NUMERICAL LINEAR ALGEBRA WITH APPLICATIONS
Numer. Linear Algebra Appl. 2007; **00**:1–25 *Prepared using* nlaauth.cls *[Version: 2002/09/18 v1.02]*

Efficiency-based *h*- and *hp*-refinement strategies for finite element methods

H. De Sterck[1], Thomas A. Manteuffel[2], Stephen F. McCormick[2], Josh Nolting[2], John Ruge[2], and Lei Tang[1]

[1]*Department of Applied Mathematics, University of Waterloo, Waterloo, Ontario, Canada*
[2]*Department of Applied Mathematics, University of Colorado at Boulder, Boulder, Colorado, USA*

SUMMARY

Two efficiency-based grid refinement strategies are investigated for adaptive finite element solution of partial differential equations. In each refinement step, the elements are ordered in terms of decreasing local error, and the optimal fraction of elements to be refined is determined based on efficiency measures that take both error reduction and work into account. The goal is to reach a pre-specified bound on the global error with minimal amount of work. Two efficiency measures are discussed, 'work times error' and 'accuracy per computational cost'. The resulting refinement strategies are first compared for a one-dimensional model problem that may have a singularity. Modified versions of the efficiency strategies are proposed for the singular case, and the resulting adaptive methods are compared with a threshold-based refinement strategy. Next, the efficiency strategies are applied to the case of *hp*-refinement for the one-dimensional model problem. The use of the efficiency-based refinement strategies is then explored for problems with spatial dimension greater than one. The 'work times error' strategy is inefficient when the spatial dimension, d, is larger than the finite element order, p, but the 'accuracy per computational cost' strategy provides an efficient refinement mechanism for any combination of d and p. Copyright © 2007 John Wiley & Sons, Ltd.

KEY WORDS: adaptive refinement; finite element methods; *hp*-refinement

1. Introduction

Adaptive finite element methods are being used extensively as powerful tools for approximating solutions of partial differential equations (PDEs) in a variety of application fields, see, e.g., [1, 2, 3]. This paper investigates the behavior of two efficiency-based grid refinement strategies for adaptive finite element solution of PDEs. It is assumed that a sharp, easily computed local *a posteriori* error estimator is available for the finite element method. In each refinement step, the elements are ordered in terms of decreasing local error, and the optimal fraction of elements to be refined in the current step is determined based on efficiency measures that take both error reduction and work into account. The goal is to reach a pre-specified bound on the global error with a minimal amount of work. It is assumed that optimal solvers are used for the discrete linear systems, and that the computational work for solving these systems is, thus, proportional to the number of degrees of freedom (DOF). Two efficiency measures are discussed. The first efficiency measure is 'work times error' efficiency (WEE), which was

Received April 19, 2007

Copyright © 2007 John Wiley & Sons, Ltd.

originally proposed in [4]. A second measure proposed in this paper is called 'accuracy per computational cost' efficiency (ACE). In the first part of the paper, the performance of the two measures is compared for a standard one-dimensional (1D) model problem with solution x^α, which may exhibit a singularity at the origin, depending on the value of the parameter α. The accuracy of the resulting grid is compared with the asymptotically optimal 'radical grid' [3, 5]. Modified versions of the efficiency strategies are proposed for the singular case, and the resulting adaptive methods are compared with a threshold-based refinement strategy. The efficiency strategies are also applied to the *hp*-refinement case for the 1D model problem, and the results are compared with the 'optimal geometric grid' for *hp*-refinement that was derived in [5]. In the last part of the paper, the use of the efficiency-based refinement strategies is explored for problems with spatial dimension $d > 1$. The 'work times error' strategy turns out to be inefficient when the spatial dimension, d, is larger than the finite element order, p, but the 'accuracy per computational cost' strategy provides an efficient refinement mechanism for any combination of d and p. This is illustrated for a model problem in two dimensions (2D).

This paper is organized as follows. In the next section, the efficiency based h-refinement strategies are described, along with the notation used in this paper, the model problem, and assumptions on the PDE problems, finite element methods, error estimators and linear solvers considered. The performance of the WEE and ACE refinement strategies for the 1D model problem are discussed in Section 3. Modified WEE and ACE refinement strategies for the singular case are considered in Section 4. In Section 5, efficiency-based *hp*-refinement strategies are discussed and illustrated for the 1D test problem. Sec. 6 describes how the efficiency-based refinement strategies can be applied for 2D problems. Conclusions are formulated in Section 7.

2. Efficiency-based h-refinement strategies

2.1. Assumptions on PDE problem, error estimate, refinement process, and linear solver

Consider a PDE written abstractly as

$$Lu = f \text{ in } \Omega \subset \mathcal{R}^d, \tag{1}$$

with appropriate boundary conditions and solution space V. Assume that continuity and coercivity bounds for the corresponding bilinear form can be verified in some suitable norm. Let $\mathcal{T}_h$ be a regular partition of the domain, Ω, into finite elements [3, 6], i.e., $\Omega = \bigcup_{\tau\in\mathcal{T}_h} \tau$ with $h = \max\{\text{diam}(\tau) : \tau \in \mathcal{T}_h\}$. In this paper we assume, for simplicity, that the elements are squares in 2D, and cubes in three dimensions (3D). Let V_h be a finite-dimensional subspace of V and $u_h \in V_h$ a finite element approximation such that the following error estimate holds:

$$||u - u_h||_{H^m(\Omega)} \leq Ch^{s-m}||u||_{H^s(\Omega)}, \tag{2}$$

where $0 \leq m < s < p+1$, with m and p integers and s a real number. Here, p is the polynomial order of the finite element method. Furthermore, assume that we obtain a sharp a-posteriori error estimate $E(u_h, f)$ that is equivalent to $||u - u_h||_{H^m(\Omega)}$. The associated error functional is given by $\mathcal{F}(u_h, f) = E^2(u_h, f)$. For example, the L^2 functional is a natural a-posteriori error estimate for First-Order System Least Squares (FOSLS) finite element methods, and equivalence to the H^1 norm has been proved for several relevant second-order PDE systems of elliptic type [4, 7, 8, 9]. The local value of the error, E, on element τ_j is denoted by ϵ_j.

Copyright © 2007 John Wiley & Sons, Ltd.

Prepared using nlaauth.cls

Consider an adaptive hp refinement process of the following form. The refinement process starts on a coarse grid with uniform element size h and order $p = 1$ (level 0), and proceeds through levels $\ell = 1, 2, \ldots$ until the error measure, $E_\ell(u_h, f)$, has value less than a given bound. In each step, some elements may be refined in h by splitting them into 2^d sub-elements, and some elements may be refined in p by doubling the element order. The decision of which elements to refine is based on the information provided by the local error estimator, and by heuristics that may take into account predicted error reduction and work. In particular, we consider strategies where the elements are ordered in terms of decreasing local error, such that elements with larger error are considered for refinement first. Standard threshold-based approaches then may refine, for example, a fixed fraction of the elements in every step, or a fixed fraction of the total error functional. Let the work needed to solve the discrete linear system on level ℓ be given by W_ℓ. Our goal is to reach a pre-specified bound on the global error, $E_\ell(u_h, f)$, with a minimal amount of total work, $\sum_{\ell=1}^{L} W_\ell$. Finding this optimal grid sequence may be difficult, even if we restrict the process to h-refinement alone. Hence, we turn to seeking nearly optimal solutions by using heuristics of greedy type. We consider refinement heuristics that determine the fraction of elements to be refined based on optimizing an efficiency measure in every step. We expect that a desirable grid sequence needs to be a high accuracy sequence, *i.e.*, a grid sequence for which the error, $E_\ell(N_\ell)$, decreases with nearly optimal order as a function of the number of DOF, N_ℓ, on grid level ℓ.

We allow the domain to contain singularities, *i.e.*, points in whose neighborhood the full convergence order of the finite element method cannot be attained due to lack of smoothness of the solution. For simplicity, assume that those singular points can only be located at coarse-level grid points, and that their power and location are known. If this information is not known in advance, the location and strength of singularities can be estimated by monitoring reduction rates of local error functionals during a few steps of initial uniform refinement.

It is assumed that optimal solvers, *e.g.*, multigrid, are used for the discrete linear systems. The computational work for solving these systems is, thus, assumed to be a fixed constant times the number of DOF: $W_\ell = c\, N_\ell$.

2.2. 1D model problem and finite element method

In the first part of this paper, we study the performance of the proposed efficiency based refinement strategies for a standard model problem in 1D [3, 5]:

$$u'' = \alpha(\alpha - 1)x^{\alpha-2}, u(0) = 0, u(1) = 1, \tag{3}$$

with exact solution given by $u = x^\alpha$. While the efficiency-based refinement strategies can be applied to various types of finite element methods and associated error estimates, we choose to illustrate the refinement strategies for model problem (3) using standard Galerkin finite element methods of order p, with the error estimated by the H^1 seminorm of the actual error, $e = u - u_h$, i.e., $\mathcal{F}(u_h, f) = ||u' - u'_h||^2_{L^2(\Omega)}$ and $\epsilon_j^2 = ||u' - u'_h||^2_{L^2(\tau_j)}$. These are equivalent to the H^1 norm, since it turns out that $e(x_i) = 0$ at each grid point for our model problem [3, 5]. Note that $u \in H^{1+\alpha-\frac{1}{2}-\epsilon}((0,1))$, for any $\epsilon > 0$. If we choose $\frac{1}{2} < \alpha \leq \frac{3}{2}$ such that $u \in H^1((0,1))$ but $u \notin H^2((0,1))$, then there is an x^α-type singularity at $x = 0$. We choose this model problem and this error estimator because asymptotically optimal h and hp finite element grids have been developed for them [3, 5], which can be used as a point of comparison for the refinement strategies to be presented in this paper. In addition, it turns out that the

Copyright © 2007 John Wiley & Sons, Ltd.

Prepared using nlaauth.cls

finite element approximations can be obtained easily, namely, by interpolation for $p = 1$, and by integrating a truncated Legendre expansion of $u'(x)$ for $p \geq 1$. The refinement strategies presented in this paper can be equally applied to other finite element methods, as is illustrated in the second part of the paper, where we present results for a 2D problem using the FOSLS finite element method [8, 4].

