Numer. Math. (2004) 97: 219–268
Digital Object Identifier (DOI) 10.1007/s00211-003-0492-7

Numerische Mathematik

Adaptive Finite Element Methods with convergence rates

Peter Binev[1], **Wolfgang Dahmen**[2], **Ron DeVore**[1]

[1] Department of Mathematics, University of South Carolina, Columbia, SC 29208, USA; e-mail:{binev, devore}@math.sc.edu
[2] Institute für Geometrie und Praktische Mathematik, RWTH Aachen, Templergarben 55, 52056 Aachen, Germany; e-mail: dahmen@igpm.rwth-aachen.de

Received June 10, 2002 / Revised version received April 29, 2003 /
Published online January 8, 2004 –

Summary. Adaptive Finite Element Methods for numerically solving elliptic equations are used often in practice. Only recently [12], [17] have these methods been shown to converge. However, this convergence analysis says nothing about the rates of convergence of these methods and therefore does, in principle, not guarantee yet any numerical advantages of adaptive strategies versus non-adaptive strategies. The present paper modifies the adaptive method of Morin, Nochetto, and Siebert [17] for solving the Laplace equation with piecewise linear elements on domains in $\mathbb{R}^2$ by adding a coarsening step and proves that this new method has certain optimal convergence rates in the energy norm (which is equivalent to the H^1 norm). Namely, it is shown that whenever $s > 0$ and the solution u is such that for each $n \geq 1$, it can be approximated to accuracy $O(n^{-s})$ in the energy norm by a continuous, piecewise linear function on a triangulation with n cells (using complete knowledge of u), then the adaptive algorithm constructs an approximation of the same type with the same asymptotic accuracy while using only information gained during the computational process. Moreover, the number of arithmetic computations in the proposed method is also of order $O(n)$ for

This work has been supported by the Office of Naval Research Contract Nr. N00014-03-10051, the Army Research Office Contract Nr. DAAD 19-02-1-0028, the National Science Foundation Grants DMS 0221642, DMS 9872890 the Deutsche Forschungsgemeinschaft grant SFB 401, the European Community's Human Potential Programme under Contract HPRN-CT-2002-00286, "Breaking Complexity".
Correspondence to: R. DeVore

each $n \geq 1$. The construction and analysis of this adaptive method relies on the theory of nonlinear approximation.

Mathematics Subject Clasification (2000): 65N30, 65Y20, 65N12, 65N50, 68W40, 68W25.

1 Introduction

Adaptive methods are frequently used to numerically compute solutions to elliptic equations. While these methods have been shown to be very successful computationally, the theory describing the advantages of such methods over their non-adaptive counterparts is still not complete. For example, only recently [12], [17] have there even been proofs of convergence of such methods. These proofs of convergence still do not prove any guaranteed advantage of these adaptive methods since there is no analysis of their *rate of convergence* in terms of the number of degrees of freedom or the number of computations.

Recently, an analysis of rates of convergence for wavelet based adaptive methods was given in [6],[7]. These papers derive an adaptive wavelet based algorithm for solving elliptic problems and show that this algorithm has optimal efficiency in the sense that if the solution u can be approximated (using complete knowledge of u) in the energy norm by an n-term wavelet expansion to accuracy $O(n^{-s}), n \to \infty$, then the adaptive method will do the same using *only* knowledge of u gained through the adaptive iteration. Wavelet methods vary from their FEM counterparts in that they can be viewed as solving linear systems that are finite sections of one fixed infinite dimensional matrix problem whose solution gives the wavelet coefficients of u.

The theoretical foundation of Adaptive Finite Element Methods (AFEM) is less satisfying. There is no known algorithm with a proven rate of convergence save for the univariate case [4]. The purpose of the present paper is to give an AFEM and prove convergence rates for this method which are the analogue of the wavelet case. Our algorithm is not much different from existing adaptive methods based on bulk chasing of a posteriori error estimators. The one main difference is the utilization of a coarsening strategy. We should mention that coarsening also played an important role in the analysis of adaptive wavelet methods. However, in the practical implementation of the adaptive wavelet methods, for many problems, coarsening is not needed. The same may be the case for AFEM.

We primarily view the present paper as a contribution to the theoretical analysis of AFEM rather than the construction of an adaptive method that outperforms other adaptive methods in practice. In particular, we wish to clarify whether a-posteriori information can lead to an adaptive algorithm that exhibits asymptotically optimal performance. In spite of the theoretical emphasis it should not be excluded that some of the ideas of the present

paper may be useful in practice. One of these tools is the theory of nonlinear approximation by piecewise polynomials. Since adaptive methods are a form of nonlinear approximation, this theory will on the one hand help us to provide a benchmark for measuring the success of adaptive methods, and on the other hand, provide an effective implementation for the coarsening (see § 4.5).

Adaptive Finite Element Methods have several complications that make their analysis more cumbersome. These include the need for graded meshes, the problem of hanging nodes, and the analysis of a-posteriori error estimators. If not for these complications, the analysis in this paper would be considerably simplified. In order to present the ideas of this paper in their simplest form, we shall try to minimize these obstacles. In particular, we shall restrict ourselves to the Poisson problem

$$-\Delta u = f \quad \text{in } \Omega, \quad u = 0 \text{ on } \partial\Omega, \tag{1.1}$$

where Ω is a polygonal domain in $\mathbb{R}^2$ and $\partial\Omega$ is its boundary. We shall also consider only approximations of the solution u by piecewise linear elements using a very specific adaptive refinement strategy (called newest vertex bisection) well-known in the FEM literature. In this way, the essentials of our arguments will be clear and we can also call on several known results concerning a-posteriori error estimates that can be found in the literature.

We conclude this introduction by briefly describing the structure of this paper. In §2 we discuss the general form of adaptive Finite Element Methods which is marking, subdivision, and completion (to remove hanging nodes). We then consider in detail the structure of subdivision using newest vertex bisection. We shall introduce a simple labelling for edges that will facilitate the analysis of this type of subdivision. The main result of this section is Theorem 2.4 which bounds the number of cells in the completion process by the number of marked cells. This bound is vital in proving optimal convergence rates. §3 recalls Galerkin approximations.

In §4 we study adaptive approximation by piecewise linear functions on adaptively generated triangulations. The spirit of this section is to understand how to construct good adaptive approximations to a *known* function w. In particular, we introduce the algorithm of [2] which will be used heavily in our adaptive Finite Element Algorithm for solving (1.1). Namely, it is used to approximate the right hand side f (see §4.4) and for our coarsening strategy (see §4.5). We also discuss in this section what are optimal approximation rates to a known function. This will provide a benchmark for our analysis of adaptive Finite Element algorithms.

In §5, we recall the adaptive FEM method in [17] (we call it the **MNS** algorithm), and record some of the proven facts about this method that will be used in the present paper. This includes their introduction of local a-posteriori error estimators and their analysis of how these estimators can be used to

bound global errors. In §6, we make some minor modifications of the **MNS** algorithm described in §5.

In §7, we describe the main ingredients of our new adaptive algorithm. We show its optimal rates of convergence in §8. We conclude the paper with an appendix which discusses the smoothness conditions that govern rates of convergence by adaptive methods. These results are not important for the analysis in the present paper but may be of interest to the reader.

2 Newest vertex bisection and completion

This section has three purposes. The first is to set forward some of the notation we shall use in this paper. The second is to introduce the main form of adaptive Finite Element Methods which is marking, subdividing, and completing. The third is to introduce and analyze the particular form of subdivision we shall use in this paper, the so called *newest vertex bisection* method.

Let Ω be a polygonal domain in $\mathbb{R}^2$. We shall use P to denote a partition of Ω into triangular cells Δ. This means that $\Omega = \cup_{\Delta\in P}\Delta$ and any two $\Delta, \Delta' \in P$ satisfy $\text{meas}(\Delta \cap \Delta') = 0$ where here and later in this paper meas denotes the Euclidean measure in $\mathbb{R}^2$. Given such a partition, we let $\mathcal{S}_P$ denote the space of continuous, piecewise linear functions subordinate to P which vanish on $\partial\Omega$. A function S is in $\mathcal{S}_P$ if and only if S is a linear function on each $\Delta \in P$, S is continuous on Ω, and S vanishes on $\partial\Omega$, i.e. $\mathcal{S}_P \subset H_0^1(\Omega)$.

We denote by $\mathcal{E}_P$ the set of edges of P and by $\dot{\mathcal{E}}_P$ the set of *interior edges*. Thus, $E \in \dot{\mathcal{E}}_P$ means that E is an edge of some $\Delta \in P$ and that the interior of E is in the interior of Ω. All other edges are called *boundary edges*. We also denote by $\mathcal{V}_P$ the set of all vertices v of P and by $\dot{\mathcal{V}}_P$ the set of *interior vertices*. Thus, $v \in \dot{\mathcal{V}}_P$ means that v is a vertex of one of the $\Delta \in P$ and v is in the interior of Ω. All other vertices are called *boundary vertices*.

There are two special conditions that we shall impose on a partition P that are important in Finite Element constructions. First, we say that a partition satisfies a *minimal angle condition* if for each $\Delta \in P$ all of its angles are $\geq a_0$ for some positive number a_0. Second, we shall require a partition P to be *conforming* which means that the intersection of any two cells is either empty or a common edge or a common vertex. A *family* of partitions whose elements are all conforming and which satisfy a minimal angle condition with respect to a common constant $a_0 > 0$, is called *admissible*.

The uniform minimal angle condition implies that for each cell Δ in any partition P from an admissible family $\mathcal{P}$, the ratio of the radii of the smallest circumscribed and the largest inscribed circle of Δ is uniformly bounded independent of Δ and P. This is sometimes referred to as *local quasi-uniformity*

or *shape regularity*. In particular, this implies the existence of a constant $\hat{C} = \hat{C}(\mathcal{P})$ such that

$$1 \leq \operatorname{diam}(\Delta)^2/|\Delta| \leq \hat{C} \quad \text{for all } \Delta \in P,\ P \in \mathcal{P}, \tag{2.1}$$

where $|\Delta| = \operatorname{meas}(\Delta)$ denotes the Lebesgue measure of Δ. Moreover, there exists a constant $G_0 = G_0(\mathcal{P})$ such that for any two cells $\Delta, \Delta' \in P$, for which $\Delta \cap \Delta' \neq \emptyset$, we have

$$\operatorname{diam}(\Delta) \leq G_0 \operatorname{diam}(\Delta'). \tag{2.2}$$

Locally quasi-uniform partitions allow one to control the global error in approximating a function by local errors. Typical estimates in FEM depend on a_0 and deteriorate if a_0 is small.

Conformity favors common finite element data structures by conveniently relating local and global stiffness matrices since global basis functions are composed of the local shape functions on each element in a simple way. In our specific context the global basis functions will be the Courant "hat functions" (nodal functions) ϕ_v, $v \in \mathcal{V}_P$. The function ϕ_v is the unique element in $\mathcal{S}_P$ which is one at v and is zero at all other vertices in $\dot{\mathcal{V}}_P$. The nodal basis functions are locally supported on the union of all triangles which share v as a vertex. Moreover, one can construct locally supported dual functionals with the same supports consisting of (discontinuous) piecewise linear functions thereby giving rise to local linear projectors on $\mathcal{S}_P$ that are bounded in $H^s(\Omega)$ for $s \leq 1$ (in fact even beyond 1). Again, it is important for estimating errors that for any partition P from an admissible family, each basis function is overlapped by a uniformly bounded number of other basis functions since the *valence* of the vertices, i.e. the number of edges emanating from a given vertex, remains uniformly bounded.

The adaptive procedures we shall consider in this paper will generate a family of partitions which is admissible.

A typical AFEM generates a sequence of partitions $P_0, P_1, \ldots, P_n$ by using rules for subdividing triangles. Given the partition P_k, the algorithm marks certain of the triangular cells $\Delta \in P_k$ for subdivision. We shall denote by $\mathcal{M}_k$ the collection of marked cells. These marked cells are subdivided using certain subdivision rules. This process, however, creates *hanging nodes*. We say that $v \in \mathcal{V}_P$ is a hanging node for $\Delta \in P$ if v appears in the interior of one of the sides of Δ. Since hanging nodes obviously violate conformity, in a second step a certain collection $\mathcal{M}_k'$ of additional cells are subdivided in order to guarantee that the resulting partition P_{k+1} is admissible. The partition P_n is the final admissible partition associated with this application of the adaptive algorithm.

2.1 Newest vertex bisection

We shall restrict ourselves in this paper to a very specific method of subdivision known as *newest vertex bisection*. We shall call on certain properties of this method of subdivision in what follows. We could not find some of these properties (or proofs of these properties) in the FEM literature and therefore our discussion and development of newest vertex bisection will be somewhat lengthy. The book of Verfürth [22] and the research article of Mitchell [16] describe this subdivision method and give some of its properties.

Given an initial partition P_0 of Ω, to each $\Delta \in P_0$, we assign exactly one of its vertices $v(\Delta)$ as the *newest vertex* for that cell. This initial labelling can be made in an arbitrary way. The edge in Δ opposite to $v(\Delta)$ will be denoted by $E(\Delta)$. In Figure 1 the newest vertex assigned in any triangle of P_0 is indicated by an arrow pointing to $E(\Delta)$.

Each triangular cell that arises in the adaptive process will also have exactly one of its vertices designated as a newest vertex. If this cell is to be subdivided then the subdivision is a simple bisection done by connecting the newest vertex and the side $E(\Delta)$ opposite. Thus the cell produces two new cells and their newest vertex (assigned to each new triangular cell) is by definition the midpoint of $E(\Delta)$.

The partitions which arise when using newest vertex bisection satisfy a uniform minimal angle condition. This is established by showing that all triangles that arise in newest vertex bisection can be classified into a set of similarity classes depending only on the initial partition P_0 (see Mitchell [16]). Also note that if a partition P is created by a sequence of newest vertex bisections and if P has no hanging nodes, then it is conforming. Thus, it is admissible. We shall show in the next subsection how any given partition generated by newest vertex bisection can be completed to a partition with no hanging nodes by subdividing certain other triangular cells. This process is called *completion*. Furthermore, we shall bound the number of additional subdivisions necessary to remove hanging nodes. But first we want

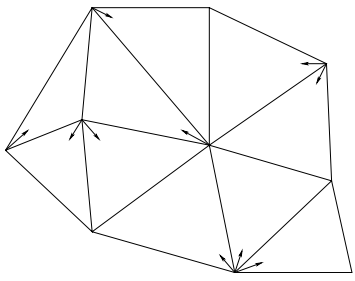

Fig. 1. Assignment of newest vertices in P_0

to examine another important property of newest vertex bisection which is its tree structure.

We can represent newest vertex bisection subdivision by an infinite binary tree T_* (which we call the *master tree*). The master tree T_* consists of all triangular cells which can be obtained by a sequence of subdivisions. The roots of the master tree are the triangular cells in P_0. When a cell Δ is subdivided, it produces two new cells which are called the children of Δ and Δ is their parent. It is very important to note that, no matter how a cell arises in a subdivision process, its associated newest vertex is unique and only depends on the initial assignment of newest vertices in P_0. This means that the children of Δ are uniquely determined and do not depend on how Δ arose in the subdivision process, i.e., it does not depend on the preceding sequence of subdivisions. The reason for this is that any subdivision only assigns newest vertices for the new triangular cells produced by the subdivision and does not alter any previous assignment. It follows that T_* is unique and does not depend at all on the order of subdivisions.

The *generation* of a triangular cell Δ is the number $g(\Delta)$ of ancestors it has in the master tree. Thus cells in P_0 have generation 0, their children have generation 1 and so on. The generation of a cell is also the number of subdivisions necessary to create this cell from its corresponding root cell in P_0.

A *subtree* $T \subset T_*$ is a collection of triangular cells $\Delta \in T_*$ with the following two properties: (i) whenever $\Delta \in T$ then its sibling is also in the tree; (ii) when $\Delta \subset \Delta'$ are both in the tree then each triangular cell $\bar{\Delta} \in T_*$ with $\Delta \subset \bar{\Delta} \subset \Delta'$ is also in T. The roots of T are all the cells $\Delta \in T$ whose parents are not in T. We say that T is *proper* if it has the same roots as T_*, i.e., it contains all $\Delta \in P_0$.

If $T \subset T_*$ is a finite subtree, we say $\Delta \in T$ is a *leaf* of T if T contains none of the children of Δ. We denote by $\mathcal{L}(T)$ the collection of all leaves of T.

For a proper subtree T, we define $N(T)$ to be the number of subdivisions made to produce T.

Any partition $P = P_n$ which is obtained by the application of an adaptive procedure based on newest vertex bisection (such as the algorithms we consider in this paper) can be associated to a proper subtree $T = T(P)$ of T_* consisting of all triangular cells that were created during the algorithm, i.e. all of the cells in $P_0, \ldots, P_n$. The set of leaves $\mathcal{L}(T)$ form the final partition $P = P_n$.

We shall say that $T = T(P)$ is admissible if P is admissible. We denote the class of all proper trees by $\mathcal{T}$ and all admissible trees by $\mathcal{T}^a$. We also let $\mathcal{T}_n$ be the set of all proper trees T with $N(T) = n$ and by $\mathcal{T}_n^a$ the corresponding class of admissible trees from $\mathcal{T}_n$. We denote by $\mathcal{P}$ the class of all

partitions P that can be generated by newest vertex bisection and by $\mathcal{P}^a$ the set of all admissible partitions. Similarly, $\mathcal{P}_n$ and $\mathcal{P}_n^a$ are the subclasses of those partitions that are obtained from P_0 by using n subdivisions. There is a precise identification between $\mathcal{P}_n$ and $\mathcal{T}_n$. Any $P \in \mathcal{P}_n$ can be given by a tree, i.e. $P = P(T)$ for some $T \in \mathcal{T}_n$. Conversely any $T \in \mathcal{T}_n$ determines a $P = P(T)$ in $\mathcal{P}_n$. The same can be said about admissible partitions and trees.

