Science in China: *Series A Mathematics* 2006 Vol. 49 No. 1 1–28

DOI: 10.1007/s11425-005-0036-y

Uniform convergence of multigrid V-cycle on adaptively refined finite element meshes for second order elliptic problems

Haijun Wu & Zhiming Chen

Institute of Computational Mathematics, Chinese Academy of Sciences, Beijing 100080, China.
Current address of the first author is Department of Mathematics, Nanjing University, Nanjing, 210093, China.
(email: hjw@nju.edu.cn; zmchen@lsec.cc.ac.cn)
Received , 2005; accepted , 2006

Abstract In this paper we prove the uniform convergence of the standard multigrid V-cycle algorithm with Gauss-Seidel relaxation performed only on new nodes and their "immediate" neighbors for discrete elliptic problems on adaptively refined finite element meshes using the newest vertex bisection algorithm. The proof depends on sharp estimates on the relationship of local mesh sizes and a new stability estimate for the space decomposition based on Scott-Zhang interpolation operator. Extensive numerical results are reported which confirm the theoretical analysis.

Keywords: Multigrid V-cycle algorithm, adaptive finite element meshes, local relaxation, Scott-Zhang interpolation.

1 Introduction

Let Ω be a bounded polygonal domain in R^2 with possibly reentrant corners. Consider the variational problem of finding $u \in H_0^1(\Omega)$ such that

$$A(u,v) = F(v) \quad \forall v \in H_0^1(\Omega), \tag{1.1}$$

where $F \in H^{-1}(\Omega)$, the dual space of $H_0^1(\Omega)$, and

$$A(u,v) = \int_\Omega \left[p(x)\nabla u \cdot \nabla v + r(x)uv\right]\, dx \quad \forall u, v \in H_0^1(\Omega).$$

We assume that $p \in C^1(\overline{\Omega}), r \in C^0(\overline{\Omega}), p(x) > 0$ on $\overline{\Omega}$, and $r(x) \geqslant 0$ on $\overline{\Omega}$.

In this paper we study the V-cycle multigrid methods for discretized elliptic problems on adaptively refined finite element meshes. The adaptive finite element method based on a posteriori error estimates initiated in [1] provides a systematic way to refine or coarsen the mesh according to the local a posteriori error estimator on the elements. Recent studies (cf. e.g. [2], [3]) indicate that for appropriately designed adaptive finite element procedures, the meshes and the associated numerical complexity are quasi-optimal in the sense that the finite element discretization error is proportional to $N^{-1/2}$ in terms of the energy norm and

proportional to N^{-1} in terms of the maximum norm, where N is the number of degrees of freedom of the underlying mesh. Therefore, in order to achieve an optimal solution method for elliptic problems, it is imperative to study efficient algorithms for solving the linear system of equations arising from the adaptive finite element discretization of elliptic problems.

Let $\mathcal{M}_j,\ 0 \leq j \leq J$, be a sequence of nested finite element meshes of Ω, $\mathcal{N}_j$ the set of interior nodes of $\mathcal{M}_j$, and $X_j \subset H_0^1(\Omega)$ the piecewise linear finite element space over $\mathcal{M}_j$ with dimension n_j. The distinct feature of applying multigrid methods on adaptively refined finite element meshes is that the number of nodes of the mesh $\mathcal{M}_j$ may not grow exponentially with respect to the number of the mesh refinements j. Thus the number of operations used for the multigrid method in which the relaxation is performed on all nodes can be as bad as $O(n_j^2)$ [4]. To reduce the computational cost, various local relaxation schemes are proposed in applying multigrid methods on adaptively refined finite element meshes. Numerical experiments in [4] strongly suggest that the "local" multigrid algorithm which, at each level j, performs relaxation only at new nodes (i.e. the nodes in $\mathcal{N}_j \backslash \mathcal{N}_{j-1}$) and their neighboring nodes can guarantee uniform convergence of the multigrid methods for discrete elliptic problems with smooth coefficients. Here the set of neighboring nodes is defined conventionally as all those nodes, whose basis functions have a support that intersects with the support of basis functions of the new nodes.

The purpose of this paper is to show that the multigrid V-cycle algorithm which performs Gauss-Seidel relaxation at new nodes and the old nodes whose support of nodal basis function have changed can already guarantee uniform convergence of the multigrid V-cycle algorithm. Note that the set of nodes where relaxation is performed in our multigrid V-cycle algorithm is smaller than the set of nodes where the relaxation is performed in the "local" multigrid V-cycle algorithm [4]. Similarly, we also obtain the uniform convergence of the "local" multigrid V-cycle algorithm in [4] with Gauss-Seidel relaxations. We recall that performing relaxation only at new nodes can not lead to uniform convergence [5].

The proof of uniform convergence of our V-cycle algorithm is based on the practical condition that the adaptive mesh refinements are carried out by the "newest vertex bisection" algorithm developed in [6], [7], [4] and [8]. This refinement algorithm has been widely used in the adaptive finite element community (see e.g. [2], [3], [9]–[12]) and has been implemented in the package ALBERT [13]. Our result is quite general and can apply to any adaptive finite element strategy that use the newest vertex bisection algorithm for mesh refinements. In each local refinement step of the adaptive finite element method, the elements marked for refinements can be anywhere in the whole domain. As long as the local relaxation at each level (step) is performed at new nodes and the old nodes whose support of nodal basis function have changed, our V-cycle algorithm converges uniformly. In this paper, the old nodes whose support of nodal basis function have changed are also called the "immediate" neighboring nodes of new nodes, since they are endpoints of some

old edges that are bisected.

To the best of our knowledge, the uniform convergence result of our V-cycle algorithm as well as that of the "local" multigrid V-cycle algorithm on the newest vertex bisection does not fall under previous results. In particular, it is not assumed that the subdomains where the refinements take place at different levels are nested. The analysis in the paper contains substantial differences from the analysis developed for the fast adaptive composite grid (FAC) methods studied in [14], [15], [16], [17] or the multilevel adaptive technique (MLAT) studied in [18], [19], [20], [21], [22]. For example, the extension operators Q_j introduced in [20, Section 5] for the analysis cannot be defined for the meshes generated by the "newest vertex bisection" algorithm. We also remark that the multigrid convergence analyses for locally refined finite element meshes in [23], [24], [25], [26] are under more restrictive conditions on the meshes which are not satisfied by the "newest vertex bisection" algorithm. Other references on the convergence analysis for multigrid methods can be found in [27], [28], [29], [30], [31], [32], [33], [34].

Our convergence proof depends on a powerful identity of Xu and Zikatanov [35] for estimating the norm of the product of non-expansive operators and the following property of a properly defined Scott-Zhang interpolation operator $\Pi_j : X_J \to X_j$ (see Lemma 3.3 below)

$$\sum_{j=1}^{J} \sum_{z \in \widetilde{\mathcal{N}}_j} |(\Pi_j v - \Pi_{j-1} v)(z)|^2 \leqslant C A(v,v) \quad \forall v \in X_J,$$

Here $\widetilde{\mathcal{N}}_j$ is the set of new nodes (i.e. the nodes in $\mathcal{N}_j \backslash \mathcal{N}_{j-1}$) and their "immediate" neighboring nodes. This estimate is proved based on establishing appropriate connection between the adaptive meshes and uniformly refined meshes which extends similar idea in the analysis in [36], [20], and [22] but contains essential differences.

Throughout the paper, we use the notation $A_1 \lesssim B_1$ to represent the inequality $A_1 \leqslant constant \times B_1$, where the *constant* is positive and independent of all the variables in the inequality, and it is always assumed to be mesh-independent. The notation $A_1 \eqsim B_1$ is equivalent to the statement $A_1 \lesssim B_1$ and $B_1 \lesssim A_1$.

The rest of the paper is organized as follows. In §2 we recall the "newest vertex bisection" algorithm for mesh refinements, introduce the multigrid V-cycle algorthim with Gauss-Seidel relaxation performed only on the new nodes and their "immediate" neighbors, and present the main convergence theorems for our algorithm and the "local" multigrid algorithm in [4], respectively. In §3 we prove the uniform convergence theorems. In §4 we discuss implementation details and present several numerical examples to confirm our theoretical analysis. We also compare the performance of our algorithm with the "local" multigrid algorithm in [4]. In §5 we present some concluding remarks.

2 The main result

In this section we consider the uniform convergence of the multigrid V-cycle algorithm with local Gauss-Seidel relaxation performed only on new nodes and their "immediate" neighbors. This leads to an algorithm of optimal complexity for solving the linear system of equations resulting from the discretization of (1.1) by adaptive finite element methods. Here "optimal" means that one step of multigrid iteration can reduce the norm of the error of the approximate solution of the linear system by a factor that is bounded away from 1 independent of N, the size of the linear system, while using only $O(N)$ operations. We will first recall the "newest vertex bisection" algorithm for the mesh refinements in §2.1, and then introduce our multigrid V-cycle with local relaxation and present the main convergence theorems for our algorithm and the "local" multigrid algorithm respectively in §2.2.

2.1 The newest vertex bisection algorithm

We now recall the "newest vertex bisection" algorithm for the mesh refinements. A detailed description of the algorithm can be found in [6], [4] or [13]. The "newest vertex bisection" algorithm consists of two steps:

1. The marked triangles for refinements are bisected by the edge opposite to the newest vertex a fixed number of times (the newest vertex of an element in the initial mesh is the vertex opposite to the longest edge). The resultant triangulation may have nodes that are not the common vertices of two triangles. Such nodes are called hanging nodes.

2. All triangles with hanging nodes are bisected by the edge opposite to the newest vertex, this process is repeated until there are no hanging nodes.

It is proved in [6] that the iteration in the second step to remove the hanging nodes can be completed in finite number of steps. Let $\mathcal{M}_j, j = 1, 2, \cdots$, be a sequence of nested meshes generated by the newest vertex bisection algorithm. It is clear that each element $K \in \mathcal{M}_j$ is obtained by refining some element $K' \in \mathcal{M}_{j-1}$ finite number of times so that $h_{K'} \lesssim h_K$, where h_K is the diameter of the element K. It is also proved in [6] that there exists a constant $\theta > 0$ such that

$$\theta_K \geq \theta \quad \forall K \in \mathcal{M}_j, \;\; j = 1, 2, \cdots, \tag{2.1}$$

where θ_K is the minimum angle of the element K.

An important property of the newest vertex bisection algorithm is that the algorithm generates a sequence of meshes that all the descendants of an original triangle fall into four similarity classes indicated in Figure 1.

To conclude this subsection, we remark that the meshes considered in this paper are determined through a posteriori error estimators. In each refinement step, the elements marked for refinements can be anywhere in the whole domain, not necessarily confined

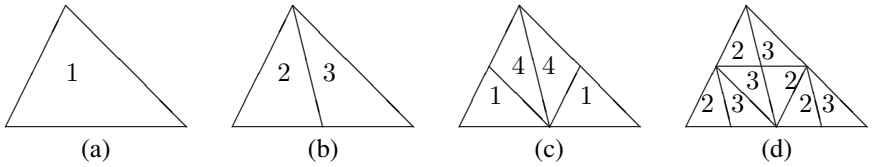

Figure 1: *Four similarity classes of triangles generated by "newest vertex bisection".*

to some specific subdomain. Moreover, the mesh size of the elements refined in each refinement step can vary largely. Depending on different refinement strategies based on a posteriori error estimators, it is well possible that some element is refined again after many refinement steps done in the other places of the domain. This a posteriori nature of doing mesh refinements makes the multigrid convergence analysis difficult, but it provides the great flexibility for applying the adaptive finite element methods based on a posteriori error estimates also to nonlinear problems in which the singularities of the solutions are a priori unknown. In Figure 2, we show two successional meshes generated by the newest vertex bisection algorithm for solving the Laplace equation on a domain with a crack (Example 4.2 in §4) by using adaptive finite element method based on a posteriori error estimates.

