Adaptive Non-Nested Multigrid Methods

MARCO L. BITTENCOURT (*)
CRAIG C. DOUGLAS (**)
RAÚL A. FEIJÓO (***)

* Center for Computational Sciences
University of Kentucky
325 McVey Hall, Lexington, KY, 40506-0045, USA
e-mail: mlb@ccs.uky.edu

** Department of Mathematics
University of Kentucky
715 Patterson Office Tower, Lexington, KY, 40506-0027, USA
e-mail: douglas@ccs.uky.edu

*** Laboratório Nacional de Computação Científica (LNCC/CNPq)
Av. Getulio Vargas 333, CEP 25651-070, Petrópolis, RJ, Brazil
e-mail: feij@alpha.lncc.br

SUMMARY

This paper presents some linear adaptive non-nested multigrid methods which are applied to linear elastic problems discretized with triangular and tetrahedral finite elements using unstructured and Delaunay mesh generators. The Zienkiewicz-Zhu error estimator and a h-refinement procedure are used to obtain the non-nested meshes used by the multigrid methods. We solve problems with a specified percentage error in the energy norm using the optimal performance of multigrid methods.

1 Introduction

The application of the finite element method [1] for solving linear elliptic problems requires the solution of

$$\mathbf{A}\mathbf{u} = \mathbf{b}, \tag{1}$$

where $\mathbf{A}$ is a symmetric matrix of order N and $\mathbf{u}$ and $\mathbf{b}$ are the vectors of unknown and independent terms. Direct, iterative, and multigrid algorithms are commonly used to solve (1) when it is sufficiently similar enough to Poisson's equation on an uniform mesh [1, 4, 9].

Multigrid methods use a sequence of meshes with different refinement levels to solve a problem on the finest mesh. One of the main features of multigrid method is a linear asymptotic cost $\mathcal{O}(N)$ for the solution of (1).

It is well known that the stationary relaxation procedures have the smoothing property [14]. For an approximation $\mathbf{v}$ to the solution $\mathbf{u}$, the oscillatory components of the algebraic error $\mathbf{e} = \mathbf{u} - \mathbf{v}$ are eliminated with just a few iterations.

When a smooth function on a fine mesh is projected onto a coarse mesh it becomes more oscillatory on the coarse mesh. After damping, the oscillatory components of the approximation $\mathbf{v}$ on the finest mesh, the algebraic error is mapped onto a coarser mesh. Then relaxations are performed on the error equation $\mathbf{Ae} = \mathbf{r}$, where $\mathbf{r} = \mathbf{b} - \mathbf{Av}$ is the residual associated with the approximation $\mathbf{v}$. The same procedure can be applied recursively. On the coarsest mesh, an exact solution is calculated and the solution $\mathbf{e}$ is used to correct the approximation on the next level. This is repeated until reaching the finest mesh. This procedure is called a coarse grid correction scheme. (The actual procedure described is known as a V cycle.)

There are other multigrid procedures. One way (cascadic) multigrid calculates a better initial approximation for a finest mesh by relaxation on a sequence of coarser meshes. Relaxation schemes, nested iterations, coarse grid correction scheme, and transfer operators are the main elements of multigrid methods.

Traditionally, multigrid methods have been used with a sequence of nested meshes where the coarse mesh unknowns are also fine mesh unknowns. This feature simplifies the definition of the transfer operators. However, engineering problems often have complex geometries. This makes it difficult to generate a sequence of nested meshes. Unstructured and non-nested multigrid procedures were defined in [4, 5, 6, 7] using frontal [12] and Delaunay [10] mesh generators for the analysis of two and three dimensional linear elastic problems.

Error estimators and adaptive analysis procedures are available for many kinds of problems. Given a solution on a given mesh, local and global a posteriori error estimates can be calculated and the mesh refined. The solution for this refined mesh will have a global error close to a specified admissible error.

For coercive problems, a posteriori error estimators are equivalent in some sense with lower and upper error bounds for the finite element discretization errors. Verfürth [18] classified them as weighting methods [21], residual techniques [2, 3], local solution methods [3], or hierarchical base procedures [22].

In this work, the Zienckiewicz-Zhu (ZZ) error estimator for coercive problems (based on a weighting method) is considered [21]. This error estimator calculates an improved approximation for the derivative of the finite element solution. The global error is obtained by a superposition of local element errors and the mesh is refined.

The main objective of this paper is to use multigrid methods, error estimators, and adaptive refinement procedures to define adaptive multigrid strategies. These strategies are applied to two and three dimensional elastic problems discretized by triangles and tetrahedron non-nested meshes. Initially, the Zienckiewicz-Zhu (ZZ) error estimator is presented in Section 2. Afterwards standard and adaptive multigrid strategies are defined in Sections 3 and 4. Finally, some results are presented for elastic problems in Section 5.

2 Zienkiewicz-Zhu Error Estimator

The equilibrium equation for a continuous media with domain Ω is given by

$$\mathrm{div}\mathbf{T} + \mathbf{f} = \mathbf{0}, \tag{2}$$

where $\mathbf{T}$ is the Cauchy stress tensor and $\mathbf{f}$ is the body force vector field [13]. For a linear elastic problem, the stress tensor is related to the infinitesimal strain tensor $\mathbf{E} = \frac{1}{2}\left(\nabla\mathbf{u} + \nabla\mathbf{u}^T\right) = \nabla^s\mathbf{u}$ linearly as $\mathbf{T} = \mathbf{D}\left[\mathbf{E}\right]$. The essential and natural boundary conditions on $\partial\Omega$ are defined by $\mathbf{u} = \mathbf{0}$ ($\mathbf{x} \in \Gamma^1$) and $\mathbf{Tn} = \mathbf{\Phi}$ ($\mathbf{x} \in \Gamma^2$). We partition the boundary into $\partial\Omega = \Gamma^0 \cup \Gamma^1 \cup \Gamma^2$ and $\Gamma^0 \cap \Gamma^1 \cap \Gamma^2 = \emptyset$. Γ^0 is a part of the boundary $\partial\Omega$ without any prescribed boundary condition.