2.2 Completion of subdivision

The adaptive algorithms we consider in this paper will be of the following type. We begin with P_0 and mark certain cells in P_0 for subdivision. After doing these subdivisions we arrive at the partition P_1'. This partition is not necessarily admissible and so we shall make some additional subdivisions which will complete P_1' to an admissible partition P_1. We then repeat this process of marking and completing. It will be important for us to see that the completion process does not seriously inflate the number of triangular cells in P_n. We have not found any result in the literature saying that the overall number of triangles created through completion always stays proportional to the number of cells marked throughout the refinement process. To establish this will be a bit technical and will be the subject of this subsection.

Suppose that P is an admissible partition with $\#(P) > 2$ (the case $\#(P) = 2$ is trivial in what follows). To each $\Delta \in P$, we associate a triangular cell $F(\Delta) \in P$ as follows. Let $v(\Delta)$ be the newest vertex of Δ and $E(\Delta)$ the edge of Δ opposite to $v(\Delta)$. If $E(\Delta)$ is a boundary edge then we define $F(\Delta) = \emptyset$. Otherwise, there is a unique triangular cell $\Delta' \neq \Delta$ which has $E(\Delta)$ as one of its edges and we define $F(\Delta) = \Delta'$. One can visualize the mapping $\Delta \to F(\Delta)$ as a flow determined by the vector which serves to bisect Δ in the subdivision process.

By a *chain* $C(\Delta)$ (with starting cell Δ) in P, we mean a sequence $\Delta, F(\Delta), \ldots, F^m(\Delta)$ with no repetition of the cells in this chain and with $F^{m+1}(\Delta) = F^k(\Delta)$, for some $k \in \{0, \ldots, m-1\}$ or $F^{m+1}(\Delta) = \emptyset$.

Remark. We shall see below that by starting with a particular assignment of newest vertices in P_0, for any of the subsequent partitions $P = P_k$, the only way $F^{m+1}(\Delta) = F^n(\Delta)$ is for n to be equal to $m-1$. We shall therefore assume this property in going further.

The completion of a chain $C(\Delta)$ is a collection $\bar{C}(\Delta)$ of cells produced by two sets of subdivisions. In the first set, each cell $\Delta' = F^k(\Delta)$ in this chain is subdivided using the newest vertex bisection (i.e. the insertion of the line segment connecting $v(\Delta')$ to the midpoint on $E(\Delta')$). This subdivision of Δ' produces two new cells (the children of Δ').

After this first set of subdivisions has been completed, there will generally be cells with hanging nodes. The second part of the subdivision process is to subdivide each of the children that have a hanging node. Hanging nodes occur inside a cell $\Delta' = F^k(\Delta)$, when $E(F^{k-1}(\Delta)) \neq E(F^k(\Delta))$. In this case the new edge we need to add in $F^k(\Delta)$ is the one connecting the midpoints of these two edges. This part of the subdivision process removes all remaining hanging nodes. By the above remarks, this has the effect of subdividing (into two grandchildren) the child of $F^k(\Delta)$, $k = 1, \ldots, m'$,which has $E(F^{k-1}(\Delta))$ as an edge. Here $m' = m - 1$, when $F^{m+1}(\Delta) = F^{m-1}(\Delta)$, and $m' = m$, when $F^{m+1}(\Delta) = \emptyset$.

We shall make some further observations about the structure of $\bar{C}(\Delta)$ and the resulting flow structure. For this purpose, we shall introduce a way of labelling all edges that arise in the subdivision process.

We shall label the edges in the partitions $P_0, P_1, \ldots, P_n$ by nonnegative integers. This labelling will give us a simple way to keep track of the subdivision and completion process. Given any triangular cell Δ in one of these partitions, the sides of Δ will be labelled by $(i+1, i+1, i)$ where $i = g(\Delta)$ is the nonnegative integer that represents the generation of Δ (i.e. how many subdivisions of a cell in P_0 were needed to create Δ). The labelling will be such that the lowest labelled side will be $E(\Delta)$, i.e. the side opposite the newest vertex of Δ. At the outset, it may appear that the labelling of a side will depend on the triangle Δ and so a side will get two labels depending on which triangle we view it to be in. However, as we shall see, for admissible partitions the labelling of an edge can be *independent* of the triangle to which it belongs provided we start with a suitable labelling of P_0.

To start the labelling process, we describe how to label the edges in P_0, see Figure 2 which displays a labelling which is consistent with the assignment of newest vertices shown in Figure 1.

We begin with the following lemma.

Lemma 2.1 *For any initial partition P_0 there is a labelling of the edges in P_0 such that each edge is given a label of either* 0 *or* 1 *and whenever a triangle $\Delta \in P_0$ then exactly two of its edges are labelled with a* 1 *and the other edge is labelled with a* 0*.*

Proof. We shall show the existence of such a labelling by using the theory of matchings in cubic graphs. A cubic graph is one in which every vertex of the graph has exactly three edges. Since the labelling refers only to the combinatorial structure we can map the domain Ω covered by P_0 homeomorphically onto a portion of the sphere in $\mathbb{R}^3$ which induces a triangulation of that portion of the sphere. Let $\hat{P}_0$ denote any fixed completion of P_0 to a triangulation of the whole sphere and consider the *dual graph* of $\hat{P}_0$. This is the graph whose vertices are the triangles of $\hat{P}_0$. Two vertices in this dual

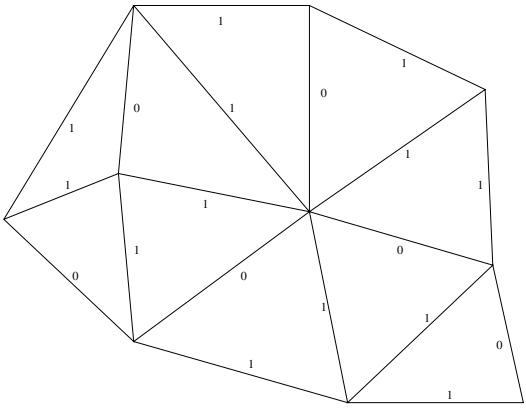

Fig. 2. Assignment of newest vertices in P_0

graph are connected if the corresponding triangles share an edge. This graph is a cubic graph.

Given any graph, a set of edges $\mathcal{E}$ of the graph such that no two edges from $\mathcal{E}$ share the same vertex is called a matching. The matching is perfect if all vertices are covered. The existence of perfect matchings in cubic graphs has been established by Petersen [20], see also Theorem 3.4.1 on page 110 in [15]. Given such a perfect matching $\mathcal{E}$ for the cubic graph generated by P_0 (as described above), we assign a label 0 to each edge of $\hat{P}_0$ which corresponds to an edge in $\mathcal{E}$. Every other edge of $\hat{P}_0$ is assigned the label 1. By Petersen's theorem, for each triangle Δ in $\hat{P}_0$, exactly one of the edges of Δ will be labelled 0 and the other two will be labelled 1. Clearly this labelling of $\hat{P}_0$ induces a labelling with the desired property also for any subset of $\hat{P}_0$ and, in particular, also for P_0 which establishes the claimed existence. $\ominus$

There remains the problem of constructing a practical scheme for giving a labelling of the form expressed in Lemma 2.1. One way is to employ methods from combinatorial optimization. By the above proof, it suffices to construct a maximal matching for the dual graph of P_0. Such schemes can be found, for instance, in [15, p. 238, Theorem 10.6].

To explain a simple alternative let us consider yet another interpretation of the labelling in Lemma 2.1. For any collection A of triangular cells in P_0 we denote by Ω_A their union. Suppose we could find a set $Q \subseteq P_0$ with the following properties:

(i) All triangles in $P_0 \setminus Q$ have at least one edge on the boundary of Ω.
(ii) The domain Ω_Q can be decomposed into an essentially disjoint union of quadrilaterals formed by pairs of adjacent triangles from Q.

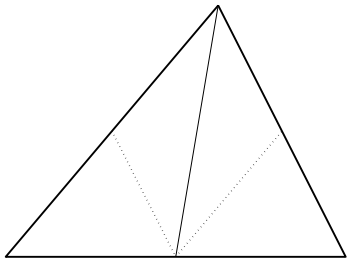

Fig. 3. Refinement for labelling

Given such a collection Q, we can assign the following labels to the edges of P_0. For each pair of triangles from Q whose union forms one of the quadrilaterals, we assign 0 to their common edge and 1 to all other edges. By (i) we have missed at most edges on the boundary Γ of Ω. If such an edge E belongs to a triangle with two interior edges, they must have the label 1, so we assign the label 0 to E. If the edge belongs to a triangle with two edges on the boundary we label one by 0 and the other one by 1 to obtain the type of labelling asserted by the lemma.

So the labelling problem now reduces to finding a collection Q in P_0 satisfying (i) and (ii). Instead of finding such a set Q for P_0 whose existence is guaranteed by Petersen's Theorem, we shall mention a simple way of constructing such a Q for a refinement of P_0. In fact, we can subdivide each Δ in P_0 into four triangles such that the new partition P_0' consisting of these new triangles satisfies (i) and (ii) for $Q = P_0'$. Indeed, given $\Delta \in P_0$, we use a bisection to divide Δ into two triangles (the side chosen for this bisection can be chosen arbitrarily). Each of the two resulting triangles (*they will be our "quadrilaterals" from* Ω_Q) are then again subdivided by bisecting the side the triangle has in common with Δ The last two bisecting edges receive labels 0, and all the other edges resulting from this procedure receive labels 1. If this is done for each $\Delta \in P_0$, the resulting partition P_0' has no hanging nodes since all the edges of P_0 are divided into two. Thus P_0' can be labelled as described above, see Figure 3.

Given the labelling of sides in P_0 by Lemma 2.1, we define the newest vertex of a triangular cell $\Delta \in P_0$ to be the vertex opposite the side which is labelled by 0. In going further, we shall always assume that the initial labelling of newest vertices in P_0 has been done in accordance with Lemma 2.1. Notice that this means that any chain in P_0 has at most two cells and that the subdivision of these cells gives an admissible partition (i.e. there is no need to go to the second subdivisions which generate grandchildren). This is illustrated in Figure 4 where one triangle (from P_0 which is labelled with an "x") has been marked for subdivision.

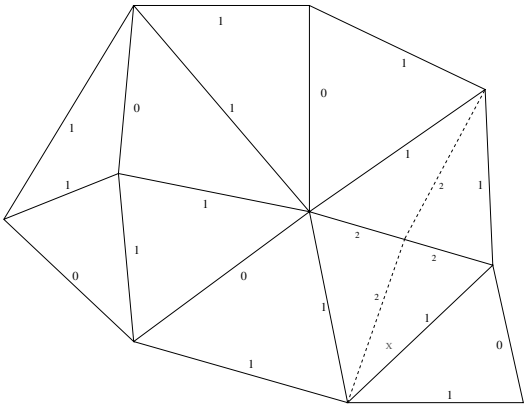

Fig. 4. Labelling of new edges

We now give a rule to label any edges that arise from the subdivision-completion process. There will be two main properties of this labelling. The first is that each triangular cell will have sides with labels $(i, i, i-1)$ for some positive integer i. The second is that the newest vertex for this cell will be the vertex opposite the side with lowest label. Certainly the edges in P_0 have such a labelling as we have just shown.

Suppose that we have such a labelling for the edges in P_k and let us describe how to label the edges in P_{k+1}. Suppose that a triangular cell $\Delta \in P_k$ has sides which have been labelled $(i, i, i-1)$ and the newest vertex for this cell is the one opposite the side labelled $i-1$. When this cell is subdivided (using newest vertex bisection) the side labelled $i-1$ is bisected and we label each of the two new sides $i+1$. We also label the *bisector* by $i+1$, i.e. the new edge connecting the newest vertex of Δ with the midpoint of the edge $E(\Delta)$ labelled by $i-1$. Thus each new triangle now has sides labelled $(i, i+1, i+1)$ with the newest vertex opposite the side with the lowest label. We note the important fact that if a cell has label $(i+1, i+1, i)$ then it is of generation i (i.e. it has been obtained from a cell in P_0 by i subdivisions). Therefore, specifying that the generation of the cell is i is the same as specifying that its label is $(i+1, i+1, i)$. Figure 5 gives an example for the completion process which corresponds to marking one of the triangles of P_1 in Figure 4 which is labelled with an "x" in Figure 5.

Lemma 2.2 *Suppose P_0 is an arbitrary partition and its edges and newest vertices are labelled in accordance with Lemma 2.1. Suppose that $P_1, \ldots, P_n$ are partitions which are generated from P_0 using the marking, subdivision, and completion process. We label edges in $P_1, \ldots, P_n$ as described above. Then there holds for each $k = 0, 1, \ldots, n$:*

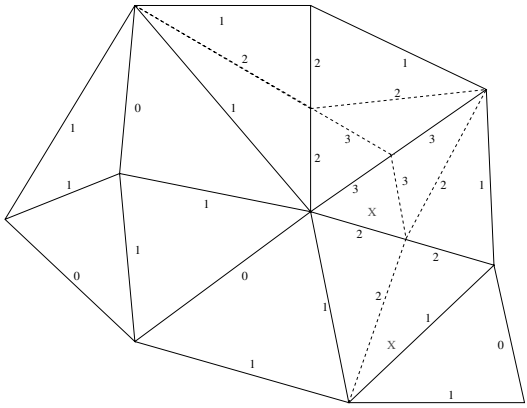

Fig. 5. Labelling in the completion process

(i) each edge in P_k has a unique label independent of the two triangles which share this edge.

(ii) If Δ is a triangular cell in P_k of generation $g(\Delta) = i$, i.e. the edge with label i is the side shared by Δ and $F(\Delta)$, then $g(F(\Delta)) \in \{i, i-1\}$. If $g(F(\Delta)) = i$ the flow ends at $F(\Delta)$.

(iii) For any $\Delta \in P_k$ of generation $g(\Delta) = i$, the cells in its chain $C(\Delta) = \{\Delta, F(\Delta), \ldots, F^m(\Delta)\}$ have the property that $g(F^j(\Delta)) = i - j$, $j = 0, \ldots, m-1$, and the terminating cell $F^m(\Delta)$ for this chain is either of generation $i - m + 1$ or it is a boundary cell with lowest labelled edge an edge of the boundary.

Proof. This is proved by induction on k. All three assertions are clear for $k = 0$ by the construction of the labelling of P_0 given by Lemma 2.1. Suppose that we have proven the lemma for P_{k-1} and consider P_k.

Proof of (i). Note that the above rules leave the labels of all those edges unchanged that are not effected by subdivisions. Any edge E in P_k which was not in P_{k-1} was obtained in one of two ways. The first is that it is a new edge which was added as a bisector in the subdivision-completion process. In this case there is nothing to prove about its labelling being unique. The second possibility is that the new edge E was obtained by bisecting an edge E', say with label i, from P_{k-1}. Let Δ and Δ' be the two triangular cells in P_{k-1} which shared E'. For one of these triangular cells, which we can assume is Δ', we have $E' = E(\Delta')$. So Δ' had label $(i+1, i+1, i)$ and therefore viewed from Δ', E is assigned the label $i+2$. By our induction assumption, as an edge of Δ, E' is also labelled by i, so that Δ has either the label $(i, i+1, i+1)$ or $(i, i, i-1)$. In the first case the situation is symmetric to Δ' since $E' = E(\Delta)$ is bisected and E as one of the halves is labelled $i+2$.

In the second case E' is still an edge of one of the children of Δ obtained by bisecting the edge with label $i-1$. Thus the midpoint of E' is still a hanging node in that child which is now labelled $(i+1, i+1, i)$. This is the situation described by the first case which finishes the proof of (i).

Proof of (ii). Since Δ has generation i, $E(\Delta)$ has label i. Since this edge is shared with $F(\Delta)$, we conclude that the sides of $F(\Delta)$ have either labels $(i, i, i-1)$ or $(i+1, i+1, i)$. Also, in this latter case $F^2(\Delta) = \Delta$.

Proof of (iii). This follows from (ii). □

Note that all admissible partitions, generated by newest vertex bisection based on an initial labelling according to Lemma 2.1, are *graded* in the sense that any two cells sharing an edge differ in generation by at most one.

We shall next give a bound for the number of cells in P_n. In preparation for this, let us note that there are constants c_1, C_1 depending only on P_0 such that for each Δ of generation i, we have $c_1 2^{-i} \leq |\Delta| \leq C_1 2^{-i}$ where $|\Delta| = \text{meas}(\Delta)$ is the area of Δ. Indeed, each subdivision of a cell gives two cells with each having half the area of the original cell. By adjusting the constants if necessary, we also infer from (2.1) that

$$c_1 2^{-i/2} \leq \text{diam}(\Delta) \leq C_1 2^{-i/2}, \quad g(\Delta) = i, \tag{2.3}$$

because P_n belongs to an admissible family with parameters $\hat{C}$, G_0, a_0 depending only on the initial partition P_0.[1]

Using (2.3), we can bound the distance between any two cells from a chain $C(\Delta)$. Let the cell $F^p(\Delta)$ be of generation $\gamma = g(F^p(\Delta))$ and let $F^q(\Delta)$ be another cell from $C(\Delta)$, where $0 \leq q < p \leq m$. Then the distance between these two cells can be bounded by the sum of the diameters of the cells in the chain $C(\Delta)$ between them. According to (iii) from Lemma 2.2, the generation of these cells decrease exactly by one, and therefore

$$\text{dist}(F^p(\Delta), F^q(\Delta)) \leq \sum_{i=1}^{p-q-1} C_1 2^{-(\gamma+i)/2} \leq \frac{\sqrt{2}}{\sqrt{2}-1} C_1 2^{-\gamma/2}. \tag{2.4}$$

Let $\mathcal{M} := \cup_{j=0}^{n-1} \mathcal{M}_j$ be the collection of all cells that were marked in going from P_0 to P_n. Here is the way to view the following argument that will bound the number of cells in P_n. We give each cell $\Delta' \in \mathcal{M}$ a fixed number $C > 0$ of dollars to spend. These cells will spend these dollars in such a way that each new cell that was created in going from P_0 to P_n will

[1] Generic constants whose value may vary on each occurrence will be denoted by C. Whenever the specific value of a constant matters we shall use subscripts. Using the same subscript for different constants indicates that they occur in the same type of estimates. It is then understood that they may need to be adjusted at each occurrence.

get at least $c > 0$ dollars where c is an absolute constant. This means, we can bound the number of new cells created by the number of marked cells.