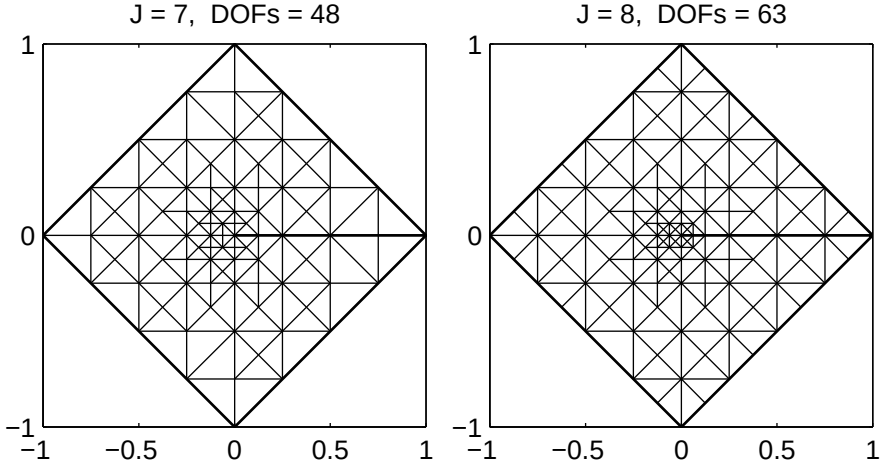

Figure 2: *The adaptively refined mesh of* 48 *interior nodes after* 7 *adaptive iterations (left) and the mesh of* 63 *interior nodes after* 8 *adaptive iterations (right) for the Laplace equation on the domain with a crack.*

2.2 The multigrid V-cycle algorithm

Let $\mathcal{M}_j, 0 \leqslant j \leqslant J$, be a sequence of nested conforming finite element triangulations of the domain Ω such that $\mathcal{M}_j$ is a refinement of $\mathcal{M}_{j-1}$. Let $X_j \subset H_0^1(\Omega)$ be the standard continuous piecewise linear finite element space over $\mathcal{M}_j$. Denote by $\mathcal{N}_j$ the collection of interior nodes of $\mathcal{M}_j$. For any node $z \in \mathcal{N}_j$, we use the notation ϕ_j^z to represent the associated nodal finite element basis function of X_j which takes the value 1 at the node z and the value 0 at all other nodes. Let $\widetilde{\mathcal{N}}_j$ be the set of nodes on which local Gauss-Seidel relaxation are carried out:

$$\widetilde{\mathcal{N}}_j = \left\{ z \in \mathcal{N}_j : \, z \in \mathcal{N}_j \setminus \mathcal{N}_{j-1} \text{ or } z \in \mathcal{N}_{j-1} \text{ but } \phi_j^z \neq \phi_{j-1}^z \right\}, \tag{2.2}$$

i.e., $\widetilde{\mathcal{N}}_j$ is the set of new nodes and their "immediate" neighboring nodes — the old nodes whose support of nodal basis function have changed. For convenience, we denote $\widetilde{\mathcal{N}}_j = \left\{x_j^k, k = 1, \cdots, \tilde{n}_j\right\}$ and let $\phi_j^k = \phi_j^{x_j^k}$ be the nodal finite element basis function corresponding to x_j^k. We remark that our set of nodes on which local relaxation are carried out is smaller than the set of nodes for relaxation in the "local" multigrid algorithm in [4].

For any $0 \leq j \leq J$, we define $A_j : X_j \to X_j$ by

$$(A_j w, v) = A(w, v), \quad \forall w, v \in X_j,$$

where the pairing $(\cdot, \cdot)$ is the inner product in $L^2(\Omega)$. Then the j^{th} level finite element discretization of (1.1) reads as: Find $u_j \in X_j$ such that

$$A_j u_j = f_j, \tag{2.3}$$

where $f_j \in X_j$ satisfying $(f_j, v) = F(v), \forall v \in X_j$. We also define the orthogonal projections $Q_j, P_j : X_J \to X_j$ by

$$(Q_j w, v) = (w, v), \quad A(P_j w, v) = A(w, v), \quad \forall v \in X_j, \ \forall w \in X_J.$$

For $k = 1, \cdots, \tilde{n}_j$, let $P_j^k : X_J \to X_j^k := \operatorname{span}\left\{\phi_j^k\right\}$ be defined by

$$A(P_j^k w, \phi_j^k) = A(w, \phi_j^k) \quad \forall w \in X_J.$$

Let $R_j : X_j \to X_j$ be the operator defined by

$$R_j = \Big(I - \prod_{k=1}^{\tilde{n}_j}(I - P_j^k)\Big) A_j^{-1}.$$

We remark that R_j is the smoothing operator which performs Gauss-Seidel relaxation only at new nodes and their "immediate" neighboring nodes.

The standard V-cycle multigrid algorithm solves the system (2.3) by the iterative method

$$u_j^{(m+1)} = u_j^{(m)} + B_j(f_j - A_j u_j^{(m)}).$$

The operators $B_j : X_j \to X_j, 0 \leqslant j \leqslant J$ are recursively defined as follows:

Algorithm 2.1 (V-cycle). *Let $B_0 = A_0^{-1}$. For $j > 0$ and $g \in X_j$, we define $B_j g = w_3$.*

(i) Pre-smoothing: $w_1 = R_j g$,

(ii) Correction: $w_2 = w_1 + B_{j-1} Q_{j-1}(g - A_j w_1)$,

(iii) Post-smoothing: $w_3 = w_2 + R_j^t(g - A_j w_2)$.

It is easy to see that the multigrid V-cycle operator B_J in Algorithm 2.1 satisfies the following relation (cf. e.g. [36, (3.4)])

$$I - B_J A_J = \Big[(I - P_0)\prod_{j=1}^{J}\prod_{k=1}^{\tilde{n}_j}(I - P_j^k)\Big]^* \Big[(I - P_0)\prod_{j=1}^{J}\prod_{k=1}^{\tilde{n}_j}(I - P_j^k)\Big],$$

where L^* denotes the adjoint operator of the linear operator $L : H_0^1(\Omega) \to H_0^1(\Omega)$ with respect to $A(\cdot, \cdot)$. Denote by $\|\cdot\|_A = A(\cdot, \cdot)^{\frac{1}{2}}$. Then since $I - B_J A_J$ is self-adjoint with

respect to the inner product $A(\cdot,\cdot)$, we have

$$\|I - B_J A_J\|_A = \sup_{0\neq v\in X_J} \frac{A((I-B_JA_J)v,v)}{\|v\|_A} = \Big\|(I-P_0)\prod_{j=1}^{J}\prod_{k=1}^{\tilde{n}_j}(I-P_j^k)\Big\|_A^2.$$

To proceed, we recall the following result of Xu and Zikatanov [35, Corollary 4.3].

Lemma 2.2. *Let V be a Hilbert space and $V_i \subset V$ $(i = 1, \cdots, M)$ a number of closed subspaces satisfying $V = \sum_{i=1}^{M} V_i$. Let $P_i : V \to V_i$ $(i = 1, \cdots, M)$ be the orthogonal projection with respect to the inner product of V. Then the following identity holds:*

$$\|(I-P_M)\cdots(I-P_1)\|^2_{\mathcal{L}(V,V)} = \frac{c_0}{1+c_0},$$

where

$$c_0 = \sup_{\|v\|=1}\inf_{v=\sum_{i=1}^{M} v_i} \sum_{i=1}^{M}\|P_i\sum_{j=i+1}^{M} v_j\|^2.$$

Note that $\|(I-P_1)\cdots(I-P_M)\|_{\mathcal{L}(V,V)} = \|(I-P_M)\cdots(I-P_1)\|_{\mathcal{L}(V,V)}$, and any function $v \in X_J$ can be decomposed as

$$v = v_0 + \sum_{j=1}^{J}\sum_{k=1}^{\tilde{n}_j} v_j^k, \text{ where } v_0 \in X_0,\ v_j^k \in X_j^k,$$

we deduce from Lemma 2.2 that

$$\|I - B_J A_J\|_A = \frac{c_0}{1+c_0} \tag{2.4}$$

with

$$c_0 = \sup_{\|v\|_A=1}\inf_{v_0+\sum_{j=1}^{J}\sum_{k=1}^{\tilde{n}_j} v_j^k = v} \Big(\Big\|P_0\sum_{j=1}^{J}\sum_{k=1}^{\tilde{n}_j} v_j^k\Big\|_A^2 + \sum_{j=1}^{J}\sum_{k=1}^{\tilde{n}_j}\Big\|P_j^k\sum_{(i,l)>(j,k)} v_i^l\Big\|_A^2\Big),$$

where $(i,l) > (j,k)$ means either $i > j$ or $i = j$ but $l > k$. The identity (2.4) is the starting point of our convergence analysis.

The following theorem whose proof will be given in the next section is the main result of this paper.

Theorem 2.3. *Let the meshes $\mathcal{M}_j, 1 \leqslant j \leqslant J$, be a sequence of nested conforming finite element triangulations of the domain Ω such that $\mathcal{M}_j$ is a refinement of $\mathcal{M}_{j-1}$ obtained by the "newest vertex bisection" algorithm. Then there exists a constant $\delta < 1$ independent of the meshes $\mathcal{M}_j$ and J such that*

$$\|I - B_J A_J\|_A < \delta.$$

Now we consider the uniform convergence of the "local" multigrid V-cycle algorithm [4] when performing Gauss-Seidel relaxations on the set

$$\bar{\mathcal{N}_j} = \Big\{z \in \mathcal{N}_j : \operatorname{support}(\phi_j^z)\bigcap \operatorname{support}(\phi_j^{z'}) \neq \emptyset \text{ for some new node } z' \text{ in } \mathcal{N}_j\Big\}.$$

It is clear that $\widetilde{\mathcal{N}}_j \subset \bar{\mathcal{N}}_j$. Without loss of generality, we denote $\bar{\mathcal{N}}_j = \{x_j^k : k = 1, \cdots, \bar{n}_j\}$, where $\bar{n}_j \geq \tilde{n}_j$. Denote by $\bar{B}_j : X_j \to X_j, 1 \leq j \leq J$, the operators defined as in Algorithm 2.1 with the smoothing operator R_j being replaced by the smoothing operator $\bar{R}_j$ which performs Gauss-Seidel relaxations on $\bar{\mathcal{N}}_j$. By Lemma 2.2, we have

$$\left\|I - \bar{B}_J A_J\right\|_A = \frac{\bar{c}_0}{1 + \bar{c}_0}$$

with

$$\bar{c}_0 = \sup_{\|v\|_A=1} \inf_{v_0+\sum_{j=1}^J \sum_{k=1}^{\bar{n}_j} v_j^k = v} \Big(\Big\| P_0 \sum_{j=1}^J \sum_{k=1}^{\bar{n}_j} v_j^k \Big\|_A^2 + \sum_{j=1}^J \sum_{k=1}^{\bar{n}_j} \Big\| P_j^k \sum_{(i,l)>(j,k)} v_i^l \Big\|_A^2 \Big).$$

We have the following theorem on the uniform convergence of the "local" multigrid V-cycle algorithm with Gauss-Seidel relaxations [4].

Theorem 2.4. *Let the meshes $\mathcal{M}_j, 1 \leqslant j \leqslant J$, be a sequence of nested conforming finite element triangulations of the domain Ω such that $\mathcal{M}_j$ is a refinement of $\mathcal{M}_{j-1}$ obtained by the "newest vertex bisection" algorithm. Then there exists a constant $\delta < 1$ independent of the meshes $\mathcal{M}_j$ and J such that*

$$\|I - \bar{B}_J A_J\|_A < \delta.$$

This theorem can be proved by following the proof in the next section of Theorem 2.3 with merely $\widetilde{\mathcal{N}}_j$'s replaced by $\bar{\mathcal{N}}_j$'s. We omit the details.

