# On parallel solution of linear elasticity problems. Part III: Higher order finite elements

I. Gustafsson* and G. Lindskog*

*Department of Mathematics, Chalmers University of Technology, S-41296 Göteborg, Sweden*

SUMMARY

This is the third part of a trilogy on parallel solution of the linear elasticity problem. We consider the separate displacement ordering for a plain isotropic problem with Dirichlet boundary conditions. The parallel solution methods presented in the first two parts of the trilogy are here generalised to higher order by using hierarchical finite elements. We discuss node-numberings for high degree of parallelity and even processor load as well as the problem of stability of the modified incomplete Cholesky factorisations used. Several preconditioning techniques are studied and compared. Bounds for the condition numbers of the corresponding preconditioning methods are derived and computer experiments are performed in order to confirm the theory and give recommendations on the choice of method. 

KEY WORDS: elasticity problem; separate displacement component; hierarchical finite elements; preconditioned conjugate gradient method; parallel algorithms

## 1. INTRODUCTION

We consider the plain strain case of the linear elasticity problem with isotropic material and Dirichlet boundary conditions. The definition of this problem and the theoretical basis for the study was given in the first part of this trilogy [1], where also generalisations are discussed and relevant references are given. In the second part [2] the focus was more on implementation details and parallel computer experiments.

So far we have considered only linear finite element approximations, but in this third part of the trilogy we generalise the ideas to second order finite elements. We may then use the hierarchical two-level method presented in [3]. Since the generalisation to $p$-order hierarchical finite elements is straigthforward we restrict the presentation to the case $p = 2$.

The finite element discretisation is based on a variational formulation with the bilinear form

$$a(u,v) = \int_\Omega \{\sum_i \frac{\partial u_i}{\partial x_i}\frac{\partial v_i}{\partial x_i} + \sum_{j\neq i}[\frac{1-\tilde{\nu}}{2}\frac{\partial u_i}{\partial x_j}\frac{\partial v_i}{\partial x_j} + \tilde{\nu}\frac{\partial u_j}{\partial x_j}\frac{\partial v_i}{\partial x_i} + \frac{1-\tilde{\nu}}{2}\frac{\partial u_i}{\partial x_j}\frac{\partial v_j}{\partial x_i}]\}d\Omega \qquad (1)$$

* Correspondence to: ivar@math.chalmers.se, lindskog@math.chalmers.se

where $\tilde{\nu}$ is a transformed contraction ratio. Following [1], we assume a separate displacement ordering i.e. we order the displacement vector in blocks, one for each of the translation degrees of freedom along the coordinate axes. In the plain strain case we then obtain an assembled stiffness matrix of 2 by 2 block structure

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{12}^T & A_{22} \end{bmatrix} \tag{2}$$

It was demonstrated in [1] and [2] how this matrix can be preconditioned efficiently using the block diagonal part

$$C_D = \begin{bmatrix} A_{11} & 0 \\ 0 & A_{22} \end{bmatrix} \tag{3}$$

or a block incomplete factorisation, called full block factorisation in [2]. In fact the matrix $C_D$ corresponds to the finite element approximation of the "diagonal" form

$$\tilde{a}(u,v) = \int_\Omega \{\sum_i \frac{\partial u_i}{\partial x_i}\frac{\partial v_i}{\partial x_i} + \frac{1-\tilde{\nu}}{2}\sum_{j\neq i}\frac{\partial u_i}{\partial x_j}\frac{\partial v_i}{\partial x_j}\}d\Omega \tag{4}$$

One important result in [1], which we will use further in this paper, is the following bounds between the forms (1) and (4):

$$(1-\sigma)\tilde{a}(u,u) \leq a(u,u) \leq (1+\sigma)\tilde{a}(u,u) \tag{5}$$

where $\sigma = \frac{1+\tilde{\nu}}{3-\tilde{\nu}}$.

We also considered a modification of the form in (4):

$$\hat{a}(u,v) = \int_\Omega \{\sum_i \frac{\partial u_i}{\partial x_i}\frac{\partial v_i}{\partial x_i} + \tau_i\sum_{j\neq i}\frac{\partial u_i}{\partial x_j}\frac{\partial v_i}{\partial x_j}\}d\Omega \tag{6}$$

giving a modification of $C_D$:

$$\hat{C}_D = \begin{bmatrix} C_{11} & 0 \\ 0 & C_{22} \end{bmatrix} \tag{7}$$

where $C_{ii} \neq A_{ii}$ if $\tau_i \neq \frac{1-\tilde{\nu}}{2}$ fore some $i$. One of the main results in [1] is the estimate for the condition number

$$\kappa(\hat{C}_D^{-1/2} A \hat{C}_D^{-1/2}) \leq \frac{2}{1-\tilde{\nu}} \tag{8}$$

valid for $\frac{1-\tilde{\nu}}{3+\tilde{\nu}} \leq \tau_1 = \tau_2 \leq 1$. In particular, for $\tau_1 = \tau_2 = 1$, $C_{ii}$ corresponds to discretisation of the Laplace equation. Observe that the bound (8) is a factor $1/3$ less then the more generally applicable bound given in [4] when $\tilde{\nu} \to 1$.

As an alternative to the diagonal block preconditioning (3) the following full block factorisation was presented in [2]:

$$C_F = \begin{bmatrix} L_1 & 0 \\ A_{12}^T L_1^{-T} & L_2 \end{bmatrix} \begin{bmatrix} L_1^T & L_1^{-1}A_{12} \\ 0 & L_2^T \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} \\ A_{12}^T & \bar{A}_{22} \end{bmatrix} \tag{9}$$

where

$$\bar{A}_{22} = A_{12}^T A_{11}^{-1} A_{12} + A_{22}$$

is the so called Schur complement and $L_1 L_1^T = A_{11},\ L_2 L_2^T = A_{22}$ are Cholesky factorisations.

For this preconditioning the condition number is proved to satisfy the bound

$$\kappa(C_F^{-1/2} A C_F^{-1/2}) \leq \frac{(3-\tilde{\nu})^2}{8(1-\tilde{\nu})}.$$

In this paper, in contrast to [2], we only consider the full block factorisation in the case $\tau_1 = \tau_2 = \frac{1-\tilde{\nu}}{2}$ i.e for the form $\tilde{a}$ in (4).

The theory for the two-level method in [3] is based on the generalised Cauchy-Buniakowski-Schwarz (C-B-S) inequality

$$|a(u,v)| \leq \{a(u,u)a(v,v)\}^{\frac{1}{2}},\ u,v \in V$$

valid for the bilinear form defining the variational formulation on $V$, the variational space under consideration. Here we will use this inequality for the diagonal separate displacement part i.e. for the form $\tilde{a}$ in (4) or its modification $\hat{a}$ in (6).

