INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS, VOL. 17, 1019–1033 (1993)

ACCURACY AND CONVERGENCE OF ELEMENT-BY-ELEMENT ITERATIVE SOLVERS FOR INCOMPRESSIBLE FLUID FLOWS USING PENALTY FINITE ELEMENT MODEL

M. P. REDDY* AND L. G. REIFSCHNEIDER†
Technalysis, 7120 Waldemar Dr. Indianapolis, IN 46268, U.S.A.

J. N. REDDY‡
Department of Mechanical Engineering, Texas A&M University, College Station, TX 77843, U.S.A.

AND

H. U. AKAY§
Department of Mechanical Engineering, Indiana University-Purdue University at Indianapolis, Indianapolis, IN 46202, U.S.A.

SUMMARY

The ability of two types of Conjugate Gradient like iterative solvers (GMRES and ORTHOMIN) to resolve large-scale phenomena as a function of mesh density and convergence tolerance limit is investigated. The flow of an incompressible fluid inside a sudden expansion channel is analysed using three meshes of 400, 1600 and 6400 bilinear elements. The iterative solvers utilize the element-by-element data structure of the finite element technique to store and maintain the data at the element level. Both the mesh density and the penalty parameter are found to influence the choice of the convergence tolerance limit needed to obtain accurate results. An empirical relationship between the element size, the penalty parameter, and the convergence tolerance is presented. This relationship can be used to predict the proper choice of the convergence tolerance for a given penalty parameter and element size.

KEY WORDS Convergence behaviour Element-by-element data structure GMRES ORTHOMIN Mesh density Penalty function model

INTRODUCTION

Background

Finite element analysis of practical engineering problems requires discretizing the problem domain into a finely divided mesh. The finite element approximation of the governing equations results in a system of algebraic equations. Solving the system of equations is the most time-consuming part of a numerical analysis and accounts for nearly 80 per cent of the CPU time and

* Current address: Sr. Research Engineer, Computational Mechanics Company, Inc., Austin, TX, U.S.A.
† Research & Development Engineer
‡ Oscar S Wyatt, Jr. Chair Professor
§ Professor

0271–2091/93/241019–15$12.50

Received 19 March 1992
Revised 10 March 1993

storage. In general the system of equations can be solved using either direct elimination methods or iterative methods.

Direct solvers like the Gauss elimination methods provide the solution to the system of equations after a fixed number of steps and are less sensitive to the conditioning of the matrix. However, the main deficiencies of direct solvers become clear when used to solve a large system equations because they require the matrix to be stored in a ordered format. Moderately large problems can be solved using refined direct solvers such as frontal solver, skyline solver, and others (see Reference 1). These refined direct solvers demand out-of-core storage of the equations and hence require large data transfer. The limitations on CPU time and storage requirements make the use of direct solvers uneconomical and impractical to solve a complex problem with more than quarter million equations.

Iterative solvers, on the other hand, require less storage and CPU time while giving comparable accuracy of the solution for a large system of equations. For these solvers the global matrix formulation is avoided. Here, the major operation is the matrix–vector multiplication instead of the matrix inversion or elimination used in direct solvers. However, the drawbacks of the iterative solvers are that the accuracy depends on the convergence parameters used and the convergence rate depends on the condition number (ratio of the largest to the smallest eigenvalue) of the matrix. These drawbacks discouraged active use of iterative solvers for last few decades. In spite of these difficulties, the demand for numerical solution of large, complex problems has renewed interest in the iterative solvers due to the CPU-intensive nature of direct solvers. Much of the current study of the iterative solvers involves improvements to the convergence and accuracy of various solvers.

Among the various iterative methods, the Conjugate Gradient (CG) method[2] is most widely used because it is a finite step method (i.e. apart from round-off errors, solution is achieved in a fixed number of iterations). The convergence of the conjugate gradient method, and iterative methods in general, can be improved by preconditioning and/or scaling the equations.[3,4]

The storage requirements for the iterative solvers can be further reduced by solving the equations at element level, i.e. using the Gauss–Seidel iteration method for the set of variables associated with the element. An advantage of doing calculations at element level is that assembly of element matrices to form the global coefficient matrix is eliminated. This idea of using the Element-By-Element (EBE) data structure of the coefficient matrix was first pointed out by Fox and Stanton[5] and Fried.[8] In this method the matrix–vector multiplications are carried out at the element level and the assembly is carried out on the resultant vector. The advantage of using the EBE data structure of the finite element mesh becomes apparent when solving large problems, because the matrix–vector multiplication can be done in parallel on series of processors.

Several different algorithms have been developed which use the element-by-element data structure with some type of conjugate gradient method (such as ORTHOMIN, ORTHORES[4] or GMRES[7]). The algorithms are described in detail by Hayes[8] and Wathen.[9] Iterative solution methods using the EBE data structure are proposed for solving problems in solid mechanics,[10,11] heat transfer,[12] compressible fluid flows,[13–16] and incompressible fluid flows using stream function–vorticity formulation and primitive variable formulations.[17–20] In majority of these articles, the savings in computer run time and disk storage by using iterative solvers are highlighted without discussing the sensitivity of the solver to mesh density.

Recently, Reddy *et al.*,[21,22] analysed incompressible fluid flow problems in primitive variables using the EBE data structure of the finite elements and iterative solvers (GMRES, ORTHOMIN and ORTHORES). The problem of having a large penalty parameter was overcome by using an iterative penalty function method. The EBE scheme with iterative penalty function approach was found to be very efficient and accurate for a wide range of problems. The accuracy of the procedure and the iterative solvers was determined by comparing the results with an efficient direct solver (e.g. frontal solver). A least-square finite element model for pressure was also presented there (see References 21–24 for details).

The convergence rate of iterative solvers decreases significantly after resolving the Fourier components of the errors whose wavelengths are comparable to the size of the mesh. Therefore, for a specified convergence tolerance limit, the accuracy of a given method and the time required to satisfy the convergence criterion depends on its behaviour as a function of mesh density. Mesh density sensitivity information for an iterative solution method can be provided by mesh refinement studies. Mesh refinement studies are expensive for three-dimensional problems, and one quickly exhausts the computer resources. In addition, validation of the results becomes difficult due to lack of experimental results or inability to perform similar studies using more robust direct solvers. However, such studies can be carried out on two-dimensional problems.

Present study

Mesh refinement studies comparing the behaviour of different iterative solvers are not available in literature at the time of this work. In this paper, we report the results of mesh refinement studies for two different iterative solvers (GMRES and ORTHOMIN). The accuracy of the GMRES and the ORTHOMIN solvers as a function of mesh density and convergence criterion is examined. This study is confined to two-dimensional problems because of the aforementioned reasons. However, the conclusions made in this study can be generalized and are applicable for three-dimensional problems.