2.3. 'Work times error' and 'accuracy per computational cost' efficiency strategies

On each level, order the elements such that the local error, ϵ_j, satisfies $\epsilon_1 \geq ... \geq \epsilon_{N_\ell}$. With $r \in (0, 1]$ denoting the to-be-determined fraction of elements that will be refined, let $f(r) \in [0, 1]$ be the fraction of the total error functional in the refinement region, $\gamma(r) \in [0, 1]$ the predicted functional reduction, and $\eta(r) \in [1, 2]$ the ratio of the number of DOF on level $\ell+1$ and level ℓ, *i.e.*, $N_{\ell+1} = \eta(r)\, N_\ell$. The first refinement strategy, 'work times error' efficiency (WEE), was initially proposed in [4]. Here, the fraction, r, of elements to be refined on the current level is determined by minimizing the following efficiency measure:

$$\text{work} \times \text{error reduction} = \eta(r)\, \sqrt{\gamma(r)}, \tag{4}$$

i.e.,

$$r_{\text{opt}} = \arg \min_{r \in (0,1]} \eta(r) \sqrt{\gamma(r)}. \tag{5}$$

The motivation for this heuristic is as follows: more work on the current level is justified when it results in increased error reduction that offsets the extra work. While this choice does not guarantee that a globally optimal grid sequence is obtained, this local optimization in each step results in an overall strategy of greedy type, which can be expected to lead to a reasonable approximation to the optimal grid sequence.

We also propose a second strategy, 'accuracy per computational cost' efficiency (ACE). We define the predicted effective functional reduction factor

$$\gamma(r)^{\text{eff}} = \gamma(r)^{1/\eta(r)}. \tag{6}$$

The fraction, r, of elements to be refined on the current level is determined by minimizing this effective reduction factor, which is the same as minimizing $\log(\gamma(r)^{\text{eff}})$, *i.e.*,

$$r_{\text{opt}} = \arg \min_{r \in (0,1]} \frac{\log(\gamma(r))}{\eta(r)}. \tag{7}$$

The effective functional reduction factor, $\gamma(r)^{\text{eff}}$, measures the functional reduction per unit work. Indeed, compare two hypothetical error-reducing processes with functional reduction factors γ_1 and γ_2, and work proportional to η_1 and η_2. Assume that process 2 requires double the work of process 1, $\eta_2 = 2\,\eta_1$. Then the two processes would be equally effective when $\gamma_2 = \gamma_1^2$, because process 1 could be applied twice to obtain the same error reduction as process 2, using the same total amount of work as process 2. Minimizing the effective functional reduction in every step, thus, chooses the fraction, r, of elements to be refined by locally minimizing the functional reduction per unit work.

Both the strategies of minimizing work times error reduction, and minimizing the effective functional reduction factor, are ways for optimizing the efficiency of the refinement process at each level. Hence, we call the two proposed refinement strategies efficiency-based.

Copyright © 2007 John Wiley & Sons, Ltd.

Prepared using nlaauth.cls

2.4. Error and work estimates for h-refinement in 1D

The predicted functional reduction ratio, $\gamma(r)$, and element growth ratio, $\eta(r)$, can be determined as follows for the case of h-refinement in 1D with fixed finite element order p.

The element growth ratio, $\eta(r)$, can be determined easily. We have N_ℓ elements on level ℓ. Of these, rN_ℓ are refined into two new elements each, while $(1-r)N_\ell$ elements are left unrefined. Thus, the number of elements on level $\ell+1$ is $N_{\ell+1} = (1-r)N_\ell + 2rN_\ell = (1+r)N_\ell$. This yields

$$\eta(r) = 1 + r. \tag{8}$$

The predicted functional reduction factor, $\gamma(r)$, depends on the error estimate and the smoothness of the solution. As mentioned above, we consider the case that the error estimate is equivalent to the H^1 norm of $u-u_h$, i.e., $\mathcal{F}(u_h, f) \approx ||u-u_h||^2_{H^1(\Omega)}$ and $\epsilon_j^2 \approx ||u-u_h||^2_{H^1(\tau_j)}$. The error has the following asymptotic behavior [6].

For elements τ_j in which the solution is smooth (at least in $H^{p+1}(\tau_j)$ if order p elements are used), we have

$$\begin{aligned} \epsilon_j^2 &\approx ||u_h - u||^2_{H^1(\tau_j)} \\ &\leq C h_j^{2p} ||u||^2_{H^{p+1}(\tau_j)} \\ &\leq C M_{p+1} h_j^{2p} h_j. \end{aligned} \tag{9}$$

Here, we can take $M_{p+1} = \| \sum_{i=0}^{p+1} u^{(i)2} \|_{\infty,\tau_j}$, such that $||u||^2_{H^{p+1}(\tau_j)} \leq M_{p+1}\, h_j$. If τ_j is split into two equal parts, we have two new elements, $\tau_{j,1}$ and $\tau_{j,2}$, and we can assume that

$$\frac{\epsilon_{j,1}^2 + \epsilon_{j,2}^2}{\epsilon_j^2} \approx \left(\frac{1}{2}\right)^{2p}. \tag{10}$$

However, if u is less smooth in some element τ_j, i.e., if we can only assume that $u \in H^s(\tau_j)$ with $s \in \mathcal{R},\ s < p+1$, then we have

$$\epsilon_j^2 \leq C h_j^{2(s-1)} ||u||^2_{s,\tau_i}. \tag{11}$$

For simplicity, we only consider the highly singular case here, for which $s << p+1$. If, again, τ_j is split in two, assuming element $\tau_{j,1}$ contains the singularity, then $\epsilon_{j,1} >> \epsilon_{j,2}$ and $||u||_{s,\tau_{j,1}} \approx ||u||_{s,\tau_j}$. We then obtain

$$\frac{\epsilon_{j,1}^2 + \epsilon_{j,2}^2}{\epsilon_i^2} \approx \frac{\epsilon_{j,1}^2}{\epsilon_j^2} \approx \left(\frac{1}{2}\right)^{2(s-1)}. \tag{12}$$

Suppose the solution is sufficiently smooth in the whole domain. Then the predicted functional reduction factor, $\gamma(r)$, can be obtained as follows. We apply (10) to the elements that are refined. A fraction $1 - f(r)$ of elements do not get refined, and so we assume that their errors are not reduced. This results in

$$\gamma(r) = 1 - f(r) + \left(\frac{1}{2}\right)^{2p} f(r). \tag{13}$$

It is cumbersome to give a general expression for the singular case. However, assuming that we know the power and location of the singularities in advance, one can easily compute $\gamma(r)$ using (10) and (12).

Copyright © 2007 John Wiley & Sons, Ltd.

Prepared using nlaauth.cls

3. Performance of the WEE and ACE h-refinement strategies in 1D

3.1. Performance of WEE and ACE for smooth solutions

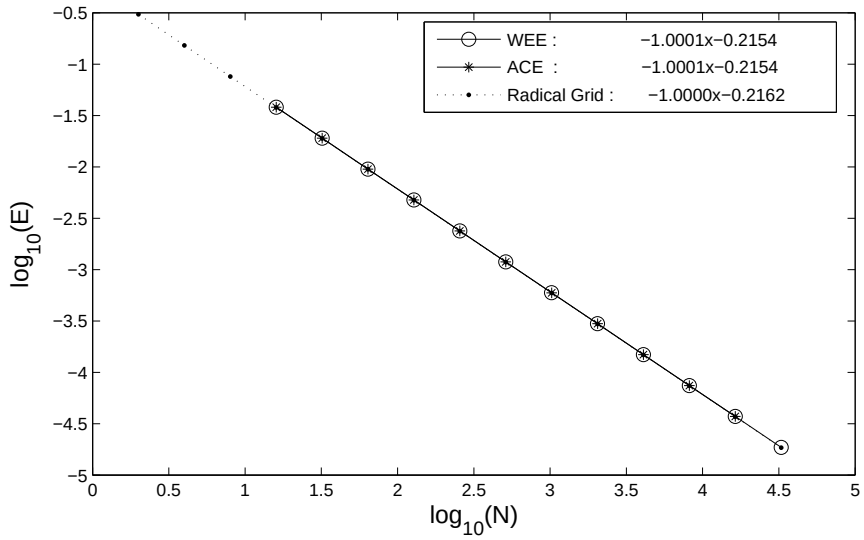
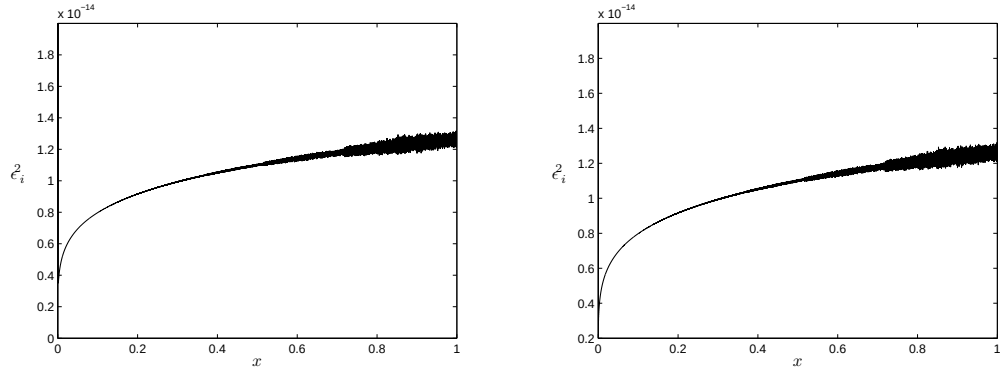


Figure 1. Error versus DOF, $\alpha = 2.1$ (no singularity), $p = 1$.



(a) WEE: $N_L = 32,741$, $E_L = 1.859e-5$, $L = 18$, total work $= 102,313$

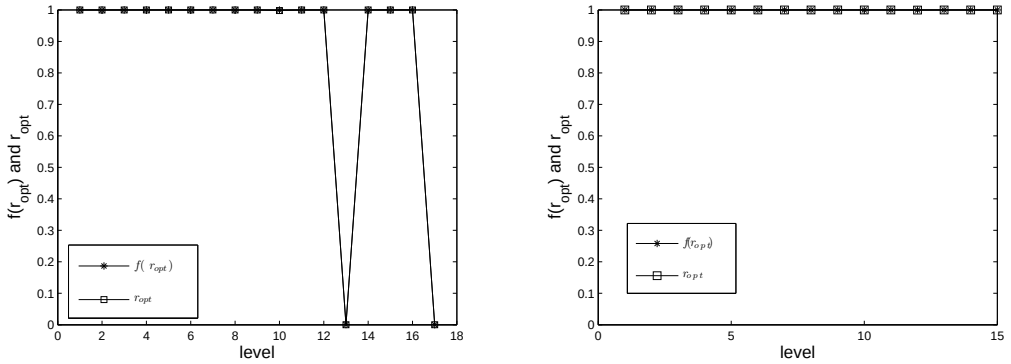
(b) ACE: $N_L = 32,760$, $E_L = 1.858e-5$, $L = 16$, total work $= 65,520$

Figure 2. Local error functional, ϵ_i^2, versus grid location on the final grid, $\alpha = 2.1$ (no singularity), $p = 1$.