We now describe how a cell $\Delta' \in \mathcal{M}$ will spend its money. We define

$$A := 7C_1 \sum_{j=-1}^{\infty} 2^{-j/4} \tag{2.5}$$

where C_1 is the constant in (2.3). We define now a function $\lambda : P_n \times \mathcal{M} \to \mathbb{R}$ as follows:

$$\lambda(\Delta, \Delta') := \begin{cases} (j-k+2)^{-2}, & \text{if } g(\Delta') = j,\ g(\Delta) = k, \\ & \operatorname{dist}(\Delta, \Delta') \le A2^{-k/2} \text{ and } k \le j+1; \\ 0, & \text{otherwise.} \end{cases} \tag{2.6}$$

The quantity $\lambda(\Delta, \Delta')$ is the portion of money which is spent by the marked cell Δ' on nearby cells $\Delta \in P_n$ of generation at most $g(\Delta') + 1$. Given $\Delta' \in \mathcal{M}$ with $g(\Delta') = j$, there are for any $k \le j+1$ at most C' cells $\Delta \in P_n$ of generation $g(\Delta) = k$ which satisfy $\operatorname{dist}(\Delta, \Delta') \le A2^{-k/2}$ (see (2.3)), where C' is an absolute constant. Hence for any such $\Delta' \in \mathcal{M}$ one has

$$\sum_{\Delta \in P_n} \lambda(\Delta, \Delta') \le C' \sum_{\nu=1}^{\infty} \nu^{-2} = C \tag{2.7}$$

with C an absolute constant. It follows then from (2.7) that

$$\sum_{\Delta' \in \mathcal{M}} \sum_{\Delta \in P_n} \lambda(\Delta, \Delta') \le C\#(\mathcal{M}), \tag{2.8}$$

i.e. the total amount of money spent by all the marked cells is proportional to their number. Conversely, each cell in P_n receives at least a minimum share bounded away from zero as will be shown next.

Lemma 2.3 *For any $\Delta \in P_n \setminus P_0$ we have*

$$\sum_{\Delta' \in \mathcal{M}} \lambda(\Delta, \Delta') \ge c \tag{2.9}$$

where $c > 0$ is an absolute constant.

Proof. We fix Δ and let $k := g(\Delta) \ge 1$ be its generation. We are going to define a sequence of marked cells $\Delta_1, \ldots, \Delta_s \in \mathcal{M}$ associated to Δ with each Δ_ν of generation $\ge k-1$. Δ_1 is the marked cell such that $\Delta \in \bar{C}(\Delta_1)$, i.e. Δ arose by subdividing the chain associated to Δ_1. Given that Δ_j has been defined, we let $\Delta_{j+1} \in \mathcal{M}$ be the marked cell such that $\Delta_j \in \bar{C}(\Delta_{j+1})$. We let s be the smallest integer such that $g(\Delta_s) = k-1$. Note that there must

be such an integer because subdividing a chain can only increase the highest generation in the chain by one. For each j, we let $\tilde{\Delta}_j$ denote the parent of Δ_j from the chain $C(\Delta_{j+1})$, i.e. $\Delta_j \subset \tilde{\Delta}_j$.

For each $j \geq -1$, we keep a running count $m(i, j)$ of the number of cells Δ_ν with $g(\Delta_\nu) = j + k$, $\nu \leq i$.

Case 1: There is a $j \in \{-1, 0, \ldots\}$ such that $m(i, j) > 2^{j/4}$ for some $1 \leq i \leq s$. In this case, we choose j^* as the integer which has the smallest i (defined to be $i^* := i$) with this property. In other words, j^* is the integer j whose count first exceeds $2^{j/4}$ and i^* is the smallest i for which $m(i, j^*) \geq 2^{j^*/4}$. It follows from (2.4) and $g(\Delta_\mu) - g(\tilde{\Delta}_\mu) \leq 2$ that

$$\text{dist}(\Delta, \Delta_\nu) \leq \frac{\sqrt{2}}{\sqrt{2}-1} C_1 \sum_{\mu=1}^{\nu} 2^{-(g(\Delta_\mu)-2)/2} \leq 7C_1 \sum_{\mu=1}^{\nu} 2^{-g(\Delta_\mu)/2} . \tag{2.10}$$

and therefore for $\nu < i^*$

$$\begin{aligned} \text{dist}(\Delta, \Delta_\nu) &\leq 7C_1 \sum_{j=-1}^{\infty} m(\nu, j) 2^{-(j+k)/2} \\ &\leq 7C_1 2^{-k/2} \sum_{j=-1}^{\infty} 2^{-j/4} = A 2^{-k/2}. \end{aligned} \tag{2.11}$$

Thus, for each Δ_ν, $\nu < i^*$, with $g(\Delta_\nu) = j^* + k$, we have $\lambda(\Delta, \Delta_\nu) = (j^* + 2)^{-2}$. Since there are at least $2^{j^*/4}$ such values of ν, we obtain

$$\sum_{\Delta' \in \mathcal{M}} \lambda(\Delta, \Delta') \geq (j^* + 2)^{-2} 2^{j^*/4} \geq c \tag{2.12}$$

with $c := \min_{\nu \geq -1} (\nu + 2)^{-2} 2^{\nu/4}$ an absolute constant. This is (2.9) in this case.

Case 2: In this case, for all $j \geq -1$ we have $m(s, j) \leq 2^{j/4}$. Therefore, as in (2.11), we have

$$\text{dist}(\Delta, \Delta_s) \leq 7C_1 2^{-k/2} \sum_{j=-1}^{\infty} 2^{-j/4} = A 2^{-k/2}. \tag{2.13}$$

This means $\lambda(\Delta, \Delta_s) = 1$ and again we have (2.9). □

Theorem 2.4 *Suppose that $P_0, \ldots, P_n$ is a sequence of partitions generated as described above. Then, there is a constant $C_2 > 0$ depending only on P_0 such that*

$$\#(P_n) \leq \#(P_0) + C_2(\#(\mathcal{M}_0) + \ldots + \#(\mathcal{M}_{n-1})). \tag{2.14}$$

Proof. It follows from Lemma 2.3 and (2.8) that the number of new cells added in the subdivision and completion process does not exceed $C(\#(\mathcal{M}_0)+\ldots+\#(\mathcal{M}_{n-1}))/c$ and so (2.14) holds with $C_2 := C/c$. □

We shall use the remainder of this section to spell out different settings in which we shall use completion and formulate variants of Theorem 2.4 for these settings.

It follows from Theorem 2.4 that to bound the cardinality $\#(P)$ of a partition P that arises through markings, refinements, and then completion, we need only keep control of the number of markings used in the creation of P. Given a partition P and a second partition P' that is obtained from P by first marking some cells for subdivision and then doing a completion, we shall use the notation $m(P'|P)$ to denote the number of markings that were used to go from P to P'. Of course, there may be many ways to mark and complete to go from P to P' but the particular method (and therefore the number $m(P'|P)$) will always be clear in the context of the algorithm under discussion. Thus, (2.14) can be rewritten as

$$\#(P_n) \leq \#(P_0) + C_2 \sum_{k=1}^{n} m(P_k|P_{k-1}). \tag{2.15}$$

Up until now, we have restricted our discussion to the setting where a completion is done after each set of markings and elementary subdivisions. We can also apply completion in the following setting. Suppose P is an admissible partition and $\bar{P}$ is a refinement of P which is not necessarily admissible. We complete $\bar{P}$ to an admissible partition P' by a sequence of markings and elementary refinements as follows. We define $\mathcal{M}_0$ to be the set of all $\Delta \in P$ which are not in $\bar{P}$ (i.e. the cells $\Delta \in \mathcal{M}_0$ were subdivided in going from P to $\bar{P}$). We perform the subdivisions for the markings $\mathcal{M}_0$ and then form the completion to arrive at the admissible partition P_1 which is a refinement of P. We let $\mathcal{M}_1$ denote the set of all $\Delta \in P_1$ that are further refined in going from P to $\bar{P}$ and let P_2 be the admissible partition obtained after marking the cells $\mathcal{M}_1$ and then completing. Continuing in this way we arrive at the first admissible partition P_m which is a refinement of $\bar{P}$.

Lemma 2.5 *Let P be any admissible partition and let $\bar{P}$ be a refinement of P. Then, there is a completion P' of $\bar{P}$ which is an admissible partition that refines $\bar{P}$ and satisfies*

$$\begin{aligned} \#(P') &\leq \#(P_0) + C_2(m(P|P_0) + m(\bar{P}|P)) \\ &\leq \#(P_0) + C_2(\#(\bar{P}) - \#(P_0)) \leq C_2\#(\bar{P}). \end{aligned} \tag{2.16}$$

Proof. Each of the cells in $\mathcal{M}_0 \cup \mathcal{M}_1 \cup \cdots \cup \mathcal{M}_{m-1}$ was subdivided in going from P to $\bar{P}$ and hence $\sum_{k=0}^{m-1} \#(\mathcal{M}_k) = m(\bar{P}|P) \leq \#(\bar{P}) - \#(P)$. Also $m(P|P_0) \leq \#(P) - \#(P_0)$. Using this in (2.15) we obtain (2.16). □

3 Galerkin approximations

Numerical methods based on Galerkin approximations to (1.1) begin with the weak formulation of (1.1) which is to find $u \in H_0^1(\Omega)$ such that

$$a(u, w) = (f, w), \quad w \in H_0^1(\Omega), \tag{3.1}$$

where $a(y, w) := (\nabla y, \nabla w)$, $(y, w) = (y, w)_\Omega := \int_\Omega ywdx$. We use the notation

$$|||w|||^2 := a(w, w) = \|\nabla w\|^2_{L_2(\Omega)}. \tag{3.2}$$

By Poincaré's inequality there exists a constant c_Ω, depending on Ω, such that for any $w \in H_0^1(\Omega)$,

$$c_\Omega \|w\|_{H^1(\Omega)} \leq |||w||| \leq \|w\|_{H^1(\Omega)}, \tag{3.3}$$

where $\|w\|^2_{H^1(\Omega)} = \|w\|^2_{L_2(\Omega)} + \|\nabla w\|^2_{L_2(\Omega)}$.

Given an admissible partition P, we shall denote the Galerkin solution to (3.1) by u_P throughout this paper. Thus u_P is the unique element in $\mathcal{S}_P \subset H_0^1(\Omega)$ which satisfies

$$a(u_P, w) = (f, w), \quad w \in \mathcal{S}_P. \tag{3.4}$$

From a practical point of view it would be better in (3.4) to replace f by an approximation and consider the solution of this modified variational problem as u_P. But we want to conform exactly to the algorithm in [17] so that we do not have to detour through a new development of a posteriori error analysis in this modified setting.

The Galerkin solution u_P is a best approximation to u from $\mathcal{S}_P$ in the energy norm

$$|||u - u_P||| = \inf_{S \in \mathcal{S}_P} |||u - S|||. \tag{3.5}$$

We cannot calculate u_P exactly. We therefore introduce the following numerical scheme in which the constant $0 < \delta < 1$ is at this point arbitrary but will be specified later.

GAL: *This algorithm takes as input an admissible partition P, an error tolerance μ and an initial approximation $\bar{u}_P \in \mathcal{S}_P$ to u_P that satisfies*

$$|||u - \bar{u}_P||| \leq A'\mu \tag{3.6}$$

with A' a fixed constant. It applies a preconditioned conjugate gradient scheme with initial guess $\bar{u}_P$ to obtain an approximation $\hat{u}_P := \mathbf{GAL}(P, \mu, \bar{u}_P)$ to u_P that satisfies

$$|||u_P - \hat{u}_P||| \leq \delta\mu. \tag{3.7}$$

Remark 3.1 The number of iterations of the preconditioned conjugate gradient scheme needed to achieve (3.7) depends only on the quotient A'/δ. Since each iteration requires at most $C\#(P)$ computations, it follows that the number of computations $N(\mathbf{GAL}, P, \mu, \bar{u}_P)$ satisfies

$$N(\mathbf{GAL}, P, \mu, \bar{u}_P) \leq C_3(A'/\delta)\#(P), \tag{3.8}$$

where $C_3 : t \to C_3(t)$ increases as a function of t.

Proof. Since $|||u - u_P||| \leq |||u - \bar{u}_P|||$ we have $|||u_p - \bar{u}_p||| \leq 2A'\mu$. Thus the target accuracy in (3.7) is a fixed fraction of the deviation of the initial guess from the exact Galerkin solution u_P. The BPX-scheme allows one to precondition the stiffness matrices corresponding to nonuniformly refined meshes in such a way that the condition numbers stay uniformly bounded, [11, 13, 18]. Since the error reduction in one iteration of the conjugate gradient scheme then reduces the current error by a multiplicative factor strictly less than one, only a fixed number of iterations depending only on A'/δ is needed to achieve (3.7). Each application of the preconditioner requires only $C\#(P)$ operations. □

We shall see that in our later applications of **GAL** the constant δ will be fixed once and for all and that there exists an A' such that (3.6) is satisfied in each application.

4 Adaptive approximation

In this section, we shall discuss adaptive approximation of a function w which is known to us and for which we can compute local polynomial approximants. These results do not apply directly to u since it is unknown but they serve to tell us what is the best we can expect in terms of approximating u. We shall also use the approximation methods we develop in this section in parts of our adaptive algorithm, namely to numerically approximate the right hand side f and to execute our coarsening step.

We limit ourselves to adaptive methods based on subdivision using the newest vertex subdivision rule starting with an initial partition P_0 and a labelling of vertices as given in Lemma 2.1. We recall that any adaptively generated partition P can be associated to a tree $T(P)$ which is a proper subtree of T_*. The leaves of $T(P)$ give the partition P. Conversely, any finite proper subtree T gives a partition P consisting of the leaves of T.

Recall that $\mathcal{P}_n$ denotes the set of all adaptively generated partitions which are obtained from P_0 by applying at most n elementary subdivisions. $\mathcal{P}_0$ consists of the single partition P_0. The partitions in $\mathcal{P}_n$ correspond to trees $T = T(P)$ which satisfy $N(T) = n$. Similarly, we denote by $\mathcal{P}_n^a$ all the partitions in $\mathcal{P}_n$ which are admissible. We also recall our notation $\mathcal{S}_P$ of continuous piecewise linear functions subordinate to P which vanish on $\partial\Omega$.

4.1 Adaptive approximation in the $H^1(\Omega)$-norm

Given a function $w \in H^1_0(\Omega)$ and a partition P, we define

$$E(w, \mathcal{S}_P)_{H^1(\Omega)} := \inf_{S \in \mathcal{S}_P} \|w - S\|_{H^1(\Omega)}, \tag{4.1}$$

which is the smallest error we can achieve by approximating w in the $H^1(\Omega)$ norm by the elements of $\mathcal{S}_P$. In the case that $w = u$ is our solution to (1.1) and P is admissible then, in view of (3.3),

$$|||u - u_P||| \le E(u, \mathcal{S}_P)_{H^1(\Omega)} \le c_\Omega^{-1} |||u - u_P|||, \tag{4.2}$$

where u_P is the Galerkin approximation associated to P.

Returning to the general case of a $w \in H^1_0(\Omega)$, we enter a competition over all partitions $P \in \mathcal{P}_n$ and introduce the error

$$\sigma_n(w) := \inf_{P \in \mathcal{P}_n} E(w, \mathcal{S}_P)_{H^1(\Omega)} \tag{4.3}$$

of best adaptive approximation.

It is unreasonable to expect any adaptive algorithm to perform exactly the same as $\sigma_n(w)$. However, we may expect the same *asymptotic* behavior. To quantify this, we introduce for any $s > 0$, the class $\mathcal{A}^s := \mathcal{A}^s(H^1_0(\Omega))$ of functions $w \in H^1_0(\Omega)$ such that

$$\sigma_n(w) \le M n^{-s}, \quad n = 1, 2, \ldots. \tag{4.4}$$

The smallest M for which (4.4) is satisfied is the norm in $\mathcal{A}^s$:

$$\|w\|_{\mathcal{A}^s} := \sup_{n \ge 1} n^s \sigma_n(w). \tag{4.5}$$

We have a similar measure of approximation when we restrict ourselves to admissible partitions. Namely,

$$\sigma^a_n(w) := \inf_{P \in \mathcal{P}^a_n} E(w, \mathcal{S}_P)_{H^1(\Omega)} \tag{4.6}$$

now measures the best nonlinear approximation error obtained from admissible partitions and $\dot{\mathcal{A}}^s := \dot{\mathcal{A}}^s(H^1_0(\Omega))$ consists of all w which satisfy

$$\sigma^a_n(w) \le M n^{-s}, \quad n = 1, 2, \ldots. \tag{4.7}$$

The smallest M for which (4.7) holds serves to define the norm $\|w\|_{\dot{\mathcal{A}}^s}$.

Remark 4.1 We have $\mathcal{A}^s(H^1_0(\Omega)) = \dot{\mathcal{A}}^s(H^1_0(\Omega))$ with equivalent norms.

Indeed, given any partition $P \in \mathcal{P}_n$ (not necessarily admissible) which achieves the error $E(w, \mathcal{S}_P)_{H^1(\Omega)} = \sigma_n(w)$, we can use Lemma 2.5 to complete P to an admissible partition P'. Since $\#(P') \leq C_2\#(P)$ (see (2.16)), and the error on the partition P' does not exceed the error on P and therefore

$$\sigma^a_{C_2 n}(w) \leq \sigma_n(w) \tag{4.8}$$

from which the remark easily follows.

The reader should be interested to know what functions are in $\mathcal{A}^s$. It turns out that these classes are related to certain Besov spaces. Since we do not use or need this information in the construction of the algorithm, we postpone this discussion to the Appendix.