For finite dimensional subspaces $V_1$ and $V_2$ of $V$ with only the trivial element in common and such that $V_1$ contains the constant function, we have the strengthened C-B-S inequality

$$|a(u,v)| \leq \gamma\{a(u,u)a(v,v)\}^{\frac{1}{2}},\ u \in V_1,\ v \in V_2 \tag{10}$$

where $0 < \gamma < 1$.

Here, $\gamma$ depends not only on the form $a$ i.e. on the underlaying pde-problem, but also on the subspaces $V_1$ and $V_2$ i.e. on the choice of finite elements. For a uniformly refined mesh, however, we have the nice result that $\gamma$ is independent of the mesh parameter $h$. For details on meshrefining of this kind we refer to [3]. The kind of hierarchical finite elements we use are linear basis functions associated with the vertex nodes in the triangulation for the space $V_1$ and quadratic basis functions associated with other nodes for the space $V_2$.

In practice $\gamma$ can be calculated on element level by solving local generalised eigenproblems, see [5]. For example, the Laplace problem in the plain, $p = 2$ and quadratic triangles give $\gamma = (2/3)^{1/2}$. For higher order ($p > 2$) or deformated triangles with one large angle, $\gamma$ is closer to 1, see [6]. As we will see in the resulting bounds for the condition numbers it is preferable to keep $\gamma$ away from 1.

For the form (4), $\gamma$ in general depends on $\tilde{\nu}$. In fact, $\gamma \to 1$ as $\tilde{\nu} \to 1$ for the triangulation just mentioned, in general. For an axiparallel orientation of the quadratic triangles, however, $\gamma = (2/3)^{1/2}$ is independent of $\tilde{\nu}$. The reason for this is symmetry in the $x_1$- and $x_2$-directions of the elements and the basis functions. Similar results hold for the generalised form (6) i.e. in general $\gamma = \gamma_i$ depends on $\tau_i$.

For the complete form (1) bounds for $\gamma$ are derived in [7], [6] and [8] for various finite element discretisations.

It is well known that modified incomplete Cholesky (MIC)-factorisations exist for diagonally dominant $M$-matrices, see eg [9]. When solving the inner systems in the preconditioning matrices $C_D$ or $\hat{C}_D$ we use MIC-CG technique and in order to obtain M-matrices i.e. existence of the MIC-factorisation and still preserve the rate of convergence we use the idea of modification on element level presented in [10]. An alternative could be to use the generalised

stable incomplete factorisation in [11]. This latter method, however, does not preserve the fast rate of convergence of the preconditioned method with respect to the mesh size parameter $h$.

The remainder of the paper is organised as follows: In Section 2 we derive condition number estimates for the diagonal as well as full block factorisation method in combination with hierarchical 2-level finite elements. In Section 3 we discuss suitable orderings of the unknowns in order for high degree of parallelity of the method. Numerical results concerning the rate of convergence and work per unknown are given in Section 4 and in Section 5 we make some concluding remarks.

## 2. CONDITION NUMBER ESTIMATES FOR VARIOUS PRECONDITIONING METHODS

We consider two-level linear-quadratic finite elements associated with the spaces $V_1$ and $V_2$ in (10) applied to the form (6), including (4) as a particular case. If the basis functions for $V_2$ are numbered first, the stiffness matrices $C_{ii}$ in $\hat{C}_D$ take the form

$$C_{ii} = \begin{bmatrix} C_{ii}^{(2)} & G_{ii}^T \\ G_{ii} & C_{ii}^{(1)} \end{bmatrix}, \quad i = 1,\ 2 \tag{11}$$

where $C_{ii}^{(1)}$ is the stiffness matrix corresponding to linear finite elements. An important fact, see [3], is that the condition number $\kappa(C_{ii}^{(2)})$ is of order $O(1)$, $h \to 0$ while, as is well known, the condition numbers $\kappa(C_{ii}^{(1)})$ and $\kappa(C_{ii})$ are of order $O(h^{-2})$, $h \to 0$. Furthermore, the size of the matrix $C_{ii}^{(1)}$ is about $1/4$ of the size of the matrix $C_{ii}$ (in two dimensions, in three dimensions $C_{ii}^{(1)}$ is even smaller compared to $C_{ii}$).

One of the main results in [3] is that if the block diagonal matrix

$$D_{ii} = \begin{bmatrix} C_{ii}^{(2)} & 0 \\ 0 & C_{ii}^{(1)} \end{bmatrix} \tag{12}$$

is used as preconditioning to $C_{ii}$ in (11) then the following bounds hold for the quotient of the quadratic forms:

$$1 - \gamma_i \leq \frac{x_i^T C_{ii} x_i}{x_i^T D_{ii} x_i} \leq 1 + \gamma_i, \quad \forall x_i \neq 0 \tag{13}$$

i.e. the condition number $\kappa(D_{ii}^{-1/2} C_{ii} D_{ii}^{-1/2}) \leq \frac{1+\gamma_i}{1-\gamma_i}$. Here $\gamma_i$ depends on $\tau_i$ in the form (6) in general and it now becomes clear why $\gamma_i$ should be well away from 1.

We now combine the diagonal block preconditioning (12) with the diagonal block separate displacement component preconditioning (7) and get the following preconditioning, denoted diagonal-diagonal, for the system (2):

$$C_{DD} = \begin{bmatrix} D_{11} & 0 \\ 0 & D_{22} \end{bmatrix} = \begin{bmatrix} C_{11}^{(2)} & 0 & 0 & 0 \\ 0 & C_{11}^{(1)} & 0 & 0 \\ 0 & 0 & C_{22}^{(2)} & 0 \\ 0 & 0 & 0 & C_{22}^{(1)} \end{bmatrix} \tag{14}$$

From (8) and (13) we then easily obtain, with $\gamma = max(\gamma_1,\ \gamma_2)$, the following bound for the condition number of the diagonal-diagonal preconditioning metod:

$$\kappa_{DD} = \kappa(C_{DD}^{-1/2} A C_{DD}^{-1/2}) \leq \frac{2(1+\gamma)}{(1-\tilde{\nu})(1-\gamma)} \tag{15}$$

and we recall the nice property that this condition number is independent of the mesh size parameter $h$ in the finite element discretisation.

