Testing Line Search Techniques for Finite Element Discretizations for Unsaturated Flow

Fred T. Tracy

Information Technology Laboratory
Engineer Research and Development Center (ERDC)
3909 Halls Ferry Road
Vicksburg, MS USA 39180
`Fred.T.Tracy@usace.army.mil`

Abstract. Unsaturated flow in porous media is often modeled using the finite element (FE) method. When employing an implicit version of the computational equations resulting from the FE discretization, a nonlinear system of equations is generated that must then be solved by techniques such as Newton's method. This paper reveals results of the effectiveness of three line search techniques when employed within the framework of the approximate Newton method. The methods of a bisection line search, a quadratic variation of the norm of the residual line search, and a relaxation technique are considered, all in the context of a parallel computing environment.

1 Introduction

Unsaturated flow in porous media creates special difficulties for any computational model, including the finite element method [4]. Often, an implicit method is used to prevent the need for extremely small time-steps. When this technique is employed, a significantly difficult system of nonlinear equations is generated from the discretization, often having millions of unknowns. A common solution to this system of nonlinear equations is to use the approximate Newton method [5]. Often, however, the computed change in the unknown variable (total head in the case of flow in porous media with constant density of the water) is too much, so a globalization technique of some kind [6] must be employed. Globalization can involve both line search algorithms and trust region methods. As it is not always practical to consider all points inside a trust region to get the optimum point, ways to limit the search have been developed. One such idea is the dog-leg method [7]. However, the scope of this work is to consider only line search methods. The purpose of this paper is to show results of using the three line search techniques of bisection, a quadratic variation of the norm of the residual, and a relaxation technique on a test problem that has proven especially difficult to solve. Results were obtained from a research version of a three-dimensional (3-D) finite element (FE) groundwater program developed at ERDC and running on the Cray XT3 using 16 cores.

2 Governing Equations

The version of Richards' equation used in the finite element code is

$$\nabla \cdot (k_r \mathbf{K}_s \cdot \nabla \phi) = \eta \frac{\partial S}{\partial t} + S S_s \frac{\partial \phi}{\partial t} , \quad (1)$$

with

$$\phi = h + z , \quad (2)$$

where k_r is the relative hydraulic conductivity, $\mathbf{K}_s$ is the saturated hydraulic conductivity tensor, ϕ is the total head, η is the porosity, S is the saturation, S_s is the specific storage, h is the pressure head, and z is the z coordinate. k_r and S are functions of h , thus creating the strong nonlinearity.

3 Finite Element Solution

The following equation represents the completely implicit Euler discretization of the standard continuous Galerkin technique used in the study:

$$\frac{1}{\Delta t} \mathbf{M}(\mathbf{u}^{n+1}) \cdot [\mathbf{u}^{n+1} - \mathbf{u}^n] + \mathbf{K}(\mathbf{u}^{n+1}) \cdot \mathbf{u}^{n+1} = \mathbf{q}(\mathbf{u}^{n+1}) , \quad (3)$$

where Δt is the time-step size; n is the time-step number; $\mathbf{M}$ is the mass matrix as a function of $\mathbf{u}$; $\mathbf{u}^{n+1}$ is the vector of unknown total head at the nodes at time-step, $n+1$; and $\mathbf{q}$ is the vector of known flow type terms at the nodes as a function of $\mathbf{u}$. Since $\mathbf{M}$, $\mathbf{K}$, and $\mathbf{q}$ are functions of the unknown vector, $\mathbf{u}^{n+1}$, the nonlinearity is evident. The residual at the nodes as a function of $\mathbf{u}$ is therefore

$$\mathbf{r}(\mathbf{u}) = \mathbf{q}(\mathbf{u}) - \frac{1}{\Delta t} \mathbf{M}(\mathbf{u}) \cdot [\mathbf{u} - \mathbf{u}^n] - \mathbf{K}(\mathbf{u}) \cdot \mathbf{u} . \quad (4)$$

