k}) \in \mathcal{R} \times V_{h_k}$.

Algorithm 3.1. Cascadic type of One Correction Step

1. Define the following auxiliary source problem: Find $\widehat{u}^{h_{k+1}} \in V_{h_{k+1}}$ such that

$$a(\widehat{u}^{h_{k+1}}, v_{h_{k+1}}) = \lambda^{h_k} b(u^{h_k}, v_{h_{k+1}}), \qquad \forall v_{h_{k+1}} \in V_{h_{k+1}}. \tag{3.8}$$

   In order to solve (3.8), perform the smoothing process (3.5) to obtain a new eigenfuction approximation $\widetilde{u}^{h_{k+1}} \in V_{h_{k+1}}$ by

$$\widetilde{u}^{h_{k+1}} = Smooth(V_{h_{k+1}}, \lambda^{h_k} u^{h_k}, u^{h_k}, m_{k+1}, S_{h_{k+1}}). \tag{3.9}$$

2. Define a new finite element space $V_H^{h_{k+1}} = V_H + \text{span}\{\widetilde{u}^{h_{k+1}}\}$ and solve the following eigenvalue problem: Find $(\lambda^{h_{k+1}}, u^{h_{k+1}}) \in \mathcal{R} \times V_H^{h_{k+1}}$ such that $b(u^{h_{k+1}}, u^{h_{k+1}}) = 1$ and

$$a(u^{h_{k+1}}, v_H^{h_{k+1}}) = \lambda^{h_{k+1}} b(u^{h_{k+1}}, v_H^{h_{k+1}}), \qquad \forall v_H^{h_{k+1}} \in V_H^{h_{k+1}}. \tag{3.10}$$

Summarize the above two steps by defining

$$(\lambda^{h_{k+1}}, u^{h_{k+1}}) = SmoothCorrection(V_H, V_{h_{k+1}}, \lambda^{h_k}, u^{h_k}, m_{k+1}, S_{h_{k+1}}).$$

Based on the above algorithm, i.e., the cascadic type of one correction step, we can construct a cascadic multigrid method as follows:

Algorithm 3.2. Eigenvalue Cascadic Multigrid Method

1. Find $(\lambda^{h_1}, u^{h_1}) \in \mathcal{R} \times V_{h_1}$ such that

$$a(u^{h_1}, v_{h_1}) = \lambda^{h_1} b(u^{h_1}, v_{h_1}), \quad \forall v_{h_1} \in V_{h_1}.$$

2. For $k = 1, \cdots, n-1$, do the following iteration

$$(\lambda^{h_{k+1}}, u^{h_{k+1}}) = SmoothCorrection(V_H, V_{h_{k+1}}, \lambda^{h_k}, u^{h_k}, m_{k+1}, S_{h_{k+1}}).$$

Finally, we obtain an eigenpair approximation $(\lambda^{h_n}, u^{h_n}) \in \mathcal{R} \times V_{h_n}$.

In order to analyze the convergence of Algorithm 3.2, we introduce an auxiliary algorithm and then show its superapproximate property.

Similarly, assume we have obtained an eigenpair approximations $(\widetilde{\lambda}_{h_k}, \widetilde{u}_{h_k}) \in \mathcal{R} \times V_{h_k}$. We introduce the following auxiliary one correction step.

Algorithm 3.3. Auxiliary One Correction Step

1. Define the following auxiliary source problem: Find $\widehat{u}_{h_{k+1}} \in V_{h_{k+1}}$ such that

$$a(\widehat{u}_{h_{k+1}}, v_{h_{k+1}}) = \widetilde{\lambda}_{h_k} b(\widetilde{u}_{h_k}, v_{h_{k+1}}), \qquad \forall v_{h_{k+1}} \in V_{h_{k+1}}. \tag{3.11}$$

1. Define a new finite element space $\widetilde{V}_{H,h_{k+1}} = V_H + \mathrm{span}\{\widehat{u}_{h_{k+1}}\} + \mathrm{span}\{\widetilde{u}^{h_{k+1}}\}$ and solve the following eigenvalue problem: Find $(\widetilde{\lambda}_{h_{k+1}}, \widetilde{u}_{h_{k+1}}) \in \mathcal{R} \times \widetilde{V}_{H,h_{k+1}}$ such that $b(\widetilde{u}_{h_{k+1}}, \widetilde{u}_{h_{k+1}}) = 1$ and

$$a(\widetilde{u}_{h_{k+1}}, \widetilde{v}_{H,h_{k+1}}) = \widetilde{\lambda}_{h_{k+1}} b(\widetilde{u}_{h_{k+1}}, \widetilde{v}_{H,h_{k+1}}), \qquad \forall \widetilde{v}_{H,h_{k+1}} \in \widetilde{V}_{H,h_{k+1}}. \tag{3.12}$$

Summarize the above two steps by defining

$$(\widetilde{\lambda}_{h_{k+1}}, \widetilde{u}_{h_{k+1}}) = \mathit{AuxiliaryCorrection}(V_H, V_{h_{k+1}}, \widetilde{\lambda}_{h_k}, \widetilde{u}_{h_k}, \widetilde{u}^{h_{k+1}}).$$

Algorithm 3.4. Eigenvalue Auxiliary Multilevel Correction Method

1. Find $(\widetilde{\lambda}_{h_1}, \widetilde{u}_{h_1}) \in \mathcal{R} \times V_{h_1}$ such that

$$a(\widetilde{u}_{h_1}, v_{h_1}) = \widetilde{\lambda}_{h_1} b(\widetilde{u}_{h_1}, v_{h_1}), \quad \forall v_{h_1} \in V_{h_1}.$$

2. For $k = 1, \cdots, n-1$, do the following iteration

$$(\widetilde{\lambda}_{h_{k+1}}, \widetilde{u}_{h_{k+1}}) = \mathit{AuxiliaryCorrection}(V_H, V_{h_{k+1}}, \widetilde{\lambda}_{h_k}, \widetilde{u}_{h_k}, \widetilde{u}^{h_{k+1}}).$$

Finally, we obtain an eigenpair approximation $(\widetilde{\lambda}_{h_n}, \widetilde{u}_{h_n}) \in \mathcal{R} \times V_{h_n}$.

