COMMUNICATIONS IN NUMERICAL METHODS IN ENGINEERING
Commun. Numer. Meth. Engng 2000; **16**:1–12 *Prepared using* cnmauth.cls *[Version: 2000/03/22 v1.0]*

Component-wise algebraic multigrid preconditioning for the iterative solution of stress analysis problems from microfabrication technology

S. Mijalković[1] and M. D. Mihajlović[2]

[1] *Faculty of Information Technology and Systems, Delft University of Technology, Delft, The Netherlands*
[2] *Centre for Novel Computing, Department of Computer Science, The University of Manchester, Manchester M13 9PL, UK*

SUMMARY

A methodology for preconditioning discrete stress analysis systems using robust scalar algebraic multigrid (AMG) solvers is evaluated in the context of problems that arise in microfabrication technology. The principle idea is to apply an AMG solver in a segregated way to the series of scalar block matrix problems corresponding to different displacement vector components, thus yielding a block diagonal AMG preconditioner. We study the component-wise AMG preconditioning in the context of the space decomposition and subspace correction framework [22]. The subspace problems are solved approximately by the scalar AMG solver and the subspace correction is performed either in block diagonal (block Jacobi) or lower triangular (block Gauss–Seidel) fashion. In our test examples we use fully unstructured grids of different sizes. The numerical experiments show robust and efficient convergence of the Krylov iterative methods with component-wise AMG preconditioning for both 2D and 3D problems. Copyright © 2000 John Wiley & Sons, Ltd.

KEY WORDS: stress analysis; linear elasticity; algebraic multigrid; preconditioning; Krylov methods

1. INTRODUCTION

Finite element stress analysis in industrial applications typically involves problems with complex geometries and employs general unstructured grids in the discretisation. This leads to the solution of large, sparse systems of algebraic equations, where the coefficient matrix has a highly non-regular sparsity pattern. It is recognised that, as computing power continues to increase, enabling larger and more complex applications to be tackled, the solver performance for the corresponding discrete problems becomes much poorer. That is, while the direct solvers are requiring unacceptable

*Correspondence to: S. Mijalković, Department of Microelectronics/ECTM Laboratory-DIMES, Faculty of Information Technology and Systems, Delft University of Technology, Mekelweg 4, 2628 CD Delft, The Netherlands. E-mail: S.Mijalkovic@ITS.TUDelft.NL

Contract/grant sponsor: Dutch Technology Foundation (STW); contract/grant number: DEL 4415

Contract/grant sponsor: The University of Manchester

Received 2 August 2000

computational resources (together with a deterioration of asymptotic complexity and MFlops rate when the problem size increases), the standard iterative solvers have the problems of ill-conditioning.

It is well known that multigrid methods offer the prospect of optimal scaling with problem size, both in terms of asymptotic complexity and memory requirements [9]. However, multigrid methods require a hierarchical grid structure, which is not readily available in unstructured grids. In order to overcome this problem, the idea of algebraic multigrid has been introduced in [18]. In AMG methods, the required discrete hierarchy is generated in the automatic coarsening process, based exclusively on algebraic relationships between the discrete variables. However, the coarsening heuristics of the standard AMG algorithm was essentially developed in the context of a stiffness matrix corresponding to scalar elliptic PDEs, and it is generally not directly applicable to non-scalar PDE systems. Although important progress has been achieved in straightforward applications of the AMG algorithm to some systems of PDEs (see, for example, [19] for some general remarks, and [5] for experiences in stress analysis problems), the well-established formal approach capable of exhibiting the robustness and efficiency of the original scalar AMG does not exist (see [19]).

An alternative to adapting the scalar AMG coarsening strategy to non-scalar stress PDEs, is to decompose the global matrix problem into a sequence of scalar problems suitable for the standard AMG algorithm. Such a decomposition is most naturally associated with the displacement vector components, which results in a sequence of block diagonal matrix subproblems. This problem splitting also fits into the abstract framework of space decomposition and subspace correction methods for preconditioning of general sparse linear algebraic systems, that is introduced in [22]. The effectiveness of block diagonal preconditioning is justified in [3] for elasticity problems, where approximate factorisation methods are used for the solution of individual block matrices. For the application and efficiency of block diagonal preconditioning in the context of some other problems (including fluid mechanics), see [1, 2, 13, 15, 16, 21] and references contained therein. In the problem that we consider here the standard AMG coarsening procedure is suitable for the component-wise block diagonal matrices, these subproblems can be submitted, either independently or successively, to robust scalar AMG solvers. The idea of using either MG or AMG applied as a block diagonal preconditioner for the Krylov solvers has been used in various contexts (see [1, 2, 11, 15, 20, 21] and references contained therein), but, to the best of our knowledge, it has not been applied in the context of stress analysis problems arising in microfabrication technology.

