XVIII International Conference on Water Resources
CMWR 2010
J. Carrera (Ed)

AN ACCELERATED FIXED-POINT ITERATION FOR SOLUTION OF VARIABLY SATURATED FLOW

Homer F. Walker*, Carol S. Woodward† and Ulrike M. Yang†

*Department of Mathematical Sciences, Worcester Polytechnic Institute
Worcester, MA 01609-2280, USA
e-mail: walker@wpi.edu

†Center for Applied Scientific Computing, Lawrence Livermore National Laboratory
P.O. Box 808, L-561, Livermore, CA 94551, USA
e-mail: cswoodward@llnl.gov, umyang@llnl.gov

Key words: Richards' equation, Picard iteration, acceleration methods, Newton method

Summary. *We investigate effectiveness of an acceleration method applied to the modified Picard iteration for simulations of variably saturated flow. We solve nonlinear systems using both unaccelerated and accelerated modified Picard iteration as well as Newton's method. Since Picard iterations can be slow to converge, the advantage of acceleration is to provide faster convergence while maintaining advantages of the Picard method over the Newton method. Results indicate that the accelerated method provides a robust solver with significant potential computational advantages.*

1 INTRODUCTION

We consider the mixed form of Richards' equation (RE) [10],

$$\frac{\partial \theta(h)}{\partial t} - \nabla \cdot (K k_r(h) \nabla(h + z)) = q, \tag{1}$$

where h is pressure head; $\theta(h) = s(h)\phi$ is the water content, with water saturation $s(h)$ and porosity of the medium ϕ; K is the saturated hydraulic conductivity; $k_r(h)$ is the relative permeability of water to air; t is time; z is the vertical direction; and q is a source/sink term. Common numerical discretizations of (1) employ finite-element or finite-difference methods along with backward-Euler implicit time integration. With these schemes, an algebraic system of highly nonlinear equations must be solved within each time step.

Authors have approached solving this nonlinear system using a Newton method [7], a Picard (or fixed-point) iteration [6], or some combination. Although Newton's method can exhibit quadratic convergence, it suffers some drawbacks. First, even though the nonlinear problem formed from common discretizations of RE is symmetric, the linear Jacobian systems in Newton's method are nonsymmetric and are more problematic to

solve than symmetric systems. Also, computing the Jacobians or their approximations can be computationally expensive. Furthermore, the provably fast convergence of Newton's method requires certain continuity properties of the Jacobian. Some parameterizations of the van Genuchten curves for relative permeability [11] do not result in such continuity. In contrast, the Picard method involves a symmetric linear system (provided the discretized nonlinear system is symmetric) and has lower continuity requirements than Newton's method. The main drawback of Picard iteration has been its linear convergence rate.

A number of authors have looked at the use of these and related methods for RE [4, 9, 8]. In general, authors have found that no method is definitively better than others, although more recent work has focussed on use of either a modified Picard method or Newton's method for large-scale simulations. The modified Picard algorithm [4] linearizes the saturation with a first-order Taylor expansion, such as one would do in a Newton method, but then applies a standard fixed-point approach to the relative permeability.

Recently, an acceleration method originating in work of Anderson [1] has been shown to be potentially useful in a variety of applications [12]. Applied to the modified Picard iteration, this scheme shows promise to retain many of the advantages of the Picard scheme over Newton's method, such as symmetric linear systems and no relative permeability derivative requirements, while speeding up convergence of the iterates. In this paper, we present initial results showing robustness of the accelerated modified Picard method on steady-state and RE test cases. In addition, we explore the convergence properties of the scheme as compared with modified Picard and Newton methods. The next section outlines the Newton and modified Picard schemes as well as the Anderson acceleration method. Section 3 presents numerical results comparing the methods. In the last section, we make some concluding remarks.

2 NONLINEAR SOLVER METHODS

With our focus on convergence properties of nonlinear solvers, we limit discussion to 1D problems, uniform grids in space and time, second-order finite-difference schemes for spatial discretization, and backward-Euler time discretization. We expect findings on convergence to readily carry over to more general discretizations and grids.

Applying these discretizations to (1), for each grid point i we have the following discretized equation in terms of the vector of pressure-head values, $\mathbf{h}^n = [h_i^n], i = 1, ..., N$, ($N$ is the number of grid points) at time t^n:

$$\begin{aligned} F_i(\mathbf{h}^n) &\equiv \Delta z[\theta(h_i^n) - \theta(h_i^{n-1})] - \Delta t(u_{i+1/2} - u_{i-1/2}) - \Delta t \Delta z q_i = 0, \\ u_{i+1/2} &= (K(z)k_r(\mathbf{h}^n))\Big|_{i+1/2} \left[\frac{h_{i+1}^n - h_i^n}{\Delta z} + 1\right]. \end{aligned} \tag{2}$$

We write this equation in vector form as,

$$\mathbf{F}(\mathbf{h}^n) \equiv (\theta(\mathbf{h}^n) - \theta(\mathbf{h}^{n-1}))\Delta z + (A(\mathbf{h}^n)\mathbf{h}^n)\Delta t - \mathbf{q}\Delta z = 0, \tag{3}$$

where $A(\mathbf{h}^n)$ is the matrix arising from application of the finite difference spatial discretization, $\mathbf{q}$ is the vector formed from $[q_i \Delta t]$, and $\mathbf{h}^{n-1}$ is the pressure head at time t^{n-1}. We now derive the three iteration approaches. For each method, we consider advancing an approximation of $\mathbf{h}^n$ at time step t^n from iteration k to $k+1$; we thus replace the time superscript with an iteration superscript unless necessary to indicate a time other than t^n. We terminate the algorithms when $\|\mathbf{F}(\mathbf{h}^k)\| < tol$, where tol is a specified tolerance.