Assuming that there is no prescribed traction ($\Gamma^2 = \emptyset$), the weak form of (2) is given by

$$a(u, v) = l(v).$$

The bilinear form is defined by

$$a(u,v) = \int_\Omega \mathbf{D}\nabla^s u \nabla^s v d\Omega = \int_\Omega \mathbf{D}^{-1}\mathbf{T}(u)\cdot\mathbf{T}(v)\, d\Omega.$$

The linear functional is defined by

$$l(v) = (\mathbf{f}, v) = \int_\Omega \mathbf{f} v\, d\Omega.$$

The energy norm is defined by

$$|||u||| = \sqrt{a(u,u)} = \left(\int_\Omega \mathbf{D}^{-1}\mathbf{T}(u)\cdot\mathbf{T}(u)\, d\Omega\right)^{\frac{1}{2}}. \tag{3}$$

Given the exact solution $(u, \mathbf{T})$ and the finite element solution $(u_h, \mathbf{T}_h)$, (3) can be used to calculate the norms $|||u|||$ and $|||u_h|||$. Subtracting $|||u_h|||^2$ from $|||u|||^2$, we get the energy norm of the error:

$$|||e||| = |||u - u_h||| = (a(u-u_h, u-u_h))^{\frac{1}{2}} = \left(\int_\Omega \mathbf{D}^{-1}(\mathbf{T}-\mathbf{T}_h)\cdot(\mathbf{T}-\mathbf{T}_h)\, d\Omega\right)^{\frac{1}{2}}. \tag{4}$$

Assuming that the exact solution $(u, \mathbf{T})$ is not known, an error estimate is calculated using a stress recovery procedure. We get a better approximation $\mathbf{T}^*$ to $\mathbf{T}$ compared to $\mathbf{T}_h$. An estimate of the error norm,

$$|||e^*||| = \left(\int_\Omega \mathbf{D}^{-1}(\mathbf{T}^*-\mathbf{T}_h)\cdot(\mathbf{T}^*-\mathbf{T}_h)\, d\Omega\right)^{\frac{1}{2}},$$

is obtained from (4).

The global relative percentual error is defined by

$$\eta = \frac{|||e|||}{|||u|||} \times 100\%. \tag{5}$$

Given a percentual admissible error $\bar{\eta}$, the final solution should be such that $\eta \le \bar{\eta}$. Using e^* in (5) gives us

$$\eta \approx \frac{|||e^*|||}{(|||u_h|||^2 + |||e^*|||^2)^{\frac{1}{2}}} \le \bar{\eta}. \tag{6}$$

An uniform error distribution is desired in adaptive algorithms. Thus, the local error in all elements in a mesh should be smaller than a given specified value. The square of the error norm can be calculated by the local contribution of the N^{els} finite elements of the mesh by $|||e^*|||^2 = \sum_{i=1}^{N^{els}} |||e_i^*|||^2$. Assuming an uniform distribution of the error,

$$|||e^*|||^2 = N^{els}\, |||e^*_{ad}|||^2, \tag{7}$$

where $|||e^*_{ad}|||$ is the admissible energy norm of element error.

Substituting (7) into (6) gives us

$$\eta = \frac{(N^{els}|||e^*_{ad}|||^2)^{\frac{1}{2}}}{(|||u_h|||^2 + |||e^*|||^2)^{\frac{1}{2}}} \le \bar{\eta} \rightarrow |||e^*_{ad}||| \le \bar{\eta}\frac{(|||u_h|||^2 + |||e^*|||^2)^{\frac{1}{2}}}{\sqrt{N^{els}}}. \tag{8}$$

For each element $i = 1, \ldots, N^{els}$, ζ_i is defined by

$$\zeta_i = \frac{|||e_i^*|||}{|||e_{ad}^*|||}. \tag{9}$$

ζ_i is the refinement indicator. When $\zeta_i > 1$, the element must be refined. When $\zeta_i \leq 1$, the element size can be increased or left unchanged.

The mathematical analysis of the FEM tells us that the error norm is bound by

$$|||e||| \leq \alpha h^p. \tag{10}$$

This assumes that there are no discontinuities, h is the element size, and p is the interpolation function order. From (10) and (9), the new element size h_i' is given by

$$\zeta_i = \frac{\alpha (h_i)^p}{\alpha (h_i')^p} \quad \rightarrow \quad h_i' = \frac{h_i}{\zeta_i^{1/p}}. \tag{11}$$

For $\zeta_i > 1$, the new element size h_i' will be smaller than h_i. For $\zeta_i < 1$, it follows that $h_i' < h_i$. A damping factor is defined in order to limit element sizes. This factor results in higher quality meshes. For $\zeta_i < 1$, we use $\zeta_i \leftarrow \zeta_i + \frac{(1-\zeta_i)}{2}$. Hence, the element size will not increase more than $2^{1/p}$ times its original size for each refinement.

After calculating the new element sizes, the respective values are projected onto the mesh nodes and a continuous function is obtained on the analysis domain. The new element size at any point of the domain is interpolated using the shape functions of the element containing the given point. This function is called the background mesh and is used by the mesh generator software in the refinement procedure [12].

A fundamental point is the stress recovery procedure to calculate the estimate $\mathbf{T}^*$ from $\mathbf{T}_h$. The cost of the stress recovery procedure must be smaller than the cost of obtaining the approximated solution $(u_h, \mathbf{T}_h)$. Some recovery procedures are described in [11, 19, 20, 21, 23].

3 Multigrid Techniques

Multigrid techniques use a sequence of meshes to calculate the solution on the finest mesh. Transfer operators must be defined to map functions between two generic meshes Ω^k and Ω^{k-1}. The restriction operator I_k^{k-1} is used to project functions (e.g., the residual $\mathbf{r}$) from a fine mesh onto a coarse mesh. The prolongation (interpolation) operator I_{k-1}^k maps functions (e.g., corrections $\mathbf{e}$ and initial approximations $\mathbf{v}$) from a coarse to a fine mesh. A variational formulation and implementation details for non-nested operators are discussed in [5, 6, 8].

Figure 1 shows some commonly used multigrid strategies. The V and W cycles are based on the recursive application of the coarse grid correction scheme. The V cycle begins on the finest mesh k. Then ν_1 pre-relaxations are performed to obtain an initial approximation $\mathbf{v}$ for the solution $\mathbf{u}$ of (1). The residual $\mathbf{r} = \mathbf{b} - \mathbf{A}\mathbf{v}$ is calculated and projected onto the coarse mesh $k-1$ using the restriction operator I_k^{k-1}. On the coarser meshes the same procedure is used except that the residual equation $\mathbf{A}\mathbf{e} = \mathbf{r}$ is solved. On the coarsest mesh the error equation is solved by a direct or iterative numerical method. A correction $\mathbf{e}$ is mapped to the other levels using the prolongation operator I_{k-1}^k. The current approximation is updated using $\mathbf{v}^k \leftarrow \mathbf{v}^k + I_{k-1}^k \mathbf{e}^{k-1}$. Finally ν_2 post-relaxations are performed on each mesh.