We apply the WEE and the ACE strategies to our 1D model problem (3) with $p = 1$. On each level ℓ, each element is allowed to be refined at most once. We first consider the

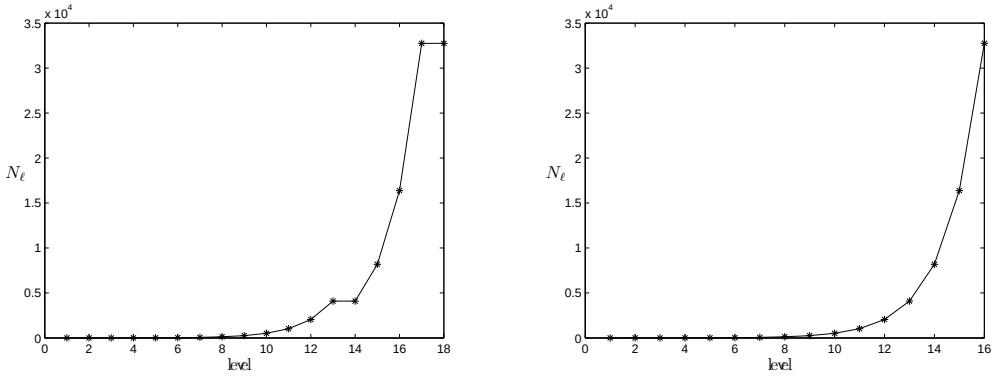
Copyright © 2007 John Wiley & Sons, Ltd.

Prepared using nlaauth.cls



(a) WEE (b) ACE

Figure 3. Refined fraction of error functional, $f(r_{\text{opt}})$, versus level, ℓ, and refined fraction of elements, r_{opt}, versus level, ℓ, $\alpha = 2.1$ (no singularity), $p = 1$.



(a) WEE (b) ACE

Figure 4. Number of elements, N_ℓ, versus level, ℓ, $\alpha = 2.1$ (no singularity), $p = 1$.

nonsingular case and choose $\alpha > 3/2$ such that $u \in H^2((0,1))$. It follows that the predicted functional reduction factor, $\gamma(r)$, is given by

$$\gamma(r) = 1 - \frac{3}{4} f(r). \tag{14}$$

Note that, for a given error bound, our ultimate goal is to choose a grid sequence that minimizes the total work, $\sum_{\ell=1}^{L} \mathcal{W}_\ell$, which is the same as minimizing $\sum_{\ell=1}^{L} N_\ell$, based on our assumption that the work is proportional to N_ℓ. For a given error bound, the number of elements on final grid N_L is determined by the convergence rate of the global error w.r.t. the DOF, which in fact is determined by the refinement strategy. For our model problem, it has been shown in [5] that the rate of convergence is never better than $(Np)^{-p}$, where N is the number of elements and p is the degree of the polynomial.

Copyright © 2007 John Wiley & Sons, Ltd.

Prepared using nlaauth.cls

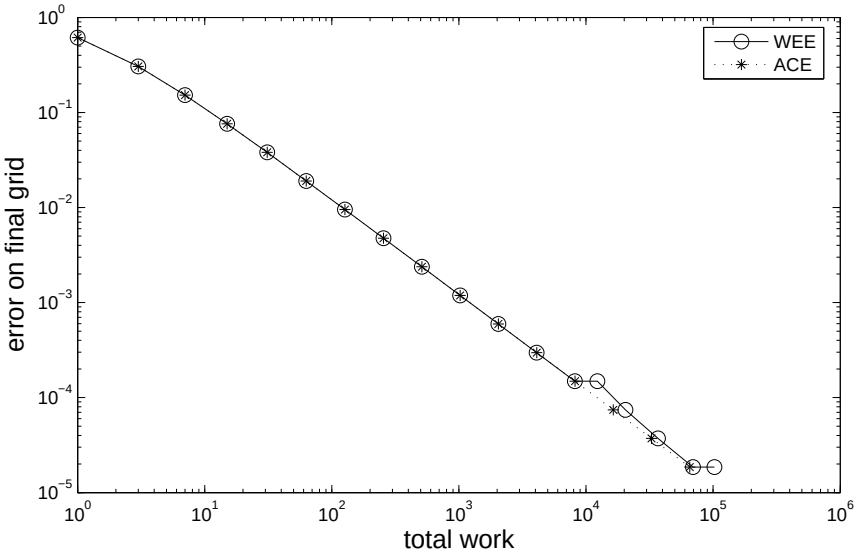
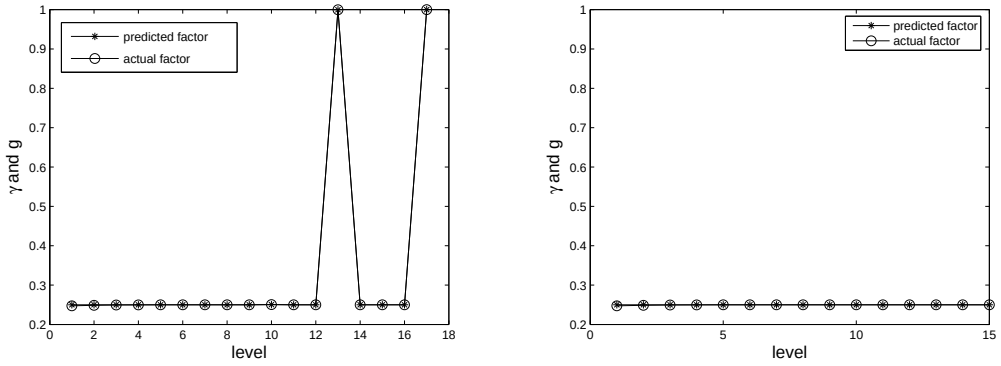


Figure 5. Final error, E_L, versus total work, $\sum_{\ell=1}^{L} N_\ell$, $\alpha = 2.1$ (no singularity), $p = 1$.



(a) WEE (b) ACE

Figure 6. Predicted functional reduction factor, $\gamma(r_{opt})$, and actual functional reduction factor, g, versus level, ℓ, $\alpha = 2.1$ (no singularity), $p = 1$.

THEOREM 1. *[5] Let* $E = (\sum \epsilon_i^2)^{\frac{1}{2}}$. *Then there is a constant,* $C = C(\alpha, p) > 0$, *for any grid* $\{0 = x_0 < x_1 < ... < x_N = 1\}$, *such that*

$$E \geq C\,(Np)^{-p}. \tag{15}$$

For our example problem, an asymptotically optimal final grid, called a radical grid, is described in [3, 5]:

$$x_j = (j/N)^{(p+1/2)/(\alpha-1/2)},\ j = 0, ..., N. \tag{16}$$

Copyright © 2007 John Wiley & Sons, Ltd.

Prepared using nlaauth.cls

This grid is optimal in the sense that, in the limit of large N, it results in the smallest error as a function of the number of DOF. If the WEE or the ACE strategy results in a grid sequence with approximately optimal convergence rate of the global error w.r.t. DOF, then the number of elements on the final grid must be close to the optimal number of elements, which only depends on the given error bound. Because we want to minimize work, it follows that, among the methods with approximately optimal convergence rate, the methods for which the sequence $\{N_\ell\}$ increases fast are preferable. Large refinements are, thus, advantageous.

We compare the numerical results of the WEE strategy and the ACE strategy, and the radical grid, for $\alpha = 2.1$ and $p = 1$ in Fig. 1 to Fig. 6. In the numerical results, we carry out the refinement process until $E_L(u_h, f) \leq 2e{-}5$ on final grid level L.

From Fig. 1, it can be observed that both strategies result in a highly accurate grid sequence. Thus, for a given error bound, the difference in the number of elements on the final grid is very small. This can be verified on Fig. 2. Fig. 3 and Fig. 4 show that the ACE strategy is slightly more efficient than the WEE strategy for our model problem in the smooth case. There are two small refinements in the WEE refinement process, while there are no small refinements for the ACE strategy. It follows that for a given error bound on the final grid, the WEE strategy may require slightly more total work than the ACE strategy, see Fig. 5. Fig. 2 shows that, for both strategies, the local errors in all elements tend to be equally distributed. This explains why the values of $f(r_{opt})$ and r_{opt} are close in Fig. 3. From Fig. 6 one can see that the predicted reduction factor $\gamma(r_{opt})$ is very accurate. This suggests a modification of the refinement process that can be considered to increase performance: one does not need to solve the linear systems until the new level is refined enough to have a significant number of additional elements in it. In this way complexity is never a problem, and we can still have a highly accurate grid sequence.

3.2. Performance of WEE and ACE for singular solutions

Next, we consider a singular example: let $\alpha = 0.6$, so that $u \in H^{1.1}((0,1))$. In the numerical results, we carry out the refinement process until $E_L(u_h, f) \leq 7e{-}4$ on final grid level L. For $p = 1$, the error reduction in the element that contains $x = 0$ can be approximately given by $\left(\frac{1}{2}\right)^{0.2}$, see (12). The predicted reduction factor $\gamma(r)$ is given by

$$\gamma(r) = 1 - \frac{3}{4} f(r) + \left[\left(\frac{1}{2}\right)^{0.2} - \frac{1}{4} \right] f(\frac{1}{N_\ell}). \tag{17}$$

Here, we assume that the local error in the element that contains $x = 0$ is always the largest.

The numerical results in Fig. 7 to Fig. 12 show that the two refinement strategies fail for this singular case. Fig. 7 shows that the WEE strategy results in a highly accurate grid sequence, while the ACE strategy becomes inaccurate by comparison with the radical grid. For both strategies, the local error in the first element, which contains the singularity, is always the largest, see Fig. 8. Hence, it is refined by the WEE and the ACE in every step. This also confirms that the predicted reduction factor can be given by (17). The WEE strategy generates a grid sequence with local errors being nearly equally distributed, but the ACE strategy does not: more than 90% of the global error accumulates in only 10% of the elements; see Fig. 8 and Fig. 9. Most refinement steps of the WEE strategy are small refinements: only the first element (possibly with a few other elements) is continuously being refined (see Figs. 9 and 10). This implies that the number of elements increases slowly as a function of refinement level. It

Copyright © 2007 John Wiley & Sons, Ltd.