Before analyzing the convergence of Algorithm 3.2, we show a superapproximate property of $\widetilde{u}_{h_k}$ obtained by Algorithm 3.4.

Theorem 3.1. *Assume $\widetilde{u}_{h_k}$ $(k = 1, \cdots, n)$ are obtained by Algorithm 3.4 and $\bar{u}_{h_k}$ $(k = 1, \cdots, n)$ the standard finite element solution in V_{h_k}. If the sequence of finite element spaces $V_{h_1}, \cdots, V_{h_n}$ and the coarse finite element space V_H satisfy the following condition*

$$C\eta_a(H)\beta^2 < 1, \tag{3.13}$$

the following estimate holds

$$\|\bar{u}_{h_k} - \widetilde{u}_{h_k}\|_a \le C\eta_a(h_k)\delta_{h_k}(\lambda), \qquad k = 1, \cdots, n, \tag{3.14}$$

$$\|\bar{u}_{h_k} - \widetilde{u}_{h_k}\|_b \le C\eta_a(H)\eta_a(h_k)\delta_{h_k}(\lambda), \qquad k = 1, \cdots, n, \tag{3.15}$$

where C is a constant only depending on the eigenvalue λ. The eigenvalue approximations $\widetilde{\lambda}_{h_k}$ and $\bar{\lambda}_{h_k}$ have the following estimates

$$\big|\bar{\lambda}_{h_k} - \widetilde{\lambda}_{h_k}\big| \le \|\bar{u}_{h_k} - \widetilde{u}_{h_k}\|_a^2, \qquad k = 1, \cdots, n. \tag{3.16}$$

Proof. Define $\epsilon_{h_k} := |\widetilde{\lambda}_{h_k} - \bar{\lambda}_{h_k}| + \|\widetilde{u}_{h_k} - \bar{u}_{h_k}\|_b$, $k = 1, \cdots, n$. It is obvious that $\epsilon_{h_1} = 0$. From (2.5) and (3.11), we have

$$\begin{aligned}
&\|\bar{u}_{h_{k+1}} - \widehat{u}_{h_{k+1}}\|_a^2 = a(\bar{u}_{h_{k+1}} - \widehat{u}_{h_{k+1}}, \bar{u}_{h_{k+1}} - \widehat{u}_{h_{k+1}}) \\
&= \bar{\lambda}_{h_{k+1}} b(\bar{u}_{h_{k+1}}, \bar{u}_{h_{k+1}} - \widehat{u}_{h_{k+1}}) - \widetilde{\lambda}_{h_k} b(\widetilde{u}_{h_k}, \bar{u}_{h_{k+1}} - \widehat{u}_{h_{k+1}}) \\
&= b(\bar{\lambda}_{h_{k+1}} \bar{u}_{h_{k+1}} - \widetilde{\lambda}_{h_k} \widetilde{u}_{h_k}, \bar{u}_{h_{k+1}} - \widehat{u}_{h_{k+1}}) \\
&\le C\|\bar{\lambda}_{h_{k+1}} \bar{u}_{h_{k+1}} - \widetilde{\lambda}_{h_k} \widetilde{u}_{h_k}\|_b \|\bar{u}_{h_{k+1}} - \widehat{u}_{h_{k+1}}\|_a,
\end{aligned} \tag{3.17}$$

where Poincaré's inequality is used in the last inequality above. Note that the eigenvalue problem (3.12) can be regarded as a finite dimensional approximation of the eigenvalue problem (2.5). Similarly to Lemma 2.1 (see [2,13]), from the second step in Algorithm 3.3, the following estimate holds

$$\begin{aligned}&\|\bar{u}_{h_{k+1}}-\widetilde{u}_{h_{k+1}}\|_a\\ &\leq C\inf_{\widetilde{v}_{H,h_{k+1}}\in\widetilde{V}_{H,h_{k+1}}}\|\bar{u}_{h_{k+1}}-\widetilde{u}_{H,h_{k+1}}\|_a\leq C\|\bar{u}_{h_{k+1}}-\widehat{u}_{h_{k+1}}\|_a.\end{aligned}\tag{3.18}$$

Then combining (3.17) and (3.18) leads to

$$\begin{aligned}\|\bar{u}_{h_{k+1}}-\widetilde{u}_{h_{k+1}}\|_a&\leq C\|\bar{\lambda}_{h_{k+1}}\bar{u}_{h_{k+1}}-\widetilde{\lambda}_{h_k}\widetilde{u}_{h_k}\|_b\\ &=C\|\bar{\lambda}_{h_{k+1}}\bar{u}_{h_{k+1}}-\bar{\lambda}_{h_{k+1}}\widetilde{u}_{h_k}+\bar{\lambda}_{h_{k+1}}\widetilde{u}_{h_k}-\widetilde{\lambda}_{h_k}\widetilde{u}_{h_k}\|_b\\ &\leq C\Big(|\bar{\lambda}_{h_{k+1}}-\widetilde{\lambda}_{h_k}|+\|\bar{u}_{h_{k+1}}-\widetilde{u}_{h_k}\|_b\Big)\\ &\leq C\Big(|\bar{\lambda}_{h_{k+1}}-\bar{\lambda}_{h_k}|+|\bar{\lambda}_{h_k}-\widetilde{\lambda}_{h_k}|+\|\bar{u}_{h_{k+1}}-\bar{u}_{h_k}\|_b+\|\bar{u}_{h_k}-\widetilde{u}_{h_k}\|_b\Big)\\ &\leq C\Big(|\bar{\lambda}_{h_{k+1}}-\bar{\lambda}_{h_k}|+\|\bar{u}_{h_{k+1}}-\bar{u}_{h_k}\|_b+\epsilon_{h_k}\Big).\end{aligned}\tag{3.19}$$