The FMV algorithm uses the concept of one way multigrid. FMV obtains a better initial approximation for the intermediate V cycle. Each V cycle is preceded by one or more V cycles

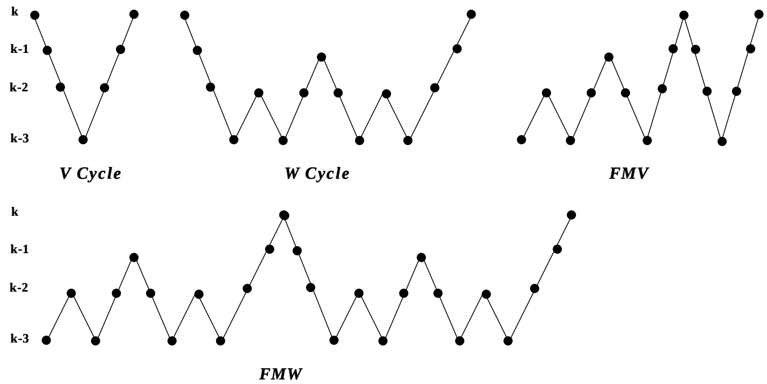


Figure 1: Multigrid strategies.

on coarser grids, depending on the value of a parameter ν_0. $\mathbf{Au} = \mathbf{b}$ is solved on coarser grids and their solutions are interpolated as the initial approximation on the next finer level using the prolongation operator I_{k-1}^{k}. Then ν_0 V cycles are performed. The result is a better approximation on the next finer mesh. This procedure is repeated until the finest mesh is reached.

The FMW algorithm uses W cycles on coarser grids to obtain a better initial approximation on a finer level. It is similar to the FMV cycle. These procedures are illustrated in the Figure 1 for $\nu_0 = 1$. Some variations of these algorithms may be used, e.g., the FMVV algorithm uses a FMV cycle followed by many V cycles.

4 Adaptive Multigrid Algorithms

Error estimators like the Zienkiewicz-Zhu (ZZ) procedure may be used with multigrid strategies to produce top down and bottom up adaptive multigrid procedures. Thus, we do not need to know the meshes in advance since they can be generated during the adaptive process.

The top down approach uses the finest mesh with a multigrid strategy. This mesh can be obtained by an adaptive procedure using an error estimator, a refinement technique, and a direct or iterative solution method. The final solution error η in the adaptive procedure should be close to a specified admissible error $\bar{\eta}$.

Our top down refinement procedure uses a new average size of element $h' \approx h/2$. Sometimes the sequence of meshes obtained by this process cannot assure the convergence of the multigrid procedure [4, 6]. In this case intermediate meshes are obtained by taking the finest mesh and using the mesh generator program with a new element size that is a factor of two larger [12]. After generating the meshes, the multigrid algorithm is used to obtain the final solution.

A similar procedure without adaptive refinement is defined in other works [15, 16, 17]. The intermediate levels are generated from the finest mesh by applying frontal and Delaunay techniques. The idea is to solve the problem on the finest mesh using the optimal order of a multigrid method. The number of equations in the finest mesh is assumed to be sufficiently large so that the cost of mesh generation is negligible when compared to the cost of solving the problem.

The main idea of the bottom up approach is to start from the coarsest mesh and solve the problem by applying multigrid strategies for a given percentage error $\bar{\eta}$ and an accuracy ξ.

Since the number of unknowns on the coarsest mesh is small, sparse Gaussian elimination can be used efficiently to calculate the solution of the system of equations. A good initial approximation is thus obtained for the problem. The second mesh is generated using the error distribution given by the ZZ error estimator with a specified error $\bar{\eta}$ and a constrained refinement procedure. For two dimensional problems the following refinement constraint was used to determine the new element size h' in order to assure the convergence of the multigrid methods:

$$\frac{1}{3}h \leq h' \leq \frac{3}{2}h. \tag{12}$$

The FMV cycle with $\nu_0 = 2$ (see the Figure 2) was chosen as the multigrid strategy to generate the other meshes. We also tried $\nu_0 = 1$ and $\nu_0 = 3$, but the best computational performance in terms of number of operations was achieved with $\nu_0 = 2$. We observed that using two V cycles did not guarantee a solution with accuracy ξ. The FMV cycle with $\nu_0 = 2$ was used only for intermediate mesh definition and it was not necessary to obtain an exact solution on each level. However, when the finest mesh was obtained, a multigrid method (though not necessarily FMV with $\nu_0 = 2$ as shown in Figure 2) was used to solve the system of equations with accuracy ξ.

In summary, the coarsest mesh is solved by sparse Gaussian elimination, the ZZ error estimator is applied, and the constrained refinement procedure generates the second mesh. Two intermediate V cycles are used and the third mesh is obtained similarly after the application of the ZZ procedure. This process is repeated until the specified maximum number of meshes is reached or the estimated error is close enough to the admissible value $\bar{\eta}$.

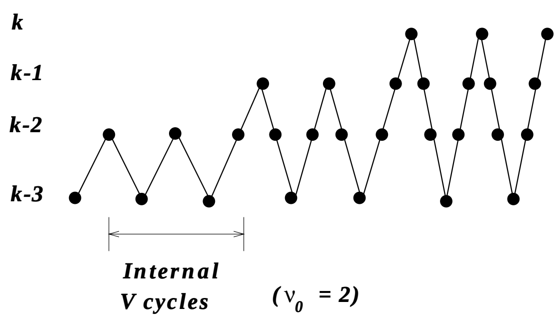


Figure 2: An adaptive FMV scheme.

5 Numerical Experiments

Adaptive multigrid procedures are applied to two and three dimensional linear elastic examples. Figure 3 shows two dimensional examples consisting of a plate with a hole problem and a fracture problem. Both are solved in Sections 5.1 and 5.2 by a bottom up adaptive multigrid procedure, an adaptive sparse Gaussian elimination (SGE) method, and conjugate gradients with a symmetric Gauss-Seidel preconditioner (CGGS). All procedures are implemented in C++ and the sparse matrices are stored in a row compressed format [4, 8].

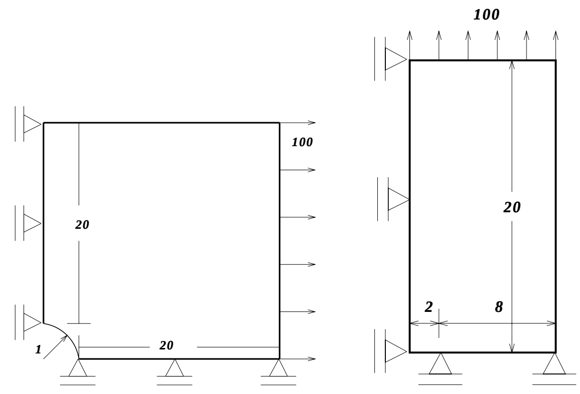
Since the adaptive multigrid refinement criterion (12) is different from than that used with the SGE and CGGS methods, different sequences of meshes were obtained. Both methods used the same initial coarsest mesh and attempted to get an estimated error close to the specified error $\bar{\eta}$. The performance of the adaptive multigrid procedure and the adaptive SGE/CGGS procedure are compared in this section.

The top down adaptive multigrid strategy was applied to three dimensional fracture problems in Section 5.3. Multigrid results are compared with ones for sparse Gaussian elimination and iterative procedures based on conjugate gradient method.

