Performance of *hp*-Adaptive Strategies for 3D Elliptic Problems

William F. Mitchell *

National Institute of Standards and Technology, Gaithersburg, MD 20899, USA

Abstract. The *hp* version of the finite element method (*hp*-FEM) combined with adaptive mesh refinement is a particularly efficient method for solving partial differential equations (PDEs) because it can achieve an exponential convergence rate in the number of degrees of freedom. *hp*-FEM allows for refinement in both the element size, h, and the polynomial degree, p. Like adaptive refinement for the h version of the finite element method, a posteriori error estimates can be used to determine where the mesh needs to be refined, but a single error estimate can not simultaneously determine whether it is better to do the refinement by h or p. Several strategies for making this determination have been proposed over the years. In a recent study [Mitchell, W.F. and McClain, M.A., ACM Trans. Math. Software 41(1), 2:1-2:39 (Oct 2014)], the effectiveness of 13 strategies for 2D elliptic PDEs were compared in terms of the number of degrees of freedom and computation time needed to reach an error tolerance. This paper presents the results of a similar study for 3D elliptic PDEs. It was found that the results are very similar to those of the 2D study. For minimizing the computation time, the method based on a priori knowledge is very effective when there are known point singularities, and the method based on the decay rate of the expansion coefficients is very effective across all categories of problems. For minimizing the number of degrees of freedom, the method based on a priori knowledge works well for easy problems and problems with singularities, and the method that uses a "type parameter" works well for hard problems.

Keywords: adaptive mesh refinement, *hp*-adaptive strategy, *hp*-FEM, 3D elliptic partial differential equations

1 Introduction

The use of adaptive mesh refinement with the finite element method for the numerical solution of partial differential equations (PDEs) has been studied for more than 35 years now. For many years, most of the work focused on h-adaptive

 Preprint submitted to the proceedings of the Computation and Information Science Conference, Portaria, Greece, 2016.

methods where the mesh size, h, is adapted locally by means of a local error estimator with the goal of placing the smallest elements in the areas where they will do the most good. Recently, the research community has begun to focus more attention on *hp*-adaptive methods. In these methods, one not only locally adapts the size of the mesh, but also the degree of the polynomials, p. The attraction of *hp*-adaptivity is that the error converges at an exponential rate in the number of degrees of freedom, as opposed to a polynomial rate for fixed p. The new complication is that the local error estimator is no longer sufficient to guide the adaptivity. It tells you which elements should be refined, but it does not indicate whether it is better to refine the element by h or by p. A method for making that determination is called an *hp*-adaptive strategy. A number of strategies have been proposed over the years. Mitchell and McClain [13] performed a numerical experiment to compare the effectiveness of 13 such strategies for 2D elliptic PDEs under different situations. In this paper we present the results of a similar experiment for 3D elliptic PDEs. We consider 7 of the *hp*-adaptive strategies from [13]; those that did not perform well in terms of minimizing the computation time to reach an error tolerance in the 2D experiment are not included.

As in [13], the comparison will be restricted to steady-state linear elliptic partial differential equations on bounded domains. The standard Galerkin finite element method will be used with the space of continuous piecewise polynomial functions over tetrahedra that are refined by the bisection method of Arnold, Mukherjee and Pouly [3].

The remainder of the paper is organized as follows. In Sect. 2 we briefly review the *hp*-adaptive finite element algorithm used in the experiments. Sect. 3 defines the *hp*-adaptive strategies to be compared. Sect. 4 contains the results of the experiments. Finally, we draw our conclusions in Sect. 5.

2 The Finite Element Method

In this section we briefly review the *hp*-adaptive finite element method used in the experiment. This is a natural extension to 3D of the method described in more detail in [13].

We consider the elliptic partial differential equation

$$\mathcal{L}u := -\nabla^2 u + c(x,y,z)u = f(x,y,z) \quad \text{in } \Omega \tag{1}$$

$$u = g(x,y,z) \quad \text{on } \partial\Omega \tag{2}$$

where Ω is a bounded, connected, polyhedral, open region in $\mathbb{R}^3$ with boundary $\partial\Omega$.

We denote by $H^m(\Omega)$ the usual Sobolev spaces consisting of those functions whose derivatives up to order m are in $L^2(\Omega)$, with the usual Sobolev norms $||\cdot||_{H^m(\Omega)}$. Define the bilinear form

$$B(u,v) = \int_\Omega \frac{\partial u}{\partial x}\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y}\frac{\partial v}{\partial y} + \frac{\partial u}{\partial z}\frac{\partial v}{\partial z} + cuv \tag{3}$$

and the linear form

$$L(v) = \int_{\Omega} fv \ . \tag{4}$$

The energy norm is defined by $||u||_{E(\Omega)} = B(u,u)^{1/2}$.

The finite element space is defined by partitioning Ω into a mesh, G_{hp}, consisting of a set of N_T tetrahedral elements, $\{T_i\}_{i=1}^{N_T}$ with $\bar{\Omega} = \cup_{i=1}^{N_T} \bar{T}_i$. We only consider compatible meshes, i.e. $\bar{T}_i \cap \bar{T}_j$, $i \neq j$, is either empty, a common face, a common edge, or a common vertex. The diameter of the element is denoted h_i. With each element we associate an integer degree $p_i \geq 1$. The finite element space V_{hp} is the space of continuous piecewise polynomial functions on Ω such that over element T_i it is a polynomial of degree p_i. The degree of a face or an edge is determined by applying the minimum rule in which the face or edge is assigned the minimum of the degrees of the adjacent elements.

