

PSEUDO-TRANSIENT CONTINUATION FOR NONLINEAR TRANSIENT ELASTICITY

MICHAEL W. GEE*, C. T. KELLEY†, AND R. B. LEHOUCQ‡

Abstract. We show how the method of pseudo-transient continuation can be applied to improve the robustness of the Newton iteration within a nonlinear transient elasticity simulation. Pseudo-transient continuation improves efficiency and robustness of the transient analysis by enabling larger time steps than possible with a Newton iteration. We illustrate the algorithm by reporting on a simulation of a buckling cylinder.

Key words. Transient elasticity, Finite Element Approximation, Pseudo-transient Continuation, Newton's Method

1. Introduction. Let the nonlinear set of equations arising in nonlinear transient elasticity be given by

$$\mathbf{r}(\mathbf{u}) = \mathbf{0}, \quad (1.1)$$

where $\mathbf{r}(\mathbf{u})$ represents the sum of transient, internal and external forces, and \mathbf{u} the corresponding displacement field. A standard approach for the solution of (1.1) is to employ Newton's iteration for (1.1), e.g.

$$\mathbf{r}'(\mathbf{u}^k)\Delta\mathbf{u} = -\mathbf{r}(\mathbf{u}^k), \quad \mathbf{u}^{k+1} = \mathbf{u}^k + \Delta\mathbf{u}, \quad k \leftarrow k + 1 \quad (1.2)$$

where \mathbf{r}' denotes the Jacobian of \mathbf{r} , and \mathbf{u}^k is the k -th iterate. If the initial iterate \mathbf{u}^0 is within the region of local convergence, then Newton's iteration is viable, assuming that the linear set of equations given by (1.2) are efficiently solved.

The purpose of this paper is to demonstrate the utility of pseudo-transient continuation (Ψ tc) as a robust variation of Newton's method for use within a nonlinear transient elasticity simulation. We present numerical examples that show that Ψ tc enables the use of a larger time step than that associated with the standard Newton iteration used within a nonlinear transient elasticity simulation.

Ψ tc is a way to recover convergence of Newton's method when the initial iterate \mathbf{u}^0 is not within the region of local convergence, and improve the solution efficiency of the associated set of linear equations. Ψ tc recasts the nonlinear set of equations (1.1) to compute the steady-state solution of an initial value problem of the form

$$\frac{d\mathbf{u}}{d\tau} = -\mathbf{r}(\mathbf{u}), \quad \mathbf{u}(0) = \mathbf{u}^0, \quad (1.3)$$

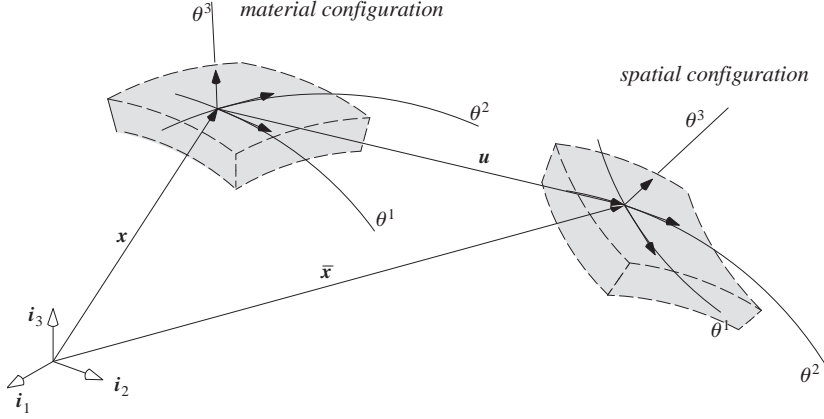
if such a solution exists. By a steady-state solution we mean \mathbf{u}^* such that

$$\lim_{\tau \rightarrow \infty} \mathbf{u}(\tau) = \mathbf{u}^*,$$

*Chair of Computational Mechanics, Technical University of Munich Boltzmannstrasse 15, D-85747 Garching, Germany, (gee@lrm.mw.tum.de)

† North Carolina State University, Department of Mathematics, Box 8205, Raleigh, N. C. 27695-8205, USA (Tim.Kelley@ncsu.edu). This research was supported in part by National Science Foundation grant DMS-0707220.

