INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN BIOMEDICAL ENGINEERING
Int. J. Numer. Meth. Biomed. Engng. 2010; **26**:534–553
Published online 10 July 2008 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/cnm.1140
COMMUNICATIONS IN NUMERICAL METHODS IN ENGINEERING

An algebraic multigrid method for isotropic linear elasticity problems on anisotropic meshes

Yingxiong Xiao[1], Shi Shu[2,*,†], Ping Zhang[1] and Min Tan[3]

[1]*Civil Engineering and Mechanics College, Xiangtan University, Xiangtan 411105, People's Republic of China*
[2]*School of Mathematics and Computational Science, Xiangtan University, Xiangtan 411105, People's Republic of China*
[3]*School of Mathematics and Computational Science, Hunan University of Science and Technology, Xiangtan 411201, People's Republic of China*

SUMMARY

Anisotropic meshes are frequently used in many practical problems in order to provide a good resolution of the solutions on the structured mesh and thus to assure the accuracy of the discretization. However, the attributes of anisotropic meshes may have a serious effect on the convergence of linear algebraic solvers. In this paper, we focus mainly on studying the effect of the anisotropic quadrilateral meshes in two dimensions and the anisotropic hexahedral meshes in three dimensions on algebraic multigrid (AMG) solvers. We find that the commonly used AMG methods are less efficient for some locally strong anisotropic meshes. A modified algorithm of coarse grid selection to obtain a more efficient AMG method is then proposed based on the geometric grid information as nodal coordinates. The results of various numerical experiments in linear elasticity are presented. It is shown from the numerical results that the constructed AMG method is robust and efficient not only for meshes containing thin and stretched elements but also for other meshes with locally strong anisotropy. Copyright © 2008 John Wiley & Sons, Ltd.

Received 20 May 2007; Revised 10 April 2008; Accepted 11 April 2008

KEY WORDS: algebraic multigrid; linear elasticity; FEM; locally anisotropic mesh; distance matrix

1. INTRODUCTION

A linear elasticity analysis is often required during simulations of multi-physics problems in applications such as micro-electro-mechanical systems (MEMS) [1]. Typically, these modern devices involve complicated geometries, extremely high aspect ratios (length-to-height ratio) and disparate material properties. A particular example of such a case is a cantilever beam with aspect ratios in the range of 50:1–500:1. Finite element method (FEM) is the most commonly used numerical method for their analysis [2]. However, the use of FEM discretization often leads to systems of

*Correspondence to: Shi Shu, School of Mathematics and Computational Science in Xiangtan University, Xiangtan 411105, People's Republic of China.
†E-mail: shushi@xtu.edu.cn

Contract/grant sponsor: National Natural Science Foundation of China; contract/grant numbers: NSF-10771178, NSF-10672138, NSAF-10676031
Contract/grant sponsor: Basic Research Program of China; contract/grant number: 2005CB321702
Contract/grant sponsor: Key Project of Chinese Ministry of Education and Scientific Research Fund of Hunan Provincial Education Department; contract/grant numbers: 208093, 07A068
Contract/grant sponsor: Provincial Natural Science Foundation of Hunan; contract/grant number: 07JJ6004

equations that are sparse, large and ill-conditioned. In addition, anisotropic meshes are also used in the solution of some typical practical problems such as crack problems in solid mechanics and simulation of excavation process in rock mechanics and engineering. The finite element solution usually involves the mesh generation, the assembly of discrete algebraic systems and the solution of these systems by some algebraic solvers. Generally speaking, the performance of finite element computations depends critically on both the geometric meshes and the algebraic solvers. In practice, however, the attributes of the anisotropic meshes may have a serious effect on the convergence of linear algebraic solvers [3]. Most of the linear solvers are not concerned with the mesh geometry and mesh quality, except the multigrid methods with semi-coarsening or line smoothing [4, 5].

Multigrid method is by far known as one of the most efficient methods for solving large-scale algebraic systems of equations [6–9]. However, it is difficult to construct a sequence of nested discretizations or meshes needed for geometric multigrid method due to the complexities for practical application problems and the requirements for efficient 'plug-in' solvers in numerical business softwares. Algebraic multigrid (AMG) [10–13] is a method for solving matrix equations, which is based on multigrid concepts, but constructs the coarsening process in an algebraic way that requires no explicit knowledge of the geometry. Such methods have been successfully applied to many linear systems corresponding to discretizations of a scalar partial differential equation (PDE). However, the AMG method for systems of PDEs, such as the equations of linear elasticity, seems to be premature and the naive use of the scalar AMG does not lead to the robust and efficient solver, rather it deteriorates and often its convergence breaks down. We would like to refer the readers to [10, 14–19] for the recent efforts to apply AMG methods for systems of PDEs.

There exist many works on the design of efficient AMG methods for anisotropic problems using ideas such as semi-coarsening [20–22] or line smoothing [23, 24]; still, there are very few systematic studies on the effect of mesh qualities on these iterative schemes [25]. In a recent work [26], the relationship between mesh generation and algebraic solver is studied using generic numerical examples pertaining to the finite element solution of some anisotropic problems and some techniques are proposed, which are applicable to both the anisotropic triangular mesh on different domains and the linear solvers in order to obtain both accurate and efficient numerical solutions simultaneously.

In this paper, we mainly focus on studying the effect of some anisotropic quadrilateral and hexahedral meshes that contain thin and stretched elements or some other meshes with locally strong anisotropy on AMG solvers for several typical problems in linear elasticity. A modified algorithm of coarse grid selection to obtain a more efficient AMG method is proposed based on the geometric grid information as nodal coordinates. The basic idea is to use both the condensed point-block stiffness matrix according to References [10, 27] and the distance between neighboring grid points (stored in a distance matrix) in order to obtain a modified coarsening algorithm and define appropriate interpolation schemes. The results of various numerical experiments in linear elasticity are presented. It is shown from the numerical results that the constructed AMG method is robust and efficient not only for meshes containing thin and stretched elements but also for other meshes with locally strong anisotropy.

Classical AMG method selects coarse grid points algebraically according to strong couplings. It has better convergence for both isotropic meshes and stretched meshes, but converges very slowly for those locally strong anisotropic meshes. Thus, how to define appropriate strong couplings is important for the selection of coarse grid points and for the improvement of the AMG efficiency. In our modified algorithm, the role of distance matrix is to define again strong couplings between grid points and gradually decrease or eliminate the multi-scale property of the coefficient matrix on the coarse grid level or eliminate the locally strong anisotropy of meshes used in FEM discretization.

The remainder of this paper is organized as follows. In the following section we introduce the model problem considered in this paper. Some typical anisotropic meshes are then presented in Section 3. We numerically study the effect of some locally strong anisotropic meshes on the performance of a commonly used AMG and CG methods in Section 4. In Section 5, an efficient AMG method is proposed based on the geometric grid information as nodal coordinates. Some concluding remarks are discussed in the final section.

DOI: 10.1002/cnm

2. MODEL PROBLEM AND FINITE ELEMENT DISCRETIZATION

Let $\Omega \subset \mathbf{R}^d$ $(d=2,3)$ be a polygonal domain, and the strain tensor $\varepsilon(\mathbf{u}) := (\varepsilon(\mathbf{u})_{ij})_{i,j=1}^d$ be defined as $\varepsilon_{ij} := (\partial_i u_j + \partial_j u_i)/2$ with $i, j = 1, 2, \ldots, d$. We consider the equation for the displacement vector $\mathbf{u}$ as given by

$$\begin{aligned} \mu \Delta \mathbf{u} + (\lambda+\mu) \nabla \operatorname{div} \mathbf{u} &= \mathbf{f} \quad \text{in } \Omega \\ \mathbf{u} &= 0 \quad \text{on } \Gamma_0 \\ \sigma(\mathbf{u}) \cdot \mathtt{n} &= \mathbf{g} \quad \text{on } \Gamma_1 \end{aligned} \tag{1}$$

with the Lamé constants $\lambda > 0$ and $\mu > 0$, where $\mathbf{f}$ is an external force and $\mathbf{g}$ is a surface force on the boundary $\Gamma_1 \subset \partial\Omega$. Here, the body is fixed at the boundary $\Gamma_0 \subset \partial\Omega$ in which $\operatorname{mes}(\Gamma_0) \neq 0$, and $\mathtt{n}$ is the unit outward normal vector to boundary $\partial\Omega$, where $\partial\Omega = \Gamma_0 + \Gamma_1$ and $\Gamma_0 \cap \Gamma_1 = \phi$. The stress tensor $\sigma(\mathbf{u})$ in three dimensions is defined as

$$\begin{bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{33} \\ \sigma_{12} \\ \sigma_{13} \\ \sigma_{23} \end{bmatrix} = \frac{E}{(1+\nu)(1-2\nu)} \cdot \begin{bmatrix} 1-\nu & \nu & \nu & 0 & 0 & 0 \\ \nu & 1-\nu & \nu & 0 & 0 & 0 \\ \nu & \nu & 1-\nu & 0 & 0 & 0 \\ 0 & 0 & 0 & 1-2\nu & 0 & 0 \\ 0 & 0 & 0 & 0 & 1-2\nu & 0 \\ 0 & 0 & 0 & 0 & 0 & 1-2\nu \end{bmatrix} \begin{bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ \varepsilon_{33} \\ \varepsilon_{12} \\ \varepsilon_{13} \\ \varepsilon_{23} \end{bmatrix} \tag{2}$$

with $\nu \in [0, 0.5)$ being Poisson's ratio and E being Young's modulus. An analogous problem formulation, which has the same structure as (2), for the two-dimensional case corresponds to the well-known plane strain formulation [28].

