Newton-Krylov Method for Compressible Euler Equations on Unstructured Grids

Sungho Kim and Jang Hyuk Kwon

Abstract

The Newton-Krylov method on the unstructured grid flow solver using the cell-centered spatial discretization of compressible Euler equations is presented. This flow solver uses the reconstructed primitive variables to get the higher order solutions. To get the quadratic convergence of Newton method with this solver, the careful linearization of face flux is performed with the reconstructed flow variables.

The GMRES method is used to solve large sparse matrix and to improve the performance ILU preconditioner is adopted and vectorized with level scheduling algorithm. To get the quadratic convergence with the higher order schemes and to reduce the memory storage, the matrix-free implementation and Barth's matrix-vector method are implemented and compared with the traditional matrix-vector method. The convergence and computing times are compared with each other.

1 Introduction

As a convergence acceleration method the Newton-Krylov methods are widely spreaded recently. This is because the implicit schemes can increase the stability over explicit ones. However, the implicit methods usually require the solution procedure of a large linear system of equations and memory to store the Jacobian matrix. For the structured grid flow solver, there are many methods to simplify the system of equations using factorization or diagonalization. Nevertheless, the unstructured grid flow solver can not use these simplifying procedures because of its sparsity of global assembled matrix. That is the reason the multigrid method was a first choice as convergence acceleration techniques on unstructured grid flow solvers.

During last ten years the iterative matrix solvers are developed drastically using Krylov subspace methods such as Conjugate Gradients, GMRES, GMR or BiCG. Using these iterative matrix solvers the unstructured grid flow solver can solve the system of equations fast. The iterative matrix solver was widely applied to many area of computational fluid dynamics and it gives reasonable results for those areas [1, 3, 9–11, 14].

Nowadays Krylov subspace method is used as a convergence acceleration technique, which is combined with the traditional Netwon's method. This Newton-Krylov method gives a solution in several iterations. However, just a few researchers can obtain the quadratic convergence with Newton-Krylov Method. These successful results are presented by several authors [2, 5, 9, 11]. They have investigated about the linear and nonlinear Newton-Krylov methods with finite difference evaluation of the matrix-vector or with numerically evaluated flux Jacobian.

In this paper, the importance of linearization of local residuals will be described in the aspects of cell-centered discretization on unstructured grids. The feasibility of this linearization is checked with the flows around a airfoil.

2 Governing equations and solution algorithm

Governing Equations and Discretization

The Navier-Stokes equations in the integral form for a control volume Ω with boundary $\partial\Omega$ read

$$\frac{d}{dt}\int_{\Omega}\mathbf{Q}dV + \oint_{\partial\Omega}\mathbf{F}_c(\mathbf{Q},\mathbf{n})dS = 0. \tag{1}$$

Here $\mathbf{Q}$ is the solution vector comprised of the conservative variables, the density and two components of momentum and the total energy. The vectors $\mathbf{F}_c$ represents the inviscid flux vectors for a surface with the normal vector $\mathbf{n}$. The control volume is the triangular elements itself and the variables $\mathbf{Q}$ are stored at the center of triangular elements.

The flow solver uses an upwind implicit algorithm in which the fluxes are obtained using Roe's flux difference splitting(FDS). Hence, the numerical face flux is defined as

$$\mathbf{F}_k = \frac{1}{2}\left[\mathbf{F}(\mathbf{Q}_L) + \mathbf{F}(\mathbf{Q}_R) - |\tilde{\mathbf{A}}|(\mathbf{Q}_R - \mathbf{Q}_L)\right] \tag{2}$$

where, $|\tilde{\mathbf{A}}|$ is a Roe averaged matrix.

The construction of numerical flux across each cell face can be done in two steps. Firstly, a linear reconstruction of the variables is performed. In this reconstruction step, the variables at the cell centers are distributed to the cell vertices. Secondly the variables on either side of the cell face are interpolated as an initial data for Roe's approximate Riemann solver with that of cell vertices.

