



ELSEVIER

Comput. Methods Appl. Mech. Engrg. 133 (1996) 247–257

**Computer methods
in applied
mechanics and
engineering**

Iterative methods for p -version finite elements: preconditioning thin solids

Jan Mandel^{a,b}

^a*Solvers International, Inc., 1920 Kohler Dr., Boulder, CO 80303, USA*

^b*Center for Computational Mathematics, University of Colorado at Denver, Denver, CO 80217-3364, USA*

Received 21 July 1995

Abstract

We present new preconditioning strategy for thin 3D p -version elements used to model plate and shell structures as well as the microstructure of composite materials. The strategy is incorporated in an adaptive preconditioner suitable for large real-world problems. The method is shown to perform well for several large real world problems, including a model of the skin of an aircraft with over 1.6 million degrees of freedom, and a highly accurate model of a 28 ply laminated fiber plate.

1. Introduction

The use of hierarchical high-order finite elements, known as the p -version finite element method, has numerous advantages: readily available refinement by simply adding new shape functions, simplified local a posteriori error estimation, avoidance of locking, and fast convergence to the exact solution, cf. the recent monograph by Szabo and Babuška [34]. Consequently, p -version is a major step towards the goal of fully automated structural analysis which would avoid the major problem of state-of-the-art finite element analysis, namely the need for human expert judgement to use the right elements and to fiddle with the mesh to obtain correct results, especially in the case of complicated shell problems. Instead, the p -version makes it possible to model plates and shells by thin 3D elements and thus to avoid the numerous pitfalls associated with the usual 2D models. Locking is avoided by the use of sufficiently high degree in the lateral direction of the polynomials approximating the transversal displacement. Polynomials linear or quadratic in the transversal direction are generally sufficient. A p -version package with automatic p -refinement based on a posteriori error estimates should create such models automatically. Structures previously very difficult to analyze accurately such as shells of variable thickness, junctures, and layers of thin elements in composite materials now can be treated in a completely automated way, with no parameters to be set and no modeling expertise required of the user.

So powerful and versatile a method is, however, not cheap. A major part of the computational cost is the solution of the system of linear equations with the stiffness matrix. For the same number of degrees of freedom, the p -version has much higher accuracy but also many more non-zero entries in the stiffness matrix. While the advantages of iterative solvers are now well established and almost all major finite element vendors offer iterative solvers based on conjugate gradients with standard preconditioners such as Jacobi and Incomplete Choleski, preconditioning of systems of equations arising from high-order thin elements encounters particular difficulties. The method presented in this paper is the first practical iterative method for thin 3D elements. Our algorithms choose adaptively between a number of

strategies, including new methods for thin elements, presented here, as well as earlier preconditioning methods of the author [17–19, 21, 22].

Our goal is to solve linear algebraic systems of equations arising from finite element discretizations produced by existing software. Then, the solver must take the algebraic system *as is* and take advantage of the special character of the problem only indirectly. Therefore, we use only information in the stiffness matrix and as little extra information as possible, which limits the choice of available algorithms.

The preconditioners proposed in this article are based on the *abstract additive Schwarz method*, which is essentially a generalization of the block Jacobi method to allow for overlapping of the blocks. It is easily seen that the preconditioned operator depends only on the selection of the subspaces associated with the blocks rather than on the particular basis vectors within the subspaces. Consequently, all our algorithms are *invariant to the selection of specific finite element shape functions*, which makes it easy to deliver the same performance for linear algebraic systems of equations coming from different finite element packages. The algorithms are implemented to require only the stiffness matrix and limited information about shape functions, such as their polynomial degree and an identification of their position on the element.

Additive Schwarz methods have been extensively studied in the domain decomposition literature and many methods can be formulated and analyzed in the additive Schwarz framework (cf. Dryja, Smith and Widlund [7, 8, 33]). A very useful iterative method for p -version elements based on the Schwarz method was proposed by Pavarino [28]; we have used a generalization of this method as our strategy for layers of thin elements. For a recent review of related methods, see Xu [36].

For interpretation of a related preconditioner in terms of special shape functions, see Babuška et al. [3]. For theory in the 2D case, see Babuška et al. [2]. The preconditioner and analysis from [2] were recently extended to a domain decomposition for the hp -version by Oden et al. [27]. Preconditioners for thin elements in 2D were studied by Mandel and Lett [14, 23]. Widlund and Pavarino [30] have proposed and derived theoretical results for other preconditioners for the p -version. The design of their preconditioner, however, is more complicated and their computations require additional information about the shape functions. For example, to construct vertex functions with almost minimal L^2 norm, needed in [30], one needs either the mass matrix or specific information about the shape functions.

Iterative methods for the p -version that use fast matrix–vector multiplication were developed at IBM by Morris et al. [26], and by Foresti et al. [9, 10]. Those methods require many hundreds of iterations and must therefore run in-core. In comparison, the advantage of our approach is that the preconditioner is judiciously chosen to give a small number of more expensive iterations. This makes out-of-core iterative solvers for very large problems practical.

