Computing and Visualization in Science manuscript No.
(will be inserted by the editor)

On the Convergence of Adaptive Feedback Loops

Randolph E. Bank · **Harry Yserentant**

Received: November 21, 2018 / Accepted: date

Abstract We present a technique for proving convergence of h and hp adaptive finite element methods through comparison with certain reference refinement schemes based on interpolation error. We then construct a testing environment where properties of different adaptive approaches can be evaluated and improved.

Keywords Adaptive Feedback Loop, Finite Elements, Convergence Proof

Mathematics Subject Classification (2010) 65N30, 65N15, 65N50

1 Overview

Adaptive feedback loops used in the numerical solution of partial differential equations have two main components related specifically to the creation of a sequence of refined subspaces: an a posteriori error estimate, and a strategy for enriching the finite element space. While there has been extensive work on the development of a posteriori error estimates in various situations, relatively little attention has been paid to the systematic study of the adaptive strategies themselves. However, it has been shown that certain strategies lead to optimally convergent adaptive algorithms, some rigorously through mathematical proof [8,11,10,22,20,21,14,15] and others empirically through numerical experiments [18,16,19,17,3]. In this work, we study adaptive refinement procedures using interpolation error as the error indicator, comparing a reference refinement procedure to a target adaptive feedback loop that could be implemented in an application. In this way we can evaluate and compare different adaptive strategies in an environment that eliminates much of the noise (e.g., the approximate solution of the finite element system, and the effects of different a posteriori error estimates) that can influence such comparisons. The link to the adaptive solution of partial differential equations is our former result [7] on the comparability of the local interpolants and quasi-optimal approximants. Our analysis covers a variety of (inner product) norms in both 2D and 3D, a variety of finite element spaces, and we consider both h and hp adaptive refinement schemes. We begin with a shape regular triangulation $\mathcal{T}_0$ of our domain Ω consisting of N_0 elements. Let $\mathcal{S}_0$ denote our initial conforming finite element space corresponding to $\mathcal{T}_0$, with D_0 degrees of freedom. Let $\{\mathcal{S}\} \equiv \{\mathcal{S}(\mathcal{T}_0)\}$ denote the nested family of subspaces generated through sequential refinement of $\mathcal{S}_0$ using some known refinement procedure (e.g. red-green, longest edge bisection, some particular hp refinement strategy, etc). We note that the refined spaces in $\{\mathcal{S}\}$ need not be conforming; since interpolation is local, discontinuous finite element spaces are allowed.

The function u is to be approximated, and we denote its interpolant by u_I and the interpolation error by $e = u - u_I$. Let $\|\cdot\|$ be an appropriately chosen inner product norm. Let $\mathcal{S} \in \{\mathcal{S}(\mathcal{T}_0)\}$ denote the current subspace, $\mathcal{T}$ the current mesh, and $u_I \in \mathcal{S}$. The current error is

$$\|e\|^2 = \sum_{t\in\mathcal{T}} \|e\|_t^2.$$

Our central assumption is a local saturation assumption [9,12]. If some element t is refined, we assume that the local interpolation error is decreased. If u_I is the original in-

Bank: The work of this author was supported by the National Science Foundation under contract DMS-1318480, and the Alexander von Humboldt Foundation through a Humboldt Research Award.

Yserentant:

Bank: Department of Mathematics, University of California, San Diego, La Jolla, California 92093-0112. Email:rbank@ucsd.edu
Yserentant: Institut für Mathematik, Technische Universität Berlin, 10623 Berlin, Germany. Email:yserentant@math.tu-berlin.de

terpolant and $\bar{u}_I$ the refined interpolant, we assume for each element t, there exists some $0 \le \beta_t \le \beta < 1$ such that

$$\|u - \bar{u}_I\|_t \le \beta_t \|u - u_I\|_t \tag{1}$$

We note that adaptive hp algorithms allow the possibility of either h or p refinement, and thus the possibility of two values $\beta_{t,h}$ and $\beta_{t,p}$. Where necessary we will make this distinction.

Our reference adaptive procedure is defined inductively. Given $u_I \in \mathcal{S}$, with N elements, we choose a new subspace $\mathcal{S} \subset \mathcal{S}'$ as follows: we choose some element $\bar{t} \in \mathcal{T}$ such that

$$\|e\|_{\bar{t}} = \max_{t \in \mathcal{T}} \|e\|_t$$

and refine $\bar{t}$ using our chosen refinement scheme [1]. The new error $\|\bar{e}\|$ is given by

$$\|\bar{e}\|^2 = \|e\|^2 - \|e\|_{\bar{t}}^2 + \|\bar{e}\|_{\bar{t}}^2 \le \|e\|^2 + (\beta_{\bar{t}}^2 - 1)\|e\|_{\bar{t}}^2$$

Let e_0 denote the error for the initial subspace $\mathcal{S}_0$. In our adaptive process, we start from $\mathcal{S}_0$ and sequentially choose elements to refine as described above. Newly refined elements are immediately eligible for further refinement. While clearly assumption (1) is sufficient for the interpolation error to converge monotonically to zero, it does not allow us to characterize the rate of convergence as a function of the dimension of the approximating subspace. To do this we study the convergence of a particular subsequence. We choose some particular $\gamma < 1$. Through our refinement process, eventually we will arrive at a special error e_1 and corresponding subspace $\mathcal{S}_1$. This is characterized by some refinement step that satisfies

$$\|e_1\| \le \gamma \|e_0\|. \tag{2}$$

We let R_1 denoted to number of refinement steps necessary to compute e_1, N_1 the number of elements in $\mathcal{T}_1$, and D_1 the dimension of $\mathcal{S}_1$.

We continue the refinement process starting from e_1 until we achieve e_2 that satisfies

$$\|e_2\| \le \gamma \|e_1\| \le \gamma^2 \|e_0\|$$

This requires at most R_2 refinement steps with N_2 elements in $\mathcal{T}_2$ of dimension D_2.

Inductively, we generate a subsequence $\{e_k\}$ that satisfies

$$\|e_k\| \le \gamma^k \|e_0\| \tag{3}$$

with corresponding subsequences $\{\mathcal{S}_k\}$, $\{R_k\}$, $\{N_k\}$, and $\{D_k\}$. Our challenge is to make a selection for γ that allows us to bound the growth in R_k, N_k, and D_k, and from this to quantify the rate of convergence. To do this we need additional details concerning the particular adaptive algorithm to be studied. To see the value of these results in practice, we propose the following scenario for evaluating a given adaptive refinement scheme.

1. Determine the class of functions $\mathcal{C}$, based upon regularity and singularity types, to be addressed by the given adaptive method. We assume (1) holds for $u \in \mathcal{C}$.
2. Develop a reference adaptive scheme such that for $u \in \mathcal{C}$, the resulting sequence $\{\mathcal{S}\}$ is optimally convergent. The h and hp methods based on regular refinement analyzed in Section 3 are examples. See [3] for reference schemes based on relaxed longest edge refinement. Since reference algorithms adaptively refine known functions using interpolation error, one can make use of this extensive knowledge; e.g, one can compute $\beta_{t,h}$ and $\beta_{t,p}$ and use this information in the refinement process.
3. Implement refinement rules for a target (practical) adaptive refinement scheme using interpolation error as the a posteriori error estimate. For $u \in \mathcal{C}$ the resulting sequence is denoted $\{\hat{\mathcal{S}}\}$. This scheme must include all important features including simulation of an adaptive feedback loop if such a loop is present in the target application. Although it uses interpolation error, it should only employ information that is available in the actual target setting.