‡Sandia National Laboratories, P.O. Box 5800, MS 1320, Albuquerque, NM 87185-1320 (rblehou@sandia.gov). Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the U.S. Department of Energy under contract DE-AC04-94AL85000.

FIG. 2.1. *Geometry and kinematics*

where $\mathbf{u}(\tau)$ is the solution of the initial value problem (1.3). Hence \mathbf{u}^* is a solution to (1.1). A first-order Rosenbrock [10] scheme applied to the time integration of (1.3) leads to

$$(\delta_k^{-1} \mathbf{I} + \mathbf{r}'(\mathbf{u}^k)) \Delta \mathbf{u} = -\mathbf{r}(\mathbf{u}^k), \quad \mathbf{u}^{k+1} = \mathbf{u}^k + \Delta \mathbf{u}, \quad k \leftarrow k + 1. \quad (1.4)$$

In this sense, Ψtc modifies the Jacobian with a “pseudo-time step” δ_k . Note that addition of $\delta_k^{-1} \mathbf{I}$ to the Jacobian \mathbf{r}' regularizes the linear set of equations and so may lead to a more efficient solution when the Jacobian \mathbf{r}' is ill conditioned. We refer the reader to [12, 5, 9] for more details including a Ψtc convergence theory.

Ψtc advances a pseudo-temporal discretization of (1.3), but in a very different way from a fully temporally accurate integration. Rather Ψtc manages the “pseudo time step” to make it as large as possible with a view toward rapid convergence to steady-state. Ψtc is different from integrating (1.3) to steady-state with a conventional initial value problem code [10, 1, 19] where the time step is controlled with stability and accuracy in mind. Hence Ψtc is motivated by the goal of fast convergence to steady-state and not that of local temporal accuracy. Ψtc has been applied to computational fluid dynamics [6, 18, 17, 24, 13, 16], combustion [14, 21], plasma dynamics [15], radiation transport [20], and hydrology [8].

Our paper is organized as follows. Section 2 briefly reviews a finite element formulation for nonlinear transient elasticity resulting in a semi-discrete set of equations. Section 3 describes the time integrator applied to the semi-discrete set of equations. Section 4 recasts the (fully) discrete set of equations using Ψtc , and two management schemes for δ_k . Finally, section 5 discusses our numerical examples.

2. Nonlinear transient elasticity. The local form of the problem of interest is the initial boundary value problem

$$\rho \ddot{\mathbf{u}} = \nabla \cdot (\mathbf{F} \cdot \mathbf{S}) + \rho \mathbf{b} \quad \text{in } \Omega \times [0, T], \quad (2.1a)$$

$$\mathbf{S} = \mathbb{C} : \mathbf{E} = \mathbb{C} : \frac{1}{2} (\mathbf{F}^T \mathbf{F} - \mathbf{I}), \quad (2.1b)$$

$$\mathbf{u} = \mathbf{u}_D \quad \text{in } \Gamma_D \times [0, T], \quad (2.1c)$$

$$\mathbf{t} = \mathbf{t}_N \quad \text{in } \Gamma_N \times [0, T], \quad (2.1d)$$

$$\mathbf{u} = \mathbf{u}_0; \quad \dot{\mathbf{u}} = \dot{\mathbf{u}}_0 \quad \text{in } \Omega \quad \text{at } t = 0. \quad (2.1e)$$

Here, \mathbf{F} denotes the deformation gradient that maps the material to the spatial configuration (see Figure 2.1).

$$\mathbf{F} = \frac{d\mathbf{x}}{d\bar{\mathbf{x}}} = \nabla_0 \mathbf{x}. \quad (2.2)$$

\mathbf{S} is the Second Piola Kirchhoff stress tensor and \mathbf{b} a body force. The stresses \mathbf{S} are related to the strains \mathbf{E} via the material law described through \mathbb{C} . The Green-Lagrange strains \mathbf{E} are computed from the deformation gradient \mathbf{F} by

$$\mathbf{E} = \frac{1}{2} (\mathbf{F}^T \mathbf{F} - \mathbf{I}) . \quad (2.3)$$

The relationship between \mathbf{E} and $\bar{\mathbf{x}} = \mathbf{x} + \mathbf{u}$ represents the nonlinearity. The material law is potentially another source of nonlinearity though for simplicity we assume that \mathbb{C} represents a linear stress-strain relationship.