Prepared using nlaauth.cls

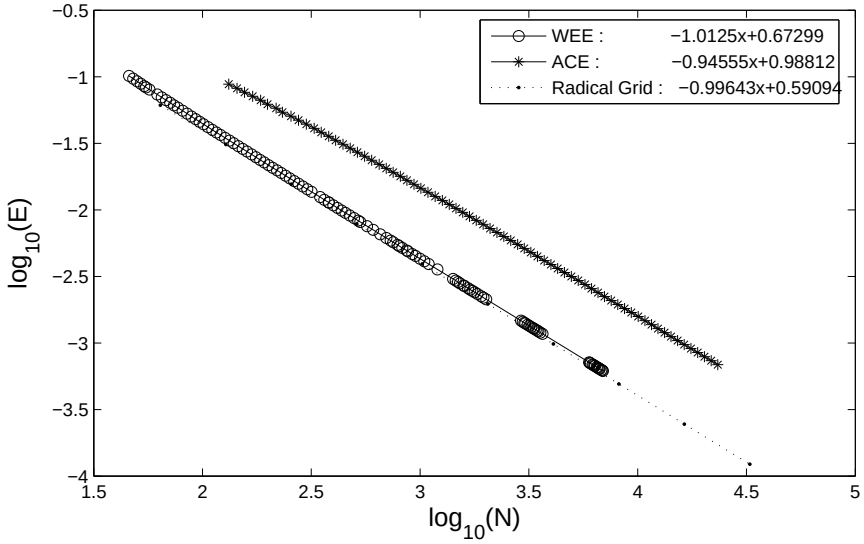
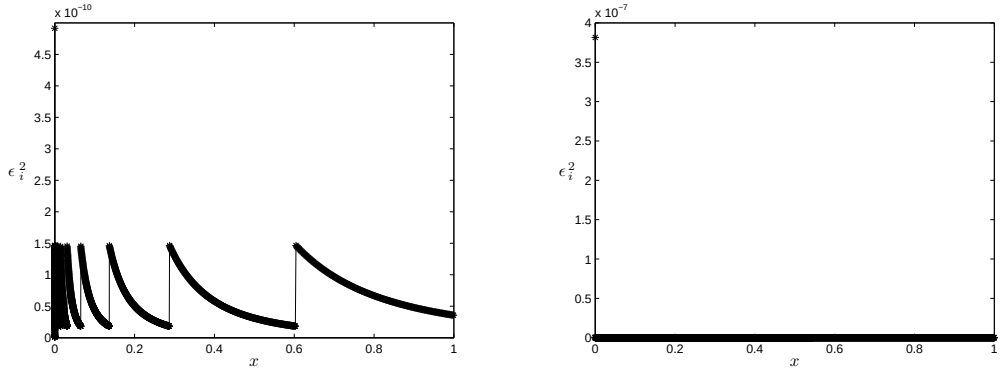


Figure 7. Error versus DOF, $\alpha = 0.6$ (singular case), $p = 1$.



(a) WEE: $N_L = 6,925$, $E_L = 6.169e-4$, $L = 154$, total work $= 192,775$

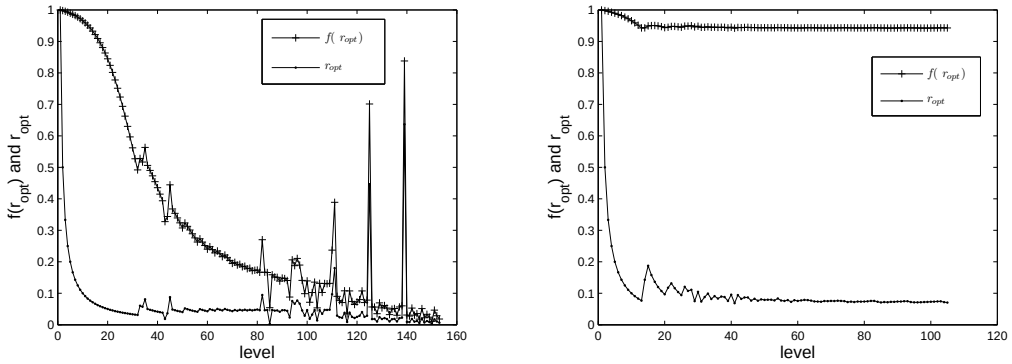
(b) ACE: $N_L = 24,986$, $E_L = 6.411e-4$, $L = 106$, total work $= 365,420$

Figure 8. Local error functional, ϵ_i^2, versus grid location on the final grid, $\alpha = 0.6$ (singular case), $p = 1$.

follows that the total work is very large. The ACE strategy does choose a refinement region with large fraction of the error in it. However, this large fraction of error is only contained in a few elements. As a result, only a small fraction of elements are refined. Thus, the required total work is still large; see Figs. 10 and 11. Compared to the nonsingular case (Fig. 5), the slope of the error versus total work plot in Fig. 11 is much less steep, especially in the initial phase of the refinement process. The predicted reduction factors for both strategies are accurate,

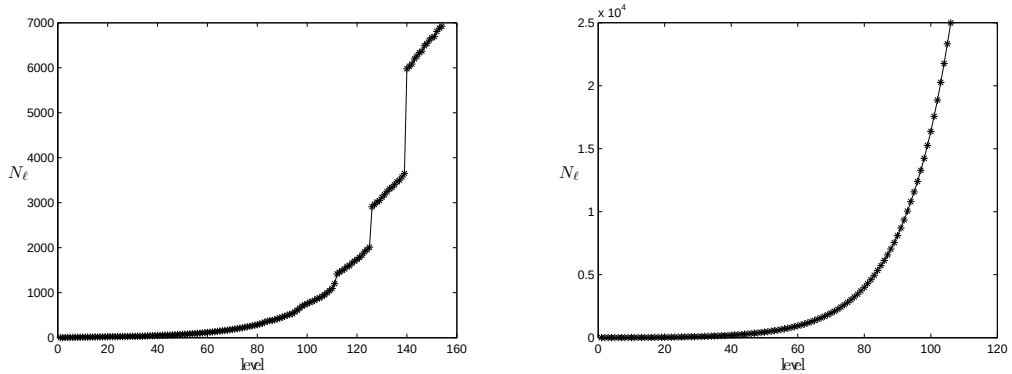
Copyright © 2007 John Wiley & Sons, Ltd.

Prepared using nlaauth.cls



(a) WEE (b) ACE

Figure 9. Refined fraction of error functional, $f(r_{\text{opt}})$, versus level, ℓ, and refined fraction of elements, r_{opt}, versus level ℓ, $\alpha = 0.6$ (singular case), $p = 1$.



(a) WEE (b) ACE

Figure 10. Number of elements, N_ℓ, versus level, ℓ, $\alpha = 0.6$ (singular case), $p = 1$.

see Fig. 12. This suggests that we can make the same modification as for the smooth case to increase performance: one can wait on solving the linear systems until the number of elements has increased sufficiently. In this way, one can assure that complexity is never a problem, but calculating and minimizing the WEE and ACE functions many times may be costly as well. In conclusion, for the highly singular case, the WEE strategy results in an accurate grid sequence but is not efficient due to too many small refinements; the ACE strategy is worse than the WEE strategy in this case, because the grid sequence is not accurate and many small refinements are performed.

Copyright © 2007 John Wiley & Sons, Ltd.

Prepared using nlaauth.cls

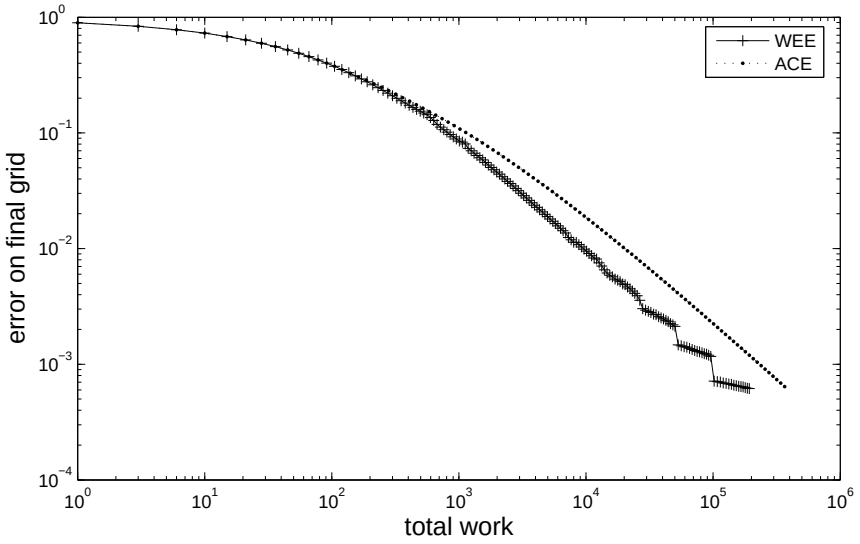
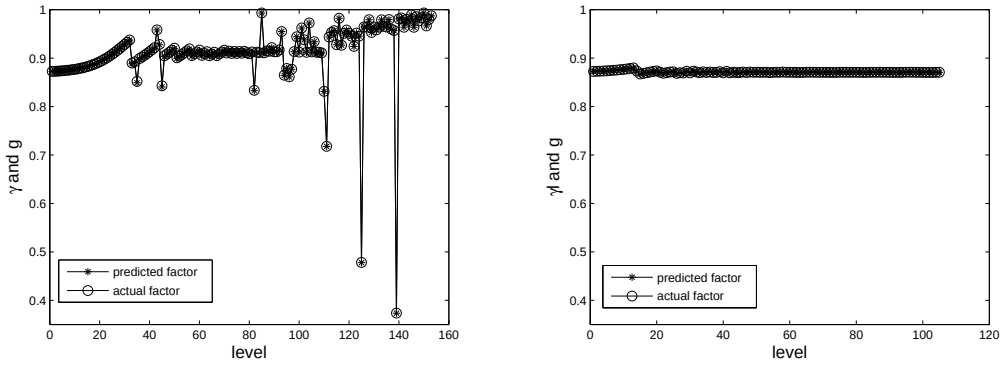


Figure 11. Final error, E_L, versus total work, $\sum_{\ell=1}^{L} N_\ell$, $\alpha = 0.6$ (singular case), $p = 1$.



(a) WEE (b) ACE

Figure 12. Predicted functional reduction factor, $\gamma_\ell(r_{opt})$, and actual functional reduction factor, g_ℓ, versus level, ℓ, $\alpha = 0.6$ (singular case), $p = 1$.

4. Modified WEE and ACE h-refinement strategies for singular solutions in 1D

4.1. Modified WEE and ACE h-refinement strategies

The inefficiency of the WEE and ACE strategies for the highly singular solution is due to many steps of small refinement for the singular elements. Therefore, we attempt to avoid these steps by using a geometrically graded grid starting from the singular point, with the aim of

Copyright © 2007 John Wiley & Sons, Ltd.

Prepared using nlaauth.cls

saving work while attempting to keep the grid sequence accurate.