From the properties of $V_{h_k}\subset V_{h_{k+1}}$, $V_{H,h_k}\subset V_{h_k}$, Lemma 2.1 and (3.3), we have

$$\begin{aligned}&\|\bar{u}_{h_{k+1}}-\bar{u}_{h_k}\|_a\leq C\delta_{h_k}(\lambda),\\ &\|\bar{u}_{h_{k+1}}-\bar{u}_{h_k}\|_b\leq C\eta_a(h_k)\|\bar{u}_{h_{k+1}}-\bar{u}_{h_k}\|_a,\\ &\big|\bar{\lambda}_{h_{k+1}}-\bar{\lambda}_{h_k}\big|\leq C\|\bar{u}_{h_{k+1}}-\bar{u}_{h_k}\|_a^2\leq C\delta_{h_k}(\lambda)^2\leq C\eta_a(h_k)\delta_{h_k}(\lambda)\\ &\|\bar{u}_{h_k}-\widetilde{u}_{h_k}\|_b\leq C\eta_a(H)\|\bar{u}_{h_k}-\widetilde{u}_{h_k}\|_a,\\ &\big|\bar{\lambda}_{h_k}-\widetilde{\lambda}_{h_k}\big|\leq C\|\bar{u}_{h_k}-\widetilde{u}_{h_k}\|_a^2.\end{aligned}$$

Substituting above inequalities into (3.19) leads to the following estimates

$$\begin{aligned}\|\bar{u}_{h_{k+1}}-\widetilde{u}_{h_{k+1}}\|_a&\leq C\Big(\delta_{h_k}^2(\lambda)+\eta_a(h_k)\delta_{h_k}(\lambda)+\epsilon_{h_k}\Big)\\ &\leq C\Big(\eta_a(h_k)\delta_{h_k}(\lambda)+\eta_a(H)\|\bar{u}_{h_k}-\widetilde{u}_{h_k}\|_a\Big).\end{aligned}\tag{3.20}$$

When $k=1$, since $\widetilde{u}_{h_1}:=\bar{u}_{h_1}$ and $\widetilde{\lambda}_{h_1}:=\bar{\lambda}_{h_1}$, we have

$$\|\bar{u}_{h_2}-\widetilde{u}_{h_2}\|_a\leq C\eta_a(h_1)\delta_{h_1}(\lambda).\tag{3.21}$$

Based on (3.3), (3.20), (3.21) and recursive argument, we have the following estimates:

$$\begin{aligned}\|\bar{u}_{h_k}-\widetilde{u}_{h_k}\|_a&\leq C\sum_{j=2}^{k}C^{k-j}\eta_a^{k-j}(H)\eta_a(h_{j-1})\delta_{h_{j-1}}(\lambda)\\ &\leq C\sum_{j=2}^{k}C^{k-j}\eta_a^{k-j}(H)\beta^{k-j+1}\eta_a(h_k)\beta^{k-j+1}\delta_{h_k}(\lambda)\\ &\leq C\beta^2\Big(\sum_{j=2}^{k}\big(C\eta_a(H)\beta^2\big)^{k-j}\Big)\eta_a(h_k)\delta_{h_k}(\lambda)\\ &\leq\frac{C\beta^2}{1-C\beta^2\eta_a(H)}\eta_a(h_k)\delta_{h_k}(\lambda).\end{aligned}\tag{3.22}$$

Therefore, the desired result (3.14) holds under the condition $C\eta_a(H)\beta^2 < 1$. Furthermore, (3.15) and (3.16) can be obtained directly from Lemmas 2.1 and 2.2, respectively. Then the proof is completed. $\Box$

Note that $V_H^{h_k} \subset \widetilde{V}_{H,h_k}$, then we can obtain the following estimates which play an important role in our analysis.

Lemma 3.1. ([2]) *Let u^{h_k}, $V_H^{h_k}$ and $\widetilde{u}_{h_k}$, $\widetilde{V}_{H,h_k}$ be defined in Algorithms 3.1 and 3.3. Then the following estimates hold:*

$$\|u^{h_k} - \widetilde{u}_{h_k}\|_a \le C\|\widehat{u}_{h_k} - \widetilde{u}^{h_k}\|_a, \tag{3.23}$$

$$\|u^{h_k} - \widetilde{u}_{h_k}\|_b \le C\eta_a(H)\|u^{h_k} - \widetilde{u}_{h_k}\|_a, \tag{3.24}$$

$$|\lambda^{h_k} - \widetilde{\lambda}_{h_k}| \le \|u^{h_k} - \widetilde{u}_{h_k}\|_a^2. \tag{3.25}$$

Proof. Since $V_H^{h_k} \subset \widetilde{V}_{H,h_k}$, according to (3.10) and (3.12), u^{h_k} can be viewed as the spectral projection of $\widetilde{u}_{h_k}$ (cf. [2]). Then from Lemma 2.1 and the definitions of $\widetilde{V}_{H,h_k}$ and $V_H^{h_k}$, we have

$$\begin{aligned}\|\widetilde{u}_{h_k} - u^{h_k}\|_a &\le C \inf_{v_H^{h_k} \in V_H^{h_k}} \|\widetilde{u}_{h_k} - v_H^{h_k}\|_a \\ &\le C \inf_{v_H^{h_k} \in V_H^{h_k}} \|\widehat{u}_{h_k} - v_H^{h_k}\|_a \le C\|\widehat{u}_{h_k} - \widetilde{u}^{h_k}\|_a,\end{aligned} \tag{3.26}$$

which is the desired result (3.23). Similarly, we also have (3.24) by the following argument

$$\|\widetilde{u}_{h_k} - u^{h_k}\|_b \le C\eta_a(V_H^{h_k})\|\widetilde{u}_{h_k} - u^{h_k}\|_a \le C\eta_a(H)\|\widetilde{u}_{h_k} - u^{h_k}\|_a,$$

where

$$\eta_a(V_H^{h_k}) := \sup_{f\in L^2(\Omega),\|f\|_b=1} \inf_{v\in V_H^{h_k}} \|Tf - v\|_a \le \eta_a(H).$$

Furthermore, (3.25) can be obtained directly from Lemma 2.2 and the proof is completed. $\Box$

Remark 3.2. Since $V_H \subset V_H^{h_k}$ and $V_H \subset \widetilde{V}_{H,h_k}$, from Lemma 2.1, we have

$$\|u^{h_k} - \widetilde{u}_{h_k}\|_a \le \|u^{h_k} - u\|_a + \|u - \widetilde{u}_{h_k}\|_a \le C\delta_H(\lambda). \tag{3.27}$$

Now, we come to give error estimates for Algorithm 3.2.