Using, (4), column j of the approximate Jacobian matrix can now be determined by

$$\mathbf{J}_j^{n+1,k+1} = \frac{\mathbf{r}(\mathbf{u}^{n+1,k} + \Delta u_j \hat{\mathbf{e}}_j) - \mathbf{r}(\mathbf{u}^{n+1,k} - \Delta u_j \hat{\mathbf{e}}_j)}{2\Delta u_j} , \quad (5)$$

where $\mathbf{u}^{n+1,k}$ is the value of $\mathbf{u}^{n+1}$ at the k^{th} nonlinear iteration, Δu_j is a small change in the j^{th} element of the vector, $\mathbf{u}^{n+1,k}$, and $\hat{\mathbf{e}}_j$ is a vector containing all zeroes except with a one in the j^{th} element. The linear system,

$$\mathbf{J}^{n+1,k+1} \cdot \delta \mathbf{u}^{n+1,k+1} = -\mathbf{r}(\mathbf{u}^{n+1,k}) , \quad (6)$$

can now be solved for the change in $\mathbf{u}$, and the new value of $\mathbf{u}^{n+1}$ at nonlinear iteration, $k+1$, now at least temporarily becomes

$$\mathbf{u}^{n+1,k+1} = \mathbf{u}^{n+1,k} + \delta \mathbf{u}^{n+1,k+1} . \quad (7)$$

4 Line Search Techniques

If $\|\delta \mathbf{u}\|_\infty$ becomes sufficiently small, the nonlinear iteration has converged. Otherwise, the process must be continued in some way. Three techniques were considered in this study, and they will now be briefly described.

4.1 Relaxation

Here, a parameter, β , where $0 < \beta_{\min} \leq \beta \leq \beta_{\max} \leq 1$, is used to reduce the change in $\mathbf{u}$ from one nonlinear iteration to the next. That is,

$$\mathbf{u}^{n+1,k+1} = \mathbf{u}^{n+1,k} + \beta_{k+1} \delta \mathbf{u}^{n+1,k+1}, \quad (8)$$

where β_{k+1} is the value of β at nonlinear iteration, $k+1$. For this option, only one value of β is considered in a given nonlinear iteration. Various values of β have been considered [2]. In this study, β was started at β_{init} and increased by β_{add} if $\|\delta \mathbf{u}^{n+1,k+1}\|_\infty \leq \|\delta \mathbf{u}^{n+1,k}\|_\infty$ (that is, $\beta_{k+2} = \beta_{k+1} + \beta_{\text{add}}$) and multiplied by β_{reduce} if $\|\delta \mathbf{u}^{n+1,k+1}\|_\infty > \|\delta \mathbf{u}^{n+1,k}\|_\infty$ (that is, $\beta_{k+2} = \beta_{\text{reduce}} \beta_{k+1}$). Care was also taken not to go smaller than $\beta_{\min}$ or larger than $\beta_{\max}$. Values that were found to work well are $\beta_{\min} = 0.1$, $\beta_{\max} = 1.0$, $\beta_{\text{add}} = 0.005$, and $\beta_{\text{reduce}} = 0.677$. β_{init} varies depending on how aggressive the user chooses to be. β_{init} was set to 0.2 in this study.

4.2 Bisection Line Search

This technique is a line search where β_{k+1} is successively reduced during the given nonlinear iteration, $k+1$. First, using the notation, $\mathbf{r}(\mathbf{u}^{n+1,k+1}) = \mathbf{r}^{n+1,k+1}$, $\|\mathbf{r}^{n+1,k+1}\|_2$ is computed from (8) and (4) with β_{k+1} initially set to 1. If $\|\mathbf{r}^{n+1,k+1}\|_2$ is greater than $\|\mathbf{r}^{n+1,k}\|_2$, then β_{k+1} is reduced by one-half, and (8) and (4) are again used to compute the residual. If the norm of this new residual is smaller than $\|\mathbf{r}^{n+1,k}\|_2$, the process ends. Otherwise, β_{k+1} is again cut by one-half, and the above algorithm is repeated. After a relatively small number of bisections (maximum of 10), the attempt at a line search for this nonlinear iteration is discontinued, and the next time-step is started.