When solving the preconditioning system with matrix $C_{DD}$ in (14) the systems $C_{ii}^{(1)}$, the so called inner systems, can be solved by any suitable method, direct or iterative. We use preconditioned conjugate gradient methods also for these inner problems with preconditioning by modified incomplete factorisation. In fact, one can avoid inner iterations by just replacing the matrices $C_{ii}^{(1)}$ by their modified incomplete factors. As pointed out above, the condition numbers of the matrices $C_{ii}^{(2)}$ are of order $O(1), h \to 0$ i.e. these matrices can be approximated by incomplete factors or even by diagonal matrices without violating the order of convergence of the outer iterations, for details see Section 4.

We now continue with alternative preconditionings for the system (2). For the two-level linear-quadratic finite element matrices (11) we can use a full block incomplete factorisation of the Schur complement kind

$$F_{ii} = \begin{bmatrix} L_i^{(2)} & 0 \\ G_{ii} L_i^{(2)^T} & L_i^{(1)} \end{bmatrix} \begin{bmatrix} L_i^{(2)^T} & L_i^{(2)^{-1}} G_{ii}^T \\ 0 & L_i^{(1)^T} \end{bmatrix} = \begin{bmatrix} C_{ii}^{(2)} & G_{ii}^T \\ G_{ii} & C_{ii}^{(1)} + G_{ii} C_{ii}^{(2)^{-1}} G_{ii}^T \end{bmatrix} \tag{16}$$

where $L_i^{(k)} L_i^{(k)^T} = C_{ii}^{(k)},\ k = 1,\ 2$ are (formal) complete Cholesky factorisations of $C_{ii}^{(k)}$. We analyse the case complete factorisations but have in mind that incomplete factorisations may be used or even preferable. Also observe that the notion complete factorisation is just formal. It is fairly easily seen that solving systems with the matrix $F_{ii}$ in (16) involves solving systems with $L_i^{(k)} L_i^{(k)^T} = C_{ii}^{(k)}$ as coefficient matrix and these systems can be solved by iteration too, so called inner iterations (again). Hence, the expensive complete factorisation can be avoided anyhow (like in the diagonal-diagonal method above).

It is shown in [3] that the following bounds hold for the quotients of the quadratic forms with the matrices (11) and (16):

$$1 \leq \frac{x_i^T F_{ii} x_i}{x_i^T C_{ii} x_i} \leq \frac{1}{1-\gamma_i^2},\ \forall x_i \neq 0 \tag{17}$$

i.e the condition number $\kappa(F_{ii}^{-1/2} C_{ii} F_{ii}^{-1/2}) \leq \frac{1}{1-\gamma_i^2}$. Combining this, in a straigtforward way, with the diagonal block separate displacement component preconditioning (7) gives the following preconditioning for the system (11), denoted diagonal-full:

$$C_{DF} = \begin{bmatrix} F_{11} & 0 \\ 0 & F_{22} \end{bmatrix}$$

From (8) and (17) we now easily derive the following bound for the diagonal-full preconditioning method, where (again) $\gamma = max(\gamma_1,\ \gamma_2)$:

$$\kappa_{DF} = \kappa(C_{DF}^{-1/2} A C_{DF}^{-1/2}) \leq \frac{2}{(1-\tilde{\nu})(1-\gamma^2)} \tag{18}$$

Next, we intend to combine the full block separate displacement component factorisation (4) with the hierarchical finite element block diagonal preconditioning (12) To this aim let $\tilde{L}_i\tilde{L}_i^T = (1+\gamma)D_{ii},\ i = 1,\ 2$ and let

$$C_{FD} = \begin{bmatrix} \tilde{L}_1 & 0 \\ A_{12}^T\tilde{L}_1^{-T} & \tilde{L}_2 \end{bmatrix} \begin{bmatrix} \tilde{L}_1^T & \tilde{L}_1^{-1}A_{12} \\ 0 & \tilde{L}_2^T \end{bmatrix} \tag{19}$$

be a modification of (9). It follows that

$$C_{FD} = \begin{bmatrix} (1+\gamma)D_{11} & A_{12} \\ A_{12}^T & \bar{D}_{22} \end{bmatrix} \tag{20}$$

where

$$\bar{D}_{22} = A_{12}^T(1+\gamma)^{-1}D_{11}^{-1}A_{12} + (1+\gamma)D_{22} \tag{21}$$

and we will now derive a bound for the condition number $\kappa(C_{FD}^{-1/2}AC_{FD}^{-1/2})$. Recall that we here only consider the case $\tau_1 = \tau_2 = \frac{1-\tilde{\nu}}{2}$ so $C_{ii} = A_{ii}$ in (7) and onwards. From (5) and (13) we readily get, for arbitrary $x = (x_1, x_2) \neq 0$,

$$(1-\sigma)(1+\gamma)\{x_1^TD_{11}x_1 + x_2^TD_{22}x_2\} \leq (1+\gamma)x_1^TD_{11}x_1 + 2x_1^TA_{12}x_2 + (1+\gamma)x_2^TD_{22}x_2$$

$$\leq (1+\sigma)(1+\gamma)\{x_1^TD_{11}x_1 + x_2^TD_{22}x_2\}. \tag{22}$$

where $\sigma = \frac{1+\tilde{\nu}}{3-\tilde{\nu}}$.

Following ideas used in [2] for the analysis of the full block separate displacement preconditioning, we conclude from (22) that the eigenvalues of the matrix

$$\begin{bmatrix} \tilde{\gamma}^{-1/2}D_{11}^{-1/2} & 0 \\ 0 & \tilde{\gamma}^{-1/2}D_{22}^{-1/2} \end{bmatrix} \begin{bmatrix} \tilde{\gamma}D_{11} & A_{12} \\ A_{12}^T & \tilde{\gamma}D_{22} \end{bmatrix} \begin{bmatrix} \tilde{\gamma}^{-1/2}D_{11}^{-1/2} & 0 \\ 0 & \tilde{\gamma}^{-1/2}D_{22}^{-1/2} \end{bmatrix}$$

$$= \begin{bmatrix} I & F \\ F^T & I \end{bmatrix} \tag{23}$$

where $\tilde{\gamma} = (1+\gamma)$ and $F = (1+\gamma)^{-1}D_{11}^{-1/2}A_{12}D_{22}^{-1/2}$, are all in the interval $[1-\sigma, 1+\sigma]$ and hence the eigenvalues of $\begin{bmatrix} 0 & F \\ F^T & 0 \end{bmatrix}$ are in the interval $[-\sigma, \sigma]$, or equivalently,

$$\|F\|_2 = \| \begin{bmatrix} 0 & F \\ F^T & 0 \end{bmatrix} \|_2 \leq \sigma. \tag{24}$$