Multigrid methods have proved to be extremely efficient iterative methods for systems arising from discretizations of elliptic partial differential equations [12, 24]. The present method is different in important aspects from multigrid methods, although the main idea is the same: a global but small auxiliary problem is constructed to provide resolution of global interactions while local computations resolve short range interactions. Multigrid methods have been applied to *spectral methods* and related spectral element methods by Rønquist and Patera [32], using spaces of polynomials of various degrees in place of the mesh hierarchy. Mitchell [25] uses a mesh hierarchy with high-order elements.

2. Preconditioner description

The linear elasticity problem is to find the small displacement field of a solid body under a load and with given boundary conditions. After discretization by the finite element method, one obtains the system of linear equations

$$Au = f \tag{1}$$

for the vector of coefficients u of the displacements. The matrix A is the stiffness matrix and it is symmetric and positive definite if the boundary conditions exclude rigid body motions, which will be assumed.

The method of choice is then the *preconditioned conjugate gradients method* [11, 13]. In each step, this method invokes the evaluation of the matrix–vector products Ax and Br , where B is another symmetric, positive definite operator, called *preconditioner*. It is well known that the number of steps to achieve a given precision in the energy norm $\|x\|_A = \sqrt{x^T Ax}$ is at most proportional to $\sqrt{\kappa}$, where κ is the condition number

$$\kappa = \kappa(B, A) = \lambda_{\max}(BA) / \lambda_{\min}(BA)$$

where λ_{\min} and λ_{\max} denote the least and the largest eigenvalue, respectively. In practice, the number of iterations is often much lower because of gaps in the spectrum of BA , but this luck disappears as the spectrum fills the interval $[\lambda_{\min}(BA), \lambda_{\max}(BA)]$ with increasing size of the problem and with better preconditioning that decreases the condition number κ .

The task of designing an efficient preconditioned conjugate gradients methods is thus one of finding a preconditioner B such that κ is small and the action of B is inexpensive to evaluate.

2.1. General form of preconditioner

We build an efficient preconditioner by the *abstract additive Schwarz method* [4, 7]. Let V be the finite element space, which we identify with the space of degrees of freedom, so that the stiffness matrix A is understood to be a linear mapping $A: V \rightarrow V$. Let V_0, \dots, V_m be subspaces of the space V such that

$$V = V_0 + \dots + V_m. \tag{2}$$

The action of the preconditioner B is defined by solving the problem on each subspace in a variational sense and adding the results to compute $u = Br$:

$$u_i \in V_i: \quad v_i^T A u_i = v_i^T r, \quad \forall v_i \in V_i, \quad i = 0, \dots, m, \tag{3}$$

$$u = \sum_{i=0}^m u_i. \tag{4}$$

The method is implemented using a linear operator $M: V_0 \times \dots \times V_m \rightarrow V$, leading to the representation

$$Br = MD^{-1}M^T r. \tag{5}$$

The operator M corresponds to the summation (4), including a possible change of basis, since, in general, the subspaces V_i are not spanned by subsets of the basis of V . The matrix D is the block diagonal of $M^T A M$ corresponding to the decomposition (2), and the action of D^{-1} represents the simultaneous solution of the subproblems (3) on the spaces V_i . The operator M is not stored as a matrix but rather in a factored form as a recording of the multipliers in the change of basis followed by assembly.

A specific preconditioning strategy now consists of the selection of the subspaces V_i .

Numerous other preconditioning methods are of the present form, with the matrix D possibly replaced by an approximation that is cheaper to invert and with various selections of the spaces V_i , see the discussion and references in [8, 16, 33].

2.2. Principles of strategy selection

The key observation for the selection of the strategy is the *P.L. Lions Lemma* [15], cf. [4, 7] for the present form and proof.

LEMMA 2.1. *If there is a constant c_0 such that every $v \in V$ has a decomposition satisfying*

$$v = \sum_{i=0}^m v_i, \quad v_i \in V_i, \quad \sum_{i=0}^m \|v_i\|_A^2 \leq c_0 \|v\|_A^2, \tag{6}$$

then $\lambda_{\min}(BA) \geq 1/c_0$.

The maximal eigenvalue is easily bounded as the maximum number of the spaces V_i that have a non-trivial intersection [4, 35]:

$$\lambda_{\max}(BA) \leq \max_{i=1, \dots, m} |\{V_j : V_j \cap V_i \neq \{0\}\}|. \quad (7)$$

In other words, the condition number bound depends on the boundedness in energy of the decomposition (2) and on the connectivity of the overlaps of the spaces V_i .

We select the space $V_0 \subset V$ as a *coarse space*, which is spanned by at least all linear shape functions. The problem (3) with $i = 0$ is then a lower-order discretization of the same elasticity problem as (1). We have shown in [22] that the *condition number* κ can be bounded independently of the number of elements only in terms of local quantities computable on each element and from the constant in Korn's inequality.

The spaces V_i , $i > 0$, are chosen as *local spaces* which are spanned by sets of modified shape functions with support in small subregions of the domain. In this case, the subproblem (3) corresponds to the Dirichlet problem for the subregion, discretized by the modified shape functions.