As was discussed before, we assume that singularities can only be located at coarse-level grid points, and that we know the location and the power of the singularities in advance. We propose to do graded-grid refinement for elements containing a singularity, in such a way that we obtain the same error reduction factor as in elements in which the solution is smooth. For example, for a singularity located at a domain boundary, the element at the boundary is split in two, and then, within the same refinement step, the new element at the singularity is repeatedly split in two again, until the predicted error reduction factor matches the desired error reduction. We modify the predicted functional reduction factor, $\gamma(r)$, and the work increase ratio, $\eta(r)$, accordingly. We expect the correspondingly modified WEE and ACE strategies (MWEE and MACE) to generate a highly accurate grid sequence in an efficient way. This results in the following modified efficiency-based refinement strategies:

1) Order the elements such that the local error, ϵ_j, satisfies $\epsilon_1 \geq \epsilon_2 \geq ... \geq \epsilon_{N_\ell}$.

2) Perform graded grid refinement for elements containing a singularity, i.e., if $u \in H^{s_j}(\tau_j)$, then graded grid refinement with m_j levels is used for any τ_j that needs to be refined, with m_j satisfying

$$\left(\frac{1}{2}\right)^{2m_j(s_j-1)} \approx \left(\frac{1}{2}\right)^{2p} \Rightarrow m_j = \lceil \frac{p}{s_j - 1} \rceil.$$

Note that we assume here that the error in the first, singular new element dominates the sum of the errors in the other new elements of the graded grid. This is a good approximation for a strong singularity. For elements in which the solution is smooth, single refinement is performed: $m_j = 1$. Let k_j be the number of new elements after τ_j is refined: $k_j = m_j + 1$.

3) The predicted functional reduction factor, $\gamma(r)$, and the work increase ratio, $\eta(r)$, are given by

$$\begin{aligned} \eta(r) &= 1 - r + \frac{\sum_{j \leq rN_\ell} k_j}{N_\ell}, \\ \gamma(r) &= 1 - f(r) + \left(\frac{1}{2}\right)^{2p} f(r). \end{aligned} \tag{18}$$

4) Find the optimal r defined in (5) for the MWEE strategy, and in (7) for the MACE strategy.

5) Repeat.

4.2. Peformance of the modified WEE and ACE h-refinement strategies for singular solutions

We again choose $\alpha = 0.6$ and $p = 1$ for our example problem. There is a singularity at $x = 0$, with error reduction factor bound $(\frac{1}{2})^{0.2}$. Therefore, for the element that contains $x = 0$, we use $11-$graded refinement $(m = \lceil \frac{1}{0.1} \rceil)$. Numerical results are shown in Figs. 13-18.

By comparing the numerical results for the modified strategies with the results for the original methods, we see the following. Both the MWEE and MACE strategies results in highly accurate grid sequences: the convergence rate is very close to the optimal rate (Fig. 13). Local error functionals on the final MWEE grid are more equally distributed than for the MACE grid. For the MWEE strategy, the local error functional in the singular element is only 3 times larger than in the smooth elements. However, for the MACE strategy, that ratio is as large as $1,000$ (Fig. 14). For the MWEE strategy, the number of elements, N_ℓ, increases much

Copyright © 2007 John Wiley & Sons, Ltd.

Prepared using nlaauth.cls

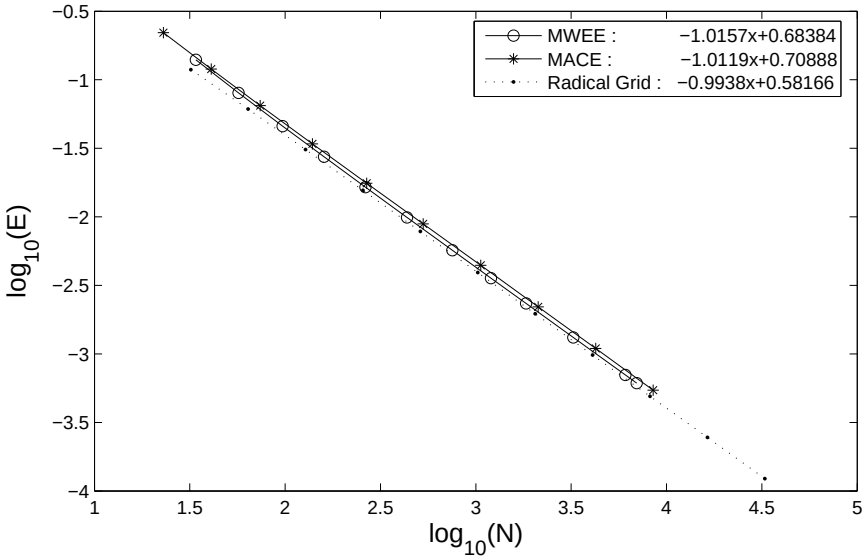
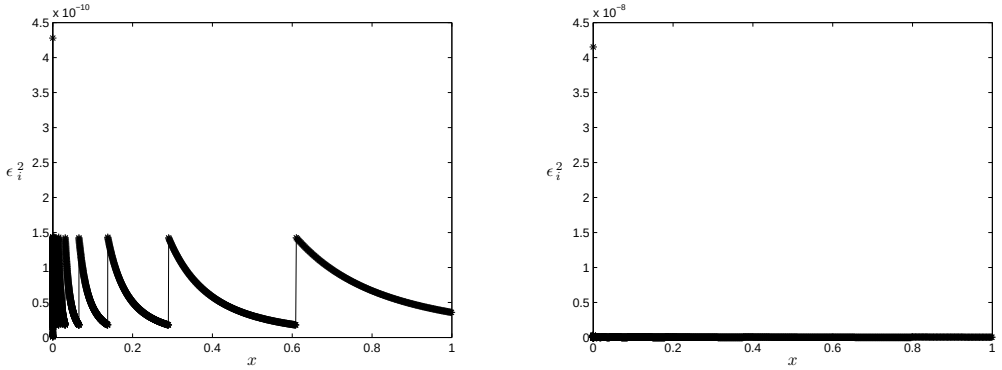


Figure 13. Error versus DOF, $\alpha = 0.6$ (singular case), $p = 1$.



(a) MWEE: $N_L = 6,975$, $E_L = 6.125e-4$, $L = 15$, total work $= 21,176$

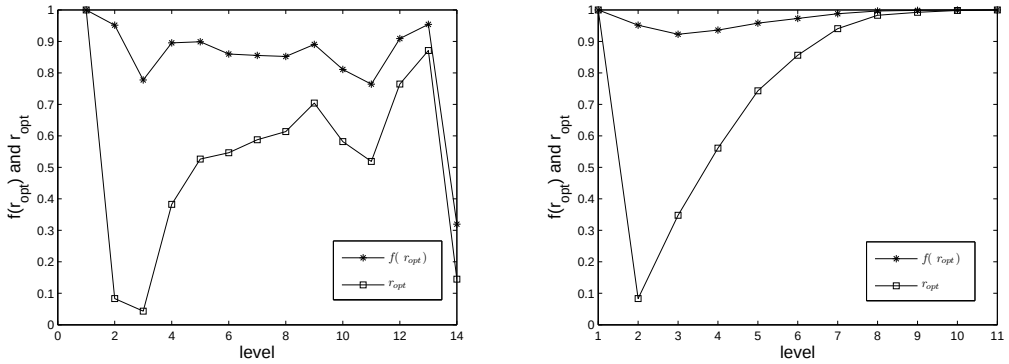
(b) MACE: $N_L = 8,517$, $E_L = 5.443e-4$, $L = 12$, total work $= 17,044$

Figure 14. Local error functional, ϵ_i^2, versus grid location on the final grid, $\alpha = 0.6$ (singular case), $p = 1$.

faster than for the WEE strategy, which reduces the work considerably (Fig. 15). However, there still exist a few small refinement steps. For the MACE strategy, it seems that the strategy tends to do uniform refinement after several initial steps (Fig. 15(b)). Similar to the smooth solution case, the MWEE strategy may need slightly more work to reach the same error bound than the MACE strategy due to a few steps of small refinement (Fig. 17). However, since the

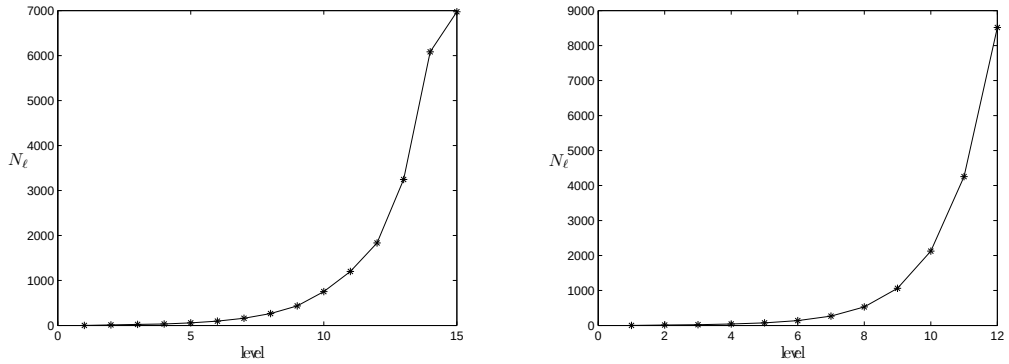
Copyright © 2007 John Wiley & Sons, Ltd.

Prepared using nlaauth.cls



(a) MWEE (b) MACE

Figure 15. Refined fraction of error functional, $f(r_{\rm opt})$, versus level, ℓ, and refined fraction of elements, $r_{\rm opt}$, versus level, ℓ, $\alpha = 0.6$ (singular case), $p = 1$.



(a) MWEE (b) MACE

Figure 16. Number of elements, N_ℓ, versus level, ℓ, $\alpha = 0.6$ (singular case), $p = 1$.

MWEE strategy is slightly more accurate, the difference is very small. Again, the predicted functional reduction factors are good approximations of the actual factors for both strategies (Fig. 18).

4.3. Comparison with threshold-based refinement strategy

It is instructive to compare the MWEE and the MACE strategies with the threshold-based refinement strategy that chooses to refine a fixed fraction of the error functional on each level, i.e., $f_\ell(r) \equiv \theta$. The same graded grid refinement strategy is used for the elements that contain a singularity. We find the following for our example problem.

If we choose to refine a fixed fraction of the global error that is too small (less than the average of $f(r_{\rm opt})$ in the modified efficiency-based strategies), *e.g.*, $\theta = 0.2$ in Fig. 19, then the resulting

Copyright © 2007 John Wiley & Sons, Ltd.