Finally the discretized equations of Eq (1) give a coupled ordinary differential equations as following

$$V\frac{d\mathbf{Q}}{dt} = \mathbf{R}(\mathbf{Q}). \tag{3}$$

Herein, the residual is defined as

$$\mathbf{R} = -\oint_{\partial\Omega} \mathbf{F}_c(\mathbf{Q}) \cdot \mathbf{n} dS. \tag{4}$$

The backward-Euler time-differencing scheme is used to march time steps. The residual at time $n+1$ step is linearized locally using Taylor series expansion with respect to time.

$$\mathbf{R}^{n+1} = \mathbf{R}^n + \frac{\partial \mathbf{R}}{\partial \mathbf{Q}}\Delta\mathbf{Q} \tag{5}$$

where $\Delta\mathbf{Q} = \mathbf{Q}^{n+1} - \mathbf{Q}^n$ is the change of conservative variable.

Applying Eq (5) to Eq (3) gives a large sparse non-symmetric system of linear equation such as

$$\left[\frac{V}{\Delta t}\mathbf{I} - \frac{\partial \mathbf{R}}{\partial \mathbf{Q}}\right]\{\Delta\mathbf{Q}\} = \{\mathbf{R}\} \tag{6}$$

where, $\mathbf{I}$ is identity matrix. This system of linear equations can be solved directly with Gauss elimination, or iteratively with well-known iterative solvers.

A linear system solver and Newton's method

The performance of implicit method depends on its performance of linear matrix solver. In the last few years a great variety of Krylov subspace methods were developed and used in different flow solvers [3, 5, 10, 14]. Among the Krylov subspace method, GMRES is used which is developed by Saad [12]. With the right preconditioner matrix P the procedure of Flexible GMRES (FGMRES) algorithm to solve $AP^{-1}(Px) = b$ is directly referred from Reference [13]:

1. $r_0 = b - Ax_0, \beta = ||r_0||_2, q_1 = r_0/\beta$
2. $For\ j = 1, \cdots, m\ Do:$
3. $\quad z_j := P^{-1}v_j$
4. $\quad w := Az_j$
5. $\quad For\ i = 1, \cdots, j\ Do:$
6. $\quad h_{i,j} := (w, v_i)$
7. $\quad w := w - h_{i,j}v_i$
8. $\quad EndDo$
9. $\quad Compute\ h_{j+1,j} = ||w||_2\ and\ v_{j+1} = w/h_{j+1,j}$
10. $\quad Define\ Z_m := [z_1, \cdots, z_m],\ \bar{H}_m = \{h_{i,j}\}_{1\le i\le j+1; 1\le j\le m}$
11. $EndDo$
12. $Compute\ y_m = argmin_y||\beta e_1 - \bar{H}_m y||_2,\ and\ x_m = x_0 + Z_m y_m.$
13. $If\ satisfied\ Stop,\ else\ set\ x_0 \leftarrow x_m\ and\ GoTo\ 1.$

This FGMRES algorithm has some advantages which can apply the different preconditioner every iterations and can be possible restarting with user-defined matrix-vector routines. The Block ILU(0) is used as preconditioner matrix P and any other convergence acceleration methods such as local preconditioning, multigrid method or mesh sequencing, are not used.

If the complete linearization of residual in Eq (6) is performed, then as $\Delta t \to \infty$, this process approaches Newton's method as following

$$\mathbf{Q}^{n+1} = \mathbf{Q}^n - \left[\frac{\partial \mathbf{R}}{\partial \mathbf{Q}}\right]^{-1}\mathbf{R}(\mathbf{Q}). \tag{7}$$

The inverse of flux Jacobian matrix $\frac{\partial \mathbf{R}}{\partial \mathbf{Q}}$ is obtained using Krylov subspace method.

In general the left hand side Jacobian matrix is constructed in first order accurate form for the implicit solver. However, in order to obtain quadratic convergence with the higher order accurate scheme, the left and the right hand side of linear system of equations have same spatial accuracy and the accurate linearization of higher order residual combining with Krylov subspace methods is needed.

Hence, to increase the accuracy of left hand side matrix of linear system, the full storing of Jacobian matrix or the accurate matrix-vector are needed. The former method is called full Newton method which needs huge memory to store Jacobian matrix and the later is called matrix-free or nonlinear Newton-Krylov method which needs accurate matrix-vector product during the procedure of GMRES. This matrix-vector can be done analytically [8] or numerically [2].