4.3 Quadratic Variation of the Norm of the Residual Line Search

Using the definitions,

$$\begin{aligned} q_1 &= \|\mathbf{r}(\mathbf{u}^{n+1,k})\|_2 = \|\mathbf{r}^{n+1,k}\|_2, \\ q_2 &= \left\| \mathbf{r} \left(\mathbf{u}^{n+1,k} + \frac{1}{2} \delta \mathbf{u}^{n+1,k+1} \right) \right\|_2, \\ q_3 &= \left\| \mathbf{r}(\mathbf{u}^{n+1,k} + \delta \mathbf{u}^{n+1,k+1}) \right\|_2, \end{aligned} \quad (9)$$

the norm of the residual is assumed to vary quadratically as

$$q = (2q_1 - 4q_2 + 2q_3) \zeta^2 - (3q_1 - 4q_2 + q_3) \zeta + q_1, \quad (10)$$

when $0 \leq \zeta \leq 1$. Also, $\frac{dq}{d\zeta} = 0$ at

$$\zeta_0 = \frac{3q_1 - 4q_2 + q_3}{4(q_1 - 2q_2 + q_3)} . \quad (11)$$

Now if $\frac{d^2q}{d\zeta^2} > 0$ and $0 < \zeta_0 < 1$, then $\beta_{k+1} = \zeta_0$ and

$$\mathbf{u}^{n+1,k+1} = \mathbf{u}^{n+1,k} + \zeta_0 \delta \mathbf{u}^{n+1,k+1} . \quad (12)$$

However, if this fails but $q_3 < q_1$, then $\beta_{k+1} = 1$ and

$$\mathbf{u}^{n+1,k+1} = \mathbf{u}^{n+1,k} + \delta \mathbf{u}^{n+1,k+1} \quad (13)$$

is used. Otherwise, the $(k+1)^{\text{th}}$ nonlinear iteration is advanced one-fourth of the way with $\beta_{k+1} = 1/4$ and

$$\mathbf{u}^{n+1,k+1} = \mathbf{u}^{n+1,k} + \frac{1}{4} \delta \mathbf{u}^{n+1,k+1} . \quad (14)$$

5 Test Problem with Analytical Solutions

The problem consists of a rectangular box-shaped soil sample with dimensions, $a \times b \times L$, that is initially dry until water is poured at the top of the sample (see the vertical cross-sectional view in Fig. 1). This 3-D problem has analytical solutions [9,10], and it can be made arbitrarily difficult by changing a parameter, α . In this particular test, b is made small, so the two-dimensional (2-D) version of the boundary conditions and analytical solution is used. The initial condition is $\phi = h_d + z$, where h_d is the pressure head of the soil when it is very dry. The boundary condition on the sides and bottom are the same as the initial condition, and the boundary condition on the top is

$$\phi = \frac{1}{\alpha} \ln \left[e^{\alpha h_d} + (1 - e^{\alpha h_d}) \sin \frac{\pi x}{a} \right] + L , \quad (15)$$

where x is the x coordinate. Also, k_r is modeled by the quasi-linear equation [1,8],

$$k_r = e^{\alpha h} , \quad (16)$$

and S varies linearly with k_r [3,11] as

$$\frac{S - S_d}{1 - S_d} = k_r , \quad (17)$$

where S_d is the saturation when the soil is very dry.