Let E be the set of edges of the mesh that are on $\partial\Omega$. Let $V_D = \{v \in V_{hp} : v = 0 \text{ on all } T_i \text{ for which } \bar{T}_i \cap \partial\Omega = \emptyset\}$. Let $\hat{u}_D \in V_D$ be a lift function that approximates the Dirichlet boundary conditions via

- $\hat{u}_D = \hat{u}_v + \hat{u}_e + \hat{u}_f$ where $\hat{u}_v$ is piecewise linear and $\hat{u}_f$ is 0 on E,
- $\hat{u}_v = g$ at the element vertices on $\partial\Omega$,
- $\int_E (\hat{u}_e - (g - \hat{u}_v))v = 0 \ \ \forall v \in V_D$, and
- $\int_{\partial\Omega} (\hat{u}_f - (g - \hat{u}_v - \hat{u}_e))v = 0 \ \ \forall v \in V_D$.

The finite element solution is the unique function $u_{hp} \in \hat{u}_D + (V_{hp} \setminus V_D)$ that satisfies

$$B(u_{hp}, v_{hp}) = L(v_{hp}) \ \ \forall v_{hp} \in V_{hp} \setminus V_D \ . \tag{5}$$

The error is defined by $e_{hp} = u - u_{hp}$.

The finite element solution is expressed as a linear combination of basis functions $\{\phi_i\}_{i=1}^{N}$ that span V_{hp},

$$u_{hp} = \sum_{i=1}^{N} \alpha_i \phi_i(x,y,z) \ . \tag{6}$$

We use the p-hierarchical basis of Szabo and Babuška [16].

The discrete form of the problem is a linear system of algebraic equations

$$Ax = b \tag{7}$$

where the matrix A is given by $A_{ji} = B(\phi_i, \phi_j)$ and the right hand side is given by $b_j = L(\phi_j)$.

If h and p are uniform over the mesh, $u \in H^m(\Omega)$, and the other usual assumptions are met, then the a priori error bound is [4]

$$||e_{hp}||_{H^1(\Omega)} \leq C \frac{h^{\mu}}{p^{m-1}} ||u||_{H^m(\Omega)} \tag{8}$$

where $\mu = \min(p, m-1)$ and C is a constant that is independent of h, p and u, but depends on m.

With a suitably chosen hp mesh, and other typical assumptions, the error can be shown [6] to converge exponentially in the number of degrees of freedom, N,

$$||e_{hp}||_{H^1(\Omega)} \leq C_1 e^{-C_2 N^{1/3}} \tag{9}$$

where C_1 and $C_2 > 0$ are constants that are independent of N.

```
begin with a very coarse mesh
form and solve the linear system
repeat
   determine which elements to refine and whether to refine by h or p
   refine elements
   form and solve the linear system
until some termination criterion is met
```

Fig. 1. Basic form of an hp-adaptive algorithm.

The basic form of the hp-adaptive algorithm is given in Fig. 1.

For h-refinement, we use the bisection method of Arnold, Mukherjee and Pouly [3], while p-refinement means increasing the degree of the element by one, followed by enforcing the minimum rule for the faces and edges. Adaptive refinement is guided by an explicit a posteriori error estimate (see, for example, [1]). For element T_i with degree p_i, the error estimate, η_i, is given by

$$\eta_i^2 = \frac{h_i^2}{p_i^2} ||r||^2_{L_2(T_i)} + \frac{h_i}{2p_i} ||R||^2_{L_2(\partial T_i)} \tag{10}$$

where r is the interior residual

$$r = f + \nabla^2 u_{hp} - c u_{hp} \tag{11}$$

and R is the jump in the outward normal derivative of u_{hp} across the element boundary. The elements chosen to be refined are those for which the error estimate is larger than one tenth of the largest error estimate. The strategies for determining whether to refine by h or p will be described in Sect. 3.

The primary criterion for program termination is that the relative energy norm of the error be smaller than a given tolerance τ, i.e., $||e_{hp}||_{E(\Omega)}/||u||_{E(\Omega)} < \tau$. An upper bound on the number of degrees of freedom is used as a secondary criterion to avoid run away programs when convergence is slow.

3 The *hp*-Adaptive Strategies

This section describes the hp-adaptive strategies. The extension of the strategies from 2D to 3D is straight forward, so there is little difference between these

descriptions and those in [13]. For brevity, many of the details have been omitted. For a detailed description of the strategies, see [12]. The values used for the parameters of the strategies were determined by a preliminary experiment to determine a single value (or possibly two values dependent on possible singularity) that generally works best using a subset of the test problems, and are generally different than those used in the 2D experiment.

3.1 Use of a priori Knowledge of Solution Regularity

A consequence of the a priori error bound in 8 is that for smooth solutions p-refinement will produce an exponential rate of convergence, but near singularities p-refinement is less effective than h-refinement. For this reason, many of the hp strategies use h-refinement in areas where the solution is irregular (i.e., locally fails to be in H^m for some finite m) or nearly irregular, and p-refinement elsewhere. The simplest strategy is to use any a priori knowledge about irregularities. Ainsworth and Senior [2] simply flag vertices in the initial mesh as being possible trouble spots, and use h-refinement for elements that contain a flagged vertex and p-refinement elsewhere. We will refer to this strategy by the name APRIORI.

3.2 Type Parameter

Gui and Babuška [5] presented an hp-adaptive strategy using what they call a type parameter, γ. We will refer to this strategy as TYPEPARAM.