Let the trial and test spaces be $\mathcal{S} = \{\mathbf{u} \in H^1(\Omega) \mid \mathbf{u}|_{\Gamma_D} = \mathbf{u}_D\}$ and $\mathcal{V} = H_0^1(\Omega)$, respectively. The principal of virtual work for (2.1) is: Find $\mathbf{u} \in \mathcal{S}$ so that $\forall \delta \mathbf{u} \in \mathcal{V}$

$$\int_{\Omega} \rho \ddot{\mathbf{u}} \cdot \delta \mathbf{u} \, d\Omega + \int_{\Omega} \mathbf{S} : \delta \mathbf{E} \, d\Omega - \int_{\Gamma_N} \mathbf{t}_N \cdot \delta \mathbf{u} \, d\Gamma - \int_{\Omega} \rho \mathbf{b} \cdot \delta \mathbf{u} \, d\Omega = 0 . \quad (2.4)$$

Introducing finite element approximations $\mathbf{u}^h \in \mathcal{S}^h \subset \mathcal{S}$ and $\delta \mathbf{u}^h \in \mathcal{V}^h \subset \mathcal{V}$ where $\mathcal{S}^h, \mathcal{V}^h$ are finite element subspaces yields a discrete version of (2.4). Under a finite element discretization, the first and second integrals in (2.4) result in the transient and internal forces $\mathbf{f}^{\text{trans}}(\mathbf{u}^h, t)$ and $\mathbf{f}^{\text{int}}(\mathbf{u}^h)$, respectively, and the remaining two integrals represent the external force \mathbf{f}^{ext} . The discrete analogue of (2.4) is then

$$\mathbf{f}^{\text{trans}}(\mathbf{u}^h, t) + \mathbf{f}^{\text{int}}(\mathbf{u}^h) + \mathbf{f}^{\text{ext}} = 0 \quad (2.5)$$

The internal force $\mathbf{f}^{\text{int}}(\mathbf{u}^h)$ represents the nonlinear function. The external force \mathbf{f}^{ext} may also represent a nonlinearity but for simplicity, we neglect this case in our paper. In the remainder of our paper, we omit the use of the superscript h denoting the finite element approximation of \mathbf{u}^h to \mathbf{u} .

3. Time discretization and Newton iteration. We employ the Generalized- α method introduced in [4] and described in [7] for time integration. In a discrete time interval $[t_n, t_{n+1}]$ with $\Delta t = t_{n+1} - t_n$, the ansatz is

$$\mathbf{u}_{n+1} = \mathbf{u}_n + \Delta t \dot{\mathbf{u}}_n + \Delta t^2 \left(\left(\frac{1}{2} - \beta \right) \ddot{\mathbf{u}}_n + \beta \ddot{\mathbf{u}}_{n+1} \right), \quad (3.1a)$$

$$\dot{\mathbf{u}}_{n+1} = \dot{\mathbf{u}}_n + \Delta t ((1 - \gamma) \ddot{\mathbf{u}}_n + \gamma \ddot{\mathbf{u}}_{n+1}), \quad (3.1b)$$

$$\ddot{\mathbf{u}}_{n+\alpha_m} = (1 - \alpha_m) \ddot{\mathbf{u}}_{n+1} + \alpha_m \ddot{\mathbf{u}}_n, \quad (3.1c)$$

$$\dot{\mathbf{u}}_{n+\alpha_f} = (1 - \alpha_f) \dot{\mathbf{u}}_{n+1} + \alpha_f \dot{\mathbf{u}}_n, \quad (3.1d)$$

where β, γ, α_m and α_f are constants controlling the type of time integration used and the amount of numerical energy dissipation desired. Inserting (3.1) into (2.5) yields the residual expression

$$\begin{aligned} \mathbf{r}(\mathbf{u}_{n+1}) = & (1 - \alpha_f) \mathbf{f}^{\text{int}}(\mathbf{u}_{n+1}) - (1 - \alpha_f) \mathbf{f}^{\text{trans}}(\mathbf{u}_{n+1}, t_{n+1}) \\ & + \alpha_f \mathbf{f}^{\text{int}}(\mathbf{u}_n) - \alpha_f \mathbf{f}^{\text{trans}}(\mathbf{u}_n, t_n) + \mathbf{f}^{\text{ext}}, \end{aligned} \quad (3.2a)$$