Since $F^TF = (1+\gamma)^{-2}D_{22}^{-1/2}A_{12}^TD_{11}^{-1}A_{12}D_{22}^{-1/2}$ we see that $\bar{D}_{22}$ in (21) becomes $\bar{D}_{22} = (1+\gamma)^{-1}A_{12}^TD_{11}^{-1}A_{12} + (1+\gamma)D_{22} = (1+\gamma)D_{22}^{-1/2}(I+F^TF)D_{22}^{-1/2}$ and then by (24)

$$(1+\gamma) \leq \frac{x_2^T\bar{D}_{22}x_2}{x_2^TD_{22}x_2} \leq (1+\gamma)(1+\sigma^2) \tag{25}$$

Using (13) and (25) we now derive the following bounds for the quotient of the quadratic forms with the matrices (2) and (20):

$$1 \leq \frac{x^TC_{FD}x}{x^TAx} = \frac{(1+\gamma)x_1^TD_{11}x_1 + 2x_1^TA_{12}x_2 + x_2^T\bar{D}_{22}x_2}{x^TAx}$$

$$= 1 + \frac{(1+\gamma)x_1^T D_{11} x_1 - x_1^T A_{11} x_1}{x^T A x} + \frac{x_2^T \bar{D}_{22} x_2 - x_2^T A_{22} x_2}{x^T A x}$$

$$\leq 1 + (\frac{1+\gamma}{1-\gamma} - 1)\frac{x_1^T A_{11} x_1}{x^T A x} + \{\frac{1+\gamma}{1-\gamma}(1+\sigma^2) - 1\}\frac{x_2^T A_{22} x_2}{x^T A x}$$

$$= 1 + (\frac{1+\gamma}{1-\gamma})\frac{x_1^T A_{11} x_1 + (1+\sigma^2) x_2^T A_{22} x_2}{x^T A x} - \frac{x_1^T A_{11} x_1 + x_2^T A_{22} x_2}{x^T A x} \tag{26}$$

In order to estimate the second term in the last row of (26) we use a generalisation of the lower bound in (5), proved in [2] to be valid for $u = (u_1, u_2)$ and for all $\delta > 0$:

$$a(u,u) \geq (1 - \sigma\delta)\tilde{a}(u_1, u_1) + (1 - \sigma\delta^{-1})\tilde{a}(u_2, u_2)$$

which should be adopted to the 'discretised' quotient

$$\frac{x_1^T A_{11} x_1 + (1+\sigma^2) x_2^T A_{22} x_2}{x^T A x}. \tag{27}$$

A proper choice of $\delta$ is then given by the equation $\frac{1-\sigma\delta}{1-\sigma\delta^{-1}} = \frac{1}{1+\sigma^2}$ with solution $\delta = \frac{1+\sqrt{\sigma^2+4(1+\sigma)}}{2(1+\sigma)}$. Then the quotient (27) takes the upper bound

$$\frac{x_1^T A_{11} x_1 + (1+\sigma^2) x_2^T A_{22} x_2}{x^T A x} \leq (1 - \sigma\delta)^{-1} = \frac{4(3-\tilde{\nu})}{(1-\tilde{\nu})(7-\tilde{\nu})}$$

For the last term in the last row of (26) we now use (5) once more to get

$$1 \leq \frac{x^T C_{FD} x}{x^T A x} \leq 1 + \frac{(1+\gamma)}{(1-\gamma)}\frac{4(3-\tilde{\nu})}{(1-\tilde{\nu})(7-\tilde{\nu})} - \frac{1}{1+\sigma}$$

Hence, the condition number for the full-diagonal preconditioning is bounded by

$$\kappa_{FD} = \kappa(C_{FD}^{-1/2} A C_{FD}^{-1/2}) \leq \frac{\sigma}{1+\sigma} + \frac{4(3-\tilde{\nu})(1+\gamma)}{(1-\tilde{\nu})(7-\tilde{\nu})(1-\gamma)} \tag{28}$$

and this bound behaves like $\frac{4(1+\gamma)}{3(1-\tilde{\nu})(1-\gamma)}$ as $\tilde{\nu} \to 1$ i.e. this bound is about a factor $2/3$ smaller than the bound for $\kappa_{DD}$ in (15), for $\tilde{\nu}$ close to 1.

Finally, we can combine the full block separate component preconditioning (9) with the hierarchical full block preconditioning (16). We now let $\tilde{L}_i \tilde{L}_i^T = F_{ii}$, $i = 1, 2$ be factors of $F_{ii}$ in (16) and get very similar to (19)

$$C_{FF} = \begin{bmatrix} \tilde{L}_1 & 0 \\ A_{12}^T \tilde{L}_1^{-T} & \tilde{L}_2 \end{bmatrix} \begin{bmatrix} \tilde{L}_1^T & \tilde{L}_1^{-1} A_{12} \\ 0 & \tilde{L}_2^T \end{bmatrix} = \begin{bmatrix} F_{11} & A_{12} \\ A_{12}^T & \bar{F}_{22} \end{bmatrix}$$

where $\bar{F}_{22} = A_{12}^T F_{11}^{-1} A_{12} + F_{22}$. Then by (17) we get the bound corresponding to (22) as follows:

$$(1-\sigma)\{x_1^T F_{11} x_1 + x_2^T F_{22} x_2\} \leq x_1^T F_{11} x_1 + 2x_1^T A_{12} x_2 + x_2^T F_{22} x_2$$

$$\leq (1+\sigma)\{x_1^T F_{11} x_1 + x_2^T F_{22} x_2\}$$

and analogous to (25) and (26), respectively, we get

$$1 \leq \frac{x_2^T \bar{F}_{22} x_2}{x_2^T F_{22} x_2} \leq 1 + \sigma^2$$

and

$$1 \leq \frac{x^T C_{FF} x}{x^T A x} = \frac{x_1^T F_{11} x_1 + 2 x_1^T A_{12} x_2 + x_2^T \bar{F}_{22} x_2}{x^T A x}$$

$$\leq 1 + (\frac{1}{1-\gamma^2} - 1)\frac{x_1^T A_{11} x_1}{x^T A x} + \{\frac{1}{1-\gamma^2}(1+\sigma^2) - 1\}\frac{x_2^T A_{22} x_2}{x^T A x}.$$

A completely similar analysis as that one giving us (28) from (26) now gives the bound for the condition number of the full-full preconditioning:

$$\kappa_{FF} = \kappa(C_{FF}^{-1/2} A C_{FF}^{-1/2}) \leq \frac{\sigma}{1+\sigma} + \frac{4(3-\tilde{\nu})}{(1-\gamma^2)(1-\tilde{\nu})(7-\tilde{\nu})} \tag{29}$$

We see, by comparing this bound to the bound (28), that the condition number is just affected by the more accurate hierarchical full block factorisation (16), with condition number given by (17), compared to the hierarchical diagonal preconditioning (12), with condition number given by (13).