The spaces V_i are selected adaptively element by element based on various a priori numerical indicators and a posteriori estimators of the condition number of the preconditioned problem. The element level strategies are then merged to form a global strategy. The selection algorithms are designed so that in particular cases of practical interest, the spaces V_i are selected according to one of the *basic strategies* described in Section 2.3 below. The basic strategies are known to work well for the case of quasisuniform meshes and asymptotically optimal theoretical bounds on the energy of the decomposition can be either proved or at least justified by plausible heuristic arguments. If the selection algorithms are unable to find a suitable basic strategy for an element, a mixed strategy which combines the approaches of several basic strategies will be chosen. The basic tool for creating mixed strategies is merging the spaces V_i selected by a basic strategy to form fewer new larger spaces V_i . If the heuristic algorithms fail to find an acceptable element preconditioning strategy, the solver will give up on the element and all degrees of freedom of the element will be included in the coarse space V_0 . This will hopefully happen only very rarely and it indicates the need to build a new basic strategy for that type of element. In the extreme case, if the solver gives up on a large proportion of elements, it will switch over to a direct method.

2.3. Basic preconditioning strategies

Our principal assumption is that the shape functions on an element can be naturally identified as vertex, edge, face and interior functions. That is, a vertex function is non-zero at the vertex and zero on all faces that are not adjacent to the vertex. An edge function is given by its values on an edge and it is zero on all faces of the element that are not adjacent to the edge. A face function is determined by its values on a given face and it is zero on all other faces. Interior function is given by its values in the interior of the element and it is zero on the element boundary. We also need to assume that the shape functions are *hierarchical*, that is, the set of shape functions for $p + 1$ is obtained by adding new shape functions to the functions for p . From this follows the existence of *linear vertex* functions and *quadratic edge and face* functions.

Many of the strategies below use vertex, edge and face spaces, which consist of all corresponding vertex, edge or face functions, respectively. These are spaces of *global* shape functions, glued between neighboring elements as is usual in the conforming finite element method. So, for example, there is a space of edge functions for each edge in the finite element mesh.

For preconditioning purposes, the vertex, edge and face functions are modified by so-called *partial orthogonalization*, which we have introduced in [17]. Let $a(\cdot, \cdot)$ be the energy inner product. Partial orthogonalization of a shape function ϕ with respect to a collection of shape functions ψ_i is defined as replacing the shape function ϕ by the function

$$\tilde{\phi} = \phi + \sum_i c_i \psi_i \quad (8)$$

where the multipliers c_i are determined from the orthogonality condition $a(\tilde{\phi}, \psi_i) = 0$ for all ψ_i , or, equivalently, from the condition that the energy $a(\tilde{\phi}, \tilde{\phi}) = 0$ is minimal among all functions of the form (8). Partial orthogonalization corresponds to the transformation of the global stiffness matrix by $A \leftarrow XAX^T$ for a transformation matrix X ; see [17, 22] for details.

A particular important case is partial orthogonalization of all vertex, edge and face functions with respect to all interior functions, which is easily seen to be equivalent to static condensation of the element interiors. This will be done in all strategies below even if it may not be mentioned explicitly.

2.3.1. 2D partial orthogonalization with linear coarse

The coarse space consists of all P_1 functions present in the element. There are no vertex spaces, and edge spaces consist of functions energy orthogonal to interior functions.

This method was proposed by Babuška et al. [2], who have proved the estimate $\kappa \leq C(1 + \log^2 p)$ in the case of triangle and tensor product rectangle. This method is an analogue of a substructuring method of Bramble et al. [5] for the h -version, with each element corresponding to a subdomain.

The theory from [2] did not use the additive Schwarz framework. We restate the proof in this framework using the technical inequalities proved in [2], which will serve as reference for following heuristic arguments. To construct the decomposition (6) for a given v , note the equivalence of the energy norms $\|\cdot\|_\lambda$ and the Sobolev norm $\|\cdot\|_{H^1(\Omega)}$, from Poincaré inequality (in case of scalar problems), and from Korn’s inequality (for elasticity). Define $v_0 = I_h v$ by interpolation using the linear vertex functions. From the *discrete Sobolev inequality*

$$\|u\|_{L^\infty(K)}^2 \leq C(1 + \log p) \left(|u|_{H^1(K)}^2 + \frac{1}{h^2} \|u\|_{L^2(K)}^2 \right) \tag{9}$$

for all u piecewise polynomial of degree at most p on element K with diameter h ,

$$\|v_0\|_{H^1(K)}^2 \leq C(1 + \log p) \left(|v|_{H^1(K)}^2 + \frac{1}{h^2} \|v\|_{L^2(K)}^2 \right) \tag{10}$$

and using the fact that $v - v_0$ as well as $|v|_{H^1(K)}$ are invariant to adding a constant to v ,

$$\|v - v_0\|_{L^\infty(K)}^2 \leq C(1 + \log p) |v|_{H^1(K)}^2.$$

The function $v - v_0$ is zero on all vertices; then, for every element K , the function u_E which coincides with $v - v_0$ on edge E of the element K and is zero on the rest of ∂K , satisfies the estimate in the trace norms,

$$\|u_E\|_{H^{1/2}(\partial K)}^2 \leq C(1 + \log p) \|v - v_0\|_{L^\infty(\partial K)}^2 + C(1 + \log p) |v - v_0|_{H^{1/2}(\partial K)}^2.$$

