Journal of Computational Physics 229 (2010) 2649–2658

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

An adaptive inverse iteration for Maxwell eigenvalue problem based on edge elements

Junqing Chen [a], Yifeng Xu [b], Jun Zou [c,*,1]

[a] Department of Mathematical Sciences, Tsinghua University, Beijing 100084, China
[b] Department of Mathematics, Shanghai Jiao Tong University, Shanghai 200240, China
[c] Department of Mathematics, The Chinese University of Hong Kong, Shatin, N.T., Hong Kong

ARTICLE INFO

Article history:
Received 19 February 2009
Received in revised form 28 November 2009
Accepted 8 December 2009
Available online 29 December 2009

Keywords:
Maxwell eigenvalue problem
Adaptive inverse iteration
Edge elements
A posterior error estimates

ABSTRACT

We propose and analyze an adaptive inverse iterative method for solving the Maxwell eigenvalue problem with discontinuous physical parameters in three dimensions. The adaptive method updates the eigenvalue and eigenfunction based on an a posteriori error estimate of the edge element discretization. At each iteration, the involved saddle-point Maxwell system is transformed into an equivalent system consisting of a singular Maxwell equation and two Poisson equations, for both of which preconditioned iterative solvers are available with optimal convergence rate in terms of the total degrees of freedom. Numerical results are presented, which confirms the quasi-optimal convergence of the adaptive edge element method in terms of the numerical accuracy and the total degrees of freedom.

1. Introduction

This work is concerned with the numerical solution of the Maxwell eigenvalue problem in computational electromagnetism. The Maxwell eigenvalue problem arises from the electromagnetic waveguides and resonances in cavities (see, e.g. [20,21]) and the governing equations can be stated as follows:

$$\nabla \times (\mu^{-1}\nabla \times \mathbf{E}) = \omega^2 \varepsilon \mathbf{E} \quad \text{in } \Omega, \tag{1.1}$$
$$\nabla \cdot (\varepsilon \mathbf{E}) = 0 \quad \text{in } \Omega, \tag{1.2}$$
$$\mathbf{n} \times \mathbf{E} = 0 \quad \text{on } \partial\Omega, \tag{1.3}$$

where Ω is a bounded domain with Lipschitz boundary $\partial\Omega$, μ and ε are the magnetic permeability and the electric permittivity respectively and ω is the resonant angular frequency of the electromagnetic wave for the cavity Ω.

Finite element methods for computing Maxwell eigenvalue problem have been widely used and their convergences have been well studied; see [4,5,22,25] and the references therein. The main focus of this work is on the practically important cases where the physical domain Ω may have reentrant corners or the coefficients μ, ε may be highly discontinuous, thus resulting in the singularities of the eigenfunction $\mathbf{E}$; see, e.g. [11,12]. The singularities affect greatly the overall accuracy of finite element solutions when regular finite element meshes are used.

Adaptive edge element methods have been proved to be very successful in resolving the local singularities of the Maxwell equations; see [1,7] for the eddy current problem, [6,10] for the time dependent and time-harmonic problems, and [8] for the

* Corresponding author. Tel.: +852 2609 7985; fax: +852 2603 5154.
E-mail addresses: jqchen@math.tsinghua.edu.cn (J. Chen), yfxuma@yahoo.com.cn (Y. Xu), zou@math.cuhk.edu.hk (J. Zou).
[1] The work of this author was substantially supported by Hong Kong RGC Grants (Projects 404407 and 404606).

0021-9991/$ - see front matter © 2010 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2009.12.013

saddle-point Maxwell system. The success and popularity of the adaptive finite element methods lie in the following two features: they are mostly based on a posteriori error estimates, which reflect the actual error between the true solution and the discrete one and are usually represented by some computable quantities such as the discrete solution and the given data of the equation; they can reach the quasi-optimality in terms of the numerical accuracy and computational complexity.

Based on an a posteriori error estimate of the edge element discretization, we will construct an adaptive inverse iteration method to solve the first eigenvalue and the corresponding eigenfunction of system (1.1)–(1.3).

The basic procedure of the adaptive method is as follows:

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

SOLVE will tackle a variational problem associated with system (1.1)–(1.3), which is constrained with a divergence law (1.2). A traditional approach is to formulate this system as a saddle-point problem [3,13] by introducing a Lagrangian multiplier to reinforce the divergence-free property and the uniqueness of the solutions. But it is still quite challenging how to effectively solve the algebraic system resulting from the finite element approximation of the saddle-point problem [18,19].

In this work, we plan to overcome this obstacle in a novel approach. At each iteration of the new method we first solve a Laplace equation to precorrect the iterate from the previous step, then solve a singular curl–curl system, which is followed by a weak divergence correction to ensure that the numerical solution is divergence-free in the discrete sense. It is worth noting that in each step we solve a singular curl–curl system once and a Laplace equation twice, for both of which there are mature preconditioned solvers with optimal preconditioners available [16,23]. In addition, we will show that the solution produced by the aforementioned procedure is nothing but the finite element solution to a desired saddle-point problem. Exploiting this equivalence, we shall derive a reliable a posteriori error estimate for ESTIMATE. With the help of the error estimate, MARK determines all refining elements for refinement. The step REFINE generates a finer triangulation by dividing those elements marked in the step MARK.

The layout of this paper is as follows. In Section 2, we present an exact inverse iterative method for the continuous eigenvalue problem and prove its convergence. A special solution process for the edge element discretization and the corresponding adaptive inverse iterative algorithm are given in Section 3. Section 4 is devoted to deriving the a posteriori error estimate for the edge element approximation of the curl–curl system in the inverse iteration. Finally, we present some numerical experiments in Section 5 to show the robustness and quasi-optimal convergence of the new adaptive algorithm in terms of the numerical accuracy and the total degrees of freedom used.

2. Inverse iterative method

In this section we introduce the inverse iterative method for computing the first eigenvalue and the corresponding eigenfunction of system (1.1)–(1.3). For this we need the following Sobolev spaces:

$$\mathbf{H}_0(\mathbf{curl};\Omega) = \{\mathbf{u} \in L^2(\Omega)^3; \nabla \times \mathbf{u} \in L^2(\Omega)^3, \mathbf{u} \times \mathbf{n} = \mathbf{0} \text{ on } \partial\Omega\},$$
$$\mathbf{X} = \{\mathbf{u} \in \mathbf{H}_0(\mathbf{curl};\Omega); \nabla \cdot (\varepsilon\mathbf{u}) = 0 \text{ in } \Omega\}.$$

Then the variational formulation of equations (1.1)–(1.3) reads as follows:

Find $\mathbf{E} \in \mathbf{X}$ and $\omega^2 > 0$ such that

$$(\mu^{-1}\nabla \times \mathbf{E}, \nabla \times \mathbf{F}) = \omega^2(\varepsilon\mathbf{E}, \mathbf{F}) \quad \forall \mathbf{F} \in \mathbf{X}. \tag{2.1}$$

The existence of eigenvalues and eigenfunctions of (2.1) is the conclusion of the Hilbert–Schmidt theory. To see this, we define an operator $K : L^2(\Omega)^3 \rightarrow \mathbf{X}$ by requiring that if $\mathbf{f} \in L^2(\Omega)^3$, then $K\mathbf{f} \in \mathbf{X}$ satisfies

$$(\mu^{-1}\nabla \times (K\mathbf{f}), \nabla \times \mathbf{v}) = (\varepsilon\mathbf{f}, \mathbf{v}) \quad \forall \mathbf{v} \in \mathbf{X}. \tag{2.2}$$