Accuracy of the iterative solvers is determined by comparing the results with the available values in literature and also with those obtained by solving the system of equations using a refined direct solver (frontal solver). Direct solvers and their refined versions are very efficient in analysing two-dimensional problems. On the other hand, the iterative solvers require more CPU time than the direct solvers and this discourages the use of iterative solvers for two-dimensional problems. Since the main objective of this investigation is to determine the sensitivity of two CG like iterative solvers to the mesh density, we are not concerned with the amount of CPU time required by each solver to provide acceptable converged solution.

The governing equations and the solution procedure are explained in the next section. The effect of mesh density and convergence criterion on the resolution of large-wavelength errors (large-scale phenomenon) is examined by using GMRES and ORTHOMIN solvers. Two-dimensional flow of incompressible fluid over a backward facing step at a Reynolds number of 60 is used as a test problem.

GOVERNING EQUATIONS

Conservation equations

The laws governing the flow of isothermal, incompressible Newtonian fluids are the basic laws of conservation of mass and momentum. These can be expressed analytically in terms of the velocity field and pressure in a rectangular Cartesian co-ordinate system (x_1, x_2, x_3):

$$\frac{\partial u_i}{\partial x_i}=0 \quad \text{in } \Omega, \tag{1}$$

$$\rho\left(\frac{\partial u_i}{\partial t}+u_j\frac{\partial u_i}{\partial x_j}\right)=\frac{\partial \sigma_{ij}}{\partial x_j}-\rho g_i \quad \text{in } \Omega, \tag{2}$$

where Ω is a two-dimensional domain, u_i is the velocity vector, ρ is the mass density of the fluid, g_i is the gravitational acceleration, σ_{ij} is the total stress tensor. This work examines a steady-flow problem, so the time derivative of the velocities is dropped from equation (2).

The constitutive equation for the total stress tensor, σ_{ij}, is given by

$$\sigma_{ij}=\sigma'_{ij}-P\,\delta_{ij}, \tag{3}$$

where σ'_{ij} are the components of the deviatoric stress tensor and P is the pressure. The components of the deviatoric stress tensor can in turn be expressed as

$$\sigma'_{ij}=2\eta D_{ij}, \tag{4}$$

where

$$D_{ij}=\frac{1}{2}\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right) \tag{5}$$

are the components of the rate of strain tensor and η is the viscosity, which is, in general, a function of temperature and rate of strain.

Boundary conditions

To complete the set of equations, equations (1) and (2) need to be combined with an appropriate set of boundary conditions. For the momentum equations, the velocities (essential boundary condition) or the surface traction (natural boundary condition) must be specified along the boundary Γ of the flow domain Ω. The boundary conditions on the velocity and stress are mutually exclusive. Let Γ_u denote the boundary on which the velocities are specified, and Γ_σ be the boundary on which the stresses are specified. We have $\Gamma=\Gamma_u\cup\Gamma_\sigma$.

$$u_i=\bar{u}_i \quad \text{on } \Gamma_u; \quad \text{or} \quad t_i=\sigma_{ij}n_j \quad \text{on} \quad \Gamma_\sigma, \tag{6}$$

where n_j denote the components of the outward unit normal on the boundary Γ. Note that pressure enters the natural boundary conditions through the total stress components.

COMPUTATIONAL ASPECTS

Formulation

The governing equations (1) and (2) are solved here using a penalty finite element model. In the penalty function formulation, the incompressibility constraint is used as a constraint on the velocity fields (see Reference 25 for details). This approach results in replacing the pressure with the expression

$$P=-\lambda\frac{\partial u_i}{\partial x_i} \tag{7}$$

and omitting the equation of continuity, equation (1). Here λ denotes the penalty parameter. For increasingly larger values of λ, the continuity equation is satisfied more accurately. The advantage of the penalty function formulation is that the pressure does not enter the formulation as a primary unknown variable. This penalty finite element model is termed the traditional penalty model, and it is solved here using the frontal solver.

The convergence rate of iterative solvers depends on the condition number of the coefficient matrix. For large values of penalty parameter, the iterative solvers perform poorly. This difficulty can be overcome by using an iterative penalty model (see Reference 21 for details). When using iterative penalty model, the pressure at nth iteration is written as

$$P^{n+1}=-\lambda\frac{\partial u_i}{\partial x_i}+P^n. \tag{8}$$

A value of 10^8 is used for λ in the present analysis for the frontal solver and a value of 10^4 is used for iterative solvers. A smaller value of penalty parameter is sufficient because the continuity equation is satisfied gradually in iterative penalty model.

Solution strategies

The penalty finite element formulation of equations (1) and (2) results in the element equations of the form

$$K(\Delta)\Delta=F, \tag{9}$$

where the coefficient matrix K is a non-linear function of nodal velocity components, $\Delta=(\mathbf{u}, \mathbf{v})^{\mathrm{T}}$. The non-linear equations (9) are solved using the Picard type iterative method. The type of convergence criterion used is

$$\left\{\frac{\sum_{i=1}^{N}|\Delta_i^{n+1}-\Delta_i^n|^2}{\sum_{i=1}^{N}|\Delta_i^n|^2}\right\}^{1/2}\leq\varepsilon, \tag{10}$$

where Δ_i^n denotes the nodal value at node i and iteration n, ε is the tolerance (say 0·01) and N is the number of equations.

Pressure can be calculated from either equation (7) or equation (8). However, the pressures calculated by these equations exhibit oscillatory behaviour as they depend on the error in the divergence-free condition. In order to suppress these oscillations, the pressure is calculated by recasting the momentum equations into a least-square type finite element model with the pressure as the unknown and the velocities known from the solution of (9):

$$[K^*]\{P\}=\{F^*\}. \tag{11}$$

This method is found to yield more accurate results for pressure in a wide variety of problems.

Four-noded bilinear elements are used for $\mathbf{u}$, $\mathbf{v}$ and P in this study. The linearized system of equations (9) and pressure equations (11) are solved using both frontal and iterative solution methods. For iterative solvers, the convergence criterion for velocity and pressure were taken to be $\varepsilon=10^{-5}$, unless stated otherwise. In this study, the pressure is computed at every non-linear iteration when the iterative solvers are used, whereas it is computed in the frontal solver after the solution for velocities is converged.

The iterative solvers implemented herein exploit the EBE data structure of the finite element mesh. The iterative solvers GMRES and ORTHOMIN are used here. These methods are generalizations of the CG algorithms for non-symmetric systems and may also be regarded as some variants of the method of Lanczos. The ORTHOMIN solver minimizes the energy norm of the system of equations. On the other hand, the GMRES solver minimizes the residual norm. The solution algorithms for these two iterative solvers can be found in References 4 and 7.