The following relative errors are used for comparing the two dimensional solutions calculated with direct ($\mathbf{u}_{dir}$), iterative ($\mathbf{u}_{ite}$), and multigrid ($\mathbf{u}_{mg}$) methods:

$$||e_r^{dir/ite}||_2 = \frac{||\mathbf{u}_{dir}-\mathbf{u}_{ite}||_2}{||\mathbf{u}_{dir}||_2}, \qquad ||e_r^{dir/mg}||_2 = \frac{||\mathbf{u}_{dir}-\mathbf{u}_{mg}||_2}{||\mathbf{u}_{dir}||_2}, \quad \text{and} \quad ||e_r^{ite/mg}||_2 = \frac{||\mathbf{u}_{ite}-\mathbf{u}_{mg}||_2}{||\mathbf{u}_{ite}||_2}. \tag{13}$$

The convergence criterion $\frac{||\mathbf{Au}-\mathbf{b}||_2}{||\mathbf{b}||_2} < \xi$ with precision $\xi = 10^{-4}$ was used in the multigrid and iterative procedures. A floating point operation (flop) is a multiplication. For all multigrid strategies, the residual was calculated during the Gauss-Seidel relaxations [6]. This reduced by about 15% the number of operations as compared to [7] where the residual was calculated separately.



(a) Plate with a hole. (b) Fracture.

Figure 3: Two dimensional examples ($E = 2.1 \times 10^5$; $\nu = 0,3$).

5.1 Plate with a hole problem

Figures 4 and 5 show the mesh sequences obtained by the adaptive multigrid strategy and an adaptive procedure based on SGE or CGGS with $\bar{\eta} = 1\%$. The mesh features are given in Table 1: the estimated error (η), the numbers of nodes, elements and equations, the total number of global matrix coefficients for direct (NCoef^{dir}) and iterative/multigrid (NCoef^{ite}) methods, the average number of coefficients per row of the sparse global matrix for direct (m^{dir}) and iterative/multigrid (m^{ite}) methods. For the multigrid adaptive strategy, the given η values were calculated at the end of the mesh generation process. The final solution value was $\eta = 1.1$ with a precision of $\xi = 10^{-4}$.

Four meshes were necessary for both adaptive procedures to reach a solution with an estimated error close to the specified admissible error $\bar{\eta}$. Table 2 gives the results for SGE, CGGS, and multigrid methods. The total number of cycles (NC), the value of ν_0, and the number of pre- and post-relaxations (ν_1 and ν_2) are also given for the multigrid strategies in Table 2. Figure 6 shows bar plots for the equivalent number of finest mesh iterations, number of operations in megaflops, and memory space in kilobytes.

5.2 Fracture problem

Following the same steps that were used Section 5.1, the fracture problem was solved adaptively. Figures 7 and 8 illustrate the mesh sequences obtained by the adaptive multigrid and

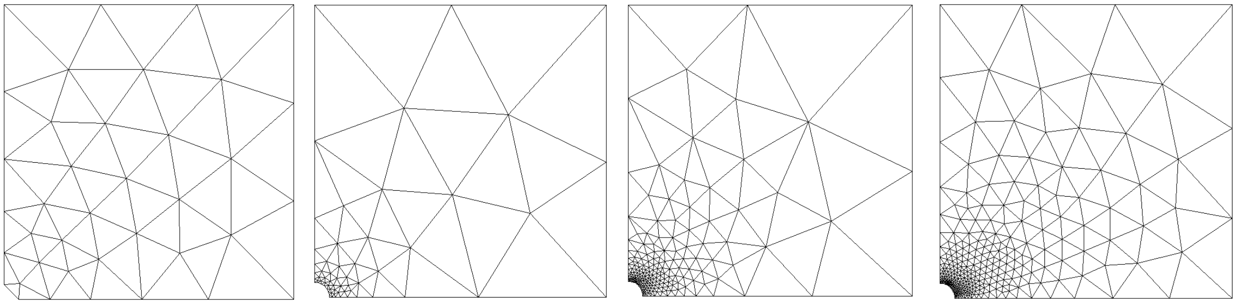


Figure 4: Meshes for plate with hole using the adaptive SGE/CGGS procedure.

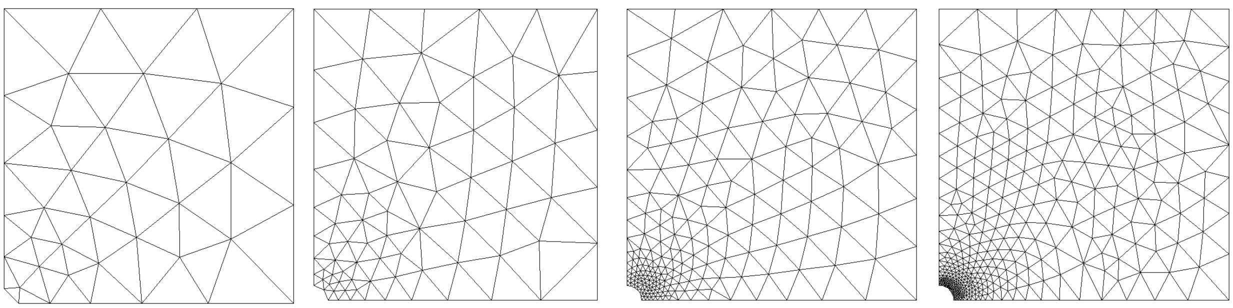


Figure 5: Meshes for plate with hole using the adaptive multigrid procedure.

Adaptive direct/iterative procedure								
Mesh	η (%)	Nodes	Elements	Equations	m^{dir}	NCoef^{dir}	m^{ite}	NCoef^{ite}
1	2.9	38	57	64	9.3	593	6.0	381
2	2.6	62	95	106	10.4	1104	6.1	650
3	1.4	231	414	434	19.6	8514	6.9	3003
4	1.1	379	694	723	21.7	15709	7.0	5094
Adaptive multigrid procedure								
Mesh	η (%)	Nodes	Elements	Equations	m^{dir}	NCoef^{dir}	m^{ite}	NCoef^{ite}
1	2.9	38	57	64	9.3	593	6.0	381
2	3.1	80	129	141	12.0	1686	6.4	905
3	1.8	208	368	386	19.0	7313	6.9	2652
4	1.0	438	799	832	22.9	19063	7.0	5848

Table 1: Mesh attributes for the plate with hole.