For representative functions $u \in \mathcal{C}$, the resulting refinement sequence $\{\hat{\mathcal{S}}\}$ and the reference sequence $\{\mathcal{S}\}$ can then be compared. If interpolation error is both an upper and lower bound for the exact error in the target setting, the behavior of $\{\hat{\mathcal{S}}\}$ in this very controlled setting using interpolation error indicates its behavior in the target setting for functions in the target class $\mathcal{C}$. The advantage of studying the behavior of $\{\hat{\mathcal{S}}\}$ in this restricted environment is that many of the external influences that can effect an adaptive scheme are controlled, allowing one to focus on the refinement strategy. In this way, strengths and weaknesses of different strategies are revealed, and one can modify the refinement rules to improve the performance of the method. While it is unlikely that practical schemes can match a reference scheme in terms of performance, this nonetheless provides a good environment for evaluating the effectiveness of a given approach. In Section 4 we employ this strategy to analyze some example adaptive feedback loops. In Section 5 we present some numerical experiments using a modified version of the *PLTMG* software package [2]. A much more extensive set of numerical experiments based on this procedure appears in Bank and Deotte [3].

[1] Finding the maximum will introduce a logarithm into the complexity (e.g., we keep all the elements in a heap based on their errors), but this is not important in this context.

2 Analysis

Suppose that we have an a posteriori error estimator

$$\eta = \left(\sum_t \eta_t^2 \right)^{1/2}$$

that is both efficient and reliable [20,21]. Thus given a function $u \in \mathcal{C}$ and $u_h \in \mathcal{S}$ that satisfies a (quasi) best approximation property,

$$\|u - u_h\| \leq C \inf_{v \in \mathcal{S}} \|u - v\|,$$

there exist positive constants c_1 and C_1, independent of N, such that

$$\|u - u_h\| \leq C_1 \eta$$
$$c_1 \eta_t \leq \|u - u_h\|_t .$$

Since

$$c_1^2 \eta^2 = c_1^2 \sum_t \eta_t^2 \leq \sum_t \|u - u_h\|_t^2 = \|u - u_h\|^2$$

it follows that the lower bound is also global. From [7], it was shown that interpolation error is both reliable and efficient in a wide variety of circumstances, and we assume that to be the case for the class $\mathcal{C}$. Thus we assume there exist constants positive c_0 and C_0 such that

$$\|u - u_h\| \leq C_0 \|u - u_I\|$$
$$c_0 \|u - u_I\|_t \leq \|u - u_h\|_t$$
$$c_0 \|u - u_I\| \leq \|u - u_h\|$$

From this it follows that the a posteriori error estimate η and interpolation error are comparable

$$\frac{c_0}{C_1} \|u - u_I\| \leq \eta \leq \frac{C_0}{c_1} \|u - u_I\| \tag{4}$$

and we have the following simple theorem.

Theorem 1 *Let $u \in \mathcal{C}$, $u_h \in \mathcal{S}$ a (quasi) best approximation, and η an posteriori error estimate, Assume both η and $\|u - u_I\|$ are reliable and efficient. Then the exact error $\|u - u_h\|$, the interpolation error $\|u - u_I\|$ and the a posteriori error estimate η all behave in the same fashion on any sequence of refinement spaces. In particular, if any one of the three converges at a given rate, the other two also converge at the same rate.*

Theorem 2 *Suppose the a posteriori error estimate η_t and local interpolation error satisfy*

$$\mu \|u - u_I\|_t \leq \eta_t \leq \|u - u_I\|_t / \mu \tag{5}$$

for some $0 < \mu \leq 1$. We also assume for adaptive hp-refinement schemes that both make the same choice between h and p refinement for the same given element. Let $\{\mathcal{S}\}$ denote the sequence of subspaces generated by our reference refinement procedure using interpolation error, and let $\{\hat{\mathcal{S}}\}$ denote the sequence of subspaces generated by our reference refinement procedure using the a posteriori error estimate η. Then the sequence $\{\hat{\mathcal{S}}\}$ converges at the same rate as $\{\mathcal{S}\}$.

Proof We begin by constructing a sequence of elements t that are chosen for refinement. This list contains both the original elements and all their descendants ordered according to the selection order in creating the sequence of subspaces $\{\mathcal{S}\}$. Let t_{max} be the first element chosen for refinement, an element with largest interpolation error at any stage of the refinement process. Next we construct a sequence of bins $\{\mathcal{B}_k\}$ as follows: elements in $t \in \mathcal{B}_k$ satisfy

$$\mu^{k+1} \|u - u_I\|_{t_{max}} < \|u - u_I\|_t \leq \mu^k \|u - u_I\|_{t_{max}},$$

for $k = 0, 1, 2 \ldots$. We note the refinement process producing the sequence $\{\mathcal{S}\}$ must refine in order all the elements in $\mathcal{B}_k$ before proceeding to bin $\mathcal{B}_{k+1}$. We replace $\|u - u_I\|_t$ by η_t, and construct $\{\hat{\mathcal{S}}\}$ using the same refinement procedure. Associated with $\{\hat{\mathcal{S}}\}$, we construct a similar set of bins $\{\hat{\mathcal{B}}_k\}$ corresponding to the refinement order used in creating the $\{\hat{\mathcal{S}}\}$ using the a posteriori error indicators η_t. From (5), a given element $t \in \mathcal{B}_k$ must appear in $\hat{\mathcal{B}}_j$ for $k-1 \leq j \leq k+1$. Conversely, a given element $t \in \hat{\mathcal{B}}_k$ must appear in $\mathcal{B}_j$ for $k-1 \leq j \leq k+1$.

Let $\{\mathcal{S}_k\}$ denote a subsequence of the $\{\mathcal{S}\}$, with $\mathcal{S}_k$ denoting the space after the last element in bin $\mathcal{B}_{k-1}$ has been refined and before the first element in $\mathcal{B}_k$ is refined. $\mathcal{S}_0$ is the initial subspace before the first refinement. Let D_k denote the dimension of the subspace $\mathcal{S}_k$ for $k = 0, 1, \ldots$. Let $\{\hat{\mathcal{S}}_k\}$ denote the subsequence of subspaces corresponding to the refinement process using a posteriori error indicators, and $\{\hat{D}_k\}$ their corresponding dimensions. We note that $D_0 = \hat{D}_0$ and

$$\hat{D}_{k-1} \leq D_k \leq \hat{D}_{k+1}$$
$$D_{k-1} \leq \hat{D}_k \leq D_{k+1}$$

for $k \geq 1$. It follows that

$$\max(\hat{D}_{k-1}, D_{k-1}) \leq D_k \leq \min(\hat{D}_{k+1}, D_{k+1})$$
$$\max(\hat{D}_{k-1}, D_{k-1}) \leq \hat{D}_k \leq \min(\hat{D}_{k+1}, D_{k+1}).$$

Thus although individual bins might be unusually large, unusually small, or even empty, size differences between D_k and $\hat{D}_k$ are offset by compensating differences in nearby bins, so the two sequences of subspaces grow in dimension in roughly the same way.