The main goal of this paper is to demonstrate experimentally the effectiveness of this preconditioning methodology in the case of stress analysis problems in 2D and 3D from microfabrication technology. Theoretical justification and spectral analysis of the proposed preconditioning methodology may be found in [12]. In our implementation we put an emphasise on use of publicly available codes for problem assembly [8], Krylov solvers [4], and standard AMG solvers [18]. The codes are used as a "black-box" solvers. This makes our approach a suitable alternative for engineers that work in the field of microfabrication technology. An extensive testing of our algorithm on problems in both 2D and 3D (the representative examples are given in Section 4) demonstrates robustness and applicability of our methodology in various industrial-type problems.

2. PROBLEM FORMULATION

The deformation of a continuous material body occupying a bounded domain $\Omega \in \mathbb{R}^d$, with a smooth boundary Γ, is governed by the boundary value problem

$$-\sigma_{ij,j} = f_i \quad \text{in} \quad \Omega \tag{1}$$

$$\sigma_{ij} n_j = g_i \quad \text{on} \quad \Gamma_g \tag{2}$$

$$u_i = h_i \quad \text{on} \quad \Gamma_h \tag{3}$$

given in the indicial notation where the sum over the repeated index is implied. Here, σ is the Cauchy stress tensor, u is the displacement vector, f is a body force vector, h represents the prescribed displacement of the boundary segment $\Gamma_h \subset \Gamma$, and g is the surface traction of the boundary segment $\Gamma_g \subset \Gamma$, ($\Gamma_d \cap \Gamma_g = \emptyset$) with outward unit normal vector n.

For the linear elasticity problems, the stress tensor can be expressed as

$$\sigma_{ij} = \mu \left(u_{i,j} + u_{j,i} \right) + \lambda u_{k,k} \delta_{ij} + \sigma^0_{ij} \tag{4}$$

where $\mu > 0$ and $\lambda > 0$ are Lamé's coefficients, δ is the Kronecker delta, while tensor σ^0 takes into account the previous stress history. In the stress analysis of microfabricated structures σ^0 specifies the intrinsic stress in the deposited material layers or accounts for the intrinsic strain due to the variation of processing temperature [7].

Multiplying (1) by an arbitrary test function $\psi \in V(\Omega)$ and integrating by parts over Ω, one can also obtain a weak form of the problem (1)–(3) which is: find $u \in V(\Omega)$ such that

$$\int\limits_{\Omega} \sigma_{ij} \psi_{,j} \, d\Omega = \int\limits_{\Omega} f_i \psi \, d\Omega + \int\limits_{\Gamma_g} g_i \psi \, d\Gamma \tag{5}$$

where $V(\Omega)$ is the space of admissible displacement fields in Ω that satisfy the Dirichlet boundary condition (3). In our case $V(\Omega) = [H^1_0(\Omega)]^d$, where

$$H^1_0(\Omega) = \left\{ v \, : \, \int_{\Omega} (v^2 + \mid \nabla v \mid^2) \, dx < \infty, \; v = 0 \text{ on } \Gamma \right\}. \tag{6}$$