Thanks to the symmetry of the bilinear form involved in (2.2), the compactness result by Weber [31] and the Lax–Milgram lemma, K is self-adjoint and compact as an operator: $L^2(\Omega)^3 \rightarrow L^2(\Omega)^3$ (see [15,24]). Thus, K converts the eigenvalue problem (2.1) to an operator eigenvalue problem

$$K\mathbf{E} = \nu\mathbf{E}, \tag{2.3}$$

with $\nu = \omega^{-2}$. Thus by virtue of Hilbert–Schmidt theory we have [15,24]:

Theorem 2.1. *There exist an infinite discrete set of eigenvalues $\nu_k > 0, k = 1, 2, \ldots$ and the corresponding eigenfunctions $\mathbf{E}_k \in \mathbf{X}, \mathbf{E}_k \neq \mathbf{0}$ such that*

1. *Eq. (2.3) holds for each pair $(\nu_k, \mathbf{E}_k)$;*
2. *$\nu_1 \geqslant \nu_2 \geqslant \cdots > 0$, and $\lim_{k\rightarrow\infty}\nu_k = 0$;*
3. *$\{\mathbf{E}_k\}_{k=1}^{\infty}$ are orthonormal in inner product $(\varepsilon\cdot,\cdot)$;*
4. *$\mathbf{X} = \text{span}\{\mathbf{E}_1, \mathbf{E}_2, \ldots\}$ and $\ker(K) = \nabla H_0^1(\Omega)$.*

Now by applying the inverse iterative algorithm (cf. [28]) to the Maxwell eigenvalue problem (2.1), we can formulate

Algorithm 2.1 (*Exact inverse iteration algorithm*). Given a tolerance $tol > 0$ and set $j := 0$;

1. Given $\mathbf{u}_0$ such that $(\varepsilon\mathbf{u}_0, \mathbf{u}_0) = 1$ and $\nabla \cdot (\varepsilon\mathbf{u}_0) = 0$; Set $j := 1$.
2. Find $\hat{\mathbf{u}}_j \in \mathbf{X}$ such that
$$(\mu^{-1}\nabla \times \hat{\mathbf{u}}_j, \nabla \times \mathbf{v}) = (\varepsilon\mathbf{u}_{j-1}, \mathbf{v}) \quad \forall \mathbf{v} \in \mathbf{X}. \tag{2.4}$$
3. Compute $\lambda_j = \frac{(\mu^{-1}\nabla\times\hat{\mathbf{u}}_j, \nabla\times\hat{\mathbf{u}}_j)}{(\varepsilon\hat{\mathbf{u}}_j, \hat{\mathbf{u}}_j)}$, set $\mathbf{u}_j = \frac{\hat{\mathbf{u}}_j}{(\varepsilon\hat{\mathbf{u}}_j, \hat{\mathbf{u}}_j)^{1/2}}$.
4. If $\left|\frac{\lambda_j - \lambda_{j-1}}{\lambda_j}\right| < tol$, output λ_j and $\mathbf{E}_j$; else set $j := j + 1$ and go to Step 2.

We have the following convergence for Algorithm 2.1, whose proof is standard (cf. [9]).

Theorem 2.2. *Suppose $\{(\omega_k^2, \mathbf{E}_k)\}_{k=1}^{\infty}$ are the eigenvalues and eigenfunctions of Problem (2.1) such that the first eigenvalue ω_1^2 is simple and $\omega_1^2 < \omega_2^2 \leqslant \omega_3^2 \leqslant \cdots$ If the initial guess $\mathbf{u}_0$ satisfies that $(\varepsilon\mathbf{u}_0, \mathbf{E}_1) \neq 0$, then the sequences $\{\mathbf{u}_j\}_{j=1}^{\infty}$ and $\{\lambda_j\}_{j=1}^{\infty}$ generated by Algorithm 2.1 converge in the followings sense*

$$\|\varepsilon^{1/2}(\mathbf{u}_j - \sigma\mathbf{E}_1)\|_0 \to 0 \quad \text{and} \quad \lambda_j \to \omega_1^2 \quad \text{as } j \to \infty, \tag{2.5}$$

where σ is a constant given by

$$\sigma = 1 \quad \text{if } (\varepsilon\mathbf{u}_0, \mathbf{E}_1) > 0; \quad \sigma = -1 \quad \text{if } (\varepsilon\mathbf{u}_0, \mathbf{E}_1) < 0.$$

3. Edge element based adaptive inverse iterative method

In this section, we propose an adaptive inverse iterative method based on Algorithm 2.1 and the edge element discretization. First, one may realize a major difficulty in the approximation of the Sobolev space $\mathbf{X}$ used in the variational equation (2.4): there are no conveniently implementable finite element spaces with reasonable degrees of freedom in three dimensions available in the literature that may reinforce the divergence constraint in space $\mathbf{X}$.

A natural approach to reinforce the divergence constraint condition in $\mathbf{X}$ is to transform the variational equation (2.4) into the following equivalent saddle-point formulation: Find $(\hat{\mathbf{u}}_j, \hat{p}_j) \in \mathbf{H}_0(\mathbf{curl}; \Omega) \times H_0^1(\Omega)$ such that

$$\begin{cases} (\mu^{-1}\nabla \times \hat{\mathbf{u}}_j, \nabla \times \mathbf{v}) + (\varepsilon\mathbf{v}, \nabla\hat{p}_j) = (\varepsilon\mathbf{u}_{j-1}, \mathbf{v}) \quad \forall \mathbf{v} \in \mathbf{H}_0(\mathbf{curl}; \Omega), \\ (\varepsilon\hat{\mathbf{u}}_j, \nabla q) = 0 \quad \forall q \in H_0^1(\Omega). \end{cases} \tag{3.1}$$

This saddle-point system is uniquely solvable; see, e.g. [8].

Now we are going to discuss the discretization of the saddle-point system (3.1) by the edge element method. Let $\{\mathcal{T}_j\}_{j\geqslant 0}$ be a nested sequence of shape regular tetrahedral triangulations over Ω such that ε and μ are constants on each $K \in \mathcal{T}_j$, and $\mathcal{F}_j$ be the collection of all interior faces in $\mathcal{T}_j$. As it will be seen, the nested meshes $\{\mathcal{T}_j\}_{j\geqslant 0}$ will be recursively obtained by adaptive refinements of the initial mesh $\mathcal{T}_0$ based on an a posteriori error estimate.

On each mesh $\mathcal{T}_j$, we introduce the lowest order edge element space of the first family (cf. [26]):

$$\mathbb{V}_j^h = \{\mathbf{v} \in \mathbf{H}_0(\mathbf{curl}; \Omega); \mathbf{v}|_K = \mathbf{a}_K + \mathbf{b}_K \times \mathbf{x}, \ \forall K \in \mathcal{T}_j\}.$$

It is known that each function in $\mathbb{V}_j^h$ is uniquely determined by the degrees of freedom $\{\int_e \mathbf{v} \cdot d\mathbf{l}, e \in \mathcal{E}_j\}$, where $\mathcal{E}_j$ is the set consisting of all interior edges of $\mathcal{T}_j$. To deal with the divergence constraint we need to introduce a Lagrange multiplier space $\mathbb{S}_j^h$, which is taken to be the standard piecewise linear finite element subspace of $H_0^1(\Omega)$ over mesh $\mathcal{T}_j$.

With the above preparations, we are ready to approximate the saddle-point problem (3.1) by the edge element method.