Method	NIT	NC, ν_0, ν_1, ν_2	$\|\|e_r^{dir/ite}\|\|_2$	$\|\|e_r^{dir/mg}\|\|_2$	$\|\|e_r^{ite/mg}\|\|_2$
SGE	36	–	–	–	–
CGGS	157	–	2.60×10^{-5}	–	–
FMV	39	1,2,1,1	–	1.40×10^{-4}	1.44×10^{-4}
	43	2,1,1,1	–	7.00×10^{-5}	7.65×10^{-5}
V	46	5,1,1,1	–	1.59×10^{-4}	1.63×10^{-4}
	47	4,1,2,1	–	1.77×10^{-4}	1.81×10^{-4}
W	45	3,1,1,1	–	2.29×10^{-5}	3.60×10^{-5}

Table 2: Results for the plate with hole.

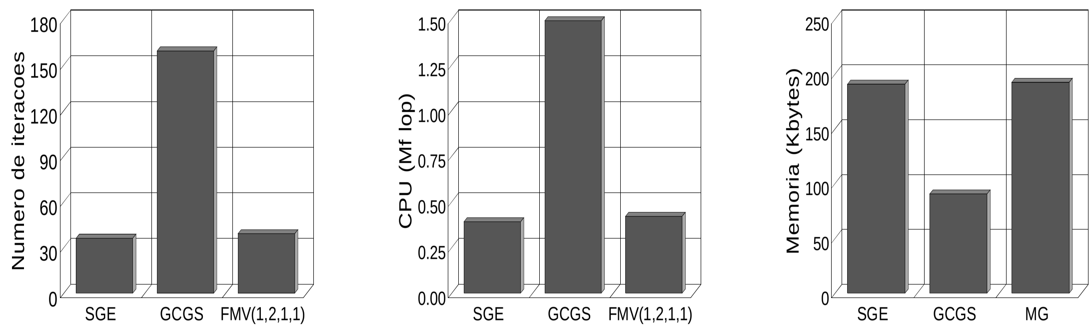


Figure 6: Results for the plate with hole.

adaptive SGE procedures with $\bar{\eta} = 2\%$. The mesh features are given in Table 3. The results of SGE, CGGS, and multigrid methods for $\xi = 10^{-3}$ and $\xi = 10^{-4}$ are given in Table 4 and Figure 9. For the multigrid case, estimated errors of $\eta = 2.3\%$ and $\eta = 2.5\%$ were obtained after the mesh generation and solution processes with a sequence of 5 meshes.

The fracture problem was also solved with a percentage error equal to $\bar{\eta} = 1\%$ and accuracy $\xi = 10^{-4}$. Three and six meshes were necessary for the adaptive direct and multigrid methods. The final estimated errors were $\eta = 1.1\%$ and $\eta = 1.3\%$. The meshes obtained are in Figures 10 and 11. The main features are given in Table 5 where η is the error obtained at the end of the mesh generation process. The results are shown in Table 6 and Figure 12.

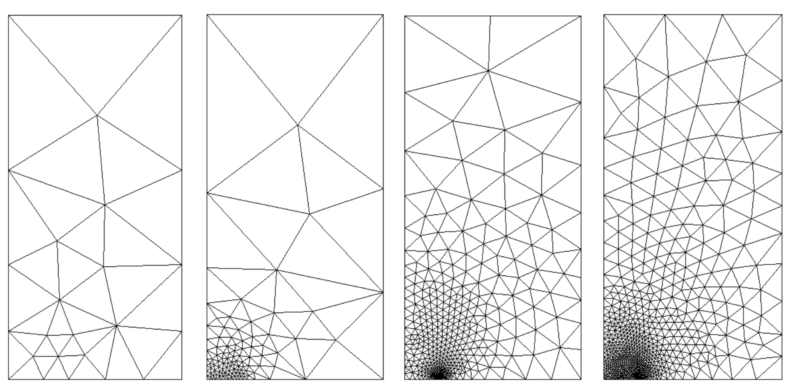


Figure 7: Meshes for the adaptive SGE fracture problem with $\bar{\eta} = 2\%$.

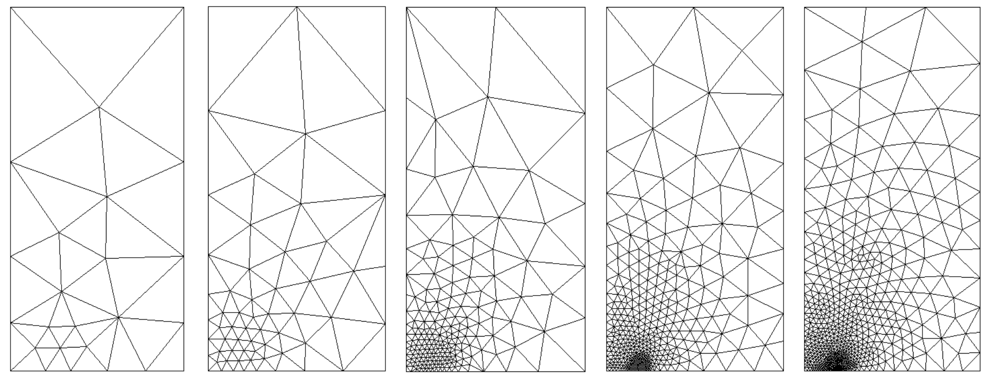


Figure 8: Meshes for the adaptive multigrid fracture problem with $\bar{\eta} = 2\%$.

Adaptive direct/iterative procedure								
Mesh	$\eta\,(\%)$	Nodes	Elements	Equations	m^{dir}	$NCoef^{dir}$	m^{ite}	$NCoef^{ite}$
1	9.3	25	34	40	7.1	283	5.5	220
2	6.3	115	196	212	14.7	3122	6.7	1413
3	3.0	556	1043	1074	27.5	29495	7.2	7702
4	2.1	968	1818	1869	29.6	55370	7.2	13433
Adaptive multigrid procedure								
Mesh	$\eta\,(\%)$	Nodes	Elements	Equations	m^{dir}	$NCoef^{dir}$	m^{ite}	$NCoef^{ite}$
1	9.3	25	34	40	7.1	283	5.5	220
2	7.7	61	97	107	11.8	1261	6.3	676
3	4.9	200	359	375	19.4	7263	6.9	2597
4	3.2	461	858	885	27.6	24386	7.1	6310
5	2.3	883	1672	1714	30.9	52930	7.2	12381

Table 3: Mesh attributes for the fracture problem with $\bar{\eta} = 2\%$.

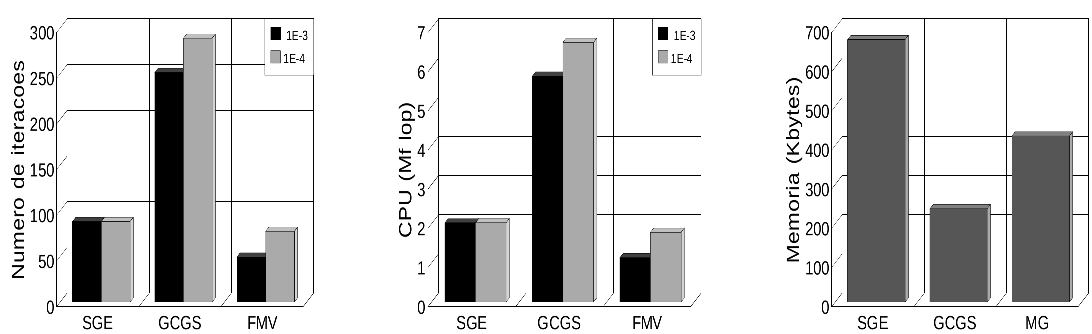


Figure 9: Results for the fracture problem with $\bar{\eta} = 2\%$.