In order to formulate the corresponding discrete finite element problem, we introduce a finite dimensional space of piecewise polynomial functions $V_h(\Omega) \subset V(\Omega)$ with the basis $\{\phi^1 \, , \, \dots \, , \, \phi^N\}$, $N = \dim V_h(\Omega)$. The ith discrete displacement vector components is approximated in $V_h(\Omega)$ as

$$u_i = u^s_i \phi^s \tag{7}$$

where $\{u^s_i\}^N_{s=1}$ are unknown discrete solution coefficients associated with the basis functions. Notice that the indicial summation convention is implied in (7) also for the basis functions' indices. Employing the Galerkin method, where the weighting functions ψ in (5) are identical to the trial functions $\{\phi^r\}^N_{r=1}$, and using the identities

$$u_{i,j} \phi^r_{,j} = u^s_i \phi^s_{,j} \phi^r_{,j} = \phi^r_{,k} \phi^s_{,k} \delta_{ij} u^s_j \tag{8}$$

$$u_{j,i} \phi^r_{,j} = \phi^r_{,j} \phi^s_{,i} u^s_j \tag{9}$$

$$u_{k,k} \phi^r_{,j} \delta_{ij} = u^s_k \phi^s_{,k} \phi^r_{,j} \delta_{ij} = u^s_k \phi^s_{,k} \phi^r_i = \phi^r_{,i} \phi^s_{,j} u^s_j \tag{10}$$

we obtain a linear algebraic system of $d \cdot N$ equations

$$A_{ij}^{rs} u_j^s = b_i^r \qquad i = 1, \ldots, d;\ r = 1, \ldots, N \tag{11}$$

with coefficients

$$A_{ij}^{rs} = \int_\Omega \left(\mu \phi_{,k}^r \phi_{,k}^s \delta_{ij} + \mu \phi_{,j}^r \phi_{,i}^s + \lambda \phi_{,i}^r \phi_{,j}^s \right) \, d\Omega \tag{12}$$

and right hand sides

$$b_i^r = \int_\Omega \left(f_i \phi^r + \sigma_{ij}^0 \phi_{,j}^r \right) \, d\Omega + \int_{\Gamma_g} g_i \phi^r \, d\Gamma \tag{13}$$

We can also express (11) in matrix notation as

$$\mathbf{A}\mathbf{u} = \mathbf{b} \tag{14}$$

where

$$\mathbf{A} = \begin{pmatrix} \mathbf{A}_{11} & \cdots & \mathbf{A}_{1d} \\ \vdots & \ddots & \vdots \\ \mathbf{A}_{d1} & \cdots & \mathbf{A}_{dd} \end{pmatrix}, \qquad \mathbf{A}_{ij} = \begin{pmatrix} A_{ij}^{11} & \cdots & A_{ij}^{1N} \\ \vdots & \ddots & \vdots \\ A_{ij}^{N1} & \cdots & A_{ij}^{NN} \end{pmatrix} \quad i, j = 1, \ldots, d \tag{15}$$

are the global and component stiffness matrices, respectively, and

$$\mathbf{u} = \left(\mathbf{u}_1^T, \ldots, \mathbf{u}_d^T \right)^T, \qquad \mathbf{u}_i = (u_i^1, \ldots, u_i^N)^T, \quad i = 1, \ldots, d, \tag{16}$$

$$\mathbf{b} = \left(\mathbf{b}_1^T, \ldots, \mathbf{b}_d^T \right)^T, \qquad \mathbf{b}_i = (b_i^1, \ldots, b_i^N)^T, \quad i = 1, \ldots, d \tag{17}$$

are correspondingly partitioned vectors of unknowns and right-hand sides.

3. PRECONDITIONING METHODOLOGY

In order to solve the sparse linear system (14) we can apply a host of iterative and direct methods that are currently available. However, in this paper we constrain ourselves to the Krylov subspace iterative methods. Given a linear algebraic system $\mathbf{A}\mathbf{u} = \mathbf{b}$, and the initial guess $\mathbf{u}^{(0)}$ for the solution, with the corresponding residual $\mathbf{r}^{(0)} = \mathbf{b} - \mathbf{A}\mathbf{u}^{(0)}$, one constructs a sequence of iterates $\mathbf{u}^{(k)}$ as

$$\mathbf{u}^{(k)} = \mathbf{u}^{(0)} + \mathbf{p}^{(k)} \tag{18}$$

where

$$\mathbf{p}^{(k)} \in \mathcal{K}(\mathbf{A}, \mathbf{r}^{(0)}) = \mathrm{span}\{\mathbf{r}^{(0)}, \mathbf{A}\mathbf{r}^{(0)}, \ldots, \mathbf{A}^{k-1}\mathbf{r}^{(0)}\}. \tag{19}$$