Let $\mathbf{U}_{j-1} \in \mathbb{V}_{j-1}^h$ be an approximation of the iterate $\mathbf{u}_{j-1}$ in (3.1). Then we suggest to approximate (3.1) by the following discrete problem:

Find $(\widetilde{\mathbf{U}}_j, \widetilde{P}_j) \in \mathbb{V}_j^h \times \mathbb{S}_j^h$ such that

$$\begin{cases} (\mu^{-1}\nabla \times \widetilde{\mathbf{U}}_j, \nabla \times \mathbf{V}) + (\varepsilon\mathbf{V}, \nabla\widetilde{P}_j) = (\varepsilon\mathbf{U}_{j-1}, \mathbf{V}) \quad \forall \mathbf{V} \in \mathbb{V}_j^h, \\ (\varepsilon\widetilde{\mathbf{U}}_j, \nabla Q) = 0 \quad \forall Q \in \mathbb{S}_j^h. \end{cases} \tag{3.2}$$

For the convenience of the subsequent analysis, we introduce also the auxiliary continuous counterpart of the discrete system (3.2):

Given $\mathbf{U}_{j-1} \in \mathbb{V}_{j-1}^h$ for $j \geqslant 1$, find $(\hat{\mathbf{u}}_j^*, \hat{p}_j^*) \in \mathbf{H}_0(\mathbf{curl}; \Omega) \times H_0^1(\Omega)$ such that

$$\begin{cases} (\mu^{-1}\nabla \times \hat{\mathbf{u}}_j^*, \nabla \times \mathbf{v}) + (\varepsilon\mathbf{v}, \nabla\hat{p}_j^*) = (\varepsilon\mathbf{U}_{j-1}, \mathbf{v}) \quad \forall \mathbf{v} \in \mathbf{H}_0(\mathbf{curl}; \Omega), \\ \left(\varepsilon\hat{\mathbf{u}}_j^*, \nabla q\right) = 0 \quad \forall q \in H_0^1(\Omega). \end{cases} \tag{3.3}$$

The discrete saddle-point system (3.2) is a natural choice for the approximation of the system (2.4) in Algorithm 2.1. Unfortunately there are still no fast solvers of preconditioning type available in the literature for the discrete system (3.2). Some

efficient iterative preconditioned solvers with relaxation parameters were proposed in [17] and [19] for the discrete saddle-point systems similar to (3.1). However the efficiency of those algorithms depends strongly on the availability of efficient preconditioners for the stiffness matrix associated with the symmetric bilinear form in the first equation of (3.2) and the Schur complement matrix associated with (3.2). It is still open and challenging how to construct those efficient preconditioners, especially for the cases considered in this work that the physical parameters μ and ε in (3.2) may have large jumps across the interfaces of different media. Another difficult issue one needs to take care of in designing an efficient solver for the discrete system (3.2) lies in the fact that the stiffness matrix associated with the symmetric bilinear form in the first equation of (3.2) is singular, while most existing preconditioned solvers work only for nonsingular stiffness matrices; see [17,19] and the references therein.

To overcome the aforementioned two difficulties in solving the saddle-point system (3.1), we are now going to propose an equivalent implementation of (3.2), for which fast preconditioned solvers are available. This will become an essential step in the adaptive inverse iterative algorithm to be proposed at the end of this section.

3.1. Splitting procedure

Given $j \geqslant 1$ and some $\mathbf{U}_{j-1} \in \mathbb{V}_{j-1}^h$ for approximating the iterate $\mathbf{u}_{j-1}$ in (3.1).

1. Find $P_j \in \mathbb{S}_j^h$ such that
$$(\varepsilon \nabla P_j, \nabla Q) = (\varepsilon \mathbf{U}_{j-1}, \nabla Q) \quad \forall Q \in \mathbb{S}_j^h; \tag{3.4}$$
2. Find $\widehat{\mathbf{U}}_j^* \in \mathbb{V}_j^h$ such that
$$(\mu^{-1} \nabla \times \widehat{\mathbf{U}}_j^*, \nabla \times \mathbf{V}) = (\varepsilon(\mathbf{U}_{j-1} - \nabla P_j), \mathbf{V}) \quad \forall \mathbf{V} \in \mathbb{V}_j^h; \tag{3.5}$$
3. Find $\Phi_j \in \mathbb{S}_j^h$ such that
$$(\varepsilon \nabla \Phi_j, \nabla Q) = (\varepsilon \widehat{\mathbf{U}}_j^*, \nabla Q) \quad \forall Q \in \mathbb{S}_j^h, \tag{3.6}$$
and set $\widehat{\mathbf{U}}_j := \widehat{\mathbf{U}}_j^* - \nabla \Phi_j$.

As we will describe at the end of this section, each step of the Splitting Procedure above can be realized by a PCG solver with optimal convergence independent of the finite element mesh size.

Now we demonstrate that the Splitting Procedure is equivalent to the discrete saddle-point problem (3.2).

Lemma 3.1. *The discrete saddle-point problem (3.2) has a unique solution $(\widetilde{\mathbf{U}}_j, \widetilde{P}_j) \in \mathbb{V}_j^h \times \mathbb{S}_j^h$, and $(\widetilde{\mathbf{U}}_j, \widetilde{P}_j)$ is the same as $(\widehat{\mathbf{U}}_j, P_j)$ produced by the Splitting Procedure.*

Proof. We begin with the existence of a unique solution to (3.2). As the spaces concerned are all finite dimensional, it suffices to prove that (3.2) has a trivial solution $(\widetilde{\mathbf{U}}_j, \widetilde{P}_j) = (0, 0)$ for a vanishing right-hand side, namely $\mathbf{U}_{j-1} = 0$. To see this, taking $\mathbf{V} = \widetilde{\mathbf{U}}_j$ in the first equation of (3.2) and noting the second equation, we see $(\mu^{-1} \nabla \times \widetilde{\mathbf{U}}_j, \nabla \times \widetilde{\mathbf{U}}_j) = 0$, hence $\nabla \times \widetilde{\mathbf{U}}_j = \mathbf{0}$. Then by means of the discrete potential representation [16,24], there exists a $\widetilde{\Phi} \in \mathbb{S}_j^h$ such that $\widetilde{\mathbf{U}}_j = \nabla \widetilde{\Phi}$. Substituting it into the second equation of (3.2) yields $\widetilde{\Phi} = 0$, thus we have $\widetilde{\mathbf{U}}_j = \mathbf{0}$. The result $\widetilde{P}_j = 0$ follows by taking $\mathbf{V} = \nabla \widetilde{P}_j$ in the first equation of (3.2).

Next we show $(\widetilde{\mathbf{U}}_j, \widetilde{P}_j)$ is the same as $(\widehat{\mathbf{U}}_j, P_j)$ produced by the Splitting Procedure. Thanks to the unique solvability of the problem (3.2), it suffices to verify that $(\widehat{\mathbf{U}}_j, P_j)$ is uniquely defined and satisfies the discrete saddle-point problem (3.2). Noting that the space $\nabla \mathbb{S}_j^h$ is embedded in $\mathbb{V}_j^h$ [15], we can take $\mathbf{V} = \nabla Q$ with $Q \in \mathbb{S}_j^h$ in the first equation of (3.2) to see that $\widetilde{P}_j$ and P_j both solve the same equation (3.4), which is clearly uniquely solvable and so $P_j = \widetilde{P}_j$. Now it follows from (3.4) that $\mathbf{U}_{j-1} - \nabla P_j$ is orthogonal to the space $\nabla \mathbb{S}_j^h$ in $\mathbb{V}_j^h$ with respect to the inner product $(\varepsilon \cdot, \cdot)$. This orthogonality, along with the fact that $\nabla \mathbb{S}_j^h$ is the kernel of $\nabla \times$, implies that (3.5) is trivially satisfied for the subspace $\nabla \mathbb{S}_j^h$ of $\mathbb{V}_j^h$. Therefore, we may restrict (3.5) to $\mathbb{X}_j^h$, the orthogonal complement of $\nabla \mathbb{S}_j^h$ in $\mathbb{V}_j^h$ with respect to the inner product $(\varepsilon \cdot, \cdot)$ and show (3.5) is uniquely solvable in $\mathbb{X}_j^h$. This is equivalent to proving $\widehat{\mathbf{U}}_j^* = \mathbf{0}$ in $\mathbb{X}_j^h$ for a vanishing right-hand side in (3.5). Clearly in this case we know from (3.5) that $\nabla \times \widehat{\mathbf{U}}_j^* = \mathbf{0}$. Then by virtue of the discrete potential representation [16,24] again, there exists a $\widehat{\Phi} \in \mathbb{S}_j^h$ such that $\widehat{\mathbf{U}}_j^* = \nabla \widehat{\Phi}$. Since (3.5) is now restricted in $\mathbb{X}_j^h$, the orthogonal complement of $\nabla \mathbb{S}_j^h$ in $\mathbb{V}_j^h$, it can only happen that $\widehat{\mathbf{U}}_j^* = \mathbf{0}$. This proves the solvability of (3.5) and that $\widehat{\mathbf{U}}_j^*$ is unique in $\mathbb{X}_j^h$.