Let $H^m(\Omega)$ be the usual Sobolev space of functions with $L_2(\Omega)$ derivatives up to order m, see, e.g. [29, 30]. In what follows, we consider discretizations that are obtained from the weak formulation:

Find $\mathbf{u} \in H^1_{\Gamma_0} := \{\mathbf{v} \in H^1(\Omega)^d | \mathbf{v}|_{\Gamma_0} = \mathbf{0}\}$ such that

$$a(\mathbf{u}, \mathbf{v}) = \langle \tilde{\mathbf{f}}, \mathbf{v} \rangle \quad \forall \mathbf{v} \in H^1_{\Gamma_0} \tag{3}$$

where

$$a(\mathbf{u}, \mathbf{v}) = \int_\Omega \varepsilon(\mathbf{u}) : \sigma(\mathbf{v}) \, \mathrm{d}\Omega, \quad \langle \tilde{\mathbf{f}}, \mathbf{v} \rangle = \int_\Omega \mathbf{f} \cdot \mathbf{v} \, \mathrm{d}\Omega - \int_{\partial\Omega} \mathbf{g} \cdot \mathbf{v} \, \mathrm{d}\partial\Omega$$

$$\varepsilon(\mathbf{u}) : \sigma(\mathbf{v}) = \sum_{ij} \varepsilon(\mathbf{u})_{ij} \sigma(\mathbf{v})_{ij}$$

The variational problem (3) is discretized using standard finite elements and the corresponding point-block system of linear algebraic equations (the matrix is usually sparse) can be expressed as

$$\mathbf{AU} = \mathbf{F} \tag{4}$$

or

$$\begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} & \cdots & \mathbf{A}_{1n_h} \\ \mathbf{A}_{21} & \mathbf{A}_{22} & \cdots & \mathbf{A}_{2n_h} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{A}_{n_h 1} & \mathbf{A}_{n_h 2} & \cdots & \mathbf{A}_{n_h n_h} \end{bmatrix} \begin{bmatrix} \mathbf{U}_1 \\ \mathbf{U}_2 \\ \vdots \\ \mathbf{U}_{n_h} \end{bmatrix} = \begin{bmatrix} \mathbf{F}_1 \\ \mathbf{F}_2 \\ \vdots \\ \mathbf{F}_{n_h} \end{bmatrix} \tag{5}$$

DOI: 10.1002/cnm

where n_h is the total number of nodes (after the Dirichlet boundary conditions have been applied); $\mathbf{U}_i$, $\mathbf{F}_i$ are $d\times 1$ vectors, respectively, for $i=1,2,\ldots,n_h$; and $\mathbf{A}_{ij}$ are $d\times d$ matrices with $i,j=1,2,\ldots,n_h$.

Note that for zero displacements it is easy to prove that the solution of finite element equations (4) exists and is unique. This follows from the positive definiteness by using Korn's inequality. Currently, the system of equations (4) is being solved by iterative methods. However, since the number of unknowns in a practical application will be extremely large, those commonly used iterative methods such as conjugate gradient (CG) and Gauss–Seidel methods [31] will converge very slowly and are inefficient for practically interesting large-scale problems.

In this paper we concentrate on studying the effect of anisotropic meshes including some locally anisotropic meshes as presented in the following section on the convergence of linear algebraic solvers such as CG and AMG methods for several typical problems in linear elasticity, and then present more efficient and robust AMG solver for such problems.

3. FINITE ELEMENT MESH WITH LOCALLY STRONG ANISOTROPY

Let us first consider a typical rectangular element with h_x and h_y denoting the mesh size in the x- and y-directions, respectively. If $h_x/h_y=O(1)$, then we call it an isotropic element; if $h_x/h_y\ll 1$, we call it the so-called x-type anisotropic element; and if $h_x/h_y\gg 1$, then we call it a y-type anisotropic element, as shown in Figure 1(a)–(c), respectively.

In this paper, we mainly discuss several types of representative uniform meshes with different element sizes for the unit square and for the unit cube, which are shown in Figures 2 and 3, respectively. Such meshes are often used in many practical applications by engineers. We can easily generate five uniform meshes as given in Figure 2(a)–(e) by choosing variable element lengths of the five forms. For instance, we can define the mesh spacing for Figure 2(b) by using the formulas presented as follows:

$$\textbf{(Fb)}\quad h^i_{x_b}=\begin{cases}\left(\dfrac{n_x}{4}+2-i\right)h^{\min}_{x_b}, & 1\leqslant i\leqslant\dfrac{n_x}{4}\\ h^{\min}_{x_b}, & \dfrac{n_x}{4}<i\leqslant\dfrac{3n_x}{4}\\ \left(i-\dfrac{3n_x}{4}+1\right)h^{\min}_{x_b}, & \dfrac{3n_x}{4}<i\leqslant n_x\end{cases}\quad\text{and}\quad h^j_{y_b}=1/n_y\ \ (1\leqslant j\leqslant n_y)$$

where $h^{\min}_{x_b}=1/(n_x^2/16+5n_x/4)$, and n_x and n_y are the numbers of elements in the x- and y-directions, respectively.

As we all know mesh (a) in Figure 2 and mesh (a) in Figure 3 are two typical finite element meshes with isotropy in two and three dimensions, respectively. In what follows, we shall discuss

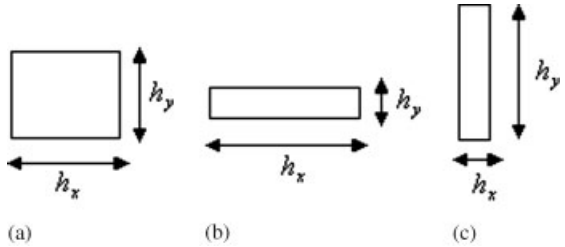

Figure 1. (a) An isotropic element; (b) the y-type anisotropic element; and (c) the x-type anisotropic element.

DOI: 10.1002/cnm

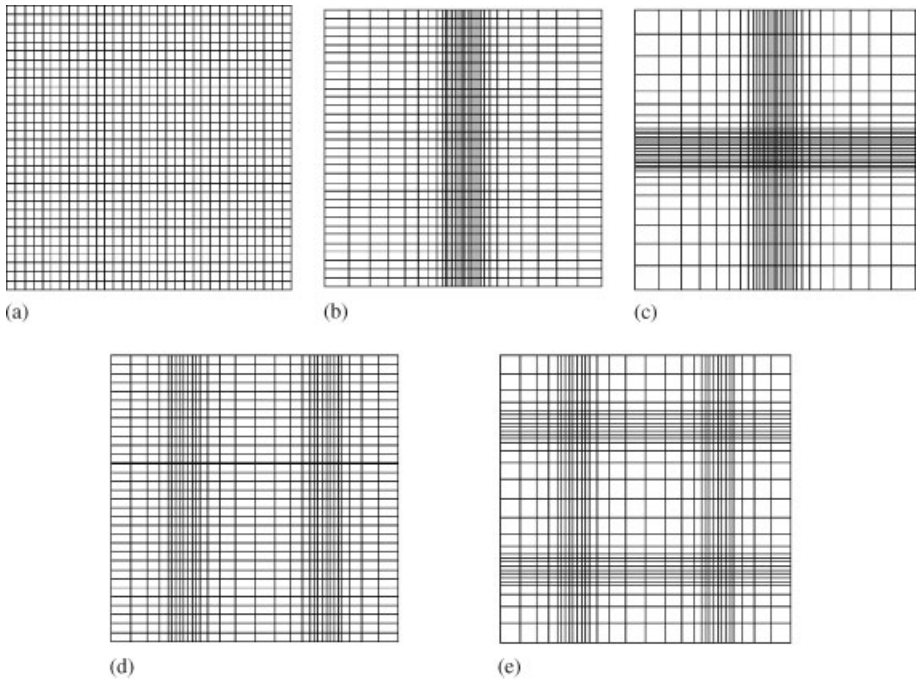

Figure 2. Structured isotropic mesh (a) and different locally anisotropic meshes (b)–(e) for the unit square.

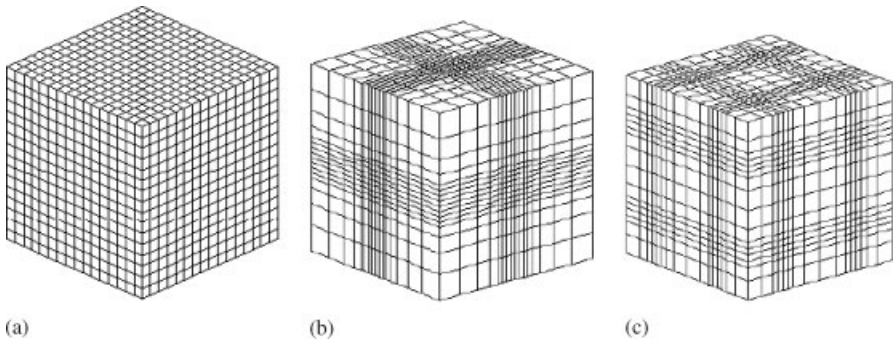

Figure 3. Structured isotropic mesh (a) and locally anisotropic meshes (b) and (c) for the unit cube.

the special properties of anisotropy only for mesh (b) in Figure 2. From the aforementioned formulas (**Fb**), we know that when $1\leqslant i\leqslant n_x/4$ or $n_x/4<i\leqslant n_x$, the maximal mesh ratio can be given as

$$R_y := \max_i \left(\frac{h^i_{x_b}}{1/n_y}\right) = \frac{(n_x/4+1)h^{\min}_{x_b}}{1/n_y} = \frac{4n_x+16}{n_x^2+5n_x}\cdot n_y \tag{6}$$

and when $n_x/4<i\leqslant 3n_x/4$, the corresponding maximal mesh ratio is given as

$$R_x := \max_i \left(\frac{1/n_y}{h^i_{x_b}}\right) = \frac{1/n_y}{h^{\min}_{x_b}} = \frac{n_x^2+5n_x}{16n_y} \tag{7}$$

Thus, if $n_x = O(n_y)$, it is easy to see that $R_y = O(1)$ and $R_x = O(n_x)$. Therefore, if $n_x \gg 1$, any element whose nodes belong to the local index set $\{(i, j)|n_x/4<i\leqslant 3n_x/4, 1\leqslant j\leqslant n_y\}$ is an x-type anisotropic element, and the rest of the elements are all the so-called isotropic elements. For convenience, we call this mesh a typical finite element mesh with locally strong anisotropy.