There is a host of various Krylov-type methods aimed to solve both symmetric and non-symmetric linear systems. For an overview of convergence characteristics of Krylov methods see [16]. When applied to the linear system (14), Krylov methods converge to the solution in at most $d \cdot N$ iterations (providing that exact arithmetic is used). However, this limit could be quite impractical for large systems of equations. The situation can be radically improved by introducing preconditioning of the original system (14) as

$$\mathbf{M}^{-1}\mathbf{A}\mathbf{u} = \mathbf{M}^{-1}\mathbf{b}, \tag{20}$$

where $\mathbf{M}$ is some matrix which is spectrally close to $\mathbf{A}$ but simple to assemble and to compute an action of its inverse. The main idea behind preconditioning is to obtain more favourable spectral properties of the preconditioned matrix $\mathbf{M}^{-1}\mathbf{A}$ which enables us to solve the system (14), within some tolerance, in a substantially lower number of iterations than $d \cdot N$. Note that preconditioning increases the computational cost per step, since the matrix-vector product involves an action of $\mathbf{M}^{-1}$ to a vector. Thus, if the preconditioner is meant to be effective, the improvement in convergence speed must compensate for this extra cost per iteration.

The optimal (or nearly optimal) preconditioners are often constructed within the framework of the *space decomposition and subspace correction* (SSC) methods [22]. The principle idea is to decompose the global finite dimensional space into a set of local subspaces, introducing the corresponding restriction of the global problem into a set of local subspace problems. The global discrete solution is then obtained by appropriate combinations of the solutions for the local subspace problems. Typical examples are the *domain decomposition methods* with subspaces based on the geometrical partition of the domain, and the *multigrid methods* where the hierarchy of coarser grids represents a basis for the space decomposition.

For the successful implementation of SSC methods, it is important to provide subspace solvers which are capable of efficiently resolving the whole range of the spectrum spanned by the subspace. In that sense, the intuition behind the construction of a preconditioner for the stress analysis system based on scalar AMG solvers, is that the global finite dimensional space $V(\Omega)$ can be split into subspaces which could be suitable for the application of the scalar AMG methodology. To this end, it seems quite natural to introduce a component-wise space decomposition

$$V(\Omega) = \sum_{i=1}^{d} V_i(\Omega) \tag{21}$$

where $u_i \in V_i(\Omega)$. Namely, the space of displacement functions is decomposed into the subspaces that correspond to the displacements restricted along each of the Cartesian coordinates. While the standard AMG coarsening algorithm [18, 19] could be quite inappropriate for the global stiffness matrix $\mathbf{A}$, the subspace displacement matrices $\mathbf{A}_{ii}$, $i = 1, \ldots, d$ are generally quite convenient for the standard AMG coarsening and solving procedure. It means that the subspaces obtained by the component-wise decomposition are further decomposed into the subsubspaces that are generated by the AMG coarsening process. Therefore, a preconditioning methodology in our case assumes a recursive implementation of a space decomposition idea from [22]. In the sequel we concentrate only on the implementation of the component-wise decomposition step. For algorithmic details of the scalar AMG algorithms see [18, 19].

Following the classification of SSC methods into *parallel subspace correction* and *successive subspace correction* [22], the component-wise scalar AMG preconditioners have been also formulated as parallel ($\mathbf{M}_P$) and successive ($\mathbf{M}_S$). Their action is algorithmically expressed as

$\mathbf{M}_P^{-1}$: **for** $i := 1$ **to** d **do**
$\quad \mathbf{u}_i = \mathbf{AMG}_{\mathbf{A}_{ii}}(\mathbf{b}_i)$
end

Prepared using cnmauth.cls

which could be also intepreted as the block Jacobi preconditioner, and

$$\mathbf{M}_S^{-1}: \quad \underline{\textbf{for}}\ i := 1\ \underline{\textbf{to}}\ d\ \underline{\textbf{do}} \quad \mathbf{u}_i = \mathbf{AMG}_{\mathbf{A}_{ii}} \left(\mathbf{b}_i - \sum_{j=1}^{i-1} \mathbf{A}_{ij} \mathbf{u}_j \right) \quad \underline{\textbf{end}}$$