Now filtering all the solutions $\widehat{\mathbf{U}}_j^*$ to (3.5) by (3.6), we obtain a unique output $(\widehat{\mathbf{U}}_j, P_j)$ generated by the Splitting Procedure. In fact, we know from (3.5) and (3.6) that $(\widehat{\mathbf{U}}_j, P_j)$ satisfies

$$(\mu^{-1} \nabla \times \widehat{\mathbf{U}}_j, \nabla \times \mathbf{V}) = \left(\mu^{-1} \nabla \times \widehat{\mathbf{U}}_j^*, \nabla \times \mathbf{V}\right) = (\varepsilon(\mathbf{U}_{j-1} - \nabla P_j), \mathbf{V}) \quad \forall \mathbf{V} \in \mathbb{V}_j^h$$

and

$$(\varepsilon \widehat{\mathbf{U}}_j, \nabla Q) = (\varepsilon \widehat{\mathbf{U}}_j^*, \nabla Q) - (\varepsilon \nabla \Phi_j, \nabla Q) = 0 \quad \forall Q \in \mathbb{S}_j^h.$$

So $(\widehat{\mathbf{U}}_j, P_j)$ is indeed a solution to the discrete saddle-point problem (3.2), or equal to the unique solution $(\widetilde{\mathbf{U}}_j, \widetilde{P}_j)$. □

We see from the equivalence in Lemma 3.1 that the Splitting Procedure is actually an approximation of the continuous problem (3.3), which is the perturbed version of (3.1), i.e., the equivalent saddle-point formulation of the original problem (2.4). But from the numerical implementation point of view, the Splitting Procedure provides a quite different and much more efficient solver to the system (2.4), compared to the solver (3.2).

The equivalence in Lemma 3.1 inspires a central idea of this work: one may use the Splitting Procedure to generate the approximate solution $(\widehat{\mathbf{U}}_j, P_j)$ to the continuous system (2.4), while use the edge element system (3.2) to derive the desired a posteriori error estimate for its discrete solution $(\widetilde{\mathbf{U}}_j, \widetilde{P}_j)$.

With the above preparations, we are now ready to formulate our major adaptive inverse iterative algorithm based on Algorithm 2.1 and an a posteriori error estimate of the edge element system (3.2). The derivations of the a posteriori error estimate and the resulting local a posteriori error indicators will be carried out in the next section; see Theorem 4.1. But for the formulation of the adaptive inverse iterative algorithm, we need to first introduce these local error indicators over each element $K \in \mathcal{T}_j$:

$$\eta_{1,K}^2 = h_K^2 \|\varepsilon(\mathbf{U}_{j-1} - \nabla P_j)\|_{L^2(K)}^2 + \sum_{F \subset \partial K \cap \Omega} h_F (\|[\![\mathbf{n} \times \mu^{-1} \nabla \times \widehat{\mathbf{U}}_j]\!]\|_{L^2(F)}^2 + \|[\![\varepsilon(\mathbf{U}_{j-1} - \nabla P_j) \cdot \mathbf{n}]\!]\|_{L^2(F)}^2),$$

$$\eta_{2,K}^2 = \sum_{F \subset \partial K \cap \Omega} h_F \|[\![\mathbf{n} \cdot \varepsilon \widehat{\mathbf{U}}_j]\!]\|_{L^2(F)}^2$$

and

$$\eta_K = \left(\eta_{1,K}^2 + \eta_{2,K}^2\right)^{\frac{1}{2}}, \quad \eta_{\max} = \max_{K \in \mathcal{T}_j} \eta_K,$$

where $[\![\cdot]\!]$ represents the jump across each face F, and h_K and h_F are the diameters of element K and face F respectively.

With these notations, we can now formulate our adaptive inverse iterative algorithm for the Maxwell's eigenvalue problem (1.1)–(1.3).

Algorithm 3.1 (*Adaptive inverse iterative algorithm*). Choose $\lambda_0 > 0, 0 < \theta < 1$, tolerance *tol* and an initial mesh $\mathcal{T}_0$; set $j := 0$.

1. Choose a $\mathbf{U}_0 \in \mathbb{V}_0^h$ such that $(\varepsilon \mathbf{U}_0, \mathbf{U}_0) = 1$, and set $\mathcal{T}_1 := \mathcal{T}_0$;
2. Set $j := j + 1$;
3. Solve (3.2) on mesh $\mathcal{T}_j$ by the Splitting Procedure to get $(\widehat{\mathbf{U}}_j, P_j)$;
4. Compute $\lambda_j = \frac{(\mu^{-1}\nabla\times\widehat{\mathbf{U}}_j, \nabla\times\widehat{\mathbf{U}}_j)}{(\varepsilon\widehat{\mathbf{U}}_j, \widehat{\mathbf{U}}_j)}$, $\mathbf{U}_j = \frac{\widehat{\mathbf{U}}_j}{(\varepsilon\widehat{\mathbf{U}}_j, \widehat{\mathbf{U}}_j)^{1/2}}$;
5. If $\left|\frac{\lambda_j - \lambda_{j-1}}{\lambda_j}\right| < tol$, output λ_j and $\mathbf{U}_j$; else, do the following:
   (a) Compute the local a posteriori error indicator η_K on each $K \in \mathcal{T}_j$;
   (b) Select a subset $\mathcal{M}_j \subset \mathcal{T}_j$ such that
   $$\eta_K \geqslant \theta \eta_{\max} \quad \forall K \in \mathcal{M}_j;$$
   (c) Refine each element in $\mathcal{M}_j$ and some additional elements in $\mathcal{T}_j \setminus \mathcal{M}_j$ for conformity to obtain $\mathcal{T}_{j+1}$, go to Step 2.

In the remainder of this section, we will describe briefly some existing fast solvers for each equation involved in the Splitting Procedure. First for the two Poisson systems in (3.4) and (3.6), fast solvers have been well developed which have optimal convergence in terms of the total degrees of freedom used, such as (algebraic) multigrid or nonoverlapping domain decomposition preconditioned conjugate gradient (PCG) method; see [32]. Thanks to the nodal auxiliary space preconditioner [16] and the two-level implementation for a singular system in PHG [27] and Hypre [23], it is also possible for us to solve (3.5) by a PCG method, which converges with optimal rate independent of the finite element mesh sizes. Next we describe briefly such an optimal preconditioner. By $\mathbf{A}_h$ we denote the stiffness matrix associated with the bilinear form in (3.5) and $\mathbf{L}_h$ the stiffness matrix associated with the bilinear form in (3.4) or (3.6), then an optimal multilevel preconditioner of the form

$$\mathbf{\Lambda}_h^{-1} + \mathbf{P}_h \mathbf{B}_h^{-1} \mathbf{P}_h^T$$

was proposed in [16] (see also [23]), where $\mathbf{\Lambda}_h$ stands for the Gauss–Seidel Smoother for $\mathbf{A}_h$, $\mathbf{B}_h$ represents the (algebraic) multigrid preconditioner for $\mathbf{L}_h$, and $\mathbf{P}_h$ represents the interpolant from vector linear Lagrange element space $(\mathbb{S}_j^h)^3$ to the Nédélec element space $\mathbb{V}_j^h$.