Prepared using nlaauth.cls

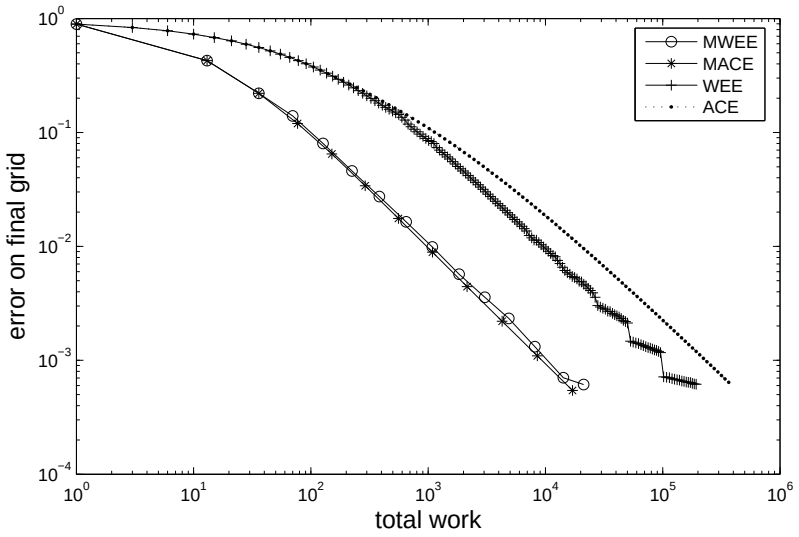
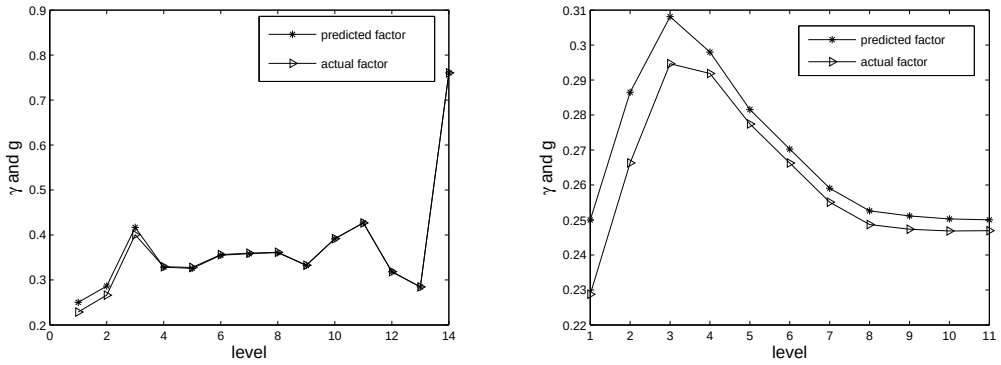


Figure 17. Final error, E_L, versus total work, $\sum_{\ell=1}^{L} N_\ell$, $\alpha = 0.6$ (singular case), $p = 1$.



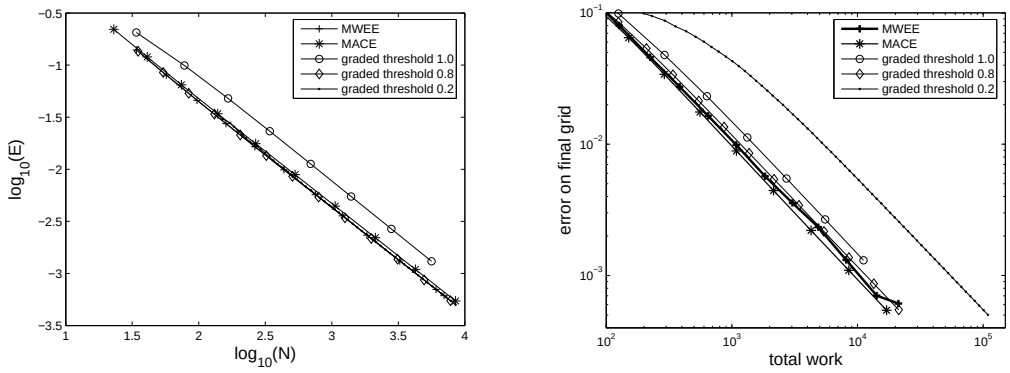
(a) MWEE (b) MACE

Figure 18. Predicted functional reduction factor, $\gamma(r_{opt})$, and actual functional reduction factor, g, versus level, ℓ, $\alpha = 0.6$ (singular case), $p = 1$.

grid sequence is almost of optimal accuracy, but the total work increases significantly since N_ℓ increases slowly. A threshold value that is too large (larger than the average of $f_\ell(r_{\text{opt}})$ in the modified efficiency-based strategies), *e.g.*, $\theta = 1.0$ in Fig. 19, makes the number of elements, $\{N_\ell\}_{\ell=1}^{L}$, increase faster, but the large threshold results in a less accurate grid sequence. This implies that more total work is required to reach the same error bound. A threshold value that is close to the average of $f(r_{\text{opt}})$ in the modified efficiency-based strategies, namely, $\theta = 0.8$ in Fig. 19, results in a refinement process that performs similar to the efficiency-based refinement

Copyright © 2007 John Wiley & Sons, Ltd.

Prepared using nlaauth.cls



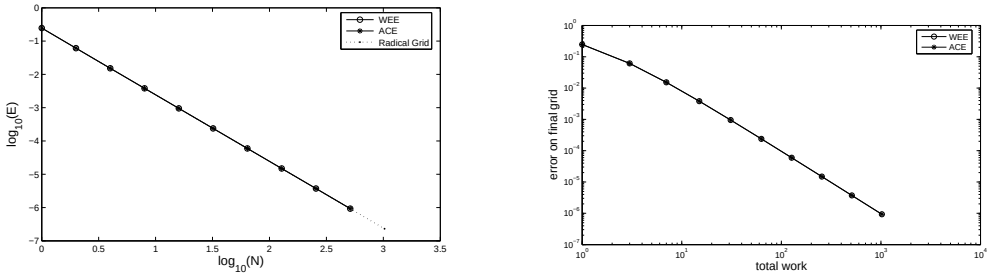
(a) (b)

Figure 19. Efficiency-based and threshold-based refinement strategies: (a) Error versus DOF. (b) Final error, E_L, versus total work, $\sum_{\ell=1}^{L} N_\ell$. (Both $\alpha = 0.6$ (singular case), $p = 1$.)

processes.

In conclusion, the efficiency-based refinement strategies automatically and adaptively choose a nearly optimal fraction of the error to be refined. As a result, they generate nearly optimal grid sequences in an efficient way, and there is no need to determine the optimal value of a threshold parameter.

4.4. Results for $p = 2$



(a) (b)

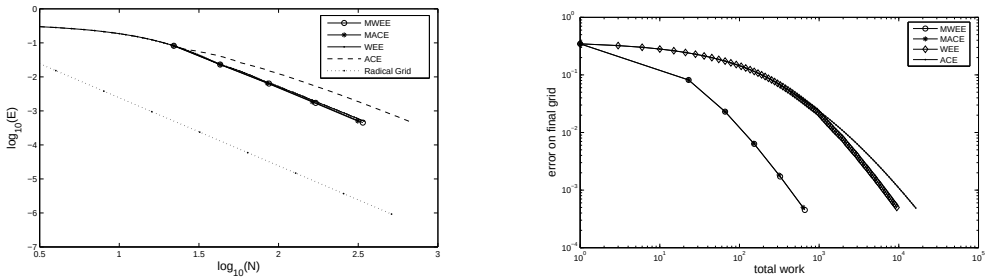
Figure 20. Efficiency-based refinement strategies for a smooth problem with $p = 2$ ($\alpha = 3.1$). (a) Error versus DOF. (b) Final error, E_L, versus total work, $\sum_{\ell=1}^{L} N_\ell$.

In this section, we briefly illustrate how the (M)WEE and (M)ACE strategies perform for finite element polynomial order $p = 2$.

First, consider a smooth case with $\alpha = 3.1$, such that $u \in H^3$ and $u \notin H^4$. Error versus DOF and total work are plotted for WEE and ACE in Fig. 20. Both strategies lead to global refinement in every step for this example, and produce a sequence of grids that are very close

Copyright © 2007 John Wiley & Sons, Ltd.

Prepared using nlaauth.cls



(a) (b)

Figure 21. Efficiency-based refinement strategies for a singular problem with $p = 2$ ($\alpha = 0.6$). (a) Error versus DOF. (b) Final error, E_L, versus total work, $\sum_{\ell=1}^{L} N_\ell$.

to optimal radical grids.

Fig. 21 shows results for a highly singular case, with $\alpha = 0.6$, such that $u \in H^1$ and $u \notin H^2$. WEE and ACE produce small refinements, but this is remedied by the MWEE and MACE strategies, resulting, as before, in much less work for the modified strategies. It has to be noted, however, that the MWEE and MACE grids contain many more elements than optimal graded grids. This is probably due to the fact that the singularity is very strong for $\alpha = 0.6$ and $p = 2$, such that a geometrically graded grid with a grading factor of $\frac{1}{2}$ does not decrease the grid size fast enough in the vicinity of the singularity. Nevertheless, we can conclude that, within the constraint of refinement based on splitting cells in two, the MWEE and MACE strategies lead to an efficient refinement process.

5. Efficiency-based *hp*-refinement strategies in 1D

Assuming that we know a good approximation for the *p*-refinement error reduction factor for each element, we can apply the efficiency-based refinement strategies to *hp*-refinement processes.

5.1. hp-version of the (M)WEE and (M)ACE refinement strategies

Consider an *hp*-finite element method for our simple example problem (3). Let $\mathcal{T}_h = \{0 = x_0 < x_1 < ... < x_N = 1\}$ be the grid and let $\mathbf{p} = \{p_1, p_2, ..., p_N\}$ be the degrees of the polynomials in the elements. Let u_h be the Galerkin finite element solution of (3) and $\epsilon_j^2(p_j) = ||u' - u'_s||^2_{0,\tau_j}$ the local error functional in element $\tau_j = [x_{j-1}, x_j]$ with polynomial of degree p_j. We choose local Legendre polynomials as the modal base functions [3]. Then we have the following theorem:

THEOREM 2. *[5] Let $\epsilon_j^2(p_j)$ be the local error of the finite element solution of problem* (12)*, and let*

$$\tau_j = [x_{j-1}, x_j],\ \theta_j = \frac{\sqrt{x_i} - \sqrt{x_{i-1}}}{\sqrt{x_i} + \sqrt{x_{i-1}}}.$$

Copyright © 2007 John Wiley & Sons, Ltd.