resembling the block Gauss-Seidel preconditioner. It should be emphasised that we do not discuss any parallelisation issues of the proposed preconditioning methodology, and the words *parallel* and *successive* should be read in the context used in [22]. In the above algorithms $\mathbf{AMG}_{\mathbf{A}_{ii}}$ denotes an action of scalar AMG algorithm applied as a preconditioner for the local stiffness matrix $\mathbf{A}_{ii}$, i.e. the operator $\mathbf{AMG}_{\mathbf{A}_{ii}}$ is an approximation for the operator $\mathbf{A}_{ii}^{-1}$. Notice that the parallel component-wise space decomposition approach could be interpreted as an evaluation of the displacement along each Cartesian coordinate provided that the displacements in other directions are fixed. The successive subspace correction utilises the displacements that have been already computed for other directions. In practical implementations, the subspace problems could actually be solved only approximately using scalar AMG solvers as the local subspace preconditioners, typically performing only a small fixed number of AMG cycles in this context (1 or 2), with an appropriate number of pre- and post smoothing steps. AMG is essentially used as an internal preconditioner of subspace problems. The AMG setup procedure, in which the coarse level multigrid components are assembled, can be performed for each submatrix $\mathbf{A}_{ii}$ separately at the beginning of an iterative solution procedure.

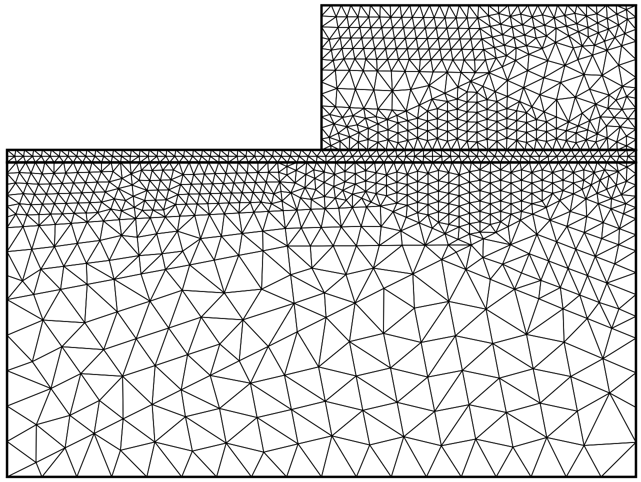
4. CASE STUDY

The proposed preconditioning methodology has been practically applied to stress analysis problems that arise in microfabrication technology. Microfabrication technology employs sequences of processes that utilise a variety of structural elements by embedding, butting and overlaying material layers which have widely differing mechanical properties and intrinsic built-in stress distributions [10]. There is an increasing interest in simulating the stress evolution of microfabrication structures. The principle goals are to predict and eliminate the harmful effects that stress can impose on the produced microelectronics and micromechanical devices, and to provide guidelines for the development of new technologies that may take some advantage of the presence of stress.

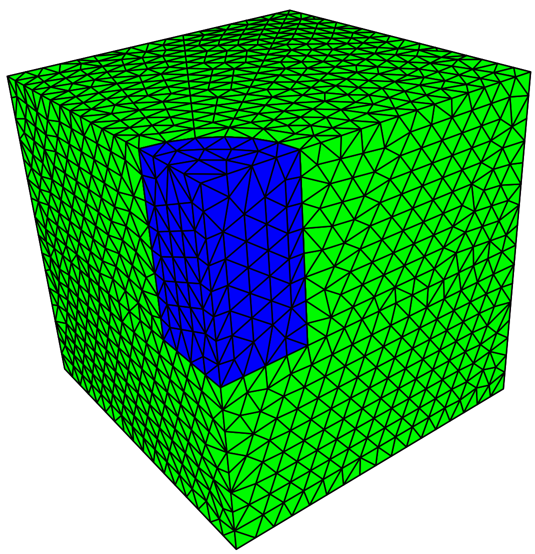
The principal sources of stress fields in microfabrication technology are intrinsic stress fields (σ_{ij}^0) due to the deposition of material layers with built-in stress, or due to the mismatch of thermal expansion coefficients between different material layers [7]. Accordingly, two model problems, with the characteristic features of both stress sources, have been selected for our numerical experiments.

The geometry and typical grid structure for the two selected test examples are shown in Figure 1 (a) and (b), respectively. In the first example, we consider a 2D structure consisting of three different material layers with the specified built-in stress distribution in the top layer. The second example involves stress analysis after temperature variation in the 3D structure consisting of two material layers with different thermal expansion coefficients.