4. The a posteriori error analysis

In this section we establish the a posteriori error estimate for the solution to the edge element system (3.2), which provides the essential computable local quantities in each element required in Algorithm 3.1 (see Step 5) to determine which elements at the current level to be refined. Unless specified otherwise, C will always denote a generic constant independent of the functions under consideration and may be different at each occurrence.

Let $\hat{\mathbf{u}}_j^*$ and $\widehat{\mathbf{U}}_j$ be respectively the exact solution of (3.3) and the edge element solution achieved by the Splitting Procedure, then our task is to bound the energy norm of the error $\hat{\mathbf{u}}_j^* - \widehat{\mathbf{U}}_j$ in terms of some computable quantities (called a posteriori

error indicators) from the computed edge element solution and the given data of the Maxwell system. Due to the equivalence in Lemma 3.1, we shall also write $(\widehat{\mathbf{U}}_j, P_j)$ for the solution $(\widetilde{\mathbf{U}}_j, \widetilde{P}_j)$ to (3.2).

For the a posteriori error estimate, we will make use of two discrete operators: the first one is the Scott–Zhang operator $I_j^s : H_0^1(\Omega) \to \mathbb{S}_j^h$ [29] and the second is the Beck–Hiptmair–Hoppe–Wohlmuth operator $\Pi_j : H^1(\Omega)^3 \cap \mathbf{H}_0(\mathbf{curl};\Omega) \to \mathbb{V}_j^h$ [1]. For any $K \in \mathcal{T}_j, F \in \mathcal{F}_j$, and any $\phi \in H_0^1(\Omega), \phi_h \in \mathbb{S}_j^h, I_j^s$ satisfies the following stable and approximation properties:

$$\begin{aligned} &I_j^s \phi_h = \phi_h, \quad \|\nabla I_j^s \phi\|_{L^2(K)} \leqslant \|\nabla \phi\|_{L^2(D_K)}, \\ &\|\phi - I_j^s \phi\|_{L^2(K)} \leqslant Ch_K \|\nabla \phi\|_{L^2(D_K)}, \quad \|\phi - I_j^s \phi\|_{L^2(F)} \leqslant Ch_F^{1/2} \|\nabla \phi\|_{L^2(D_F)}, \end{aligned} \tag{4.1}$$

while for any $\mathbf{w}_h \in \mathbb{V}_j^h, \mathbf{w} \in H^1(\Omega)^3 \cap \mathbf{H}_0(\mathbf{curl};\Omega), \Pi_j$ admits the following stable and approximation properties:

$$\begin{aligned} &\Pi_j \mathbf{w}_h = \mathbf{w}_h, \quad \|\Pi_j \mathbf{w}\|_{H(\mathbf{curl};K)} \leqslant \|\mathbf{w}\|_{H^1(D_K)}, \\ &\|\mathbf{w} - \Pi_j \mathbf{w}\|_{L^2(K)} \leqslant Ch_K \|\mathbf{w}\|_{H^1(D_K)}, \quad \|\mathbf{w} - \Pi_j \mathbf{w}\|_{L^2(F)} \leqslant Ch_F^{1/2} \|\mathbf{w}\|_{H^1(D_F)}, \end{aligned} \tag{4.2}$$

where D_K and D_F is the union of elements in $\mathcal{T}_j$, which have a non-empty intersection with K and F respectively.

The next lemma is important to the a posteriori error analysis for the edge element approximation of (3.3).

Lemma 4.1. *For $j \geqslant 1$, suppose that $\widehat{\mathbf{U}}_j$ is the solution produced by the Splitting Procedure at the jth iteration of Algorithm 3.1 and $\hat{\mathbf{u}}_j^*$ is the solution of the corresponding continuous problem (3.3). Then there exists a constant C depending only on the minimum angle of $\mathcal{T}_j$ and the coefficient ε in (1.1) such that*

$$\|\hat{\mathbf{u}}_j^* - \widehat{\mathbf{U}}_j\|_{H(\mathbf{curl};\Omega)} \leqslant C\|\nabla \times (\hat{\mathbf{u}}_j^* - \widehat{\mathbf{U}}_j)\|_{L^2(\Omega)} + C\left(\sum_{F \in \mathcal{F}_j} h_F \|[\![\varepsilon \widehat{\mathbf{U}}_j \cdot \mathbf{n}]\!]\|_{L^2(F)}^2\right)^{1/2}. \tag{4.3}$$

Proof. Let $\mathbf{e} = \hat{\mathbf{u}}_j^* - \widehat{\mathbf{U}}_j$, and $\phi \in H_0^1(\Omega)$ be the solution of the problem

$$(\varepsilon \nabla \phi, \nabla q) = (\varepsilon \widehat{\mathbf{U}}_j, \nabla q) \quad \forall q \in H_0^1(\Omega). \tag{4.4}$$

Noting $\hat{\mathbf{u}}_j^* - \widehat{\mathbf{U}}_j - \nabla \phi \in \mathbf{X}$, we have

$$\|\mathbf{e} - \nabla \phi\|_{H(\mathbf{curl};\Omega)} \leqslant C\|\nabla \times \mathbf{e}\|_{L^2(\Omega)},$$

which implies

$$\|\mathbf{e}\|_{H(\mathbf{curl};\Omega)} \leqslant C\|\nabla \times \mathbf{e}\|_{L^2(\Omega)} + \|\nabla \phi\|_{L^2(\Omega)}. \tag{4.5}$$

By (4.4), the equivalence in Lemma 3.1 and the second equation of (3.2), it is easy to see that

$$\left\|\varepsilon^{\frac{1}{2}} \nabla \phi\right\|_{L^2(\Omega)}^2 = \left(\varepsilon \widehat{\mathbf{U}}_j, \nabla\left(\phi - I_j^s \phi\right)\right).$$

Using the standard arguments in the a posteriori error analysis for second-order elliptic equations (see, e.g., [30]), we know

$$\|\nabla \phi\|_{L^2(\Omega)} \leqslant C\left(\sum_{F \in \mathcal{F}_j} h_F \|[\![\varepsilon \widehat{\mathbf{U}}_j \cdot \mathbf{n}]\!]\|_{L^2(F)}^2\right)^{1/2}. \tag{4.6}$$

Now (4.3) follows from (4.5) and (4.6). □

Next we derive the main a posteriori error estimate of this section.

Theorem 4.1. *For $j \geqslant 1$, let $\left(\hat{\mathbf{u}}_j^*, \hat{p}_j^*\right)$ be the solution of (3.3), and $(\widehat{\mathbf{U}}_j, P_j)$ the solution of (3.2) in the step 3 of the jth iteration of Algorithm 3.1, then we have the following a posteriori error estimate:*

$$\left\|\hat{\mathbf{u}}_j^* - \widehat{\mathbf{U}}_j\right\|_{H(\mathbf{curl};\Omega)} \leqslant C\left\{\sum_{K \in \mathcal{T}_j} (\eta_{1,K}^2 + \eta_{2,K}^2)\right\}^{1/2}, \tag{4.7}$$

where C depends only on the shape regularity of the mesh and the coefficients ε and μ.