4.2 Adaptive approximation in $H^{-1}(\Omega)$

We shall also need approximation by piecewise constants which will be our vehicle for resolving the right hand side f in our numerical algorithm for (1.1). The approximation will take place in the $H^{-1}(\Omega)$ norm which is defined for a tempered distribution g by duality:

$$\|g\|_{H^{-1}(\Omega)} := \sup_{\phi \in H^1_0(\Omega)} \frac{\langle g, \phi \rangle}{|||\phi|||}, \tag{4.9}$$

where $\langle \cdot, \cdot \rangle$ denotes the duality pairing induced by the standard L_2-inner product. Given a partition P, we let $\mathcal{S}^0_P$ denote the class of piecewise constant functions subordinate to P. For a function $g \in H^{-1}(\Omega)$, we have

$$E(g, \mathcal{S}^0_P)_{H^{-1}(\Omega)} := \inf_{S \in \mathcal{S}^0_P} \|g - S\|_{H^{-1}(\Omega)}, \tag{4.10}$$

which is the best error we can achieve by approximating f in the $H^{-1}(\Omega)$-norm by elements of $\mathcal{S}^0_P$. Analogously, we have

$$\sigma_n(g)_{H^{-1}(\Omega)} := \inf_{P \in \mathcal{P}_n} E(g, \mathcal{S}^0_P)_{H^{-1}(\Omega)} \tag{4.11}$$

the error of best nonlinear approximation to g by piecewise constants.

As in the case of piecewise linear approximation, we introduce for any $s > 0$, the approximation class $\mathcal{A}^s(H^{-1}(\Omega))$ and its norm exactly as in (4.4) and (4.5) except that we use $\sigma_n(g)_{H^{-1}(\Omega)}$ in place of $\sigma_n(w)$. We have a similar measure of approximation when we restrict ourselves to admissible partitions. We denote the approximation class in this case by $\dot{\mathcal{A}}^s(H^{-1}(\Omega))$

Suppose now that $g \in L_2(\Omega)$. Then $g \in H^{-1}(\Omega)$. If P is any partition of Ω and $\Delta \in P$, we define

$$g_\Delta := \frac{1}{|\Delta|} \int_\Delta g \tag{4.12}$$

which is the average of g over Δ. Also, g_Δ is the best approximation to g in $L_2(\Delta)$ by constant functions. Therefore,

$$S_P^0(g) := \sum_{\Delta\in P} g_\Delta \chi_\Delta \tag{4.13}$$

is the best $L_2(\Omega)$ approximation to g by piecewise constants subordinate to P.

We can use $S_P^0(g)$ also to approximate g in the $H^{-1}(\Omega)$-norm. In fact, for any admissible partition P, we have the following bound for the approximation error

$$E(g, \mathcal{S}_P^0)^2_{H^{-1}(\Omega)} \le \|g - S_P^0(g)\|^2_{H^{-1}(\Omega)} \le C_0 \bar{E}(g, P) \tag{4.14}$$

where

$$\bar{E}(g, P) := \sum_{\Delta\in P} |\Delta| \|g - g_\Delta\|^2_{L_2(\Delta)}. \tag{4.15}$$

Indeed, to see that this is true, let ϕ be any function in $H_0^1(\Omega)$ of unit norm. Then, since S_P^0 is the orthogonal projection to $\mathcal{S}_P^0$, we have

$$\begin{aligned} \langle g - S_P^0(g), \phi\rangle &= \langle g - S_P^0(g), \phi - S_P^0(\phi)\rangle \\ &= \sum_{\Delta\in P} \int_\Delta (g - g_\Delta)(\phi - \phi_\Delta). \end{aligned} \tag{4.16}$$

We use the Cauchy-Schwartz inequality on each of the terms in the last sum and then the Poincaré inequality to bound such a term by

$$\|g - g_\Delta\|_{L_2(\Delta)} \|\phi - \phi_\Delta\|_{L_2(\Delta)} \le C|\Delta|^{1/2} \|\nabla\phi\|_{L_2(\Delta)} \|g - g_\Delta\|_{L_2(\Delta)}.$$

Here C is an absolute constant because all the triangular cells Δ are uniformly shape regular (2.1). Using this in (4.16) and again applying Cauchy-Schwarz, we arrive at (4.14) by taking a supremum over all ϕ of norm one in $H_0^1(\Omega)$.

We can use $\bar{E}(g, P)$ to define another nonlinear approximation error:

$$\bar{\sigma}_n^2(g) := \inf_{P\in\mathcal{P}_n} \bar{E}(g, P). \tag{4.17}$$

Analogous to the approximation classes defined above, we define $\bar{\mathcal{A}}^s(H^{-1}(\Omega))$ using $\bar{\sigma}_n(g)$ and the norm for this class as before. When dealing with admissible partitions we shall denote this class by $\dot{\bar{\mathcal{A}}}^s(H^{-1}(\Omega))$. As before $\dot{\bar{\mathcal{A}}}^s(H^{-1}(\Omega)) = \bar{\mathcal{A}}^s(H^{-1}(\Omega))$ with equivalent norms.

4.3 An algorithm for adaptive approximation of a given function

In this section, we wish to describe some of the results of [2] which give adaptive approximation algorithms for approximating a given target function v defined on Ω in a specified norm. These algorithms are different from an AFEM since they assume that the target function v is fully known (whereas our solution u to (1.1) is not). We shall use these algorithms in two different settings which we shall describe in the following subsections.

Although the algorithms in [2] apply in a more general setting, we shall limit our discussion to the case of newest vertex bisection and its associated master tree T_*. The starting assumption is that there is a functional e which associates to each triangular cell Δ in the master tree a nonnegative real number $e(\Delta)$. In applications, $e(\Delta)$ is usually some local approximation error (or a bound for this local error) associated to Δ.

There are two algorithms, called **Second Algorithm** and **Modified Second Algorithm** in [2], that we shall utilize. We shall limit our discussion in this section to the **Second Algorithm**. The **Modified Second Algorithm**, which is a variant of the **Second Algorithm** will bee introduced in §4.5 when we shall need it for coarsening.

The **Second Algorithm** in [2] can be applied under the assumption that e satisfies the following property: for each $\Delta \in T_*$ and each subtree T which contains Δ as its only root, we have

$$\sum_{\Delta' \in \mathcal{L}(T)} e(\Delta') \leq C\, e(\Delta), \tag{4.18}$$

where C is an absolute constant and $\mathcal{L}(T)$ is the set of leaves of T. Note that (4.18) follows, in particular, when e is *subadditive*, i.e.,

$$e(\Delta_1) + e(\Delta_2) \leq e(\Delta). \tag{4.19}$$

holds for any $\Delta \in T_*$ and its children Δ_1 and Δ_2. In this case, $C = 1$ in (4.18). We shall use the **Second Algorithm** in §4.4 when we approximate the right hand side f of (1.1).

For any proper subtree $T \subset T_*$, we define

$$E(T) := \sum_{\Delta \in \mathcal{L}(T)} e(\Delta) \tag{4.20}$$

as the error associated to T. In applications to adaptive partitioning, $E(T)$ would correspond to the square of the error associated to the partition given by the leaves of T.

Recall that $T_0 := T(P_0)$ and that for any tree T which is a refinement of T_0 the number $N(T)$ is the number of subdivisions necessary to create T

from T_0; it is also the number of interior nodes in T starting from the root T_0. We let $\mathcal{T}_n(T_0)$ be the set of all trees T with $N(T) = n$.

We enter the following competition among all proper trees in $\mathcal{T}_n(T_0)$:

$$E_n := E_n(T_0) := \inf_{T \in \mathcal{T}_n(T_0)} E(T). \tag{4.21}$$

So E_n is the smallest error we could achieve using trees from $\mathcal{T}_n(T_0)$. Although the problem is finite, it is numerically too intensive to find a best tree which achieves the minimum error E_n because $\#(\mathcal{T}_n(T_0))$ is exponential in n.

Suppose that given any Δ, we are able to compute $e(\Delta)$. We would like to find a tree which performs almost as well as the best tree from $\mathcal{T}_n(T_0)$ and to do so while only computing $O(n + \#(T_0))$ values $e(\Delta)$. The main result of [2] is to show that this is possible. Namely, that paper gives a numerically realizable algorithm which yields *near best* trees T from $\mathcal{T}_n(T_0)$ by which we mean

$$E(T) \leq C_0 E_{c_0 n} \tag{4.22}$$

where C_0, $c_0 > 0$ are absolute constants (in particular they do not depend on the starting partition P_0).

The algorithm finds a near best tree for the given value of n by creating a sequence of trees $T = T_j$, $j = 1, 2, \ldots$ starting from the initial tree T_0. Roughly speaking, at any given stage in the algorithm, it computes $e(\Delta)$ for all leaves in the current tree and then subdivides the leaves with the largest e-value. However, to avoid a lengthy sequence of subdivisions with insufficient reduction in the error, the algorithm in actuality uses a modified functional $\tilde{e}$. What is important in the context of the present paper is that to create the tree in (4.22) requires the computation of at most $\#(T_0) + Cn$ values of e where T_0 is the starting tree.

We shall use this algorithm in the following setting described in [2] as the **Thresholding Second Algorithm**. Given an admissible partition P which is not necessarily P_0 and the corresponding tree $T(P)$, let now $T_0 := T(P)$ be our set of root nodes. For any tree T which is a refinement of T_0 the number $N(T|T_0)$ denotes now the number of subdivisions necessary to create T from T_0. So $N(T|T_0) = N(T)$ when $T_0 = T(P_0)$. Given a tolerance $\mu > 0$, and an admissible partition P the algorithm produces a tree T_μ which is a refinement of $T_0 = T(P)$ with the following properties (see Corollary 5.4 of [2]):

P1: T_μ satisfies

$$E(T_\mu) \leq \mu. \tag{4.23}$$

P2: For absolute constants c_1 and C_1 (independent of P), it holds that whenever $\tilde{T}$ is a refinement of $T(P)$ satisfying

$$E(\tilde{T}) \leq c_1 \mu, \tag{4.24}$$

then

$$N(T_\mu|T_0) \le C_1 N(\tilde{T}|T_0). \tag{4.25}$$

Since we are interested in admissible partitions, we shall add a completion step to the end of this algorithm to obtain the following algorithm.

Approximation Algorithm (AA) *Given an initial partition P, an error functional e satisfying (4.19), and an error tolerance $\mu > 0$,* **AA** *produces as output an admissible partition $P' = \mathbf{AA}(P, \mu)$ which is a refinement of P in the following way. First it uses the* **Thresholding Second Algorithm** *to produce a tree T_μ which satisfies* **P1** *and* **P2**. *We can write $T_\mu = T(\bar{P})$ where $\bar{P}$ is the partition formed by the leaves of T_μ. We now apply completion to $\bar{P}$ as described in Lemma 2.5 to get the admissible partition P' from $\bar{P}$.*

The following lemma describes the properties of this algorithm.

Lemma 4.2 *The output $P' = \mathbf{AA}(P, \mu)$ satisfies*

$$E(T(P')) \le \mu. \tag{4.26}$$

In addition, there are absolute constants $c_1, C_1 > 0$ such that whenever $\tilde{P}$ is any refinement of P which satisfies $E(T(\tilde{P})) \le c_1\mu$, then we have

$$m(P'|P) \le C_1 N(T(\tilde{P})|T(P)). \tag{4.27}$$

Also, the number of evaluations $N(\mathbf{AA}, P, \mu)$ of e needed to compute the output P' of **AA** *satisfies*

$$N(\mathbf{AA}, P, \mu) \le 2(\#(\bar{P})) \le 2\#(P'). \tag{4.28}$$

Proof. The subadditivity property (4.19) guarantees that the error for the admissible tree $T(P')$ is no larger than that of T_μ. Hence (4.26) follows from (4.23). If $\tilde{P}$ is any refinement of P which satisfies $E(T(\tilde{P})) \le c_1\mu$, then by virtue of (4.25),

$$N(T(\bar{P})|T(P)) \le C_1 N(T(\tilde{P})|T(P)). \tag{4.29}$$

That is the number of subdivisions needed to create $\bar{P}$ from P does not exceed the right side of (4.29). By definition, $m(P'|P) = N(T(\bar{P})|T(P))$. Finally, (4.28) is obvious since the only values of e that need to be computed in the execution of the **Second Thresholding Algorithm** are those on the nodes of $T(\bar{P})$ and the number of these nodes does not exceed $2\#(\bar{P})$. □

4.4 Approximation of the right hand side

We shall describe in this section an algorithm for approximating the right hand side f whenever $f \in L_2(\Omega)$. This algorithm will be a step in our adaptive algorithm for solving (1.1). We shall approximate f by piecewise constants on admissible partitions that are adaptively generated using newest vertex bisection. To achieve this approximation we shall use the adaptive approximation algorithm **AA** of §4.3. Let P_0 be any fixed initial partition and let T_* denote the master tree for newest vertex bisection. As before, we let $\mathcal{P}^a$ denote the class of all admissible partitions that can be obtained using newest vertex bisection with initial partition P_0.

For each $\Delta \in T_*$, we define

$$e(f, \Delta) = e(\Delta) := |\Delta| \|f - f_\Delta\|^2_{L_2(\Delta))}. \tag{4.30}$$

The subadditivity property (4.19) follows easily from the fact that f_Δ is the best $L_2(\Delta)$ approximation to f by constants.

The following statements describe our algorithm for approximating f and its properties:

APPROX$(f, P, \mu) \to P'$: *The input of this algorithm is the function $f \in L_2(\Omega)$, an admissible partition $P \in \mathcal{P}^a$, and the error tolerance $\mu > 0$. The algorithm uses the inputs f, P and μ^2 in algorithm* **AA** *with the functional e defined by* (4.30) *and receives the output partition $P' =$* **APPROX**$(f, P, \mu) :=$ **AA** (P, μ^2) *which, by the properties of* **AA**, *is admissible and satisfies:*

$$\bar{E}(f, P') = E(T(P')) \le \mu^2. \tag{4.31}$$

From the properties of **AA** we immediately have the following facts, see Lemma 4.2.

Proposition 4.3 *If $\tilde{P}$ is any partition which is a refinement of P and satisfies $\bar{E}(f, \tilde{P}) \le c_1 \mu^2$, then*

$$m(P'|P) \le C_1 N(T(\tilde{P})|T(P)), \tag{4.32}$$

where c_1, C_1 are the constants associated to **AA**. *The number of evaluations $N($***APPROX***$, f, P, \mu)$ of e needed to compute the output P' to* **APPROX** *satisfies*

$$N(\textbf{APPROX}, f, P, \mu) \le 2(\#(P')). \tag{4.33}$$

The following lemma describes how we shall use **APPROX**.

Lemma 4.4 *If we apply* **APPROX** *to an input function $f \in \bar{\mathcal{A}}^s = \bar{\mathcal{A}}^s(H^{-1}(\Omega))$, $s > 0$, an input partition P, and an input tolerance $\mu > 0$, then the output partition $P' =$* **APPROX**(f, P, μ) *satisfies*

$$m(P'|P) \le C_1(s) \|f\|^{1/s}_{\bar{\mathcal{A}}^s} \mu^{-1/s} \tag{4.34}$$

with $C_1(s)$ a constant depending only on s. Also,

$$\#(P') \leq C_2(s)(\#(P) + \|f\|_{\bar{\mathcal{A}}^s}^{1/s}\mu^{-1/s}) \tag{4.35}$$

with $C_2(s)$ a constant depending only on s.

Proof. Let P^* be the smallest partition in $\mathcal{P}$ which satisfies

$$\bar{E}(f, P^*) \leq c_1\mu^2. \tag{4.36}$$

where c_1 is the constant appearing in algorithm **AA**. Then, by the definition of the class $\bar{\mathcal{A}}^s$, we have

$$\#(P^*) \leq \|f\|_{\bar{\mathcal{A}}^s}^{1/s}(\sqrt{c_1}\mu)^{-1/s}. \tag{4.37}$$

We let $\tilde{P}$ be the smallest partition which is a common refinement of P and P^*. Then, $\tilde{P}$ can be obtained from P using at most $\#(P^*)$ refinements. Also, $\bar{E}(f, \tilde{P}) \leq c_1\mu^2$ because $\tilde{P}$ is a refinement of P^*. Therefore, the output partition P' of **APPROX** satisfies (4.32) which means that

$$m(P'|P) \leq C_1 N(T(\tilde{P})|T(P)) \leq C_1\|f\|_{\bar{\mathcal{A}}^s}^{1/s}(\sqrt{c_1}\mu)^{-1/s}. \tag{4.38}$$

This proves (4.34). To estimate the cardinality of P' we can use (2.16) and (4.37) to obtain

$$\begin{aligned}\#(P') &\leq \#(P_0) + C_2(m(P|P_0) + m(P'|P)) \\ &\leq C_2(\#(P) + C_1(s)\|f\|_{\bar{\mathcal{A}}^s}^{1/s}\mu^{-1/s})\end{aligned} \tag{4.39}$$

which gives (4.35). □

4.5 Coarsening

In this section, we describe our second application of the adaptive approximation algorithm. The setting is the following. Suppose that P is any admissible partition and u_P is the Galerkin solution to (1.1) on this partition. We assume that we have a function $\Phi(P)$ which provides an upper bound for the Galerkin error

$$|||u - u_P|||^2 \leq \Phi(P). \tag{4.40}$$

Such an upper bound Φ is given in §5 (see (5.2)).

The algorithm we shall construct in this subsection starts with the input of two admissible partitions P, P' with P' a refinement of P, and a tolerance μ for which we know the bounds

$$|||u - u_P|||^2 \leq \Phi(P) \leq \mu^2 \tag{4.41}$$

and

$$|||u - u_{P'}|||^2 \leq \Phi(P') \leq \alpha\mu^2. \tag{4.42}$$

At this stage, the constant $0 < \alpha < 1$ is arbitrary but fixed. We shall specify the value of α, and the two constants β and γ that follow, at the end of §7. For now the reader should think of these as arbitrary but fixed constants satisfying the relations we specify below.