The discretization is performed using piecewise linear finite elements based on unstructured grids created by the grid generation software GEOMPACK [8]. The Krylov iterative solvers that we have employed are Generalised Minimal Residual (GMRES) [14] and BiConjugate Gradient (BCGS) [17] as provided by the PETSc library [4]. The well established and publicly available code AMG1R5



(a)



(b)

Figure 1. Material distribution and triangulation (a) 2D grid with 1290 grid nodes, (b) 3D grid with 5425 nodes

[18] has been used as a scalar AMG solver. The approximate AMG solving procedure consists of 1 V(1,1) multigrid cycle. The component-wise AMG preconditioner, implemented in both parallel (AMG-P) and successive (AMG-S) versions, have been compared to the standard incomplete LU (ILU) preconditioner [6] from PETSc library.

Figure 2 gives the convergence history for various Krylov solvers and various preconditioners for one particular discretisation of the 2D and 3D examples (that is presented on Figure 1). It can be clearly seen that the Krylov solvers preconditioned by the AMG converge rapidly in a monotonic fashion (nearly straight line plots), while the ILU preconditioned solvers exhibit much slower and less smooth convergence characteristics.

Table I presents convergence results for various discretisations of the 2D problem. The results are given in terms of iteration counts needed for the two Krylov solvers (with various preconditioners) to reduce the initial ℓ_2 norm of the residual by the factor 10^{-10}. Table II contains the appropriate results for various discretisations of the 3D problem that is outlined above. From these tables it can be concluded that the iteration counts in 2D case for the AMG preconditioning are virtually independent of the problem size. In the 3D case, a similar conclusion may be derived for the BCGS method, whilst in our examples GMRES iteration counts slowly increase with the problem size. We hypothesise, however, that this number tends to a constant in the limit. This behaviour can be explained by the fact that the spectrum of a preconditioned matrices $\mathbf{M}_P^{-1}\mathbf{A}$ and $\mathbf{M}_S^{-1}\mathbf{A}$ can be bounded independently of the problem size (see [12] for details). Spectral bounds, however, depend upon the domain geometry.

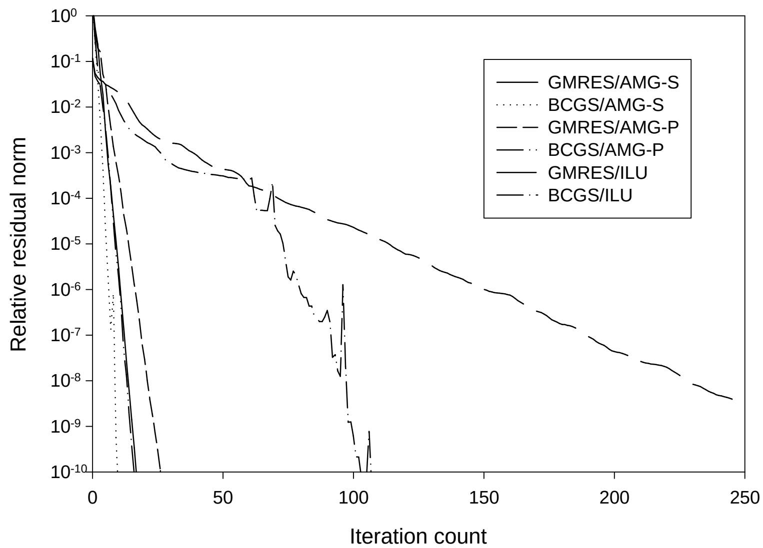
Table I. Iteration counts for the 2D problem.