DOI: 10.1002/cnm

Analogously, meshes (c)–(e) in Figure 2 and meshes (b) and (c) in Figure 3 are also several typical finite element meshes with locally strong anisotropy in two and three dimensions, respectively. In what follows, we shall study the effect of both the locally strong anisotropic meshes presented in this section and the commonly used stretched meshes on the performance of several basic iterative solvers including CG and AMG methods by using some typical experiments in linear elasticity.

4. A COMMONLY USED AMG METHOD AND NUMERICAL RESULTS

AMG methods have been successfully applied to many linear systems corresponding to discretizations of the scalar PDE. However, most of these methods cannot be directly applied to the discrete equations arising from discretizations of systems of PDEs. In this section, we briefly introduce AMG01 method, which is referred as one commonly used AMG method for the finite element equations (4); see [14, 32] for more details. What we propose here is to develop a multigrid solver that may be called AMG method based on geometric and algebraic information.

First, using the coefficient matrix $\mathbf{A}=(\mathbf{A}_{ij})$ in the system of equations (5), we can define an auxiliary grid matrix $\overline{\mathbf{A}}=(\overline{a}_{ij})$ whose entries can be defined as

$$(\overline{a}_{ij}):=\begin{cases}0 & \text{if } \mathbf{A}_{ij}=\mathbf{0}\\ 1 & \text{if } \mathbf{A}_{ij}\neq\mathbf{0}\end{cases} \tag{8}$$

where the symbol '$\mathbf{0}$' denotes a $d\times d$ zero matrix. We consider the graph that corresponds to $\overline{\mathbf{A}}=(\overline{a}_{ij})$, namely $G(\overline{\mathbf{A}})=(V,E)$ in which V is the set of vertices and E is the set of edges. Associated with the graph $G(\overline{\mathbf{A}})$, a maximal independent set (MIS) (see [16, 33] for more details) of vertices can be obtained.

We can then construct the following two main ingredients of the multigrid method, namely, for $k=1,2,\ldots,M$, we shall discuss coarse grid selections and construction of prolongation operators, where M is the number of grid levels. Let $\mathbf{F}_1:=\mathbf{F}$, $\mathbf{U}_1:=\mathbf{U}$ and $G^1=(V^1,E^1)=G(\overline{\mathbf{A}})$, and assume that $\mathbf{A}_1:=\mathbf{A}$ and $\overline{\mathbf{A}}_1:=\overline{\mathbf{A}}$ denote the stiffness matrix and the grid matrix on the finest grid level, respectively. Algorithm 4.1 describes how to construct the aforementioned two main ingredients.

Algorithm 4.1 (*Coarsening and prolongation*)
The following procedures are performed starting with $\overline{\mathbf{A}}_k$ and $\mathbf{A}_k$ for $k=1$.

Step 1: Find the MIS of the graph $G^k=(V^k,E^k)$ and take it as the set of coarse points for the $(k+1)$th grid level.

Step 2: Construct the initial prolongation matrix $\mathbf{I}_{k+1}^k$ by using simple piecewise constant interpolation, namely

$$(\mathbf{I}_{k+1}^k v^{k+1})_i=\begin{cases}v_i^{k+1} & \text{if } i\in C^k\\ \dfrac{1}{m}\displaystyle\sum_{j\in C^k} v_j^{k+1} & \text{if } i\in F^k\end{cases} \tag{9}$$

where C^k and F^k are the sets of coarse and fine grid points, respectively, m is the number of coarse grid points adjacent to fine grid point i.

Step 3: Construct the initial full prolongation matrix $\mathbf{P}_{k+1}^k$ by the way given in Algorithm 4.2.

Step 4: Construct the full prolongation matrix $\hat{\mathbf{P}}_{k+1}^k$ from the initial full prolongation matrix $\mathbf{P}_{k+1}^k$ in the sense of minimal energy using the smoothing technique in [34].

Step 5: Construct the next coarser space matrix by the usual Galerkin approach, namely

$$\mathbf{A}_{k+1}=(\hat{\mathbf{P}}_{k+1}^k)^{\mathrm{T}}\mathbf{A}_k\hat{\mathbf{P}}_{k+1}^k \tag{10}$$

Step 6: Get the auxiliary grid matrix $\overline{\mathbf{A}}_{k+1}$ and the corresponding graph G^{k+1} with the definition as shown in (8), and set $k:=k+1$, if $k<M$, then go to *Step 1*; otherwise, stop.

DOI: 10.1002/cnm

Algorithm 4.2 (*Extending algorithm for the full prolongation in three dimensions*)
For $i=1,2,\ldots,n_k$
For $j=1,2,\ldots,n_{k+1}$
$$(\mathbf{P}_{k+1}^k)_{3i-2,3j-2}=(\mathbf{P}_{k+1}^k)_{3i-1,3j-1}=(\mathbf{P}_{k+1}^k)_{3i,3j}=(\mathbf{I}_{k+1}^k)_{i,j}$$
$$(\mathbf{P}_{k+1}^k)_{3i-2,3j-1}=(\mathbf{P}_{k+1}^k)_{3i-2,3j}=(\mathbf{P}_{k+1}^k)_{3i-1,3j-2}=(\mathbf{P}_{k+1}^k)_{3i-1,3j}$$
$$=(\mathbf{P}_{k+1}^k)_{3i,3j-2}=(\mathbf{P}_{k+1}^k)_{3i,3j-1}=0$$
End For
End For

Remark 4.1
Construction of the full prolongation matrix $\hat{\mathbf{P}}_{k+1}^k=(\hat{p}_{ij})$ from the initial full prolongation matrix $\mathbf{P}_{k+1}^k$ is made by solving the following optimization problem:

$$\hat{\mathbf{P}}_{k+1}^k=\arg\min_{\mathbf{q}} J(\mathbf{q})=\frac{1}{2}\sum_j \mathbf{q}_{*j}^{\mathrm{T}}\mathbf{A}_k\mathbf{q}_{*j} \tag{11}$$

where $\mathbf{q}_{*j}$ is the jth column of matrix $\mathbf{q}$. The constraints on the matrix $\mathbf{q}$ are (i) the sparsity pattern of $\mathbf{q}$ is the same as that of $\mathbf{P}_{k+1}^k$ and (ii) $\sum_j q_{ij}=1$ for all nodes i, for which i and its neighbors do not satisfy a Dirichlet boundary condition. The second constraint is the requirement that the coarse space basis reproduces a constraint function. Note that problem (11) is in general solved using some iterative methods with the full initial approximation of $\mathbf{P}_{k+1}^k$. This is a typical construction of a coarse grid basis function by minimization of its energy [34].

Having constructed all the components for the multigrid algorithm, we can get the AMG V-cycle iteration as usual for the system of equations (4). Note that the point-block Gauss–Seidel iteration is used as an effective smoother in our numerical experiments. For convenience, we denote by AMG01 the AMG V-cycle method with three pre- and three post-smoothing steps.

In what follows, we perform some experiments for studying the effect of anisotropic meshes on the performance of AMG01 and CG methods. Here, we mainly consider isotropic problems in linear elasticity on different anisotropic meshes presented in Figures 2 and 3.

We first describe our notation convention to be used to present the results of numerical experiments. In all the tables throughout this paper, N_{eqs} denotes the total number of unknowns; 'Cond' denotes the condition number of the stiffness matrix; N_{CG} denotes the number of iterations for the CG convergence, defined by $\|\mathbf{r}^k\|/\|\mathbf{r}^0\|\leqslant 10^{-6}$ in which $\mathbf{r^k}=\mathbf{F}-\mathbf{AU^k}$ is the residual vector at the kth iteration and the same iterative tolerance on the residual is used for AMG method; and ρ denotes the average convergence factor defined as

$$\rho:=\left(\frac{\|\mathbf{r}^{\mathrm{iter}}\|_{L_2}}{\|\mathbf{r}^0\|_{L_2}}\right)^{1/\mathrm{iter}} \tag{12}$$

where 'iter' denotes the number of iterations needed to achieve the desired stopping criteria. We choose $\mathbf{U}^0=\mathbf{0}$ as our initial guess. All the experiments throughout this paper have been performed on the computer with Interl(R) Xeon(TM) CPU 3.20 GHz and 1.00 GB RAM.

Example 1
In this example, we study the effect of those locally strong anisotropic meshes presented in Section 3 on the performance of AMG01 and CG methods for a discretization of (3) both on the unit square and on the unit cube with Young's modulus $E=1.0\times 10^6$ MPa and Poisson's ratio $v=0.25$ and with zero boundary conditions on all boundaries, where a constant body force $\mathbf{f}=2.78\times 10^6$ MPa. The corresponding numerical results are given in Tables I and II, respectively. Since the condition number of stiffness matrix becomes very large when the size of problems increases, CG method converges very slowly and is inefficient for large-scale problems.

From these results above, it can be seen that the AMG01 method is efficient only for those isotropic meshes, e.g. Figures 2(a) and 3(a), but this is not the case for those locally strong

DOI: 10.1002/cnm

Table I. Numerical results of AMG01 method for the two-dimensional problem of Example 1, where the symbol '*' denotes convergence for more than 1000 iterations.