$$\mathbf{f}^{\text{trans}} = \mathbf{M} \left[\frac{1 - \alpha_m}{\beta \Delta t^2} (\mathbf{u}_{n+1} - \mathbf{u}_n) - \frac{1 - \alpha_m}{\beta \Delta t} \dot{\mathbf{u}}_n - \left(\frac{1 - \alpha_m}{2\beta} - 1 \right) \ddot{\mathbf{u}}_n \right] \quad (3.2b)$$

where \mathbf{M} is the mass matrix that is independent of t and \mathbf{u}_n . A Newton iteration (1.2) determines \mathbf{u}_{n+1} , and is given by

$$\mathbf{r}'(\mathbf{u}_{n+1}^k) \Delta \mathbf{u} = -\mathbf{r}(\mathbf{u}_{n+1}^k), \mathbf{u}_{n+1}^{k+1} = \mathbf{u}_{n+1}^k + \Delta \mathbf{u}, k \leftarrow k + 1 \quad (3.3)$$

where the Jacobian is given by

$$\mathbf{r}'(\mathbf{u}_{n+1}^k) = \frac{1 - \alpha_m}{\beta \Delta t^2} \mathbf{M} + (1 - \alpha_f) \mathbf{K}_T, \quad (3.4a)$$

$$\mathbf{K}_T = \frac{\partial \mathbf{f}_{n+1}^{\text{int}}(\mathbf{u})}{\partial \mathbf{u}} \Big|_{\mathbf{u}_{n+1}^k}. \quad (3.4b)$$

The Jacobian represents the effective tangent stiffness matrix given the tangent stiffness matrix \mathbf{K}_T . Once the Newton iteration (3.3) is converged, velocities and accelerations are updated using

$$\dot{\mathbf{u}}_{n+1} = \frac{\gamma}{\beta \Delta t} (\mathbf{u}_{n+1} - \mathbf{u}_n) - \frac{\gamma - \beta}{\beta} \dot{\mathbf{u}}_n - \frac{\gamma - 2\beta}{2\beta} \Delta t \ddot{\mathbf{u}}_n, \quad (3.5a)$$

$$\ddot{\mathbf{u}}_{n+1} = \frac{1}{\beta \Delta t^2} (\mathbf{u}_{n+1} - \mathbf{u}_n) - \frac{1}{\beta \Delta t} \dot{\mathbf{u}}_n - \frac{1 - 2\beta}{2\beta} \ddot{\mathbf{u}}_n. \quad (3.5b)$$

to advance the time step.

The efficiency of the above nonlinear transient elastic simulation crucially depends upon the size of the time step Δt , and the ability to solve the linear set of equations in (3.3). A robust method of solution for (3.3) enables large time steps and the ability to efficiently solve the linear set of equations.

The time step is bounded from above by two factors: First and foremost, the amount of tolerated time discretization error and error propagation. This aspect is usually either addressed by time adaptivity using time error estimators or indicators, or is simply ignored and the time step size is chosen by rule of thumb (a common approach). The second factor that limits the size of the time step is the efficiency associated with the numerical solution of the Newton iteration (3.3). For instance, the condition number of the linear set of equations in (3.4a) becomes that of the tangent stiffness matrix \mathbf{K}_T as the time step is increased. Hence if \mathbf{K}_T is ill-conditioned, then so is the Jacobian.

The first time step size upper bound mentioned above is of minor relevance when forces of inertia play a minor role and the problem at hand may be addressed by a quasistatic approach. In such cases, transient structural dynamics is applied to circumvent certain numerical difficulties such as loss of definiteness of \mathbf{K}_T in (3.4b) that arise in quasistatic nonlinear mechanics in the context of structural instability e.g. buckling. In such cases \mathbf{M} has a strong regularizing effect on the numerical solution method. In this case, the size of Δt is limited by the ability of the nonlinear solution method to converge (3.2a).