$2N$	GMRES/ AMG-S	BCGS/ AMG-S	GMRES/ AMG-P	BCGS/ AMG-P	GMRES/ ILU	BCGS/ ILU
2580	17	10	26	16	345	108
4838	17	10	26	16	401	135
9534	17	10	26	17	752	185
18746	18	10	27	16	823	263
37378	19	11	28	18	1274	758
74810	19	11	28	18	2219	762

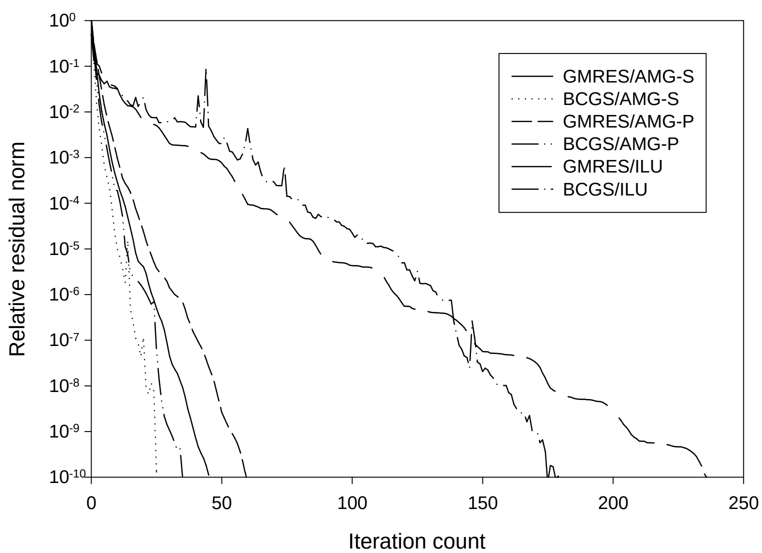
Table II. Iteration counts for the 3D problem.

$3N$	GMRES/ AMG-S	BCGS/ AMG-S	GMRES/ AMG-P	BCGS/ AMG-P	GMRES/ ILU	BCGS/ ILU
16275	45	26	60	35	251	180
27120	53	26	68	35	1290	324
53412	55	28	77	36	8472	1404

Finally, Tables III and IV present the total wall clock time in seconds for the preconditioned Krylov solvers with various discretisations of the 2D and 3D problem. All our numerical experiments are carried out on a single processor HP 9000/782 architecture. It should be emphasised that the AMG



(a)



(b)

Figure 2. Convergence history (a) 2D problem (b) 3D problem

Table III. Execution times for the 2D problem.

$2N$	GMRES/ AMG-S	BCGS/ AMG-S	GMRES/ AMG-P	BCGS/ AMG-P	GMRES/ ILU	BCGS/ ILU
2580	0.53	0.58	0.78	0.89	1.18	0.41
4838	1.09	1.16	1.53	1.76	3.41	1.58
9534	2.19	2.31	3.15	3.51	18.60	6.26
18746	4.50	5.13	6.48	7.64	42.74	19.33
37378	9.74	10.98	13.95	16.43	148.98	118.80
74810	20.87	22.71	28.61	33.55	489.01	240.28

Table IV. Execution times for the 3D problem.

$3N$	GMRES/ AMG-S	BCGS/ AMG-S	GMRES/ AMG-P	BCGS/ AMG-P	GMRES/ ILU	BCGS/ ILU
16275	17.91	18.47	21.72	22.99	17.81	21.08
27120	37.67	40.96	45.18	43.78	156.29	62.29
53412	92.75	89.39	116.40	111.88	2079.37	552.65

setup times and ILU factorisation time are also included in the execution times. Notice that for the smaller problems, ILU preconditioned iterative methods are the fastest. However, component-wise AMG preconditioned Krylov methods become faster than the ILU methods for sufficiently large discrete problems. From the Tables III and IV it can be concluded that the AMG preconditioned Krylov solvers scale linearly with problem size and are faster than their ILU preconditioned counterparts. In all numerical experiments the successive implementation of the component-wise AMG preconditioning is superior to the corresponding parallel version. It should be emphasised that the results obtained compare favourably, both in terms of the execution time and the memory requirements, with the direct sparse solvers provided in the PETSc library (although, for clarity of presentation, we did not include these comparisons here). This makes the method that we propose a suitable choice for the class of problems under consideration in this paper.

5. CONCLUSIONS

In this paper we present a methodology for solving stress analysis problems based on the component-wise algebraic multigrid preconditioners. The problem under consideration has an important industrial application in microfabrication technology. The principal idea lies in the implementation of a robust scalar AMG solver as a preconditioner in a component-wise fashion (i.e. as the local subspace preconditioner). We emphasise that our implementation is one particular realisation within a global framework of component decomposition preconditioning. Consequently, various scalar factorisations can be implemented component-wise. Theoretical justification of our method is a topic of our forthcoming paper [12]. In our experiments we find the AMG solvers to be a suitable alternative for satisfying both the requirement for optimal scaling property and the flexibility requirement in terms of a domain complexity and grid construction. The numerical results obtained, and comparison with some other standard preconditioning approaches, justify our choice.