3 Convergence analysis

We start by introducing some notation. For any $z \in \mathcal{M}_j$, we denote by Ω_j^z the closure of the support of the corresponding finite element basis function ϕ_j^z. Recall that elements K in $\mathcal{M}_j$ are considered as closed. Define $h_j(z)$ the length of the shortest edge of $\mathcal{M}_j$ with one vertex at z and introduce

$$\rho_j(z) = \Big[\frac{\ln(h_j(z)/h_0)}{\ln(1/2)}\Big], \quad h_0 = \max_{K \in \mathcal{M}_0} h_K. \tag{3.1}$$

Here $[a]$ stands for the largest integer less than or equal to a, h_K is the diameter of the element K. Roughly speaking, $\rho_j(z)$ characterizes the actual number of refinements made for the elements inside Ω_j^z. It is easy to see that

$$\Big(1/2\Big)^{\rho_j(z)+1} h_0 < h_j(z) \leqslant \Big(1/2\Big)^{\rho_j(z)} h_0. \tag{3.2}$$

For any $x_j^k \in \widetilde{\mathcal{N}}_j$, for simplifying the notation, we also denote by $\Omega_j^k = \Omega_j^{x_j^k}$ the closure of the support of the basis function ϕ_j^k, and h_j^k the length of the shortest edge of $\mathcal{M}_j$ with one vertex at x_j^k.

Lemma 3.1. *Let the meshes $\mathcal{M}_j, 1 \leqslant j \leqslant J$, be a sequence of nested conforming finite element triangulations of the domain Ω such that $\mathcal{M}_j$ is a refinement of $\mathcal{M}_{j-1}$ obtained by the "newest vertex bisection" algorithm. Then*

(i) *Given an integer* $m \geqslant 0$ *and* $z \in \mathcal{N}_J$*, the cardinality of the set* $\sigma_1(m,z) = \big\{j : z \in \widetilde{\mathcal{N}}_j,\ \rho_j(z) = m,\ 0 \leqslant j \leqslant J\big\}$ *is bounded by some constant independent of* m, z *and* J.

(ii) *Given an integer* $m \geqslant 0$ *and* $z \in \mathcal{N}_J$*, the cardinality of the set* $\sigma_2(m,z) = \big\{(k,j) : x_j^k \in \widetilde{\mathcal{N}}_j,\ \rho_j(x_j^k) = m,\ z \in \Omega_j^k,\ 0 \leqslant j \leqslant J\big\}$ *is bounded by some constant independent of* m, z *and* J.

Recall that $\widetilde{\mathcal{N}}_j$ is the set of new nodes and their "immediate" neighboring nodes in $\mathcal{N}_j$, it is well possible that $z \in \widetilde{\mathcal{N}}_j$ and $z \in \widetilde{\mathcal{N}}_{j+l}$ for some $l \geq 1$ such that $\rho_j(z) = \rho_{j+l}(z)$. $\sigma_1(m,z)$ is defined as the number of such levels j that $z \in \widetilde{\mathcal{N}}_j$ and $\rho_j(z) = m$. On the other hand, for any $z \in N_J$, z possibly belongs to many sets Ω_j^k corresponding to some $x_j^k \in \widetilde{\mathcal{N}}_j$, $\sigma_2(m,z)$ is defined as the number of such nodes that $\rho_j(x_j^k) = m$.

Proof. We first notice that the set $\sigma_1(m,z)$ or $\sigma_2(m,z)$ may be empty, in which case, the lemma is trivial. To prove (i), we first note that if $z \in \widetilde{\mathcal{N}}_{j_1}$ and $z \in \widetilde{\mathcal{N}}_{j_2}$ with $j_1 < j_2$, then at least one edge in $\Omega_{j_1}^z$ will be bisected to obtain the new mesh $\mathcal{M}_{j_2}$. Let $j \in \sigma_1(m,z)$, by (2.1) there exists an integer d_1 depending only on θ in (2.1) such that each edge in Ω_j^z will be bisected after d_1 bisections. Note that if $h_{j+k}(z) \leqslant h_j(z)/2$ for some k, then $\rho_{j+k}(z) \geqslant \rho_j(z) + 1 = m + 1$. Thus the number of integers in the set $\sigma_1(m,z)$ is bounded by d_1. This proves (i).

To show (ii), we first prove that the number of nodes in the set $\mathcal{N}(m,z) = \big\{y : y \in \widetilde{\mathcal{N}}_j, \rho_j(y) = m, |y - z| \leqslant H_j(y)$ for some $0 \leqslant j \leqslant J\big\}$ is bounded by some constant independent of m, z and J. Here $H_j(y)$ is the length of the longest edge of $\mathcal{M}_j$ with one vertex at y. Note that by (2.1) and (3.2), $H_j(y) \leqslant \beta h_j(y) \leqslant \beta(1/2)^m h_0$ for some constant β depending only on θ in (2.1). Now suppose that $y_1 \in \widetilde{\mathcal{N}}_{j_1}, \rho_{j_1}(y_1) = m$ and $y_2 \in \widetilde{\mathcal{N}}_{j_2}, \rho_{j_2}(y_2) = m$. Without loss of generality, we may assume $j_1 \geqslant j_2$, thus y_2 is also a node of $\mathcal{M}_{j_1}$. Therefore $|y_1 - y_2| \geqslant h_{j_1}(y_1) > (1/2)^{m+1} h_0$. This implies that the number of nodes in $\mathcal{N}(m,z)$ is bounded by some constant independent of m, z and J, i.e. $\#\mathcal{N}(m,z) \lesssim 1$. Now

$$\begin{aligned}
\#\sigma_2(m,z) &= \#\Big\{(y,j) : y \in \widetilde{\mathcal{N}}_j,\ \rho_j(y) = m,\ z \in \Omega_j^y,\ 0 \leqslant j \leqslant J\Big\} \\
&\lesssim \#\left\{(y,j);\ y \in \mathcal{N}(m,z),\ j \in \sigma_1(m,y)\right\} \\
&\lesssim \#\mathcal{N}(m,z) \times \max_{y \in \mathcal{N}_J} \#\sigma_1(m,y) \lesssim 1.
\end{aligned}$$

This completes the proof of (ii). ∎

Denote by $\mathcal{E}_j^k$ the collection of edges of $\mathcal{M}_j$ that are included in Ω_j^k. The following lemma whose proof may be of independent interest.

Lemma 3.2. *Let the meshes* $\mathcal{M}_j, 1 \leqslant j \leqslant J$*, be a sequence of nested conforming finite element triangulations of the domain* Ω *such that* $\mathcal{M}_j$ *is a refinement of* $\mathcal{M}_{j-1}$ *obtained*

by the "newest vertex bisection" algorithm. Then we have, for any $x_j^k \in \widetilde{\mathcal{N}}_j$,

$$\sum_{i=j+1}^{J}\sum_{x_i^l\in\widetilde{\mathcal{N}}_i, x_i^l\in\mathcal{E}_j^k}\left(h_i^l/h_j^k\right)^{3/2}\lesssim 1,\quad \sum_{i=j+1}^{J}\sum_{x_i^l\in\widetilde{\mathcal{N}}_i, x_i^l\in\Omega_j^k}\left(h_i^l/h_j^k\right)^{3}\lesssim 1, \tag{3.3}$$

and, for any $x_i^l \in \widetilde{\mathcal{N}}_i$,

$$\sum_{j=1}^{i-1}\sum_{x_j^k\in\widetilde{\mathcal{N}}_j, x_i^l\in\mathcal{E}_j^k}\left(h_i^l/h_j^k\right)^{1/2}\lesssim 1,\quad \sum_{j=1}^{i-1}\sum_{x_j^k\in\widetilde{\mathcal{N}}_j, x_i^l\in\Omega_j^k}\left(h_i^l/h_j^k\right)\lesssim 1. \tag{3.4}$$

Proof. By (2.1), as in the proof of (ii) of Lemma 3.1 we have there exists a constant $\beta>1$ depending only on the minimum angle of the meshes such that $H_j(x_j^k)\leqslant \beta h_j^k$. Then for any $x_i^l\in\widetilde{\mathcal{N}}_i, x_i^l\in\mathcal{E}_j^k$, we have $h_i^l\leqslant H_j(x_j^k)\leqslant \beta h_j^k$. Thus by the definition (3.1), $\rho_i(x_i^l)\geqslant \rho_j(x_j^k)-[\ln\beta/\ln 2]-1$. Let $n_0=[\ln\beta/\ln 2]+1>0$. By (3.2) we have

$$\begin{aligned}\sum_{i=j+1}^{J}\sum_{x_i^l\in\widetilde{\mathcal{N}}_i, x_i^l\in\mathcal{E}_j^k}\left(h_i^l\right)^{3/2}&\lesssim h_0^{3/2}\sum_{i=j+1}^{J}\sum_{x_i^l\in\widetilde{\mathcal{N}}_i, x_i^l\in\mathcal{E}_j^k}\left(\frac{1}{\sqrt{2}}\right)^{3\rho_i(x_i^l)}\\&\leqslant h_0^{3/2}\sum_{m=\rho_j(x_j^k)-n_0}^{\infty}\sum_{\substack{z\in\widetilde{\mathcal{N}}_i, z\in\mathcal{E}_j^k\\ \rho_i(z)=m, j+1\leqslant i\leqslant J}}\left(\frac{1}{\sqrt{2}}\right)^{3m}\\&\leqslant h_0^{3/2}\sum_{m=\rho_j(x_j^k)-n_0}^{\infty}\sum_{\substack{z\in\mathcal{N}_J, z\in\mathcal{E}_j^k\\ i\in\sigma_1(m,z)}}\left(\frac{1}{\sqrt{2}}\right)^{3m}.\end{aligned}$$

Since the distance of any two distinct nodes $z_1, z_2\in\mathcal{E}_j^k$ such that $\rho_{j_1}(z_1)=\rho_{j_2}(z_2)=m$ for some $j_1, j_2\geqslant 0$ is greater than $(1/2)^{m+1}h_0$, the number of the nodes $z\in\mathcal{E}_j^k$ satisfying $\rho_i(z)=m$ for some $i\geqslant 0$ is $\lesssim h_j^k/((1/2)^{m+1}h_0)$. Thus by (i) of Lemma 3.1, we deduce that

$$\begin{aligned}\sum_{i=j+1}^{J}\sum_{x_i^l\in\widetilde{\mathcal{N}}_i, x_i^l\in\mathcal{E}_j^k}\left(h_i^l\right)^{3/2}&\lesssim h_0^{3/2}\sum_{m=\rho_j(x_j^k)-n_0}^{\infty}\left(\frac{1}{\sqrt{2}}\right)^{3m}\cdot\frac{h_j^k}{(1/2)^{m+1}h_0}\\&\lesssim h_0^{1/2}\sum_{m=\rho_j(x_j^k)-n_0}^{\infty}\left(\frac{1}{\sqrt{2}}\right)^{m}\cdot h_j^k\lesssim h_0^{1/2}\left(\frac{1}{2}\right)^{\rho_j(x_j^k)/2}h_j^k\\&\lesssim (h_j^k)^{3/2}.\end{aligned}$$

This proves the first estimate in (3.3). The second estimate in (3.3) can be proved similarly. Here we omit the details.