Linearization of the flux Jacobian matrix and the matrix-vector product

The first order flux Jacobian matrix $\mathcal{J}$ can be evaluated numerically which can be found in the book of Schnabel [4]. This is presented as following

$$\begin{aligned} \mathcal{J}_n(\mathbf{Q}) &= \frac{\partial \mathbf{F}(\mathbf{Q})}{\partial \mathbf{Q}} \\ &= \frac{\mathbf{F}(\mathbf{Q} + \epsilon_{mz}\mathbf{l_m}) - \mathbf{F}(\mathbf{Q})}{\epsilon} \end{aligned} \tag{8}$$

where, ϵ_{mz} is a small number which is usually machine zero, $\mathbf{l}_m$ is a unit vector to perturb flow variables and $\mathcal{J}_n(\mathbf{Q})$ is a numerical Jacobian matrix.

The similar procedure to product the Jacobian matrix with Krylov subspace vector can be defined numerically as

$$\begin{aligned} \mathcal{J}_{ij}(\mathbf{Q}) \cdot \mathbf{v}_j &= \sum_k \left[\frac{\partial \mathbf{F}}{\partial \mathbf{Q}}\right]_{ik} \cdot \mathbf{v}_k \\ &= \frac{\mathbf{F}(\mathbf{Q}_i + \epsilon \mathbf{v}_j) - \mathbf{F}(\mathbf{Q}_i)}{\epsilon} \end{aligned} \tag{9}$$

where, $\mathbf{v}$ is a Krylov subspace vector and the parameter ϵ is computed at every inner iterations as a function of machine zero, ϵ_{mz} and the norm of $\mathbf{v}$ as proposed by Nielsen et al. [2]

$$\epsilon = \frac{\sqrt{\epsilon_{mz}}}{L_2(\mathbf{v})} \tag{10}$$

On the other hand, the production of analytic Jacobian matrix with Krylov vector proposed by Barth [8] can be described in following form of

$$\tilde{\mathcal{J}}_{Barth}(\mathbf{Q}) \cdot \mathbf{v} = \sum \frac{\partial \mathbf{F}}{\partial \tilde{\mathbf{Q}}} \cdot \tilde{\mathbf{v}}. \tag{11}$$

This matrix-vector product is performed in edge-wise. Hence, for a single edge this matrix-vector product can be written as

$$\left(\tilde{\mathcal{J}}_{Barth}(\mathbf{Q}) \cdot \mathbf{v}\right)_{L,R} = \left(\frac{\partial \mathbf{F}}{\partial \tilde{\mathbf{Q}}}\right)_L \cdot \tilde{\mathbf{v}}_L \cdot + \left(\frac{\partial \mathbf{F}}{\partial \tilde{\mathbf{Q}}}\right)_R \cdot \tilde{\mathbf{v}}_R. \tag{12}$$

where $\frac{\partial \mathbf{F}}{\partial \tilde{\mathbf{Q}}}$ is a Jacobian matrix which is evaluated with reconstructed conservative flow variables, $\tilde{\mathbf{v}}$ is a reconstructed Krylov vector and the subscript L and R mean the index number of left and right cells with respect to a edge.

The flux Jacobian matrix $\frac{\partial \mathbf{F}}{\partial \tilde{\mathbf{Q}}}$ can be obtained in several manners such as a constant treatment of Roe averaged matrix [10], direct linearization of face flux by Barth [7] or numerical evaluation [4]. In this paper the Roe averaged matrix is treated as a constant.

3 Results

In order to verify the linearization effect of residual and to check the feasibility of flow solver, the flows around a NACA 0012 airfoil are computed on the subsonic and transonic flow region. The unstructured grids are generated using the Frontal Delaunay Method [6]. The total number of nodes is 4333, the number of triangular elements is 8410 and 192 nodes are distributed on the airfoil surface. The subsonic flow condition is $M = 0.63$ and $\alpha = 2.$ degree and the transonic flow condition is $M = 0.8$ and $\alpha = 1.25$ degree. Figure (1) depicts enlarged view of grids near a NACA 0012 airfoil.