Given the error estimates η_{i,p_i} and η_{i,p_i-1}, define

$$R(T_i) = \begin{cases} \frac{\eta_{i,p_i}}{\eta_{i,p_i-1}} & \eta_{i,p_i-1} \neq 0 \\ 0 & \eta_{i,p_i-1} = 0 \end{cases} \tag{12}$$

where by convention, $\eta_{i,0} = 0$, which forces p-refinement if $p_i = 1$.

The quantity R is used to assess the perceived solution smoothness. Given the type parameter, $0 \leq \gamma < \infty$, element T_i is refined by h-refinement if $R(T_i) > \gamma$ and by p-refinement if $R(T_i) \leq \gamma$.

For the results of Sect. 4, we use $\gamma = 0.6$ if the solution has a singularity, and $\gamma = 1.2$ otherwise.

3.3 Estimate Regularity Using Smaller p Estimates

One approach that estimates the regularity is given by Süli, Houston and Schwab [15]. This strategy is based on 8 and an estimate of the convergence rate in p using error estimates based on $p_i - 2$ and $p_i - 1$. We will refer to this strategy as PRIOR2P.

Suppose the error estimate in 8 holds on individual elements and that the inequality is an approximate equality. Let η_{i,p_i-2} and η_{i,p_i-1} be a posteriori error estimates for partial approximate solutions over triangle T_i using the bases up to

degree $p_i - 2$ and $p_i - 1$, respectively. Then assuming equality in 8, substituting in the error estimates and p_i's, and taking the ratio, one gets

$$\frac{\eta_{i,p_i-1}}{\eta_{i,p_i-2}} \approx \left(\frac{p_i - 1}{p_i - 2}\right)^{-(m_i-1)} \tag{13}$$

and thus the regularity is estimated by

$$m_i \approx 1 - \frac{\log(\eta_{i,p_i-1}/\eta_{i,p_i-2})}{\log((p_i-1)/(p_i-2))} \ . \tag{14}$$

Use p-refinement if $p_i \leq m_i - 1$ and h-refinement otherwise.

3.4 Predict Error Estimate on Assumption of Smoothness

Melenk and Wohlmuth [9] proposed a strategy based on a prediction of what the error should be if the solution is smooth. We call this strategy SMOOTH_PRED.

When refining element T_i, assume the solution is locally smooth and that the optimal convergence rate is obtained. If h-refinement is performed, then the expected error on the two children of T_i is reduced by a factor of $\sqrt[3]{2}^{\,p_i}$ as indicated by 8. (This is different from the 2D case where it is $\sqrt{2}$ instead of $\sqrt[3]{2}$.) Thus if η_i is the error estimate for T_i, predict the error estimate of the children to be $\gamma_h \eta_i / \sqrt[3]{2}^{\,p_i}$ where γ_h is a user specified parameter. If p-refinement is performed on T_i, exponential convergence is expected, so the error should be reduced by some constant factor $\gamma_p \in (0, 1)$, i.e., the predicted error estimate of the p-refinement of T_i is $\gamma_p \eta_i$. When the actual error estimate of a child of T_i or p-refinement of T_i becomes available, it is compared to the predicted error estimate. If the error estimate is less than or equal to the predicted error estimate, then p-refinement is indicated for the child. Otherwise, h-refinement is indicated since presumably the assumption of smoothness was wrong. For the results in Sect. 4 we use $\gamma_h = 4$ and $\gamma_p = \sqrt{0.4}$.

3.5 Larger of *h*-Based and *p*-Based Error Indicators

In 1D, Schmidt and Siebert [14] proposed a strategy that uses two a posteriori error estimates to predict whether h-refinement or p-refinement will reduce the error more. We extend this strategy to bisected tetrahedra and refer to it as H&P_ERREST.

One error estimate is given by the local Neumann problem

$$\mathcal{L} e_i = f - \mathcal{L} u_{hp} \qquad \text{in } T_i \tag{15}$$

$$e_i = 0 \qquad \text{on } \partial T_i \cap \partial \Omega \tag{16}$$

$$\frac{\partial e_i}{\partial n} = -\frac{1}{2}\left[\frac{\partial u_{hp}}{\partial n}\right] \qquad \text{on } \partial T_i \setminus \partial \Omega \tag{17}$$

where $\left[\frac{\partial u_{hp}}{\partial n}\right]$ is the jump in the outward normal derivative of u_{hp} across the element boundary. The solution of this equation is approximated using the p-hierarchical bases of exact degree $p_i + 1$, i.e., bases up to degree p_i are omitted. It can be shown [13] that the norm of this approximation of e_i is actually an estimate of how much the error will be reduced if this element is p-refined.

A second error estimate is given by approximately solving the local problem

$$\mathcal{L}u_i = f \qquad \text{in } T_i \tag{18}$$

$$\frac{\partial u_i}{\partial n} = 0 \qquad \text{on } \partial T_{i,1} \tag{19}$$

$$u_i = 0 \qquad \text{on } \partial T_{i,2} \tag{20}$$

where $\partial T_{i,1}$ is the edge and two faces that would be bisected if T_i is h-refined, and $\partial T_{i,2}$ is the rest of the boundary of T_i. The solution of this equation is approximated using an h-refinement of T_i and the error estimate $e_i = u_i - u_{hp}$ is computed. The norm of this approximate solution is actually an estimate of how much the error will be reduced if this element is h-refined.

For more flexibility, one of the error estimates can be multiplied by a user specified constant to bias the refinement toward h- or p-refinement. In the results of Sect. 4 the p-based error estimate is multiplied by 2.