Using the discrete Sobolev inequality (9), the polynomial extension theorem

$$\inf_{u|_{\partial K} = v - v_0, u \in P_p} |u|_{H^1(K)} \leq C |v - v_0|_{H^{1/2}(\partial K)}, \quad \forall u \in P_p(\partial K), \tag{11}$$

and the standard trace theorem

$$|u|_{H^{1/2}(\partial K)} \leq C |u|_{H^1(K)} \tag{12}$$

one obtains for the extension v_i with minimal energy of the edge functions u_{E_i} into the interiors of the elements that

$$\sum_{E_i \text{ cdge}} |v_i|_{H^1(K)}^2 \leq C(1 + \log^2 p) \|u\|_{H^{1/2}(K)}^2.$$

Adding the interpolation estimate (10) and noting that the spaces of interior functions for each element are energy orthogonal to each other as well as to all edge spaces, we obtain for each element K , that

$$\sum_i |v_i|_{H^1(K)}^2 \leq C(1 + \log^2 p) |v|_{H^1(K)}^2.$$

Adding these inequalities over all elements gives the inequality needed in Lions' lemma, with $c_0 = C(1 + \log^2 p)$.

It should be noted that the argument goes through only for the case of triangles or tensor product quadrilaterals where the extension theorem (11) is known. This theorem is not known for the important case of the reduced element (also called serendipity element).

2.3.2. 3D partial orthogonalization with linear coarse

Here, the coarse space consists of all linear functions present in the element; there is a vertex space for each vertex consisting of the vertex shape functions orthogonalized to all functions on adjacent edges, faces and interiors; edge spaces of functions orthogonalized to the adjacent faces and the interiors; and face spaces orthogonalized to the interior functions. This method was proposed by the author in [22]. It results in a smaller coarse problem than the quadratic coarse strategy (see Section 2.3.3) and it often has similar convergence properties.

The addition of vertex spaces is required by the fact that in 3D, the discrete Sobolev inequality (9) no longer holds; indeed, the logarithmic term in (9) has to be replaced by Cp^2 [30]. In the absence of vertex spaces, there is no other choice than to define the coarse component v_0 by linear interpolation, resulting in a bound on the condition number of the same order. Such fast growth of the condition number with p in the absence of vertex spaces is indeed observed numerically [22].

Using the technical inequalities developed by Pavarino and Widlund [30] (to analyze a different method), one can prove the bound $\kappa \leq C(1 + \log^2 p)$ for the case of tensor product elements similarly as in Section 2.3.1.

This method is related to a special case (with 1D vertex spaces) of the substructuring method of Smith [33], analyzed by Dryja and Widlund [8].

2.3.3. 2D partial orthogonalization with quadratic coarse

The coarse space consists of all quadratic functions present in the element. For every edge, the edge space consists of all functions on the edge except the quadratic, modified by partial orthogonalization to the interior functions. This method is useful only in the rather special cases where the other cheaper strategies do not work well enough and the quadratic coarse functions help capture the rigid body motions.

2.3.4. 3D partial orthogonalization with quadratic coarse

The coarse space consists of all quadratic functions present in the element. For each edge or face, the coarse functions are further modified by partial orthogonalization to all remaining functions on the edge or face. For each edge, there is the edge space consisting of all edge functions of degree higher than quadratic, modified by partial orthogonalization to the adjacent face functions and to the interior. For each face, the face space consists of the shape functions of degree higher than quadratic, orthogonalized to the interior functions.

This was the method first proposed by the author for the 3D case [17]. The use of quadratic coarse space is motivated by the fact that since the discrete Sobolev inequality (9) fails in 3D, a linear interpolant v_0 would have unacceptably high energy. Since the spaces V_i form a direct sum, each component v_i is determined by v uniquely. It is then expected and confirmed by observed low condition numbers for $p \leq 8$ that the energy of the quadratic component v_0 is reasonable.

2.3.5. Overlap

In this strategy, there is one space for each vertex, consisting of all functions associated with the vertex as well as all adjacent edges and faces (in 3D). The coarse space consists of linear functions. This is essentially the method of Pavarino [28, 29], which includes in the vertex spaces also all functions in the adjacent interiors. The elimination of the interiors up front saves us considerable computational expense and results in a mathematically equivalent method. Pavarino has proved that because of the generous overlap, (6) holds with c_0 independent of p , so $\kappa \leq C$. This strategy has been observed to be quite robust but it is very expensive. It is easy to see that preconditioning system is quite large since, in general, every edge and face function belongs to several subspaces V_i .

2.3.6. Dimensional reduction of thin elements

The classical approach to plates and shells relies on neglecting the deformation through the thickness and approximating the shear by a low-order polynomial [31]. The displacement and stiffness are then represented in terms of a 2D model. We use a similar approach to find a preconditioner for thin elements in a single layer.