Fig. 1. Vertical cross-section of the test problem

5.1 Computational Details

The above described problem was run with $a = 50$ cm, $b = 0.25$ cm, $L = 50$ cm, $\mathbf{K}_s$ = a diagonal matrix with each diagonal element being 0.1 cm day^{-1} , $h_d = -50$ cm, $\eta = 0.45$, $S_d = 1/3$, and $S_s = 0$. The 2-D cross-section in Fig. 1 was first divided into a grid of $200 \times 1 \times 200$ cells with $\Delta x = \Delta y = \Delta z = 0.25$ cm, and then each cell was divided into 6 tetrahedral elements, which is what the 3-D FE groundwater program requires. Data sets for relative hydraulic conductivity given in (16) and saturation modeled by (17) were then computed using 1001 values of pressure head and provided to the FE program input file. Inside the program, these curves are treated as piecewise linear.

The linear solver that was used is BiCG-Stab. Before the linear system of equations, $\mathbf{Ax} = \mathbf{b}$, as given in (6) is solved, it is normalized by

$$(\mathbf{FAF})(\mathbf{F}^{-1}\mathbf{x}) = \mathbf{Fb} \quad (18)$$

or

$$\hat{\mathbf{A}}\hat{\mathbf{x}} = \hat{\mathbf{b}} \quad (19)$$

where $\mathbf{F}$ is a diagonal matrix whose i^{th} term is

$$\mathbf{F}_{ii} = \frac{1}{\sqrt{\max(\max_j |a_{ij}|, \epsilon)}} \quad (20)$$

where a_{ij} is the $(ij)^{\text{th}}$ component of $\mathbf{A}$, and ϵ is a small number. After the linear solve is completed, the final solution is computed by

$$\mathbf{x} = \mathbf{F}\hat{\mathbf{x}} \quad (21)$$

Since the equations are normalized with most of the diagonal terms of $\mathbf{A}$ now being 1, no further preconditioning is done in the BiCG-Stab solver.

5.2 Line Search Test Results

Table 1 shows nonlinear count results and timings for the first time-step for different values of α , Δt , and nonlinear iteration types when using a rather strong nonlinear convergence criterion of 10^{-5} for $\|\delta \mathbf{u}^{n+1,k+1}\|_\infty$. Each nonlinear iteration requires at least one linear solve of (6) and a computation of $\|\delta \mathbf{u}^{n+1,k+1}\|_\infty$. The bisection line search requires repeated calculation of the residual norm, $\|\mathbf{r}(\mathbf{u})\|_2$, which is computationally equivalent to computing the norm of the right-hand side of (6). This is an element-by-element computation of (4). The quadratic line search requires the computation of $\|\mathbf{r}(\mathbf{u})\|_2$ at the middle and end of the Newton step. The relaxation method does not require any extra computations of (4).

Each linear iteration in BiCG-Stab is dominated by two matrix-vector multiplications of the type, $\hat{\mathbf{A}}\mathbf{v}$, where $\mathbf{v}$ represents utility temporary vectors. Both $\hat{\mathbf{A}}$ and $\mathbf{v}$ are distributed over the parallel cores, thus requiring ghost node updating at strategic times using MPI. The linear solve is typically the dominant computation.

The running times given in Table 1 do not precisely delineate the impact of the different factors. To get at least a clearer understanding of these timing results, the number of linear iterations in BiCG-Stab for the first five nonlinear iterations

Table 1. Nonlinear iteration count and time in seconds for the first time-step for the three line search types, two values of α , and three values of Δt

α (cm ⁻¹)	Δt (day)	Line search type	Nonlinear iteration count	Time (sec)
0.05	0.1	Bisection	7	3.27
0.05	0.1	Quadratic	14	5.18
0.05	0.1	Relaxation	42	12.20
0.05	0.01	Bisection	8	2.77
0.05	0.01	Quadratic	15	4.21
0.05	0.01	Relaxation	45	8.37
0.05	0.001	Bisection	5	2.24
0.05	0.001	Quadratic	12	3.33
0.05	0.001	Relaxation	41	6.81
0.2	0.1	Bisection	23	5.58
0.2	0.1	Quadratic	29	7.97
0.2	0.1	Relaxation	168	27.89
0.2	0.01	Bisection	12	3.51
0.2	0.01	Quadratic	18	4.73
0.2	0.01	Relaxation	52	9.08
0.2	0.001	Bisection	∞	-
0.2	0.001	Quadratic	17	4.27
0.2	0.001	Relaxation	46	7.83