With these inputs, we shall generate an admissible partition $\bar{P}$ which is a coarsening of P' such that a certain further refinement $r(\bar{P})$ satisfies

$$|||u - u_{r(\bar{P})}|||^2 \leq \beta\mu^2, \tag{4.43}$$

where $\alpha < \beta < 1$. The main property (not held by P') we will gain in the construction of $\bar{P}$ is that we will be able to favorably bound its number of elements. The reason we need the refinement $r(\bar{P})$ of $\bar{P}$ and not just $\bar{P}$ in (4.43) is a technical one that will be explained later in this section.

To find $\bar{P}$, we are going to use an adaptive approximation algorithm **SAA** which is a modification of **AA** that was introduced and analyzed in [2]. The algorithm provides an approximation to $u_{P'}$ on a coarser partition (namely $\bar{P}$) than P'. We will explain the algorithm **SAA** in more detail as we proceed.

We start by defining an error functional e for approximating functions $w \in H^1_0(\Omega)$ by functions from $\mathcal{S}_P$ with P an admissible partition. To do this, we introduce the *minimal ring* associated to a triangular cell $\Delta \in T_*$. Given any admissible partition P from $\mathcal{P}^a$ and $\Delta \in P$, we define

$$R(\Delta, P) := \bigcup_{\Delta' \in P, \Delta \cap \Delta' \neq \emptyset} \Delta' \tag{4.44}$$

which is the first ring about Δ. This ring depends on P. However, we can find a minimal ring about Δ which does not depend on P. Namely, we define

$$R_-(\Delta) := \bigcap_{P \in \mathcal{P}^a, \Delta \in P} R(\Delta, P) = \bigcup_{\Delta' \in P_-(\Delta)} \Delta' \tag{4.45}$$

where $P_-(\Delta)$ is the collection of cells from T_* which touch Δ and make up $R_-(\Delta)$.

Before returning to the definition of e that we will use, we take this opportunity to develop a little further the properties of the set $P_-(\Delta)$ because we will need these properties later in this section. Let m_0 be the maximal valence that can occur for any vertex v in any admissible partition. Because of the uniform boundedness from below of the angles in triangular cells for admissible partitions, it follows that m_0 is a fixed finite constant.

Lemma 4.5 *Suppose P is an admissible partition and $\Delta \in P$ has the label $(k+1, k+1, k)$. Any cell $\Delta' \in P$ such that $\Delta' \cap \Delta \neq \emptyset$ has label $(k+\nu, k+\nu, k+\nu-1)$ where $-\frac{m_0-2}{2} < \nu \leq \frac{m_0+1}{2}$.*

Proof. Any such Δ' shares a vertex v with Δ. This vertex is common to two edges of Δ, one, which we denote by ε_0, is labelled $k+1$, and the other, which we denote by ε_1, is labelled k, or $k+1$. We can assume that all other edges that share v arise when traversing the edges from ε_0 to ε_1 in a clockwise direction. We claim that the maximum label of any of the edges containing v is $k+\ell$ where $\ell \leq \frac{m_0+1}{2}$. To see this note first that the labels of edges that share a cell can differ by at most 1. Now suppose that the number of edges incident at v is r. Let ε be such an edge with label $k+\ell$. Stepping from ε_0 to ε, we need at least $\ell-2$ more edges between these two. Moving from ε to ε_1 at least another $\ell-2$ more edges. That makes a total count of at least $3+2(\ell-2) = 2\ell-1 \leq r$, and therefore $\ell \leq \frac{r+1}{2}$. This confirms the upper bound for v. To obtain the lower bound, suppose now that v is common to an edge ε with label $k-\ell$ and that again r edges meet at v. It takes at least ℓ intermediate edges to go from ε_0 to ε and at least $\ell-1$ edges to go from ε to ε_1. Thus, the count is at least $3+\ell+(\ell-1) = 2\ell+2 \leq r$ and therefore $\ell \leq \frac{r-2}{2}$. This means that $\ell \leq \frac{m_0-2}{2}$ which confirms the lower bound. □

For the formulation of the next lemma, we introduce the notion of a *full refinement*. Given an admissible partition P, by a full refinement of P we mean the subdivision of each cell in P into six cells using newest vertex bisection. These new cells consist of two grandchildren of Δ and four great grandchildren of Δ and result in the bisection of each original edge of Δ as well as the bisection of the new edge added when Δ is subdivided using newest vertex bisection (see Figure 6).

Lemma 4.6 *Assume that P is any admissible partition and $r(P)$ is the partition obtained from P by subjecting each cell in P to $\lceil \frac{m_0}{2} \rceil$ full refinements. Then $r(P)$ is admissible and any ring $R(\Delta', r(P))$ of a cell $\Delta' \in r(P)$ is contained in the minimal ring $R_-(\Delta)$ of the ancestor $\Delta \in P$ of Δ'.*

Proof. The admissibility of $r(P)$ follows from the fact that one full refinement of every cell in an admissible partition leaves no hanging nodes. For the second property note that a full refinement to a triangular cell with label $(k+1, k+1, k)$ produces cells whose new edges carry labels that are at

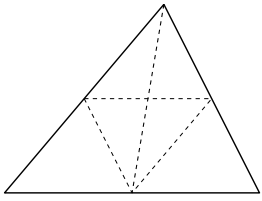

Fig. 6. Full refinement of a triangle

least $k+2$. Thus ℓ successive full refinements produce triangles with generations at least $k+2\ell$. Next note that the minimal ring of Δ consists of triangles touching Δ and having the highest possible generation admitted by T_*. We have shown in Lemma 4.5 that the minimal generation of cells in $R(\Delta, P)$ is greater than $k - \frac{m_0-2}{2} - 1 = k - \frac{m_0}{2}$. Thus, applying $\lceil \frac{m_0}{2} \rceil$ full refinements the smallest possible generation appearing in $R(\Delta', r(P))$ will increase to $k - \frac{m_0}{2} + 2\lceil \frac{m_0}{2} \rceil \geq k + \frac{m_0}{2}$. This is larger than the highest possible generation appearing in $R(\Delta, P)$ which, by Lemma 4.5, is at most $k + \frac{m_0+1}{2} - 1$. The proof is complete. □

We now return to our problem of coarsening. Given a function $w \in H_0^1(\Omega)$ (later we shall take w as an approximation to $u_{P'}$), we define

$$e(\Delta) := e(w, \Delta) := \inf_S \|w - S\|^2_{H^1(R_-(\Delta))} = \|w - S_\Delta\|^2_{H^1(R_-(\Delta))}, \tag{4.46}$$

where the infimum is taken over all continuous piecewise linear functions S defined on $R_-(\Delta)$ which are subordinate to $P_-(\Delta)$ and vanish on $\partial\Omega \cap \partial R_-(\Delta)$ whenever this set is not empty.

This error functional e will not satisfy (4.18) but does satisfy the following weaker version of this property:

Weak subadditivity: *For any $\Delta \in T_*$ and any* **admissible** *tree T which has Δ as its only root node, we have*

$$\sum_{\Delta' \in \mathcal{L}(T)} e(\Delta') \leq C\, e(\Delta) \tag{4.47}$$

where C is an absolute constant and $\mathcal{L}(T)$ is the set of leaves of T.

Thus the only distinction in this weak subadditivity is that (4.47) is only required to hold for admissible trees.

Proposition 4.7 *The local error functional e, defined by (4.46), satisfies the weak subadditivity property (4.47). Moreover, for $E = E(w, \cdot)$ defined by (4.20) for this choice of e, and any admissible partition $P \in \mathcal{P}^a$ one has for any $w \in H_0^1(\Omega)$*

$$E(w, T(P)) \leq C_4' \inf_{S \in \mathcal{S}_P} \|w - S\|^2_{H^1(\Omega)}, \tag{4.48}$$

where the constant C_4' depends only on the initial partition P_0 which consists of the roots of T_*.

Proof. Let T be a finite admissible tree with single root Δ. The leaves $\mathcal{L}(T)$ of T form an admissible partition of Δ. For each $\Delta' \in \mathcal{L}(T)$, we have that $P_-(\Delta')$ is a refinement of $P_-(\Delta)$ on the set $R_-(\Delta')$, and $R_-(\Delta') \subset R_-(\Delta)$. Therefore,

$$\|w - S_{\Delta'}\|_{H^1(R_-(\Delta'))} \leq \|w - S_\Delta\|_{H^1(R_-(\Delta'))}, \tag{4.49}$$

by the very definition of $S_{\Delta'}$.

Let us observe that a given point $x \in R_-(\Delta)$ appears in at most C of the sets $R_-(\Delta')$, $\Delta' \in \mathcal{L}(T)$, with C an absolute constant. Indeed, if $x \in \Delta^*$ with $\Delta^* \in \mathcal{L}(T)$, then, because of the minimality of the rings, x will not appear in any $R_-(\Delta')$ unless Δ' touches $R_-(\Delta^*)$. Since the partition $\mathcal{L}(T)$ is admissible, there are at most C cells which touch $R_-(\Delta^*)$ with C an absolute constant. We use this property and the set subadditivity of $\|\cdot\|^2_{H^1}$ to find,

$$\begin{aligned}\sum_{\Delta' \in \mathcal{L}(T)} \|w - S_{\Delta'}\|^2_{H^1(R_-(\Delta'))} &\leq \sum_{\Delta' \in \mathcal{L}(T)} \|w - S_\Delta\|^2_{H^1(R_-(\Delta'))} \\ &\leq C\|w - S_\Delta\|^2_{H^1(R_-(\Delta))}. \end{aligned} \tag{4.50}$$

This inequality verifies property (4.18) for this choice of e.

As for (4.48), let $w_P \in \mathcal{S}_P$ be the minimizer of the right side of (4.48) and note that only a finite uniformly bounded number of the minimal rings $R_-(\Delta)$, $\Delta \in P$, overlap at any given point in Ω. Hence,

$$\sum_{\Delta \in P} \|w - S_\Delta\|^2_{H^1(R_-(\Delta))} \leq \sum_{\Delta \in P} \|w - w_P\|^2_{H^1(R_-(\Delta))} \leq C\|w - w_P\|^2_{H^1(\Omega)},$$

as claimed. □

We cannot directly reverse the inequality (4.48) which is why we introduced the refinement $r(P)$ which will allow us to do the reversal. It is obvious that $r(P)$ is again admissible.

Proposition 4.8 *There exists a constant C_4 depending only on the initial partition P_0 such that for any admissible refinement $P \in \mathcal{P}^a$ of P_0, one has*

$$\inf_{S \in \mathcal{S}_{r(P)}} \|w - S\|^2_{H^1(\Omega)} \leq C_4 E(w, T(P)). \tag{4.51}$$

Proof. We shall employ quasi-interpolants for bounding the best approximation in the energy norm. To this end, recall that the Courant elements ϕ_v, $v \in \dot{\mathcal{V}}_{r(P)}$, form a basis for $\mathcal{S}_{r(P)}$. Thus each $S \in \mathcal{S}_{r(P)}$ has the unique representation

$$S = \sum_{v \in \dot{\mathcal{V}}_{r(P)}} S(v)\phi_v = \sum_{v \in \dot{\mathcal{V}}_{r(P)}} \lambda_v(S)\tilde{\phi}_v, \tag{4.52}$$

where the $\tilde{\phi}_v$ are normalized to have norm one in $H^1(\Omega)$, i.e. $\|\tilde{\phi}_v\|_{H^1(\Omega)} = 1$, and where the λ_v are dual functionals, i.e. $\lambda_v(\tilde{\phi}_{v'}) = \delta_{v,v'}$. This means that the λ_v also have norm one as linear functionals on $\mathcal{S}_{r(P)}$ when this space is equipped with the $H^1(\Omega)$ norm. There is a norm preserving extension λ_v (which we continue to denote by λ_v) to all of $H^1_0(\Omega)$ and we can require that this extension is given as the integral with an L_2 function Λ_v:

$$\lambda_v(g) = \int_{B_v} g\Lambda_v, \tag{4.53}$$

with

$$B_v := \operatorname{supp}(\Lambda_v) \subset \bigcup \{\Delta \in r(P) : v \in \Delta\}, \tag{4.54}$$

the *star* of v. It follows that

$$|\lambda_v(g)| \le \|g\|_{H^1(B_v)}. \tag{4.55}$$

Clearly, $Q_{r(P)}(w) := \sum_{v \in \dot{\mathcal{V}}_{r(P)}} \lambda_v(w)\tilde{\phi}_v$ defines a projector from $L_2(\Omega)$ into $\mathcal{S}_{r(P)}$. Let us now bound $\|w - Q_{r(P)}(w)\|_{H^1(\Omega)}$. Given any triangular cell $\Delta \in P$, we take an extension of S_Δ to a continuous piecewise linear function on Ω, vanishing on $\partial\Omega$ which is subordinate to $r(P)$. This is possible since $P_-(\Delta)$ is contained in $T(r(P))$. We denote this extension also by S_Δ. We then have $Q_{r(P)}(S_\Delta) = S_\Delta$ so that, in particular, both quantities agree on Δ. Therefore it follows that

$$\begin{aligned} \|w - Q_{r(P)}(w)\|^2_{H^1(\Delta)} &\le 2\|w - S_\Delta\|^2_{H^1(\Delta)} + 2\|Q_{r(P)}(w - S_\Delta)\|^2_{H^1(\Delta)} \\ &\le 2e(\Delta) + 2\|Q_{r(P)}(w - S_\Delta)\|^2_{H^1(\Delta)}. \end{aligned} \tag{4.56}$$

We need to bound the second term in (4.56). Let us note that on Δ only finitely many terms $\lambda_v(w - S_\Delta)\tilde{\phi}_v$ are nonzero. These correspond to the vertices from $r(P)$ that are in Δ and, in view of the fact that each full refinement subdivides each triangle into six and introduces four new vertices on it, there are at most $C = 3 + 4\left(6^{\lceil \frac{m_0}{2} \rceil} - 1\right)$ vertices from $r(P)$ in Δ. Let v be one of these vertices, then

$$\|\lambda_v(w - S_\Delta)\tilde{\phi}_v\|^2_{H^1(\Delta)} \le \|w - S_\Delta\|^2_{H^1(B_v)} \le \|w - S_\Delta\|^2_{H^1(R_-(\Delta))}, \tag{4.57}$$

where we have used that, by Lemma 4.6, $B_v \subset R_-(\Delta)$, $v \in \Delta \cap \dot{\mathcal{V}}_{r(P)}$. Hence, summing over all vertices $v \in \dot{\mathcal{V}}_{r(P)}$ in Δ, we obtain

$$\|Q_{r(P)}(w - S_\Delta)\|^2_{H^1(\Delta)} \le Ce(\Delta). \tag{4.58}$$

Using this back in (4.56) and summing over all $\Delta \in P$, gives

$$\|w - Q_{r(P)}(w)\|^2_{H^1(\Omega)} \le C_4\, E(w, T(P)). \tag{4.59}$$

This proves the assertion. □

Throughout the remainder of this section let E be defined by (4.20) for e, defined by (4.46) with w an arbitrary but fixed $H^1(\Omega)$ function. We return now to our problem of generating the set $\bar{P}$. For this we shall use the **Modified Second Algorithm** of §7 in [2]. This algorithm exactly matches our setting of newest vertex bisection. We recall the properties of this algorithm (see Corollary 7.4 of [2]).

Modified Second Algorithm (MSA): *Given any function* $w \in H_0^1(\Omega)$ *and any error tolerance* $\mu > 0$, **MSA** *produces as output an admissible partition* $P' = \mathbf{MSA}(w, \mu)$ *such that*

$$E(w, T(P')) \leq \mu. \tag{4.60}$$

Moreover, there are absolute constants $c_1, C_1 > 0$ *such that whenever* $\tilde{P}$ *is any admissible partition which satisfies* $E(w, T(\tilde{P})) \leq c_1\mu$, *then we have*

$$\#(P') - \#(P_0) = N(T(P')) \leq C_1 N(T(\tilde{P})) = C_1(\#(\tilde{P}) - \#(P_0)). \tag{4.61}$$

The number of evaluations $N(\mathbf{MSA}, w, \mu)$ *of* e *to compute the output* P' *to* **MSA** *satisfies*

$$N(\mathbf{MSA}, w, \mu) \leq C_3(\#(P')). \tag{4.62}$$

In contrast to **APPROX**, the algorithm **MSA** always starts from the root partition P_0. We could also have developed an algorithm which started from any given initial partition P but this would be at the expense of more technicalities which we wish to avoid.

We shall use the algorithm **MSA** in the following coarsening algorithm which is the main algorithm of this section. The constant γ in this algorithm will be specified later. The reader should think of γ as any fixed constant $0 < \gamma < 1$ until it is later specified. The algorithm **COARSE** will also use **GAL**.

COARSE: *We take as inputs the admissible partition* P', *the tolerance* μ *such that (4.42) holds, and an initial approximation* $\bar{u}_{P'} \in \mathcal{S}_{P'}$ *to* $u_{P'}$ *satisfying* $|||u_{P'} - \bar{u}_{P'}|||^2 \leq (A'\mu)^2$. *The algorithm* **COARSE** *outputs the admissible partition* $\bar{P} = \mathbf{COARSE}(P', \mu, \bar{u}_{P'})$ *as follows. We apply* **GAL** *to obtain the numerical approximation* $\hat{u}_{P'} = \mathbf{GAL}(P', \mu, \bar{u}_{P'})$. *We take* $w = \hat{u}_{P'}$ *and apply* **MSA** *with inputs* w *and* $\gamma\mu^2$ *to obtain* $\bar{P} = \mathbf{MSA}\,(w, \gamma\mu^2)$ *where* γ *will be specified below. It follows that*

$$E(\hat{u}_{P'}, T(\bar{P})) \leq \gamma\mu^2. \tag{4.63}$$

The rest of this subsection will be devoted to deriving the important properties of the output $\bar{P}$ of **COARSE**. Let us first observe that given P', the number of computations necessary to invoke the algorithm, is bounded by

$$N(\mathbf{COARSE}, P', \mu, \bar{u}_{P'}) \leq C_3(A'/\delta)\#(P') \tag{4.64}$$

with $C_3(A'/\delta)$ a constant depending only on the ratio A'/δ, see (3.6) and (3.7). We have already observed that the number of computations to calculate $\hat{u}_{P'}$ is bounded by the right side of (4.64) (see (3.8)). To compute S_Δ, $\Delta \in P'$, from the least squares problem uses a fixed number of computations because $\hat{u}_{P'}$ is continuous and piecewise linear and the number of cells in

$P_-(\Delta)$ has an absolute bound (see our discussion of the structure of $P_-(\Delta)$ given above). We have observed in (4.62) that the number of evaluations of e needed to execute **MSA** is bounded by $C_3\#(P')$. The estimate (4.64) follows now from Remark 3.1.