The motivation of dimensional reduction is given by the following asymptotical expansion due to Ciarlet [6]. Consider the domain $\Omega_\varepsilon = \mathcal{O} \times [-\varepsilon, +\varepsilon]$, $\mathcal{O} \subset \mathbb{R}^2$. After the scaling $(\tilde{x}_1, \tilde{x}_2, \tilde{x}_3) \mapsto (x_1, x_2, \varepsilon x_3)$, and a corresponding scaling of the displacements, the quadratic form in the strain energy becomes

$$\int_{\Omega} (\lambda e_{\alpha\alpha} e_{\rho\rho} + 2\mu e_{\alpha\beta} e_{\alpha\beta}) dx + \frac{1}{\varepsilon^2} \int_{\Omega} (2\lambda e_{\rho\rho} e_{33} + 4\mu e_{\alpha 3} e_{\alpha 3}) dx + \frac{1}{\varepsilon^4} \int_{\Omega} (\lambda + 2\mu) e_{33} e_{33} dx, \tag{13}$$

where the summation is over the set $\{1, 2\}$, $\Omega = \mathcal{O} \times [-1, +1]$, λ, μ are Lamé coefficients, ε is the thickness of the plate, and $e_{ij} = \frac{1}{2}(\partial u_i / \partial x_j + \partial u_j / \partial x_i)$ are the components of the scaled linear strain tensor.

By considering functions that are constant in x_3 , it becomes clear that if the decomposition $V = V_0 + \dots + V_m$ is such that a function $u \in V$ is split into components where some components are zero on the lower surface $\mathcal{O} \times \{-\varepsilon\}$ and some zero on the upper surface $\mathcal{O} \times \{+\varepsilon\}$, the constant c_0 in (6) will grow at least as ε^{-4} for $\varepsilon \rightarrow 0$ because of the last term in (13). Consequently, the decomposition must not decouple the problem in the x_3 direction. This can be achieved by considering the using nodes opposite each other on the upper and lower surface to be one *supernode*, containing only the edge between the nodes. Similarly, a lower and upper edge and the face between them form a supernode, and the upper and lower face and the interior of an element are also considered one supernode.

Our algorithms recognize a thin element in a single layer automatically and preprocess such element by merging the vertices, edges, and faces on top and bottom surfaces as described above. The element is then further treated as a 2D entity employing the next strategy.

2.3.7. 2D partial orthogonalization with vertex spaces

In this strategy, there is a vertex space for each vertex consisting of functions orthogonalized to all adjacent edges and interiors. The edge spaces for each edge are orthogonalized to adjacent interiors. This is the strategy used for thin elements after dimensional reduction (Section 2.3.6).

Note that for 2D elements, the difference between the overlapping strategy (Section 2.3.5) and the 2D partial orthogonalization with vertex spaces is essentially that the latter decouples adjacent edges. In the overlapping strategy, the functions on adjacent edges are contained in a common subspace. So, the partial orthogonalization method is computationally less expensive than the overlapping strategy but the condition numbers are higher. The coarse space is chosen as follows.

2.3.8. Quadratic and anisotropic coarse for thin elements

From the asymptotic expansion of the energy of the thin element (13), it follows that a kink in the transversal displacement field has energy proportional to l/ε^2 , since it introduces shear. To assure continuous derivative of the coarse component v_0 in (6), the coarse space should contain at least cubic functions on the element edges; however, in practice, we have found the quadratic coarse space satisfactory.

For large problems, like modeling of the whole aircraft, the quadratic coarse space results in too big a coarse problem. The key to further improvement is to realize that what we really need in the coarse space are only functions that describe the transversal component of the deformation at the midplane. The orientation of the thin element is determined automatically and such coarse functions are chosen as linear combinations of the quadratic and cubic shape functions with the coefficients computed based on the local stiffness matrix data.

2.3.9. Overlap chains

The dimensional reduction strategy (Section 2.3.6) fails when the thin elements occur in more than one layer. Fortunately, the overlapping strategy from Section 2.3.5 can be used here. For each vertex in

the thin element, define the space spanned by all vertex functions and all adjacent edge and face functions. Then merge these spaces for every pair of vertices that are across the thickness.

Similarly as in Section 2.3.6, merging the upper and lower vertex spaces will again prevent the decomposition of a function constant through the thickness into two components that would have energy of the order $1/\epsilon^4$. For a detailed related argument for 2D thin elements and scalar problems, cf. Mandel and Lett [23].

Since every vertex belongs to several elements, merging the vertex spaces results in larger spaces associated with chains of vertices through the thicknesses of the elements. Because of this 1D topology, the linear system associated with the solution of the local problem (3) in such space is banded. This strategy is in fact less expensive than the original overlapping strategy, since the overlapping systems along the chains were replaced by one banded system, so it is advantageous to merge the vertex spaces in 1D chains even if this is not necessary because of the energy considerations above.

3. Numerical results

This section contains the results of tests on several difficult problems. In all cases, the stopping criterion was that the estimated relative error in energy norm, the relative maximum residual, and the relative Euclidean residual norm are all less than 10^{-4} .