Precision $\xi = 10^{-3}$					
Method	NIT	NC, ν_0, ν_1, ν_2	$\|\|e_r^{dir/ite}\|\|_2$	$\|\|e_r^{dir/mg}\|\|_2$	$\|\|e_r^{ite/mg}\|\|_2$
SGE	88	–	–	–	–
CGGS	251	–	6.59×10^{-4}	–	–
FMV	58	2,2,1,1	–	1.96×10^{-4}	6.15×10^{-4}
	45	1,2,2,1	–	7.18×10^{-4}	7.95×10^{-4}
	41	2,1,1,1	–	8.05×10^{-4}	8.28×10^{-4}
	50	2,1,2,1	–	3.38×10^{-4}	6.26×10^{-4}
V	38	4,1,1,1	–	1.89×10^{-3}	1.78×10^{-3}
	38	3,1,2,1	–	1.96×10^{-4}	1.87×10^{-3}
Precision $\xi = 10^{-4}$					
Method	NIT	NC, ν_0, ν_1, ν_2	$\|\|e_r^{dir/ite}\|\|_2$	$\|\|e_r^{dir/mg}\|\|_2$	$\|\|e_r^{ite/mg}\|\|_2$
SGE	88	–	–	–	–
CGGS	289	–	3.24×10^{-5}	–	–
FMV	96	4,2,1,1	–	3.34×10^{-5}	4.45×10^{-5}
	73	2,2,2,1	–	6.03×10^{-5}	6.57×10^{-5}
	77	5,1,1,1	–	2.93×10^{-5}	4.19×10^{-5}
	67	3,1,2,1	–	6.95×10^{-5}	7.39×10^{-5}
V	64	9,1,1,1	–	2.13×10^{-4}	2.11×10^{-4}
	66	7,1,2,1	–	1.96×10^{-4}	1.94×10^{-4}

Table 4: Results for the fracture problem with $\bar{\eta} = 2\%$.

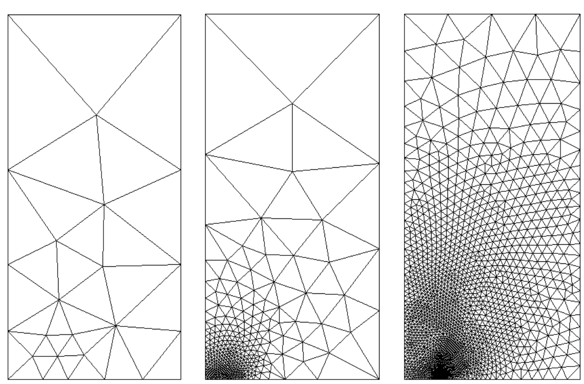


Figure 10: Meshes for the adaptive SGE fracture problem with $\bar{\eta} = 1\%$.

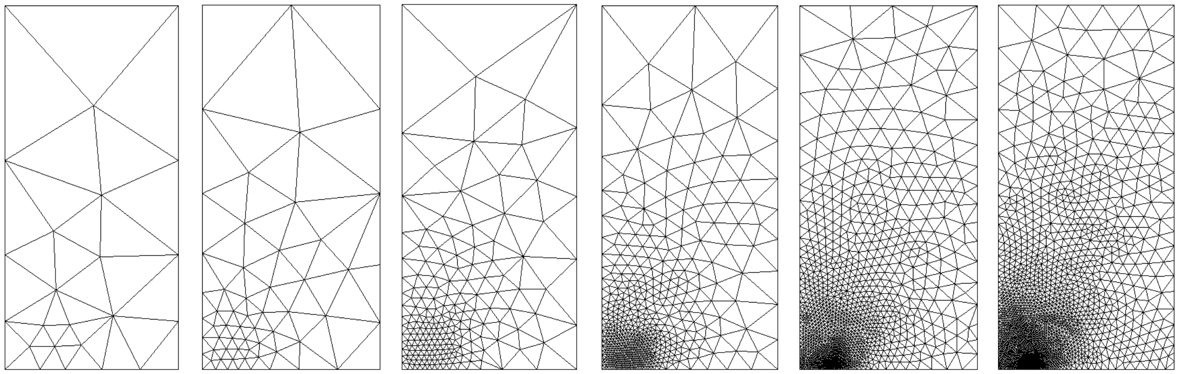


Figure 11: Meshes for the adaptive multigrid fracture problem with $\bar{\eta} = 1\%$.

Adaptive direct/iterative								
Mesh	η (%)	Nodes	Elements	Equations	m^{dir}	NCoef^{dir}	m^{ite}	NCoef^{ite}
1	9.3	25	34	40	7.1	283	5.5	220
2	4.2	404	747	775	25.9	20062	7.1	5503
3	1.3	3772	7350	7440	49.7	369737	7.4	54836
Adaptive multigrid								
Mesh	η (%)	Nodes	Elements	Equations	m^{dir}	NCoef^{dir}	m^{ite}	NCoef^{ite}
1	9.3	25	34	40	7.1	283	5.5	220
2	7.7	61	97	107	11.8	1261	6.3	676
3	5.6	198	358	371	19.9	7393	6.9	2575
4	3.8	592	1119	1144	29.4	33666	7.2	8228
5	2.2	1681	3240	3293	40.3	132808	7.3	24076
6	1.4	3369	6560	6641	46.7	310313	7.4	48904

Table 5: Mesh attributes for the adaptive fracture problem with $\bar{\eta} = 1\%$.

Method	NIT	NC, ν_0, ν_1, ν_2	$\|\|e_r^{dir/ite}\|\|_2$	$\|\|e_r^{dir/mg}\|\|_2$	$\|\|e_r^{ite/mg}\|\|_2$
SGE	207	–	–	–	–
CGGS	409	–	3.24×10^{-5}	–	–
FMV	89	4,2,1,1	–	3.34×10^{-5}	4.45×10^{-5}
	80	2,2,2,1	–	6.03×10^{-5}	6.57×10^{-5}
	88	5,1,1,1	–	2.93×10^{-5}	4.19×10^{-5}
	74	3,1,2,1	–	6.95×10^{-5}	7.39×10^{-5}
V	73	9,1,1,1	–	2.13×10^{-4}	2.11×10^{-4}
	73	7,1,2,1	–	1.96×10^{-4}	1.94×10^{-4}

Table 6: Results for the fracture problem with $\bar{\eta} = 1\%$.