To improve the convergence of the iterative solvers the coefficient matrix and the force vectors are scaled using diagonal (Jacobi) preconditioning. Other levels of preconditioning (multilevel) are also possible.[16] The Krylov space in GMRES is set to 50. The program was run on IBM/RS6000 530 Power Station.

NUMERICAL RESULTS

Model problem solved

The laminar, incompressible flow through a channel with sudden expansion is used as a test problem for this study. The problem geometry and the flow conditions are given in Figure 1. The widths of the channel inlet and outlet sections are one unit and two units, respectively. The length of the channel is equal to six units. The sudden expansion is located two units downstream from the inlet. The Reynolds number of 60 is based on the maximum inlet velocity (U) and the inlet width. A fully developed, parabolic velocity profile is specified at the inlet and the zero traction boundary conditions are imposed at the exit. No slip boundary conditions for velocity are imposed at the solid walls. A reference zero pressure is specified at the exit. A finite element solution for this problem is given by Lee *et al.*[26] An interesting feature of this flow is the separation at the expansion and the reattachment downstream of the expansion. At higher Reynolds numbers, multiple separation regions will form and eventually the flow becomes unsteady. The reattachment length depends on the Reynolds number of the flow.

The first portion of this section will discuss in detail the observed interaction of the element size (Δh), the convergence criterion (ε), and the penalty parameter (λ) on the convergence rate and the accuracy of the conjugate gradient solvers. In the second part of this section, an empirical relationship between δh, ε and λ is presented.

Interaction and choice of δh, ε and λ

The velocity field and pressure distribution are computed for three different meshes (400, 1600 and 6400 elements) using both direct (frontal solver) and iterative solver (EBE/GMRES and EBE/ORTHOMIN) solvers. The finite element mesh for the coarsest mesh (400 elements: $\delta x = 0{\cdot}125$ and $\delta y = 0{\cdot}1$) is shown in Figure 2. The same element aspect ratio is maintained for the refined meshes. Namely, the element dimensions for 1600 elements mesh are half that of 400 element mesh. Similarly, the element dimensions for 6400 element mesh are half that of 1600

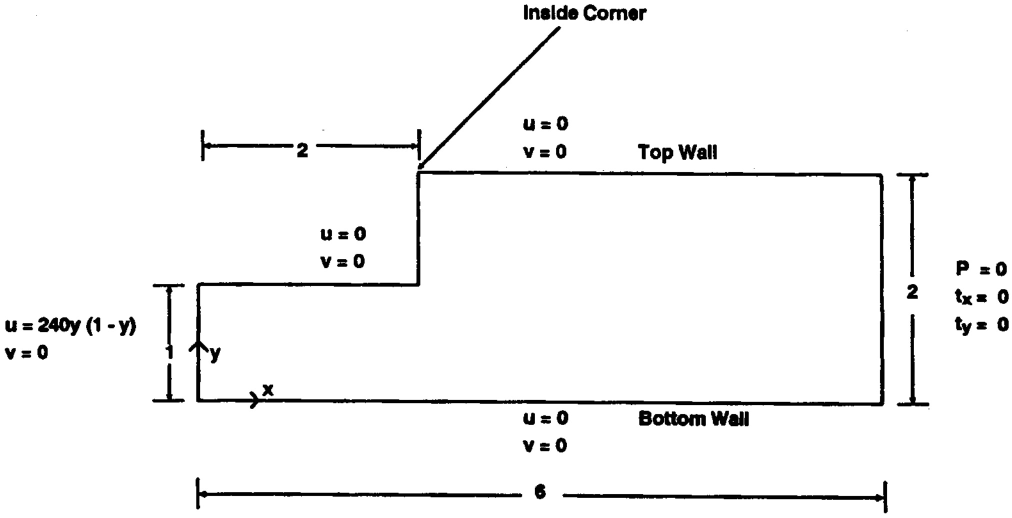

Figure 1. Schematic diagram and the boundary conditions used for expansion flow problem

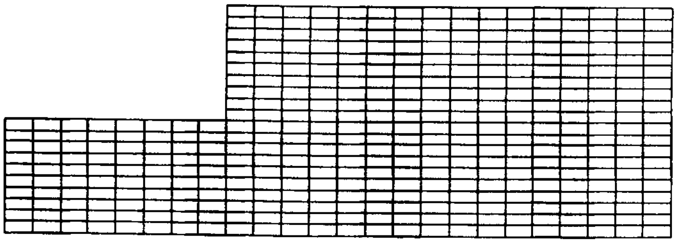

Figure 2. Coarsest finite element mesh (400/445 elements/nodes) for expansion flow problem. In addition, 1600/1689 and 6400/6577 elements/nodes meshes were used

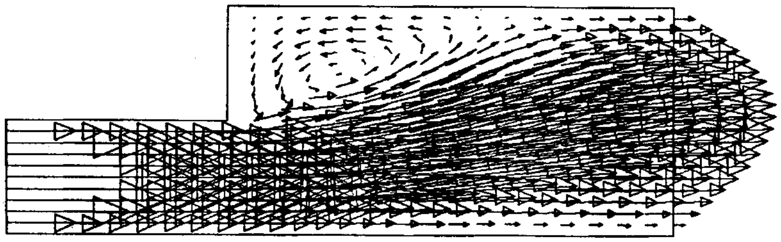

Figure 3. Representative velocity solution for sudden expansion flow problem (Reynolds number = 60). Sudden expansion at $x=2$

element mesh. The results for the iterative solvers are compared with both those given in Reference 26 and those from the frontal solver using the finest mesh (6400 elements).

A representative velocity solution for the sudden expansion flow problem is shown in Figure 3.

The pressure distribution along the bottom wall of 400 and 1600 element meshes with iterative solvers are plotted in Figure 4. The iterative solution results were obtained using $\varepsilon=10^{-5}$. The pressure profiles for the coarse meshes are compared to the fine mesh results from the direct solver and the results from Lee *et al.*[26] Lee *et al.* used a nine-noded quadratic elements to obtain the velocity field while four-noded elements were used in this study. All results in Figure 4 exhibit the same behaviour observed by Lee *et al.* The major differences in the pressure profiles are noticed in the inlet section, with the coarsest mesh results showing the poorest agreement. These differences are attributed to the error in the way the smoothing algorithm computes the velocity gradient at the boundaries and corner nodes. However, the accuracy in the inlet pressure distribution is seen to improve by increasing the mesh density. The pressure gradient for a fully developed flow between parallel plates is a function of wall shear stress. Therefore, by refining the mesh, the wall shear stresses are computed more accurately and this improves the pressure solution. In other words, the wall shear stress is a small-scale phenomenon and therefore mesh refinement produces more accurate solution.