Theorem 3.2. *Assume the eigenpair approximation (λ^{h_n}, u^{h_n}) is obtained by Algorithm 3.2, $(\widetilde{\lambda}_{h_n}, \widetilde{u}_{h_n})$ is obtained by Algorithm 3.4 and the smoother selected in each level V_{h_k} satisfy the smoothing property (3.4) for $k = 1, \cdots, n$. Under the conditions of Theorem 3.1, we have the following estimate:*

$$\|\widetilde{u}_{h_n} - u^{h_n}\|_a \le C\sum_{k=2}^{n} \frac{\big(1 + C\eta_a(H)\big)^{n-k}}{m_k^\alpha}\delta_{h_k}(\lambda), \tag{3.28}$$

and the corresponding eigenvalue error estimate

$$\big|\widetilde{\lambda}_{h_n} - \lambda^{h_n}\big| \le C\|\widetilde{u}_{h_n} - u^{h_n}\|_a^2. \tag{3.29}$$

Proof. Define $e_{h_k} := u^{h_k} - \widetilde{u}_{h_k}$ for $k = 1, \cdots, n$. Then it is easy to see that $e_{h_1} = 0$. From Lemma 3.1, the following inequalities hold

$$\begin{aligned}\|e_{h_{k+1}}\|_a &= \|u^{h_{k+1}} - \widetilde{u}_{h_{k+1}}\|_a \le C\|\widehat{u}_{h_{k+1}} - \widetilde{u}^{h_{k+1}}\|_a \\ &\le C\Big(\|\widehat{u}_{h_{k+1}} - \widehat{u}^{h_{k+1}}\|_a + \|\widehat{u}^{h_{k+1}} - \widetilde{u}^{h_{k+1}}\|_a\Big). \end{aligned} \tag{3.30}$$

For the first term in (3.30), together with (3.8), (3.11), Lemma 3.1 and (3.27), we have

$$\begin{aligned}&\|\widehat{u}_{h_{k+1}} - \widehat{u}^{h_{k+1}}\|_a \le C\|\lambda^{h_k} u^{h_k} - \widetilde{\lambda}_{h_k}\widetilde{u}_{h_k}\|_b \\ &\le C\Big(\|u^{h_k} - \widetilde{u}_{h_k}\|_a^2 + \|u^{h_k} - \widetilde{u}_{h_k}\|_b\Big) \le C\eta_a(H)\|u^{h_k} - \widetilde{u}_{h_k}\|_a = C\eta_a(H)\|e_{h_k}\|_a. \end{aligned} \tag{3.31}$$

For the second term in (3.30), due to (3.4) and (3.31), the following estimates hold

$$\begin{aligned}&\|\widehat{u}^{h_{k+1}} - \widetilde{u}^{h_{k+1}}\|_a = \|S_{h_{k+1}}^{m_{k+1}}(\widehat{u}^{h_{k+1}} - u^{h_k})\|_a \\ &\le \|S_{h_{k+1}}^{m_{k+1}}(\widehat{u}^{h_{k+1}} - \widetilde{u}_{h_k})\|_a + \|S_{h_{k+1}}^{m_{k+1}}(\widetilde{u}_{h_k} - u^{h_k})\|_a \\ &\le \|S_{h_{k+1}}^{m_{k+1}}(\widehat{u}^{h_{k+1}} - \widehat{u}_{h_{k+1}})\|_a + \|S_{h_{k+1}}^{m_{k+1}}(\widehat{u}_{h_{k+1}} - \widetilde{u}_{h_k})\|_a + \|\widetilde{u}_{h_k} - u^{h_k}\|_a \\ &\le \|\widehat{u}_{h_{k+1}} - \widehat{u}^{h_{k+1}}\|_a + \frac{C}{m_{k+1}^\alpha}\frac{1}{h_{k+1}}\|\widehat{u}_{h_{k+1}} - \widetilde{u}_{h_k}\|_b + \|\widetilde{u}_{h_k} - u^{h_k}\|_a \\ &\le \Big(1 + C\eta_a(H)\Big)\|e_{h_k}\|_a + \frac{C}{m_{k+1}^\alpha}\frac{1}{h_{k+1}}\|\widehat{u}_{h_{k+1}} - \widetilde{u}_{h_k}\|_b. \end{aligned} \tag{3.32}$$

According to Lemma 2.1, (3.3), Theorem 3.1 and its proof,

$$\begin{aligned}\|\widehat{u}_{h_{k+1}} - \widetilde{u}_{h_k}\|_b &\le \|\widehat{u}_{h_{k+1}} - \bar{u}_{h_{k+1}}\|_b + \|\bar{u}_{h_{k+1}} - \bar{u}_{h_k}\|_b + \|\bar{u}_{h_k} - \widetilde{u}_{h_k}\|_b \\ &\le C\eta_a(h_{k+1})\delta_{h_{k+1}}(\lambda). \end{aligned} \tag{3.33}$$

Combining (3.30), (3.31), (3.32), (3.33) and (3.7), we have

$$\|e_{h_{k+1}}\|_a \le \Big(1 + C\eta_a(H)\Big)\|e_{h_k}\|_a + \frac{C}{m_{k+1}^\alpha}\delta_{h_{k+1}}(\lambda), \quad k = 1, \cdots, n-1. \tag{3.34}$$

Based on (3.34), the fact $e_{h_1} = 0$ and the recursive argument, the following estimates hold

$$\begin{aligned}\|e_{h_n}\| &\le \big(1 + C\eta_a(H)\big)\|e_{h_{n-1}}\|_a + \frac{C}{m_n^\alpha}\delta_{h_n}(\lambda) \\ &\le \big(1 + C\eta_a(H)\big)^2\|e_{h_{n-2}}\|_a + \big(1 + C\eta_a(H)\big)\frac{C}{m_{n-1}^\alpha}\delta_{h_{n-1}}(\lambda) + \frac{C}{m_n^\alpha}\delta_{h_n}(\lambda) \\ &\le C\sum_{k=2}^{n}\big(1 + C\eta_a(H)\big)^{n-k}\frac{1}{m_k^\alpha}\delta_{h_k}(\lambda). \end{aligned}$$