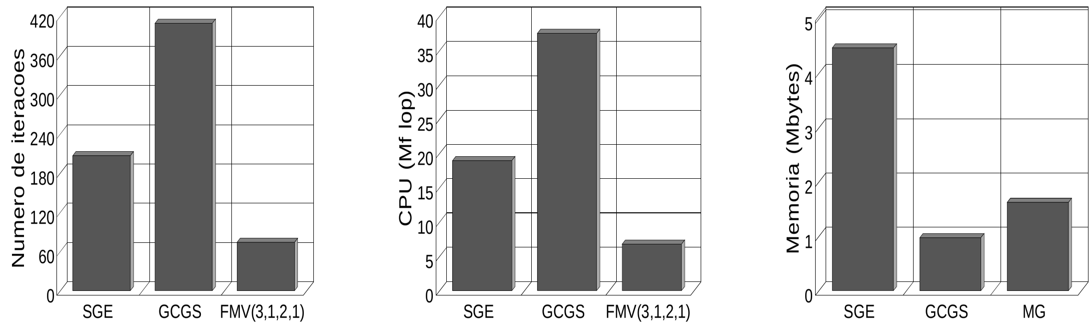


Figure 12: Results for the fracture problem with $\bar{\eta} = 1\%$.

5.3 Three dimensional problems

Figure 13 shows the linear tetrahedron meshes for a planar elliptic fracture problem in an infinite domain represented by a parallelepiped. The meshes for another planar fracture problem in a cylindrical bar (a penny shaped crack) are illustrated in Figure 14. In these two examples, symmetry conditions were considered and the sequences of meshes were generated using an adaptive procedure based on the Zienkiewicz-Zhu error estimator [21]. The mesh features are given in Table 7. These problems were also solved by conjugate gradient (CG) procedures with diagonal preconditioning (CGD), SSOR (CGSS), and symmetric Gauss-Seidel (CGGS) preconditioners. For the multigrid strategies, the coarsest mesh problem was solved using the CGGS procedure with precision $\xi = 10^{-4}$ in order to avoid the high cost of sparse Gaussian elimination. Figure 15 shows the behavior in terms of number of operations for SGE, conjugate gradient, and multigrid methods.

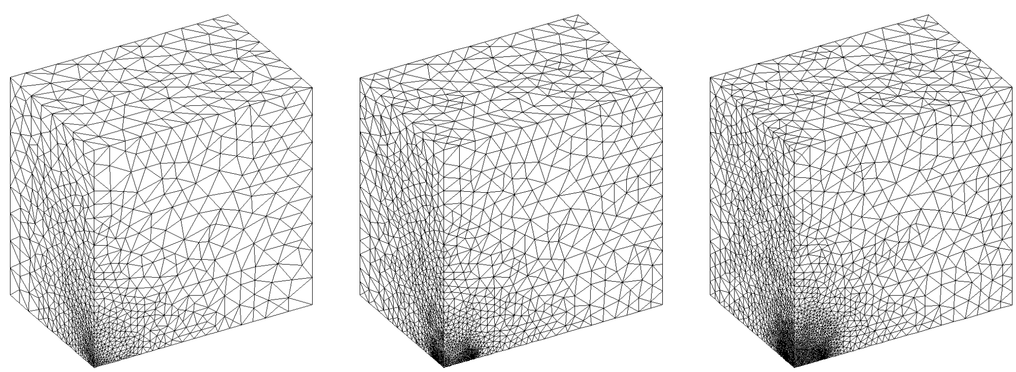


Figure 13: Meshes for the parallelepiped example ($E = 2.1 \times 10^5$; $\nu = 0.3$).

6 Conclusions

In the adaptive procedures presented here, sequences of meshes are obtained by applying the ZZ error estimator in order to solve the proposed problems by means of SGE, conjugated gradient, and multigrid methods.

For the plate with a hole problem of Section 5.1, the performance of SGE and the multigrid methods (in terms of the number of operations and memory requirements) is very similar. For

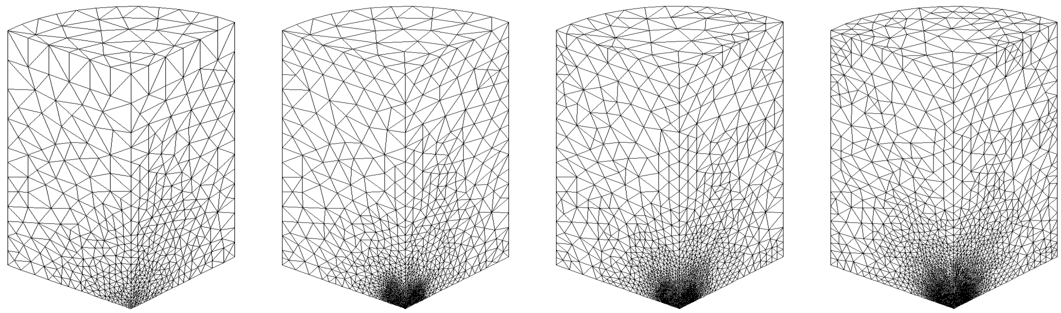
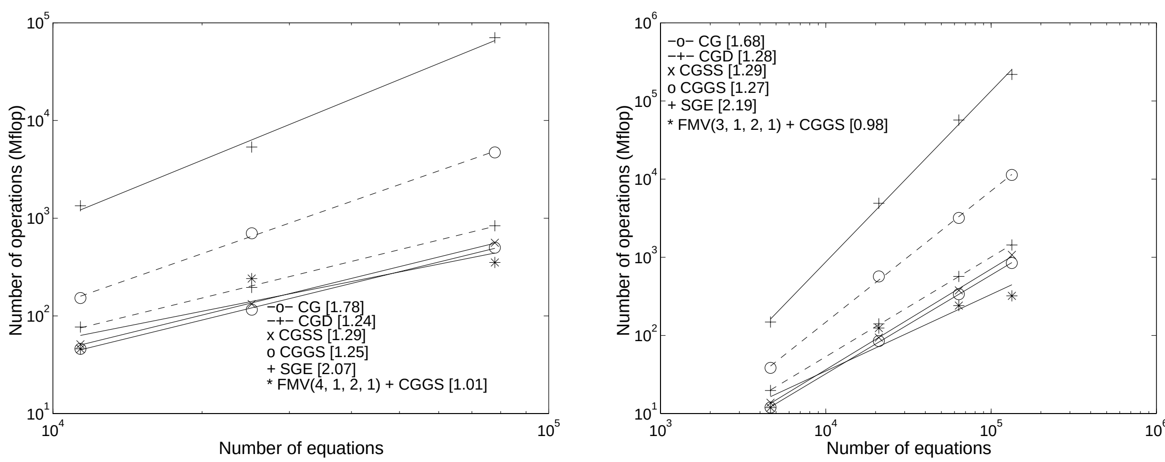


Figure 14: Meshes for the cylinder example ($E = 2.1 \times 10^5$; $\nu = 0.3$).