When using a direct solver, mesh refinement produces a more accurate solution that captures both the large-scale and the small-scale phenomena. However, when using iterative solvers, mesh refinement does not necessarily produce more accurate results for complex flows. This solution behaviour is counter-intuitive to conventional numerical methods. Examples of this behaviour are demonstrated in the remainder of this paper.

The first example of the unique convergence behaviour of iterative solvers is illustrated in Figure 4. Here, the pressure solution near the exit for the 1600 element mesh with the GMRES solver ($\varepsilon=10^{-5}$) is seen to differ slightly from the 400 element results. This indicates the loss of

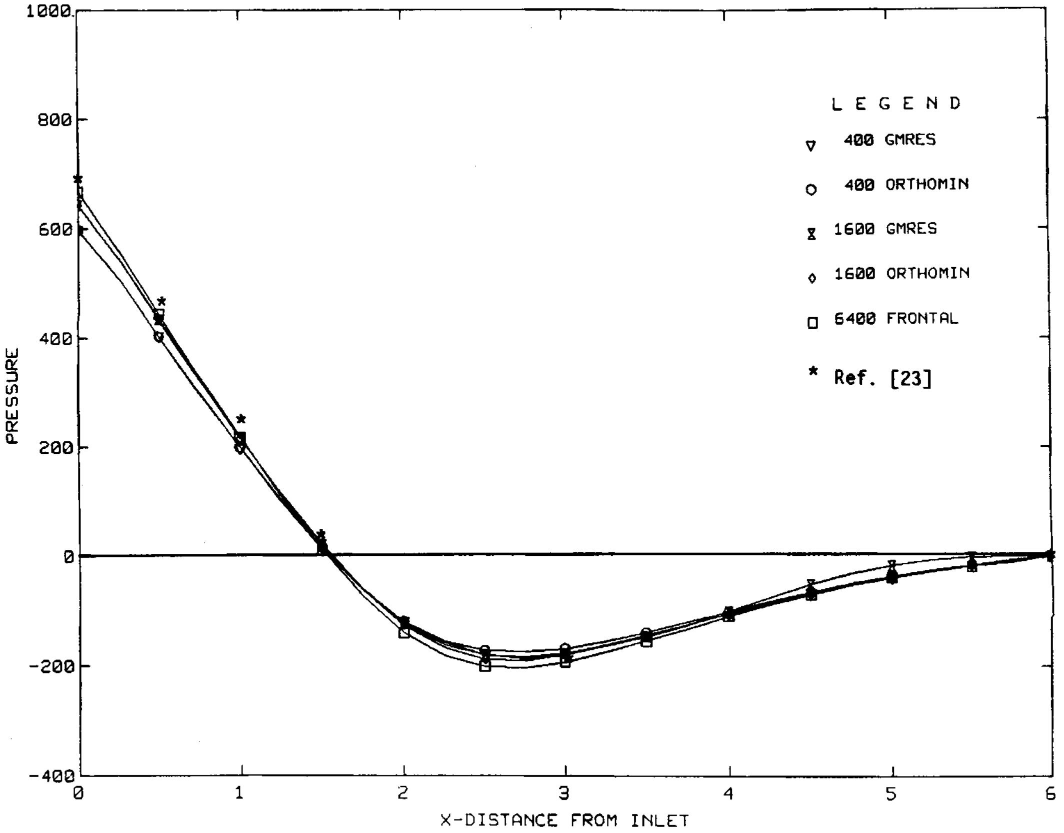

Figure 4. The predicted pressure distributions along the bottom wall using GMRES and ORTHOMIN solvers for 400 and 1600 element meshes ($\varepsilon = 10^{-5}$). The relative accuracy of iterative solvers is determined by comparing with the frontal solver results (6400 elements mesh) and values from Reference 26

accuracy by refining the mesh for a fixed convergence tolerance. The error in the GMRES solution is more apparent and occurs immediately downstream of the expansion for the finer mesh as seen in Figure 5. In Figures 5–8, the legend contains the mesh size, the solver and the convergence tolerance. The pressure profile for the GMRES solver with $\varepsilon = 10^{-5}$ in Figure 5 indicates that the flow reached a nearly fully developed state. However, this is correct only for creeping flows with a very small Reynolds number. The results indicate that the effect of the convective terms (large-scale phenomenon) is not captured by the GMRES solver when a fine mesh is used with $\varepsilon = 10^{-5}$. By using a tighter convergence criterion for the residual calculations of the GMRES solver, $\varepsilon = 10^{-6}$, this problem was overcome as shown in Figure 5. The pressure distribution obtained using ORTHOMIN solver with both $\varepsilon = 10^{-5}$ and $\varepsilon = 5 \times 10^{-6}$ shows good agreement with the frontal solver results. The ORTHOMIN results are better than the GMRES results for a fixed residual solver with both $\varepsilon = 10^{-5}$ and $\varepsilon = 5 \times 10^{-6}$ showing good agreement with the frontal solver results. The ORTHOMIN results are better than the GMRES results for a fixed residual tolerance, ε. It is evident from Figure 5 that the resolution of the large wavelength error components is mesh and solver dependent.

The coefficient of friction ($\tau_w/0{\cdot}5\,\rho U^2$) along the top wall for the 1600 element mesh is plotted in Figure 6, where τ_w is the wall shear stress. The iterative solver results are compared with the