Prepared using nlaauth.cls

Then

$$\epsilon_1^2(p_1) \approx \frac{h_1^{2\alpha-1}}{p_1^{4\alpha-2}}. \tag{19}$$

If θ_j $(2 \le j \le N)$ *is not close to* 1, *then*

$$\epsilon_j^2(p_j) \approx \left\{ h_j^{\alpha-1/2} \left(\frac{1-\theta_j^2}{2\theta_j} \right)^{\alpha-1} \frac{\theta_j^{p_j}}{p_j^{\alpha}} \right\}^2. \tag{20}$$

We only consider h-refinement for the first element, which contains the singularity. Then we have the error functional reduction factor bound $(\frac{1}{2})^{2\alpha-1}$ as in (12). For an element τ_j that does not contain the singularity, note that θ_j is small, and again we obtain the same h-reduction factor bound, $(\frac{1}{2})^{2p_j}$, as before (see (10)). Moreover, if we double the degree of polynomial p_j, we obtain the p-reduction factor bound as follows:

$$\frac{\epsilon_j(2p_j)}{\epsilon_j(p_j)} \approx \left(\frac{\theta_j^{p_j}}{2^{\alpha}} \right)^2. \tag{21}$$

We can then develop an hp-version of the MWEE strategy as follows:

1) Order the elements such that the local error, ϵ_j, satisfies $\epsilon_1 \ge \epsilon_2 \ge ... \ge \epsilon_{N_\ell}$.

2) Let $p_{\max}$ be the maximal polynomial order to be used in the refinement process. Three types of refinement are used, depending on the element. We use a graded grid with $p = 1$ for the elements containing a singularity, in such a way that the predicted error-reduction factor attains $\frac{1}{4}$. (Note that a target reduction factor of up to $\frac{1}{2^{pmax}}$ could be used, but we choose $\frac{1}{4}$ for simplicity in our numerical tests.) For elements without a singularity, p-refinement (doubling p) is used if the solution is locally smooth enough (which, in general, can be detected *a posteriori* by comparing predicted and observed error-functional reduction ratios) and $p < p_{\max}$. Otherwise, $h-$refinement is used and the degree p is inherited by both sub-elements. As before, we assume that the work of solving the linear systems is proportional to the number of DOF. Then, doubling p or splitting the element into two elements with order p has the same computational complexity.

3) Calculate the MWEE or MACE efficiency functions and find the optimal fraction of elements to be refined, r_{opt}.

4) Refine elements τ_j, $1 \le j \le r_{opt} N_\ell$.

5) Repeat.

For a general problem different from (3), it may be difficult to find a sharp approximation formula for the error reduction in the case of p-refinement. Hence we are interested in seeking a more general but possibly less sharp p-error reduction factor. Recall that for elements τ_j in which the solution is smooth (at least in $H^{p_j+1}(\tau_j)$ if order p_j elements are used), we have

$$\epsilon_j^2(p_j) \le C(p_j) h_j^{2p_j} ||u||^2_{H^{p_j+1}(\tau_j)}.$$

More precisely, we have the following approximation [3]:

$$\epsilon_j^2(p_j) \le c \left(\frac{h_j}{2} \right)^{2p_j} \frac{1}{(2p_j)!} ||u||^2_{H^{p_j+1}(\tau_j)}. \tag{22}$$

Copyright © 2007 John Wiley & Sons, Ltd.

Prepared using nlaauth.cls

Assuming that $\frac{1}{(2p_j)!}||u||^2_{H^{p_j+1}(\tau_j)} \leq M$, where M is a constant, we obtain the following general p-error reduction factor

$$\frac{\epsilon_j^2(2p_j)}{\epsilon_j^2(p_j)} \approx \left(\frac{h_j}{2}\right)^{2p_j} \tag{23}$$

for elements τ_j that do not contain a singularity.

5.2. Optimal geometric hp-grid for the model problem

Just as in the case of h-refinement, we seek some kind of optimal grid for comparison. Suppose the locations of the grid points are given by

$$x_j = q^{N-j}, \quad 0 < q < 1, \ j = 1, 2, ..., N. \tag{24}$$

Let $\theta_j = \theta = \frac{1-\sqrt{q}}{1+\sqrt{q}}, \forall j : 1 \leq j \leq N$. It was shown in [5] that the optimal degree distribution of $\mathbf{p}$ for these grid locations tends to a linear distribution with slope

$$s_o = (\alpha - 1/2)\frac{\log q}{\log \theta}. \tag{25}$$

Furthermore, the optimal geometric grid factor q and linear slope s_o combination is given by

$$q_{opt} = (\sqrt{2} - 1)^2, \ s_{opt} = 2\alpha - 1. \tag{26}$$

5.3. Numerical results and comparisons

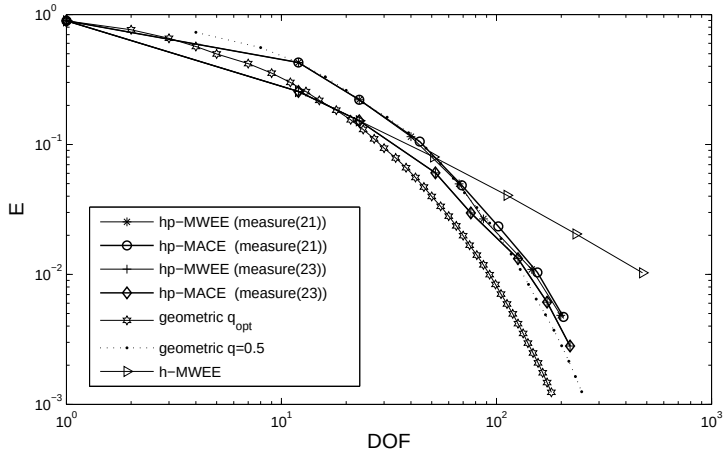


Figure 22. Error versus DOF, $\alpha = 0.6$ (singular case), $p = 1$.

We apply the *hp*-version MWEE and MACE strategies with the two $p-$refinement reduction factors given by (21) and (23) to our model problem 3 with $\alpha = 0.6$, and compare the numerical results with the optimal geometric grid with $q = q_{opt}$ and $q = \frac{1}{2}$; see Figs. 22 and 23. In the

Copyright © 2007 John Wiley & Sons, Ltd.

Prepared using nlaauth.cls

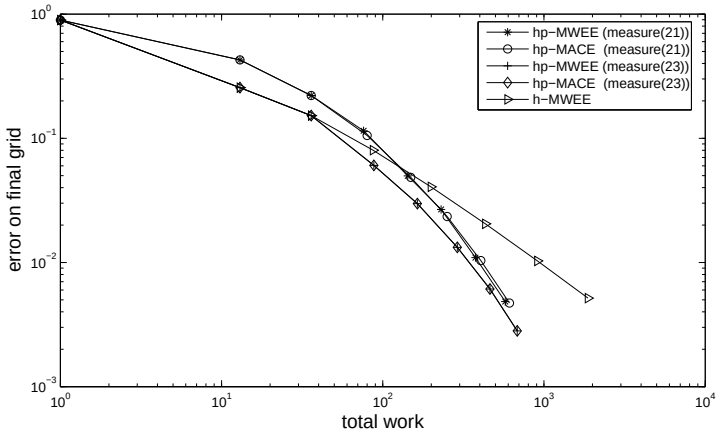


Figure 23. Final error, E_L, versus total work, $\sum_{\ell=1}^{L} N_\ell$, $\alpha = 0.6$ (singular case), $p = 1$.

numerical results, we carry out the refinement process until $E_L(u_h, f) \leq 5e{-}3$ on final grid level L.

Observe that the *hp*-finite element methods result in much faster error convergence rates than the *h*-finite element method. Both the *hp*-MACE and *hp*-MWEE strategies result in a highly accurate grid sequence with rate-of-error convergence very close to the geometrical grid with grading number $q = 0.5$. Also, the refinement process is efficient, *i.e.*, the number of DOF increases fast w.r.t. the refinement level. Surprisingly, $hp-$refinement strategies using the more general, but less accurate, error reduction factor (23), result in a better grid sequence than with the more accurate Babuška factor, (21). The results are even better than the optimal geometric grid sequence when the number of DOF is small. More work needs to be done to verify whether the general factor (23) works well for more general problems.

6. 2D results

In this section, we explore the use of the proposed efficiency-based refinement strategies in two spatial dimensions. In these initial considerations, we only discuss problems with sufficiently smooth solutions.

6.1. Efficiency strategies in $\mathcal{R}^d$

The efficiency-based WEE and ACE refinement strategies presented above can readily be applied to problems in d spatial dimensions. Let $\Omega \subset \mathcal{R}^d$. Assume again that the error estimator, $\mathcal{F}(u_h, f)$, is equivalent to the H^1 norm of $u - u_h$: $\mathcal{F}(u_h, f) \approx ||u - u_h||^2_{H^1(\Omega)}$. Assume that the refinement process, in each step, splits elements into 2^d sub-elements. Then the element growth ratio, $\eta(r)$, is given by

$$\eta(r) = 1 + (2^d - 1)\, r. \tag{27}$$

Copyright © 2007 John Wiley & Sons, Ltd.

Prepared using nlaauth.cls

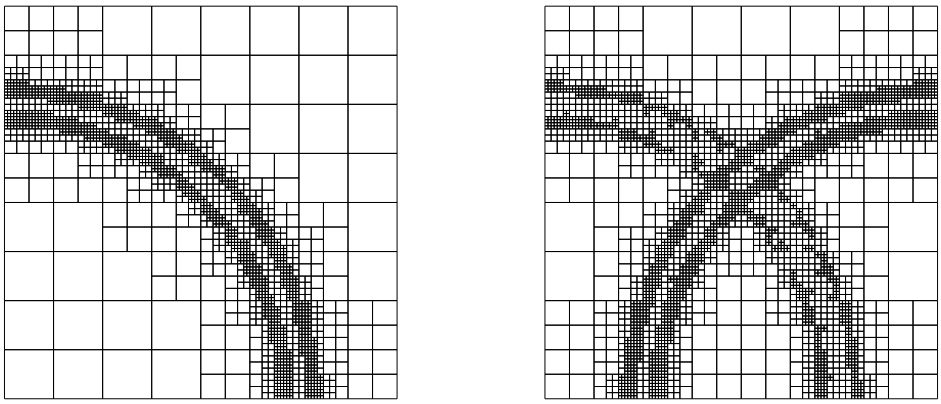
Suppose the solution is sufficiently smooth in the whole domain. As in the 1D case, the predicted functional reduction factor, $\gamma(r)$, is given by

$$\gamma(r) = 1 - f(r) + \left(\frac{1}{2}\right)^{2p} f(r). \tag{28}$$

The WEE and ACE strategies can then be used to determine the fraction of elements to be refined, r_{opt}, according to Eqs. (5) and (7), respectively.