We now consider the space $\hat{\mathcal{S}}_{k+1}$. This space contains all the refinement through bin $\hat{\mathcal{B}}_k$, and thus all the refinements in $\mathcal{B}_j$ for $0 \leq j \leq k-1$, and hence is some further refinement

of $\mathcal{S}_k$. Similarly $\mathcal{S}_{k+1}$ contains all the refinements in $\hat{\mathcal{B}}_j$ for $0 \le j \le k-1$, and hence is some further refinement of $\hat{\mathcal{S}}_k$. Thus we see that as $k \to \infty$ the two sequences must behave in a similar fashion and in particular exhibit the same rate of convergence. □

Assumption (5) in Theorem 2 is the local analog of estimate (4). Since both η and $\|u-u_I\|$ are assumed to be both reliable and efficient, the local lower bound assumption is quite reasonable, and for example holds even if η is the exact error $\|u-u_h\|$. On the other hand, the local upper bound estimate in (5) will not generally hold for all choices of η and in particular with $\eta = \|u-u_h\|$ due to possible pollution effects. However, we show below that (5) does hold for the choice of η used in Theorem 3 and our subsequent analysis of some adaptive refinement strategies in Section 4.

As $\mu \to 0$, Theorem 2 breaks down. For very small but fixed μ the bins $\mathcal{B}_k$ and $\hat{\mathcal{B}}_k$ will both tend to become very large, with the potential effect that many elements in $\{\hat{\mathcal{S}}\}$ will be refined prematurely in comparison with the reference method. The sequence $\{\hat{\mathcal{S}}\}$ still converges at the same asymptotic rate as $\{\mathcal{S}\}$, although this behavior may not become apparent until the error is very small, and the spaces have very large dimension, perhaps so large that it cannot be observed in practical calculations. For example, for sufficiently small errors in a $\log-\log$ plot of error as a function of degrees of freedom, the curves for $\{\mathcal{S}\}$ and $\{\hat{\mathcal{S}}\}$ will asymptotically have the same slope, but that for $\{\hat{\mathcal{S}}\}$ will be displaced to the right of the one for $\{\mathcal{S}\}$ by a large number of degrees of freedom.

Finally, we note that our proof assumes all error indicators are updated at every refinement step. In practice this is not the case. Indeed, the typical scenario involves an adaptive feedback loop [1,20,21,23] of the form

solve $\to$ estimate $\to$ refine.

In the solve phase the approximate solution u_h is computed on a given mesh. The a posteriori error estimate for this subspace is computed in the estimate phase. Then this estimate is employed in the refine phase. Typical sets of refinement rules allow for several elements to be refined during the refine phase. Sometimes the refine phase itself is divided into two parts

solve $\to$ estimate $\to$ mark $\to$ refine.

In the mark phase, some subset of the elements in the existing mesh are marked for refinement, and then that set is refined during the refine phase. In either scenario, the process is not as tightly controlled as in the case of the reference procedure. The solve and estimate phases are usually relatively expensive, and one does not normally want to invoke them on a sequence of subspaces growing by one refinement in each feedback loop, or even a small number of refinements. On the other hand, refining many elements before updating the a posteriori error estimate can potentially degrade the quality of the refined meshes in terms of the convergence of $\|u-u_h\|$. Thus one must seek a balance between the cost of the components of the adaptive feedback loop and the quality of the subspaces that it produces.

We note that by Theorem 1, if the sequence of subspaces $\{\hat{\mathcal{S}}\}$ generated by an adaptive feedback loop is optimally convergent for any of $\|u-u_h\|$, $\|u-u_I\|$, or η, it is optimally convergent for all. To analyze a particular adaptive feedback loop strategy using Theorem 2, We develop the sequence $\{\hat{\mathcal{S}}\}$ as follows. We implement the adaptive feedback loop using interpolation error as the error indicator. In this setting, much of the cost of the solve and estimate phases is avoided, and the use of interpolation error allows one to focus on the particular details of the refine (or mark/refine) phases. However, this feedback loop should follow exactly the refine (or mark/refine) phases of the target application, and in particular only use information that is available to these phases in the target application. The ordering of refined elements in $\{\hat{\mathcal{S}}\}$ should correspond to the ordering generated by the adaptive feedback loop.

Theorem 3 *Let $\{\hat{\mathcal{S}}\}$ be a sequence of spaces generated by an adaptive feedback loop using interpolation error as the error indicator, with global error converging to zero. Then there exists a related error indicator η such that the same sequence $\{\hat{\mathcal{S}}\}$ is generated by the reference refinement strategy using η as the error indicator. In the case of hp adaptive refinement, the choice between h and p for a given element must be the same.*

Proof To apply Theorem 2, the sequence $\{\hat{\mathcal{S}}\}$ must be viewed as a sequence of spaces resulting from the refinement of a single element in the current space. However, each refine (or mark/refine) phase of an adaptive feedback loop might be responsible for generating many consecutive members of the sequence, depending on how many elements are refined during that particular loop. We now construct a new (but related) error indicator η that generates the sequence $\{\hat{\mathcal{S}}\}$ by refining a single element in each step. At the beginning of each refine (or mark/refine) phase, we find an element t_{max} having the largest error in the current subspace, and set $\eta_{t_{max}} = \|u-u_I\|_{t_{max}}$. For many adaptive feedback loops, t_{max} is the first element selected for refinement (or marked for refinement). Using $\eta_{t_{max}}$, we define pseudo error indicators η as follows; all elements t chosen for refinement during this particular refine (or mark/refine) phase by the underlying adaptive feedback loop are given the pseudo error indicator $\eta_t = \eta_{t_{max}}$. A similar assignment process is repeated for each adaptive feedback loop, resulting in a sequence $\{\eta_t\}$ corresponding to $\{\hat{\mathcal{S}}\}$ that is monotonic non increasing and constant on the set of elements selected for refinement during each refine (mark/refine) phase of the original algorithm. Since $\{\hat{\mathcal{S}}\}$ is globally convergent to zero,

any element that is never refined at any stage must have $\|u-u_I\|_t = 0$, and we assign $\eta_t = 0$ to such elements. Thus every element at every step of the process has an error indicator η_t. The reference algorithm always selects an element with the maximum error indicator for refinement at every step. Thus applying the reference algorithm using the error indicators η will generate the sequence $\{\hat{\mathcal{S}}\}$, provided we break ties in the obvious fashion. □

In order to use Theorems 2 and 3 together in our analysis of adaptive feedback loops, we must estimate the value of μ in (5) of Theorem 2 for the particular error indicator η developed in Theorem 3. Since by definition $\|u-u_I\|_t \leq \eta_t$, the lower bound in (5) is satisfied for any $0 < \mu \leq 1$. To satisfy the upper bound, we choose μ as

$$\mu = \min_t \frac{\|u-u_I\|_t}{\eta_t}.$$

(If $\|u-u_I\|_t = \eta_t = 0$, we define their ratio to be 1.) In order to make μ large, a reasonable (and perhaps obvious) guideline for developing an effective strategy for an adaptive feedback loop is to avoid refining elements with errors substantially smaller than the largest element error $\eta_{t_{max}}$ in the space at the beginning of that refine (or mark/refine) phase.