Proof. For $\mathbf{e} = \hat{\mathbf{u}}_j^* - \widehat{\mathbf{U}}_j$, by the regular decomposition [2,10] there exists a vector-valued function $\mathbf{w} \in H^1(\Omega)^3 \cap H_0(\mathbf{curl};\Omega)$ and a scalar function $\psi \in H_0^1(\Omega)$ such that

$$\mathbf{e} = \mathbf{w} + \nabla \psi, \tag{4.8}$$

with the estimate

$$\|\mathbf{w}\|_{H^1(\Omega)} \leqslant C\|\mathbf{e}\|_{H(\mathbf{curl};\Omega)}, \quad \|\psi\|_{H^1(\Omega)} \leqslant C\|\mathbf{e}\|_{H(\mathbf{curl};\Omega)}. \tag{4.9}$$

Invoking the interpolation operators Π_j and I_j^s, we will write $\mathbf{e}_h \in \mathbb{V}_j^h$ for $\Pi_j \mathbf{w} + \nabla(I_j^s \psi)$. Then noting that $(\hat{\mathbf{u}}_j^*, \hat{p}_j^*)$ and $(\widehat{\mathbf{U}}_j, \widehat{P}_j) = (\widehat{\mathbf{U}}_j, P_j)$ satisfy (3.3) and (3.2) respectively, we derive

$$\left(\mu^{-1}\nabla \times \hat{\mathbf{u}}_j^*, \nabla \times \mathbf{e}\right) = \left(\varepsilon(\mathbf{U}_{j-1} - \nabla \hat{p}_j^*), \mathbf{e}\right), \tag{4.10}$$

$$(\mu^{-1}\nabla \times \widehat{\mathbf{U}}_j, \nabla \times \mathbf{e}_h) = (\varepsilon(\mathbf{U}_{j-1} - \nabla P_j), \mathbf{e}_h). \tag{4.11}$$

On the other hand, for any $q \in H_0^1(\Omega)$ and $Q \in \mathbb{S}_j^h$, we take $\mathbf{v} = \nabla q$ and $\mathbf{V} = \nabla Q$ in the first equation of (3.3) and (3.2) respectively to obtain

$$(\varepsilon \nabla \hat{p}_j^*, \nabla q) = (\varepsilon \mathbf{U}_{j-1}, \nabla q), \quad (\varepsilon \nabla P_j, \nabla Q) = (\varepsilon \mathbf{U}_{j-1}, \nabla Q).$$

By standard a posteriori error analysis for second-order elliptic problems [30], we know

$$\left\|\nabla\left(\hat{p}_j^* - P_j\right)\right\|_{L^2(\Omega)} \leqslant C\left(\sum_{F \in \mathcal{F}_j} h_F \|[\![\varepsilon(\mathbf{U}_{j-1} - \nabla P_j) \cdot \mathbf{n}]\!]\|_{L^2(F)}^2\right)^{1/2}. \tag{4.12}$$

Now we can deduce the following equivalence by using (4.10) and (4.11):

$$\begin{aligned}
(\mu^{-1}\nabla \times \mathbf{e}, \nabla \times \mathbf{e}) &= \left(\mu^{-1}\nabla \times \hat{\mathbf{u}}_j^*, \nabla \times \mathbf{e}\right) - (\mu^{-1}\nabla \times \widehat{\mathbf{U}}_j, \nabla \times \mathbf{e}) \\
&= \left(\varepsilon\left(\mathbf{U}_{j-1} - \nabla \hat{p}_j^*\right), \mathbf{e}\right) - (\mu^{-1}\nabla \times \widehat{\mathbf{U}}_j, \nabla \times (\mathbf{e} - \mathbf{e}_h)) - (\mu^{-1}\nabla \times \widehat{\mathbf{U}}_j, \nabla \times \mathbf{e}_h) \\
&= \left(\varepsilon\left(\mathbf{U}_{j-1} - \nabla \hat{p}_j^*\right), \mathbf{e}\right) - (\mu^{-1}\nabla \times \widehat{\mathbf{U}}_j, \nabla \times (\mathbf{e} - \mathbf{e}_h)) - (\varepsilon(\mathbf{U}_{j-1} - \nabla P_j), \mathbf{e}_h) \\
&= (\varepsilon(\mathbf{U}_{j-1} - \nabla P_j), \mathbf{e} - \mathbf{e}_h) - (\mu^{-1}\nabla \times \widehat{\mathbf{U}}_j, \nabla \times (\mathbf{e} - \mathbf{e}_h)) - (\varepsilon \nabla(\hat{p}_j^* - P_j), \mathbf{e}) \\
&= (\varepsilon(\mathbf{U}_{j-1} - \nabla P_j), \mathbf{w} - \Pi_j \mathbf{w}) + \left(\varepsilon(\mathbf{U}_{j-1} - \nabla P_j), \nabla\left(\psi - I_j^s \psi\right)\right) \\
&\quad - (\mu^{-1}\nabla \times \widehat{\mathbf{U}}_j, \nabla \times (\mathbf{w} - \Pi_j \mathbf{w})) - \left(\varepsilon \nabla\left(\hat{p}_j^* - P_j\right), \mathbf{e}\right).
\end{aligned}$$

Then by using the integration by parts, the approximation properties (4.1) and (4.2) of interpolation operators Π_j and I_j^s, the estimates (4.9) and (4.12), we come to

$$\|\nabla \times \mathbf{e}\|_{L^2(\Omega)}^2 \leqslant C\left(\sum_{K \in \mathcal{T}_j} \eta_{1,K}^2\right)^{1/2} \|\mathbf{e}\|_{H(\mathbf{curl};\Omega)},$$

which, along with (4.3) and Young's inequality leads to the desired estimate (4.7). □

The local error indicators $\eta_{1,K}$ and $\eta_{2,K}$ over each element K derived in Theorem 4.1 are the crucial ingredients in our adaptive Algorithm 3.1. Although these a posteriori error indicators do not directly measure the error between the discrete eigenfunction and the exact one, they provide an indirect manner to refine the mesh and proceed to the next iteration. This is one of the central novelties in this work. As we will see from the numerical experiments, the adaptive Algorithm 3.1 based on this a posteriori error estimate performs very robustly and satisfactorily, in fact it converges quasi-optimally in terms of the total degrees of freedom used.

5. Numerical experiments

In this section we carry out a few numerical experiments to verify the effectiveness and robustness of our adaptive Algorithm 3.1. The first two examples are the benchmark problems provided by Dauge [14] and the third one tests the robustness and reliability of the algorithm for the case with discontinuous coefficients ε and μ in (1.1) and (1.2).

The implementations are done with the help of the adaptive finite element package PHG [27] developed by Zhang et al. [33] at the State Key Laboratory of Scientific and Engineering Computing, the Chinese Academy of Sciences. All of the experiments were carried out on a Pentium IV PC with 3.0 GHz CPU and 1 G memory.

In all the numerical implementations, we use the preconditioned conjugate gradient (PCG) method with the nodal auxiliary subspace preconditioner proposed in [16] to solve equation (3.5); see the illustration at the end of Section 3.

5.1. Thin L-shaped domain

In this example, the domain Ω is set to be a thin L-shaped one, i.e., $[-1,1] \times [-1,1] \times [0,1] \setminus [0,1] \times [-1,0] \times [0,1]$ (see the left figure in Fig. 1). The parameters ε and μ are taken to be $\varepsilon = \mu = 1.0$. The first eigenvalue of this problem is 9.63972384472 [14]. Table 1 displays the number of PCG iterations for solving system (3.5) in each inverse iteration with

a relative reduction of the Euclidean norm of the residual vector by a factor of 10^{-10} to be a termination criterion. One can see that the PCG iteration works very well, with its convergence nearly independent of the mesh sizes and the number of PCG iterations is only 13 when the degrees of freedom reach 666,514. The left part of Fig. 2 displays the $(x-y)$ cross-section of the adaptive mesh with 120,502 degrees of freedom.