4. Pseudo-Transient Continuation. We make the assumption that β, γ, α_m and α_f in (3.1) are selected so that

$$\frac{d\mathbf{u}_{n+1}}{d\tau} = -\mathbf{r}(\mathbf{u}_{n+1}), \quad \mathbf{u}_{n+1}(0) = \mathbf{u}_{n+1}^0,$$

has a steady-state. Therefore we apply Ψtc by replacing the Newton iteration (3.3) with the Ψtc iteration (1.4) where $\mathbf{u}^k \leftarrow \mathbf{u}_{n+1}^k$ resulting in

$$(\delta_k^{-1} \mathbf{I} + \mathbf{r}'(\mathbf{u}_{n+1}^k)) \Delta \mathbf{u} = -\mathbf{r}(\mathbf{u}_{n+1}^k), \mathbf{u}_{n+1}^{k+1} = \mathbf{u}_{n+1}^k + \Delta \mathbf{u}, k \leftarrow k + 1. \quad (4.1)$$

The goal of Ψtc is to increase the robustness of the Newton iteration and therefore allow for an increase in the time step Δt . The examples in the next section demonstrate the increased robustness of Ψtc over a standard Newton iteration.

Two approaches for managing δ_k during a Ψtc iteration are now discussed. The first, and the most common approach in the literature is “Switched Evolution Relaxation” (SER) [16]. In SER the new time step is

$$\delta_{k+1} = \min(\delta_k \|\mathbf{r}(u^k)\| / \|\mathbf{r}(u^{k+1})\|, \delta_{max}) = \min(\delta_0 \|\mathbf{r}(u^0)\| / \|\mathbf{r}(u^{k+1})\|, \delta_{max}).$$

Using $\delta_{max} = \infty$ is common. At present, the convergence theory requires that the time step be updated with SER.

An alternative, not supported by theory, is the temporal truncation error approach (TTE) [14]. TTE estimates the local truncation error and manages the growth of δ_k by attempting to bound the truncation error. This differs from the time step control undertaken for an accurate numerical integration in that the bound on the truncation error for TTE is large.

5. Numerical Example. We consider a thin-walled cylinder that is hinged at the top and bottom. The cylinder radius is 200mm, the height is 225mm and the wall thickness is 0.054mm. The cylinder is discretized with hybrid bilinear quadrilateral shell elements applying a 7-parameter shell formulation [3, 2] that includes wall thickness change resulting in an extremely ill-conditioned problem [11]. A compressible Neo-Hookean hyperelastic material with Young’s modulus $E = 1000$, Poisson ratio $\nu = 0.3$ and density $\rho = 10^{-6}$ is chosen. The cylinder is deformed by a time dependent Dirichlet boundary condition at the top of the cylinder shortening along its central axis by 10.8mm within $T = 0.405s$. Figure 5.1 displays the buckling of the cylinder under two discretizations. Under either discretization, the problem results in nonlinear behavior that challenges the nonlinear transient simulation.

We solve the nonlinear problem (3.2a) with Newton (3.3) and Ψtc (4.1) iterations, respectively. The initial pseudo time step δ_0 is $\delta_0 = 0.01$ where both the SER and TTE schemes in turn manage δ_k . We terminate the iterations when both conditions

$$\|\mathbf{r}(\mathbf{u}_{n+1})\| < 10^{-6}, \quad \|\Delta \mathbf{u}\| < 10^{-6} \quad (5.1)$$

are satisfied.

We present two sets of experiments. The first set compares the Newton and two Ψtc iterations on one problem size using three different time steps held constant during each transient simulation (for a total of nine simulations). This first set of experiments uses a sparse direct solver for the associated linear sets of equations. The second set of examples performs a mesh refinement resulting in five problem sizes, and replaces the sparse direct solver with a preconditioned iterative method. The Newton and two Ψtc iterations are compared on each of the five problems using a time step held constant during the transient simulations.