This is the desired result (3.28). The estimate (3.29) can be obtained from Lemma 2.2 and (3.28). □

Corollary 3.1. *Under the conditions of Theorem 3.2, we have the following estimates:*

$$\|\bar{u}_{h_n} - u^{h_n}\|_a \le C\Big(\eta_a(h_n)\delta_{h_n}(\lambda) + \sum_{k=2}^{n}\frac{\big(1 + C\eta_a(H)\big)^{n-k}}{m_k^\alpha}\delta_{h_k}(\lambda)\Big), \tag{3.35}$$

$$|\bar{\lambda}_{h_n} - \lambda^{h_n}| \le \|\bar{u}_{h_n} - u^{h_n}\|_a^2. \tag{3.36}$$

Now we come to estimate the computational work for Algorithm 3.2. Define the dimension of each linear finite element space as

$$N_k := \dim V_{h_k}, \quad k = 1, \cdots, n.$$

Then we have

$$N_k \approx \Big(\frac{h_k}{h_n}\Big)^{-d} N_n = \Big(\frac{1}{\beta}\Big)^{d(n-k)} N_n, \quad k = 1, \cdots, n. \tag{3.37}$$

From Theorem 3.2, in order to control the global error, it is required that the number of iterations in the coarser spaces should be larger than the fine spaces. To give a precise analysis for the final error and complexity estimates, we assume the following inequality holds for the number of iterations in each level mesh:

$$\Big(\frac{h_k}{h_n}\Big)^{\zeta} \le \frac{m_k^{\alpha}}{\bar{m}^{\alpha}} \le \sigma\Big(\frac{h_k}{h_n}\Big)^{\zeta}, \qquad k = 2, \cdots, n-1, \tag{3.38}$$

where $\bar{m} = m_n$, $\sigma > 1$ and $\zeta > 1$ are some appropriate constants.

Now, we give the final error and the complexity estimates for Algorithm 3.2.

Theorem 3.3. *Under the conditions* (3.3), (3.38) *and* $\beta^{1-\zeta}(1 + CH) < 1$, *for any given* $\gamma \in (0, 1]$, *the final error estimate satisfies*

$$\|u^{h_n} - \widetilde{u}_{h_n}\|_a \le \gamma h_n \tag{3.39}$$

if we take

$$\bar{m} > \Big(\frac{CC_{\zeta}}{\gamma}\Big)^{\frac{1}{\alpha}}, \tag{3.40}$$

where $C_{\zeta} = 1/(1 - \beta^{1-\zeta}(1 + CH))$.

Assume the eigenvalue problems solved in the coarse spaces V_H *and* V_{h_1} *need work* M_H *and* M_{h_1}, *respectively. If* $\zeta/\alpha < d$, *the total computational work of Algorithm* 3.2 *can be bounded by* $\mathcal{O}(N_n + M_{h_1} + M_H \log(N_n))$ *and furthermore* $\mathcal{O}(N_n)$ *provided* $M_H \ll N_n$ *and* $M_{h_1} \le N_n$; *while if* $\zeta/\alpha = d$, *the total computational work can be bounded by* $\mathcal{O}(N_n \log(N_n) + M_{h_1} + M_H \log(N_n))$ *and furthermore* $\mathcal{O}(N_n \log(N_n))$ *provided* $M_H \ll N_n$ *and* $M_{h_1} \le N_n$.

Proof. By Theorem 3.2, together with (3.1), (3.7), (3.28) and (3.38), we have the following estimates:

$$\begin{aligned}
&\|u^{h_n} - \widetilde{u}_{h_n}\|_a \\
&\le C\sum_{k=2}^{n}(1 + C\eta_a(H))^{n-k}\frac{1}{m_k^{\alpha}}\delta_{h_k}(\lambda) \le C\sum_{k=2}^{n}(1 + CH)^{n-k}\frac{1}{\bar{m}^{\alpha}}\Big(\frac{h_k}{h_n}\Big)^{-\zeta} h_k \\
&\le C\sum_{k=2}^{n}(1 + CH)^{n-k}\beta^{(n-k)(1-\zeta)}\frac{h_n}{\bar{m}^{\alpha}} = C\frac{h_n}{\bar{m}^{\alpha}}\sum_{k=0}^{n-2}\big(\beta^{1-\zeta}(1 + CH)\big)^k \\
&\le C\frac{h_n}{\bar{m}^{\alpha}}\frac{1}{1 - \beta^{1-\zeta}(1 + CH)}.
\end{aligned} \tag{3.41}$$

When $\beta^{1-\zeta}(1 + CH) < 1$, (3.41) becomes

$$\|u^{h_n} - \bar{u}_{h_n}\|_a \le \frac{CC_{\zeta}}{\bar{m}^{\alpha}} h_n. \tag{3.42}$$

Then it is obvious that we can obtain $\|u^{h_n} - \widetilde{u}_{h_n}\|_a \le \gamma h_n$ when $\bar{m}$ satisfies the condition (3.40).

Let W denote the whole computational work of Algorithm 3.2, w_k the work on the k-th level for $k = 1, \cdots, n$. Based on the definition of Algorithms 3.1 and 3.2, (3.1), (3.38) and (3.37), the following estimates hold

$$\begin{aligned} W = \sum_{k=1}^{n} w_k &\le M_{h_1} + \sum_{k=2}^{n} m_k N_k + M_H \log_\beta(N_n) \\ &\le M_{h_1} + C M_H \log(N_n) + \bar{m}\sigma^{1/\alpha} N_n \sum_{k=2}^{n} \Big(\frac{1}{\beta}\Big)^{(n-k)(d-\zeta/\alpha)}. \end{aligned}$$

Then we know that the computation work W can be bounded by $\mathcal{O}(M_{h_1} + M_H \log(N_n) + N_n)$ when $d - \zeta/\alpha > 0$ and by $\mathcal{O}(M_{h_1} + M_H \log(N_n) + N_n \log(N_n))$ when $d - \zeta/\alpha = 0$. It is also obvious they can be bounded by $\mathcal{O}(N_n)$ and $\mathcal{O}(N_n \log(N_n))$, respectively, if $M_H \ll N_n$ and $M_{h_1} \le N_n$ are provided. □

Corollary 3.2. *Under the same conditions of Theorem* 3.3 *and* (3.40) *holding, if* $Ch_n \le \gamma$, *then we have the following estimate*

$$\|u^{h_n} - \bar{u}_{h_n}\|_a \le 2\gamma h_n. \tag{3.43}$$

If we choose the conjugate gradient method as the smoothing operator, then $\alpha = 1$ and the computation work of Algorithm 3.2 can be bounded by $\mathcal{O}(N_n + M_{h_1} + M_H \log(N_n))$ or $\mathcal{O}(N_n)$ provided $M_H \ll N_n$ and $M_{h_1} \le N_n$ for both $d = 2$ and $d = 3$ when we choose $1 < \zeta < d$.