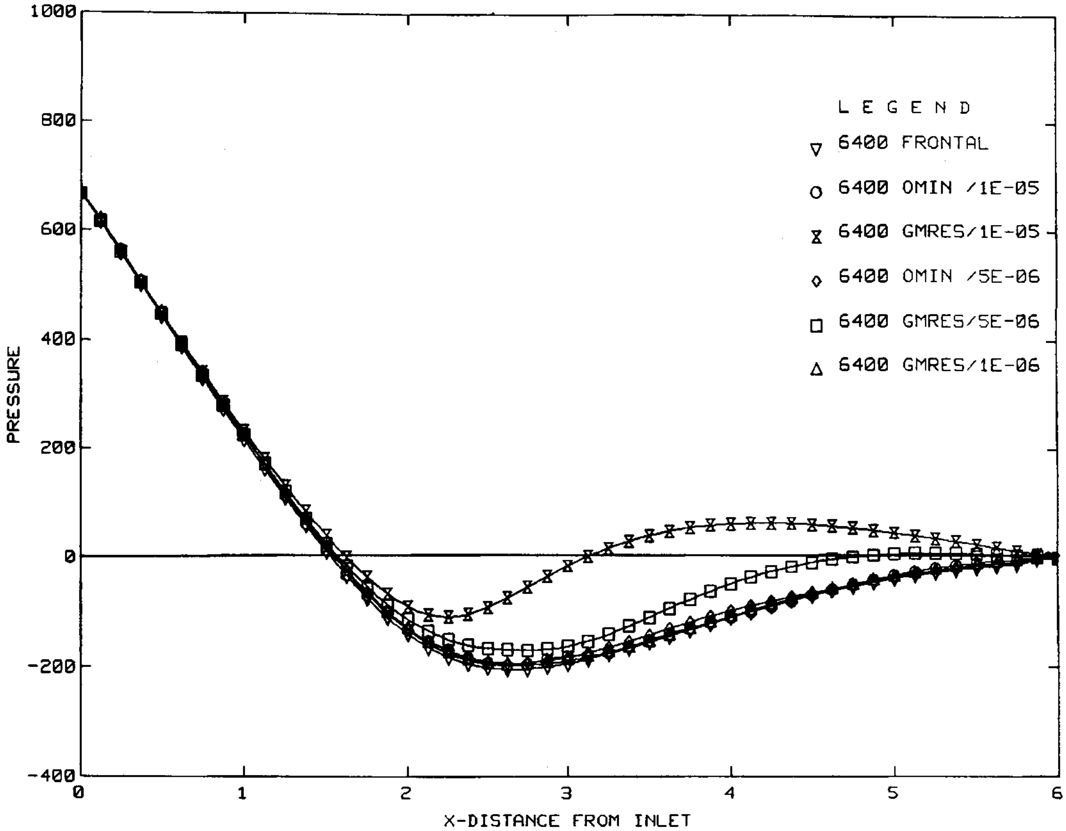

Figure 5. The predicted pressure distributions along the bottom wall using GMRES and ORTHOMIN solvers for 6400 elements mesh as a function of convergence tolerance, ε. The relative accuracy of iterative solvers is determined by comparing with the frontal solver results. The GMRES solver requires an order of magnitude smaller ε compared to the ORTHOMIN solver to achieve the same level of accuracy for the 6400 elements mesh

values obtained using the frontal solver with the finest mesh. Again, the ORTHOMIN solver exhibits a better solution compared to the GMRES solver using the same ε. The frontal solver solution shows oscillatory behaviour near the exit. This behaviour is due to the error in the calculation of velocity gradients near the corners. Further, in the case of traditional penalty model, the stresses and pressure are calculated after obtaining the converged velocity field. On the other hand, when using ORTHOMIN and GMRES solvers we use iterative penalty model where the stresses and pressures are computed for every non-linear iteration. This method provided a better smoothing of the stresses and suppresses oscillations in pressure solution. The reattachment length can be determined by calculating the location where the friction coefficient (wall shear stress) changes sign.

The friction coefficient along the top wall for the finest mesh is plotted in Figure 7. Here, it is evident that the large-scale phenomenon is not resolved with GMRES solver using $\varepsilon = 10^{-5}$ since the recirculation region is confined only to a small region close to the expansion. The error in the GMRES solution is also clear from the normalized velocity profiles at $x = 3$ (one unit downstream of the sudden expansion) shown in Figure 8. The velocities are normalized by dividing with maximum inlet velocity ($U = 60$). Similar to pressure solution, the accuracy of the shear stresses

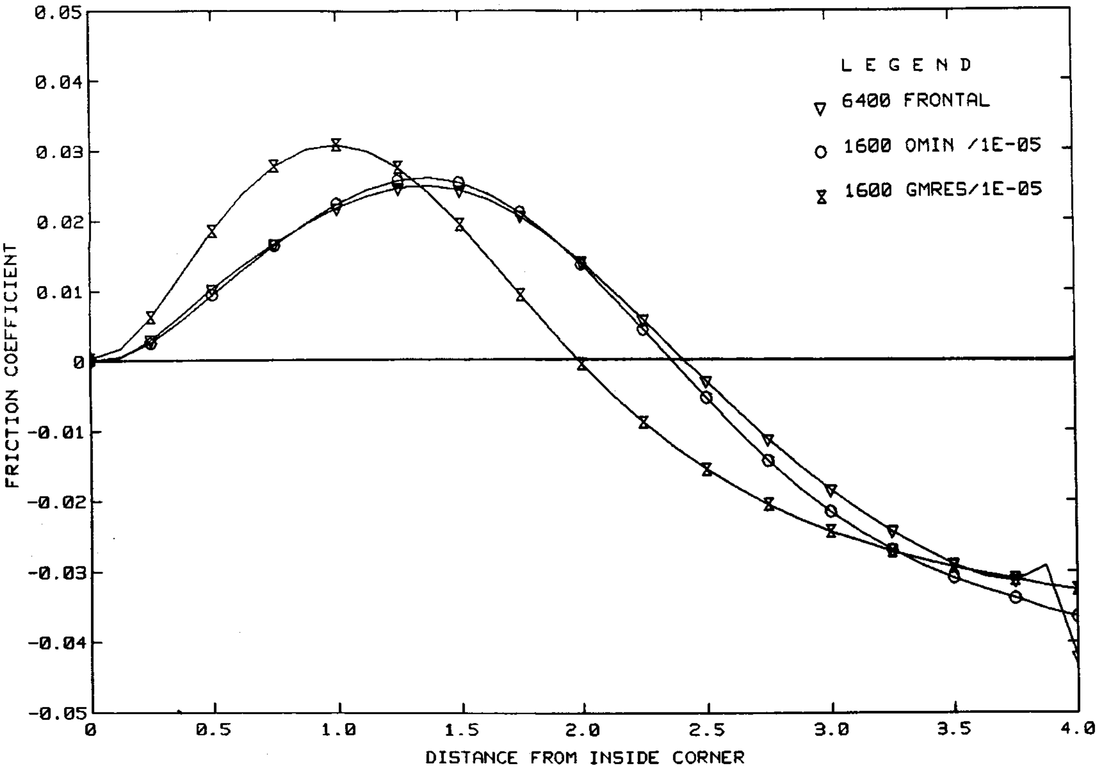

Figure 6. The predicted coefficient of friction along the top wall downstream from sudden expansion using GMRES and ORTHOMIN solvers for 1600 element mesh ($\varepsilon = 10^{-5}$). The relative accuracy of iterative solvers is determined by comparing with the frontal solver results (6400 elements mesh)

and the velocity profiles are improved by tightening the convergence tolerance. The reattachment lengths computed by the GMRES and the ORTHOMIN solvers as a function of mesh density and convergence tolerance are given in Table I. The frontal solution value for the fine mesh is also given in Table I.

The ORTHOMIN solver with $\varepsilon = 10^{-5}$ produced an accurate pressure distribution along the bottom wall (Figure 5), however, the error in the reattachment length (Table I) is nearly 10 per cent. This is because the pressure distribution on the bottom wall depends on the wall shear stress, which is a small-scale phenomenon. On the other hand, the reattachment length on the top wall depends on the strength of the convective terms and therefore is a large-scale phenomenon which was not resolved properly by using $\varepsilon = 10^{-5}$. Although the shear stress at the top wall is a small-scale phenomenon, it is coupled to the large-scale phenomenon of the flow separation. This expiains the error in the friction coefficient along the top wall in Figure 7.