Table 2. Linear iterations required for the first five nonlinear iterations for $\alpha = 0.05 \text{ cm}^{-1}$, the bisection method, and three values of Δt

Δt (day)	#1	#2	#3	#4	#5
0.1	63	44	50	30	3
0.01	24	15	13	10	6
0.001	7	4	3	1	1

of the first time-step are tabulated in Table 2 for $\alpha = 0.05 \text{ cm}^{-1}$, the bisection method, and the three values of Δt . These results show that the number of linear iterations per nonlinear time-step is significantly reduced when the time-step size is reduced, and this helps explain the reduction in the running times in Table 1. For example, the reduction of running time from 3.27 sec to 2.77 sec while the number of nonlinear iterations increased from 7 to 8 as the time-step size is changed from 0.1 day to 0.01 day is explained by the linear iterations going from 63 to 24, 44 to 15, 50 to 13, 30 to 10, and 3 to 6, respectively.

6 Conclusions and Further Research

Conclusions that can be drawn from the above results are as follows:

1. As α is increased, the number of nonlinear iterations for all three methods increase.
2. The bisection method almost always was the most efficient. However, it stalled at an unpredictable time. It stalled because 10 unsuccessful bisections generated a very small change in total head. This failure was repeated indefinitely.
3. The relaxation method almost always was the least efficient.
4. Significant improvement in running time could potentially be achieved by implementing a more sophisticated linear solver. This is an area of future work.
5. Reducing the time-step size typically reduced both the number of nonlinear iterations and the number of linear iterations per nonlinear iteration.
6. More sophisticated nonlinear solvers/globalization techniques should be investigated to improve robustness. The one nonlinear failure reported in this paper has been observed in solving other difficult problems. Further research is needed.

Acknowledgment

This work was supported in part by a grant of computer time from the DoD High Performance Computing Modernization Program.

References

1. Gardner, W.: Some steady-state solutions of the unsaturated moisture flow equation with application to evaporation from a water table. *Soil. Sci.* 85, 228–232 (1958)
2. Gill, P., Murray, W., Wright, M.: *Practical Optimization*, p. 319. Academic Press, London (1981)
3. Irmay, S.: On the hydraulic conductivity of unsaturated soils. *Econ. Transit., AGU* 35 (1954)
4. Istok, J.: *Groundwater Modeling by the Finite Element Method*. AGU (1989)
5. Kelley, C.: *Solving Nonlinear Equations with Newton's Method*. SIAM (2003)
6. Pawlowski, R., Shadid, J., Simonis, J., Walker, H.: Globalization techniques for Newton-Krylov methods and applications to the fully coupled solution of the Navier-Stokes equations. *SIAM Review* 48, 700–721 (2006)
7. Pawlowski, R., Simonis, J., Walker, H., Shadid, J.: Inexact Newton dogleg methods. *SIAM J. Numer. Anal.* 46, 2112–2132 (2008)
8. Segol, G.: *Classic Groundwater Simulations*, p. 352. Prentice Hall, Englewood Cliffs (1994)
9. Tracy, F.: Clean two- and three-dimensional analytical solutions of Richards equation for testing numerical solvers. *Water Resours. Res.* 42, W08503 (2006)
10. Tracy, F.: Three-dimensional analytical solutions of Richards' equation for a box-shaped soil sample with piecewise-constant head boundary conditions on the top. *J. Hyd.* 336, 391–400 (2007)
11. Warrick, A.: *Soil Water Dynamics*, pp. 247–258. Oxford University Press, Oxford (2003)