Now we turn to (3.4) and again we only prove the first estimate. By (ii) of Lemma 3.1

and (3.2), we have

$$\begin{aligned}\sum_{j=1}^{i-1}\sum_{x_j^k\in\widetilde{\mathcal{N}}_j,x_i^l\in\mathcal{E}_j^k}(h_j^k)^{-1/2} &\lesssim h_0^{-1/2}\sum_{j=1}^{i-1}\sum_{x_j^k\in\widetilde{\mathcal{N}}_j,x_i^l\in\mathcal{E}_j^k}\Big(\frac{1}{\sqrt{2}}\Big)^{-\rho_j(x_j^k)}\\ &\lesssim h_0^{-1/2}\sum_{m=0}^{\rho_i(x_i^l)+n_0}\sum_{(k,j)\in\sigma_2(m,x_i^l)}\Big(\frac{1}{\sqrt{2}}\Big)^{-m}\\ &\lesssim h_0^{-1/2}\sum_{m=0}^{\rho_i(x_i^l)+n_0}(\sqrt{2})^m\lesssim h_0^{-1/2}(\sqrt{2})^{\rho_i(x_i^l)+n_0}\\ &\lesssim (h_i^l)^{-1/2}.\end{aligned}$$

This completes the proof. ■

Now we introduce the Scott-Zhang interpolation operator $\Pi_j : H_0^1(\Omega)\to X_j$. For any $z\in\mathcal{N}_j$, let e_z be an edge with one vertex at z, then the Scott-Zhang interpolation operator is defined as [37]:

$$(\Pi_j v)(z)=\int_{e_z}\psi_z(x)v(x)ds\quad\forall v\in H_0^1(\Omega),\tag{3.5}$$

where the linear function $\psi_z(x)$ satisfies $\int_{e_z}\psi_z(x)w(x)ds=w(z)$ for any linear function $w(x)$ on e_z. It is easy to check that $\psi_z(x)=(4-6\,|x-z|\,/h_{e_z})/h_{e_z}$, where h_{e_z} is the length of the edge e_z, and

$$(\Pi_j v)(z)=v(z)\text{ for any function }v(x)\text{ that is linear on }e_z,\tag{3.6}$$

in particular, $\Pi_j v_j=v_j$ for any $v_j\in X_j$. For any $z\in\mathcal{N}_j\setminus\widetilde{\mathcal{N}}_j$, that is, z is also a node in $\widetilde{\mathcal{N}}_{j-1}$ so that $\phi_j^z=\phi_{j-1}^z$, we require in addition that the edge e_z in the definition (3.5) is the same as the edge used in defining Π_{j-1}. This implies the following important property

$$\left(\Pi_j v-\Pi_{j-1}v\right)(z)=0\quad\forall z\in\mathcal{N}_j\setminus\widetilde{\mathcal{N}}_j.\tag{3.7}$$

Lemma 3.3. *Let the meshes $\mathcal{M}_j, 1\leqslant j\leqslant J$, be a sequence of nested conforming finite element triangulations of the domain Ω such that $\mathcal{M}_j$ is a refinement of $\mathcal{M}_{j-1}$ obtained by the "newest vertex bisection" algorithm. Then we have*

$$\sum_{j=1}^{J}\sum_{z\in\widetilde{\mathcal{N}}_j}\left|(\Pi_j v-\Pi_{j-1}v)(z)\right|^2\lesssim\|v\|_A^2\,.$$

Proof. The proof depends on a close relation between the adaptively refined meshes $\mathcal{M}_j$, $0\leqslant j\leqslant J$, and a sequence of uniformly refined meshes $\widehat{\mathcal{M}}_j$ which is obtained by connecting the edge midpoints of $\widehat{\mathcal{M}}_{j-1}$ starting from $\widehat{\mathcal{M}}_0=\mathcal{M}_0$. Denote by $\widehat{\mathcal{N}}_j$ the set of interior nodes of $\widehat{\mathcal{M}}_j$ and $\hat{h}_j=(1/2)^j h_0$.

We start by observing from the newest vertex bisection algorithm that each element $K\in\mathcal{M}_j$ is included in some element $K_0\in\mathcal{M}_0$, for example, K_0 is the triangle in Figure 1(a), and K is similar to one of the four types of triangles $K_0^{(i)}, i=1,2,3,4$,

where $K_0^{(1)} = K_0$ and $K_0^{(i)}(i = 2, 3, 4)$ is the triangle marked by i in Figure 1(b)(c). Moreover, there exists an integer $n(K)$ such that $h_K = (1/2)^{n(K)} h_{K_0^{(i)}}$. Thus K is included in some element $\widehat{K} \in \widehat{\mathcal{M}}_{n(K)}$. For any $z \in \mathcal{N}_j$, we choose an element $K' \in \mathcal{M}_{j-1}$ which contains z and define

$$L_j(z) = \min\left\{n(K) : K \in \mathcal{M}_{j-1}, K \bigcap K' \neq \emptyset\right\},$$
$$S_j(z) = \bigcup\left\{K \in \mathcal{M}_{j-1} : K \bigcap K' \neq \emptyset\right\}.$$

Obviously, for each element $K \subset S_j(z)$ that is in $\mathcal{M}_{j-1}$, there exists some element $\widehat{K} \in \widehat{\mathcal{M}}_{L_j(z)}$ such that $K \subseteq \widehat{K}$. We make the following claims:

Claim 1°: $\rho_j(z) \leqslant L_j(z) + d$, for some constant d depending only on θ in (2.1).

Claim 2°: $z \in \widehat{\mathcal{N}}_{\rho_j(z)}$ and $h_j(z) \eqsim \hat{h}_{\rho_j(z)}$.

First, by (2.1) and the property that any element in $\mathcal{M}_j$ is obtained by refining some element in $\mathcal{M}_{j-1}$ a fixed number of times, we know that there exists a constant $\gamma > 0$ such that $h_j(z) \geqslant \gamma h_K$ for any $K \subset S_j(z)$, that is, $h_j(z) \geqslant \gamma(1/2)^{n(K)} h_{K_0^{(i)}} \geqslant \gamma'(1/2)^{n(K)} h_0$ for some constant γ' by the quasi-uniformity of the initial mesh $\mathcal{M}_0$. By the definition of $\rho_j(z)$ we see that $\rho_j(z) \leqslant n(K) + [\ln \gamma' / \ln(1/2)]$ for any $K \subset S_j(z)$, which implies Claim 1° with $d = [\ln \gamma' / \ln(1/2)]$.

Next, by the newest vertex bisection algorithm, there is an edge e_z of $\mathcal{M}_j$ started at z such that e_z is an edge of $\widehat{\mathcal{M}}_m$ for some m. Thus $z \in \widehat{\mathcal{N}}_m$ and $h_j(z) \leqslant |e_z| \leqslant \hat{h}_m = (1/2)^m h_0$. On the other hand, $\rho_j(z) > (1/2)^{\rho_j(z)+1} h_0$ by (3.2), we conclude then $m < \rho_j(z) + 1$, i.e. $m \leqslant \rho_j(z)$. Thus Claim 2° holds.

Let $\widehat{Q}_j : L^2(\Omega) \to \widehat{X}_j$ be the standard L^2–projection operator and $\widehat{Q}_j = \widehat{Q}_0$ if $j < 0$, where $\widehat{X}_j \subset H_0^1(\Omega)$ is the piecewise linear finite element space over $\widehat{\mathcal{M}}_j$. From Claim 1°, $\widehat{Q}_{\rho_j(z)-d} v \in \widehat{X}_{L_j(z)}$ is linear on each element in $\widehat{\mathcal{M}}_{L_j(z)}$, and hence is linear on each element $K \subset S_j(z)$. Therefore from (3.6),

$$\left(\Pi_j \widehat{Q}_{\rho_j(z)-d} v\right)(z) = \left(\widehat{Q}_{\rho_j(z)-d} v\right)(z) = \left(\Pi_{j-1} \widehat{Q}_{\rho_j(z)-d} v\right)(z). \tag{3.8}$$

By the stability property of the Scott-Zhang interpolation operator [37] we have then that for any $\alpha \in (\frac{1}{2}, 1)$,

$$\begin{aligned} |(\Pi_j - \Pi_{j-1}) v(z)| &= \left|(\Pi_j - \Pi_{j-1})\left(v - \widehat{Q}_{\rho_j(z)-d} v\right)(z)\right| \\ &\lesssim h_j(z)^{-1} \left\|v - \widehat{Q}_{\rho_j(z)-d} v\right\|_{L^2(S_j(z))} + h_j(z)^{\alpha-1} \left\|v - \widehat{Q}_{\rho_j(z)-d} v\right\|_{H^\alpha(S_j(z))}. \end{aligned} \tag{3.9}$$

By Claim 2°, $S_j(z) \subset B(z, c\hat{h}_{\rho_j(z)}) = \left\{y \in \Omega : |y - z| < c\hat{h}_{\rho_j(z)}\right\}$ for some constant

c depending only on θ in (2.1), summing up (3.9) we know that

$$\begin{aligned}
&\sum_{j=1}^{J}\sum_{z\in\widetilde{\mathcal{N}}_j}|(\Pi_j-\Pi_{j-1})v(z)|^2\\
&\lesssim\sum_{j=1}^{J}\sum_{z\in\widetilde{\mathcal{N}}_j}\left(\hat{h}_{\rho_j(z)}^{-2}\left\|v-\widehat{Q}_{\rho_j(z)-d}v\right\|^2_{L^2(S_j(z))}+\hat{h}_{\rho_j(z)}^{2\alpha-2}\left\|v-\widehat{Q}_{\rho_j(z)-d}v\right\|^2_{H^\alpha(S_j(z))}\right)\\
&\lesssim\sum_{m=0}^{\infty}\sum_{\substack{z\in\widetilde{\mathcal{N}}_j,\rho_j(z)=m\\1\leqslant j\leqslant J}}\left(\hat{h}_m^{-2}\left\|v-\widehat{Q}_{m-d}v\right\|^2_{L^2(B(z,c\hat{h}_m))}+\hat{h}_m^{2\alpha-2}\left\|v-\widehat{Q}_{m-d}v\right\|^2_{H^\alpha(B(z,c\hat{h}_m))}\right)\\
&\lesssim\sum_{m=0}^{\infty}\sum_{z\in\widehat{\mathcal{N}}_m\cap\mathcal{N}_J}\sum_{j\in\sigma_1(m,z)}\Big(\hat{h}_m^{-2}\left\|v-\widehat{Q}_{m-d}v\right\|^2_{L^2(B(z,c\hat{h}_m))}\\
&\qquad\qquad+\hat{h}_m^{2\alpha-2}\left\|v-\widehat{Q}_{m-d}v\right\|^2_{H^\alpha(B(z,c\hat{h}_m))}\Big).
\end{aligned}$$

By (i) of Lemma 3.1 we have then

$$\begin{aligned}
&\sum_{j=1}^{J}\sum_{z\in\widetilde{\mathcal{N}}_j}|(\Pi_j-\Pi_{j-1})v(z)|^2\\
&\lesssim\sum_{m=0}^{\infty}\sum_{z\in\widehat{\mathcal{N}}_m}\left(\hat{h}_m^{-2}\left\|v-\widehat{Q}_{m-d}v\right\|^2_{L^2(B(z,c\hat{h}_m))}+\hat{h}_m^{2\alpha-2}\left\|v-\widehat{Q}_{m-d}v\right\|^2_{H^\alpha(B(z,c\hat{h}_m))}\right)\\
&\lesssim\sum_{m=0}^{\infty}\left(\hat{h}_m^{-2}\left\|v-\widehat{Q}_{m-d}v\right\|^2_{L^2(\Omega)}+\hat{h}_m^{2\alpha-2}\left\|v-\widehat{Q}_{m-d}v\right\|^2_{H^\alpha(\Omega)}\right)\\
&\lesssim\sum_{m=0}^{\infty}\left(\hat{h}_m^{-2}\left\|v-\widehat{Q}_{m}v\right\|^2_{L^2(\Omega)}+\hat{h}_m^{2\alpha-2}\left\|v-\widehat{Q}_{m}v\right\|^2_{H^\alpha(\Omega)}\right).
\end{aligned}$$

To conclude the proof the lemma, it is enough to show that: For any $s\in[0,1)$,

$$\sum_{m=0}^{\infty}\hat{h}_m^{-2(1-s)}\left\|v-\widehat{Q}_mv\right\|^2_{H^s(\Omega)}\lesssim\|v\|^2_{H^1(\Omega)}.$$