When the symmetric Gauss-Seidel, the SSOR, the damped Jacobi or the Richardson iteration act as the smoothing operator, we know $\alpha = 1/2$. Then the computation work of Algorithm 3.2 can be bounded by $\mathcal{O}(N_n + M_{h_1} + M_H \log(N_n))$ ($\mathcal{O}(N_n)$ provided $M_H \ll N_n$ and $M_{h_1} \le N_n$) only for $d = 3$ when we choose $1 < \zeta < 3/2$. In the case of $\alpha = 1/2$ and $d = 2$, from Theorem 3.3 and its proof, we can only choose $\zeta = 1$ and then the final error has the estimate

$$\|u^{h_n} - \bar{u}_{h_n}\|_a \le Ch_n |\log(h_n)|$$

and the computational work can only be bounded by $\mathcal{O}(N_n \log(N_n) + M_{h_1} + M_H \log(N_n))$ ($\mathcal{O}(N_n \log(N_n))$ provided $M_H \ll N_n$ and $M_{h_1} \le N_n$.

4. Numerical Tests

In this section, three numerical examples are presented to illustrate the efficiency of the cascadic multigrid scheme (Algorithm 3.2) proposed in this paper. Here, for all three examples, we choose the conjugate-gradient iteration as the smoothing operator ($\alpha = 1$) and the number of iteration steps by

$$m_k = \lceil \sigma \times 2^{\zeta(n-k)} \rceil \quad \text{for } k = 2, \cdots, n$$

with $\sigma = 2$, $\zeta = 1.01$ and $\lceil r \rceil$ denoting the smallest integer which is not less than r.

4.1. Model eigenvalue problem

Here we give the numerical results of the cascadic multigrid scheme for the Laplace eigenvalue problem on a two dimensional domain $\Omega = (0,1) \times (0,1)$. The sequence of finite element spaces are constructed by using linear element on the series of mesh which are produced by regular refinement with $\beta = 2$ (connecting the midpoints of each edge). To investigate the convergence behaviors with different initial meshes, we take two meshes generated by Delaunay method as the initial mesh: one is coarser, the other is finer (see Figure 4.1).

Algorithm 3.2 is applied to solve the eigenvalue problem. For comparison, we also solve the eigenvalue problem by the direct finite element method.

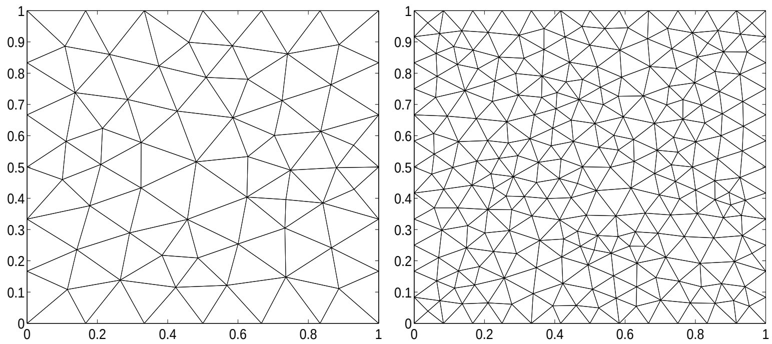

Fig. 4.1. The coarser and finer initial meshes for Example 1

Figure 4.2 gives the corresponding numerical results for the first eigenvalue $\lambda_1 = 2\pi^2$ and the corresponding eigenfunction on the two initial meshes illustrated in Figure 4.1.

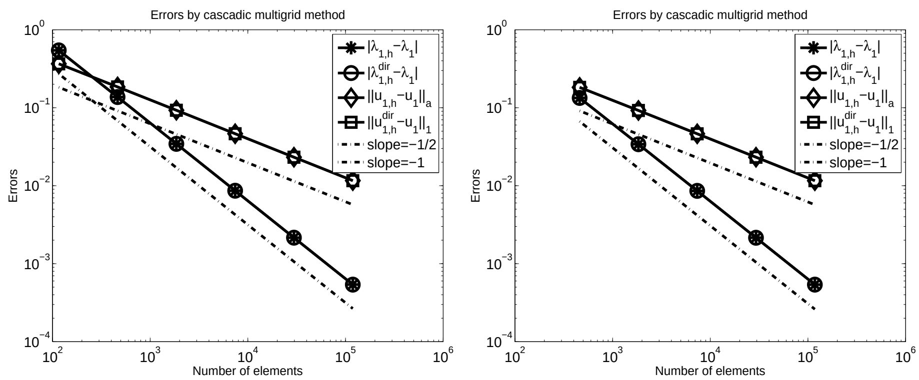

Fig. 4.2. The errors of the cascadic multigrid algorithm for the first eigenvalue $2\pi^2$ and the corresponding eigenfunction, where $u_{1,h}$ and $\lambda_{1,h}$ denote the eigenfunction and eigenvalue approximation by Algorithm 3.2, and $u_{1,h}^{\rm dir}$ and $\lambda_{1,h}^{\rm dir}$ denote the eigenfunction and eigenvalue approximation by direct eigenvalue solving (The left figure corresponds to the left mesh in Figure 4.1 and the right figure corresponds to the right mesh in Figure 4.1)

From Figure 4.2, we find the cascadic multigrid scheme can obtain the optimal error estimates as same as the direct eigenvalue solving method for the eigenvalue and the corresponding eigenfunction approximations. Furthermore, Figure 4.2 also shows the computational work of Algorithm 3.2 can arrive the optimality.