Our first test problem was the skin of an aircraft (Fig. 1, Table 1). This model was created in STRIPE

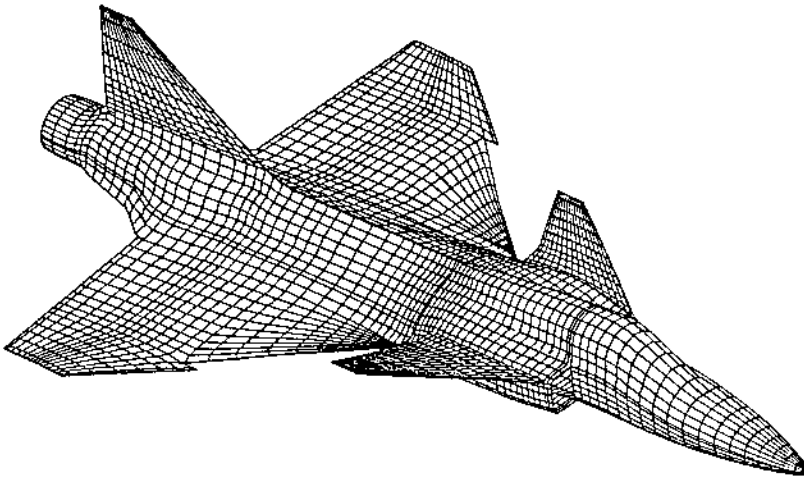


Fig. 1. Aircraft skin.

Table 1
Half aircraft skin, 4986 elements (Fig. 1)

| <i>p</i> | <i>NDOF</i> | Stiffness | | Direct | | Iterative solver | | | |
|----------|-------------|-----------|--------|---------|--------|------------------|------|------|------|
| | | Disk | CPU | Disk | CPU | Iter | Disk | CPU | Wall |
| 6 | 793590 | 1893 | (0.54) | (27498) | (445) | 109 | 1768 | 13.5 | 40 |
| 8 | 1628547 | 6322 | (2.32) | (81630) | (2087) | 125 | 5656 | 50.8 | 115 |

Disk space in MB, time in hours IBM RS/6000 550 128 MB RAM.

Direct solver estimate by STRIPE (skyline).

Stiffness matrix–vector multiplication by STRIPE.

and it consists of a single layer of thin hexa elements. One difficulty of this model for a preconditioner is the large proportion of skewed elements. The stiffness matrix was not stored; rather, fast quadrature in STRIPE was used to compute the matrix–vector products.

The second test problem (Table 2) demonstrates the performance of our solver on a realistic model of a laminated plate with a circular opening. This model was created in STRIPE and the results used by Andersson [1] for highly accurate computation of delamination stresses. The finite element mesh is highly graded in the radial and vertical directions (ratio about 1:6) when approaching the singularities at the edges of the material interface and the opening. The fiber plies themselves are modeled as anisotropic material. The directions of the fibers vary between the plies. The stiffness matrix was stored on a disk as a collection of local stiffness matrices.

The third test problem (Table 3) demonstrates the performance on a large problem solved on CRAY C-90. We have used 4 processors of the C-90 to utilize efficiently the time when the process is in memory, reduced the amount of memory to 16 Mw data area to increase the scheduling priority of the job, and linked the code with the CRI FFIO library to obtain disk striping and an intelligent 8 Mw cache for the sustained data transfer rate of about 35 MB/s. In our experience, this is the best strategy for short wall clock times in a multiuser environment. The parallelism used was on loop level and in BLAS routines. The average number of CPUs executing concurrently was 2.1 in numerically intensive parts of the code and 1.4 overall. It should be noted that parallel efficiency could only be measured from a run on a dedicated machine, since in a multiuser environment, the number of CPUs actually available to the job is not known. The model was a lap joint discretized by hexa elements in STRIPE. The aspect ratios of the elements were reasonable but about half of the elements were curved. The stiffness matrix was stored on a disk as a collection of local stiffness matrices.

The last model was created in MECHANICA to investigate performance for a problem with thin wedge elements and variable p (Fig. 2, Table 4). The model consists of layers of very thin elements, obtained by adding a third dimension to a 2D mesh of quadrilaterals and triangles. Shape functions of highest polynomial degrees are concentrated only in some areas of the model. The stiffness matrix–vector multiplication was done in MECHANICA, with the local stiffness matrices stored in a database and cached in memory.

Table 2
Laminated plate with a circular hole, 4886 elements

| p | NDOF | Stiffness | Iterative solver | | |
|-----|--------|-----------|------------------|---------|---------|
| | | MB | Iter | Disk MB | CPU (s) |
| 6 | 223872 | 710.8 | 72 | 1535 | 1310 |

Uniform p . Highly graded mesh, composite fiber material with 28 plies.
CRAY-YMP, one processor.

Table 3
Lap joint, 3504 elements

| p | NDOF | Stiffness | Iterative solver | | | |
|-----|--------|-----------|------------------|---------|---------|----------|
| | | MB | Iter | Disk MB | CPU (s) | Wall (s) |
| 6 | 590052 | 1330 | 41 | 979 | 1276 | 4444 |

CPU time total of 4 processors of CRAY-C90, multiuser batch mode, 40 Mw partition.
I/O with disk striping and intelligent cache by CRI FFIO library.