	64×64, $N_{\text{eqs}}=7936$				128×128, $N_{\text{eqs}}=32258$			
Mesh	Cond	N_{CG}	ρ	Iter	Cond	N_{CG}	ρ	Iter
Figure 2(a)	1320	143	0.247	10	5276	291	0.351	14
Figure 2(b)	14061	532	0.805	64	108559	1511	0.930	191
Figure 2(c)	72385	634	0.965	386	733453	2045	0.994	*
Figure 2(d)	4950	387	0.580	26	34071	1076	0.808	65
Figure 2(e)	14138	438	0.867	87	125067	1363	0.966	404

Table II. Numerical results of AMG01 method for the three-dimensional problem of Example 1.

	$16\times16\times16$, $N_{\text{eqs}}=10125$				$32\times32\times32$, $N_{\text{eqs}}=89373$			
Mesh	Cond	N_{CG}	ρ	Iter	Cond	N_{CG}	ρ	Iter
Figure 3(a)	53	30	0.068	6	214	62	0.105	7
Figure 3(b)	1046	93	0.544	23	18421	390	0.899	131
Figure 3(c)	230	60	0.290	12	2606	223	0.790	59

anisotropic meshes, that is to say, the convergence rate of AMG01 method often deteriorates with increasing problem size.

In general, there are two ways that can improve AMG performance for such anisotropic problems. One way is to use semi-coarsening technique. The failure of a standard multigrid is that the errors in the direction of weak anisotropy are not smoothed. Thus, they cannot be solved on the coarse grid. In the case of structured grids, one can apply semi-coarsening, which selects coarse grid points geometrically according to the direction of anisotropy. The drawback, however, is that the overall cost of the multigrid will increase. Another way is to use block/line relaxation methods. However, for those locally anisotropic meshes, since the direction of anisotropy changes frequently, the determination of the lines or planes of anisotropy is usually complicated. In the following section, we hope to use the geometric grid information as nodal coordinates to construct a highly efficient AMG method for such anisotropic problems.

5. A MODIFIED AMG METHOD AND NUMERICAL RESULTS

The failure of the aforementioned AMG01 method is that it cannot deal with strong couplings between different physical unknowns only by using the auxiliary grid matrix $\overline{\mathbf{A}}=(\overline{a}_{ij})$ with the definition as shown in (8). We need to define again strong couplings between the blocks of the resulting stiffness matrix $\mathbf{A}=(\mathbf{A}_{ij})$, which correspond to a geometric grid point rather than couplings between single scalar (unknown) coefficients. An effective way is to use a condensation approach that can handle such coupling very efficiently. The basic idea is to use both the condensed point-block stiffness matrix according to References [10, 27] and the distance between neighboring grid points (stored in a distance matrix) in order to obtain a modified coarsening algorithm and define appropriate interpolation schemes. The details of this construction are given in the following.

We first define the auxiliary grid matrix $\overline{\mathbf{A}}:=(\overline{a}_{ij})$ by a condensation approach as the following form:

$$(\overline{a}_{ij}):=\begin{cases}0 & \text{if } \mathbf{A}_{ij}=\mathbf{0}\\ |||\mathbf{A}_{ij}||| & \text{if } \mathbf{A}_{ij}\neq\mathbf{0}\end{cases} \tag{13}$$

where $|||\mathbf{A}_{ij}|||$ is certain associated matrix-norm of matrix $\mathbf{A}_{ij}$. Note that there are several types of matrix-norms such as the L_2-norm, Frobenius-norm $|||\cdot|||_F$, the column-sum norm $|||\cdot|||_1$ and

DOI: 10.1002/cnm

the row-sum norm $|||\cdot|||_\infty$. From [27], we know that the row-sum norm $|||\cdot|||_\infty$ can provide the most favorable coarse grid results. Therefore, we also use this matrix-norm, i.e. $\bar{a}_{ij} := |||\mathbf{A}_{ij}|||_\infty$ if $\mathbf{A}_{ij} \neq \mathbf{0}$, for the condensation of the point blocks throughout this paper.

With the definition as shown in (13), we can then introduce the notion of strong couplings between grid points analogous to the scalar PDE case. Given a threshold of $\alpha, 0 \leqslant \alpha \leqslant 1$, we say the grid point i is strongly coupled to the grid point j if

$$|\bar{a}_{ij}| \geqslant \alpha \cdot \max_{k \in N_i} |\bar{a}_{ik}| \tag{14}$$

with α typically set to be 0.25, where $N_i := \{j | \bar{a}_{ij} \neq 0\}$. Note that one can uniquely map the index i to the geometric grid point $\mathbf{x}_i$, where the ith basis function of the discretization is nonzero on the grid point $\mathbf{x}_i$.

We define $S_i := \{j | |\bar{a}_{ij}| \geqslant \alpha \cdot \max_{k \in N_i} |\bar{a}_{ik}|\}$ as the set of all points j to which i is strongly coupled and $S_i^{\mathrm{T}} := \{j | i \in S_j\}$ as the set of all points, which are strongly coupled to i. With these definitions we can apply the classical coarse grid selection procedure to the condensed matrix $\overline{\mathbf{A}} := (\overline{a}_{ij})$. Details on the coarsening algorithm can be found in [10, 11, 35]. The initial prolongation matrix $\mathbf{I}_C^F$ is then defined by using the following simple interpolation formula, namely

$$(\mathbf{I}_C^F v_C)_i = \begin{cases} v_i & \text{if } i \in C \\ \dfrac{1}{m} \displaystyle\sum_{j \in C \cap S_i} v_j & \text{if } i \in F \end{cases} \tag{15}$$

where C and F are the sets of coarse and fine grid points, and $m = |C \cap S_i|$, which denotes the total number of elements in the set $C \cap S_i$. Finally, we can obtain the full prolongation matrix $\mathbf{P}$ as the way given in Algorithm 4.2 and the final prolongation matrix $\hat{\mathbf{P}}$ in the sense of minimal energy by using the smoothing technique in [34]. Furthermore, we can get all the components of multigrid iteration in a recursive manner. Here, we denote by AMG02 the corresponding AMG V-cycle method with three pre- and three post-smoothing steps. This method is applied to the system of equations (4) for the aforementioned Examples 1 and 2, respectively. Numerical results given in Tables III and V and Figure 6 have shown that AMG02 method is highly efficient for those thin and stretched meshes. This resembles the semi-coarsening of the method that can also be noticed for scalar problems. However, the convergence rate of AMG02 method gets worse with the increasing problem size for those locally strong anisotropic meshes. Therefore, it is necessary to develop some new techniques to improve the AMG convergence for such anisotropic problems.

AMG02 method selects coarse grid points algebraically according to strong couplings. It has better convergence for both isotropic meshes and stretched meshes, but converges very slowly for those locally strong anisotropic meshes as shown in Figures 2(b)–(e) and 3(b) and (c). Thus, how to define appropriate strong couplings is important for the selection of coarse grid points and for the improvement of the AMG efficiency. In what follows, we propose a modified algorithm of coarse grid selection. The basic idea of the algorithm is as follows. In the first step we divide all grid points into two classes. One consists of all isotropic grid points and another consists of all

Table III. Numerical results of AMG02 and AMG03 methods for the two-dimensional problem of Example 1, where '**' denotes convergence for more than 1000 iterations.

	AMG02						AMG03					
	64×64		128×128		256×256		64×64		128×128		256×256	
Mesh	ρ	Iter	ρ	Iter	ρ	Iter	ρ	Iter	ρ	Iter	ρ	Iter
Figure 2(a)	0.149	8	0.210	9	0.284	11	0.149	8	0.210	9	0.284	11
Figure 2(b)	0.723	43	0.865	96	0.947	255	0.188	9	0.304	12	0.608	28
Figure 2(c)	0.870	100	0.967	415	**	**	0.240	10	0.337	13	0.513	21
Figure 2(d)	0.695	38	0.853	88	0.940	225	0.248	10	0.269	11	0.395	15
Figure 2(e)	0.703	40	0.891	120	0.968	319	0.300	12	0.358	14	0.394	15

DOI: 10.1002/cnm

anisotropic grid points. Here, an anisotropic point is a point belonging to at least an anisotropic mesh element.

Then, in the second step we define an appropriate coarse grid selection, namely all isotropic grid points are automatically selected as coarse grid points and the rest coarse grid points are selected algebraically according to strong couplings for those anisotropic grid points. The above two steps are done recursively until all grid points are almost isotropic grid points in some measure. Finally, we take the MIS as the sets of coarser grid points and construct the corresponding prolongation operators and coarse grid matrices for the remainder grid levels.

For the convenience of discussion here, we introduce some definitions and notations. We begin by defining the distance matrix corresponding to the auxiliary grid matrix $\overline{\mathbf{A}}$ by using nodal coordinates.

Definition 5.1
The auxiliary matrix $\mathbf{D}=(d_{ij})$ is defined as the *distance matrix* corresponding to the auxiliary grid matrix $\overline{\mathbf{A}}$, whose entries are given by

$$d_{ij}=\begin{cases}\sqrt{(x_1(i)-x_1(j))^2+(x_2(i)-x_2(j))^2+(x_3(i)-x_3(j))^2} & \text{if } j\in N_i\\ 0 & \text{otherwise}\end{cases} \tag{16}$$

where $(x_1(i), x_2(i), x_3(i))$ is the nodal coordinate of the ith grid point, $i=1,2,\ldots,n_h$.

Let

$$d_i^{\min}=\min_{j\in N_i} d_{ij},\quad d_{\min}=\min_{1\leqslant i\leqslant n_h} d_i^{\min}\quad\text{and}\quad d_{\max}=\max_{1\leqslant i\leqslant n_h,\, j\in N_i} d_{ij}$$

We then define the set $S_i:=\{j\,|\,d_i^{\min}\leqslant d_{ij}\leqslant c\cdot d_i^{\min},\forall j\in N_i\}$ as the set of all grid points j to which i is strongly coupled.