3 Example Reference Adaptive Procedures

In this section, we develop reference h and hp adaptive schemes that exhibit optimal rates of convergence, using interpolation error as the a posteriori error estimate. To define the class $\mathcal{C}$ of functions to be addressed, we consider functions $u \in \mathcal{H}^1(\Omega)$ that are smooth except for a fixed number r of isolated point singular points x_k, $1 \leq k \leq r$. where locally $u \in \mathcal{H}^{1+\alpha_k}(t)$. We assume that each singular point x_k is a vertex in the initial mesh $\mathcal{T}_0$, and that (1) is satisfied. The norm used is the $\mathcal{H}^1$ semi-norm $\|u\| = |u|_{\mathcal{H}^1} = \|\nabla u\|_{\mathcal{L}_2}$. For simplicity in these examples, we consider regular refinement of 2D triangular meshes, using the usual families of Lagrange finite elements. Regular (red) h-refinement consists of refining a given element t into four similar elements by pairwise connecting its edge midpoints. Regular p-refinement consists of increasing the polynomial degree of the element by one.

3.1 h-refinement

We first consider h refinement using elements of fixed degree p. Let E denote the set of elements with a vertex that is also one of the r singular points. Note E is typically not a static but evolves with the refinement process. Because the mesh is shape regular, the number of elements in E at any given time is bounded by some fixed constant M.

We assume for elements $t \notin E$, that $u \in \mathcal{H}^{p+1}(t)$, and that each refinement of such elements results in an optimal error reduction with $\beta_{t,h} \approx 2^{-p}$. For simplicity, we initially assume $\beta_{t,h} = 2^{-p}$ and consider the case $\beta_{t,h} \approx 2^{-p}$ later, treating it as a perturbation.

For elements $t \in E$, $1 > \beta_{t,h} > 2^{-p}$, depending on the nature of the singularity. However, error in these elements can be reduced by 2^{-p} by a fixed number of h-refinements, depending on the the local smoothness of u and p.

We now estimate the number of refinement steps required to create a subsequence $\{e_k\}$ with $\gamma = 2^{-p}$ satisfying

$$\|e_k\| \leq 2^{-p}\|e_{k-1}\|.$$

Initially, assume all elements $t \notin E$ are refined only once. Those $t \in E$ for the starting mesh used for e_{k-1} need to be refined at most a fixed number of times to reduce their errors by 2^{-p}. Then

$$N_k \leq 4N_{k-1} + 3\chi(M,r) \tag{6}$$

for some constant $\chi = \chi(M,r)$. Now suppose some elements $t \notin E$, are refined more than once; then in compensation other elements in E (having smaller errors than those chosen for refinement) are not refined at all in this phase, so (6) still holds. The solution of the majorizing difference is

$$N_k = (N_0+\chi)4^k - \chi$$

so that

$$k \approx \frac{\log((N_k+\chi)/(N_0+\chi))}{\log 4}.$$

This leads to the corresponding estimate

$$\begin{aligned}\|e_k\| &\leq \gamma^k\|e_0\| \\ &\lessapprox \left(\frac{N_k+\chi}{N_0+\chi}\right)^{\log\gamma/\log 4}\|e_0\| \\ &\lessapprox \left(\frac{N_k+\chi}{N_0+\chi}\right)^{-p/2}\|e_0\|\end{aligned} \tag{7}$$

Estimate (7) is asymptotically optimal. However, it suggests that for large p and small N_k, optimal convergence rates will not be observed during the initial stages of the adaptive refinement process. Indeed, we have empirically observed this in numerical experiments [3]. For large p and small N in the initial stages there might be insufficient elements available to grade the mesh in an optimal fashion using regular refinement or other bisection schemes that control shape regularity. In the special case when u is smooth and $r = 0$, then $\chi = 0$ and (7) reduces to

$$\|e_k\| \lessapprox \left(\frac{N_k}{N_0}\right)^{-p/2}\|e_0\|.$$

As mentioned above, our assumption $\beta_{t,h} = 2^{-p}$ exactly for h-refinement is only asymptotically correct. This could perturb our h-adaptive refinement scheme, especially at the beginning when the elements are relatively large. Thus a more reasonable bound for our h-refinement reference scheme is

$$N_k \le 4N_{k-1}(1+\varepsilon^k)+3\chi$$

for some $0 \le \varepsilon < 1$. The introduction of the term $(1+\varepsilon^k)$ makes the solution of the majorizing difference equation more complicated. The homogeneous solution is

$$N_k = 4^k N_0 \prod_{j=1}^{k}(1+\varepsilon^j) \le \rho 4^k N_0$$

for some fixed constant $\rho = \rho(\varepsilon) = 1 + O(\varepsilon)$. By a similar calculation the particular solution is bounded by

$$\rho(4^k - 1)\chi,$$

so the overall rate of convergence is bounded by

$$\|e_k\| \lessapprox \left(\frac{N_k + \rho\chi}{\rho N_0 + \rho\chi}\right)^{-p/2} \|e_0\|. \tag{8}$$

3.2 hp-refinement

We next consider hp-adaptive refinement. As in the case of our h refinement algorithm, we select an element of maximum error for each refinement step. To decide between h and p refinement, we consider the value of $\beta_{t,h}$. If element t has local degree p and $\beta_{t,h} \approx 2^{-p}$, then u is sufficiently smooth locally, and we p-refine t; otherwise we employ regular h-refinement.

Refinement near point singularities is more complicated in this setting. There may be singularities where $u \in \mathcal{H}^k(t)$ but $u \notin \mathcal{H}^{k+1}(t)$ for some $k \ge 1$. Then our hp-adaptive strategy will use p-refinement up to local polynomials of degree $k-1$ and then switch to h-refinement. We have actually observed this in practice. However, our worst case analysis below is not delicate enough to take this effect into account, resulting in overestimating the dimension growth, although our bound on the convergence rate is still of optimal order. Here we chose γ such that $\beta_{t,p} \le \gamma$ for $t \notin E$, select a subsequence $\{e_k\}$ that satisfies

$$\|e_k\| \le \gamma \|e_{k-1}\|,$$

and bound the number of degrees of freedom D_k necessary to achieve this estimate. We assume the initial space $\mathcal{S}_0$ has only piecewise linear polynomials.

We assume that elements are p-refined at most ℓ times in moving from e_{k-1} to e_k. Initially, if the error in the elements $t \notin E$ is distributed unevenly, then some elements might be p-refined several times. However, due to our choice of γ, once the error becomes more evenly distributed typically such elements will be p-refined at most once in moving from e_{k-1} to e_k. The recurrence relation for the growth of degrees of freedom is now more complicated because h and p refinements influence to number of elements and the number of degrees of freedom in different ways. Thus we need recurrence relations for both N_k and D_k. We note that the number of elements N_k changes only in response to h-refinement, and thus is of the form

$$N_k \le N_{k-1} + \chi(M, r)$$

from which it follows that

$$N_k \le N_0 + k\chi.$$

To bound D_k, if an element of degree j is p-refined, at most $j+2$ degrees of freedom are added to that element: $j-1$ interior degrees of freedom and one degree of freedom for each edge. In moving from e_{k-1} to e_k, in the worst case, all N_k elements are p-refined ℓ times to degree $\ell k + 1$, leading to the bounding recurrence relation

$$D_k \le D_{k-1} + N_k(\ell k + 2)\ell \le D_{k-1} + (N_0 + k\chi)(\ell k + 2)\ell$$

The solution of the majorizing difference equation is

$$D_k = \frac{\chi\ell^2 k^3}{3} + \frac{\{(N_0+\chi)\ell + 2\chi\}\ell k^2}{2} + \frac{\{(3N_0+\chi)(\ell+4)+2\chi\}\ell k}{6} + D_0$$

from which it follows that $k = O(D_k^{1/3})$. This leads to an estimate of the form

$$\|e_k\| \lessapprox a e^{-bD_k^{1/3}} \|e_0\|.$$

In the special case $r = 0$, $\chi = 0$ and $k = O(D_k^{1/2})$, and

$$\|e_k\| \lessapprox a e^{-bD_k^{1/2}} \|e_0\|.$$

These estimates are in agreement with the analysis of Babuška and Guo [13].