The type of refinement that is used is the one that corresponds to the larger of the two modified error estimates.

3.6 Decay Rate of Coefficients

In one dimension, the approximate solution in element T_i with degree p_i can be written

$$u_i(x) = \sum_{j=0}^{p_i} a_j P_j(x) \tag{21}$$

where P_j is the j^{th} degree Legendre polynomial scaled to the interval of element T_i. Mavriplis [8] estimates the decay rate of the coefficients by a least squares fit of the last four coefficients a_j to $Ce^{-\sigma j}$. Elements are refined by p-refinement where $\sigma > 1$ and by h-refinement where $\sigma \leq 1$. We refer to this strategy as COEF_DECAY.

In this paper we are using tetrahedral elements which have a p-hierarchical basis that is based on Legendre polynomials [16]. In this basis, an element of degree p_i has 4 basis functions of degree 1 and $(j+1)(j+2)/2$ basis functions of exact degree j, $1 < j \leq p_i$, so we don't have a unique Legendre polynomial coefficient to use for each degree. Instead, for the coefficients a_j we use the ℓ_1 norm of the coefficients of the degree j basis functions, i.e.

$$a_j = \sum_{\substack{k \text{ s.t. } \deg(\phi_k)=j \\ \text{supp}(\phi_k)\cap T_i \neq \emptyset}} |\alpha_k| \ . \tag{22}$$

3.7 Root Test on Coefficients

Houston et al. [7] present an approach which uses the Legendre coefficients to estimate the regularity of the solution. The regularity is estimated using the root test yielding

$$m_i = \frac{\log\left(\frac{2p_i+1}{2a_{p_i}^2}\right)}{2\log p_i} \quad . \tag{23}$$

If $p_i = 1$, use p-refinement. Otherwise, use p-refinement if $p_i \leq m_i - 1$ and h-refinement if $p_i > m_i - 1$. We use the modified a_j of Sect. 3.6 and refer to this strategy as COEF_ROOT.

4 Numerical Results

This section contains the results of numerical experiments to compare the hp-adaptive strategies' performance on a suite of 8 test problems with various difficulties that adaptive refinement should locate. Two metrics are used to compare the strategies: the number of degrees of freedom required to reach a given error tolerance, and the computation time required to reach the tolerance. The test problems and numerical results for each problem are given in Sect. 4.1, with a summary of the results in Sect. 4.2. Each problem is characterized as being easy, hard or singular. A problem is singular if the solution has a singularity in the closure of the domain. We consider a problem to be hard if it requires a large amount of resources to resolve the difficulty, and easy otherwise. In this paper, we consider a problem to be easy if it can be solved by most of the strategies to a tolerance of 10^{-4} with fewer than 2 million degrees of freedom. For the hard problems, many of the strategies require much more than 2 million degrees of freedom for that tolerance, so a larger tolerance is used.

These test problems come from the NIST AMR benchmark suite [11]. Recall that Poisson's equation is $-\nabla^2 u = f(x, y, z)$ and Laplace's equation is Poisson's equation with $f = 0$. All problems have Dirichlet boundary conditions.

Each problem is solved with each hp strategy using the hp-adaptive algorithm of Sect. 2. The problems are solved with low accuracy and high accuracy tolerances. At the end of each run the number of degrees of freedom and total "wall clock" time to solution are recorded.

The results are given in bar charts in Figs. 2–9. The gray bars indicate the number of degrees of freedom required to reach the tolerance, and the black bars indicate the computation time required to reach the tolerance. All results are scaled by the value of the strategy that performed best, so, for example, a value of 1.0 indicates the best strategy, and a value of 0.2 indicates the strategy needed five times as many degrees of freedom or took five times longer than the best strategy. Instances where both of the bars are missing indicate cases where the strategy was unable to achieve the given tolerance within the allowed number of degrees of freedom.

These computations were performed using the adaptive finite element code PHAML Version 1.16.1 [10] run as a sequential code on a 2.6 GHz Intel Xeon E5-2630 v2 based computer operating under the CentOS 5.11 release of Linux with kernel 2.6.18-274.3.1.el5. PHAML was compiled with the Intel Fortran compiler version 15.0.3. [1]

4.1 Test Problems and Convergence Graphs

Polynomial. This is a simple problem with a polynomial solution. It is Poisson's equation on the unit cube. The solution is the polynomial

$$64^q x^q (1-x)^q y^q (1-y)^q z^q (1-z)^q \tag{24}$$

with $q = 10$. 64^q is a normalization factor so that the L^∞ norm is 1.0. The tolerances are $\tau = 10^{-2}$ for low accuracy and $\tau = 10^{-4}$ for high accuracy. For the APRIORI strategy, we choose to always refine by p, i.e., it is just p-adaptive refinement. This is categorized as an easy problem. Results are shown in Fig. 2.

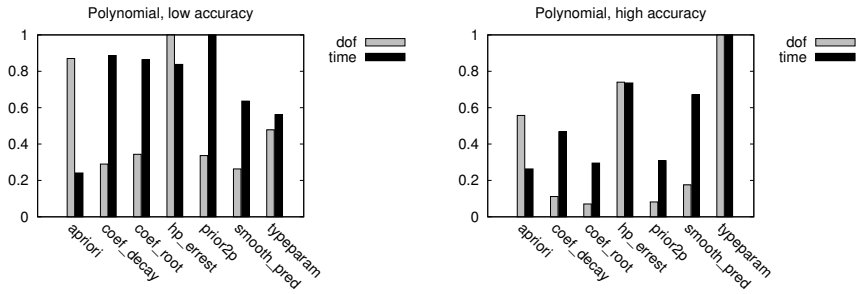

Fig. 2. Relative performance of the strategies in degrees of freedom and wall clock time for low accuracy ($\tau = 10^{-2}$) and high accuracy ($\tau = 10^{-4}$) solution of the polynomial problem.