Definition 5.2
The auxiliary matrix $\mathbf{S}=(s_{ij})$ is defined as the *relation matrix* corresponding to the auxiliary grid matrix $\overline{\mathbf{A}}$ if its entries s_{ij} satisfy

$$s_{ij}=\begin{cases}1 & \text{if } j\in S_i\\ 0 & \text{otherwise}\end{cases} \tag{17}$$

for any given index $i\in N^h$, where $N^h=\{1,2,\ldots,n_h\}$.

Assume that Ω_1 is the index set of all geometric grid points, $\overline{\mathbf{A}}_1:=\overline{\mathbf{A}}$ is the auxiliary grid matrix and $\mathbf{D}_1:=\mathbf{D}$ is the corresponding distance matrix on the finest grid level, respectively. Let $d_{\max}^{(1)}=d_{\max}$, $d_{\min}^{(1)}=d_{\min}$, $(d_i^{\min})^{(1)}=d_i^{\min}$ and $N_i^1=N_i$ $(i\in N^h)$. In this manner, we can present the modified algorithm for coarse grid selection as given in Algorithm 5.3.

Algorithm 5.3 (*Modified coarse grid selection procedure*)

Step 1: Set $k=1$.

Step 2: If the condition $d_{\max}^{(1)}\leqslant c^{k-1}\cdot d_{\min}^{(1)}$ is satisfied, then go to *Step 9*; otherwise, go to *Step 3*.

Step 3: Find the index set $T_k=\{i\,|\,d_{\min}^{(k)}\leqslant(d_i^{\min})^{(k)}\leqslant c\cdot d_{\min}^{(k)},\ i=1,2,\ldots,n_k\}$ by the distance matrix $\mathbf{D}_k$. Using the depth first search (DFS) algorithm, see [36, 37], we then divide the index set T_k into n_b connected subsets, which are denoted by $\Omega_{k,l}$, $l=1,2,\ldots,n_b$, respectively. For each connected subset $\Omega_{k,l}$, $l=1,2,\ldots,n_b$, perform the following two steps:

3.1. Generate the local distance matrix $\mathbf{D}_{k,l}$ and the corresponding relation matrix $\mathbf{S}_{k,l}$.
3.2. Apply the classical coarse grid selection procedure to the connected subset $\Omega_{k,l}$. Then denote by C_l^k and F_l^k the sets of coarse and fine grid points, respectively.

Step 4: Set $\Omega_{k,0}=\Omega_k-\bigcup_{l=1}^{n_b}\Omega_{k,l}$, then $C^k=(\bigcup_{l=1}^{n_b}C_l^k)\cup\Omega_{k,0}$ and $F^k=\Omega_k-C^k$. Here C^k and F^k denote the sets of coarse and fine grid points, respectively.

DOI: 10.1002/cnm

Step 5: Find the initial prolongation matrix $\mathbf{I}_{k+1}^{k}$ and the full prolongation matrix $\mathbf{P}_{k+1}^{k}$ by using Algorithm 5.4.

Step 6: Let $\Omega_{k+1}=C^k$ and take it as the set of all grid points at the $(k+1)$th level.

Step 7: Obtain the next coarse grid matrix $\mathbf{A}_{k+1}$ by the usual Galerkin approach (10) and then define the corresponding auxiliary grid matrix $\overline{\mathbf{A}}_{k+1}$.

Step 8: Save the nodal coordinates of the $(k+1)$th grid level and then generate the corresponding distance matrix $\mathbf{D}_{k+1}$. Let $d_{\min}^{(k+1)}=\min_{1\leqslant i\leqslant n_{k+1}}(d_i^{\min})^{(k+1)}$, where $(d_i^{\min})^{(k+1)}=\min_{j\in N_i^{k+1}} d_{ij}^{(k+1)}$. Set $k:=k+1$ and go to *Step 2*.

Step 9: Using Algorithm 4.1, construct the corresponding prolongation operators and coarse grid matrices for the remainder of grid levels, namely from the kth grid level to the Mth grid level, where level M is the coarsest grid level.

Remark 5.1

We assume that $d_{\max}^{(1)}=c^J\cdot d_{\min}^{(1)}$ for the finest grid level in Algorithm 5.3, and thus we have that $1\leqslant k\leqslant J$, where c is certain constant between 1.5 and 2.5 in general. For instance, c can be set to $\sqrt{2}$ for uniform quadrilateral mesh. In addition, the role of the distance matrix is to define again strong couplings between grid points and gradually decrease or eliminate the multi-scale property of the coefficient matrix on the coarse grid level or eliminate the locally strong anisotropy of meshes used in FEM discretization.

Remark 5.2

For certain grid level k, if the condition $d_{\max}^{(1)}\leqslant c^{k-1}\cdot d_{\min}^{(1)}$ is satisfied, it will imply that all grid points are almost isotropic ones. Thus, we can take the MIS of Ω_k as the set of coarse grid points.

Remark 5.3

The DFS algorithm is used here to solve the problem of testing whether certain graph is connected. The running time of DSF algorithm is $O(V+E)$, where V is a set whose elements are called vertices, and E is a set of unordered pairs of distinct elements of set V.

Figures 4 and 5 show two examples of the choice of coarse grid points by using the modified coarse grid selection procedure based on distance matrix. The result in Figure 4 for the usual thin and stretched meshes is essentially identical to that of semi-coarsening. For the locally anisotropic mesh as shown in Figure 5, it needs to perform only twice recursively in order to make all grid points almost isotropic ones, and thus the MIS can be taken as the sets of coarser grid points for the remainder grid levels.

The following algorithm will present the construction of the modified prolongation operator, which is used in Step 5 of Algorithm 5.3.

Algorithm 5.4 (*Modified prolongation*)

Step 1: Define the initial prolongation matrix $\mathbf{I}_{k+1}^{k}$ by using the modified simple piecewise constant interpolation, namely

$$(\mathbf{I}_{k+1}^{k}v^{k+1})_i=\begin{cases} v_i^{k+1} & \text{if } i\in C^k \\ \dfrac{1}{m}\displaystyle\sum_{j\in S_i} v_j^{k+1} & \text{if } i\in F^k \end{cases} \tag{18}$$

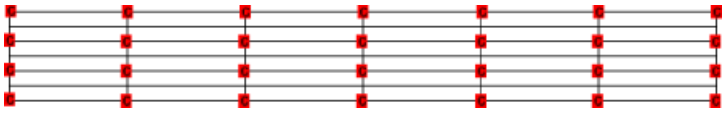

Figure 4. An illustration for the usual thin and stretched mesh, where the coarse nodes are denoted by 'c' and the rest are fine nodes.

DOI: 10.1002/cnm

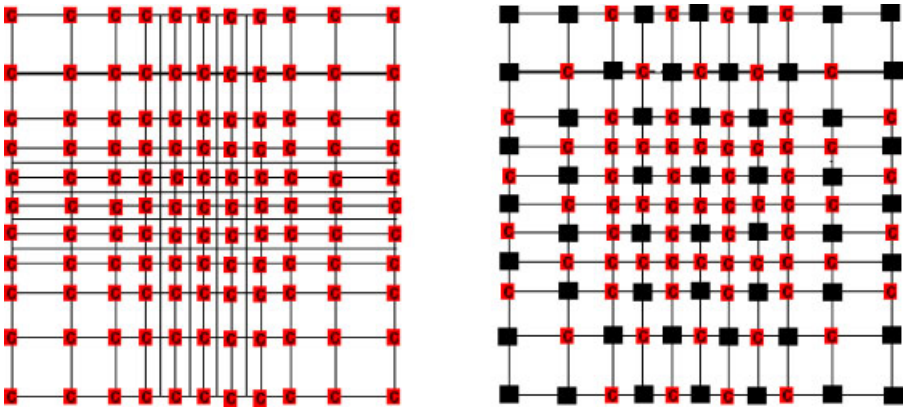

Figure 5. Sequence of the modified coarse grid selection process on a locally anisotropic mesh. Left: Result of coarse grid selection for the finest mesh, where the coarse nodes are denoted by 'c' and the rest are fine nodes. Right: Result of coarse grid selection for the next coarser mesh, where the coarse nodes are denoted by '■' and the rest are fine nodes.

Table IV. Numerical results of AMG02 and AMG03 methods for the three-dimensional problem of Example 1.

	AMG02						AMG03					
	$16\times16\times16$		$32\times32\times32$		$48\times36\times36$		$16\times16\times16$		$32\times32\times32$		$48\times36\times36$	
Mesh	ρ	Iter	ρ	Iter	ρ	Iter	ρ	Iter	ρ	Iter	ρ	Iter
Figure 3(a)	0.025	4	0.040	5	0.043	5	0.025	4	0.041	5	0.043	5
Figure 3(b)	0.518	21	0.844	82	0.924	115	0.092	6	0.150	8	0.15	9
Figure 3(c)	0.426	17	0.795	76	0.876	94	0.098	6	0.152	8	0.160	9

where C^k and F^k are the sets of coarse and fine grid points, respectively, and S_i denotes the set of all grid points j to which i is strongly coupled. Here $m=|S_i|$, which denotes the total number of elements in set S_i.

Step 2: Construct the initial full prolongation matrix $\mathbf{P}_{k+1}^k$ by the way given in Algorithm 4.2.

Step 3: Obtain the full prolongation matrix $\hat{\mathbf{P}}_{k+1}^k$ from the initial full prolongation matrix $\mathbf{P}_{k+1}^k$ in the sense of minimal energy using the smoothing technique given in [34].