By [22, Theorem 4.32] we have

$$\left\|w-\widehat{Q}_0w\right\|^2_{H^{s'}(\Omega)}\eqsim\sum_{l=1}^{\infty}\hat{h}_l^{-2s'}\left\|(\widehat{Q}_l-\widehat{Q}_{l-1})w\right\|^2_{L^2(\Omega)}\quad\forall w\in H_0^1(\Omega),\ s'\in[0,1]. \tag{3.10}$$

Hence, by taking $w=v-\widehat{Q}_mv$ in (3.10) and using $\widehat{Q}_l\widehat{Q}_mv=\widehat{Q}_{\min(l,m)}v$, we obtain

$$\left\|v-\widehat{Q}_mv\right\|^2_{H^s(\Omega)}\eqsim\sum_{l=m+1}^{\infty}\hat{h}_l^{-2s}\left\|(\widehat{Q}_l-\widehat{Q}_{l-1})v\right\|^2_{L^2(\Omega)}.$$

So

$$\begin{aligned}\sum_{m=0}^{\infty}\hat{h}_m^{-2(1-s)}\left\|v-\widehat{Q}_m v\right\|^2_{H^s(\Omega)} &\eqsim \sum_{m=0}^{\infty}\sum_{l=m+1}^{\infty}\hat{h}_m^{-2(1-s)}\hat{h}_l^{-2s}\left\|(\widehat{Q}_l-\widehat{Q}_{l-1})v\right\|^2_{L^2(\Omega)}\\ &=\sum_{l=1}^{\infty}\Big(\sum_{m=0}^{l-1}\hat{h}_m^{-2(1-s)}\Big)\hat{h}_l^{-2s}\left\|(\widehat{Q}_l-\widehat{Q}_{l-1})v\right\|^2_{L^2(\Omega)}\\ &\eqsim \sum_{l=1}^{\infty}\hat{h}_l^{-2}\left\|(\widehat{Q}_l-\widehat{Q}_{l-1})v\right\|^2_{L^2(\Omega)}\lesssim \|v\|^2_{H^1(\Omega)}.\end{aligned}$$

We have used (3.10) and $\left\|\widehat{Q}_0 v\right\|_{H^1(\Omega)}\lesssim \|v\|_{H^1(\Omega)}$ to derive the last inequality. This completes the proof. ■

Now we are ready to prove Theorem 2.3.

Proof of Theorem 2.3. We only need to estimate c_0 in (2.4). For any $v\in X_J$, we consider the decomposition

$$v=\Pi_0 v+\sum_{j=1}^{J}v_j,\quad v_j=\Pi_j v-\Pi_{j-1}v=\sum_{k=1}^{\tilde{n}_j}v_j^k,\quad v_j^k=v_j(x_j^k)\phi_j^k.$$

Then

$$c_0\leqslant \sup_{\|v\|_A=1}c(v), \tag{3.11}$$

where

$$c(v):=\left\|P_0\sum_{j=1}^{J}\sum_{k=1}^{\tilde{n}_j}v_j^k\right\|_A^2+\sum_{j=1}^{J}\sum_{k=1}^{\tilde{n}_j}\Big(\left\|P_j^k\sum_{l=k+1}^{\tilde{n}_j}v_j^l\right\|_A^2+\left\|P_j^k\sum_{i=j+1}^{J}\sum_{l=1}^{\tilde{n}_i}v_i^l\right\|_A^2\Big). \tag{3.12}$$

By the stability estimate of Scott-Zhang interpolation operator, we have

$$\left\|P_0\sum_{j=1}^{J}\sum_{k=1}^{\tilde{n}_j}v_j^k\right\|_A^2=\|P_0(v-\Pi_0 v)\|_A^2\lesssim\|v\|_A^2 \tag{3.13}$$

Since there are only finite number of nodes $x_j^l\in\Omega_j^k$ the closure of the support of the basis function ϕ_j^k, we have

$$\sum_{j=1}^{J}\sum_{k=1}^{\tilde{n}_j}\left\|P_j^k\sum_{l=k+1}^{\tilde{n}_j}v_j^l\right\|_A^2\lesssim\sum_{j=1}^{J}\sum_{k=1}^{\tilde{n}_j}\left\|v_j^k\right\|_A^2\lesssim\sum_{j=1}^{J}\sum_{k=1}^{\tilde{n}_j}|v_j(x_j^k)|^2. \tag{3.14}$$

Since for any $w\in H_0^1(\Omega)$, $P_j^k w=[A(w,\phi_j^k)/A(\phi_j^k,\phi_j^k)]\phi_j^k$ and $A(\phi_j^k,\phi_j^k)\eqsim 1$ we have

$$\left\|P_j^k\sum_{i=j+1}^{J}\sum_{l=1}^{\tilde{n}_i}v_i^l\right\|_A^2\lesssim\Big|\sum_{i=j+1}^{J}\sum_{l=1}^{\tilde{n}_i}A(v_i^l,\phi_j^k)\Big|^2\lesssim\Big(\sum_{i=j+1}^{J}\sum_{x_i^l\in\widetilde{\mathcal{N}}_i,x_i^l\in\Omega_j^k}\big|A(\phi_i^l,\phi_j^k)v_i(x_i^l)\big|\Big)^2$$

Now, for $x_i^l \in \Omega_j^k$, we know that

$$\begin{aligned} A(\phi_i^l, \phi_j^k) &= \sum_{K\subset\Omega_j^k, K\in\mathcal{M}_j} \int_K \Big(p(x)\nabla\phi_j^k\nabla\phi_i^l + r(x)\phi_j^k\phi_i^l\Big)\,dx \\ &= \sum_{K\subset\Omega_j^k, K\in\mathcal{M}_j} \int_{\partial K} p(x)(\nabla\phi_j^k\cdot\nu)\phi_i^l\,ds \\ &\quad + \sum_{K\subset\Omega_j^k, K\in\mathcal{M}_j} \int_K \Big(-\nabla p(x)\cdot\nabla\phi_j^k + r(x)\phi_j^k\Big)\phi_i^l\,dx. \end{aligned}$$

Note that for $x_i^l \notin \mathcal{E}_j^k$ and $K\subset\Omega_j^k$, $\phi_i^l = 0$ on ∂K, thus

$$\begin{aligned} &\sum_{x_i^l\in\widetilde{\mathcal{N}}_i, x_i^l\in\Omega_j^k} \Big|\sum_{K\subset\Omega_j^k, K\in\mathcal{M}_j} \Big(\int_{\partial K} p(x)(\nabla\phi_j^k\cdot\nu)\phi_i^l ds\Big) v_i(x_i^l)\Big| \\ =\ &\sum_{x_i^l\in\widetilde{\mathcal{N}}_i, x_i^l\in\mathcal{E}_j^k} \Big|\sum_{K\subset\Omega_j^k, K\in\mathcal{M}_j} \Big(\int_{\partial K} p(x)(\nabla\phi_j^k\cdot\nu)\phi_i^l ds\Big) v_i(x_i^l)\Big| \\ \lesssim\ &\sum_{x_i^l\in\widetilde{\mathcal{N}}_i, x_i^l\in\mathcal{E}_j^k} (h_i^l/h_j^k)|v_i(x_i^l)|, \end{aligned}$$

where in the second inequality we have used $|\nabla\phi_j^k| \lesssim (h_j^k)^{-1}$. Therefore, we have

$$\begin{aligned} \Big\|P_j^k \sum_{i=j+1}^{J}\sum_{l=1}^{\tilde{n}_i} v_i^l\Big\|_A^2 \lesssim &\Big(\sum_{i=j+1}^{J}\sum_{x_i^l\in\widetilde{\mathcal{N}}_i, x_i^l\in\mathcal{E}_j^k} (h_i^l/h_j^k)|v_i(x_i^l)|\Big)^2 \\ &+ \Big(\sum_{i=j+1}^{J}\sum_{x_i^l\in\widetilde{\mathcal{N}}_i, x_i^l\in\Omega_j^k} (h_i^l)^2(h_j^k)^{-1}|v_i(x_i^l)|\Big)^2. \end{aligned} \tag{3.15}$$

By (3.3) and Cauchy-Schwarz inequality, we have

$$\begin{aligned} \Big(\sum_{i=j+1}^{J}\sum_{x_i^l\in\widetilde{\mathcal{N}}_i, x_i^l\in\mathcal{E}_j^k} (h_i^l/h_j^k)|v_i(x_i^l)|\Big)^2 &\lesssim \sum_{i=j+1}^{J}\sum_{x_i^l\in\widetilde{\mathcal{N}}_i, x_i^l\in\mathcal{E}_j^k} (h_i^l/h_j^k)^{1/2}|v_i(x_i^l)|^2 \\ &= \sum_{i=j+1}^{J}\sum_{x_i^l\in\widetilde{\mathcal{N}}_i} (h_i^l/h_j^k)^{1/2}|v_i(x_i^l)|^2\delta(x_j^k, x_i^l), \end{aligned}$$

where $\delta(x_j^k, x_i^l) = 1$ if $x_i^l \in \mathcal{E}_j^k$ and $\delta(x_j^k, x_i^l) = 0$ otherwise. Thus

$$\begin{aligned}
&\sum_{j=1}^{J}\sum_{x_j^k\in\widetilde{\mathcal{N}}_j}\Big(\sum_{i=j+1}^{J}\sum_{x_i^l\in\widetilde{\mathcal{N}}_i, x_i^l\in\mathcal{E}_j^k}(h_i^l/h_j^k)|v_i(x_i^l)|\Big)^2\\
&\qquad\lesssim\sum_{j=1}^{J}\sum_{x_j^k\in\widetilde{\mathcal{N}}_j}\sum_{i=j+1}^{J}\sum_{x_i^l\in\widetilde{\mathcal{N}}_i}(h_i^l/h_j^k)^{1/2}|v_i(x_i^l)|^2\delta(x_j^k,x_i^l)\\
&\qquad=\sum_{i=2}^{J}\sum_{x_i^l\in\widetilde{\mathcal{N}}_i}\Big(\sum_{j=1}^{i-1}\sum_{x_j^k\in\widetilde{\mathcal{N}}_j}(h_i^l/h_j^k)^{1/2}\delta(x_j^k,x_i^l)\Big)|v_i(x_i^l)|^2\\
&\qquad\lesssim\sum_{i=2}^{J}\sum_{x_i^l\in\widetilde{\mathcal{N}}_i}|v_i(x_i^l)|^2,
\end{aligned}\tag{3.16}$$

where we have used the first inequality in (3.4). Similarly, we can show that

$$\sum_{j=1}^{J}\sum_{x_j^k\in\widetilde{\mathcal{N}}_j}\Big(\sum_{i=j+1}^{J}\sum_{x_i^l\in\widetilde{\mathcal{N}}_i, x_i^l\in\Omega_j^k}(h_i^l)^2(h_j^k)^{-1}|v_i(x_i^l)|\Big)^2\lesssim\sum_{i=2}^{J}\sum_{x_i^l\in\widetilde{\mathcal{N}}_i}|v_i(x_i^l)|^2.\tag{3.17}$$

It follows then from (3.11)-(3.17) and Lemma 3.3 that

$$c_0\lesssim 1+\sup_{\|v\|_A=1}\sum_{j=1}^{J}\sum_{x_j^k\in\widetilde{\mathcal{N}}_j}\big|v_j(x_j^k)\big|^2\lesssim 1.$$

This completes the proof of Theorem 2.3. ∎

Remark 3.4. The Xu-Zikatanov identity in Lemma 2.2 is used in [35] to prove the uniform convergence of multigrid methods on uniformly refined meshes. In [38] this identity is used to prove the uniform convergence of MLAT on locally refined meshes. The current work is inspired by the lecture notes of Xu in [38].