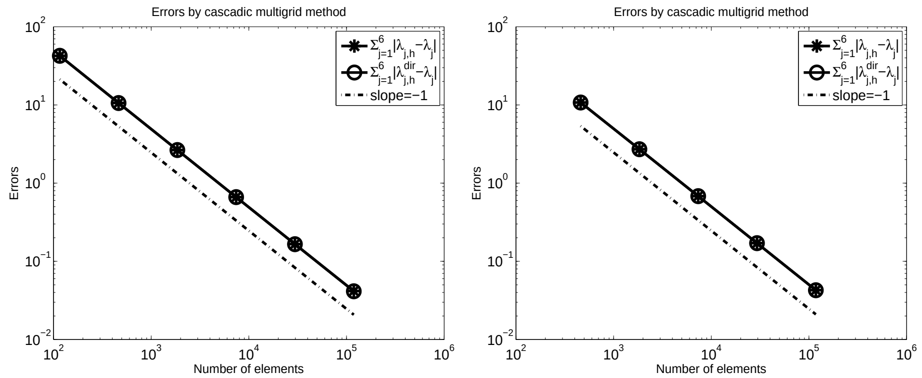

Fig. 4.3. The errors of the cascadic multigrid algorithm for the first six eigenvalues on the unit square, where $\lambda_{j,h}$ denotes the eigenvalue approximation by Algorithm 3.2, and $\lambda_{j,h}^{\rm dir}$ denotes the eigenvalue approximation by direct eigenvalue solving (The left figure corresponds to the left mesh in Figure 4.1 and the right figure corresponds to the right right mesh in Figure 4.1).

We also check the convergence behavior for multi eigenvalue approximations with Algorithm 3.2. Here the first six eigenvalues $\lambda = 2\pi^2,\ 5\pi^2,\ 5\pi^2,\ 8\pi^2,\ 10\pi^2,\ 10\pi^2$ are investigated. We also adopt the meshes shown in Figure 4.1 as the initial mesh and the corresponding numerical results are shown in Figure 4.3. Figure 4.3 also exhibits the optimal convergence and complexity of the cascadic multigrid scheme.

4.2. More general eigenvalue problem

Here we give the numerical results of the cascadic multigrid scheme for solving a more general eigenvalue problem on the unit square domain $\Omega = (0,1)\times(0,1)$: Find (λ, u) such that

$$-\nabla\cdot\mathcal{A}\nabla u + \phi u = \lambda\rho u, \quad \text{in } \Omega, \tag{4.1a}$$

$$u = 0, \quad \text{on } \partial\Omega, \tag{4.1b}$$

$$\int_\Omega \rho u^2 d\Omega = 1, \tag{4.1c}$$

where

$$\mathcal{A} = \begin{pmatrix} 1+(x_1-\frac12)^2 & (x_1-\frac12)(x_2-\frac12) \\ (x_1-\frac12)(x_2-\frac12) & 1+(x_2-\frac12)^2 \end{pmatrix},$$

$\varphi = e^{(x_1-\frac12)(x_2-\frac12)}$ and $\rho = 1+(x_1-\frac12)(x_2-\frac12)$.

In this example, we also use two coarse meshes which are shown in Figure 4.1 as the initial meshes to investigate the convergence behaviors. Since the exact solution is not known, we choose an adequately accurate eigenvalue approximations with the extrapolation method (see, e.g., [12]) as the exact eigenvalues to measure errors. Figure 4.4 gives the corresponding numerical results for the first six eigenvalue approximations. Here we also compare the numerical results with the direct algorithm. Figure 4.4 also exhibits the optimality of the error and complexity for Algorithm 3.2.

4.3. Model eigenvalue problem in three dimensional

In order to present the complexity of the proposed numerical method in this paper. Here we give the numerical results of the cascadic multigrid scheme for the Laplace eigenvalue problem

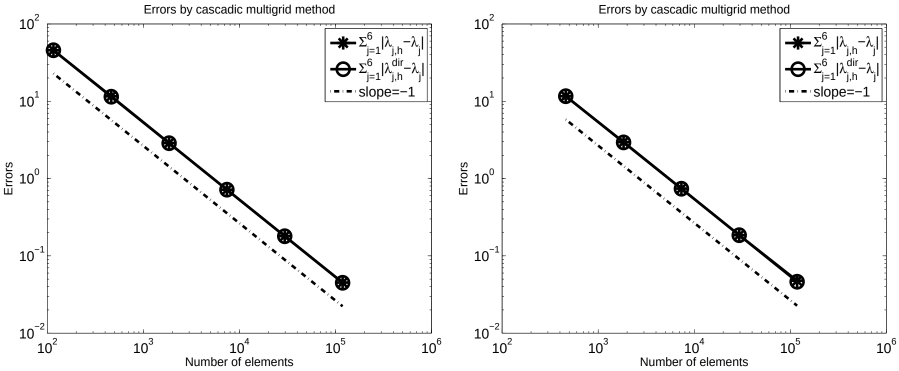

Fig. 4.4. The errors of the cascadic multigrid algorithm for the first six eigenvalues on the unit square, where $\lambda_{j,h}$ denotes the eigenvalue approximation by Algorithm 3.2, and $\lambda_{j,h}^{\rm dir}$ denotes the eigenvalue approximation by direct eigenvalue solving (The left figure corresponds to the left mesh in Figure 4.1 and the right figure corresponds to the right right mesh in Figure 4.1)