3.3 Conforming Finite Element Spaces

We now address the problem of making the spaces corresponding to the e_k conforming. Many of the usual h-refinement schemes provide refinement rules that insure the mesh remains conforming as well as controlling the shape regularity of refined elements. In [5], rules for regular (red-green) h refinement are given. For this scheme as well as others, generally few additional refinements are needed to insure a conforming triangulation; typically neighbor elements have errors of similar magnitude as those chosen for refinement, and thus often would be chosen for refinement

on the basis of their error if not chosen for refinement to make the mesh conforming. In any event, by (1) such refinements also contribute to achieving the overall error reduction of γ.

For our example h-refinement scheme based on regular refinement, a simple construction can be used to create a conforming mesh without creating any additional degrees of freedom. Here we focus on so-called transition elements, i.e., those with hanging nodes along one or more of their edges. For such an element, we begin by (virtually) refining it using regular refinement until all of the hanging nodes become vertices of this local virtual triangulation. For each hanging node (or hanging degree of freedom for $p > 1$) we add its corresponding nodal basis function to the set of nodal basis functions of the transition element. From this enlarged set of basis functions, it is easy to construct a local nodal basis for each transition element that yields a global conforming space. Since the additional basis functions correspond to hanging degrees of freedom on edges, there is no net increase in the global number of degrees of freedom. Furthermore, since the local subspace for a transition element contains all polynomials of degree p plus certain piecewise polynomials on its edges, its basic approximation properties are not impaired. We note that this construction can be applied to any individual space $\mathcal{S}$ with nonconforming interpolant u_I, producing a conforming interpolant $\tilde{u}_I$. However, we return to u_I and $\mathcal{S}$ before continuing with the next refinement step.

hp refinement schemes also have rules for making the finite element spaces conforming with respect to p. In [4] transition elements of lower degree have degrees of freedom on its edges that correspond to its higher degree neighbors. The construction is somewhat analogous to that described above. One begins with the original space of polynomials of degree p and then adds certain special higher degree polynomials along transition edges. From this enlarged basis set, one constructs a nodal basis that is globally conforming. As before, since the new local basis functions correspond to existing degrees of freedom along transition edges, there is no net increase in the global number of degrees of freedom. As in the case with h-refinement, this construction may be applied at any step to produce a conforming interpolant $\tilde{u}_I$, but we return to the nonconforming space $\mathcal{S}$ to continue the adaptive process.

4 Example Feedback Loops

We consider the class of functions $\mathcal{C}$ defined above and study the behavior of several popular adaptive feedback loop strategies using Theorems 2-3. For simplicity, we assume just one singularity with $u \in \mathcal{H}^{1+\alpha}(t)$ with $0 < \alpha < 1$ for those elements t having the singular point as one of its vertices. Local interpolation error is used for the a posteriori error estimate.

Perhaps the most classical strategy for h-adaptivity is to mark for refinement all elements whose error indicator exceeds some threshold, for example

$$\eta_t \geq \theta \eta_{t_{max}}$$

where $0 < \theta < 1$ and $\eta_{t_{max}}$ is the largest error indicator in the current mesh. This strategy will certainly produce an optimal rate of convergence if θ is sufficiently close to one, as θ can be used to control the size of μ in (5) of Theorem 2.

A potential difficulty with this scheme arises from the wide range of values for $\beta_{t,h}$, that varies from (approximately) 2^{-p} in regions where the solution is smooth to (approximately) $2^{-\alpha}$ at the singular point. To reduce the error by 2^{-p} in smooth regions typically requires just one regular refinement, while it requires (approximately) p/α regular refinements at the singular point.

Let us consider the behavior of the method in terms of reducing the global error by 2^{-p}. Initially, suppose the current mesh is roughly equilibrated, that is all elements have approximately the same error. Then all or most elements will be selected for refinement in the first feedback loop using even a very large value for θ. The error in elements in the smooth region would decrease by about 2^{-p} and those at the singularity by only $2^{-\alpha}$. This suggests that for the second feedback loop only a few elements near the singularity will be marked for refinement, and the dimension of the finite element subspace will grow by only a small amount.

In the general setting, to reduce the error globally by 2^{-p}, one will likely require p/α or more feedback loops. Within those p/α loops, very smooth elements typically will be refined at most once, while those near the singularity will be refined multiple times. Since the set of elements near the singularity is typically a very small fraction of the total number of elements, this implies that many of the p/α feedback loops will produce only a small increase in subspace dimension. This in turn creates extra costs for the solve and estimate phases of the feedback loop.

This slow growth cannot be effectively addressed by choosing a smaller value for θ. Such a strategy will likely increase the rate of growth in subspace dimension, but this will reduce the size of μ and many of the marked elements would be unnecessarily refined. In the extreme case $\theta \approx 0$, the result would be (nearly) uniform refinement, and its corresponding convergence factor of $2^{-\alpha}$.

Another common marking scheme is Dörfler marking [8]. In this scheme a set $\mathcal{M}$ of marked elements is selected such that

$$\sum_{t \in \mathcal{M}} \eta_t^2 \geq \theta^2 \eta^2,$$

for some $0 < \theta < 1$. Typically the elements chosen for $\mathcal{M}$ have the largest (or have among the largest) values of η_t.

Like the previous scheme the marking parameter θ can be used to control the size of μ in (5), resulting in optimal convergence for θ sufficiently small. There are several proofs of optimality for this method [8,11,10,22,20,21,14,15] under a variety of conditions.

As with the previous marking strategy, a main issue is the wide variation in $\beta_{t,h}$. As before in the general setting, to reduce the error by 2^{-p}, it will take p/α refinement steps near the singular point and typically one refinement in the smooth regions; as before we will likely need at least p/α adaptive feedback loops. Elements at the singularity will likely be refined in every adaptive feedback loop, while those in smooth regions will be refined at most a few times during this sequence of loops. Once again one can expect overall slow growth in subspace dimension, especially for large values of p/α, and its corresponding adverse impact on the solve and estimate phases. We have observed this slow growth in practice in numerical experiments [3]. Similar to the previous case, increasing the size of $\mathcal{M}$ by increasing θ has the potential drawback of refining many unnecessary elements and allowing the rate of convergence to become suboptimal. (Choosing $\theta \approx 1$ will cause almost uniform refinement on every feedback loop.)

One possible way to address this slow-growth issue is to allow multiple refinements of a given element during each feedback loop. This is the basis of the strategy implemented in the *PLTMG* software package [2]. The error indicators η_t are based on an interpolation error formula of the form

$$u - u_I = \sum_j \mathcal{F}_j(\partial^{p+1}u)\psi_j \tag{9}$$

for the interpolation error on a triangle t, where the ψ_j form a basis for the space of polynomials of degree $p+1$ that are zero at all nodes of $\mathcal{S}$ on t and the coefficient functions $\mathcal{F}_j$ depend in an explicitly known and computationally accessible way on potentially all derivatives of u of order $p+1$, generically denoted $\partial^{p+1}u$. Piecewise constant approximations of these derivatives can be extracted from several a posteriori error estimates and superconvergent derivative recovery schemes [6]. In this case, our superconvergent derivative recovery scheme is applied to the interpolant u_I as a proxy for the finite element solution. One then computes η_t by computing the relevant norm of this approximation.