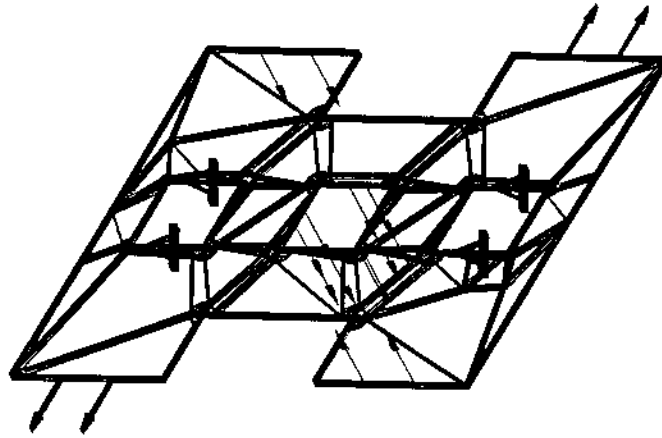


Fig. 2. H-Shaped core.

Table 4
H-Shaped core

| max p | NDOF | Stiffness | | Iterative solver | | | | |
|---------|-------|-----------|---------|------------------|------|---------|---------|----------|
| | | MB | Disk MB | CPU (s) | Iter | Disk MB | CPU (s) | Wall (s) |
| 4 | 15654 | 18.4 | | | 106 | 31.5 | 332.4 | 411.8 |
| 6 | 26313 | 39.5 | 217 | 1450 | 63 | 49.9 | 506.5 | 912.2 |

590 elements in 5 layers with aspect ratio 1:500:500. adaptive p .

Disk space in MB, time in s IBM RS/6000 250 80 MB RAM.

Stiffness matrix–vector multiplication in MECHANICA.

Direct solver by optimized skyline.

Acknowledgments

This research was done at Solvers International, Inc., supported by the National Science Foundation under grant SBIR DMI-9360015. A part of the work was done under a consulting contract with the Aeronautic Research Institute of Sweden. CRAY computer time was provided by CRAY Research, Inc. Data from STRIPE were provided and computations with STRIPE done by Dr. Börje Andersson at the Aeronautic Research Institute of Sweden. The problem in Fig. 2 was created in MECHANICA by Mr. Sanjay Kumar, who has also assisted with programming and computations. The package MECHANICA was provided by the RASNA Corporation.

References

- [1] B. Andersson, Stress σ_z at hole boundary in 28 ply composite plate under compression, Technical Report FFAPH-1240, The Aeronautical Research Institute of Sweden, PO Box 11021, S-161 11 Bromma, Sweden, October 1994.
- [2] I. Babuška, A.W. Craig, J. Mandel and J. Pitkäranta, Efficient preconditioning for the p -version finite element method in two dimensions, *SIAM J. Numer. Anal.* 28 (1991) 624–662.
- [3] I. Babuška, M. Griebel and J. Pitkäranta, The problem of selecting the shape functions for a p -type finite element, *Int. J. Numer. Methods Engrg.* 28 (1989) 1891–1908.
- [4] P.E. Bjørstad and J. Mandel, Spectra of sums of orthogonal projections and applications to parallel computing, *BIT* 31 (1991) 76–88.
- [5] J.H. Bramble, J.E. Pasciak and A.H. Schatz, The construction of preconditioners for elliptic problems by substructuring, I, *Math. Comput.* 47 (1986) 103–134.
- [6] P.G. Ciarlet, *Plates and Junctions in Elastic Multi-Structures* (Masson and Springer-Verlag, Paris, 1990).
- [7] M. Dryja and O.B. Widlund, Towards a unified theory of domain decomposition algorithms for elliptic problems, in: T. Chan, R. Glowinski, J. Périaux and O. Widlund, eds., 3rd Int. Symp. on Domain Decomposition Methods for Partial Differential Equations, held in Houston, Texas, March 20–22, 1989, SIAM, Philadelphia, PA, 1990.