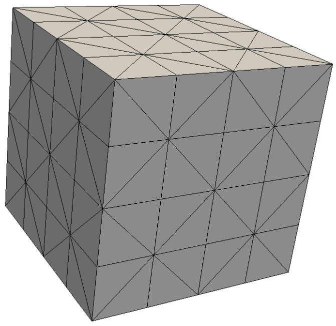

Fig. 4.5. The initial mesh for Example 3

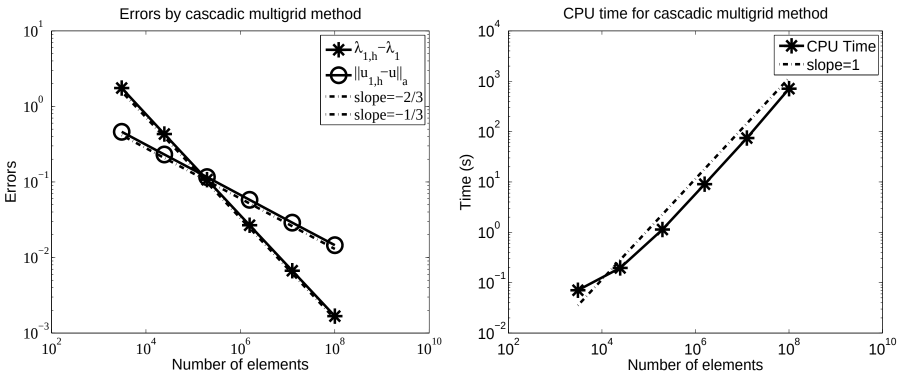

Fig. 4.6. The errors and CPU time of the cascadic multigrid algorithm for the first eigenvalue $3\pi^2$, where $\lambda_{1,h}, u_{1,h}$ denotes the eigenpair approximation obtained by Algorithm 3.2

on a three dimensional domain $\Omega = (0,1)\times(0,1)\times(0,1)$. The sequence of finite element spaces are constructed by using linear element on the series of mesh which are produced by putting some regular refinements on the following initial coarsest mesh (see Figure 4.5). Algorithm 3.2

is also applied to solve the eigenvalue problem.

The numerical results are illustrated in Figure 4.6: the left gives the error for the first eigenvalue $\lambda_1 = 3\pi^2$ and the corresponding eigenfunction, while the right gives the CPU time cost by the cascadic multigrid method (Algorithm 3.2).

From Figure 4.6, we can also find the cascadic multigrid scheme can obtain the optimal error estimates for the eigenvalue and the corresponding eigenfunction approximations. Furthermore, the CPU time results in Figure 4.6 shows the computational work of Algorithm 3.2 can really arrive the optimality.

5. Concluding Remarks

In this paper, we present a type of cascadic multigrid method for eigenvalue problems based on the combination of the cascadic multigrid for boundary value problems and the multilevel correction scheme for eigenvalue problems. The optimality of the computational efficiency has been demonstrated by theoretical analysis and numerical examples. As shown in the numerical examples, the cascadic multigrid method can also be used to obtain the multiple eigenpair approximations of the eigenvalue problem (cf. [21, 22]). Furthermore, the proposed cascadic multigrid method can be extended to more general eigenvalue problems and other types of nonlinear problems. The methods based on the multilevel correction technique only needs the same regularity of the nonlinearity as the finite element method which is different from the classical methods which depend on higher nonlinear regularity [11,14,16]. Also the extrapolation methods can be used to accelerate the smoothing steps as in [16, Section 4.7] and [5]. These will be our future work.

Acknowledgments. This work is supported in part by the National Natural Science Foundation of China (NSFC 91330202, 11371026, 11001259, 11031006, 2011CB309703) and the National Center for Mathematics and Interdisciplinary Science, CAS.

References

[1] R. Adams, *Sobolev Spaces*, Academic Press, New York, 1975.

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

[3] I. Babuška and J.E. Osborn, Eigenvalue problems. In P.G. Lions and P.G. Ciarlet, editors, *In Handbook of Numerical Analysis*, volume 2, pages 641–787. North-Holland, Amsterdam, 1991.

[4] F.A. Bornemann and P. Deuflhard, The cascadic multigrid method for elliptic problems, *Numer. Math.*, **75** (1996), 135–152.

[5] C. Chen, H. Hu, Z. Xie and C. Li, Analysis of extrapolation cascadic multigrid method (EXCMG), *Sci. China Ser. A*, **51** (2008), 1349–1360.

[6] P. Deuflhard, P. Leinen and H. Yserentant, Concepts of an adaptive hierarchical finite element code, *IMPACT Comput. Sci. Engrg.*, **1** (1989), 3–35.

[7] F. Chatelin, *Spectral Approximation of Linear Operators*, Academic Press, New York, 1983.

[8] W. Hackbusch, *Multi-grid methods and applications*, Springer-Verlag, Berlin, 1985.

[9] X. Han and H. Xie, A cascadic multigrid method for eigenvalue problem, http://arxiv.org/abs/1401.5378v1, Jan. 21, 2014.

[10] H. Chen,Y. He, Y. Li and H. Xie, A multigrid method for eigenvalue problems based on shifted-inverse power technique, *European Journal of Mathematics*, **1**:1 (2015), 207–228.

[11] Y. Huang, Z. Shi, T. Tang and W. Xue, A multilevel successive iteration method for nonlinear elliptic problem, *Math. Comp.*, **73** (2004), 525–539.

[12] Q. Lin and J. Lin, *Finite Element Methods: Accuracy and Inprovement*, Science press, Beijing, 2006.

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

[14] Q. Lin, H. Xie and F. Xu, Multilevel correction adaptive finite element method for semilinear elliptic equation, *Application of Mathematics*, **60**:5 (2015), 527–550.

[15] Q. Lin, H. Xie, and J. Xu, Lower bound of the discretization error for piecewise polynomials, *Math. Comput.*, **83**:285 (2014), 1–13.

[16] V. Shaidurov, *Multigrid Methods for Finite Elements*, Kluwer Academic, Netherlands, 1995.

[17] V. Shaidurov, Some estimates of the rate of convergence for the cascadic conjugate-gradient method, *Comput. Math. Appl.*, **31** (1996), 161–171.

[18] V. Shaidurov and L. Tobiska, The convergence of the cascadic conjugate-gradient method applied to elliptic problems in domains with re-entrant corners, *Math. Comput.*, **69** (2000), 501–520.

[19] J. Urschel, X. Hu, J. Xu and L. Zikatanov, A cascadic multigrid algorithm for computing the fiedler vector of graph laplacians, *J. Comput. Math.*, **33**:2 (2015), 209–226.

[20] L. Wang and X. Xu, *The Basic Mathematical Theory of Finite Element Methods*, Science Press (in Chinese), Beijing, 2004.

[21] H. Xie, A multigrid method for eigenvalue problem, *J. Comput. Phys.*, **274** (2014), 550–561.

[22] H. Xie, A type of multilevel method for the steklov eigenvalue problem, *IMA J. Numer. Anal.*, **34** (2014), 592–608.