Next, we want to bound $|||u - u_{r(\bar{P})}|||$ for the Galerkin solution $u_{r(\bar{P})}$. To do this, we let S be a best approximation to $\hat{u}_{P'}$ from $\mathcal{S}_{r(\bar{P})}$ in the $H^1(\Omega)$ norm. Thus,

$$\begin{aligned} |||u - u_{r(\bar{P})}||| &\le |||u - S||| \le |||u - u_{P'}||| + |||u_{P'} - \hat{u}_{P'}||| + |||\hat{u}_{P'} - S||| \\ &\le (\sqrt{\alpha}\mu + \delta\mu) + \|\hat{u}_{P'} - S\|_{H^1(\Omega)} \\ &\le (\sqrt{\alpha} + \delta + \sqrt{C_4\gamma})\mu \le \sqrt{\beta}\mu \end{aligned} \tag{4.65}$$

where in the third inequality we used (4.42) on the first term and we used (3.7) for the second term. Then we use Poincaré's (i.e. (3.3)) followed by (4.51) on the remaining term. The last inequality will hold provided α, δ, and γ are chosen so as to satisfy

$$\alpha \le \beta/9 \quad \text{and} \quad \delta \le \sqrt{\beta}/3 \quad \text{and} \quad C_4\gamma \le \beta/9. \tag{4.66}$$

We impose these requirements on α, δ and γ in what follows.

We now summarize the properties of the output $\bar{P}$ of the algorithm **COARSE**.

Theorem 4.9 *Given as inputs the admissible partition P',an input tolerance μ, and an initial approximation $\bar{u}_{P'}$ the output $\bar{P}$ =***COARSE**$(P', \mu, \bar{u}_{P'})$ *has the following properties:*

(i) The Galerkin solution $u_{r(\bar{P})}$ on $r(\bar{P})$ satisfies:

$$|||u - u_{r(\bar{P})}|||^2 \le \beta\mu^2. \tag{4.67}$$

*(ii) The number of computations $N($***COARSE**$, P', \mu, \bar{u}_{P'})$ used to compute $\bar{P}$ satisfies*

$$N(\textbf{COARSE}, P', \mu, \bar{u}_{P'}) \le C_3(A'/\delta)(\#(P')) \tag{4.68}$$

with $C_3(A'/\delta)$ a constant depending only on A'/δ.

(iii) If $u \in \mathcal{A}^s = \mathcal{A}^s(H_0^1(\Omega))$, then

$$\#(\bar{P}) \le \#(P_0) + C_2(s)\|u\|^{1/s}_{\mathcal{A}^s(H_0^1(\Omega))}\mu^{-1/s} \tag{4.69}$$

with $C_2(s)$ a constant depending only on s.

Proof. Statement (i) is just (4.65). Statement (ii) follows from (4.64). The proof of (iii) is similar to the proof in Lemma 4.4. Let P^* be the smallest admissible partition that satisfies

$$|||u - u_{P^*}|||^2 \leq \|u - u_{P^*}\|^2_{H^1(\Omega)} \leq \hat{c}_1^2 \mu^2 \tag{4.70}$$

where $\hat{c}_1$ is a constant that will be specified later in this proof. From the fact that $u \in \mathcal{A}^s$, we know

$$\#(P^*) \leq \|u\|^{1/s}_{\mathcal{A}^s} (\hat{c}_1 \mu)^{-1/s}. \tag{4.71}$$

Moreover, note that u_{P^*} also approximates $\hat{u}_{P'}$ well. In fact, by (4.42) and (3.7),

$$|||\hat{u}_{P'} - u_{P^*}||| \leq |||u_{P'} - u||| + |||u_{P'} - \hat{u}_{P'}||| + |||u - u_{P^*}||| \leq (\sqrt{\alpha} + \delta + \hat{c}_1)\mu.$$

Therefore, from (4.48) and (3.3), we obtain

$$\begin{aligned} E(\hat{u}_{P'}, T(P^*)) &\leq C_4' \|\hat{u}_{P'} - u_{P^*}\|^2_{H^1(\Omega)} \\ &\leq c_\Omega^{-2} C_4' (\sqrt{\alpha} + \delta + \hat{c}_1)^2 \mu^2 \leq c_1 \gamma \mu^2 \end{aligned} \tag{4.72}$$

provided

$$\alpha \leq \frac{(C_4')^{-1} c_\Omega^2 c_1 \gamma}{9} \quad \text{and} \quad \delta, \hat{c}_1 \leq \frac{c_\Omega \sqrt{(C_4')^{-1} c_1 \gamma}}{3}. \tag{4.73}$$

We can now apply (4.61) and obtain

$$\begin{aligned} \#(\bar{P}) - \#(P_0) &\leq C_1(\#(P^*) - \#(P_0)) \leq C_1 \#(P^*) \\ &\leq C_2(s) \|u\|^{1/s}_{\mathcal{A}^s} \mu^{-1/s} \end{aligned} \tag{4.74}$$

where the last inequality uses (4.71). □

Let us remark on the order that all of the above constants are chosen. First, we are free to choose $0 < \beta < 1$. With the value of β fixed, we now choose γ so that the second inequality in (4.66) is satisfied. Then we choose α so that both the first inequality in (4.66) and the first inequality in (4.73) is satisfied. We also choose $\hat{c}_1$ so that the second inequality in (4.73) is satisfied. Further, we require that δ satisfies (4.73). Later in §7, we shall fix the value of β and then all other constants can be specified.

5 The adaptive algorithm of Morin, Nochetto, and Siebert (MNS)

In this section, we shall recall the AFEM of Morin, Nochetto, and Siebert [17] which is based on subdividing (using the newest vertex rule) certain collections of marked cells. We shall later modify this algorithm to arrive at our new algorithm. Our notation is slightly different from theirs in places.

The strategy in [17] for marking cells for subdivision and the guaranteed error reduction that results is based on two criteria. The first of these is a local error indicator that we now describe. Suppose that $P \in \mathcal{P}^a$ is an admissible partition. Given an edge $E \in \mathcal{E}_P$, we let Ω_E denote the union of the two triangles that share E and let ℓ_E denote the length of E. We also denote by $J_E(P)$ the jump of $n_E^T \nabla u_P$ across E, where for each cell sharing E, n_E is the respective outer normal of E.

The quantity

$$\eta_E := \eta_E(P) := \|\ell_E^{1/2} J_E(u_P)\|^2_{L_2(E)} + \|\mathrm{diam}(\Omega_E) f\|^2_{L_2(\Omega_E)} \tag{5.1}$$

is an indicator of the local error in $u - u_P$. One can show that the sum of these local indicators bounds the square of the global energy error from above. More precisely, there exists an absolute constant A_3 such that

$$|||u - u_P|||^2 \leq A_3 \sum_{E \in \mathcal{E}_P} \eta_E.$$

We therefore define the global quantity

$$\Phi(P) := A_3 \sum_{E \in \mathcal{E}_P} \eta_E. \tag{5.2}$$

The second ingredient that we shall need is a bound for the global H^{-1} error in approximating $f \in L_2(\Omega)$ by piecewise constants in terms of a sum of local quantities. We have already introduced these quantities in §4.4. We recall the local error functional

$$e(\Delta) := |\Delta| \|f - f_\Delta\|^2_{L_2(\Delta)}, \tag{5.3}$$

where f_Δ is the average of f on Δ. We also have the global error

$$\bar{E}(f, P) := \sum_{\Delta \in P} e(\Delta), \tag{5.4}$$

which was used in algorithm **AA**.

The local error indicators from (5.1) together with (5.4) can be used to prove the following bounds for the Galerkin error (see [17] equations (2.7) and (2.8)):

$$A_1 \Phi(P) - A_2 \bar{E}(f, P) \leq |||u - u_P|||^2 \leq \Phi(P), \tag{5.5}$$

where $A_1, A_2 > 0$ are absolute constants.

To describe the main result in [17], we denote for each $\Delta \in T_*$ by $G(\Delta)$ the set of six triangular cells obtained from Δ when using a full refinement as described in the paragraph preceding Lemma 4.6 (see Figure 6). The following theorem is Theorem 3.1 of [17]:

Theorem 5.1 *Given $0 < \theta < 1$, there exist constants $0 < \tau < 1$ and $A_4 > 0$ with the following property. Let P be any admissible partition in $\mathcal{P}^a$ and suppose that $\mathcal{E}_0$ is a collection of edges from $\mathcal{E}_P$ such that*

$$A_3 \sum_{E \in \mathcal{E}_0} \eta_E \geq \theta \Phi(P), \tag{5.6}$$

and

$$\bar{E}(f, P) \leq A_4 \mu^2. \tag{5.7}$$

If P' is any admissible refinement of P that contains $G(\Delta)$ or a refinement of $G(\Delta)$ for every Δ which has an edge in common with $\mathcal{E}_0$, then either $|||u - u_{P'}|||^2 \leq \mu^2$ or

$$|||u - u_{P'}|||^2 \leq \tau |||u - u_P|||^2. \tag{5.8}$$

In other words, if the error in approximating f is small enough then the refinement strategy (5.6) guarantees an error reduction until the desired tolerance μ is met. Using this result, Morin, Nochetto, and Siebert build an AFEM which they prove converges whenever $f \in L_2(\Omega)$.

Given a partition P and a marked set $\mathcal{E}_0$ of edges satisfying (5.6), we shall say that a partition P' which is a refinement of P has the G-property if it is admissible and P' contains $G(\Delta)$ or a refinement of $G(\Delta)$ whenever Δ has an edge in $\mathcal{E}_0$.

6 Modifications of the MNS algorithm

We shall make some modifications of the **MNS** algorithm in this section in order to prepare for its use in our new algorithm of the next section. These modifications will center on two issues. The first is to recast the algorithm in the form of elementary markings and then completions since it is in this form that we can apply the results of §2 which bound the cost of completions. The second issue is more substantial. In a numerical algorithm we cannot compute u_P or $\Phi(P)$ exactly. We shall have to replace these computations by approximate computations $\hat{u}_P$ and $\hat{\Phi}(P)$. We shall need to examine the effect of these approximate calculations. We shall show that it is possible to calculate these quantities approximately while still retaining the error reduction in the **MNS** algorithm. Moreover, we will be able to bound the computational complexity of these approximate calculations.

To begin the analysis of this section, we first note some properties of $\bar{E}$ and Φ. First of all if P' is a refinement of P then

$$\bar{E}(f, P') \leq \bar{E}(f, P). \tag{6.1}$$

This follows from the definition of $\bar{E}$ (4.15). Secondly,

$$\bar{E}(f, P) \leq (A_1/2A_2)\Phi(P) \quad \text{implies} \quad \Phi(P) \leq A_5 |\!|\!| u - u_P |\!|\!|^2 \tag{6.2}$$

with $A_5 = 2/A_1$. This follows from the lower inequality in (5.5).

These two observations will be used several times in the following form.

Remark 6.1 Assume that for some admissible partition P we have

$$A_5 |\!|\!| u - u_P |\!|\!|^2 \leq \mu^2 \quad \text{and} \quad \bar{E}(f, P) \leq \frac{A_1}{2A_2}\mu^2. \tag{6.3}$$

Then for any admissible refinement P' of P one has

$$\Phi(P') \leq \mu^2. \tag{6.4}$$

Proof. Suppose that $\Phi(P') > \mu^2$. Then, by (6.1),

$$\frac{A_1}{2A_2}\Phi(P') > \frac{A_1}{2A_2}\mu^2 \geq \bar{E}(f, P) \geq \bar{E}(f, P').$$

Thus (6.2) applies and since P' is a refinement of P our assumption (6.3) yields

$$\Phi(P') \leq A_5 |\!|\!| u - u_{P'} |\!|\!|^2 \leq A_5 |\!|\!| u - u_P |\!|\!|^2 \leq \mu^2,$$

which is a contradiction and hence proves (6.4). □

Our first modification of **MNS** which we shall call **MMNS** is to recast it in the form of marking and completion and still have the property that the refined partition contains $G(\Delta)$ (we need this to be able to obtain the guaranteed reduction in Theorem 5.1). Here is a description of

MMNS$(P, \mu) \to P'$*: This algorithm takes as input an admissible partition* P*, the corresponding Galerkin approximation* u_P *and an error tolerance* μ *with* P *satisfying* $\bar{E}(f, P) \leq A_4\mu^2$ *and produces a new partition* $P' =$ **MMNS**(P, μ) *as follows. We mark all triangular cells which have an edge from* $\mathcal{E}_0$ *where* $\mathcal{E}_0$ *are the marked edges from* **MNS**. *Call this marked set* $\mathcal{M}_0$. *We perform a refinement and then a completion for* $\mathcal{M}_0$ *and* P *giving the new partition* P'_1. *We next mark any cells* Δ' *in* P'_1 *which have a proper descendent which is one of the cells in* $G(\Delta)$, $\Delta \in \mathcal{M}_0$. *Call this set of marked cells* $\mathcal{M}_1$. *We do a refinement and completion for* $\mathcal{M}_1$ *and* P'_1 *giving the new partition* P'_2. *We repeat this process one more time ending with the partition* $P' = P'_3$.

Here are the properties of **MMNS**.

Theorem 6.2 *For the output P' of* **MMNS** *we have that either $|||u - u_{P'}||| \leq \mu$ or $|||u-u_{P'}|||^2 \leq \tau |||u-u_P|||^2$. The number of subdivisions $N(\mathbf{MMNS}, P, \mu)$ needed in* **MMNS** *satisfies*

$$N(\mathbf{MMNS}, P, \mu) \leq C_3 \#(P), \tag{6.5}$$

with C_3 an absolute constant.

Proof. For each cell Δ corresponding to a marked edge in **MNS**, the partition P' contains a refinement of the full refinement $G(\Delta)$. Hence, Theorem 5.1 applies and we obtain the stated bounds on $|||u - u_{P'}|||$. As for (6.5), the number of cells in $\mathcal{M}_0$ does not exceed $\#(P)$ and hence $\#(P_1') \leq C\#(P)$ (see (2.16)). Repeating this argument we derive (6.5). □

Our next modifications of **MNS** center around the fact that we cannot compute u_P or $\Phi(P)$ exactly. Therefore, we shall use the approximation $\hat{u}_P := \mathbf{GAL}(P, \mu, \bar{u}_P)$. We define the local error estimators $\hat{\eta}_E$ as in (5.1) with $\hat{u}_P$ in place of u_P and similarly define $\hat{\Phi}(P)$ as in (5.2) with these new $\hat{\eta}_E$ in place of η_E. A standard trace argument shows that

$$|\eta_E - \hat{\eta}_E| \leq B_1' \|u_P - \hat{u}_P\|^2_{H^1(\Omega_E)}, \tag{6.6}$$

where B_1' is some fixed constant depending only on P_0. As a consequence one has for another constant $B_1 \geq 1$

$$A_3 \sum_{E \in \mathcal{E}_P} |\hat{\eta}_E - \eta_E| \leq B_1 \delta^2 \mu^2. \tag{6.7}$$

With this numerical version $\hat{\Phi}$ in hand, we can revisit the algorithm **MMNS** and obtain the corresponding numerical version of this algorithm which we denote by **NMMNS**.

NMMNS$(P, \mu, \bar{u}_P) \rightarrow P'$*: This algorithm takes as input an admissible partition P, an error tolerance μ with P satisfying $\bar{E}(f, P) \leq A_4\mu^2$, and the numerical approximation $\hat{u}_P := \mathbf{GAL}(P, \mu, \bar{u}_P)$, and produces a new partition $P' =$* **NMMNS**$(P, \mu, \bar{u}_P)$ *as follows. We let $\hat{\mathcal{E}}_0$ be any set of edges in $\mathcal{E}_P$ such that*

$$A_3 \sum_{E \in \hat{\mathcal{E}}_0} \hat{\eta}_E \geq \theta \hat{\Phi}(P). \tag{6.8}$$

We mark all triangular cells which have an edge from $\hat{\mathcal{E}}_0$ and call this marked set $\mathcal{M}_0$. We perform a refinement and then a completion for $\mathcal{M}_0$ and P giving the new partition P_1'. We next mark any cells Δ' in P_1' which have a proper descendent which is one of the cells in $G(\Delta)$, $\Delta \in \mathcal{M}_0$. Call this set of marked cells $\mathcal{M}_1$. We do a refinement and completion for $\mathcal{M}_1$ and P_1' giving the new partition P_2'. We repeat this process one more time ending with the partition $P' = P_3'$.

Here are the properties of **NMMNS**.

Theorem 6.3 *If $\delta^2 \le \theta(4B_1)^{-1}$, then for the output P' of* **NMMNS** *we have that either $\|\!|u - u_{P'}|\!\| \le \mu$ or $\|\!|u - u_{P'}|\!\|^2 \le \tau \|\!|u - u_P|\!\|^2$ where $0 < \tau < 1$, depends on θ. The number of subdivision and computations $N(\mathbf{NMMNS}, P, \mu)$ needed in* **NMMNS** *satisfies*

$$N(\mathbf{NMMNS}, P, \mu, \bar{u}_P) \le C_3(A'/\delta)\#(P), \tag{6.9}$$

whenever the initial guess $\bar{u}_P$ satisfies (3.6).