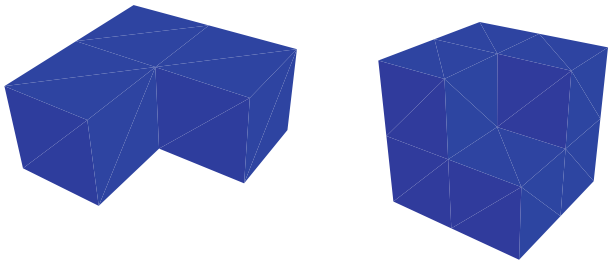

Fig. 1. The thin L-shaped domain (left) and the Fichera corner domain (right).

Table 1
The DOFs and PCG iterations with respect to the adaptive inverse iteration.

Adaptive Step	1	2	3	4	5	6
#DOF	47	262	1282	7418	16,237	39,404
PCG Iters	3	6	8	9	11	11
Adaptive Step	7	8	9	10		
#DOF	75,106	120,502	391,498	666,514		
PCG Iters	12	12	13	13		

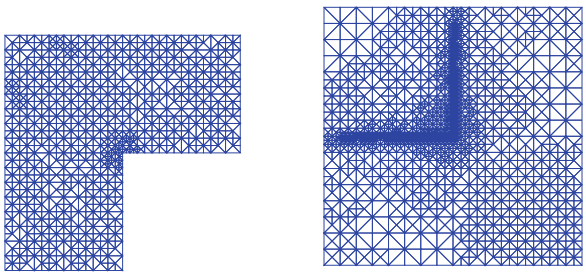

Fig. 2. Left: the $(x-y)$ cross-section centered at $(0,0,0.5)$ of the mesh generated by the adaptive inverse iteration for the thin L-shaped domain (with 120,502 unknowns). Right: The $(x-y)$ cross-section centered at $(0,0,0)$ of the mesh generated by adaptive inverse iteration for the Fichera domain (with 215,254 unknowns).

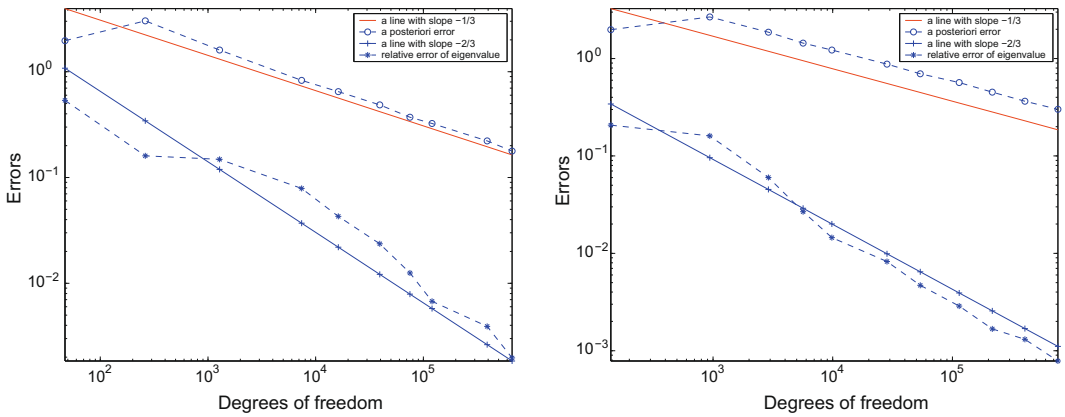

Fig. 3. The error reductions with respect to the adaptive inverse iteration for the thin L-shaped domain (left) and the Fichera domain (right).

It is known that the singularity of this problem is only caused by the reentrant edge along the z-direction. We observe that the mesh is very fine around this edge, which means that our error indicator detects the singularity very well. The left part of Fig. 3 shows the $\log \mathcal{E} - \log N$ curves with $\mathcal{E}$ representing the a posteriori error or relative error of the first eigenvalue. It indicates that the adaptive meshes and the associated numerical complexity are quasi-optimal in the sense that $\mathcal{E} \approx N^{-1/3}$ (or $\mathcal{E} \approx N^{-2/3}$ for eigenvalue).

5.2. Fichera corner domain

In this example, we take the domain to be the cube $[-1,1] \times [-1,1] \times [-1,1]$, with the cube $[0,1] \times [0,1] \times [0,1]$ cut off, as shown in the right of Fig. 1. The first eigenvalue of the problem is 3.21987401386 (with four reliable digits) [14].

Table 2 contains the number of the PCG iterations for solving the singular system (3.5) in each inverse iteration. A relative reduction of the Euclidean norm of the residual vector by a factor of 10^{-10} is chosen as a termination criterion. The PCG iteration converges again very well, nearly independent of the mesh sizes.

The right part of Fig. 2 shows that the mesh detects the singularities arising from both the non-convex edge and the corner at $(0,0,0)$. The right part of Fig. 3 shows the $\log \mathcal{E} - \log N$ curves for the a posteriori error and the relative error of the first eigenvalue and indicates that the adaptive meshes and the associated numerical complexity are quasi-optimal, i.e. $\mathcal{E} \approx N^{-1/3}$ (or $\mathcal{E} \approx N^{-2/3}$ for eigenvalue).

Table 2
The DOFs and PCG iterations with respect to the adaptive inverse iteration.

Adaptive step	1	2	3	4	5	6
#DOF	140	940	2901	5653	9843	28,381
PCG Iters	8	10	10	11	12	13
Adaptive step	7	8	9	10	11	
#DOF	53,817	113,908	215,254	402,879	759,520	
PCG Iters	13	14	14	14	15	

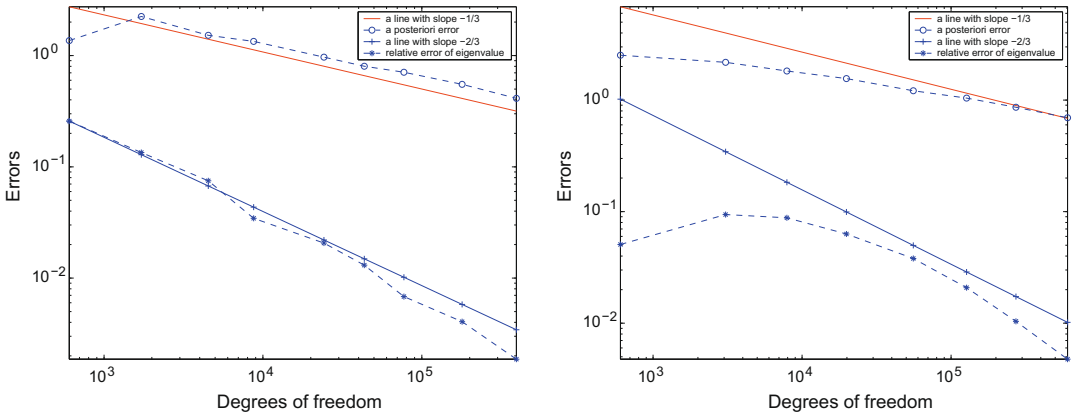

Fig. 4. Left: The error reductions with respect to the adaptive inverse iteration. $\mu_1 = 0.1,\ \mu_2 = 1.0$; Right: The error reductions with respect to adaptive inverse iterations. $\varepsilon_1 = 100,\ \varepsilon_2 = 1$.

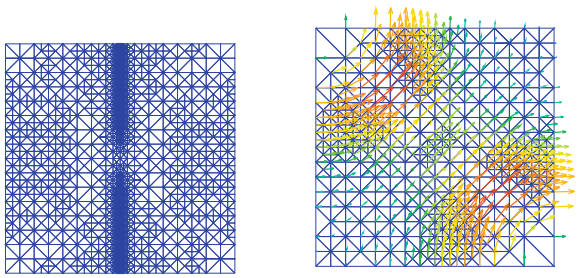

Fig. 5. Left: The $(y-z)$ cross-section centered at $(0,0,0)$ of the mesh generated by the adaptive inverse iteration (with 179,216 unknowns); Right: The eigenvector on the $(x-y)$ cross-section centered at $(0,0,0)$ (179,216 unknowns). $\mu_1 = 0.1,\ \mu_2 = 1.0$.