In a similar way as in the $C_{DD}$ and $C_{DF}$ preconditionings, the involved (formally) complete Cholesky factorisations in the $C_{FD}$ and $C_{FF}$ preconditonings, i.e. the inner iterations, can be replaced by incomplete IC or MIC factorisations, for detailed implementation see Section 4.

## 3. ORDERINGS FOR PARALLELITY AND PARALLEL COMPLEXITY

We define a suitable parallel node-numbering for the model problem considered in [1] and [2] i.e. for the plain isotropic elasticity problem with homogeneous Dirichlet boundary conditions and constant body force on a unit square with various orientation $\theta$, the angle between the $x_1$-axis and the hypotenuses in the triangulation.

For the vertex-nodes in the linear-quadratic finite elements we use the node-numbering in Figure 1, giving a partilly parallel method with almost even processor load, as pointed out in [1]. The reason for this partial parallelism is that the matrices $C_{ii}^{(1)}$ in (11) become block matrices with the block diagonal having diagonal blocks of almost the same size, for details see [1]

For the edge-nodes, indicated in Figure 1, we use a similar columnwise ordering, from right to left say. Then also the matrices $C_{ii}^{(2)}$ in (11) become block matrices with the diagonal blocks being diagonal with more varying sizes than $C_{ii}^{(1)}$ however, giving a more uneven processor load.

The partial parallelism obtained by this ordering is utilised when solving the preconditiong systems i.e. when performing the forward- and backward substitutions in the triangular modified incomplete Cholesky MIC(0) factors as described in [1]. Observe that if $C_{ii}^{(2)}$ is just approximated by its diagonal, the simple method used in Section 4, any ordering can be used

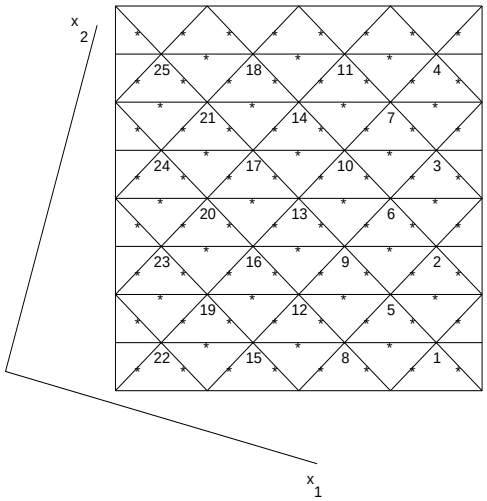


Figure 1. The test problem

for $C_{ii}^{(2)}$. We use this parallel ordering to make generalisation possible to IC factorisation of $C_{ii}^{(2)}$ as described in [3].

Next we consider the parallel complexity in solving the systems of linear equations with matrix A in (2) by the preconditioned gradient method. We use the preconditioning matrix $C_{DD}$ in (14), where the matrices $C_{ii}^{(1)}$ $i = 1,\ 2$ are replaced by their modified incomplete Cholesky, MIC(0) factors and $C_{ii}^{(2)}$ $i = 1,\ 2$ are replaced by their diagonals.

Following [2] the domain is decomposed between the processors in order to get as equal processor load as possible and also to get effective message transfers. Each processor calculates only the data elements corresponding to meshnodes in its own area. Let $m$ be the number of diagonal blocks in the matrices $C_{ii}^{(1)} i$ $i = 1,\ 2$. For the testproblem in Figure 2, $m = 5$ and the domain is partitioned between three processors $P_1$, $P_2$ and $P_3$.

For the calculation of the product $A$ times a vector, vector elements corresponding to nodes near the subdomain boundaries have to be transported to the neighbouring processor. This is due to the existence of non-zero couplings between nodes on each side of the subdomain boundaries. The nodes of current interest are denoted by $\Box$ in the Figure 2. One send and one receive are required for each such node. One processor has (in general) two neighbour processors and has to send $3m + 1$ elements and receive $3m + 1$ elements. Since the system is a $2 \times 2$ block system the total amount of communication (sends and receives) is $4(3m+1) \approx 12m$ per processor. For the solution of the preconditioning system with matrix $C_{DD}$ in (14), transportations are needed only when solving the triangular systems corresponding to $C_{ii}^{(1)}$ $i = 1,\ 2$. These matrices only involve vertex nodes and then each processor has to send $m$ elements and receive $m$ elements when solving a triangular system. Since there are 4 triangular systems the total amount of communication per iteration is $4 * 2m = 8m$ per processor. In total, for one iteration, $20m$ transports are needed per processor.

The amount of arithmetic operations per iteration per processor is $(132m^2 + 12m)/k$, where

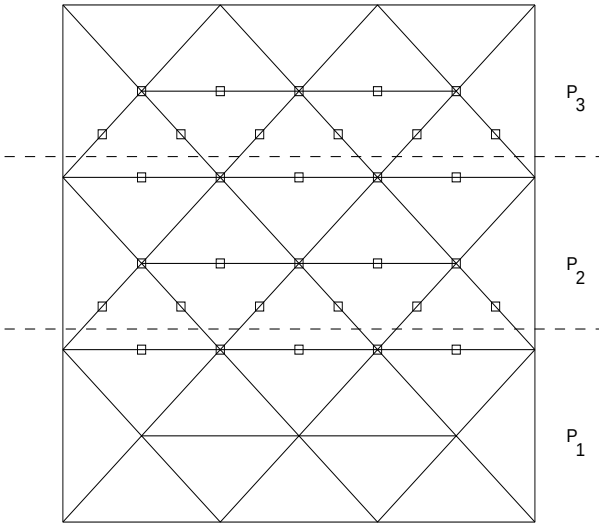


Figure 2. The test problem with decomposition of the domain between three processors.