Remark 3.5. The assumption that the meshes $\mathcal{M}_j, 0\leqslant j\leqslant J$, are obtained by using the "newest vertex bisection" algorithm can be weakened. Let $\widehat{\mathcal{M}}_j$ be a sequence of meshes of Ω obtained by uniform refinement: edge midpoints in $\widehat{\mathcal{M}}_{j-1}$ are connected by new edges to form $\widehat{\mathcal{M}}_j$. Then it is proved in [39] for the Poisson equation that $\|I-B_JA_J\|_A<\delta$ for some constant $\delta<1$ if the meshes $\mathcal{M}_j, 0\leqslant j\leqslant J$, satisfy that each element $K\in\mathcal{M}_j$ is obtained by refining some element $K'\in\mathcal{M}_{j-1}$ finite number of times so that $h_{K'}\lesssim h_K$, (2.1), and that each element $K\in\mathcal{M}_j$ is included in some element $\widehat{K}\in\widehat{\mathcal{M}}_n$ for some n so that $h_K\eqsim\hat{h}_n$, where h_K is the diameter of $K\in\mathcal{M}_j$ and $\hat{h}_n$ is the mesh size of the uniform mesh $\widehat{\mathcal{M}}_n$. The proof of Theorem 2.3 modifies and extends the ideas in [39].

4 Numerical Results

We make use of the so-called non-recursive implementation of multigrid method (cf. e.g. [22]). Let $\tilde{g}\in\mathbb{R}^{n_j}$ be the vector in $\mathbb{R}^{n_j}$, then the matrix $\widetilde{B}_j\in\mathbb{R}^{n_j\times n_j}$ which corresponds

to the operator $B_j : V_j \to V_j$ in Algorithm 2.1 is defined through the following algorithm.

Algorithm 4.1. (Non-recursive implementation)

$\texttt{rtmp} = \tilde{g}$	$\texttt{btmp} = \widetilde{A}_0^{-1}\texttt{rtmp}$
`for` $i = j : 1$	`for` $i = 1 : j$
$\tilde{v}_i = 0$	$\texttt{btmp} \leftarrow \tilde{v}_i + I_{i-1}^i\texttt{btmp}$
$\tilde{v}_i \leftarrow \tilde{v}_i + \widetilde{R}_i(\texttt{rtmp} - \widetilde{A}_i\tilde{v}_i)$	$\texttt{btmp} \leftarrow \texttt{btmp} + \widetilde{R}_i^t(\tilde{r}_i - \widetilde{A}_i\texttt{btmp})$
$\tilde{r}_i \leftarrow \texttt{rtmp}(ns_i)$	`end for`
$\texttt{rtmp} \leftarrow (I_{i-1}^i)^t(\texttt{rtmp} - \widetilde{A}_i\tilde{v}_i)$	
`end for`	$\widetilde{B}_j\tilde{g} = \texttt{btmp}$

Here $\widetilde{A}_i \in \mathbb{R}^{n_i \times n_i}$ is the stiffness matrix of the discrete problem at the i-th level, $\widetilde{R}_i \in \mathbb{R}^{n_i \times n_i}$ is the local Gauss-Seidel relaxation matrix, and $I_{i-1}^i \in \mathbb{R}^{n_i \times n_{i-1}}$ is the standard prolongation matrix. Let $\widetilde{\mathcal{N}}_i = \{x_i^{k_1}, \cdots, x_i^{k_{\tilde{n}_i}}\}$ be the collection of new nodes and their "immediate" neighboring nodes at level i, and denote by $ns_i = \{k_1, \cdots, k_{\tilde{n}_i}\}$, then $\texttt{rtmp}(ns_i) \in \mathbb{R}^{\tilde{n}_i}$ stands for the collection of the $k_1, \cdots, k_{\tilde{n}_i}$-th components of the vector $\texttt{rtmp}$. $(I_{i-1}^i)^t$ and $\widetilde{R}_i^t$ are the transpose matrix of I_{i-1}^i and $\widetilde{R}_i$, respectively.

We remark that in Algorithm 4.1, all the computations at the i-th level involve only the components corresponding to the nodes in $\widetilde{\mathcal{N}}_i$, all the other components corresponding to the nodes in $\mathcal{N}_i \backslash \widetilde{\mathcal{N}}_i$ remain unchanged. Therefore each iteration to compute $\widetilde{B}_j\tilde{g} \in \mathbb{R}^{n_j}$ for given $\tilde{g} \in \mathbb{R}^{n_j}$ uses only $O(n_j)$ operations.

4.1 Numerical examples

In this subsection we present several numerical examples to illustrate the optimality of the algorithm studied in the paper. In the computation we make use of the adaptive FEM solver in MATLAB PDE toolbox to discretize the equations but replace the linear system solver of MATLAB by our multigrid V-cycle solver whose implementation as shown in Algorithm 4.1. The local error estimator for element K of the adaptive algorithm is defined as follows

$$\eta_T = 0.15 \times \Big[h_K \| R_K \|_{L^2(K)} + \Big(\frac{1}{2} \sum_{e \subset K} h_e \| J_e \|_{L^2(e)}^2 \Big)^{1/2} \Big], \tag{4.1}$$

where h_K is the diameter of the element K, h_e is the length of the side e, R_K and J_e are the element residual and jump residual respectively. Computations are performed on a Pentium 3.4GHz computer with Linux operating system.

The stopping rule of the multigrid iteration is as follows. At the j-th level, let $u_j^{(0)} = u_{j-1}$, the multigrid solution of the previous level, then the multigrid iteration is terminated

when the following relation is satisfied

$$\frac{\| f_j - A_j u_j^{(m)} \|_\infty}{\| f_j - A_j u_j^{(0)} \|_\infty} \leq 10^{-6}.$$

Example 4.1. *The Laplace equation on the L-shaped domain of Figure 3 with the Dirichlet boundary condition so chosen that the true solution is* $r^{2/3}\sin(2\theta/3)$ *in polar coordinates.*

Figure 3 shows a sample of adaptively refined mesh of 1165 interior nodes which is obtained after 14 adaptive iterations. The area of the smallest element is 3.05×10^{-7}. Table 1 shows the number of interior nodes, the energy error $|u - u_h|_{1,\Omega}$ between the true solution u and the discrete solution u_h computed by the multigrid method, and the energy error $|\hat{u}_h - u_h|_{1,\Omega}$ between u_h and the exact discrete solution $\hat{u}_h$ computed by the direct linear system solver on various adaptive levels. It clearly shows that the multigrid solution u_h approximates the true solution well up to the discretization error.

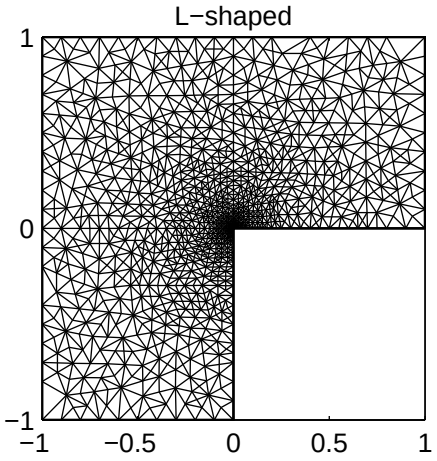

Figure 3: *The adaptively refined mesh of* 1165 *interior nodes after* 14 *adaptive iterations for the Laplace equation on L-shaped domain*

Table 1: *The level of adaptive iterations* J*, the number of interior nodes DOFs, the error* $|u - u_h|_{1,\Omega}$ *between the true solution* u *and the discrete solution* u_h *computed by the multigrid method, and the error* $|\hat{u}_h - u_h|_{1,\Omega}$ *between* u_h *and the exact discrete solution* $\hat{u}_h$ *for the Laplace equation on the L-shaped domain.*

J	14	20	26	32	38	44
DOFs	1165	4689	18688	74329	295989	1181007
$\|u - u_h\|_{1,\Omega}$	3.07e-2	1.53e-2	7.71e-3	3.88e-3	1.94e-3	9.74e-4
$\|\hat{u}_h - u_h\|_{1,\Omega}$	1.70e-9	8.57e-10	3.32e-10	1.81e-10	4.56e-11	1.99e-11

On the finest mesh with $1,181,007$ interior nodes at 44-th level, our multigrid solver solves the discrete problem using CPU time 20.14 seconds. As a comparison, the direct solver provided by MATLAB uses CPU time 138.66 seconds.

Figure 4 shows the reduction factor $\| I - B_J A_J \|_A$ (left) and the CPU time of each multigrid V-cycle iteration (right). Since $I - B_J A_J$ is symmetric with respect to the

bilinear form $A(\cdot,\cdot)$, $\|I - B_J A_J\|_A$ is equal to the maximum eigenvalue of $I - B_J A_J$ which is computed by the power method in this paper. We observe $\|I - B_J A_J\|_A$ is independent of the number of levels J as predicted by our theoretical analysis. We also observe that the CPU time of each multigrid V-cycle iteration is linear in terms of the number of interior nodes used. This clearly indicates the optimality of the algorithm: one multigrid V-cycle iteration at the J-th level requires only $O(n_J)$ operations to reduce the norm the error of the approximate solution of $A_J u_J = f_J$ by a factor independent of n_J.

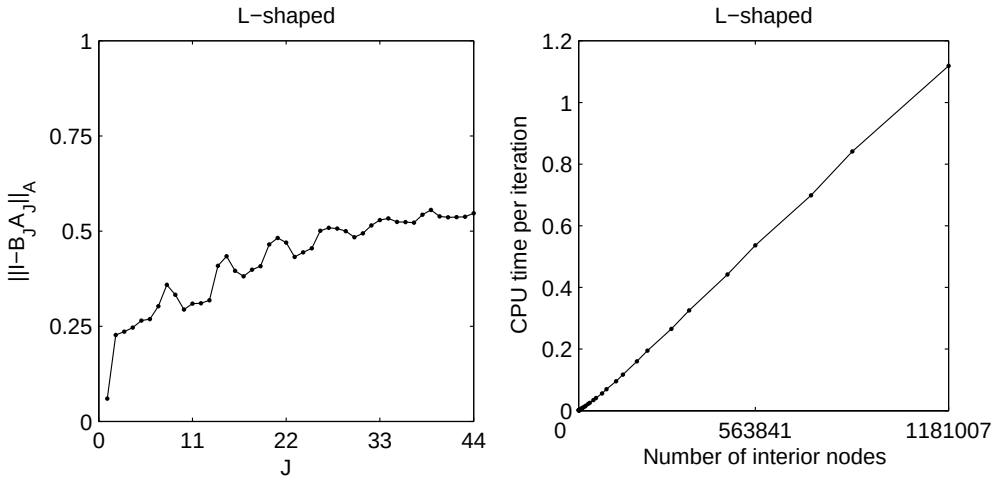

Figure 4: *The reduction factors $\|I - B_J A_J\|_A$ (left) and the CPU time (seconds) of each multigrid V-cycle iteration (right) for the Laplace equation on the L-shaped domain.*

Example 4.2. *Let $\Omega = \{(x_1, x_2) : |x_1| + |x_2| < 1\} \backslash \{(x_1, x_2) : 0 \le x_1 < 1\}$ be the domain with a crack. We consider the Poisson equation*

$$-\Delta u = 1$$

with Dirichlet boundary condition so chosen that the true solution is $r^{1/2}\sin(\theta/2) - \frac{1}{4}r^2$ in polar coordinates.

Figure 5 shows a sample of adaptively refined mesh of 1060 interior nodes which is obtained after 23 adaptive iterations. The area of the smallest element is 5.96×10^{-8}. Table 2 shows the number of interior nodes, the energy error $|u - u_h|_{1,\Omega}$ between the true solution u and the discrete solution u_h computed by the multigrid method, and the energy error $|\hat{u}_h - u_h|_{1,\Omega}$ between u_h and the exact discrete solution $\hat{u}_h$ on various adaptive levels.