So far, we can obtain all the components for AMG V-cycle iteration for the system of equations (4). Here, we also denote this method by AMG03 with three pre- and three post-smoothing steps as usual. In what follows, we perform some numerical experiments to test the effectiveness and robustness of our constructed AMG method.

We first use our constructed AMG03 method to solve the aforementioned problems presented in Example 1. Note that we always set $c=1.6$ throughout this paper. The corresponding numerical results are given in Tables III and IV, respectively.

Example 2

This example is mainly used to illustrate the robustness of our constructed AMG method with respect to small aspect ratios of the elements. It is also used in Reference [10] as an example. We discretize the two-dimensional problem on a rectangular domain with the mesh size h_y in the y-direction and $h_x=l\cdot h_y$ in the x-direction, where $l>0$ is the stretching factor of the mesh as shown in Figure 6. In three-space dimensions we distinguish the two cases

$$\Omega=[0,l]\times[0,1]^2 \quad \text{and} \quad h_x=l\cdot h_y(=h_z) \tag{C1}$$

$$\Omega=[0,l]^2\times[0,1] \quad \text{and} \quad h_x=h_y=l\cdot h_z \tag{C2}$$

DOI: 10.1002/cnm

Figure 6. Sketch of the mesh structure used in Example 2 in two dimensions.

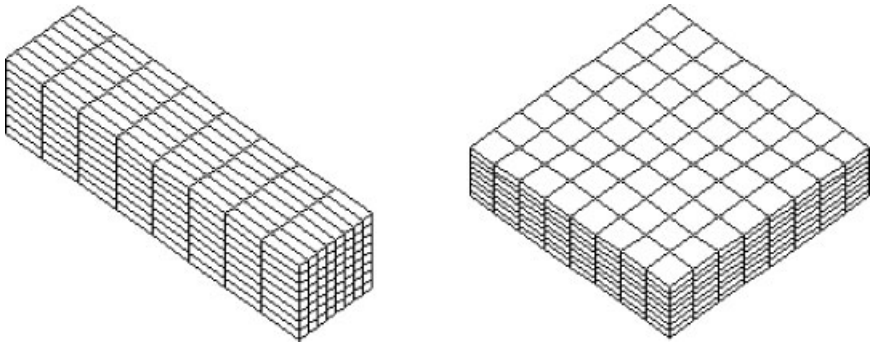

Figure 7. Two different domains for the three-dimensional problem of Example 2.

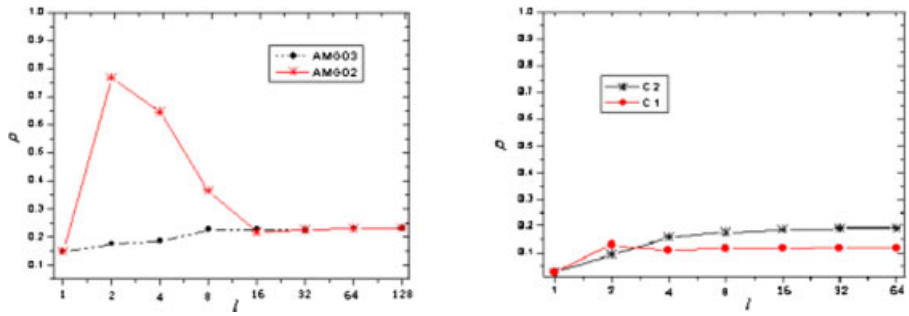

Figure 8. Left: Average convergence factor ρ of AMG02 and AMG03 methods for Example 2 for different l in two dimensions with $h=\frac{1}{64}$. Right: Average convergence factor ρ of AMG03 for Example 2 for different l in three dimensions with $h=\frac{1}{16}$ for problem types (C1) and (C2).

as shown in Figure 7. The same material parameters and Dirichlet boundary conditions and the body force as given in Example 1 are used in this experiment. The corresponding numerical results are presented in Figure 8.

Example 3

In this example, we consider again the same problems as Example 1. However, they are now discretized on non-uniform and unstructured meshes for further testing the effectiveness and robustness of our constructed AMG method.

Case 1: We discretize the two-dimensional problem on non-uniform meshes of quadrilateral elements (that are uniformly refined), one of which is shown in Figure 9, which contains some elements with small aspect ratios. The corresponding iteration counts for different mesh sizes are presented in Table V.

Case 2: We discretize the problems on unstructured meshes with different sizes. The main point here is that the constructed AMG method can deal effectively with unstructured meshes without too much degradation in convergence over the uniform case. Several isotropic meshes with 2288, 8832 and 13 376 quadrilateral elements and some locally strong anisotropic meshes with 960, 3840 and 15 360 quadrilateral elements in two dimensions are used in this numerical test. Two typical

DOI: 10.1002/cnm

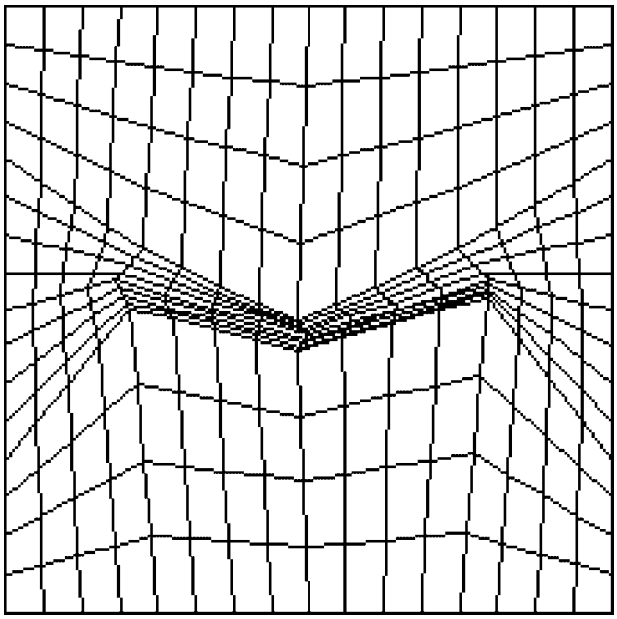

Figure 9. A typical non-uniform mesh used in Case 1 of Example 3.

Table V. Results of AMG02 and AMG03 on non-uniform meshes of quadrilateral elements.

	16×16		32×32		64×64		128×128	
	ρ	Iter	ρ	Iter	ρ	Iter	ρ	Iter
AMG02	0.695	38	0.780	56	0.855	89	0.844	82
AMG03	0.118	7	0.213	9	0.244	10	0.326	12

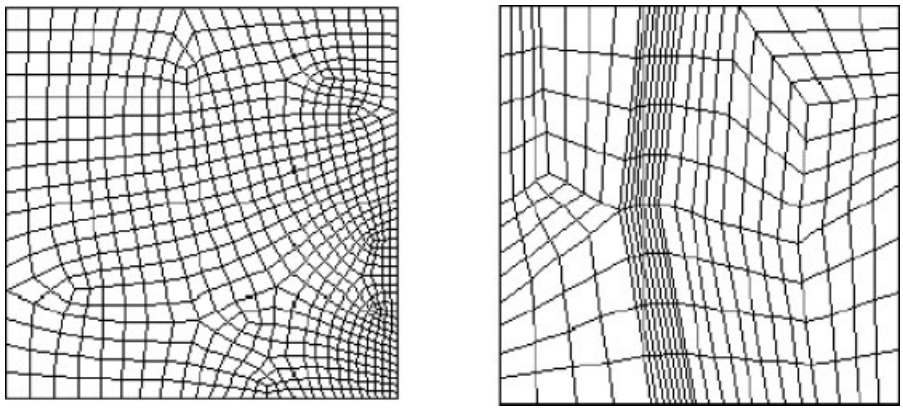

Figure 10. Two typical unstructured meshes (left: isotropic mesh; right: anisotropic mesh) in two dimensions used in Case 2 of Example 3.

examples are shown in Figure 10, respectively. The corresponding results of AMG02 and AMG03 methods are presented in Tables VI and VII, respectively.

Example 4

In many situations, we need to use a discretization on adaptive meshes, where the mesh width can vary substantially over the domain. In this example, we consider the same two-dimensional

DOI: 10.1002/cnm

Table VI. Results of AMG02 and AMG03 on different isotropic unstructured meshes.

	2288		8832		13 376	
No. of elements	ρ	Iter	ρ	Iter	ρ	Iter
AMG02	0.342	13	0.470	19	0.498	20
AMG03	0.365	14	0.461	18	0.511	21

Table VII. Results of AMG02 and AMG03 on different locally strong anisotropic unstructured meshes.

	960		3840		15 360	
No. of elements	ρ	Iter	ρ	Iter	ρ	Iter
AMG02	0.740	46	0.768	53	0.850	86
AMG03	0.419	16	0.438	17	0.537	23

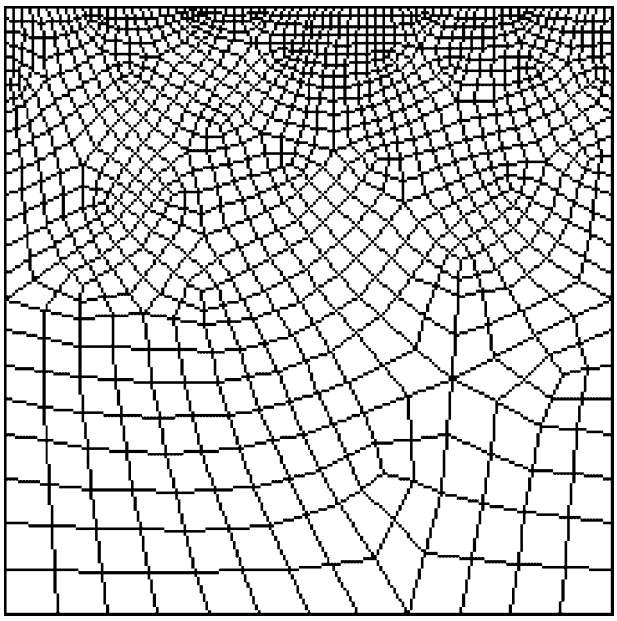

Figure 11. A locally refined mesh with 964 elements used in Example 4.