The normalized CPU time for the 6400 element mesh for the iterative solvers with different values of ε are given in Table II. The CPU times are normalized by taking the time required by ORTHOMIN solver with $\varepsilon = 10^{-5}$ as one unit. As expected, the frontal solver required less CPU time. In general, for a given number of iterations, the GMRES solver requires more CPU time due to the calculations associated with the Krylov space. Table II illustrates that the ORTHOMIN solvers require less CPU time for the same accuracy. Concerning the convergence

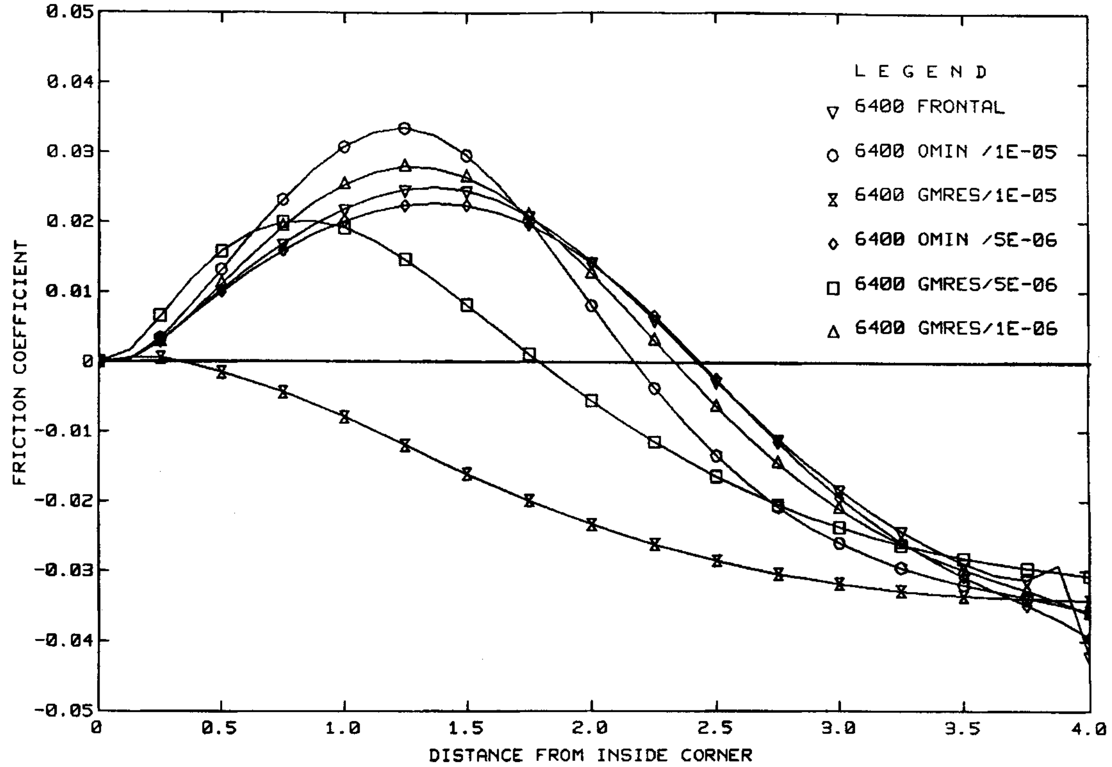

Figure 7. The predicted coefficient of friction along the top wall downstream from sudden expansion using GMRES and ORTHOMIN solvers for 6400 element mesh as a function of convergence tolerance, ε. The relative accuracy of iterative solvers is determined by comparing with the frontal solver results. The GMRES solver requires an order of magnitude smaller ε compared to the ORTHOMIN solver to achieve the same level of accuracy for the 6400 element mesh

rate of the iterative solvers, it was observed that the residual norm decreases monotonically for the GMRES solver, but the residual norm tends to oscillate for the ORTHOMIN solver.

Correlation between Δh, ε and λ

The convergence criterion needed to obtain a specified degree of accuracy in the solution is seen to be proportional to the mesh size. This relationship is apparent from the results seen in Figures 4–8 when comparing the trends for a given solver. As the element size, Δh, decreases, the convergence tolerance must also be decreased to maintain the same degree of accuracy. Thus,

$$\varepsilon = K_1 \Delta h, \tag{12}$$

where K_1 is some proportionality constant yet to be determined. This relationship is schematically illustrated in Figure 9.

Further, a relationship between ε and λ is seen when comparing the percentage error in the solution for a given solver as the convergence tolerance and the penalty parameter vary. Table III summarized the results of this comparison by using the ORTHOMIN solver for a fixed mesh size. The '100·00' in Table III denotes a solution devoid of a recirculation region. The stair step pattern in the results show that the maximum value of λ is proportional to the inverse of ε. Thus,