REFERENCES

1. Arnold D, Falk R, Winther R. Preconditioning in $H(\mathrm{div})$ and applications. *Math. Comp.*, 1997; **66**:957–984.
2. Arnold D, Falk RS, Winther R. Preconditioning discrete approximations of the Reissner–Mindlin plate model. *Math. Modelling Numer. Anal.*, 1997; **31**(4): 517–557.
3. Axelsson O, Gustafsson I. Iterative methods for the solution of the Navier equations of elasticity. *Comput. Methods Appl. Mech. Engng.*, 1978; **15**: 241–258.
4. Balay S, Gropp W, McInnes LC, Smith B. *PETSc 2.0 Users Manual*. ANL-95/11–Revision 2.0.28. Argonne National Laboratory. 1999.
5. Füllenbach T, Stüben K, Mijalković S. Application of an Algebraic Multigrid Solver to Process Simulation Problems. *Proceedings of 2000 International Conference on Simulation of Semiconductor Processes and Devices* 2000; 225–228.
6. Golub GH, Van Loan CF. *Matrix Computations.* John Hopkins University Press, Baltimore, MD, 1983.
7. Hu SM. Stress related problems in silicon technology. *Int. J. Appl. Phys.*, 1991; **15**: R53–R80.
8. Joe B. Geompack – a software package for the generation of meshes using geometric algorithms. *Adv. Eng. Software* 1991; **13**: 325–331.
9. Joppich W, Mijalković S. *Multigrid Methods for Process Simulation*. Springer-Verlag: Wien, 1993.
10. Joppich W, Mijalković S. Semiconductor Process Modeling. In *Wiley Encyclopedia of Electrical and Electronics Engineering*, Webster JG (ed.); Wiley: New York, 1999; **19**:127–139.
11. Kettler R. Analysis and comparison of relaxation schemes in robust multigrid and conjugate gradient methods. In *Multigrid Methods*, Hackbush W, Trottenberg U (ed.); Lecture Notes in Mathematics 960; Springer: Berlin, 1982; 502–534.
12. Mijalković S., Mihajlović MD. A component decomposition preconditioning for 3D stress analysis problems. *Conference on Preconditioned Robust Iterative Methods for Problems with Singularities (PRISM2001)*, May, Nijmegen, 2001.
13. Rusten T., Winther R. A preconditioned iterative method for saddle point problems. *SIAM J. Matrix Anal. Appl.* 1992; **13**:887–904.
14. Saad Y, Schultz MH. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM J. Sci. Stat. Comput.* 1986; **7**:856-869.
15. Silvester DJ, Wathen A. Fast iterative solution of stabilised Stokes systems Part II: Using general block preconditioners. *SIAM J. Numer. Anal.* 1994; **31**:1352–1367.
16. Silvester DJ, Elman H, Kay D, Wathen AJ. Efficient preconditioning of the linearized Navier–Stokes equations. *MCCM report 352*, 1999. To appear in *J. Comp. Appl. Math.* , 2001.
17. Sleijpen GLG, Fokkema DR. BiCGSTAB(ℓ) for linear equations involving unsymmetric matrices with complex spectrum. *ETNA* 1993; **1**:11–32.
18. Stüben K. Algebraic multigrid (AMG): experiences and comparisons. *Appl. Math. Comp.*, 1983; **13**: 419–452.
19. Stüben K. Algebraic Multigrid (AMG): An Introduction with Applications. *GMD Report 70*, November 1999.
20. Webster R. Efficient Algebraic Multigrid Solvers with Elementary Restriction and Prolongation. *J. Numer. Meth. Fluids.*, 1998; **28**: 317–336.
21. Wesseling P. *An introduction to multigrid methods*. John Willey & Sons: Chicester, 1992.
22. Xu J. Iterative methods by space decomposition and subspace correction. *SIAM review*, 1992; **34**: 581–613.

Prepared using cnmauth.cls