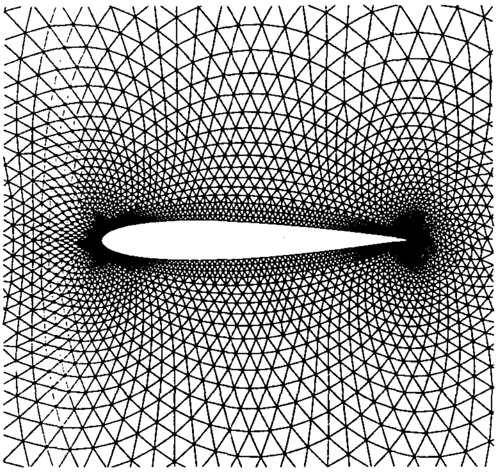


Figure 1. Unstructured Grids around a NACA 0012 airfoil

The choice of Krylov subspace dimension

The convergence tolerance of matrix solver and the number of Krylov subspace dimension have to determined to reduce the memory storage and computing time. Figure (2(a)) and (2(b)) depict the convergence history of first and second order spatial discretization with respect to Krylov subspace dimensions. The subsonic flow around a airfoil is computed when the convergence tolerance of matrix solver is 0.001.

From these figures we choose 40 Krylov subspace dimensions to guarantee the convergence for the higher order spatial discretization.

Comparison of convergence history

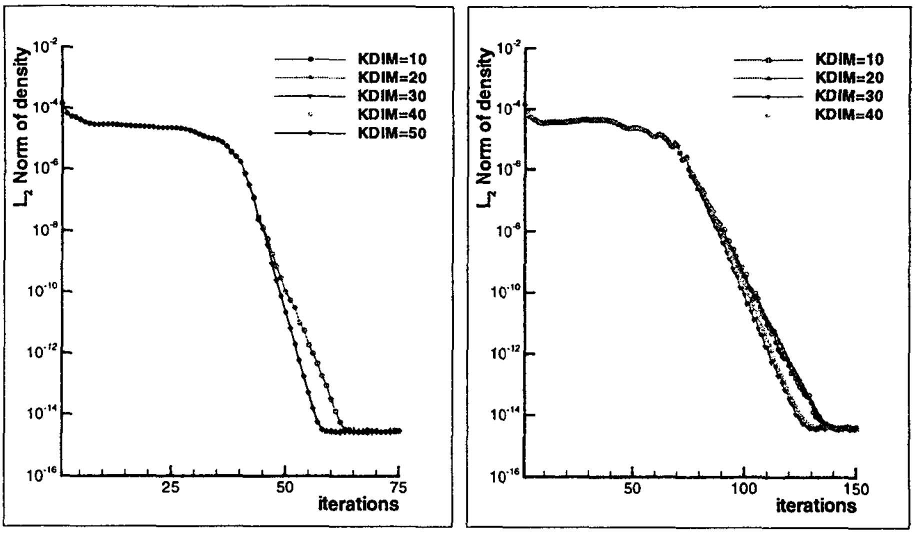
In order to check the linearization effect with the primitive variables, the first order calculations are performed for the subsonic flow. The traditional matrix-vector product which stores the flux Jacobian matrix is compared with numerical matrix-vector and Barth's matrix-vector product. Figure (3(a)) depicts the convergence history with respect to number of iterations for the 1st order discretization and Figure (3(b)) shows with respect to computing time. The numerical matrix-vector product is slightly faster than traditional matrix-vector and Barth's matrix-vector product with respect to number of iterations. In view of computing time, however, the numerical matrix-vector product is slower than the traditional method but slightly faster than Barth's method.

For the subsonic flow numerical matrix-vector product is compared with Barth's matrix-vector product. In view of iteration counts it is difficult to find out the difference between two methods. However, in view of computing time numerical matrix-vector is much faster than Barth's matrix-vector method.

For the transonic flow, the first order calculation is performed with all three method. Because of the flow characteristics, the total number of iterations and the computing times are larger than that of subsonic flows. However, the convergence characteristics are same as that of subsonic flow, that is, the numerical matrix-vector is faster than Barth's method but is slower than the traditional method. These are depicted in Figure (5(a)) and (5(b)). However, for the 2nd order scheme the numerical matrix-vector is much faster than Barth's method which are depicted in Figure (4(a)) and (4(b)).