$k$ is the number of processors. Only terms including $m$ are considered here. Hence, when the problem parameter $m$ is increased by a factor 2, the number of arithmetic operations increases by a factor $\approx 4$, while the number of data transports increases by a factor $= 2$. In [8] we have made parallel implementation of the approximate block-diagonal preconditioning method based on the use of linear finite elements. We use a Sun Enterprise 10000. Results from the implementation concerning times for the arithmetic computations and the communication showed a good agreement with the theoretically estimated times. Moreover, in [8] the analysis of the amount of transportation is shown to explain the speedup values resulting from the implementation. For the method we analyse with respect to parallel complexity in this paper, we theoretically get the times in Table I. The time per arithmetic operation and the time per transportation refer to the computer used in [8]. The resulting estimate of speedup is also given. The results show that a smaller speedup can be expected for smaller problem due to a larger percent of communication time for these problems. For larger problems a good speedup can be expected.

## 4. COMPUTER EXPERIMENTS

The numerical tests are performed on our model problem, see Section 3, with orientation $\theta = 0$ or $\theta = -\pi/4$ and various sizes of the problem i.e. various fineness of the mesh. More general angles are used in [1] and [2]. Following [1] we let $m$ be the number of diagonal blocks in the matrices $C_{ii}^{(1)}$ i.e. $m = 7$ in Figure 1. The total number of unknowns in our problem is then $4m^2 + 4m + 2$. The parallel node-numbering used is indicated in Figure 1. We use zero

| Number of processors | | m=31 | m=63 | m=127 | m=255 | m=511 |
|---|---|---|---|---|---|---|
| 4 | Computation | 0.0057s | 0.0236s | 0.0959s | 0.3864s | 1.5513s |
| | Communication | 0.0051s | 0.0103s | 0.0 208s | 0.0418s | 0.0838s |
| | Speedup | 2.1 | 2.8 | 3.3 | 3.6 | 3.8 |
| 8 | Computation | 0.0029s | 0.0118s | 0.0479s | 0.1932s | 0.7757s |
| | Communication | 0.0051s | 0.0103s | 0.0208s | 0.0418s | 0.0838s |
| | Speedup | 2.9 | 4.3 | 5.6 | 6.6 | 7.2 |
| 16 | Computation | 0.0014s | 0.0059s | 0.0240s | 0.0966s | 0.3878s |
| | Communication | 0.0051s | 0.0103s | 0.0208s | 0.0418s | 0.0838s |
| | Speedup | 3.5 | 5.8 | 8.6 | 11.2 | 13.2 |

Table I. Theoretical computational complexity per iteration in seconds and communication per iteration in seconds by use of approximate diagonal-diagonal preconditioning.

| m | $\tilde{\nu}=0$ | $\tilde{\nu}=0.5$ | $\tilde{\nu}=0.9$ | $\tilde{\nu}=0.95$ | $\tilde{\nu}=0.995$ |
|---|---|---|---|---|---|
| 7 | 16(15) | 20(20) | 35(35) | 44(45) | 104(106) |
| 15 | 17(17) | 21(21) | 40(40) | 53(54) | 146(146) |
| 31 | 17(18) | 22(22) | 42(42) | 57(57) | 158(165) |
| 63 | 18(18) | 22(23) | 44(44) | 60(60) | 171(173) |
| p | 29 | 40 | 88 | 123 | 387 |

Table II. The number of outer iterations for the diagonal-diagonal preconditioning method applied to the model problem with $\theta=-\pi/4$ and various values of the elasticity parameter $\tilde{\nu}$ and size $m$ of the problem

starting vectors in outer as well as inner iterations and a relative residual stopping criterion $\|r^{(p)}\|_2 \leq \epsilon \|r^{(0)}\|_2$ with $\epsilon_o = 10^{-4}$ in the outer iterations and $\epsilon_i = 10^{-2}$ in the inner iterations.

In Table II we present the number of outer iterations for the model problem with $\theta=-\pi/4$ and various values of $m$ and $\tilde{\nu}$, preconditioned by the diagonal-diagonal technique with $\tau_1=\tau_2=\frac{1-\tilde{\nu}}{2}$ and $C_{ii}^{(2)}$ approximated by their diagonals. Within parentheses we also give the number of iteratons for $\tau_1=\tau_2=1$. In this latter case we can give a theoretical upper bound for the number of iterations. The value of $\gamma$ in (15) is then $\gamma=(2/3)^{1/2}$ independently of $\tilde{\nu}$ and then $\kappa_{DD} \leq 9.9\ \frac{2}{1-\tilde{\nu}}$. By solving a corresponding local generalised eigenproblem it is proved in [3] that the factor 9.9 is only increased to 15.2 if $C_{ii}^{(2)}$ are approximated by their diagonals. The upper bound for the number of iterations $p$ is then calculated by the well-known formula

$$p \leq \frac{1}{2}\sqrt{\kappa}\ \ln\frac{2}{\epsilon_o}+1$$

with $\kappa = 15.2\ \frac{2}{1-\tilde{\nu}}$ and $\epsilon_o = 10^{-4}$. This value it given in the last row of the table for the vaious values of $\tilde{\nu}$.

Table III is similar to Table II but for the orientation $\theta=0$. We observe that in this case the matrices $C_{ii}^{(1)}$ in the case $\tau_1=\tau_2=\frac{1-\tilde{\nu}}{2}$ are not M-matrices and therefore we perform modification on element level before the modified incomplete Cholesky factorisation MIC(0).

| m | $\tilde{\nu}=0$ | $\tilde{\nu}=0.5$ | $\tilde{\nu}=0.9$ | $\tilde{\nu}=0.95$ | $\tilde{\nu}=0.995$ |
|---|---|---|---|---|---|
| 7 | 15(15) | 21(20) | 35(37) | 47(51) | 105(139) |
| 15 | 17(17) | 22(21) | 38(41) | 51(57) | 135(161) |
| 31 | 17(17) | 23(22) | 40(43) | 54(59) | 144(171) |
| 63 | 18(18) | 24(23) | 42(45) | 57(61) | 160(183) |

Table III. The number of outer iterations for the diagonal-diagonal preconditioning method applied to the model problem with $\theta = 0$ and various values of the elasticity parameter $\tilde{\nu}$ and size $m$ of the problem

| m | $\tilde{\nu}=0$ | $\tilde{\nu}=0.5$ | $\tilde{\nu}=0.9$ | $\tilde{\nu}=0.95$ | $\tilde{\nu}=0.995$ |
|---|---|---|---|---|---|
| 7 | 17(16) | 20(20) | 34(34) | 46(45) | 91(96) |
| 15 | 18(18) | 22(22) | 40(40) | 53(54) | 139(146) |
| 31 | 19(20) | 23(25) | 43(48) | 58(66) | 157(175) |
| 63 | 22(25) | 25(32) | 46(62) | 62(84) | 170(225) |

Table IV. The number of iteration for the approximate diagonal-diagonal preconditioning applied to the model problem with $\theta = -\pi/4$ and various values of the elasticity parameter $\tilde{\nu}$ and size $m$ of the problem

We do not present the number of inner iterations for these methods since, as we will see soon, the best method involves no inner iterations. We just comment that the number of inner iterations is much larger for $\theta = 0$ than for $\theta = -\pi/4$ for values of $\tilde{\nu}$ close to 1, because for $\theta = -\pi/4$ the MIC-facorisations become exact when $\theta$ approaches 1.