$$\varepsilon = K_2/\lambda, \tag{13}$$

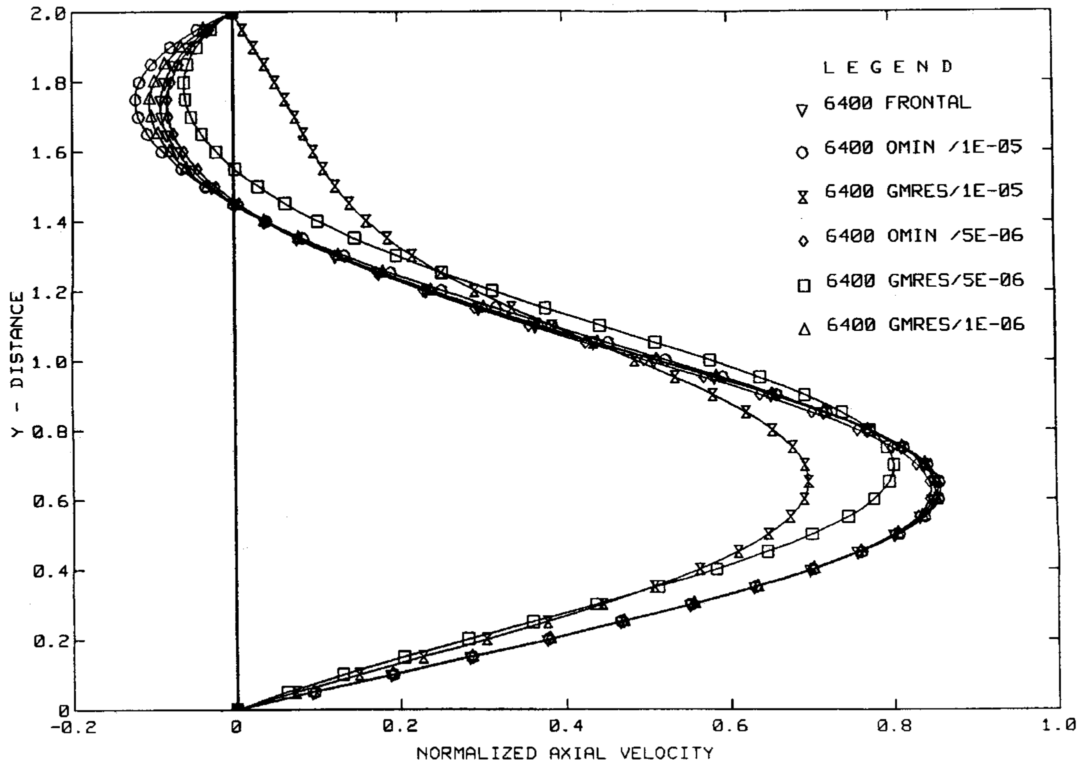

Figure 8. The predicted axial velocity profiles at $x=3$ (1 unit downstream of the sudden expansion) for GMRES and ORTHOMIN solvers as a function of convergence tolerance, ε. The relative accuracy of iterative solvers is determined by comparing with the frontal solver results. The trend agrees with results shown in the previous figures

Table I. Computed values of reattachment length

Case	Solver	Elements	Residual norm	Reattachment length (L_r)	Percentage error* in L_r
1	Frontal	6400	—	2·42093	0·00000
2	ORTHOMIN	1600	1×10^{-5}	2·36747	2·20824
3	GMRES	1600	1×10^{-5}	1·99294	17·67874
4	ORTHOMIN	6400	1×10^{-5}	2·16781	10·45548
5	GMRES	6400	1×10^{-5}	0·33772	86·05000
6	ORTHOMIN	6400	5×10^{-6}	2·43268	−0·48535
7	GMRES	6400	5×10^{-6}	1·78918	26·09534
8	GMRES	6400	1×10^{-6}	2·32677	3·88941

* Percentage error is calculated by taking frontal solver value as the reference

where K_2 is some proportionality constant yet to be determined. This relationship is schematically illustrated in Figure 10.

By combining equations (12) and (13) and expressing the penalty parameter explicitly as a product of λ and the viscosity η, a single comprehensive expressions for the interaction between

Table II. CPU time comparisons* for 6400 elements mesh

Case	Solver	Residual norm	CPU time
1	Frontal	—	0·2692
2	ORTHOMIN	1×10^{-5}	1·0000
3	GMRES	1×10^{-5}	1·9878
4	ORTHOMIN	5×10^{-6}	1·3930
5	GMRES	5×10^{-6}	4·7333
6	GMRES	1×10^{-6}	11·3959

* CPU times are normalized with the CPU time required by ORTHOMIN solver with $\varepsilon = 10^{-5}$

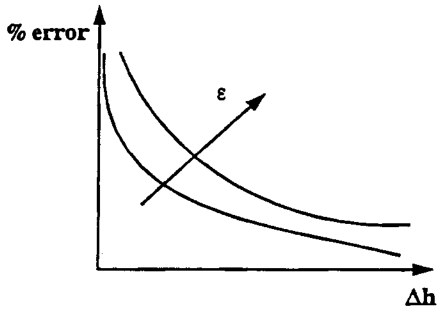

Figure 9. For a given penalty parameter, the convergence tolerance (ε) must decrease as the mesh size (Δh) decreases to maintain the same degree of accuracy. This trend is demonstrated in Figures 5–8 for a given iterative solver

Table III. Percentage error in reattachment length calculations for different values of ε and λ (ORTHOMIN solver, 400 elements mesh)

$\varepsilon \backslash \lambda$	1×10^{-3}	1×10^{-4}	1×10^{-5}	1×10^{-6}	1×10^{-7}
1×10^{0}	31·7795	45·8621	45·8365	45·8378	45·8390
1×10^{1}	23·2720	24·8706	24·5615	24·5615	24·8347
1×10^{2}	4·4462	5·0415	5·2695	5·1422	5·1422
1×10^{3}	2·4404	2·2624	1·5998	1·7014	1·6998
1×10^{4}	100·0000	3·7188	2·1673	1·1818	1·4420
1×10^{5}	100·0000	100·0000	4·8448	2·6254	1·1847
1×10^{6}	100·0000	100·0000	100·0000	−3·5776	1·9170
1×10^{7}	100·0000	100·0000	100·0000	100·0000	−2·8018
1×10^{8}	100·0000	100·0000	100·0000	100·0000	100·0000

Δh, ε and λ can be obtained:

$$\varepsilon = K' \Delta h / \lambda \eta. \tag{14}$$

Equation (14) can be made non-dimensional by assuming that K' contains the fluid density and a characteristic velocity as factors; then

$$\varepsilon = K \rho \Delta h U / \lambda \eta, \tag{15}$$

where the proportionality constant K is assumed to be a function of the geometry of the problem.

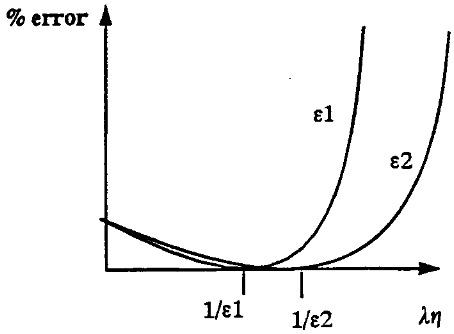

Figure 10. For a given mesh, the maximum value of penalty parameter (λ) allowed before the accuracy of the solution deteriorates rapidly is found to be proportional to the inverse of the convergence tolerance (ε)

Table IV. CPU time in seconds required for different values of ε and λ (ORTHOMIN solver, 400 elements mesh)

ε\λ	1×10^{-3}	1×10^{-4}	1×10^{-5}	1×10^{-6}	1×10^{-7}
1×10^{0}	35·21	89·77	122·30	136·76	145·33
1×10^{1}	30·63	69·51	88·77	106·79	111·99
1×10^{2}	32·07	65·13	73·12	95·91	110·23
1×10^{3}	32·69	91·38	138·06	194·90	237·66
1×10^{4}	8·19	155·29	168·11	275·60	406·85
1×10^{5}	6·33	8·75	189·56	246·06	483·68
1×10^{6}	6·11	5·95	9·75	159·23	362·99
1×10^{7}	6·61	6·12	7·00	12·57	262·69
1×10^{8}	6·98	6·32	6·87	6·72	9·74

Equation (15) can be used to determine the appropriate convergence criterion for a finer mesh solution once a satisfactory solution is obtained for a coarse mesh.