Our first set of experiments used the three time steps

$$\Delta t = 7.5 \cdot 10^{-4}, 8.0 \cdot 10^{-4}, 2.0 \cdot 10^{-3}. \quad (5.2)$$

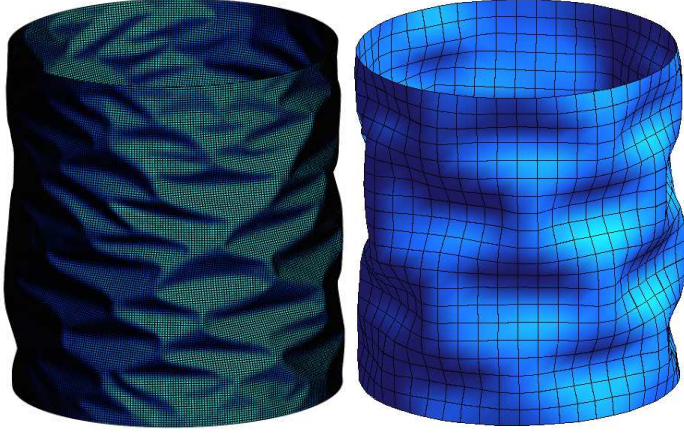


FIG. 5.1. *Buckling of cylinder. Spatial configuration at $T = 0.405$ s, left: 64000 elements, right: 1000 elements.*

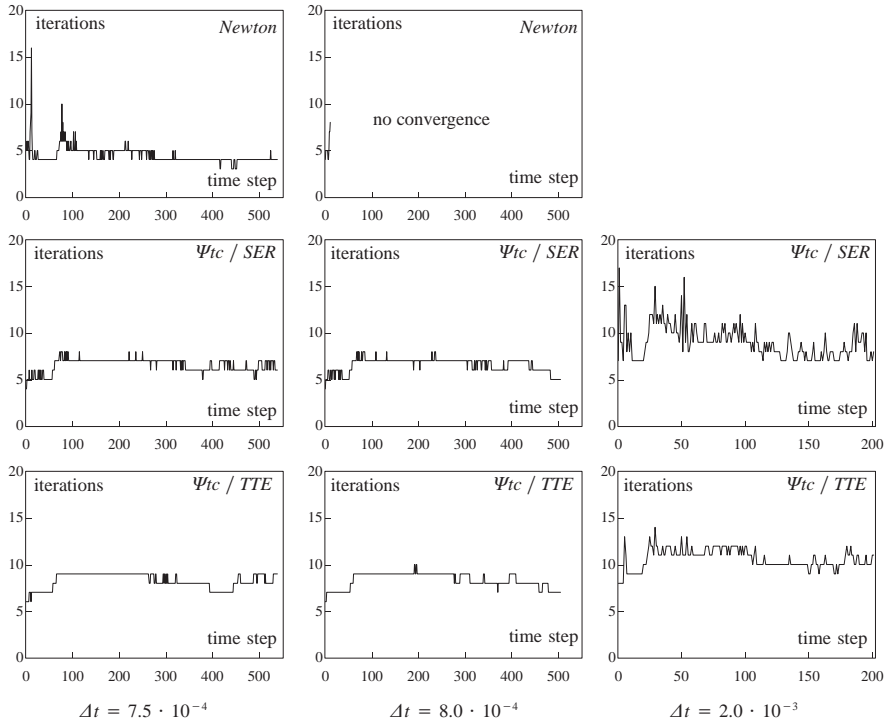


FIG. 5.2. *Comparison of iteration numbers for Newton, Ψ_{tc}/SER and Ψ_{tc}/TTE for selected time step sizes*

Figure 5.2 displays iteration numbers for each time step size associated with the Newton and two Ψ_{tc} iterations.

All three iterations converge for $\Delta t = 7.5 \cdot 10^{-4}$ with Ψ_{tc} usually taking one to two more iterations than Newton's method on average throughout the simulation. The Newton iteration does not converge for $\Delta t = 8.0 \cdot 10^{-4}$, $2.0 \cdot 10^{-3}$ while both Ψ_{tc}

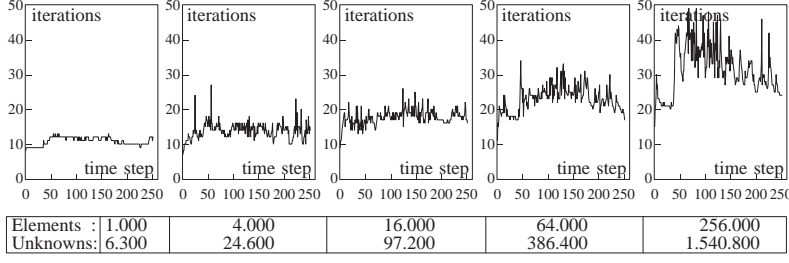


FIG. 5.3. Scalability of iteration numbers for Ψ_{tc}/SER at fixed $\Delta t = 1.6 \cdot 10^{-3}$