One immediate problem with this approach is that (9) holds only in smooth regions. Since extracting the derivative approximations is an algebraic process, it produces approximate derivatives whether or not they actually exist. In *PLTMG*, we compare an a posteriori error estimate based on interpolation error and the original a posteriori error estimate from which the derivatives were extracted. We then scale the extracted piecewise constant derivatives such that these two estimates are equal. In regions where u is smooth these scaling factors are approximately one, but they can become very large near singular points. This latter point plays an important role in our hp-adaptive procedure. Estimate (5) holds in smooth regions, and using the rescaled derivatives, also should hold near the singularity, provided both the original a posteriori estimate and interpolation error are both reliable and efficient.

The h-refinement scheme is similar to the reference scheme in that the element with largest error estimate is chosen for refinement. It is removed from the heap and its child elements are added to the heap, making them available for further refinement. The error estimates for the child elements are computed using the piecewise constant derivative approximations from their parent; all other information needed for (9) and η_t is taken from the element itself. The refine phase terminates when either of two conditions is satisfied. The first condition allows for rapid growth in subspace dimension; in particular, the current refine phase is terminated when the subspace dimension N_k satisfies

$$N_k \approx \min(4N_{k-1}, N_{trgt}) \tag{10}$$

where N_{k-1} is the subspace dimension at the beginning of the current refine phase, and N_{trgt} is a target value for the largest allowable subspace dimension provided by the user. The factor 4 references subspace dimension growth for 2D triangulations under uniform regular refinement. In this abstract setting $N_{trgt} = \infty$ and (10) simplifies. The second condition terminates the refine phase if the current largest error estimate η_t satisfies

$$\eta_t \le \theta\,\eta_{ave} \tag{11}$$

for $0 < \theta < 1$, where η_{ave} is the average value of η_t at the beginning of the current refine phase.

This approach is only partially successful at resolving the dimension growth issue for adaptive feedback loops. This stems from the fact that children inherit derivative information from their parent. While the error related to the adaptive process decreases according to (9), the error in the approximate derivative calculation remains static, at the level of the original ancestor at the beginning of the current refine phase. Eventually this becomes the dominant error in the calculated η_t and results in violation of (5). In practice, this scheme tends to make a uniform refinement of the original ancestor element, since all descendant elements use the same constant values for their derivatives in (9). When p/α is not too large, this is is good enough and allows large increases in subspace dimension while maintaining optimal convergence behavior. However, for large values of p/α the uniform refinement of the original ancestor elements p/α times would introduce many unnecessary elements, and cause degradation of the rate of convergence; hence the need for the second exit criteria, resulting in slower growth in subspace dimension with increasing p. To understand the impact of (11), we note that η_t decreases

more rapidly with increasing p in smooth regions (as 2^{-p}). The η_t also (artificially) decrease faster at the singularities as well, because the recovered derivatives, although large and inaccurate, are still fixed and finite and thus behave as in the smooth case in terms of their impact on the computed η_t. Overall the increasingly rapid decrease in the η_t allows (11) to force the refine phase to exit sooner as p increases.

In summary, this scheme has proved to be very effective for h-refinement for the case $p = 1$, allowing for relatively rapid dimension increase while maintaining optimal convergence. It becomes progressively less effective with increasing p, forcing slower growth in subspace dimension in order to retain optimal convergence behavior.

Finally, we consider the hp-adaptive feedback loop implemented in *PLTMG*. Our hp adaptive procedure is just a small extension of the h-adaptive procedure in *PLTMG*. When an element is p-refined, it is removed form the heap and is ineligible for further refinement in the current refine phase. The practical reason for this is that inheriting derivatives of order $p+1$ from a parent is not useful for p-refinement because derivatives of order $p+2$ are needed for (9) when an element of degree p is refined, and we currently have no inexpensive and accurate procedure for providing this information. On a positive note, based on our discussion of the reference hp procedure, we expect that one p-refinement per feedback loop should usually be adequate for elements in smooth regions. We note that elements can still be h-refined multiple times as in the h-adaptive algorithm.

The decision to h refine or p refine the element with largest error indicator involves two criteria. First, the constant used to scale the extracted derivatives in t must be smaller than $\hat{\theta} = 2$ as a necessary condition for p refinement. Second, we require

$$\frac{(\sum \eta_t^2)^{1/2}}{\|\nabla u_h\|} \leq \left(\frac{1}{3}\right)^p$$

where the terms on the left hand side are evaluated at the beginning of the current refine phase. This reflects the empirical observation that our derivative recovery is often not accurate enough on very coarse meshes, and therefore the scaling factors should not be trusted. If both criteria are satisfied, we choose p-refinement; otherwise we choose h-refinement. The two exit criteria used in the h-refinement algorithm are also used here. It is unlikely that our process for selecting h or p refinement always make the same choice as our reference procedure, although both criteria are motivated by the same considerations. Thus the assumption of Theorem 2 with respect to hp refinement is not likely to be satisfied. Nonetheless, empirical comparison with the reference hp procedure has proved to be quite valuable in improving our hp refinement procedure. See [3] for some details. Another experimental approach to study hp adaptive methods is given by Mitchell [18, 16, 19, 17].

5 A Numerical Example

In this section we provide an illustration of our theoretical results in a computational setting. These experiments were done using a modified version of the software package *PLTMG* [2]. A more comprehensive set of experiments, encompassing several classes of functions is given in [3]. We consider the case of a solution $u \in \mathcal{H}^{1+\alpha}$ having a single point singularity at the origin. We consider the case of the unit circle Ω with a crack along the positive x axis (see Figure 1, left). We choose as an example function $u = r^{1/4}\sin(\theta/4) \in \mathcal{H}^{5/4-\varepsilon}$. This function solves the elliptic partial differential equation

$$\begin{aligned} -\Delta u &= 0 && \text{in } \Omega \\ u &= r^{1/4}\sin(\theta/4) && \text{on } \partial\Omega_1 \\ u_n &= 0 && \text{on } \partial\Omega_2. \end{aligned}$$

The boundary $\partial\Omega_2$ is the bottom edge of the crack, and $\partial\Omega_1 = \partial\Omega \setminus \partial\Omega_2$. The initial mesh of eight elements and the solution are shown in Figure 1.

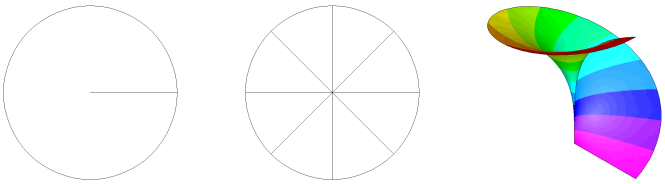


Fig. 1 The domain Ω (left), the initial mesh (center), and the solution $u = r^{1/4}\sin(\theta/4)$ (right).

In these experiments the target number of degrees of freedom was $N = 250K$. *PLTMG* is based on the usual family of triangular Lagrange elements (isoparametric versions as needed). The basic h-refinement procedures is a relaxed version of longest edge bisection. The basic p-refinement procedure is to increase the polynomial degree by one. Since *PLTMG* is typically employed to solve PDEs on conforming meshes, the refinement procedures include strategies that control the shape regularity of the elements and that insure the subspaces remain conforming in both h and p. Additional details are given in [2].