- [8] M. Dryja and O.B. Widlund, Domain decomposition algorithms with small overlap, *SIAM J. Sci. Comput.* 15 (1994) 604–620.
- [9] S. Foresti, G. Brussino, S. Hassanzadeh and V. Sonnad, Multilevel solution method for the p -version of finite elements, *Comput. Phys. Comm.* 53 (1989) 349–355.
- [10] S. Foresti, S. Hassanzadeh, H. Murakami and V. Sonnad, A comparison of preconditioned iterative techniques using rapid operator application against direct solution methods, *Int. J. Numer. Methods Engrg.* 32 (1992) 1137–1144.
- [11] G.H. Golub and C.F. Van Loan, *Matrix Computations*, 2nd edition (John Hopkins University Press, 1989).
- [12] W. Hackbusch, *Multigrid Methods—Theory and Applications* (Springer-Verlag, Berlin, 1986).
- [13] M.R. Hestenes and E. Stiefel, Methods of conjugate gradients for solving linear systems, *J. Res. Nat. Bur. Stand.* 49 (1952) 409–436.
- [14] G.S. Lett, Iterative solver for the p -version finite element method with thin rectangular elements, Ph.D. Thesis, Department of Mathematics, University of Colorado at Denver, 1990.
- [15] P.L. Lions, On the Schwarz alternating method. I., in: R. Glowinski, G.H. Golub, G.A. Meurant and J. Périaux, eds., *First Int. Symp. on Domain Decomposition Methods for Partial Differential Equations*, SIAM, Philadelphia, PA, 1988.
- [16] J. Mandel, Domain decomposition on unstructured subdomains, in: A. Quarteroni et al., eds., *Domain Decomposition Methods in Science and Engineering*, Proc. 6th Int. Symp. on Domain Decomposition Methods, Como, Italy, 1992 (American Mathematical Society, Providence, RI, 1994) 103–112.
- [17] J. Mandel, Hierarchical preconditioning and partial orthogonalization for the p -version finite element method, in: T.F. Chan, R. Glowinski, J. Périaux and O.B. Widlund, eds., *Third Int. Symp. on Domain Decomposition Methods for Partial Differential Equations*, SIAM, Philadelphia (1990) 141–156.
- [18] J. Mandel, Iterative solvers by substructuring for the p -version finite element method, *Comput. Methods Appl. Mech. Engrg.* 80 (1990) 117–128; *Int. Conf. on Spectral and High Order Methods for Partial Differential Equations*, Como, Italy, June 1989.
- [19] J. Mandel, Two-level domain decomposition preconditioning for the p -version finite element method in three dimensions, *Int. J. Numer. Methods Engrg.* 29 (1990) 1095–1108.
- [20] J. Mandel, Adaptive iterative solvers in finite elements, in: M. Paradrakakis, ed., *Solving Large Scale Problems in Mechanics. The Development and Application of Computational Solution Methods* (J. Wiley & Sons, London, 1993) 65–88.
- [21] J. Mandel, Intelligent block iterative methods, in: J. Robinson, ed., *FEM Today and the Future*, Robinson and Associates, Okhampton, Devon EX20 4NT, England, 1993, pp. 471–477; *Proc. 7th World Congress on Finite Elements*, Monte Carlo, November 1993.
- [22] J. Mandel, Iterative solver for p -version finite elements in three dimensions, *Comput. Methods Appl. Mech. Engrg.* 116 (1994) 175–183; *ICOSAHOM 92*, Montpellier, France, June 1992.
- [23] J. Mandel and G.S. Lett, Domain decomposition preconditioning for p -version finite elements with high aspect ratios, *Appl. Numer. Math.* 8 (1991) 411–425.
- [24] S.F. McCormick, ed., *Multigrid Methods*, *Frontiers in Applied Mathematics*, Vol. 3 (SIAM, Philadelphia, 1987).
- [25] W.F. Mitchell, Optimal multilevel iterative methods for adaptive grids, *SIAM J. Sci. Stat. Comput.* 13 (1992) 146–167.
- [26] R.B. Morris, Y. Tsui and P. Carnevali, Adaptive solution strategy for solving large systems of p -type finite element equations, *Int. J. Numer. Methods Engrg.* 33 (1992) 2059–2069.
- [27] J.T. Oden, A. Patra and Y. Feng, Parallel domain decomposition solver for adaptive hp finite element methods, Technical Report TICIAM 94-11, University of Texas at Austin, 1994.
- [28] L.F. Pavarino, Additive Schwarz methods for the p -version finite element method, *Numer. Math.* 66 (1994) 493–515.
- [29] L.F. Pavarino, Some Schwarz algorithms for the p -version finite element method, in: Y.A. Kuznetsov, J. Périaux, A. Quarteroni and O.B. Widlund, eds., *Domain Decomposition Methods in Science and Engineering: 6th Int. Conf. on Domain Decomposition*, AMS, 1993.
- [30] L.F. Pavarino and O.B. Widlund, A polylogarithmic bound for an iterative substructuring method for spectral elements in three dimensions, Technical Report 661, Courant Institute of Mathematical Sciences, Department of Computer Science, March 1994, *SIAM J. Numer. Anal.*, in press.
- [31] J. Pitkäranta, The problem of membrane locking in finite element analysis of cylindrical shells, *Numer. Math.* 61 (1992) 523–542.
- [32] E.M. Rønquist and A.T. Patra, Spectral element multigrid. I. Formulation and numerical results, *J. Sci. Comput.* 2(4) (1987) 389–406.
- [33] B.F. Smith, A domain decomposition algorithm for elliptic problems in three dimensions, *Numer. Math.* 60 (1991) 219–234.
- [34] B.A. Szabó and I.M. Babuška, *Finite Element Analysis* (John Wiley and Sons, New York, 1991).
- [35] O.B. Widlund, Optimal iterative refinement methods, in: T. Chan, R. Glowinski, J. Périaux and O. Widlund, eds., *Domain Decomposition Methods*, SIAM, Philadelphia, PA, 1989.
- [36] J. Xu, Iterative methods by space decomposition and subspace correction, *SIAM Rev.* 34 (1992) 581–613.