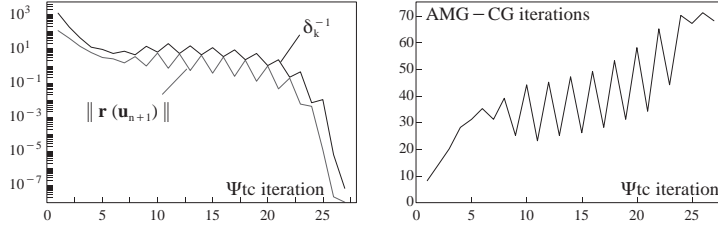


FIG. 5.4. $SER-\Psi_{tc}$ iteration in time step 176 for discretization with 64,000 elements. Left: Residual from (3.2a) and Ψ_{tc} parameter from (1.4). Right: Increasing values of δ_k^{-1} lead to a decrease in total preconditioned iterations.

variants (SER, TTE) converge. The SER variant usually uses one to two iterations less than the slightly more expensive TTE variant of Ψ_{tc} . Note that the final time step of $\Delta t = 2.0 \cdot 10^{-3}$ used during a transient simulation is more than twice as large than the first time step used during a transient simulation demonstrating that Ψ_{tc} is more robust than the Newton's iteration. Decreasing the initial pseudo time step size δ_0 in (1.4) allows for even larger Δt .

Our second set of examples performs a mesh refinement where the finite element ratio is held constant during refinement, and uses a preconditioned conjugate gradient iteration (discussed below) for the linear systems. Figure 5.1 displays two out of five discretizations investigated. The time step size is held fixed at $\Delta t = 1.6 \cdot 10^{-3}$ and the SER variant of Ψ_{tc} is applied using the convergence criteria (5.1). The Newton iteration was unable to converge with $\Delta t = 1.6 \cdot 10^{-3}$ using the preconditioned conjugate gradient iteration for the associated linear systems.

Figure 5.3 displays iteration numbers for the five discretizations. We observe that $SER-\Psi_{tc}$ converges for each discretization with the total number of iterations increasing with problems size. This growth of iterations is due to the geometric nonlinearity that becomes more severe (see Figure 5.1) for refined discretizations as these can resolve the buckling effects in more detail.

We end this section with a brief discussion of the preconditioned iterative method employed in the second set of experiments. A parallel conjugate gradient iteration preconditioned by a smoothed aggregation algebraic multigrid (SA-AMG) [23, 22] was used. The convergence criteria used was

$$\|\mathbf{r}_k^{\text{lin}}\| / \|\mathbf{r}_0^{\text{lin}}\| < 10^{-6}$$

where \mathbf{r}^{lin} is the residual of the linear system solved at each nonlinear iteration. An Additive-Schwarz-Gauss-Seidel is applied as the smoother on all multigrid levels

except the coarsest where an LU factorization is applied. Figure 5.4 demonstrates that the efficiency of the SA-AMG preconditioned conjugate gradient iteration benefits as δ_k^{-1} increases. The addition of $\delta_k^{-1}\mathbf{I}$ regularizes the linear set of equations associated with the Newton iteration when δ_k is small. This occurs during the early portion of the Newton iteration, when the iterates are possibly not within the region of local Newton convergence.