Table VIII. Results of AMG02 and AMG03 methods on different adaptive meshes.

	1348		8160		12 672	
No. of elements	ρ	Iter	ρ	Iter	ρ	Iter
AMG02	0.352	14	0.528	22	0.500	20
AMG03	0.387	15	0.494	20	0.511	21

problems as Example 1 once more, which are now discretized on some adaptive meshes (an example of such a mesh is given in Figure 11) to show that our constructed AMG method can also provide fast convergence for such discretizations. The results of AMG02 and AMG03 methods, presented in Table VIII, are similar to those obtained for the previous two-dimensional problems discretized on different isotropic unstructured meshes of quadrilateral elements.

DOI: 10.1002/cnm

From these results we can draw some conclusions as follows:

(1) AMG03 and AMG02 methods nearly have the same convergence rate for isotropic meshes including both structured and unstructured meshes.
(2) The constructed AMG03 method has faster convergence than AMG02 method for those locally strong anisotropic meshes including both structured and unstructured meshes. For example, the average convergence factor obtained with AMG02 method is $\rho=0.967$ for $h=\frac{1}{128}$, whereas we find $\rho=0.337$ for AMG03 method for the two-dimensional problems discretized as depicted in Figure 2(c). In three dimensions, for the mesh as shown in Figure 3(b) our modified AMG method converges with an average convergence factor $\rho=0.150$ for $h=\frac{1}{32}$, whereas the AMG02 method provides approximately one-tenth of the speed of convergence, compared with $\rho=0.844$. From Table IV one can see that the number of iterations is reduced from 82 to 8 in this case. For the two-dimensional problems discretized on unstructured anisotropic meshes as depicted in Figure 10 (right), the average convergence factor of AMG02 method is $\rho=0.850$ for the mesh with 15 360 elements, whereas $\rho=0.537$ for AMG03 method.
(3) For those problems discretized on the usual thin and stretched meshes, both methods have the same convergence rate when the parameter $l\geqslant 16$ in two dimensions, but AMG03 has better convergence than AMG02 method for $1\leqslant l<16$. In three dimensions, AMG03 gives better results with $\rho<0.20$ for the two different problem types (C1) and (C2).

In conclusion, the modified AMG method is often robust and efficient not only for the usual thin and stretched meshes but also for those locally anisotropic meshes including structured and unstructured meshes. Although the convergence rate of the resulting AMG method is perhaps weakly dependent on the mesh size in some situations, the numerical results obtained here are still quite encouraging.

In what follows, we shall make further numerical tests for the solution of some important problems that occur in many practical applications.

Example 5 (*Jumps in Young's modulus, see, e.g. Griebel et al.* [*10*] *and Brezina et al.* [*38*])
In practice, we often encounter elastic structures that are made of many different materials with different properties. However, this leads to jumps in the coefficients of the elasticity operator and makes the solution of the respective linear system even harder. In this example, we examine such problems in two- and three-space dimensions, respectively.

Consider the decomposition of the domain into four subdomains for the two-dimensional case and into eight subdomains for the three-dimensional case as shown in Figure 12, respectively. We set Poisson's ratio $\nu=0.25$ and Young's modulus E to $E_2=1.0\times 10^6$ MPa on the light subdomains and to $E=E_1\gg E_2$ on the dark subdomains. The body force $\mathbf{f}=2.78\times 10^6$ MPa is also used in this example.

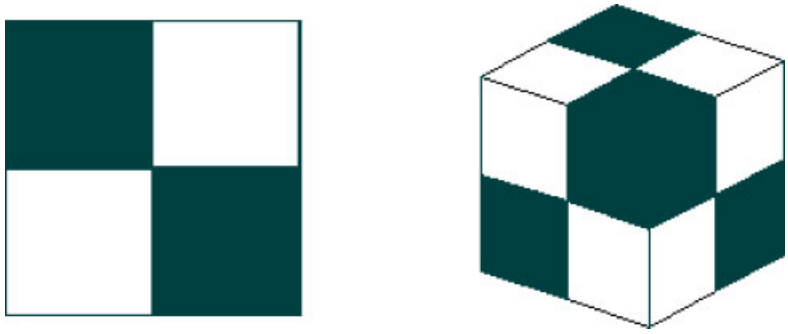

Figure 12. Jumps in Young's modulus, namely $E=E_1$ on dark subdomains and $E=E_2=1.0\times 10^6<E_1$ on light subdomains.

DOI: 10.1002/cnm

Table IX. Numerical results of AMG02 and AMG03 methods for the two-dimensional problem of Example 5.

	AMG02				AMG03			
	64×64		128×128		64×64		128×128	
E_1/E_2	ρ	Iter	ρ	Iter	ρ	Iter	ρ	Iter
10	0.914	154	0.977	583	0.546	23	0.630	30
100	0.915	157	0.977	595	0.557	24	0.627	30
1000	0.905	139	0.974	530	0.483	19	0.566	25

Table X. Numerical results of AMG02 and AMG03 methods for the three-dimensional problem of Example 5.

	AMG02						AMG03					
	$8\times8\times8$		$16\times16\times16$		$32\times32\times32$		$8\times8\times8$		$16\times16\times16$		$32\times32\times32$	
E_1/E_2	ρ	Iter	ρ	Iter	ρ	Iter	ρ	Iter	ρ	Iter	ρ	Iter
10	0.176	8	0.573	25	0.893	123	0.086	6	0.133	7	0.387	15
100	0.208	9	0.620	29	0.912	150	0.111	7	0.203	9	0.453	18
1000	0.213	9	0.648	32	0.915	156	0.098	6	0.193	9	0.427	17

Linear elasticity problem (3) has been discretized using bilinear and trilinear finite element and zero Dirichlet boundary conditions are used on all boundaries. The corresponding meshes needed are shown in Figures 2(c) and 3(b), respectively, where the locally refined meshes are used near the interface between different materials. Results of experiments are summed up in Tables IX and X. It is shown from these results that the constructed AMG03 method has substantially faster convergence than AMG02 method and it achieves the desired robustness with respect to the jumps of Young's modulus. However, it would be nice to have some numerical results included (which show the number of iterations) for different coefficient jumps, e.g. for $E_1/E_2=1$, 10, 100, 1000.

Remark 5.4

The constructed algorithm is still somewhat dependent on the problem size for linear elasticity problems with jumps in coefficients. In general, such PDEs are difficult to solve by current (algebraic) multigrid methods and it is a challenging task to come up with a method that works robustly in all practical situations. The performance of the resulting AMG method can be partially improved, for instance, by using the interface preserving coarsening procedure, which is proposed in the recent work [19].

Note that in the previous examples, we consider only homogeneous Dirichlet conditions on all boundaries. In practice, there are also Neumann boundary conditions or simply without any conditions on some parts of the boundary. In the final example, we take an elasticity problem with some free boundaries in two dimensions as an illustration to numerically demonstrate the robustness and efficiency of our constructed AMG method.

Example 6 (*Beam problem with free boundaries, see Graham and Forsyth* [*39*])

This problem consists of a rectangular beam of variable thickness attached at each end to a wall, hanging under gravity as shown in Figure 13. The beam is held in tension. Its dimensions are $0.6\,\mathrm{m}\times0.15\delta\,\mathrm{m}$ in the x- and y-directions, respectively, and it is modelled on a 64×64 mesh. The material parameters $E=100\,\mathrm{MPa}$ and $v=0.4$. In our numerical experiment, $\delta=1, \frac{1}{2}, \frac{1}{4}, \frac{1}{8}$ and $\frac{1}{16}$ are used to demonstrate the effects of small element aspect ratio on the convergence of AMG methods. A depiction of such a mesh is shown in Figure 6.

The boundary conditions are designed to simulate the effects of the weight of the beam. A downward force is prescribed at the center of the underside of the beam (the center of gravity) and the nodes adjacent to the left-hand side wall and to the right-hand side wall are fixed.

DOI: 10.1002/cnm

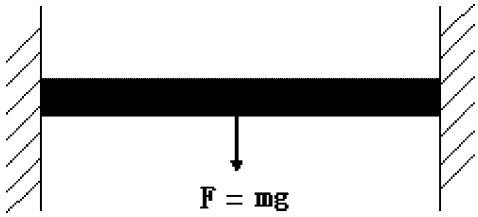

Figure 13. Beam hanging under its own weight [39].

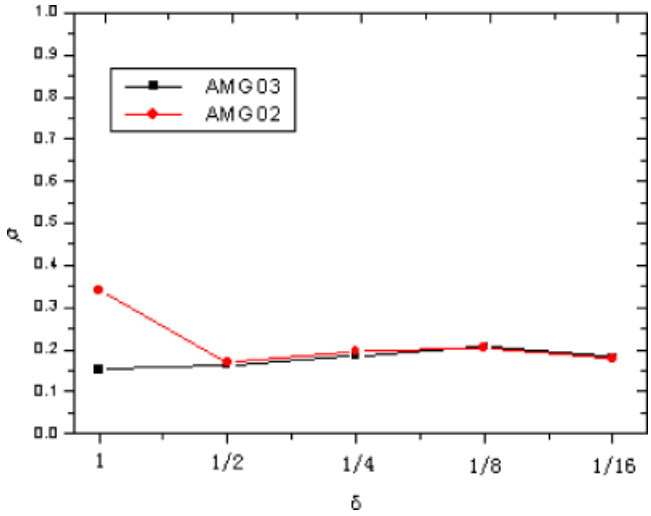

Figure 14. Average convergence factor ρ of AMG02 and AMG03 methods for different aspect ratios δ.