4 Conclusion

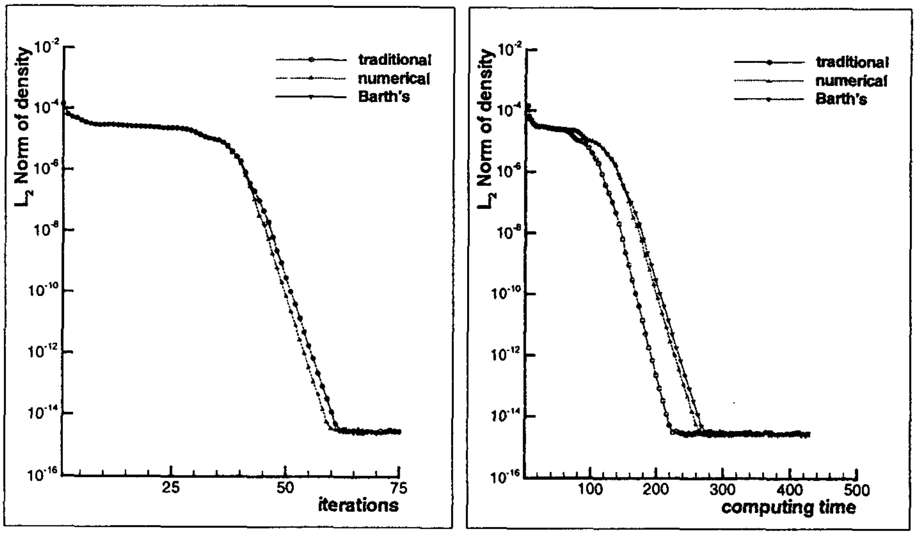
The Newton-Krylov Method on unstructured grids with cell-centered spatial discretization is presented, which can reduce the memory storage and computing time of flow solver. The quadratic convergence characteristics of Newton method is obtained with careful linearization of local residuals on the cell centers. The matrix-free and Barth's method are implemented to reduce the memory storage and to reserve the convergence characteristics of Newton method. Within a reasonable number of iterations and computing time ,we can get the flow solution on unstructured grids with this method.



(a) Convergence history of 1st order spatial discretization with respect to Krylov subspace dimensions

(b) Convergence history of 2nd order spatial discretization with respect to Krylov subspace dimensions

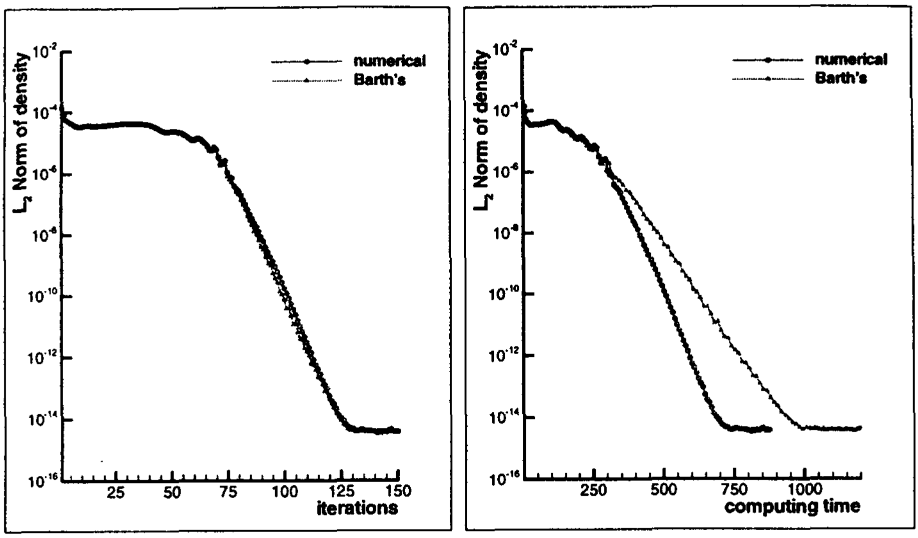
Figure 2. Convergence History



(a) Comparison of convergence history for the first order discretization with respect to number of iterations

(b) Comparison of convergence history for the first order discretization with respect to computing time