Peak. The peak problem contains a Gaussian peak in the interior of the domain. It is Poisson's equation on the unit cube. The solution is

$$e^{-\alpha((x-x_c)^2+(y-y_c)^2+(z-z_c)^2)} \tag{25}$$

where (x_c, y_c, z_c) is the location of the peak, and α determines the strength of the peak. We use $\alpha = 10^5$ and $(x_c, y_c, z_c) = (0.5, 0.5, 0.5)$. The tolerances are

[1] The mention of specific products, trademarks, or brand names is for purposes of identification only. Such mention is not to be interpreted in any way as an endorsement or certification of such products or brands by the National Institute of Standards and Technology. All trademarks mentioned herein belong to their respective owners.

$\tau = 10^{-2}$ for low accuracy and $\tau = 10^{-4}$ for high accuracy. The APRIORI strategy refines by h if the element touches the center of the peak and by p otherwise. This is categorized as an easy problem. Results are shown in Fig. 3.

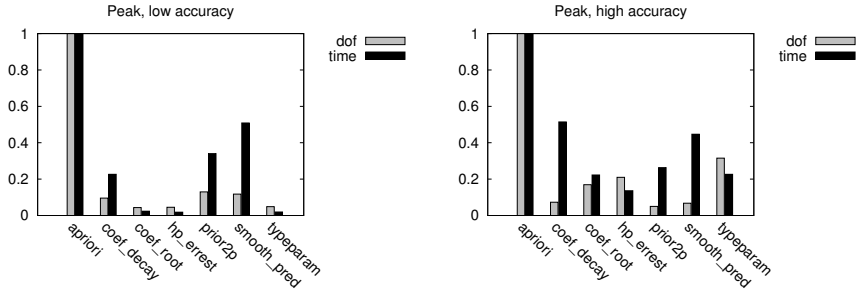

Fig. 3. Relative performance of the strategies in degrees of freedom and wall clock time for low accuracy ($\tau = 10^{-2}$) and high accuracy ($\tau = 10^{-4}$) solution of the peak problem.

Mild Wave Front. The 2D circular wave front problem is often used as an example in adaptive mesh refinement papers. This is a 3D version of it with a spherical wave front. It is Poisson's equation on the unit cube. The solution is

$$\tan^{-1}(\alpha(r - r_0)) \tag{26}$$

where $r = \sqrt{(x - x_c)^2 + (y - y_c)^2 + (z - z_c)^2}$. The location of the wave front is defined by a sphere with radius r_0 and center (x_c, y_c, z_c). The scalar α determines the steepness of the wave front. For the easy form of this problem we use $\alpha = 20$, $(x_c, y_c, z_c) = (-.05, -.05, -.05)$, and $r_0 = 0.7$. The center is chosen outside the domain so that only the wave front is a factor in the adaptivity, not the mild singularity at the center of the sphere. The tolerances are $\tau = 10^{-2}$ for low accuracy and $\tau = 2 \times 10^{-4}$ for high accuracy. For the APRIORI strategy, refine by h if the element touches the sphere that defines the location of the wave front, and by p otherwise. Results are shown in Fig. 4.

Strong Wave Front. In the hard version of the wave front problem the location of the wave front is the same, but it is much steeper. The parameters are $\alpha = 100$, $(x_c, y_c, z_c) = (-.05, -.05, -.05)$, and $r_0 = 0.7$. The tolerances are $\tau = 5 \times 10^{-2}$ for low accuracy and $\tau = 6 \times 10^{-3}$ for high accuracy. Results are shown in Fig. 5.

Boundary Layer. The boundary layer problem is a convection-diffusion equation with first order terms on $(-1, 1) \times (-1, 1) \times (-1, 1)$. The equation is

$$-\epsilon \nabla^2 u + 2\frac{\partial u}{\partial x} + \frac{\partial u}{\partial y} + \frac{\partial u}{\partial z} = f \tag{27}$$

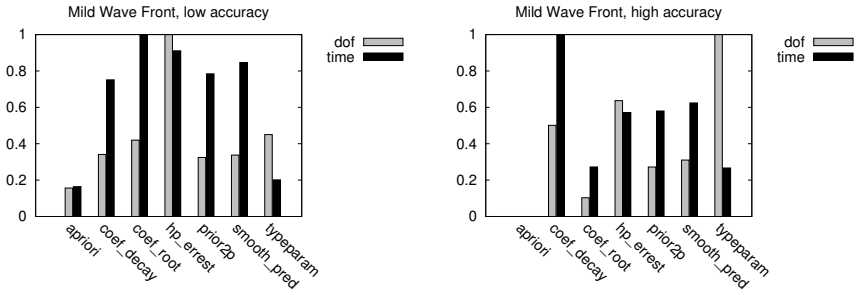

Fig. 4. Relative performance of the strategies in degrees of freedom and wall clock time for low accuracy ($\tau = 10^{-2}$) and high accuracy ($\tau = 2 \times 10^{-4}$) solution of the mild wave front problem.