On the finest mesh with $1,028,747$ interior nodes at 68-th level, our multigrid solver uses CPU time 12.25 seconds, which is about 8 times faster than the direct solver provided by MATLAB that needs CPU time 114.97 seconds.

Figure 6 shows the reduction factor $\| I - B_J A_J \|_A$ (left) and the CPU time of each multigrid V-cycle iteration (right). Here again we observe $\|I - B_J A_J\|_A$ is independent of the number of levels J and the optimality of the algorithm.

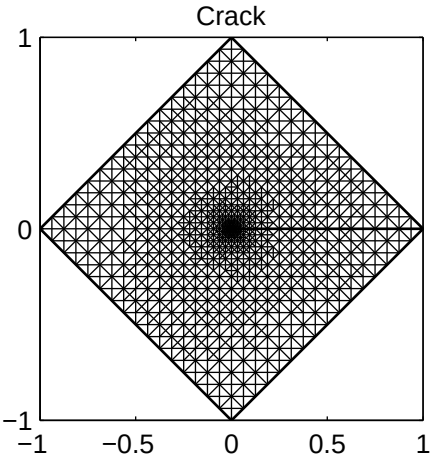

Figure 5: *The adaptively refined mesh of* 1060 *interior nodes after* 23 *adaptive iterations for the Poisson equation on the domain with a crack.*

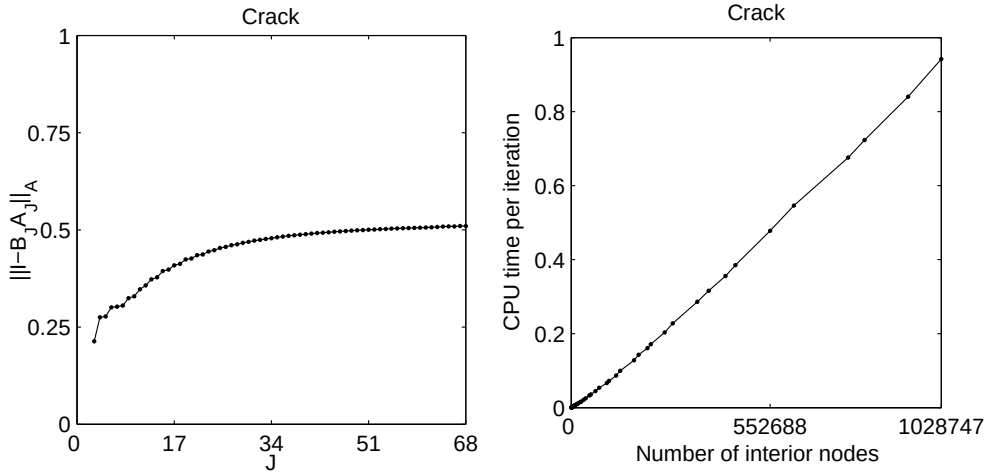

Figure 6: *The reduction factors* $\|I - B_J A_J\|_A$ *(left) and the CPU time (seconds) of each multigrid V-cycle iteration (right) for the Poisson equation on the domain with a crack.*

Table 2: *The level of adaptive iterations J, the number of interior nodes DOFs, the error $|u - u_h|_{1,\Omega}$ between the true solution u and the exact discrete solution u_h computed by the multigrid method, and the error $|\hat{u}_h - u_h|_{1,\Omega}$ between u_h and the exact discrete solution $\hat{u}_h$ for the Poisson equation on the domain with a crack.*

J	23	31	39	47	57	68
DOFs	1060	4299	17227	66780	259371	1028747
$\|u - u_h\|_1$	5.87e-2	2.91e-2	1.46e-2	7.39e-3	3.75e-3	1.88e-3
$\|\hat{u}_h - u_h\|_1$	2.12e-8	1.09e-8	4.14e-9	6.96e-10	2.30e-10	1.09e-10

Example 4.3. *We consider the approximation of Green function on the unit circle*

$$-\Delta u = \delta(x_1, x_2).$$

Figure 7 shows a sample of adaptively refined mesh of $11,158$ interior nodes which is obtained after 99 adaptive iterations. The area of the smallest element is about 3.38×10^{-33}.

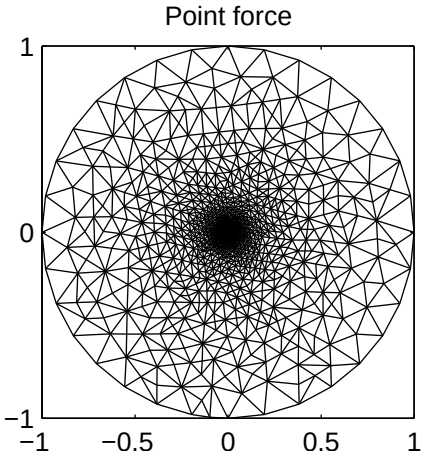

Figure 7: *The adaptively refined mesh of $11,158$ interior nodes after 99 adaptive iterations for the approximation of Green function on the unit circle.*

Figure 8 shows the reduction factor $\| I - B_J A_J \|_A$ (left) and the CPU time of each multigrid V-cycle iteration (right). Here again we observe $\|I - B_J A_J\|_A$ is independent of the number of levels J and the optimality of the algorithm.

4.2 Influence of the number of smoothing iterations

It is well-known that, for a multigrid V-cycle method on uniformly refined meshes with full relaxations, the reduction factor approaches zero when the number of smoothing iterations goes to infinity. This property is usually not true for a multigrid V-cycle method on adaptively refined meshes with local relaxations, because one usually can not obtain the exact solution of the finite element linear systems in the smoothing procedure even if local relaxations are performed infinite times. Recall that infinite times of full relaxations im-

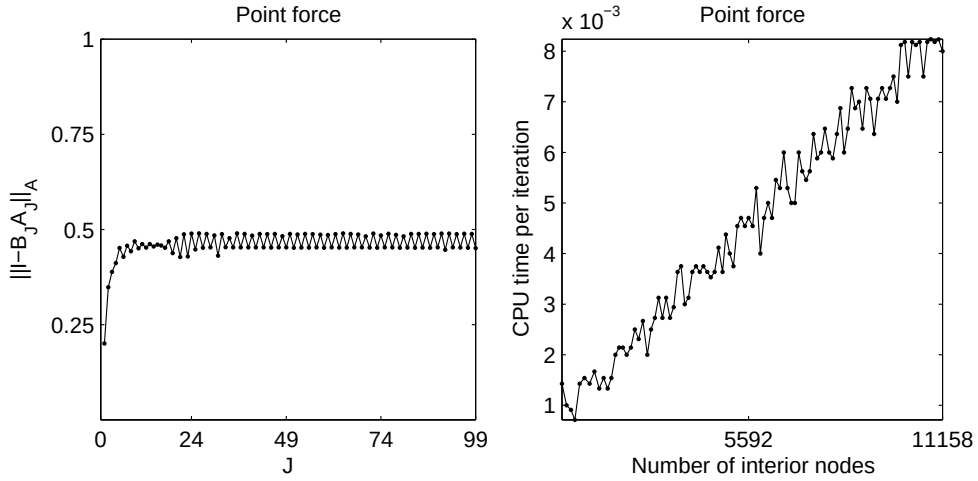

Figure 8: *The reduction factors $\|I - B_J A_J\|_A$ (left) and the CPU time (seconds) of each multigrid V-cycle iteration (right) for the approximation of Green function on the unit circle.*

plies we get the exact solution after smoothing procedure and hence the multigrid method converges in one step.

Figure 9 shows the reduction factors $\|I - B_J A_J\|_A$ for our algorithm (left) and the reduction factors $\left\|I - \bar{B}_J A_J\right\|_A$ for the "local" multigrid V-cycle algorithm (right) when 1, 2, and 100 local relaxation sweeps are carried out for the discrete problem in Example 4.1, respectively. Table 3 and 4 show the CPU times of our multigrid V-cycle solver and that

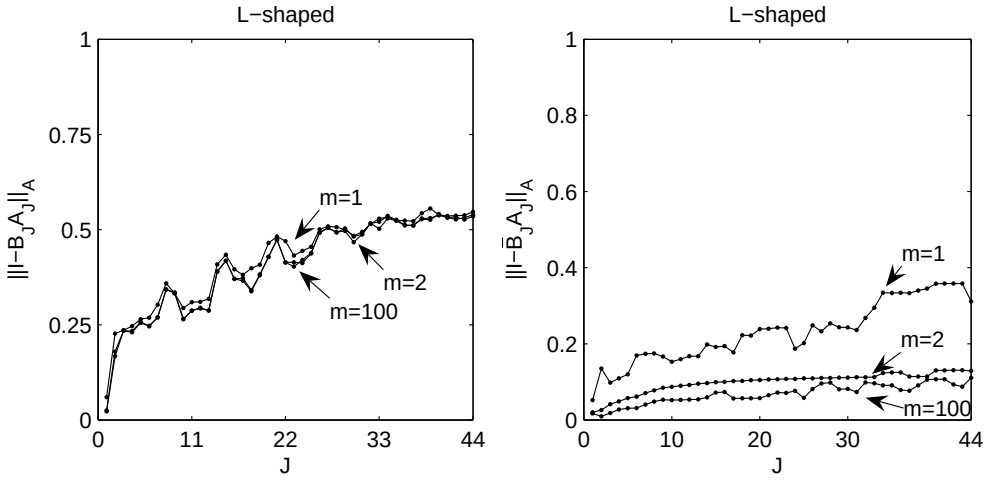

Figure 9: *The comparison of the reduction factors for our algorithm (left) and that for the "local" multigrid V-cycle algorithm (right) when 1, 2, and 100 local relaxation sweeps are performed for the Laplace equation on the L-shaped domain, respectively. Here m is the number local relaxation sweeps.*

of the "local" multigrid V-cycle solver on various adaptive levels with $1, 2$, and 3 local relaxations sweeps for the discrete problem in Example 4.1 and Example 4.2, respectively.

Table 3: *The level of adaptive iterations J, the number of interior nodes DOFs, The CPU time t_J^m of our algorithm and the CPU time t_J^{cm} of the "local" multigrid V-cycle algorithm at level J with m times of relaxations for the Laplace equation on the L-shaped domain.*

J	14	20	26	32	38	44
DOFs	1165	4689	18688	74329	295989	1181007
t_J^1 (seconds)	0.02	0.06	0.22	0.95	4.78	20.14
t_J^2 (seconds)	0.02	0.07	0.25	1.11	5.61	23.68
t_J^3 (seconds)	0.03	0.07	0.30	1.33	6.98	29.45
t_J^{c1} (seconds)	0.01	0.04	0.16	0.79	3.57	14.69
t_J^{c2} (seconds)	0.01	0.04	0.12	0.63	2.96	12.14
t_J^{c3} (seconds)	0.01	0.03	0.15	0.78	3.74	15.36

We observe that increasing the number of smoothing iterations helps little to improve the efficiency of both algorithms. We remark that our algorithm is cheaper per cycle since it restricts the smoothing to a smaller set of nodes. But the convergence rate of our algorithm may be slower than that of the "local" multigrid V-cycle algorithm. For both examples,

Table 4: *The level of adaptive iterations J, the number of interior nodes DOFs, The CPU time t_J^m of our algorithm and the CPU time t_J^{cm} of the "local" multigrid V-cycle algorithm at level J with m times of relaxations for the Poisson equation on the domain with a crack.*

J	23	31	39	47	57	68
DOFs	1060	4299	17227	66780	259371	1028747
t_J^1 (seconds)	0.01	0.03	0.14	0.67	3.05	12.25
t_J^2 (seconds)	0.01	0.04	0.17	0.73	3.45	14.91
t_J^3 (seconds)	0.01	0.05	0.21	0.88	4.24	18.49
t_J^{c1} (seconds)	0.00	0.02	0.13	0.54	2.52	10.31
t_J^{c2} (seconds)	0.00	0.03	0.13	0.56	2.67	10.59
t_J^{c3} (seconds)	0.01	0.03	0.15	0.60	2.89	13.42

our algorithm is a little slower than the "local" multigrid one.