We find that the convergence of AMG03 method deteriorates with δ decreasing (the corresponding results are not presented here). One reason is that the selection of points on two free boundaries for the coarse grids and the constructed interpolation does not reproduce the rigid body modes at such points exactly [10]. One way to overcome this problem is to develop a coarsening algorithm, which provides a standard coarsening also for free boundary points. Since the modified coarsening Algorithm 5.3 can produce good results for interior points in the previous examples, it is natural to use a standard coarsening in the first step to coarsen all free boundary points separately and then, in the second step, to coarsen the interior points using the modified coarsening Algorithm 5.3.

From [10], we know that a point i belongs to the boundary, if it satisfies

$$|N_i| < \frac{\sum_{j \in N_i} |N_j|}{|N_i|} \tag{19}$$

Thus, we can also get AMG03 method based on the coarsening processes as given above. The corresponding numerical results for different aspect ratios δ are presented in Figure 14. It is shown that the AMG03 method has the same convergence rates nearly as the AMG02 method when δ is decreasing. Therefore, the constructed AMG method in this paper is also robust and efficient for elasticity problems with free boundaries. Note that in AMG02 method, semi-coarsening was performed until there is at least one interior point.

6. CONCLUDING REMARKS

The aim of this paper has been to construct a more efficient and robust AMG solver for the solution of linear elasticity problems on several types of anisotropic meshes. In particular, a

DOI: 10.1002/cnm

modified algorithm of coarse grid selection is proposed by using geometric information and analytic properties of the underlying systems of elasticity PDEs. We have presented various numerical results of the constructed AMG method for some typical problems in two- and three-dimensional linear elasticity. It has been shown that the resulting AMG method is often robust and efficient not only for the usual thin and stretched meshes but also for those locally anisotropic meshes including structured and unstructured meshes.

Our method is less algebraic or generic than the classical AMG in the sense that we need to know *a priori* geometric grid information as nodal coordinates on the finest grid level. Although only limited numerical experiments have been performed, the numerical results are still quite encouraging. More extensive numerical experiments for general problems and developing other techniques to obtain better AMG convergence rate are also expected. Such areas will be a subject of our future research.

ACKNOWLEDGEMENTS

The first author would like to thank Dr Desheng Wang for his helpful discussions on the numerical experiments. This work was supported in part by NSF-10771178, NSF-10672138 and NSAF-10676031 in China, the Basic Research Program of China under the grant 2005CB321702, the Key Project of Chinese Ministry of Education and the Scientific Research Fund of Hunan Provincial Education Department (208093 and 07A068) and the Provincial Natural Science Foundation of Hunan (07JJ6004).

REFERENCES

1. Senturia S, Aluru N, White J. Simulating the behavior of MEMS devices: computational methods and needs. *IEEE Computational Science and Engineering* 1997; **4**(1):30–43.
2. Senturia S, Harris R, Johnson B, Songmin K, Nabors K, Shulman M, White J. A computer-aided design system for microelectromechanical systems (MEMCAD). *Journal of Microelectromechanical Systems* 1992; **1**:3–13.
3. Picasso M. A stopping criterion for the conjugate gradient algorithm in the framework of anisotropic adaptive finite elements. *Communications in Numerical Methods in Engineering*. DOI: 10.1002/cnm.1120.
4. Mavriplis D. Directional coarsening and smoothing for anisotropic Navier–Stokes problems. *Electronic Transactions on Numerical Analysis* 1997; **6**:182–197.
5. Mavriplis D. Multigrid strategies for viscous flow solvers on anisotropic unstructured meshes. *Journal of Computational Physics* 1998; **145**(1):141–165.
6. Parsons ID, Hall JF. The multigrid method in solid mechanics. *International Journal for Numerical Methods in Engineering* 1990; **29**:719–754.
7. Brandt A. Guide to multigrid development. In *Multigrid Methods*, Hackbusch W, Trottenberg U (eds). Springer: Berlin, 1982; 220–312.
8. Liu CQ. *Multigrid Method and its Application in Computational Fluid Mechanics*. Tsinghua University Press: Beijing, 1995.
9. Mohamed SA. Adaptive FAS-multigrid method for nonlinear elliptic equations on unstructured grids. *Communications in Numerical Methods in Engineering*. DOI: 10.1002/cnm.1089.
10. Griebel M, Oeltz D, Schweitzer MA. An algebraic multigrid method for linear elasticity. *SIAM Journal on Scientific Computing* 2003; **25**(3):85–407.
11. Ruge JW, Stüben K. Algebraic multigrid. In *Multigrid Methods*, McCormick S (ed.). SIAM: Philadelphia, PA, 1987; 73–130.
12. Stüben K. An review of algebraic multigrid. *Journal of Computational and Applied Mathematics* 2001; **128** (1–2):281–309.
13. Chang Q, Wong Y, Fu H. On algebraic multigrid method. *Journal of Computational Physics* 1996; **125**:279–292.
14. Xiao YX, Shu S, Zhang P, Mo Z, Xu J. A kind of semi-coarsening AMG method for two dimensional energy equations with three temperatures. *Journal on Numerical Methods and Computer Applications* 2003; **24**(4):293–303.
15. Brezina M, Cleary AJ, Falgout RD, Henson VE, Jones JE, Manteuffel TA, McCormick SF, Ruge JW. Algebraic multigrid based on element interpolation (AMGe). *SIAM Journal on Scientific Computing* 2000; **22**:1570–1592.
16. Xiao YX, Zhang P, Shu S. An algebraic multigrid method with interpolation reproducing rigid body modes for semi-definite problems in two-dimensional linear elasticity. *Journal of Computational and Applied Mathematics* 2007; **200**(2):637–652.
17. Jones J, Vassilevski P. AMGe based on element agglomeration. *SIAM Journal on Scientific Computing* 2001; **23**(1):109–133.
18. Sumant PS, Cangellaris AC, Aluru NR. A node-based agglomeration AMG solver for linear elasticity in thin bodies. *Communications in Numerical Methods in Engineering*. DOI: 10.1002/cnm.1116.

19. Xiao YX, Zhang P, Shu S. Algebraic multigrid methods for elastic structures with highly discontinuous coefficients. *Mathematics and Computers in Simulation* 2007; **76**(4):249–262.
20. Dendy J, Ida M, Rutledge J. A semi-coarsening multigrid algorithm for SIMD machines. *SIAM Journal on Scientific and Statistical Computing* 1992; **13**:1460–1469.
21. Dendy JE, McCormick SF, Ruge JW, Russell TF, Schaffer S. Multigrid methods for three-dimensional petroleum reservoir simulation. *Tenth Symposium on Reservoir Simulation*, vols 6–8. Society of Petroleum Engineers (SPE): Houston, 1989; 19–25.
22. Smith RA, Weiser A. Semi-coarsening multigrid on a hypercube. *SIAM Journal on Scientific and Statistical Computing* 1992; **13**:1314–1329.
23. Hackbusch W. *Multi-grid Methods and Applications*. Springer: New York, 1985.
24. Brandt A. Multi-level adaptive solutions to boundary-value problems. *Mathematics of Computation* 1977; **33**: 333–390.
25. Berzins M. A solution-based triangular and tetrahedral mesh quality indicator. *SIAM Journal on Scientific Computing* 1998; **19**:2051–2069.
26. Du Q, Huang Z, Wang D. Mesh and solver coadaptation in finite element methods for anisotropic problems. *Numerical Methods for Differential Equations* 2005; **21**:859–874.
27. Oeltz D. Algebraische Mehrgittermethoden füur Systeme partieller Differentialgleichungen. *Master Thesis*, Institut für Angewandte Mathematik, Universität Bonn, 2001.
28. Braess D. *Finite Elements*: *Theory*, *Fast Solvers and Applications in Solid Mechanics*. Cambridge University Press: Cambridge, MA, 2001.
29. Brezzi F, Fortin M. *Mixed and Hybrid Finite Element Methods*. Springer: Berlin, 1991.
30. Ciarlet P. *The Finite Element Method for Elliptic Problems*. North-Holland: Amsterdam, 1978.
31. Saad Y. *Iterative Methods for Sparse Linear Systems*. PWS Publishing Company: Boston, MA, 1996.
32. Cheikh NB, Beya BB, Li TL. A multigrid method for solving the Navier–Stokes/Boussinesq equations. *Communications in Numerical Methods in Engineering*. DOI: 10.1002/cnm.980.
33. Chan TF, Xu J, Zikatanov L. An agglomeration multigrid method for unstructured grids. *Contemporary Mathematics* 1998; **218**:67–81.
34. Mandel J, Brezina M, Vanek P. Energy optimization of algebraic multigrid bases. *Computing* 1999; **62**:205–228.
35. Ruge JW, Stüben K. Efficient solution of finite difference and finite element equations by algebraic multigrid. In *Multigrid Methods for Integral and Differential Equations*, Paddon DJ, Holstein H (eds). The Institute of Mathematics and its Applications Conference Series. Clarendon Press: Oxford, 1985.
36. Tarjan R. Depth-first search and linear graph algorithms. *SIAM Journal on Computing* 1972; **1**:146–159.
37. Depth First Search (DFS). Available from: http: www.personal.kent.edu/∼rmuhamma/algorithms/MyAlgorithms/GraphAlgor/depthSearch.htm.
38. Brezina M, Vanek P, Tezaur R. Two-grid method for linear elasticity on unstructured meshes. *SIAM Journal on Scientific Computing* 1999; **21**:900–923.
39. Graham E, Forsyth PA. Preconditioning methods for very ill-conditioned three dimensional linear elasticity problems. *International Journal for Numerical Methods in Engineering* 1999; **44**:77–99.