We also comment that the number of iterations can be decreased by using $\tau_1 = 1/3$ instead of $\tau_1 = 1$ when $\theta = 0$. Then the number of iterations is 155 for $m = 63$ and $\tilde{\nu} = 0.995$. In fact one can, by solving generalised eigenproblems (again), prove that the value $\tau_1 = 1/3$ is optimal in this case giving the value $\gamma = \frac{1}{\sqrt{2}}$.

In Table IV and Table V we present the results for the approximate diagonal-diagonal preconditioning i.e. when the inner systems are just replaced by their modified incomplete Cholesky MIC(0) factors. Apparently, this technique corresponds to solving the inner systems with very low accuracy. In spite of this low inner accuracy the number of outer iterations is not increased very much as can be seen in the tables. As before the figures within parentheses are for the values $\tau_1 = \tau_2 = 1$. For $\theta = 0$ the number of iterations is decreased a little by using $\tau_1 = 1/3$ instead of $\tau_1 = 1$. For $m = 63$ the number of iterations is then decreased from 241 to 220. However, since it is simpler (just corresponding to Laplace equations) we recommend the choice $\tau_1 = \tau_2 = 1$. The increase in the number of iterations compared to the choice $\tau_1 = \tau_2 = \frac{1-\tilde{\nu}}{2}$ is minor except possibly for $\theta = -\pi/4$ and $\tilde{\nu}$ close to 1. However, this case is exceptional because then the modified incomplete Cholesky factorisations become exact.

For the diagonal-full preconditioning method with $C_{ii}^{(2)}$ replaced by their diagonals, we notice a decrease in the number of (outer) iterations as is indicated by the smaller bound for the condition number (18) compared to (15). This decrease, however, does not pay off the increase in the work per iteration for this more complicated factorisation.

For instance, for the approximate diagonal-full preconditioning i.e. with $L_i^{(1)} L_i^{(1)^T}$ in (16)

| m | $\tilde{\nu}=0$ | $\tilde{\nu}=0.5$ | $\tilde{\nu}=0.9$ | $\tilde{\nu}=0.95$ | $\tilde{\nu}=0.995$ |
|---|---|---|---|---|---|
| 7 | 17(16) | 21(21) | 37(39) | 49(51) | 109(116) |
| 15 | 18(18) | 23(22) | 39(43) | 51(59) | 131(159) |
| 31 | 20(20) | 26(26) | 44(50) | 59(67) | 159(179) |
| 63 | 26(27) | 32(33) | 58(64) | 77(87) | 210(241) |

Table V. The number of iteration for the approximate diagonal-diagonal preconditioning applied to the model problem with $\theta = 0$ and various values of the elasticity parameter $\tilde{\nu}$ and size $m$ of the problem

| $\theta$ | $\tilde{\nu}=0$ | $\tilde{\nu}=0.5$ | $\tilde{\nu}=0.9$ | $\tilde{\nu}=0.95$ | $\tilde{\nu}=0.995$ |
|---|---|---|---|---|---|
| $-\pi/4$ | 16 | 19 | 37 | 51 | 152 |
| 0 | 18 | 22 | 39 | 53 | 146 |

Table VI. The number of outer iterations for the full-diagonal preconditioniung applied to the model problem with $m = 63$ and $\theta = -\pi/4$ and $\theta = 0$

| $\theta$ | $\tilde{\nu}=0$ | $\tilde{\nu}=0.5$ | $\tilde{\nu}=0.9$ | $\tilde{\nu}=0.95$ | $\tilde{\nu}=0.995$ |
|---|---|---|---|---|---|
| $-\pi/4$ | 21 | 24 | 42 | 54 | 152 |
| 0 | 25 | 31 | 53 | 69 | 170 |

Table VII. The number of outer iterations for the approximate full-diagonal preconditioniung applied to the model problem with $m = 63$ and $\theta = -\pi/4$ and $\theta = 0$

being MIC(0) factors of $C_{ii}^{(1)}$, 203 iterations are required for $m = 63$, $\theta = 0$ and $\tau_1 = \tau_2 = \frac{1-\tilde{\nu}}{2}$ compared to 210 iterations for the approximate diagonal-diagonal method. Similarly, for $m = 63$, $\theta = 0$ and $\tau_1 = \tau_2 = 1$, the approximate diagonal-full preconditioning requires 226 iteration compared to 241 for the approximate diagonal-diagonal preconditioning.

We comment that no significant improvement has been observed by approximating $C_{22}^{(2)}$ by $\lambda_0\ diag(C_{ii}^{(2)})$ with $\lambda_0 \neq 1$. In [3] $\lambda_0$ is chosen to be the largest eigenvalue of the matrix $diag(C_{ii}^{(2)})^{-1}C_{ii}^{(2)}$ in the case $\tau_1 = \tau_2 = 1$, i.e. when solving the Laplace equation.

For the full-diagonal preconditioning method with $C_{ii}^{(2)}$ replaced by their diagonals, we also get a decrease in the number of (outer) iterations compared to the diagonal-diagonal method. This decrease is in agreement with the smaller bound for the condition number (28) compared to (15). In Table VI we present the number of iterations for the model problem with $m = 63$, $\theta = -\pi/4$ and $\theta = 0$ and various values of the elasticity parameter $\tilde{\nu}$. In the diagonal scaling by $\tilde{\gamma} = 1 + \gamma$, see (20) and (23), we have used the value $\gamma = \sqrt{2/3}$ all the time although in general $\gamma$ depends on $\tilde{\nu}$. In fact the number of iterations is fairly insensitive to this scaling.

Similarly, for the approximate full-diagonal preconditioning i.e. with $C_{ii}^{(1)}$ replaced by their modified incomplete MIC(0) factors in (12), we get a smaller number of iterations than for the approximate diagonal-diagonal preconditioning. In Table VII we present the number of iterations for $m = 63$, $\theta = -\pi/4$ and $\theta = 0$ for various values of $\tilde{\nu}$.

Notice that the number of iterations is increased only slightly compared to Table VI although

the inner iterations are replaced by just the MIC(0) factors. Once more, recall that the MIC(0) factorisations become exact as $\tilde{\nu} \to 1$ when $\theta = -\pi/4$, explaining the remarkable small number of iterations in these cases.