5 Concluding remarks

In this paper we have shown the uniform convergence of our multigrid V-cycle algorithm (and the "local" multigrid V-cycle algorithm [4]) with local Gauss-Seidel relaxation performed only on new nodes and their "immediate" neighbors (their "conventional" neighbors) under the practical assumption that the adaptive meshes are generated by using the "newest vertex bisection" algorithm. This leads to an algorithm of optimal complexity for solving the linear system of equations resulting from the discretization of (1.1) by adaptive finite element methods at each level k. We remark that the analysis in the paper is applicable to the two dimensional case only. The extension of the analysis to the three dimensional case is an interesting open problem.

In practical applications, multigrid methods are often applied to solve partial differential equations by the way of full multigrid (FMG) methods [40]. To conclude this paper, we show that Theorem 2.3 implies that discretization error can be achieved in one FMG cycle based on the argument in [36]. Recall that $u_k \in X_k$ is the k^{th} level finite element solution satisfying $A_k u_k = f_k$. It is proved in [2] that for properly designed adaptive finite element procedures, there exist positive constants C_0 and $\beta < 1$, depending only the given data and the initial grid, such that

$$\|u - u_k\|_A \leqslant C_0 \beta^k, \quad k \geqslant 0. \tag{5.1}$$

The FMG method (cf. e.g. [18]) is based on the following two observations: (1) $u_{k-1} \in X_{k-1} \subset X_k$ is closed to $u_k \in X_k$ and hence can be used as an initial guess for an iterative scheme for solving u_k; and (2) Each u_k can be solved within its truncation error by a multigrid iterative scheme.

Algorithm 5.1. (FMG method)

(i) *For $k = 0$, $\hat{u}_0 = A_0^{-1} f_0$*

(ii) *For $k \geqslant 1$, let $\hat{u}_k = \hat{u}_{k-1}$, and iterate $\hat{u}_k \leftarrow \hat{u}_k + B_k(f_k - A_k \hat{u}_k)$ for l times, where B_k is defined in Algorithm 2.1.*

Now let the integer l so chosen that $\delta^l < \beta$, then by Theorem 2.3 we have

$$\begin{aligned} \|u_k - \hat{u}_k\|_A \leqslant \delta^l \|u_k - \hat{u}_{k-1}\|_A &\leqslant \delta^l \|u_k - u_{k-1}\|_A + \delta^l \|u_{k-1} - \hat{u}_{k-1}\|_A \\ &\leqslant \delta^l \|u - u_{k-1}\|_A + \delta^l \|u_{k-1} - \hat{u}_{k-1}\|_A \\ &\leqslant C_0 \delta^l \beta^{k-1} + \delta^l \|u_{k-1} - \hat{u}_{k-1}\|_A . \end{aligned}$$

Since $\|u_0 - \hat{u}_0\|_A = 0$, we conclude that

$$\|u_k - \hat{u}_k\|_A \leqslant C_0 \sum_{n=1}^{k} (\delta^l)^n \beta^{k-n} \leqslant \frac{\delta^l}{\beta - \delta^l} C_0 \beta^k.$$

This estimate implies that the discretization error can be achieved with one FMG cycle on adaptively refined finite element meshes. It is easy to see that the work involved in the FMG is $O(\sum_{j=0}^{k} n_j)$ which is optimal in the sense that the work is of the same order as the work involved in computing a posteriori error estimators in an adaptive finite element algorithm. However, the practical application of the FMG algorithm has the difficulty of choosing the number of iterations l in Algorithm 5.1 which depends on the unknown constant β in (5.1). How to adaptively choose the integer l in Algorithm 5.1 is an interesting topic for future research. We refer to [18] for more discussion on FMG in the context of MLAT methods.

Acknowledgements The authors would like to thank Jinchao Xu for many insightful discussions on the multigrid methods. The authors would also like to thank the referees of the paper for their constructive comments which improved the paper. The first author was supported in part by China NSF under the grant 10401016 and by the National Basic Research Program under the Grant 2005CB321701. The second author was supported in part by China NSF under the grant 10025102 and by China MOS under the grant G1999032802.

References

[1] Babuška, I., Rheinboldt, C., Error estimates for adaptive finite element computations, *SIAM J. Numer. Anal.,* 1978, 15: 736–754.

[2] Morin, P., Nochetto, R.H., Siebert, K.G., Data oscillation and convergence of adaptive FEM, *SIAM J. Numer. Anal.,* 2000, 38: 466–488.

[3] Chen, Z., Dai, S., On the efficiency of adaptive finite element methods for elliptic problems with discontinuous coefficients, *SIAM J. Sci. Comput.,* 2002, 24: 443–462.

[4] Mitchell, W.F., Optimal multilevel iterative methods for adaptive grids, *SIAM J. Sci. Stat. Comput.,* 1992, 13: 146–167.

[5] Bank, R. E., Dupont, T. F., Yserentant, H., The hierarchical basis multigrid method, *Numer. Math.,* 1988, 52: 427–458.

[6] Bänsch, E., Local mesh refinement in 2 and 3 dimensions, *Impact of Computing in Science and Engineering,* 1991, 3: 181–191.

[7] Rivara, M. C., Algorithms for refining triangular grids suitable for adaptive and multigrid techniques, *J. Numer. Meth. Engrg.,* 1984, 20: 745–756.

[8] Stals, L., Implementation of multigrid on parallel machines using adaptive finite element methods, in: Ninth International Conference on Domain Decomposition Methods, P. E. Bjorstad and M. S. Espedal and D. E. Keyes, eds., Bergen: DDM.org, 1997, 488–496.

[9] Chen, Z., Dai, S., Adaptive Galerkin methods with error control for a dynamical Ginzburg-Landau model in superconductivity, *SIAM J. Numer. Anal.,* 2001, 38: 1961–1985.

[10] Chen, Z., Jia, F., An adaptive finite element algorithm with reliable and efficient error control for linear parabolic problems, *Math. Comp.,* 2004, 73: 1163–1197.

[11] Chen, Z., Nochetto, R.H., Schmidt, A., A characteristic Galerkin method with adaptive error control for continuous casting problem, *Comput. Methods Appl. Mech. Engrg.,* 2000, 189: 249–276.

[12] Chen, Z., Wu, H., An adaptive finite element method with perfectly matched absorbing layers for the wave scattering by periodic structures, *SIAM J. Numer. Anal.,* 2003, 41: 799–826.

[13] Schmidt, A., Siebert, K.G., ALBERT: An adaptive hierarchical finite element toolbox, IAM, University of Freiburg, 2000, http://www.mathematik.uni-freiburg.de/IAM/Research/projectsdz/albert.

[14] Bramble, J.H., Ewing, R.E., Pasciak, J.E., et al, A preconditioneding technique for the efficient solution of problems with local grid refinement, *Comput. Methods Appl. Mech. Engrg.,* 1988, 67: 149–159

[15] McCormick, S., Fast adaptive composite grid (FAC) methods, in: Defect Correction Methods: Theory and Applications, K. Böhmer and H.J. Stetter, eds., Computing Supplementum 5, Wien: Sringer Verlag, 1984, 115–121.

[16] McCormick, S., Thomas, J., The fast adaptive composite grid (FAC) method for elliptic equations, *Math. Comp.,* 1986 46: 439–456.

[17] Widlund, O.B., Optimal iterative refinement methods, in: Domain Decomposition Methods, T. Chan, R. Glowinski, J. Périaux and O. Widlund, eds., SIAM, Philadelphia, 1989: 114–125.

[18] Bai, D., Brandt, A., Local mesh refinement multilevel techniques, *SIAM J. Sci. Stat. Comput.,* 1987, 8: 109–134.

[19] Bramble, J.H., Pasicak, J.E., New estimates for multigrid algorithms including the V-cycle, *Math. Comp.,* 1993, 60: 447–471.

[20] Bramble, J.H., Pasicak, J.E., Wang, J., et al, Convergence estimates for product iterative methods with applications to domain decomposition, *Math. Comp.,* 1991, 57: 23–45.

[21] Oswald, P., Multilevel Finite Element Approximation, Theory and Applications, Stuttgart: Teubner Verlag, 1994.

[22] Xu, J., An introduction to multigrid convergence theory, in: Iterative Methods in Scientific Computing, R. Chan, T. Chan and G. Golub, eds., Singapore: Springer Verlag, 1997.

[23] Ainsworth, M., Mclean, W., Multilevel diagonal scaling preconditioners for boundary element equations on locally refined meshes, *Numer. Math.,* 2003, 93: 387–413.

[24] Aksoylu, B., Holst, M., An odyssey into local refinement and multilevel preconditioning I: Optimality of the BPX preconditioned, Institute for Computational Engineering and Sciences, The University of Texas at Austin, ICES Technical Report 05-03, 2005.

[25] Dahmen, W., Kunoth, A., Multilevel preconditioning, *Numer. Math.,* 1992, 63: 315–344.

[26] Oswald, P., Multilevel solvers for elliptic problems on domains, in: Multiscale Wavelet Methods for PDEs, W. Dahmen, A.J. Kurdila and P. Oswald, eds., Academic Press, 1996, 3–58.

[27] Bank, R.E., Dupont, T., An optimal order process for solving elliptic finite element equations, *Math. Comp.,* 1981, 36: 35–51.

[28] Bastian, P., Wittum, G., On robust and adaptive multigrid methods, in: Multigrid Methods IV, Proceedings of the Fourth European Multigrid Conference, Amsterdam, July 6–9, 1993, P. W. Hemker and P. Wesseling, eds., Basel: Birkhäuser, 1994, 1–17.

[29] Bornemann, F. A., Yserentant, H., A basic norm equivalence for the theory of multilevel methods, *Numer. Math.,* 1993, 64: 455–476.

[30] Brandt, A., Multilevel adaptive finite element methods I: Variational problems, in: Special Topics of Applied Mathematics, J. Frehes and D. Pallaschke and U. Trotenberg, eds., Amsterdam: North-Holland, 1979, 91–128.

[31] Griebel, M., Oswald, P., On the abstract theory of additive and multiplicative Schwarz algorithms, *Numer. Math.,* 1995, 70: 163–180.

[32] Rüde, U., Mathematical and Computational Techniques for Multilevel Adaptive Method, Frontiers in Applied Mathematics 13, SIAM, Philadelphia, 1993.

[33] Yserentant, H., Old and new convergence proofs for multigrid methods, *Acta Numerica,* 1993, 3: 285–326.

[34] Xu, J., Iterative Methods by Space Decomposition and Subspace Correction, *SIAM Review,* 1992, 34: 581–613.

[35] Xu, J., Zikatanov, L., The method of alternating projections and the method of subspace corrections in Hilbert space, *J. Amer. Math. Soc.,* 2002, 15: 573–597.

[36] Bramble, J.H., Multigrid Methods, Pitman Research Notes in Mathematical Sciences 294, Essex: Longman, 1993.

[37] Scott, L.R., Zhang, S., Finite element interpolation of nonsmooth functions satisfying boundary conditions, *Math. Comp.,* 1990, 54: 483–493.

[38] Xu, J., Multigrid convergence theory, Lecture Notes of the 2001 summer program on Multigrid Methods and Computational Electromagnetics in the Morningside Center of Mathematics, Chinese Academy of Sciences, Beijing, China.

[39] Wu, H., Uniform Convergence of a multigrid V-cycle for adaptive linear finite element methods, Postdoctoral Report, Institute of Computational Mathematics, Chinese Academy of Sciences, Beijing, 2002.

[40] Brandt, A., Multigrid Techniques: 1984 Guide, with applications to fluid dynamics, monograph, Weizmann Institute of Science, Rehovot, Israel, 1984.