It should be noted here that the WEE measure may be problematic in dimensions higher than one. This can be seen as follows. The WEE measure determines r_{opt} by minimizing $M_{WEE} \equiv \eta(r)\sqrt{\gamma(r)}$ over $r \in [1/N, 1]$. For smooth solutions, $\eta(1/N) \approx 1$ and $\gamma(1/N) \approx 1$, such that $M_{WEE}(1/N) \approx 1$. For $r = 1$, however, it can be observed that $\eta(1) = 2^d$ and $\gamma(1) = (\frac{1}{2})^{2p}$, such that $M_{WEE}(1) = 2^{d-p}$. This means that $M_{WEE} > 1$ when $d > p$. $M_{WEE}(r)$ is often a very smooth function, so r_{opt} is likely to be close to $1/N$ when $d > p$, resulting in small refinements, which are inefficient. We, thus, expect that the WEE strategy may not be efficient when $d > p$. We investigate this issue in the numerical results presented below. Also, it can be noted that this problem does not occur for the ACE strategy.

6.2. Model problem and finite element method



(a) (b)

Figure 24. Adaptively refined grids using the ACE refinement strategy for 2D problems with $p = 2$. (a) Single arc on a unit square domain. (b) Double arc on a unit square domain.

The following 2D finite element problem is considered to illustrate the efficiency-based refinement strategies. We solve the Poisson boundary value problem (BVP)

$$\begin{cases} -\Delta p = f \quad \text{in} \quad \Omega, \\ \quad p = g \quad \text{on} \quad \partial\Omega, \\ \quad \Omega = (0,1) \times (0,1), \end{cases} \tag{29}$$

with the right-hand side f and boundary conditions g chosen such that the solution is given

Copyright © 2007 John Wiley & Sons, Ltd.

Prepared using nlaauth.cls

by

$$p(r,\theta) = \begin{cases} 1 & r \le r_0, \\ h(r) & r_0 \le r \le r_1, \\ 0 & r_1 \le r. \end{cases} \tag{30}$$

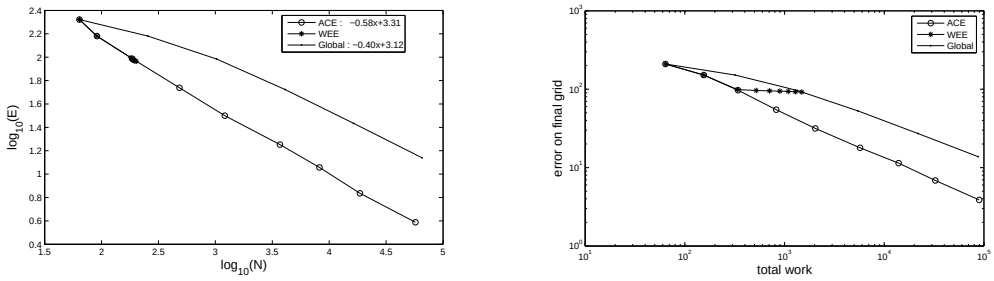
Here, (r,θ) are the usual polar coordinates and $h(r)$ is the unique polynomial of degree five such that $p \in C^2(\Omega)$. We choose $r_0 = 0.7$ and $r_1 = 0.8$. The solution of this test problem takes on the unit value in the lower left corner of the domain, and is zero elsewhere, except for a steep gradient in the thin strip $0.7 \le r \le 0.8$. Fig. 24(a) shows the grid obtained after several refinement steps for this model problem.

To illustrate the broad applicability of our refinement strategies, we solve this model problem using a FOSLS finite element method, rather than the Galerkin method that was used for the 1D test problems. BVP (29) is rewritten as a first-order system BVP [8]

$$\begin{cases} -\nabla \cdot U = f \quad \text{in} \quad \Omega, \\ \qquad U = \nabla p \\ \nabla \times U = 0 \\ \qquad p = g \quad \text{on} \quad \partial\Omega, \\ \vec{\tau} \cdot U = \dfrac{\partial g}{\partial \tau} \\ \qquad \Omega = (0,1) \times (0,1), \end{cases} \tag{31}$$

where U is a vector of auxiliary unknowns, and $\vec{\tau}$ is the unit vector tangent to $\partial\Omega$. The FOSLS error estimator is given by $\mathcal{F}(p_h, U_h; f) = \|\nabla \cdot U_h + f\|^2_{L^2(\Omega)} + \|U_h - \nabla p_h\|^2_{(L^2(\Omega))^2} + \|\nabla \times U_h\|^2_{L^2(\Omega)}$. Under certain smoothness assumptions, the FOSLS error estimator is equivalent to the H^1-norm [8]: $\mathcal{F}(p_h, U_h; f) \approx ||p - p_h||^2_{H^1(\Omega)} + \|U - U_h\|^2_{(H^1(\Omega))^2}$.

6.3. Numerical results



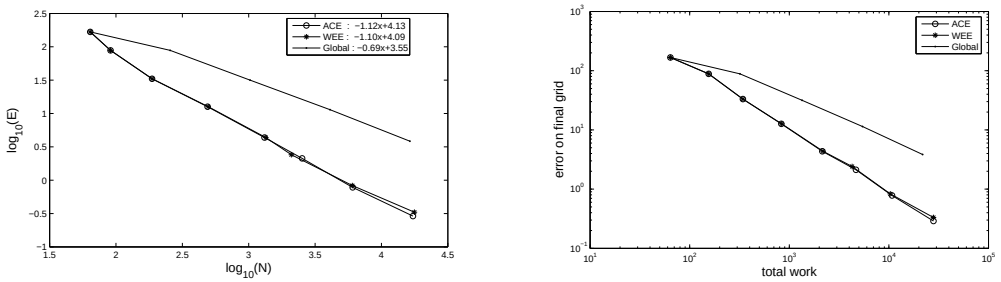
(a) (b)

Figure 25. Efficiency-based refinement strategies for the 2D model problem with $p = 1$. (a) Error versus DOF. (b) Final error, E_L, versus total work, $\sum_{\ell=1}^{L} N_\ell$.

We present numerical results for the 2D model problem with $p = 1$ and $p = 2$ in Figs. 25 and 26, respectively. The figures show error versus DOF and total work for the WEE and ACE refinement strategies, compared with global refinement in every step.

Copyright © 2007 John Wiley & Sons, Ltd.

Prepared using nlaauth.cls



(a) (b)

Figure 26. Efficiency-based refinement strategies for the 2D model problem with $p = 2$. (a) Error versus DOF. (b) Final error, E_L, versus total work, $\sum_{\ell=1}^{L} N_\ell$.

For $p = 1$, the ACE strategy results in an efficient algorithm, but, as expected, the WEE strategy produces many small refinement steps for this case where $d > p$, and is, thus, not efficient (Fig. 25). Fig. 26 shows that, for $p = 2$ ($d = p$), both the ACE and WEE strategies produce an efficient refinement process.

Fig. 24(b) shows the resulting grid when the ACE strategy is applied to a slightly more complicated test problem, in which two circular steps are superimposed ($u = 1$ in the lower left corner, $u = 2$ in the lower right corner, $u = 3$ where the two steps overlap, and $u = 0$ in the top part of the domain). The adaptive refinement process adequately captures the error generated at the steep gradients.

7. Conclusions

Two efficiency-based adaptive refinement strategies for finite element methods, WEE and ACE, were discussed. The two strategies take both error reduction and work into account. The two strategies were first compared for a 1D model problem. For the case of h-refinement with smooth solutions, the efficiency-based strategies generate a highly accurate grid sequence and an efficient refinement process. However, for singular solutions, the refinement process becomes inefficient due to many steps of small refinements. Use of a graded grid for elements with a singularity leads to significant improvement. For both the WEE and ACE strategies, this modification saves a lot of work, and also results in a highly accurate grid sequence. For the hp-refinement case, similar conclusions are obtained. However, for general problems, the difficulty here may lie in how to find a good approximation for the p-error reduction factor. Application to problems with spatial dimension larger than one shows that the WEE strategy is inefficient when the dimension, d, is larger than the finite element order, p. The ACE strategy, however, produces an efficient refinement process for any combination of d and p.

Future work will include application of these grid refinement strategies to problems with singularities in multiple spatial dimensions. Also, an idea to be explored in the future is to enhance the refinement strategies by allowing double or triple refinement for some elements, and determining, in each step, the optimal number of elements to be refined once, twice and thrice. More realistic measures for computational work must be considered, that may, for

Copyright © 2007 John Wiley & Sons, Ltd.

Prepared using nlaauth.cls

instance, take into account matrix assembly costs and multigrid convergence factors, and their dependence on the finite element order and the spatial dimension of the problem. Another topic of interest is the parallelization of the efficiency-based refinement strategies. Binning strategies need to be considered in order to reduce the work for minimizing the efficiency measures, and to reduce the communication between processors [4]. Also, load balancing issues are important for parallel adaptive methods (see, e.g., [10]). After initial solution of a coarse level problem on a single processor, the domain may be partitioned such that each parallel processor receives a subdomain with approximately the same amount of error. This may be a fruitful strategy for load balancing in that, as the grid becomes finer, the optimal refinement approaches global refinement, which requires minimal load balancing. This will be explored in future research.

REFERENCES

1. U. Ruede, Mathematical and Computational Techniques for Multilevel Adaptive Methods, Volume 13 of Frontiers in Applied Mathematics, SIAM, Philadelphia (1993).
2. R. Verfuerth, A Review of a Posteriori Error Estimation and Adaptive Mesh-Refinement Techniques, Teubner Verlag and J. Wiley, Stuttgart, (1996).
3. C. Schwab, p- and hp-Finite Element Methods, Clarendon press, Oxford (1998).
4. M. Berndt, T. A. Manteuffel and S. F. McCormick, Local Error Estimates and Adaptive Refinement for First-Order System Least Squares (FOSLS), *E.T.N.A.* 6, pp. 35-43 (1997).
5. W. Gui and I. Babuška, The h, p and hp Versions of the Finite Element Method in 1 Dimension, Parts I, II, III. *Numerische Mathematik* 49, pp. 577-683 (1986).
6. S. C. Brenner and L. R. Scott, The Mathematical Theory of Finite Element Methods, Springer-Verlag, New York (1996).
7. Z. Cai, R. Lazarov, T. A. Manteuffel, and S. F. McCormick, First-Order System Least Squares for Second-Order Partial Differential Equations. I, *SIAM J. Numer. Anal.* 31, pp. 1785–1799 (1994).
8. Z. Cai, T. A. Manteuffel, and S. F. McCormick, First-Order System Least Squares for Second-Order Partial Differential Equations. II, *SIAM J. Numer. Anal.* 34, pp. 425–454 (1997).
9. P.B. Bochev and M.D. Gunzburger, Finite Element Methods of Least-Squares Type, *SIAM Review* 40, pp. 789–837 (1998).
10. R.E. Bank and M.J. Holst, A New Paradigm for Parallel Adaptive Meshing Algorithms, *SIAM Review* 45, pp. 292-323 (2003).

Copyright © 2007 John Wiley & Sons, Ltd.

Prepared using nlaauth.cls