Regarding also the work per iteration we find that the costs for the approximate diagonal-diagonal and approximate full-diagonal are about the same for our model problem. The numbers of floating point operations (flops) per unknown and iteration are about 31 and 47, respectively, for these methods. Here, one flop consists of one multiplication plus one addition. For instance, for $m = 63$, $\tilde{\nu} = 0.995$ and $\theta = 0$ the total work per unknown is about 7471 (241 times 31) and 7990 (170 times 47), respectively and for $m = 63$, $\tilde{\nu} = 0.995$ and $\theta = -\pi/4$ the total work per unknown is 6975 (225 times 31) and 7144 (152 times 47), respectively.

We conclude that is does not pay off to use the more complicated approximate full-diagonal preconditioning compared to the simpler approximate diagonal-diagonal preconditioning.

Finally, for the full-full preconditioning with $C_{ii}^{(2)}$ replaced by $\lambda_0\ diag(C_{ii}^{(2)})$, we get the smallest number of iterations. This is expected from the smallest bound for the condition number (29). For instance, the model problem with $m = 63$, $\theta = 0$ and $\lambda_0 = 2$ only requires 121 outer iterations.

The approximate full-full preconditioning i.e. with $L_i^{(1)} {L_i^{(1)}}^T$ in (16) being MIC(0) factors of $C_{ii}^{(1)}$, however, is often slower than the approximate full-diagonal preconditioning. For instance, 174 iterations are needed for $m = 63$, $\theta = 0$ and $\lambda_0 = 1$. This value of $\lambda_0$ turnes out to be optimal in this case. The number of iterations is fairly insensitive to this value, however.

We realise that the more complicated approximate full-full preconditioning does not conquer the simpler approximate diagonal-diagonal preconditioning when solving our model problem, if the total work is considered.

## 5. CONCLUSIONS

We have demonstrated how the preconditioned conjugate gradient methods for solving the linear elasticity problem, presented in the first two parts of this trilogy, see [1] and [2], can be generalised to higher order finite elements. We have used hierarchical linear-quadratic elements in order to preserve the high degree of parallelity and the fast rate of convergence.

We have combined diagonal and full separate displacement preconditionings with diagonal and full hierarchical preconditionings. The arising four variants of preconditioning methods have been studied theoretically by their bounds for the resulting condition number and experimentally by computer implementations.

It has turned out that the simplest method, the approximate diagonal-diagonal method, is also the fastest for our model problem. Furthermore, the number of flops is increased only moderatly if this preconditioning is based on MIC(0) factors of matrices corresponding to the simple Laplace equation.

For our model problem with orientation $\theta = -\pi/4$ and elasticity parameter $\tilde{\nu} = 0.995$, i.e. the material is almost incompressible, we have also tested a finer mesh than in Section 4. With $m = 127$, that is 65026 unknowns in the problem, the approximate diagonal-diagonal method with $\tau_1 = \tau_2 = \frac{1-\tilde{\nu}}{2}$, i.e the unmodified method, required only 184 iterations (for a relative residual tolerance $10^{-4}$) and the total number of flops per unknown is then about 5700 for solving the system of equations.

When comparing the computational complexity for linear and quadratic finite elements it is reasonable (for sufficiently smooth problems) to asum nnume that $h = 1/64$ for quadratic finite elements (discretization error of order $O(h^3)$) compares to $h = 1/512$ for linear finite elements (discretization error of order $O(h^2)$). For our model problem with $\theta = 0$ this corresponds to about 4200 Mflops for linear finite elements and about 105 Mflops for quadratic finite elements. For $\theta = -\pi/4$ the corresponding figures are about 2000 and 85 Mflops, respectively. Hence we can expect that a given (sufficiently smooth) problem can be solved 20-40 times faster with quadratic finite elements than with linear finite elements.

As a side-result of this paper, we have demonstrated how well the idea of modification on element level in order to obtain M-matrices, see [10], works in practice.

Finally we conclude that, since the MIC(0) factorisations are performed only of matrices corresponding to linear finite elements, it is fairly easy to generalise the parallel implementation technique used in [2] to the higher order finite element methods presented in this paper.

## REFERENCES


1. Gustafsson I, Lindskog G. On parallel solution of linear elasticity problems, Part I: Theory. *Numerical Linear Algebra with Applications* 1998; **5**:123–139.
2. Gustafsson I, Lindskog G. On parallel solution of linear elasticity problems, Part II: Methods and some computer experiments. Report 2000:32, Department of Mathematics, Chalmers University of Technology, Göteborg, Sweden. *Numerical Linear Algebra with Applications*, to appear.
3. Axelsson O, Gustafsson I. Preconditioning and two-level multigrid methods of arbitrary degree of approximation. *Mathematics of Computation* 1983; **40**:219–242.
4. Axelsson O. On iterative solvers in structural mechanics; separate displacement orderings and mixed variable methods. *Mathematics and Computers in Simulations* 1999; **50**:11–30.
5. Gustafsson I. A class of preconditioned conjugate gradient methods applied to finite element equations. In *Preconditioned Conjugate Gradient Methods, Proceedings, Nijmegen 1989 , Lecture Notes in Mathematics,* Axelsson O, Kolotolina L Yu (eds). Springer Verlag, 1990; **1457**:44–57.
6. Jung M, Lange U. Semmler U. Two-level hierarchically preconditioned conjugate gradient methods for solving linear elasticity finite element equations. *BIT* 1989; **29**:748–768.
7. Margenov S.D. Upper bound of the constant in the strengthened C.B.S. inequality for FEM 2D elasticity equations. *Numerical Linear Algebra with Applications* 1994; **1**:65–74.
8. Achchab B. Maitre J.F. Estimate of the constant in two strengthened C.B.S. inequalities for F.E.M. systems of 2D elasticity: Application to multilevel methods and a posteriori error estimators. *Numerical Linear Algebra with Applications* 1996; **3**:147–159.
9. Gustafsson I. Modified incomplete Cholesky (MIC) factorizations. In *Preconditioning Methods; Theory and Applications,* Evans D J (eds). Gordon and Breach: New York, 1983; 265–293.
10. Gustafsson I. An incomplete factorization preconditioning method based on modification of element matrices. *BIT* 1996; **36**(1):86–100.
11. Kaporin I.E. High quality preconditioning of a general symmetric positive definite matrix based on its $U^TU + U^TR + R^TU$-decomposition. *Numerical Linear Algebra with Applications* 1998; **5**:483–509.