Proof. From (6.7), we have that

$$\theta(\Phi(P) - B_1\delta^2\mu^2) \le \theta\hat{\Phi}(P) \le A_3 \sum_{E\in\hat{\mathcal{E}}_0} \hat{\eta}_E \le A_3 \sum_{E\in\hat{\mathcal{E}}_0} \eta_E + B_1\delta^2\mu^2. \tag{6.10}$$

We consider two cases:
Case: $\|\!|u - u_P|\!\| \le \mu^2$*:* Since P' is a refinement of P, we have $\|\!|u - u_{P'}|\!\| \le \|\!|u - u_P|\!\|$ as desired.
Case: $\|\!|u - u_P|\!\| > \mu^2$ In this case $\Phi(P) \ge \mu^2$ and $2B_1\delta^2 \le \theta/2$. Therefore (6.10) gives

$$(\theta/2)\Phi(P) \le A_3 \sum_{E\in\hat{\mathcal{E}}_0} \eta_E. \tag{6.11}$$

We can therefore apply Theorem 5.1 (with $\theta/2$ used in place of θ) and obtain $\|\!|u - u_{P'}|\!\| \le \mu^2$ or $\|\!|u - u_{P'}|\!\| \le \tau\|\!|u - u_P|\!\|$ with the τ corresponding to $\theta/2$.

The bound on $N(\mathbf{NMMNS}, P, \mu, \bar{u}_P)$ follows as in the proof of (6.5) with the caveat that we need to add the number of computations used in **GAL**. But these are bounded by $C_3(A'/\delta)\#(P)$ because of (3.8) in Remark 3.1. □

The following algorithm **REDUCE** is the main algorithm of this section. In this algorithm, there appears a constant A^* which will be specified later. Its role is to make sure that the term $\bar{E}(f, P')$ appearing in (5.5) is negligible compared to $\Phi(P')$ for all of the partitions P' that are encountered in the execution of the Algorithm.

REDUCE$(P, \mu, \mu', \bar{u}_P) \to P'$*: This algorithm takes as input any tolerances $0 < \mu' < \mu$, any admissible partition P for which we know*

$$\Phi(P) \le \mu^2, \quad \bar{E}(f, P) \le A^*\mu^2 \tag{6.12}$$

and any initial approximation $\bar{u}_P \in \mathcal{S}_P$ (to be used in **GAL***) satisfying (3.6). It outputs a new partition $P' =$* **REDUCE**$(P, \mu, \mu', \bar{u}_P)$ *which is a refinement of P and satisfies*

$$\Phi(P') \le (\mu')^2, \quad \bar{E}(f, P') \le A^*(\mu')^2. \tag{6.13}$$

Here A^* is any constant satisfying

$$A^* \leq \min\left\{\frac{A_4}{A_5}, \frac{A_1}{2A_2}\right\}\left(\frac{\mu'}{\mu}\right)^2, \tag{6.14}$$

where A_1, A_2, A_4 and A_5 are the constants from (5.5), (5.7) and (6.2), respectively.

We describe now the steps used to produce P'. We take $\mu'' = \mu' A_5^{-1/2}$ and use **NMMNS** with the inputs P and μ'' and $\bar{u}_P$. The result is a partition $\tilde{P}_1 = \mathbf{NMMNS}(P, \mu'', \bar{u}_P)$ which is a refinement of P. We reapply **NMMNS** with inputs μ'', $\tilde{P}_1$, and the same $\bar{u}_P$ as the initial approximation to $u_{\tilde{P}_1}$. We obtain the output $\tilde{P}_2 = \mathbf{NMMNS}(\tilde{P}_1, \mu'', \bar{u}_P)$. We repeat this k times (each time using the same $\bar{u}_P$ as an initial approximation) where k is chosen as the smallest integer so that $\tau^k \leq (A_5)^{-1}(\mu'/\mu)^2$. We now apply **APPROX** with the input $\tilde{P}_k$ and $\sqrt{A^*}\mu'$ to get the partition $P' = \mathbf{REDUCE}(P, \mu, \mu', \bar{u}_P)$.

Let us now check that P' satisfies (6.13). First of all, we know that the second inequality in (6.13) is valid because of the **APPROX** application. So we need to verify only the first inequality. Note that (6.14) implies, in particular, that $A^* \leq A_4(\mu''/\mu)^2$ so that Theorem 6.3 is applicable to each iteration. So we have two possibilities. The first case is that $|||u - u_{\tilde{P}_j}||| \leq \mu''$ for one of the values of $j \in \{1, \ldots, k\}$. Since P' is a refinement of P_j, we have

$$A_5|||u - u_{P'}|||^2 \leq A_5|||u - u_{\tilde{P}_j}|||^2 \leq A_5(\mu'')^2 \leq (\mu')^2. \tag{6.15}$$

In view of the second relation in (6.13) and the condition (6.14) on A^*, we can invoke Remark 6.1 (with P replaced by P') to conclude that the first relation in (6.13) is also valid in this case.

The other case is that the error is reduced by the factor τ at each application of **NMMNS** which gives

$$\begin{aligned} A_5|||u - u_{P'}|||^2 &\leq A_5|||u - u_{\tilde{P}_k}|||^2 \\ &\leq A_5\tau^k|||u - u_P|||^2 \leq A_5\tau^k\Phi(P) \leq (\mu')^2, \end{aligned} \tag{6.16}$$

which, again by Remark 6.1 confirms the validity of the first inequality in (6.13).

The following proposition bounds the number of subdivisions/computations used to produce P'.

Proposition 6.4 *The number of elements $\#(P')$ satisfies*

$$\#(P') \leq \tilde{C}_3(\mu/\mu')\#(P). \tag{6.17}$$

The number $N(\mathbf{REDUCE}, P, \mu, \mu', \bar{u}_P)$ of computations used to produce P' also satisfies

$$N(\mathbf{REDUCE}, P, \mu, \mu', \bar{u}_P) \leq C_3(\rho, \mu/\mu')\#(P), \tag{6.18}$$

where ρ is the maximal ratio A'/δ for all constants A' from (3.6) encountered in the calls of **GAL** *invoked in* **REDUCE**.

Proof. From Theorem 6.3, we have that $\#(\tilde{P}_1) \leq C\#(P)$ and $\#(\tilde{P}_{j+1}) \leq C\#(P_j)$, for $j = 1, \ldots, k-1$. Since k is fixed, this gives the bound (6.17). Regarding the number of computations, we know that the number of calculations needed in **GAL** to compute $\hat{u}_P$ from $\bar{u}_P$ does not exceed $C\#(P)$ with the constant depending only on Δ and the ratio μ/μ'. The number of calculations in each iteration of **NMMNS** uses at most $C(\rho)\#(\tilde{P}_j)$ calculations as was shown in Theorem 6.3. Therefore, (6.18) holds in this case as well. □

7 The Main loop

We are now in a position to build the main iteration of our adaptive algorithm. This loop will use the **REDUCE** algorithm to generate a partition P' and then follow it with a coarsening step whose sole purpose is to give a control on the size of the output partition. The description of this algorithm will serve to set the value of the constants $\alpha, \beta, \delta, \gamma, A^*$ which have appeared earlier but were left unspecified. We call the algorithm for the main loop **MAIN**.

MAIN$(P, \mu, \bar{u}_{r(P)}) \to \widetilde{P}$*: This algorithm takes as input a tolerance μ, an admissible partition P, and an initial approximation $\bar{u}_{r(P)}$ to $u_{r(P)}$ satisfying (3.6) and*

$$\Phi(r(P)) \leq \mu^2 \tag{7.1}$$

and

$$\bar{E}(f, P) \leq A^*\mu^2. \tag{7.2}$$

It outputs an admissible partition $\tilde{P} :=$ **MAIN** $(P, \mu, \bar{u}_{r(P)})$ *which is a refinement of P and satisfies*

$$\Phi(r(\tilde{P})) \leq \mu^2/2 \tag{7.3}$$

and

$$\bar{E}(f, \tilde{P}) \leq A^*\mu^2/2. \tag{7.4}$$

Here A^* is the constant appearing in **REDUCE** *with $\mu' := \sqrt{\alpha}\mu$ and hence is assumed to satisfy*

$$A^* \leq \alpha \ \min\left\{\frac{A_4}{A_5}, \frac{A_1}{2A_2}\right\}. \tag{7.5}$$

We now describe the steps in **MAIN**.

Step 1: Apply **REDUCE** with inputs $r(P)$, μ, and $\mu' = \sqrt{\alpha}\mu$ with α the constant in **COARSE**, see (4.42) and with $\bar{u}_{r(P)}$ used as the initial approximation in any application of **GAL**. The output $P' =$ **REDUCE** $(r(P), \mu, \sqrt{\alpha}\mu, \bar{u}_{r(P)})$ is an admissible partition which satisfies (6.13) and in particular $\Phi(P') \leq \alpha\mu^2$.

Step 2: Apply **COARSE** with inputs P, P', μ, and $\bar{u}_{P'} := \bar{u}_{r(P)}$ (note that P' is a refinement of $r(P)$ and therefore $u_{r(P)}$ is in $\mathcal{S}_{P'}$), and obtain the output $\bar{P} = \mathbf{COARSE}(P', \mu, \bar{u}_{r(P)})$ which, on account of (4.67), satisfies

$$|||u - u_{r(\bar{P})}|||^2 \leq \beta\mu^2. \tag{7.6}$$

Step 3: Define $\bar{\bar{P}}$ to be the common refinement of P and $\bar{P}$.

Step 4: Apply **APPROX** with input $\bar{\bar{P}}$ and tolerance $(A^*/2)^{1/2}\mu$ and obtain as output the partition $\tilde{P} = \mathbf{APPROX}(\bar{\bar{P}}, (A^*/2)^{1/2}\mu)$ which satisfies $\bar{E}(f, \tilde{P}) \leq A^*\mu^2/2$.

Let us now check that the conclusions stated in **MAIN** are indeed valid. The condition (7.4) is an obvious consequence of *Step 4* in **MAIN**. Furthermore, (7.5) enables the application of **REDUCE** in *Step 1*. So it remains to verify (7.3). The desired bound on Φ will follow again from Remark 6.1 whose applicability will hinge on the choice of A^* and of the parameter β which has yet to be fixed. By (7.5) and (7.4), we have $(A_1/2A_2)\mu^2/2 \geq A^*\mu^2/2 \geq \bar{E}(f, \tilde{P})$. Since $\tilde{P}$ is a refinement of $\bar{P}$, we have

$$A_5|||u - u_{r(\tilde{P})}|||^2 \leq A_5|||u - u_{r(\bar{P})}|||^2 \leq A_5\beta\mu^2, \tag{7.7}$$

where we have used (7.6) in the last step. Thus whenever $\beta \leq 1/(2A_5)$, Remark 6.1 applies and yields (7.3) (with $\bar{P}$ used for both P and P'). We impose this condition and thereby specify the value of β as

$$\beta = 1/(2A_5). \tag{7.8}$$

This in turn now allows the specification of α and γ through our earlier requirements (4.66) and (4.73).

Proposition 7.1 *There is a constant C_5 such that the output $\tilde{P} = \mathbf{MAIN}(P, \mu, \bar{u}_P)$ satisfies*

$$\#(\tilde{P}) \leq C_5\#(P) + C_2 m(\tilde{P}|\bar{\bar{P}}) \tag{7.9}$$

where $m(\tilde{P}|\bar{\bar{P}})$ is the number of markings used in creating $\tilde{P}$ from $\bar{\bar{P}}$. (Here C_2 is the constant of (2.16).) Also, the number of flops $N(\mathbf{MAIN}, P, \mu, \bar{u}_{r(P)})$ used in computing $\tilde{P}$ is bounded by the right side of (7.9) times a constant $C(\rho)$ that depends only on the maximum of the ratios $\rho = A'/\delta$ where A' measures the accuracy of the initial guess in each call of **GAL**, *see (3.6). In the case that $f \in \bar{\mathcal{A}}^s(H^{-1})$ and $u \in \mathcal{A}^s = \mathcal{A}^s(H_0^1(\Omega))$, we have the inequality*

$$\#(\tilde{P}) \leq \#(P) + C_5(s)(\|u\|^{1/s}_{\mathcal{A}^s}\mu^{-1/s} + \|f\|^{1/s}_{\bar{\mathcal{A}}^s}\mu^{-1/s}) \tag{7.10}$$

and

$$N(\mathbf{MAIN}, P, \mu, \bar{u}_{r(P)}) \le C_5(\rho, s)\left\{\#(P_0) + (\|f\|_{\bar{\mathcal{A}}^s}^{1/s} + \|u\|_{\mathcal{A}^s}^{1/s})\mu^{-1/s}\right\} \tag{7.11}$$

with $C_5(\rho, s)$ a constant depending only on s and the maximum of the ratios A'/δ corresponding to the accuracies of the initial guesses in the calls of **GAL** *in* **REDUCE** *and* **COARSE**.

Proof. Since μ/μ' is fixed, (6.17) ensures that the output P' of **REDUCE** in *Step 1* of **MAIN** satisfies

$$\#(P') \le C_3\#(P).$$

Of course, the number of elements in $\bar{P}$ is also bounded by this number since $\bar{P} \subset P'$. Since $\bar{\bar{P}}$ is a common refinement of P and $\bar{P}$, it also satisfies $\#(\bar{\bar{P}}) \le C\#(P)$. If $m(\tilde{P}|\bar{\bar{P}})$ is the number of markings used in the creation of $\tilde{P}$ from $\bar{\bar{P}}$ in the application of **APPROX** in *Step 4*, then by (2.16), we have

$$\#(\tilde{P}) \le \#(P_0) + C_2(m(\bar{\bar{P}}|P_0) + m(\tilde{P}|\bar{\bar{P}})) \le C_4\#(P) + C_2 m(\tilde{P}|\bar{\bar{P}}). \tag{7.12}$$

This proves (7.9). Moreover, by Remark 3.1 the number of flops required by the various calls of **GAL** remains proportional to the cardinalities of the respective partitions times a factor $C(\rho)$ that depends only on the accuracy quotients $\rho = A'/\delta$ relating the output accuracy to that of the initial guess. For each of the algorithms **REDUCE**, **COARSE**, and **APPROX**, we have therefore shown that the number of computations is bounded by the maximum cardinalities of the inputs and outputs of these algorithms times $C(\rho)$, see (6.18), (6.9) and (4.68). Thus the right side of (7.9) multiplied by $C(\rho)$ also provides an upper bound for the number of computations used in **MAIN**.

Now suppose that $f \in \bar{\mathcal{A}}^s(H^{-1})$ and $u \in \mathcal{A}^s$. By (4.69), we have

$$\#(\bar{P}) \le \#(P_0) + C_2(s)\|u\|_{\mathcal{A}^s}^{1/s}\mu^{-1/s} \tag{7.13}$$

and so

$$\#(\bar{\bar{P}}) \le \#(P) + C_2(s)\|u\|_{\mathcal{A}^s}^{1/s}\mu^{-1/s}. \tag{7.14}$$

When we now apply *Step 4*, we can use (4.34) to obtain the bound

$$m(\tilde{P}|\bar{\bar{P}}) \le C_1(s)\|f\|_{\bar{\mathcal{A}}^s}^{1/s}\mu^{-1/s}. \tag{7.15}$$

Hence, using (2.16), we obtain

$$\begin{aligned}\#(\tilde{P}) &\le \#(\bar{\bar{P}}) + C_2(m(\tilde{P}|\bar{\bar{P}}))\\ &\le \#(P) + C_5(s)(\|u\|_{\mathcal{A}^s}^{1/s}\mu^{-1/s} + \|f\|_{\bar{\mathcal{A}}^s}^{1/s}\mu^{-1/s})\end{aligned} \tag{7.16}$$

where the last inequality uses (7.14) and (7.15). This gives (7.10). The proof of (7.11) is similar but takes in addition the cost (4.68), (6.5) and (6.18) of determining approximate Galerkin solutions in **REDUCE** and **COARSE**. □

8 An adaptive algorithm with coarsening

In this section, we shall formulate our adaptive algorithm for solving (1.1) and prove the convergence properties of this algorithm. We shall call this algorithm **ALG**. We recall the numerical approximation $\hat{\Phi}(P)$ of $\Phi(P)$ which is computed through the aid of the numerical approximation $\hat{u}_P$ to u_P. Regarding the constraints on the accuracy of this approximation reflected by the parameter δ in (3.7), note that $\delta \leq \sqrt{\alpha}$ implies the validity of both conditions in (4.66) and (4.73) on δ. Thus we fix

$$\delta^2 = \min\{\alpha, \theta/(4B_1)\}. \tag{8.1}$$

so that, in particular, $B_1\delta^2 \leq 1/4$. If μ is the input tolerance in **GAL**, then from (6.7), we have

$$|\Phi(P) - \hat{\Phi}(P)| \leq B_1\delta^2\mu^2. \tag{8.2}$$

ALG$(\varepsilon, P_0) \to (P, u_P)$*: This algorithm takes as input a desired tolerance* $\epsilon > 0$ *and an initial partition* P_0 *and outputs a partition* P *which satisfies*

$$\Phi(P) \leq \varepsilon^2/4, \tag{8.3}$$

(and hence, by (5.5), $|||u - u_P||| \leq \varepsilon/2$*), and it outputs a numerical approximation* $\hat{u}_P$ *to* u_P *which satisfies*

$$|||u - \hat{u}_P||| \leq \varepsilon \tag{8.4}$$

by using the following steps:

(I): *(Initialization) For the initial partition* P_0*, we take* $\bar{u}_{r(P_0)} := 0$ *as the initial approximation of* $u_{r(P_0)}$ *and* $\mu = 1$ *as the tolerance in* **GAL**. *We let* $\hat{u}_{r(P_0)} :=$ **GAL**$(r(P_0), 1, 0)$ *be the output of* **GAL** *for these inputs. We compute* $\hat{\Phi}(r(P_0))$ *using* $\hat{u}_{r(P_0)}$ *and we further compute* $\bar{E}(f, P_0)$. *We define*

$$\varepsilon_0^2 := \max\{\hat{\Phi}(r(P_0)) + 1/4, \bar{E}(f, P_0)/A^*\}. \tag{8.5}$$

We therefore have

$$\bar{E}(f, P_0) \leq A^*\varepsilon_0^2 \tag{8.6}$$

and by (8.2) and (8.1)

$$\Phi(r(P_0)) \leq \hat{\Phi}(r(P_0)) + 1/4 \leq \varepsilon_0^2. \tag{8.7}$$

If $\varepsilon_0 \leq \varepsilon/2$ *we stop and output* $P := r(P_0)$ *and* $\hat{u}_P := \hat{u}_{r(P_0)}$. *If not, let* n *be the smallest integer such that* $2^{-n}\varepsilon_0^2 \leq \varepsilon^2/4$ *and proceed to step* **(II)**.