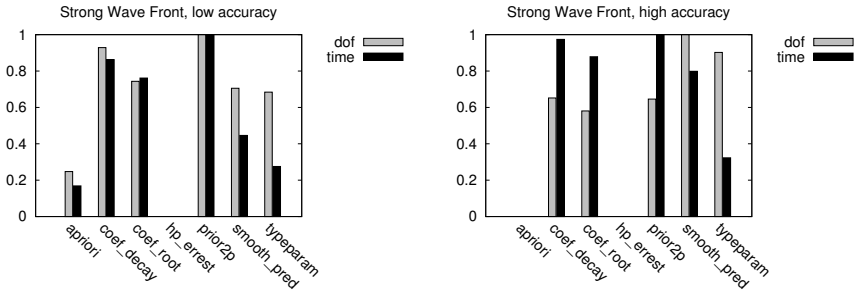

Fig. 5. Relative performance of the strategies in degrees of freedom and wall clock time for low accuracy ($\tau = 5 \times 10^{-2}$) and high accuracy ($\tau = 6 \times 10^{-3}$) solution of the strong wave front problem.

and the solution is

$$(1 - e^{-(1-x)/\epsilon})(1 - e^{-(1-y)/\epsilon})(1 - e^{-(1-z)/\epsilon}) \cos(\pi(x + y + z)) \tag{28}$$

where ϵ controls the strength of the boundary layer. We use $\epsilon = 10^{-1}$. The tolerances are $\tau = 2 \times 10^{-2}$ for low accuracy and $\tau = 10^{-3}$ for high accuracy. In the APRIORI strategy we refine by h if the element touches any of the boundaries with a boundary layer, and by p otherwise. This is categorized as a hard problem. Results are shown in Fig. 6.

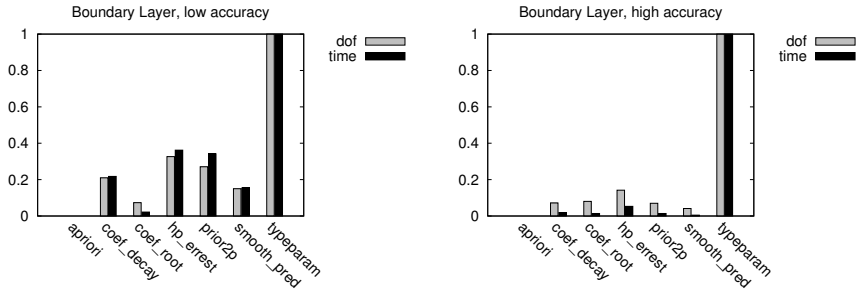

Fig. 6. Relative performance of the strategies in degrees of freedom and wall clock time for low accuracy ($\tau = 2 \times 10^{-2}$) and high accuracy ($\tau = 10^{-3}$) solution of the boundary layer problem.

Oscillatory. The oscillatory problem contains several waves that form spherical shells which get closer together as you approach the origin. The PDE is a Helmholtz equation on the unit cube. The equation is

$$-\nabla^2 u - \frac{1}{(\alpha + r)^4} u = f \tag{29}$$

and the solution is

$$\sin\left(\frac{1}{\alpha + r}\right) \tag{30}$$

where $r = \sqrt{x^2 + y^2 + z^2}$. The number of oscillations, N, is determined by the parameter $\alpha = \frac{1}{N\pi}$. We use $N = 10$. The tolerances are $\tau = 2 \times 10^{-2}$ for low accuracy and $\tau = 10^{-3}$ for high accuracy. For APRIORI, refine by h if the element touches the origin and by p otherwise. This is categorized as a hard problem. Results are shown in Fig. 7.

Fichera Corner with Vertex Singularity. The Fichera corner is the 3D analogue of the 2D L-domain problem. It is Poisson's equation on $(-1, 1)^3 \setminus [0, 1)^3$, i.e., a cube with one octant removed. In the vertex singularity form of the problem, the solution is

$$\left(\sqrt{x^2 + y^2 + z^2}\right)^q \quad . \tag{31}$$

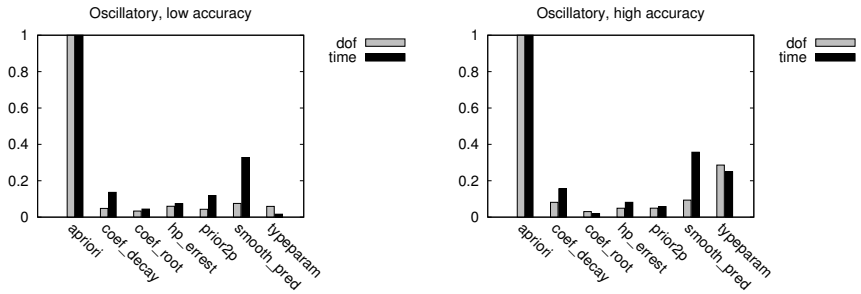

Fig. 7. Relative performance of the strategies in degrees of freedom and wall clock time for low accuracy ($\tau = 2 \times 10^{-2}$) and high accuracy ($\tau = 10^{-3}$) solution of the oscillatory problem.

The parameter q determines the strength of the singularity at the origin; the Sobolev regularity is $1.5 + q$. We use $q = 1/2$. The tolerances are $\tau = 10^{-2}$ for low accuracy and $\tau = 2.5 \times 10^{-4}$ for high accuracy. For APRIORI, refine by h if the element contains the origin and $p_i + 1 > 1.5 + q$, and by p otherwise. This is categorized as a singular problem. Results are shown in Fig. 8.