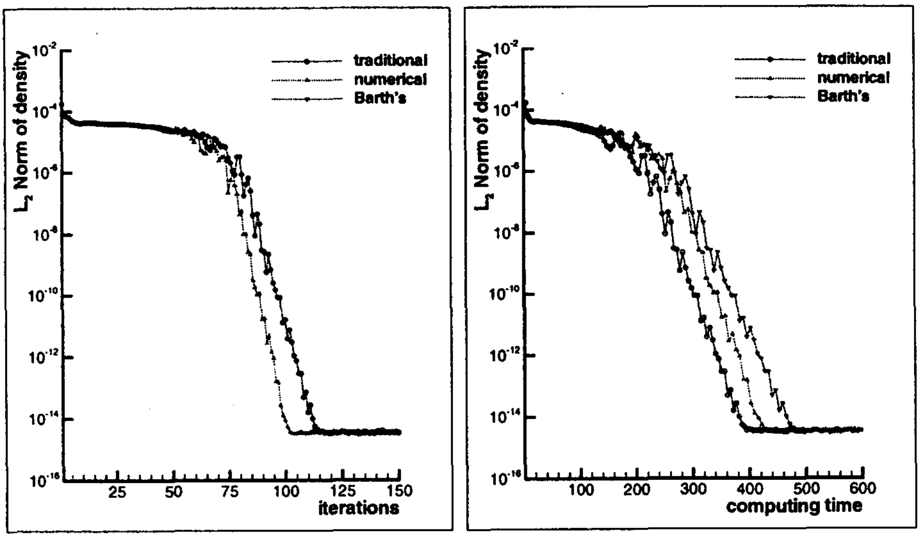
Figure 3. Convergence history of 1st order scheme



(a) Comparison of convergence history for the 2nd order discretization with respect to number of iterations

(b) Comparison of convergence history for the 2nd order discretization with respect to computing time

Figure 4. Convergence history of 2nd order scheme



(a) Convergence history of transonic flow for the 1st order with respect to number of iterations

(b) Convergence history of transonic flow for the 1st order with respect to computing time

Figure 5. Convergence history of transonic flow

References

[1] C. W. S. Bruner. *Parallelization of the Euler Equations on Unstructured Grids*. PhD thesis, Department of Aerospace Engineering, Vriginia Polytechnic Institute and State University, May 1996.

[2] Eric J. Nielsen, W. Kyle Anderson, Robert W. Walters, and David E. Keyes. Application of Newton-Krylov Methodology to a Three-Dimensional Unstructured Euler Code. AIAA Papers 95-1733, San Diego CA, 1995.

[3] Farzin Sharkib, Thomas J.R. Hughes, and Zdenek Johan. A Multi-Element Group Preconditioned GMRES Algrithm for Nonsymmetric Systems Arising in Finite Element Analysis. *Computer Methods in Applied Mechanics and Engineering*, 75:415–456, 1989.

[4] J. E. Dennis Jr. and Robert B. Schnabel. *Numberical Methods for Unconstrained Optimization and Nonlinear Equations*. Prentice-Hall Inc., New Jersey, 1983.

[5] Kirk J. Vanden and Paul D. Orkwis. Comparison of Numberical and Analytical Jacobians. *AIAA Journal*, 34(6):1125–1129, June 1996.

[6] Müller,J.D., Roe, P.L., and Deconinck, H. A Frontal Approach for internal node generation in Delaunay Triangulation. *Int. J. Num. Meth. Fluids.*, 17:241–255, 1993.

[7] Timothy J. Barth. Analysis of implicit local linearization techniques for upwind and tvd algorithms. AIAA Papers 87-0595, 1987.

[8] Timothy J. Barth. Numerical Aspects of Computing Viscous High Reynolds Number Flows on Unstructured Meshes. AIAA Papers 91-0721, 1991.

[9] Timothy J. Barth. Parallel CFD Algorithms on Unstructured Meshes. Technical Report 807, May 1995.

[10] V. Venkatakrishnan. Parallel Implicit Unstructured Grid Euler Solvers. *AIAA Journal*, 32(10):1985–1991, October 1994.

[11] W. Kyle Anderson, Russ D. Rauss, and Daryl L. Bonhaus. Implicit/Multigrid Algorithms for Incompressible Turbulent Flows on Unstructured Grids. AIAA Papers 95-1740, 1995.

[12] Youcef Saad and Martin H. Schultz. GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsysmetric Linear Systems. *SIAM J. Sci. Stat. Comput.*, 7(3):856–869, July 1986.

[13] Yousef Saad. *Iterative Methods for Sparse Linear Systems*. PWS Publishing Company., Boston. 1992.

[14] Zdenek Johan, Thomas J.R. Hughes, and Farzin Shakib. A Globally Convergent Matrix-Free Algorithm for Implicit Time Marching Schemes Arising in Finite Element Analysis in Fluids. *Computer Methods in Applied Mechanics and Engineering*, 87:281–304, 1991.