(II): *For $k = 0, 1, \ldots, n-1$, we apply* **MAIN** *with inputs P_k and ε_k, satisfying $\Phi(r(P_k)) \le \varepsilon_k^2$ and $\bar{E}(f, P_k) \le A^* \varepsilon_k^2$, to obtain the partition $P_{k+1} =$* **MAIN**$(P_k, \varepsilon_k, \bar{u}_{r(P_k)})$. *In the application of* **MAIN**, *we use as an initial guess $\bar{u}_{r(P_k)} := \hat{u}_{r(P_0)}$ when $k = 0$, and $\bar{u}_{r(P_k)} := \hat{u}_{r(P_{k-1})}$ when $k > 0$, since the latter function has already been computed in the application of* **REDUCE** *in* **MAIN** *at the previous stage. By (7.3) and (7.4) we know that $\Phi(r(P_{k+1})) \le \varepsilon_k^2/2$, $\bar{E}(f, P_{k+1}) \le A^* \varepsilon_k^2/2$.*

(III): *Define $P = r(P_n)$, and compute $\hat{u}_P :=$* **GAL**$(P, \varepsilon, \hat{u}_{r(P_{n-1})})$.

The following is the main result of this paper.

Theorem 8.1 *For any function $f \in L_2(\Omega)$ and any $\varepsilon > 0$,* **ALG** *produces a partition P for which*

$$|||u - u_P|||^2 \le \Phi(P) \le \varepsilon^2/4 \tag{8.8}$$

and

$$|||u - \hat{u}_P||| \le \varepsilon \tag{8.9}$$

If $s > 0$ and $u \in \mathcal{A}^s$, and $f \in \bar{\mathcal{A}}^s$, then

$$\#(P) \le C(s) \left(\#(P_0) + \|f\|_{\bar{\mathcal{A}}^s}^{1/s} + \|u\|_{\mathcal{A}^s}^{1/s} \right) \varepsilon^{-1/s} \tag{8.10}$$

with $C(s) > 0$ a constant depending only on s, $\|u\|_{H^1(\Omega)}$, and the initial partition P_0. Moreover, The number of computations used in producing P does not exceed $C(s)(\#(P_0) + \|f\|_{\bar{\mathcal{A}}^s}^{1/s} + \|u\|_{\mathcal{A}^s}^{1/s})\varepsilon^{-1/s}$.

Proof. The conclusion (8.8) follows immediately from the stopping criterion and the initialization step **(I)**. To show (8.9), we estimate

$$|||u - \hat{u}_P||| \le |||u - u_P||| + |||u_P - \hat{u}_P||| \le \varepsilon/2 + \delta\varepsilon \le \varepsilon \tag{8.11}$$

because $\delta \le 1/2$ (see (8.1)).

We now turn to bounding the number of elements in P and the number of operations used in the algorithm in terms of the target accuracy ε. To begin with, let us identify next the constants A' entering the estimate for the initial guesses in the various calls of **GAL**. In step **(I)** we have $\bar{u}_{r(P_0)} = 0$ so that

$$\begin{aligned} |||u - \bar{u}_{r(P_0)}|||^2 &= a(u,u) = \langle f, u \rangle \\ &\le \|u\|_{H^1(\Omega)} \|f\|_{H^{-1}(\Omega)} \le c_\Omega^{-1} |||u||| \, \|f\|_{H^{-1}(\Omega)}, \end{aligned}$$

so that in this case $A' \le c_\Omega^{-1} \|f\|_{H^{-1}(\Omega)}$.

At the kth stage of step **(II)** **GAL** is invoked several times by **NMMMS** in **REDUCE** and by **COARSE**. Each time $\bar{u}_{r(P_k)} = \hat{u}_{r(P_k)}$ is used as an initial

guess. In view of the accuracy tolerance of $\hat{u}_{r(P_{k-1})}$ chosen in the first call of **NMMNS** used in **REDUCE**, we have

$$\begin{aligned}|||u - \bar{u}_{r(P_k)}||| &\le |||u - u_{r(P_{k-1})}||| + |||\hat{u}_{r(P_{k-1})} - u_{r(P_{k-1})}||| \\ &\le \Phi(r(P_{k-1}))^{1/2} + \delta\sqrt{\frac{\alpha}{A_5}}\varepsilon_{k-1} \le \sqrt{2}\left(1 + \sqrt{\frac{\alpha}{4A_5}}\right)\varepsilon_k.\end{aligned}$$

In fact, the tightest accuracy tolerance used by **GAL** in **REDUCE**, or in **COARSE** is $\varepsilon_k\sqrt{\alpha}/\sqrt{A_5}$. The same argument applies for step **(III)**. Thus in all cases the ratio $\rho = A'/\delta$ is uniformly bounded. By Proposition 7.1 the computational cost of **GAL** remains proportional to the cardinality of the respective partition.

To verify now (8.10), suppose that $s > 0$ and $u \in \mathcal{A}^s$, and $f \in \bar{\mathcal{A}}^s$. Let n be the final value of k when the algorithm stops, i.e., $P = r(P_n)$. To start out with, we note first that $\varepsilon_0 \le C \max\{1, \|u\|_{H^1(\Omega)}, \|f\|_{L_2(\Omega)}\}$. In fact, if ε_0 is given by the second term on the right hand side of (8.5) we infer from (4.15) that $\varepsilon_0 \le C\|f\|_{L_2(\Omega)}$. When the first term on the right hand side of (8.5) dominates we distinguish two cases. If $\bar{E}(f, P_0)/A^* \le \Phi(r(P_0))$ we conclude from (6.1), (6.2) and (8.2) that $\varepsilon_0^2 - 1/2 = \hat{\Phi}(r(P_0)) - 1/4 \le \Phi(r(P_0)) \le A_5|||u|||^2$. The other case $\bar{E}(f, P_0)/A^* > \Phi(r(P_0)) \ge \varepsilon_0^2 - 1/2$ also leads to the bound claimed above. Subsequent ε_k will satisfy this same bound. From (7.16) we know that for $k = 1, 2, \ldots, n$, we have

$$\#(P_{k+1}) - \#(P_k) \le C_5(s)\left(\|u\|_{\mathcal{A}^s}^{1/s}\varepsilon_k^{-1/s} + \|f\|_{\bar{\mathcal{A}}^s}^{1/s}\varepsilon_k^{-1/s}\right). \tag{8.12}$$

Since by definition $\varepsilon_k \ge \varepsilon 2^{(n-1-k)/2}$ we have

$$\begin{aligned}\#(P_n) &= \#(P_0) + \sum_{k=0}^{n-1}(\#(P_{k+1}) - \#(P_k)) \\ &\le \#(P_0) + \frac{C_5(s)\varepsilon_0^{-1/s}}{1 - 2^{-1/2s}}\left(\|u\|_{\mathcal{A}^s}^{1/s} + \|f\|_{\bar{\mathcal{A}}^s}^{1/s}\right)\varepsilon^{-1/s},\end{aligned}$$

and since $P = r(P_n)$, we have $\#(P) \le C\#(P_n)$ which proves (8.10).

To bound the number of computations used in **ALG**, we know from (7.11) that the number of computations N_k used in computing P_k, $k = 1, \ldots, n$, is bounded by

$$N_k \le C_5(s)\left(\#(P_0) + \|u\|_{\mathcal{A}^s}^{1/s} + \|f\|_{\bar{\mathcal{A}}^s}^{1/s}\right)(\varepsilon_k)^{-1/s} \tag{8.13}$$

since, according to our initial comments, the quotient ρ is uniformly bounded and therefore the computational cost of each call of **GAL** remains proportional to the corresponding partition size. Similarly, the number of computations used in computing P_0 is bounded by a constant multiple of $\#(P_0)$. Thus the total number of computations does not exceed

$$C(s)\left(\#(P_0)+\|f\|_{\bar{\mathcal{A}}^s}^{1/s}+\|u\|_{\mathcal{A}^s}^{1/s}\right)\sum_{k=0}^{n}(\varepsilon_k)^{-1/s}$$

$$\leq C(s)\left(\#(P_0)+\|f\|_{\bar{\mathcal{A}}^s}^{-1/s}+\|u\|_{\mathcal{A}^s}^{-1/s}\right)\varepsilon^{-1/s}. \tag{8.14}$$

□

9 Appendix

Recall from (4.7) and (4.17) that the classes $\mathcal{A}^s(H^1(\Omega))$ and $\bar{\mathcal{A}}^s(H^{-1}(\Omega))$ are determined through the nonlinear approximation properties of their elements. For instance, the elements in $\mathcal{A}^s(H^1(\Omega))$ can be approximated in $H^1(\Omega)$ to accuracy ε on admissible partitions with the order of $\varepsilon^{-1/s}$ cells. Obviously not all of these elements can be approximated on uniform partitions with the same accuracy. In this section we wish to explain (without proofs) which properties make a function belong to $\mathcal{A}^s(H^1(\Omega))$, say. This amounts to relating the above approximation classes to *regularity*. The following results about the behavior of $\sigma_n(u)$, defined by (4.3), are in principle known. They invoke the Besov smoothness of functions. We refer the reader to any of the standard treatments of Besov spaces (e.g. [21,1,8]) for the definition of the Besov spaces $B_q^s(L_\tau(\Omega))$ and only remark here that such a space is a smoothness space consisting of functions with smoothness order s (number of derivatives) measured in L_τ. For example $B_2^s(L_2)$ is identical with H^s and $B_\infty^s(L_\tau)$ is a Lipschitz space in L_τ whenever s is not an integer. The role of q is secondary and only serves to give a fine grading of the spaces important in many applications such as embedding theorems.

The results about σ_n, defined by (4.3), can be formulated as follows.

Theorem 9.1 *If $u \in B_\tau^{\alpha+1}(L_\tau(\Omega))$ with $0 \leq \alpha \leq 1$ and $1/\tau < (\alpha+1)/2$, then*

$$\sigma_n(u) = \inf_{P\in\mathcal{P}_n}\inf_{S\in\mathcal{S}_P}|||u-S||| \leq C_0 n^{-\alpha/2}\|u\|_{B_\tau^{\alpha+1}(L_\tau)},$$

where the constant C_0 depends on the discrepancy $\delta := \frac{\alpha+1}{2}-\frac{1}{\tau}$ when δ tends to zero.

A few brief comments on the range of the involved parameters are in order.

Remark 9.2 The restriction on α arises because we are approximating in H^1 using piecewise linears and so $\alpha+1 \leq 2$. Thus in two spatial dimensions $N^{-1/2}$ is the highest attainable order in the class $B_\tau^2(L_\tau)$ with $\tau^{-1} < 1$. The restriction on τ arises from the Sobolev embedding theorem. It guarantees that the Besov space is embedded compactly in H^1. When $\tau^{-1} > (\alpha+1)/2$, this Besov space is no longer embedded in H^1.

The above regularity assumptions are only sufficient for u to belong to $\mathcal{A}^{\alpha/2}$, say. The following *inverse theorem* shows that, although this is not a complete characterization, it is sharp in the following sense.

Theorem 9.3 *If $u \in H^1(\Omega)$ satisfies $\sigma_n(u) \leq Cn^{-\alpha/2}$ then $u \in B_\tau^{\alpha+1}(L_\tau)$ for all τ satisfying $\tau^{-1} = (\alpha+1)/2$.*

The proofs of Theorems 9.1, 9.3 can be found in [3]. For related results see [14].

Finally, as a consequence of known results on the metric entropy of unit balls of Besov classes, the order of Besov smoothness limits the approximation order in the following sense.

Theorem 9.4 *For each $0 \leq \alpha \leq 1$ and $\tau^{-1} < (\alpha+1)/2$ we have*

$$\sup_{f \in U(B_\tau^{\alpha+1}(L_\tau(\Omega)))} \sigma_n(u) \geq CN^{-\alpha/2} \tag{9.1}$$

with $C > 0$ an absolute constant. (Here $U(X)$ denotes the unit ball of a normed space X).

The above results should be read as follows. First note that for fixed β the smoothness measure given by the space $B_\tau^\beta(L_\tau)$ becomes weaker when τ decreases. Theorem 9.1 says that for fixed α the loss of regularity incurred by decreasing τ towards the critical value $2/(\alpha+1)$ can be compensated by *nonlinear approximation* so as to retain the order $N^{-\alpha/2}$ of approximation in H^1. Theorem 9.4 says that this order is best possible with respect to the full unit ball in the respective Besov class. To obtain the same approximation order through spaces on (quasi-) uniform meshes is equivalent to u belonging in the much smaller space $B_\infty^{\alpha+1}(L_2)$ (which is close to $H^{\alpha+1}$).

Thus whenever the solution u has sufficient regularity measured in L_2 the best possible balance of accuracy versus the number of degrees of freedom can be obtained at least asymptotically by using quasi-uniform meshes. Nonlinear approximation provides asymptotically better rates whenever u has a higher regularity in L_τ for $\tau < 2$. Theorem 8.1 combined with Theorem 9.1 then say that this better rate is actually recovered by the above adaptive algorithm which is a special instance of a nonlinear process. It has recently been shown in [10,9] that, depending on the smoothness of the domain Ω, the solution to Poisson's equation indeed has typically higher Besov than Sobolev regularity in the sense that

$$\begin{aligned}\alpha^* &:= \sup\{\alpha : u \in B_\tau^{\alpha+1}(L_\tau), \tau^{-1} = (\alpha+1)/2\} \\ &> \beta^* := \sup\{\beta : u \in B_\infty^{\beta+1}(L_2)\}.\end{aligned}$$

Hence in those cases the use of the adaptive scheme gives a better asymptotic work/accuracy rate.

Acknowledgements. We are very indebted to P. Morin and R. Nochetto for valuable comments and for pointing out to us an erroneous argument in an earlier version of this paper.

References

1. Bergh, J., Löfström, J.: Interpolation Spaces. An Introduction, Springer, 1976
2. Binev, P., DeVore, R.: Fast computation in tree approximation. Numer. Math., this issue
3. Binev, P., Dahmen, W., DeVore, R., Petrushev, P.: Approximation classes for adaptive methods. Serdica Math. J. **28**, 391–416 (2002)
4. Babuška, I., Vogelius, M.: Feedback and adaptive finite element solution of one dimensional boundary value problems. Numer. Math. **44**, 75–102 (1984)
5. Babuška, I., Miller, A.: A feedback finite element method with a posteriori error estimations: Part *I*. The finite element method and some basic properties of the a posteriori error estimator. Comput. Methods Appl. Mech. Engrg. **61**, 1–40 (1987)
6. Cohen, A., Dahmen, W., DeVore, R.: Adaptive wavelet methods for elliptic operator equations - convergence rates. Math. Comp **70**, 27–75 (2001)
7. Cohen, A., Dahmen, W., DeVore, R.: Adaptive wavelet methods II – beyond the elliptic case. Foundations of Comp. Math. **2**, 203–245 (2002)
8. DeVore, R., Sharpley, R.: Besov spaces on domains in $\mathbb{R}^d$. TAMS **335**, 843–864 (1993)
9. Dahlke, S.: Besov regularity for elliptic boundary value problems on polygonal domains. Appl. Math. Lett. **12**, 31–36 (1999)
10. Dahlke, S., DeVore, R.: Besov regularity for elliptic boundary value problems. Comm. Partial Differential Equations. **22**, 1–16 (1997)
11. Dahmen, W., Kunoth, A.: Multilevel preconditioning. Numer. Math. **63**, 315–344 (1992)
12. Dörfler, W.: A convergent adaptive algorithm for Poisson's equation. SIAM J. Numer. Anal. **33**, 1106–1124 (1996)
13. Griebel, M., Oswald, P.: Remarks on the abstract theory of additive and multiplicative Schwarz algorithms. Numer. Math. **70**, 163–180 (1995)
14. Karaivanov, B., Petrushev, P.: Nonlinear piecewise polynomial approximation beyond Besov spaces. Appl. Comput. Harmon. Anal. **15**, 177–223 (2003)
15. Lovász, L., Plummer, M.D.: Matching Theory. Akadémiai Kiadó, Budapest, 1986
16. Mitchell, W.F.: A comparison of adaptive refinement techniques for elliptic problems. ACM Transaction on Math. Software **15**, 326–347 (1989)
17. Morin, P., Nochetto, R., Siebert, K.: Data Oscillation and Convergence of Adaptive FEM. SIAM J. Numer. Anal. **38**, 466–488 (2000)
18. Oswald, P.: Multilevel Finite Element Approximations. Teubner Skripten zur Numerik, Teubner-Verlag, Stuttgart, 1994
19. Papadimitrious, C., Steiglitz, K.: Combinatorial Optimization: Algorithms and Complexity. Englewood Cliffs, New Jersey, Prentice Hall, 1982.
20. Petersen, J.: Die Theorie der regulären Graphen. Acta Math **15**, 193–220 (1891)
21. Triebel, H.: Interpolation Theory, Function Spaces, and Differential Operators, Amsterdam, North–Holland, 1978
22. Verfürth, R.: A Review of A Posteriori Error Estimation and Adaptive Mesh-Refinement Techniques. Chichester, Wiley–Teubner, 1996