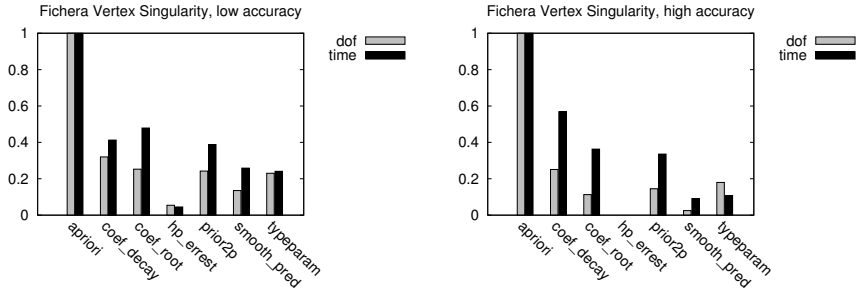

Fig. 8. Relative performance of the strategies in degrees of freedom and wall clock time for low accuracy ($\tau = 10^{-2}$) and high accuracy ($\tau = 2.5 \times 10^{-4}$) solution of the Fichera corner with vertex singularity problem.

Fichera Corner with Vertex and Edge Singularities. In the vertex and edge singularities form of the Fichera corner problem, the domain is the same, but the Poisson equation is

$$-\nabla^2 u = 1/r \tag{32}$$

where $r = \sqrt{x^2 + y^2 + z^2}$. The solution is singular along the edges of the reentrant octant as well as at the origin. The exact solution is not known. To measure

the error, the computed solution is compared to a more accurate reference solution, which has an estimated energy norm error of 1.69×10^{-3}. The tolerances are $\tau = 5 \times 10^{-2}$ for low accuracy and $\tau = 6 \times 10^{-3}$ for high accuracy. For APRIORI, refine by h if the element contains the origin or one of the reentrant edges and $p_i + 1 > 1.5$, and by p otherwise. This is categorized as a singular problem. Results are shown in Fig. 9.

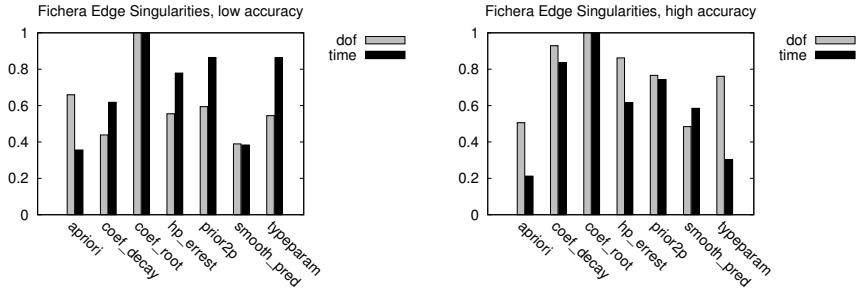

Fig. 9. Relative performance of the strategies in degrees of freedom and wall clock time for low accuracy ($\tau = 5 \times 10^{-2}$) and high accuracy ($\tau = 6 \times 10^{-3}$) solution of the Fichera corner with vertex and edge singularities problem.

4.2 Summary and Observations

Table 1. Number of problems for which each strategy required less than twice as many degrees of freedom as the best performing strategy.

	low accuracy			high accuracy		
strategy	easy	hard	singular	easy	hard	singular
APRIORI	2	1	2	2	1	2
COEF_DECAY	0	1	0	1	1	1
COEF_ROOT	0	1	1	0	1	1
H&P_ERREST	2	0	1	2	0	1
PRIOR2P	0	1	1	0	1	1
SMOOTH_PRED	0	1	0	0	1	0
TYPEPARAM	0	2	1	2	2	1

In this section, we summarize the results in Sect. 4.1 to examine the relative performance of the strategies in different situations. The test problems are grouped into six categories: easy problems, hard problems, and singular problems at low accuracy and high accuracy.

Table 2. Number of problems for which each strategy required less than twice as much computation time as the best performing strategy.

	low accuracy			high accuracy		
strategy	easy	hard	singular	easy	hard	singular
APRIORI	1	1	1	1	1	1
COEF_DECAY	2	1	1	2	1	2
COEF_ROOT	2	1	1	0	1	1
H&P_ERREST	2	0	1	2	0	1
PRIOR2P	2	1	1	1	1	1
SMOOTH_PRED	3	0	0	2	1	1
TYPEPARAM	1	1	1	1	1	0

While the reader may draw personal conclusions from the data given in Sect. 4.1, we use the following approach to attempt to summarize those results. We will consider a method to have performed well on a given problem if it is no more than a factor of two worse than the best performing method for that problem, i.e., no more than twice as many degrees of freedom or twice as much computation time. In Table 1 we give the number of problems in each of the six categories for which each strategy performed well in terms of minimizing the number of degrees of freedom to reach the error tolerance (hereinafter dof) and in Table 2 the number that performed well in terms of minimizing the computation time to reach the error tolerance (hereinafter time). Bear in mind there are three easy problems, three hard problems, and two singular problems. Based on these numbers, we make the following observations.

The overall performance of the strategies can be assessed by examining the row sums from Tables 1 and 2. For dof, APRIORI is best with TYPEPARAM a close second. SMOOTH_PRED is the worst and COEF_DECAY is tied for next to worst. But for time the results are just the opposite. COEF_DECAY is best with SMOOTH_PRED tied for second, and TYPEPARAM is worst with APRIORI tied for next to worst. This is counter intuitive, but may be due to APRIORI and TYPEPARAM tending to perform more p refinement (typically topping off at degree 8 to 10 for the low tolerance results) than SMOOTH_PRED and COEF_DECAY (typically topping off at degree 5 to 7). The higher degree elements can obtain higher accuracy with fewer degrees of freedom, but are computationally more expensive in number of operations to evaluate the basis functions, number of quadrature points needed, density of the stiffness matrix, etc.