Our reference refinement procedure is much the same as that described in Section 4. However, there are a few concessions made due to numerical considerations.

i. The maximum polynomial degree allowed in *PLTMG* is nine. This is due to limits on the family of quadrature rules employed by the package.
ii. Proposed h-refinements are evaluated in terms of round-off error, and extremely small elements are disallowed. Elements having vertex locations that agree to almost the

machine precision become problematic, e.g. the usual affine mapping to the reference element becomes degenerate due to catastrophic cancellation.

iii. We incorporated a pseudo feedback loop, where the reference procedure reported data for $N_k = \min(4N_{k-1}, 250000)$. This provides data needed for graphics routines, as well as a benchmark for the comparison refinement procedures.

For our choice $N = 250K$, items (i) and (ii) did not have significant impact on the presented results.

For the reference procedure, we made four experiments: h-refinement for fixed $p = 1,2,4$ and hp-refinement. The resulting convergence curves are shown in Figure 2, where we display $\log(\|\nabla(u-u_I)\|/\|\nabla u\|)$ vs $\log N$. As reported below, all four experiments result in optimal asymptotic convergence rates. Of particular interest is the "flat" portion of the convergence history for h-refinement for $p = 4$ and to a lesser extent $p = 2$. This behavior is predicted by our analysis in Section 3, Equation (7). This suggests that for low accuracy requirements, low order h-refinement might be preferred to higher order h-refinement methods.

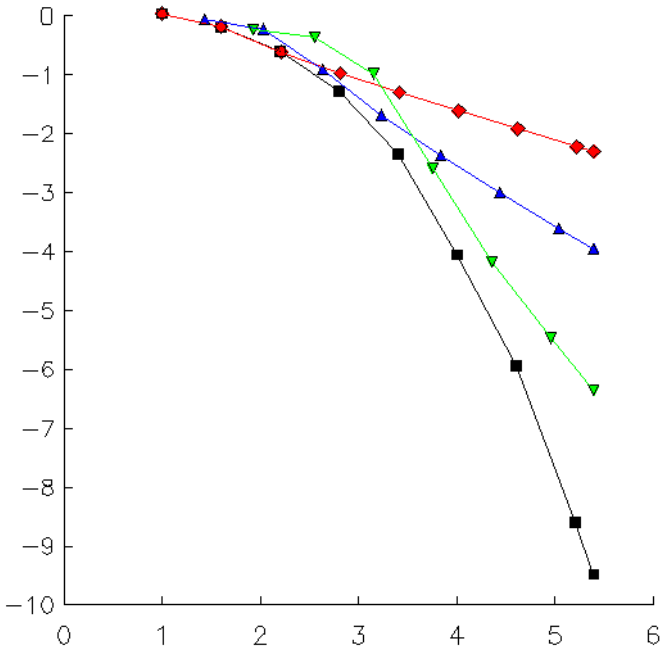


Fig. 2 reference procedure. The black curve is for hp refinement and the red, blue and green curves are for h refinement for $p = 1,2,4$ respectively. The x-axis is $\log N$ and the y-axis is $\log(\|\nabla(u-u_I)\|/\|\nabla u\|)$.

For the comparison adaptive feedback loops, we studied the *PLTMG* scheme for both h and hp-refinement, and the Dörfler marking scheme with $\theta = .5$ for h-refinement. While the reference scheme used interpolation errors exclusively, both the *PLTMG* and marking schemes use the interpolant u_I to compute recovered derivatives as in (9), and these error estimates were used in their respective adaptive feedback loops. In terms of refinement, in addition to the rules discussed in Section 4 both schemes were subject to the practical constraints for p and h refinement mentioned above for the reference scheme. In all cases the refined meshes were conforming in both h and p. The results of the four experiments are given in Figure 3.

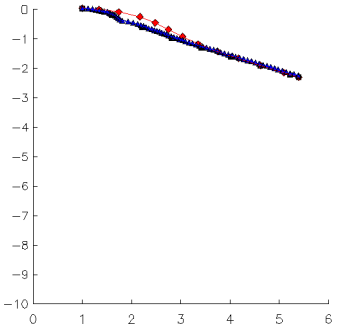
In the tables, several numbers are reported. Digits $= -\log(\|\nabla(u-u_I)\|/\|\nabla u\|)$, and Loops is the number of adaptive feedback loops used to reach the target of 250K degrees of freedom. For the h-refinement experiments, Order is a least squares estimate of the the coefficient b in the convergence rate estimate $aN^{-b/2}$. For the hp-refinement experiment, Exp is a (nonlinear) least squares estimate to the exponent c in the convergence rate estimate ae^{-bN^c}.

From the data, it appears all methods converged at asymptotically optimal rates. The convergence curves generally track the target reference feedback loop. The *PLTMG* and marking schemes do depart a bit from the reference scheme, especially for smaller values of N. This is due in part to the effect of the a posteriori error estimate. Since $u \in \mathcal{H}^{5/4-\varepsilon}$, it is not smooth enough to satisfy the assumptions underlying (9). In the initial mesh all eight elements have a vertex at the origin, and as a result we expect the a posteriori error estimates based on (9) to be poor. After the mesh has been sufficiently refined, almost all the elements do not include the origin as a vertex, u is now sufficiently smooth in most of the elements, and the curves track the reference scheme more closely. The departure from the reference curve for h-refinement becomes larger as p increases, partly due to the stronger regularity needed to justify the a posteriori error estimate, and partly because there are more degrees of freedom associated with each element. For $p = 1$ a mesh with N degrees of freedom contains T elements; for $p > 1$, N degrees of freedom corresponds to a mesh with approximately T/p^2 elements. The *PLTMG* strategy departs from the reference curve somewhat more than the marking scheme; this is likely because the *PLTMG* allows a given element to be h-refined multiple times in a single feedback loop. With increasing p, the marking scheme takes increasingly many feedback loops to reach $N = 250K$ degrees of freedom, as predicted by our analysis. Also as predicted, the *PLTMG* scheme takes fewer loops than the marking scheme, but the number of loops required also increases with p.

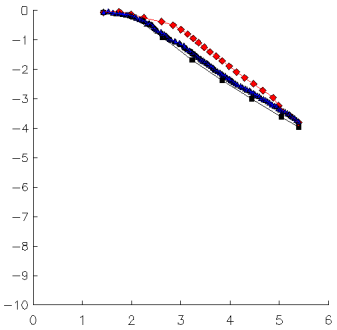
References

1. Babuška, I., Vogelius, M.: Feedback and adaptive finite element solution of one-dimensional boundary value problems. Numer.