REFERENCES

- [1] U. M. ASCHER AND L. R. PETZOLD, *Computer Methods for Ordinary Differential Equations and Differential Algebraic Equations*, SIAM, Philadelphia, 1998.
- [2] M. BISCHOFF AND E. RAMM, *Shear deformable shell elements for large strains and rotations*, International Journal for Numerical Methods in Engineering, 40 (1994), pp. 4427–4449.
- [3] N. BUECHTER, E. RAMM, AND D. ROEHL, *Three-dimensional extension of nonlinear shell formulation based on the enhanced assumed strain concept*, International Journal for Numerical Methods in Engineering, 37 (1994), pp. 2551–2568.
- [4] J. CHUNG AND G. HULBERT, *A time integration algorithm for structural dynamics with improved numerical dissipation: The generalized $-\alpha$ method*, Journal of Applied Mechanics, 60 (1993), pp. 371–375.
- [5] T. COFFEY, C. T. KELLEY, AND D. E. KEYES, *Pseudo-transient continuation and differential-algebraic equations*, SIAM J. Sci. Comput., 25 (2003), pp. 553–569.
- [6] T. S. COFFEY, R. J. McMULLAN, C. T. KELLEY, AND D. S. McRAE, *Globally convergent algorithms for nonsmooth nonlinear equations in computational fluid dynamics*, J. Comp. Appl. Math., 152 (2003), pp. 69–81.
- [7] M. CRISFIELD, *Non-linear Finite Element Analysis of Solids and Structures, Vol II*, John Wiley & Sons, 1997.
- [8] M. W. FARTHING, C. E. KEES, T. COFFEY, C. T. KELLEY, AND C. T. MILLER, *Efficient steady-state solution techniques for variably saturated groundwater flow*, Advances in Water Resources, 26 (2003), pp. 833–849.
- [9] K. R. FOWLER AND C. T. KELLEY, *Pseudo-transient continuation for nonsmooth nonlinear equations*, SIAM J. Numer. Anal., 43 (2005), pp. 1385–1406.
- [10] C. W. GEAR, *Numerical Initial Value Problems in Ordinary Differential Equations*, Prentice-Hall, Englewood Cliffs, 1971.
- [11] M. GEE, E. RAMM, AND W. WALL, *Parallel multilevel solution of nonlinear shell structures*, Computational Methods in Applied Mechanics and Engineering, 194 (2005), pp. 2513–2533.
- [12] C. T. KELLEY AND D. E. KEYES, *Convergence analysis of pseudo-transient continuation*, SIAM J. Numer. Anal., 35 (1998), pp. 508–523.
- [13] D. E. KEYES, *Aerodynamic applications of Newton-Krylov-Schwarz solvers*, in Proc. of 14th International Conference on Num. Meths. in Fluid Dynamics, R. N. et al. eds, ed., New York, 1995, Springer, pp. 1–20.
- [14] D. E. KEYES AND M. D. SMOOKE, *A parallelized elliptic solver for reacting flows*, in Parallel Computations and Their Impact on Mechanics, A. K. Noor, ed., ASME, 1987, pp. 375–402.
- [15] D. KNOLL AND P. MCHUGH, *Enhanced nonlinear iterative techniques applied to a nonequilibrium plasma flow*, SIAM J. Sci. Comput., 19 (1998), pp. 291–301.
- [16] W. MULDER AND B. V. LEER, *Experiments with implicit upwind methods for the Euler equations*, J. Comp. Phys., 59 (1985), pp. 232–246.
- [17] P. D. ORKWIS AND D. S. McRAE, *Newton's method solver for high-speed separated flowfields*, AIAA Journal, 30 (1992), pp. 78–85.
- [18] ———, *Newton's method solver for the axisymmetric Navier-Stokes equations*, AIAA Journal, 30 (1992), pp. 1507–1514.
- [19] L. F. SHAMPINE, *Numerical Solution of Ordinary Differential Equations*, Chapman and Hall, New York, 1994.
- [20] A. I. SHESTAKOV, J. L. MILOVICH, AND A. NOY, *Solution of the nonlinear Poisson-Boltzmann equation using pseudo-transient continuation and the finite element method*, J. Colloid Interface Sci, 247 (2002), pp. 62–79.
- [21] M. D. SMOOKE, R. MITCHELL, AND D. KEYES, *Numerical solution of two-dimensional axisymmetric laminar diffusion flames*, Combust. Sci. and Tech., 67 (1989), pp. 85–122.
- [22] P. VANĚK, M. BREZINA, AND J. MANDEL, *Convergence of algebraic multigrid based on smoothed aggregation*, Numer. Math., 88 (2001), pp. 559–579.
- [23] P. VANĚK, J. MANDEL, AND M. BREZINA, *Algebraic multigrid based on smoothed aggregation*

- for second and fourth order problems*, Computing, 56 (1996), pp. 179–196.
- [24] V. VENKATAKRISHNAN, *Newton solution of inviscid and viscous problems*, AIAA Journal, 27 (1989), pp. 885–891.