5.3. A cubic domain with discontinuous coefficients

This example concerns the Maxwell eigenvalue problem in a domain occupied by two different media. The computing domain is the cube $[-1,1]\times[-1,1]\times[-1,1]$, and the electric permittivity and magnetic permeability are assumed to be ε_1,μ_1 in domain $xy \geqslant 0$ and ε_2,μ_2 in domain $xy < 0$.

For the first case, we let $\mu_1 = 0.1, \mu_2 = 1.0$, and the electric permittivity $\varepsilon = 1.0$ all over the domain. Since the exact eigenvalue is unknown, we take the computing result on a very fine mesh generated by our adaptive algorithm as an exact one.

The left part of Fig. 4 shows the $\log\mathcal{E}-\log N$ curves. We observe that the reductions of a posteriori error and relative error of eigenvalue are quasi-optimal again.

Fig. 5 shows the mesh and the eigenvector generated by the adaptive inverse iteration respectively. We observe that the mesh is locally refined near z-axis, where the solution is singular due to the jump of the coefficient μ.

In the second case, we take $\varepsilon_1 = 100, \varepsilon_2 = 1$, and the discrete solution on the mesh with 1,089,777 unknowns to be the true solution. The results are presented in the right of Fig. 4 (with log–log plot). We observe that the relative errors of eigenvalue confirms the quasi-optimal convergence as before.

Acknowledgments

We would like to thank the anonymous referees for many constructive comments which have led to a great improvement of the results and presentation of the work, and Dr. T. Cui in LSEC, the Chinese Academy of Sciences, for some valuable discussions about the PCG implementation.

References

[1] R. Beck, R. Hiptmair, R.H.W. Hoppe, B. Wohlmuth, Residual based a posteriori error estimators for eddy current computation, M2AN Math. Model. Numer. Anal. 34 (2000) 159–182.
[2] M. Birman, M. Solomyak, L^2-theory of the Maxwell operator in arbitrary domains, Russian Math. Surv. 42 (1987) 75–96.
[3] D. Boffi, Fortin operators and discrete compactness for edge elements, Numer. Math. 87 (2000) 229–246.
[4] D. Boffi, P. Fernandes, L. Gastaldi, I. Perugia, Computational models of electromagnetic resonantors: analysis of edge element approximation, SIAM J. Numer. Anal. 36 (1999) 1264–1290.
[5] S. Caorsi, P. Fernandes, M. Raffetto, On the convergence of Galerkin finite element approximations of electromagnetic eigenproblems, SIAM J. Numer. Anal. 38 (2000) 580–607.
[6] J. Chen, Z. Chen, An adaptive perfectly matched layer technique for 3-d time-harmonic electromagnetic scattering problems, Math. Comput. 77 (2008) 673–698.
[7] J. Chen, Z. Chen, T. Cui, L. Zhang, An adaptive finite element method for the eddy current model with circuit/field couplings, SIAM J. Sci. Comput., accepted for publication.
[8] J. Chen, Y. Xu, J. Zou, An adaptive edge element method and its convergence for a saddle-point Maxwell's system, Technical Report 2008-01 (356), Department of Mathematics, The Chinese University of Hong Kong, 2008.
[9] J. Chen, Y. Xu, J. Zou, An adaptive inverse iteration for Maxwell eigenvalue problem based on edge elements, Technical Report, Department of Mathematics, The Chinese University of Hong Kong, 2009.
[10] Z. Chen, L. Wang, W. Zheng, An adaptive multilevel method for time-harmonic Maxwell equations with singularities, SIAM J. Sci. Comput. 29 (2007) 118–138.
[11] M. Costabel, M. Dauge, Maxwell and Lamé eigenvalues on polyhedra, Math. Methods Appl. Sci. 22 (1999) 243–258.
[12] M. Costabel, M. Dauge, Computation of resonance frequencies for Maxwell equations in nonsmooth domains, in: M. Ainsworth, P. Davies, D. Duncan, P. Martin, B. Rynne (Eds.), Topics in Computational Wave Propagation, Springer, 2004.
[13] J. Coyle, P. Ledger, Evidence of exponential convergence in the computation of Maxwell eigenvalues, Comput. Methods Appl. Mech. Eng. 194 (2005) 587–604.
[14] M. Dauge. <http://www.perso.univ-rennes1.fr/monique.dauge/>.
[15] R. Hiptmair, Finite elements in computational electromagnetism, Acta Numer. 11 (2002) 237–339.
[16] R. Hiptmair, J. Xu, Nodal auxiliary space preconditioning in $H(\mathrm{curl})$ and $H(\mathrm{div})$ spaces, SIAM J. Numer. Anal. 45 (2007) 2483–2509.
[17] Q. Hu, J. Zou, An iterative method with variable relaxation parameters for saddle-point problems, SIAM J. Matrix Anal. Appl. 23 (2001) 317–338.
[18] Q. Hu, J. Zou, Substructuring preconditioners for saddle-point problems arising from Maxwell's equations in three dimensions, Math. Comput. 73 (2004) 35–61.
[19] Q. Hu, J. Zou, Nonlinear inexact Uzawa algorithms for linear and nonlinear saddle-point problems, SIAM J. Optim. 16 (2006) 798–825.
[20] J.D. Jackson, Classical Electrodynamics, Wiley, New York, 1962.
[21] J. Jin, The Finite Element Method in Electromagnetics, John Wiley & Sons Inc., New York, 1993.
[22] F. Kikuchi, On a discrete compactness property for the Nedelec finite elements, J. Fac. Sci., Univer. Tokyo Sect. IA Math. 36 (1989) 479–490.
[23] Tz. V. Kolev, P.S. Vassilevski, Some Experience With a H1-Based Auxiliary Space AMG for $H(\mathrm{curl})$ Problems, LLNL Technical Report UCRL-TR-221841, June, 2006.
[24] P. Monk, Finite Element Methods for Maxwell's Equations, Oxford University Press, New York, 2003.
[25] P. Monk, L. Demkowicz, Discrete compactness and the approximation of Maxwell's equations in $\mathbb{R}^3$, Math. Comput. 70 (2000) 507–523.
[26] J.C. Nédélec, Mixed finite elements in $\mathbb{R}^3$, Numer. Math. 35 (1980) 315–341.
[27] PHG: Parallel Hierachical Grid. <http://www.lsec.cc.ac.cn/phg/>.
[28] Y. Saad, Iterative Methods for Sparse Linear Systems, PWS Publishing, Boston, 1996.
[29] L.R. Scott, S. Zhang, Finite element interpolation of nonsmooth functions satisfying boundary conditions, Math. Comput. 54 (1990) 483–493.
[30] R. Verfürth, A Review of A Posteriori Error Estimation and Adaptive Mesh-Refinement Techniques, Wiley-Teubner, New York, 1996.
[31] C. Weber, A local compactness theorem for Maxwell's equations, Math. Methods Appl. Sci. 2 (1980) 12–25.
[32] J. Xu, Y. Zhu, Uniform convergent multigrid methods for elliptic problems with strongly discontinuous coefficients, M3AS Math. Models Methods Appl. Sci. 18 (2008) 77–105.
[33] L. Zhang, A parallel algorithm for adaptive local refinement of tetrahedral meshes using bisection, Numer. Math.: Theory, Meth. Appl. 2 (2009) 65–89.