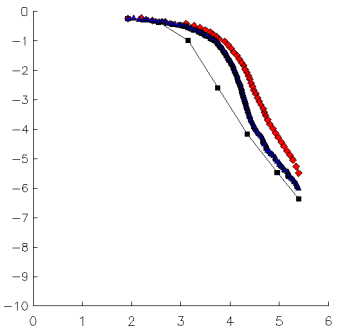
h-refinement, $p=1$			
	Digits	Order	Loops
reference	2.30	1.09	9
pltmg	2.29	1.26	13
marking	2.26	1.10	69



h-refinement, $p=2$			
	Digits	Order	Loops
reference	3.95	2.23	8
pltmg	3.81	2.30	23
marking	3.76	2.12	105



h-refinement, $p=4$			
	Digits	Order	Loops
reference	6.36	4.45	7
pltmg	5.49	4.21	46
marking	5.99	4.53	260



hp-refinement			
	Digits	Exp	Loops
reference	9.48	0.24	9
pltmg	7.57	0.30	24

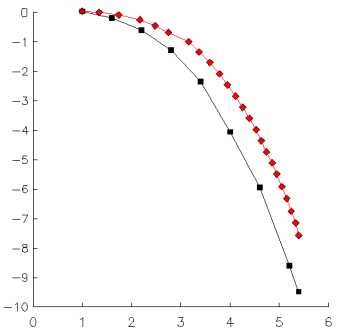


Fig. 3 The black convergence curves are for the reference scheme, red for the *PLTMG* scheme, and blue for the marking scheme. In the tables. Digits $= -\log(\|\nabla(u-u_I)\|/\|\nabla u\|)$, Loops is the number of adaptive feedback loops needed to reach 250K dofs. For the h-refinement, Order is a least squares estimate of the the convergence rate. For hp-refinement, Exp is a (nonlinear) least squares estimate of the exponent. In the graphs, the x-axis is $\log N$ and the y-axis is $\log(\|\nabla(u-u_I)\|/\|\nabla u\|)$.

Math. **44**(1), 75–102 (1984). DOI 10.1007/BF01389757. URL http://dx.doi.org/10.1007/BF01389757
2. Bank, R.E.: PLTMG: A software package for solving elliptic partial differential equations, users' guide 12.0. Tech. rep., Department of Mathematics, University of California at San Diego (2016)
3. Bank, R.E., Deotte, C.: Adventures in adaptivity. Computing and Visualization in Science (to appear)
4. Bank, R.E., Nguyen, H.: *hp* adaptive finite elements based on derivative recovery and superconvergence. Comput. Vis. Sci. **14**(6), 287–299 (2011). DOI 10.1007/s00791-012-0179-7. URL http://dx.doi.org/10.1007/s00791-012-0179-7
5. Bank, R.E., Sherman, A.H., Weiser, A.: Refinement algorithms and data structures for regular local mesh refinement. In: Scientific computing (Montreal, Que., 1982), IMACS Trans. Sci. Comput., I, pp. 3–17. IMACS, New Brunswick, NJ (1983)
6. Bank, R.E., Xu, J., Zheng, B.: Superconvergent derivative recovery for Lagrange triangular elements of degree p on unstructured grids. SIAM J. Numer. Anal. **45**(5), 2032–2046 (electronic) (2007). DOI 10.1137/060675174. URL http://dx.doi.org/10.1137/060675174
7. Bank, R.E., Yserentant, H.: A note on interpolation, best approximation, and the saturation property. Numer. Math. **131**(1), 199–203 (2015). DOI 10.1007/s00211-014-0687-0. URL http://dx.doi.org/10.1007/s00211-014-0687-0
8. Bürg, M., Dörfler, W.: Convergence of an adaptive *hp* finite element strategy in higher space-dimensions. Appl. Numer. Math. **61**(11), 1132–1146 (2011). DOI 10.1016/j.apnum.2011.07.008. URL http://dx.doi.org/10.1016/j.apnum.2011.07.008
9. Carstensen, C., Gallistl, D., Gedicke, J.: Justification of the saturation assumption. Numer. Math. **134**(1), 1–25 (2016). DOI 10.1007/s00211-015-0769-7. URL http://dx.doi.org/10.1007/s00211-015-0769-7
10. Demlow, A., Stevenson, R.: Convergence and quasi-optimality of an adaptive finite element method for controlling L_2 errors. Numer. Math. **117**(2), 185–218 (2011). DOI 10.1007/s00211-010-0349-9. URL http://dx.doi.org/10.1007/s00211-010-0349-9
11. Dörfler, W.: A convergent adaptive algorithm for Poisson's equation. SIAM J. Numer. Anal. **33**(3), 1106–1124 (1996). DOI 10.1137/0733054. URL http://dx.doi.org/10.1137/0733054
12. Dörfler, W., Nochetto, R.H.: Small data oscillation implies the saturation assumption. Numer. Math. **91**(1), 1–12 (2002). DOI 10.1007/s002110100321. URL http://dx.doi.org/10.1007/s002110100321
13. Guo, B., Babuška, I.: The *hp* version of the finite element method. part 1: the basic approximation results. Computational Mechanics **1**(1), 21–41 (1986)
14. Holst, M., Pollock, S.: Convergence of goal-oriented adaptive finite element methods for nonsymmetric problems. Numer. Methods Partial Differential Equations **32**(2), 479–509 (2016). DOI 10.1002/num.22002. URL http://dx.doi.org/10.1002/num.22002
15. Holst, M., Pollock, S., Zhu, Y.: Convergence of goal-oriented adaptive finite element methods for semilinear problems. Comput. Vis. Sci. **17**(1), 43–63 (2015). DOI 10.1007/s00791-015-0243-1. URL http://dx.doi.org/10.1007/s00791-015-0243-1
16. Mitchell, W.F.: A collection of 2D elliptic problems for testing adaptive grid refinement algorithms. Appl. Math. Comput. **220**, 350–364 (2013). DOI 10.1016/j.amc.2013.05.068. URL http://dx.doi.org/10.1016/j.amc.2013.05.068
17. Mitchell, W.F.: How high a degree is high enough for high order finite elements? Procedia Computer Science **51**, 246–255 (2015)
18. Mitchell, W.F., McClain, M.A.: A survey of *hp*-adaptive strategies for elliptic partial differential equations. In: Recent advances in computational and applied mathematics, pp. 227–258. Springer, Dordrecht (2011). DOI 10.1007/978-90-481-9981-5_10. URL http://dx.doi.org/10.1007/978-90-481-9981-5_10
19. Mitchell, W.F., McClain, M.A.: A comparison of *hp*-adaptive strategies for elliptic partial differential equations. ACM Trans. Math. Software **41**(1), Art. 2, 39 (2014). DOI 10.1145/2629459. URL http://dx.doi.org/10.1145/2629459
20. Nochetto, R.H., Siebert, K.G., Veeser, A.: Theory of adaptive finite element methods: an introduction. In: Multiscale, nonlinear and adaptive approximation, pp. 409–542. Springer, Berlin (2009). DOI 10.1007/978-3-642-03413-8_12. URL http://dx.doi.org/10.1007/978-3-642-03413-8_12
21. Nochetto, R.H., Veeser, A.: Primer of adaptive finite element methods. In: Multiscale and adaptivity: modeling, numerics and applications, *Lecture Notes in Math.*, vol. 2040, pp. 125–225. Springer, Heidelberg (2012). DOI 10.1007/978-3-642-24079-9. URL http://dx.doi.org/10.1007/978-3-642-24079-9
22. Stevenson, R.: Optimality of a standard adaptive finite element method. Found. Comput. Math. **7**(2), 245–269 (2007). DOI 10.1007/s10208-005-0183-0. URL http://dx.doi.org/10.1007/s10208-005-0183-0
23. Verfürth, R.: A posteriori error estimation techniques for finite element methods. Numerical Mathematics and Scientific Computation. Oxford University Press, Oxford (2013). DOI 10.1093/acprof:oso/9780199679423.001.0001. URL http://dx.doi.org/10.1093/acprof:oso/9780199679423.001.0001