Parallelepiped							
Mesh	Nodes	Elements	Equations	m^{dir}	NCoef^{dir}	m^{ite}	NCoef^{ite}
1	4228	19871	11365	327.0	3716698	19.5	221037
2	9241	44015	25184	422.0	10627486	19.8	495535
3	27652	139348	77862	786.0	61196124	20.5	1596409

Cylinder							
Mesh	Nodes	Elements	Equations	m^{dir}	NCoef^{dir}	m^{ite}	NCoef^{ite}
1	1810	7599	4630	179.4	830385	18.0	83141
2	7753	36746	20933	428.2	8962786	19.6	409267
3	22670	114634	63778	792.9	50566338	20.5	1309692
4	46982	238546	133471	959.3	128036645	20.6	2746298

Table 7: Attribute of the three dimensional meshes.



(a) Parallelepiped.

(b) Cylinder.

Figure 15: Number of floating point operations for three dimensional problems.

both methods the same number of meshes is required to reach an 1% admissible error. The adaptive multigrid procedure is competitive even for this low order and moderately singular problem.

The strong singularity of the fracture problem in Section 5.2 results in a larger number of equations needed to attain the 1% and 2% levels of admissible error. For this example, multigrid techniques are superior to the SGE with respect to the required number of operations, as well as to the required amount of memory, as indicated in Tables 4 and 6 and Figures 9 and 12. However, due to the limitation in modifying the element size (12), the number of meshes for the multigrid strategies is larger than the number of meshes for SGE.

The adaptive multigrid procedures are viable alternatives. They perform similarly or better than SGE even for small problems. The bottom up multigrid procedure presented here allows the automation of the mesh generation procedure in order to obtain a better sequence of meshes to achieve the specified admissible error $\bar{\eta}$. For three dimensional problems the performance for multigrid methods is nearly optimal. As shown in Figure 15 the convergence rate is almost linear.

References

[1] O. Axelsson and V. A. Barker. *Finite Element Solution of Boundary Value Problems – Theory and Computation.* Academic Press, Orlando, 1984.

[2] I. Babuška and W. C. Rheinboldt. *A posteriori* error estimates for the finite element method. *International Journal for Numerical Methods in Engineering*, 12:1597–1615, 1978.

[3] I. Babuška and W. C. Rheinboldt. Error estimates for adaptive finite element computations. *SIAM Journal on Numerical Analysis*, 15:736–754, 1978.

[4] M. L. Bittencourt. *Adaptive Iterative and Multigrid Methods Applied to Non-Structured Meshes (in Portuguese).* PhD thesis, State University of Campinas, Brazil, 1996.

[5] M. L. Bittencourt, C. C. Douglas, and R. A. Feijóo. Non-nested and non-structured multigrid methods applied to elastic problems. Part I: The two-dimensional case. Submitted for publication, 1998.

[6] M. L. Bittencourt, C. C. Douglas, and R. A. Feijóo. Non-nested and non-structured multigrid methods applied to elastic problems. Part II: The three-dimensional case. Submitted for publication, 1998.

[7] M. L. Bittencourt and R. A. Feijóo. Non-nested multigrid methods in finite element linear structural analysis. In *Virtual Proceedings of the 8th Copper Mountain Conference on Multigrid Methods*, http://www.mgnet.org, April 1997. MGNet.

[8] M. L. Bittencourt and R. A. Feijóo. Object-oriented non-nested multigrid methods. In *IV World Conference on Computational Mechanics*, Buenos Aires, June 1998.

[9] A. Brandt. Multi-level adaptive solutions to boundary-value problems. *Mathematics of Computation*, 31(138):333–390, 1977.

[10] E. Dari. *Contribuciones a la Triangulación Automática de Dominios Tridimensionales.* PhD thesis, Instituto Balseiro, Bariloche, Argentina, 1994.

[11] E. A. Fancello and R. A. Feijóo. Adapte: error estimador for two-dimensional linear elastic problems. Technical report, National Laboratory for Scientific Computation, Rio de Janeiro, Brazil, 1992.

[12] E. A. Fancello, A. C. S. Guimarães, R. A. Feijóo, and M. Venere. Automatic two-dimensional mesh generation using object-oriented programming. In *Proceedings of 11th Brazilian Congress of Mechanical Engineering*, pages 635–638, São Paulo, December 1991. Brasilian Association of Mechanical Sciences.

[13] M. E. Gurtin. *An Introduction to Continuum Mechanics*, volume 158 of *Mathematics in Science and Engineering*. Academic Press, 1981.

[14] W. Hackbush. *Multigrid Methods*. Springer-Verlag, Berlin, 1985.

[15] R. Löhner and K. Morgan. An unstructured multigrid method for elliptic problems. *International Journal for Numerical Methods in Engineering*, 24:101–115, 1987.

[16] D. J. Mavriplis. Multigrid solution of the two-dimensional Euler equations on unstructured triangular meshes. *AIAA Journal*, 26(7):325–354, 1987.

[17] J. Peraire, J. Peiro, and K. Morgan. Multigrid solution of the 3D compressible Euler equations on unstructured tetrahedral grids. *International Journal for Numerical Methods in Engineering*, 36:1029–1044, 1993.

[18] R. Verfurth. Posteriori error estimation and adaptive mesh-refinement techniques. *Journal of Computational and Applied Mathematics*, 50(1):67–83, 1994.

[19] N. Wiberg and F. Abdulwahab. Patch recovery based on superconvergent derivatives and equilibrium. *International Journal for Numerical Methods in Engineering*, 36:2703–2724, 1993.

[20] N. Wiberg, F. Abdulwahab, and S. Ziukas. Enhanced superconvergent patch recovery incorporating equilibrium and boundary conditions. *International Journal for Numerical Methods in Engineering*, 37:3417–3440, 1994.

[21] O. C. Zienkiewicz and J. Z. Zhu. A simple error estimator and adaptative procedure for practical engineering analysis. *International Journal for Numerical Methods in Engineering*, 24:337–357, 1987.

[22] O. C. Zienkiewicz, J. Z. Zhu, and N. G. Gong. Effective and practical *h-p* version adaptive analysis procedures for the finite element method. *International Journal for Numerical Methods in Engineering*, 28:879–891, 1989.

[23] O. C. Zienkiewicz, J. Z. Zhu, and N. G. Gong. The superconvergent patch recovery and *a posteriori* error estimates. Part 1: the recovery technique. *International Journal for Numerical Methods in Engineering*, 33:1331–1364, 1992.