Looking at the counts for individual categories in the dof table, APRIORI was best in the easy and singular problems for both low and high accuracy, while TYPEPARAM was best for the hard problems in both low and high accuracy. Also of note is H&P_ERREST tying for best in the easy problems, again for both low and high accuracy. We note there is very little difference between the low accuracy and high accuracy results.

In the individual results for time, SMOOTH_PRED works particularly well for easy problems, but not for the hard and singular problems, especially at low

accuracy. COEF_DECAY dominates Table 2 by tying for best in every category except for easy problems at low accuracy, where SMOOTH_PRED was best. This indicates that COEF_DECAY is a good choice as a general purpose strategy when minimizing the computation time is the objective.

From Fig. 8 we see that APRIORI dominates in both dof and time for the problem with a point singularity. We also observe this for the peak (Fig. 3) and oscillatory (Fig. 7) problems. For all three of these problems, the APRIORI strategy refines by h when an element contains a certain point and by p otherwise. This suggests that APRIORI works well not only for problems with singularities at known points, but also nonsingular problems where the difficulty is localized around known points.

5 Conclusion

In this paper we presented the results of a study of strategies for the hp-adaptive finite element method for 3D linear elliptic partial differential equations using bisection of tetrahedra. The hp-strategies are methods for determining how to select between the different possibilities of h- and p-refinement. Seven strategies were described and compared in a numerical experiment using eight test problems. Two metrics for comparison were used: the number of degrees of freedom needed to reach a tolerance in the relative energy norm of the error, and the computation time needed to reach that tolerance.

We found that if the objective is to minimize the number of degrees of freedom, APRIORI works best for easy and singular problems, and TYPEPARAM is best for hard problems. If the objective is to minimize the computation time, COEF_DECAY is a good general strategy, although SMOOTH_PRED might be considered for easy problems and APRIORI is good for problems with known point singularities and problems where the difficulty is localized around known points. These conclusions are similar to those drawn for 2D elliptic PDEs in [13], which recommends APRIORI for problems with known point singularities and COEF_DECAY as the best choice as a general strategy.

References

1. Ainsworth, M., Oden, J.T.: a posteriori Error Estimation in Finite Element Analysis. John Wiley & Sons, New York (2000)
2. Ainsworth, M., Senior, B.: hp-finite element procedures on non-uniform geometric meshes: adaptivity and constrained approximation. In: Bern, M.W., Flaherty, J.E., Luskin, M. (eds.) Grid Generation and Adaptive Algorithms. vol. 113, pp. 1–28. IMA Volumes in Mathematics and its Applications, Springer-Verlag, New York (1999)
3. Arnold, D.N., Mukherjee, A., Pouly, L.: Locally adapted tetrahedral meshes using bisection. SIAM J. Sci. Comput. 22, 431–448 (2000)
4. Babuška, I., Suri, M.: The h-p version of the finite element method with quasiuniform meshes. RAIRO Modél. Math. Anal. Numér. 21, 199–238 (1987)

5. Gui, W., Babuška, I.: The h, p and h-p versions of the finite element method in 1 dimension. Part 3: The adaptive h-p version. Numer. Math. 49, 659–683 (1986)
6. Guo, B., Babuška, I.: The h-p version of the finite element method. Part 1: The basic approximation results. Comput. Mech. 1, 21–41 (1986)
7. Houston, P., Senior, B., Süli, E.: Sobolev regularity estimation for hp-adaptive finite element methods. In: Brezzi, F., Buffa, A., Corsaro, S., Murli, A. (eds.) Numerical Mathematics and Advanced Appplications. pp. 619–644. Springer-Verlag, Berlin (2003)
8. Mavriplis, C.: Adaptive mesh strategies for the spectral element method. Comput. Methods Appl. Mech. Engrg. 116, 77–86 (1994)
9. Melenk, J.M., Wohlmuth, B.I.: On residual-based a-posteriori error estimation in hp-FEM. Adv. Comput. Math. 15, 311–331 (2001)
10. Mitchell, W.F.: PHAML home page. `http://math.nist.gov/phaml` (2012)
11. Mitchell, W.F.: NIST Adaptive Mesh Refinement Benchmark Problems. `http://math.nist.gov/amr-benchmark` (2015)
12. Mitchell, W.F., McClain, M.A.: A survey of hp-adaptive strategies for elliptic partial differential equations. In: Simos, T.E. (ed.) Recent Advances in Computational and Applied Mathematics. pp. 227–258. Springer (2011)
13. Mitchell, W.F., McClain, M.A.: A comparison of hp-adaptive strategies for elliptic partial differential equations. ACM Trans. Math. Software 41(1), 2:1–2:39 (Oct 2014)
14. Schmidt, A., Siebert, K.G.: a posteriori estimators for the h-p version of the finite element method in 1D. Appl. Numer. Math. 35, 43–66 (2000)
15. Süli, E., Houston, P., Schwab, C.: hp-finite element methods for hyperbolic problems. In: Whiteman, J. (ed.) The Mathematics of Finite Elements and Applications X. MAFELAP. pp. 143–162. Elsevier (2000)
16. Szabo, B., Babuška, I.: Finite Element Analysis. John Wiley and Sons, New York (1991)