Finally, the relative efficiencies of different combinations of ε and λ in terms of the CPU requirements for a given degree of accuracy is summarized in Table IV. The element mesh yields the accurate estimate of the reattachment length with either $\varepsilon=10^{-6}$ and $\lambda=10^{4}$ or with $\varepsilon=10^{-7}$ and $\lambda=10^{5}$. However, the former combination requires about one-half of the CPU effort as the later, refer Table IV. In fact, depending upon the requirements for accuracy, much quicker results are possible with combinations of ε and λ that are in the upper left portion of Table IV.

CONCLUSIONS

The behaviour of GMRES and ORTHOMIN solvers as a function of mesh density is studied by solving flow inside a sudden expansion channel at a Reynolds number of 60. Based on the results presented in this paper, it is clear that the convergence rate of iterative solvers depends on the mesh density and the solution algorithm. It is also observed that the ORTHOMIN solver is more efficient than the GMRES solver as it requires less CPU time to obtain accurate solutions. Further, the finest mesh for iterative solvers produces less accurate results than the coarser meshes for a given convergence tolerance. This is due to the inability of the solver to resolve large-wavelength Fourier components of the error as the element density increases. We have observed a similar behaviour for a wide range of problems.

REFERENCES

1. G. F. Carey and J. T. Oden, *Finite Elements: Computational Aspects*, Prentice-Hall, Englewood Cliffs, NJ, 1984.
2. M. R. Hestenes and E. L. Stiefel, 'Methods of conjugate gradients for solving linear systems', *Nat. Bur. Std. J. Res.*, **49**, 409–436 (1952).
3. G. H. Gloub and C. F. Van Loan, *Matrix Computations*, 2nd edn. The Johns Hopkins University Press, Baltimore, 1989.
4. K. C. Jea and D. M. Young, 'On simplification of generalized conjugate gradient methods for nonsymmetrizable linear systems', *Lin. Alg. Appl.*, **52,** 399–417 (1983).
5. R. L. Fox and E. L. Stanton, 'Developments in structural analysis by direct energy minimization', *AIAA J.*, **6,** 1036–1042 (1986).
6. I. Fried, 'A gradient computational procedure for the solution of large problems arising from the finite element discretization method', *Int. j. numer. methods eng.*, **2,** 477–494 (1970).
7. Y. Saad and M. H. Schultz, 'GMRES: A generalized minimum residual algorithm for solving nonsymmetric linear systems', *SIAM J. Sci. Comput.*, **7,** 856–868 (1986).
8. L. J. Hayes, 'Advances and trends in element-by-element techniques', in A. K. Noor and J. T. Oden (eds), *State-of-the-Art Surveys on Computational Mechanics*, ASME, New York, 1989, pp. 219–236.
9. A. J. Wathen, 'An analysis of some element-by-element techniques', *Comput. Methods Appl. Mech. Eng.*, **74,** 217–287 (1989).
10. T. J. R. Hughes, I. Levit and J. Winget, 'An element-by-element solution algorithm for problems of structural and solid mechanics', *Comput. Methods Appl. Mech. Eng.*, **36,** 241–254 (1983).
11. T. J. R. Hughes, R. M. Ferencz and J. O. Hallquist, 'Large-scale vectorized implicit calculations in solid mechanics on a Cray X-MP/48 utilizing EBE preconditioned conjugate gradients', *Comput. Methods Appl. Mech. Eng.*, **61,** 215–248 (1987).
12. J. Winget and T. J. R. Hughes, 'Solution algorithms for nonlinear transient heat conduction analysis employing element-by-element iterative strategies', *Comput. Methods Appl. Mech. Eng.*, **52,** 711–815 (1985).
13. T. J. R. Hughes, J. Winget, I. Levit and T. E. Tezduyar, 'New alternating direction procedures in finite element analysis based upon EBE approximation factorization', in S. N. Atluri and N. Perrone (eds), *Computer Methods for Nonlinear Solids and Structural Mechanics*, AMD-Vol. 54, (1983), pp. 75–109.
14. G. F. Carey and B. Jiang, 'Subcritical flow computations using an element-by-element conjugate gradient method', *Proc. 5th Int. Symp. on Finite Elements and Flow Problems*, Jan. 23–26, Univ. of Texas at Austin, 1984, pp. 103–106.
15. G. F. Carey and B. Jiang, 'Element-by-element linear and nonlinear solution schemes', *Commun. Appl. Numer. Methods*, **2,** 281–287 (1986).
16. F. Shakib, T. J. R. Hughes and Z. Johan, 'A multi-element group preconditioned GMRES algorithm for nonsymmetric systems arising in finite element analysis', *Comput. Methods Appl. Mech. Eng.*, **75,** 415–456 (1989).
17. T. E. Tezduyar and J. Liou, 'Element-by-element and implicit-explicit finite element formulations for computational fluid dynamics', in R. Glowinski, G. H. Golub, G. A. Meurant and J. Periaux (eds), *First Int. Symp. on Domain Decomposition Methods for Partial Differential Equations*, SIAM, 1988, pp. 281–300.
18. T. E. Tezduyar and J. Liou, 'Grouped element-by-element iteration schemes for incompressible flow computations', *Comput. Phys. Commun.*, **53,** 441–453 (1989).
19. T. E. Tezduyar, 'Stabilized finite element formulations for incompressible flow computations', *Adv. Appl. Mech.*, **28,** 1–44 (1991).
20. S. Mittal, A. Ratner, D. Hastreiter and T. E. Tezduyar, 'Space–time finite element computation of incompressible flows with emphasis on flows involving oscillating cylinders', *Int. Video J. Eng. Res.*, **1,** 83–96 (1992).
21. M. P. Reddy, J. N. Reddy and H. U. Akay, 'Penalty-finite element analysis of incompressible flows using element-by-element solution algorithms', *Comput. Methods Appl. Mech. Eng.*, **100,** 169–205 (1992).
22. M. P. Reddy and J. N. Reddy, 'Numerical simulation of forming processes using a coupled fluid flow and heat transfer model', *Int. j. numer. methods eng.*, **35,** 807–833 (1992).
23. E.-M. Salonen and J. Aalto, 'A pressure determination scheme', in C. Taylor, M. D. Olson, P. M. Gresho and W. G. Habashi (eds), *Proc. 5th Int. Conf. Numer. Meths. Laminar and Turbulent Flows*, John Wiley, 1985, pp. 128–139.
24. T. Shiogima and Y. Shimazaki, 'A pressure-smoothing scheme for incompressible flow problems', *Int. j. numer. methods fluids*, **9**, 557–567 (1989).
25. J. N. Reddy, 'On penalty function methods in the finite element analysis of fluid flow', *Int. j. numer. methods fluids*, **9,** 151–171 (1982).
26. R. L. Lee, P. M. Gresho and R. L. Sani, 'Smoothing techniques for certain primitive variable solutions of the Navier–Stokes equations', *Int. j. numer. methods eng.*, **14,** 1785–1804 (1979).