At the kth iteration, Newton's method proceeds by finding a root of the linear model

$$\mathbf{F}(\mathbf{h}^{k+1}) \approx \mathbf{F}(\mathbf{h}^k) + J_F(\mathbf{h}^k)(\mathbf{h}^{k+1} - \mathbf{h}^k), \tag{4}$$

where $J_F(\mathbf{h}^k)$ is the Jacobian of $\mathbf{F}$ evaluated at $\mathbf{h}^k$. The algorithm is:

Algorithm NI: Newton Iteration

Given $\mathbf{h}^0$.
For $k = 0, 1, \ldots,$ *until* $\|\mathbf{F}(\mathbf{h}^k)\| < tol$
Compute $J_F(\mathbf{h}^k)$.
Solve the system $J_F(\mathbf{h}^k)\Delta\mathbf{h}^k = -\mathbf{F}(\mathbf{h}^k)$.
Set $\mathbf{h}^{k+1} = \mathbf{h}^k + \Delta\mathbf{h}^k$.

Note that each iteration requires solution of a nonsymmetric linear system.

To define the modified Picard method, we employ the approximations

$$A(\mathbf{h}^{k+1})\mathbf{h}^{k+1} \approx A(\mathbf{h}^k)\mathbf{h}^{k+1}, \qquad \theta(\mathbf{h}^{k+1}) \approx \theta(\mathbf{h}^k) + J_\theta(\mathbf{h}^k)\Delta\mathbf{h}^k. \tag{5}$$

Combining these with the relation $\mathbf{h}^{k+1} = \mathbf{h}^k + \Delta\mathbf{h}^k$ and placing them into (3) results in

$$\tilde{\mathbf{F}}(\mathbf{h}^{k+1}) \equiv \left(\theta(\mathbf{h}^k) + J_\theta(\mathbf{h}^k)\mathbf{h}^{k+1} - J_\theta(\mathbf{h}^k)\mathbf{h}^k - \theta(\mathbf{h}^{n-1})\right)\Delta z + (A(\mathbf{h}^k)\mathbf{h}^{k+1})\Delta t - \mathbf{q}\Delta z. \tag{6}$$

Setting $\tilde{\mathbf{F}} = 0$, adding $(A(\mathbf{h}^k)\mathbf{h}^k - A(\mathbf{h}^k)\mathbf{h}^k)\Delta t$, and solving for $\Delta\mathbf{h}^k$ gives

$$\mathbf{h}^{k+1} = \mathbf{G}(\mathbf{h}^k) \equiv \mathbf{h}^k + \left(J_\theta(\mathbf{h}^k) + A(\mathbf{h}^k)\frac{\Delta t}{\Delta z}\right)^{-1}\left(\mathbf{q} - A(\mathbf{h}^k)\mathbf{h}^k\frac{\Delta t}{\Delta z} - \theta(\mathbf{h}^k) + \theta(\mathbf{h}^{n-1})\right). \tag{7}$$

Building this into an iterative method gives the modified Picard iteration. We note that the linear system is symmetric; however, the system includes a derivative of the saturation – pressure head relationship.

Algorithm MPI: Modified Picard Iteration

Given $\mathbf{h}^0$.
For $k = 0, 1, \ldots,$ *until* $\|\mathbf{F}(\mathbf{h}^k)\| < tol$
Set $\mathbf{h}^{k+1} = \mathbf{G}(\mathbf{h}^k)$.

While the modified Picard method has proven useful in many situations, it can be slower to converge than Newton. Acceleration methods can potentially alleviate slow convergence and, in some cases, divergence as well. Here, we consider an acceleration method originating in work of Anderson [1], which we formulate as follows:

Algorithm AA: Anderson Acceleration

Given $\mathbf{h}^0$ *and* $m \geq 1$.
Set $\mathbf{h}^1 = \mathbf{G}(\mathbf{h}^0)$.
For $k = 1, 2, \ldots,$ *until* $\|\mathbf{F}(\mathbf{h}^k)\| < tol$
  Set $m_k = \min\{m, k\}$.
  Set $\mathbf{F}_k = (\mathbf{f}_{k-m_k}, \ldots, \mathbf{f}_k)$, *where* $\mathbf{f}_i = \mathbf{G}(\mathbf{h}^i) - \mathbf{h}^i$.
  Compute $\alpha^{(k)} = (\alpha_0^{(k)}, \ldots, \alpha_{m_k}^{(k)})^T$ *that solves* $\min_\alpha \|\mathbf{F}_k\alpha\|_2$ s. t. $\sum_{i=0}^{m_k} \alpha_i = 1$.
  Set $\mathbf{h}^{k+1} = \sum_{i=0}^{m_k} \alpha_i^{(k)} \mathbf{G}(\mathbf{h}^{k-m_k+i})$.

The rationale for this algorithm is that, *if* $\mathbf{G}$ were linear, then $\mathbf{h}^{k+1} = \mathbf{G}(\mathbf{h}_{\min})$, where $\mathbf{h}_{\min} \equiv \sum_{i=0}^{m_k} \alpha_i \mathbf{h}^{k-m_k+i}$ has minimal fixed-point residual among all convex combinations of $\mathbf{h}_{k-m_k}, \ldots, \mathbf{h}_k$. The expense of the algorithm increases as iterations proceed, up to about $2m$ vectors of storage and (if properly implemented, see [12]) $O(Nm)$ arithmetic operations per iteration beyond those required for MPI.

Acceleration of Picard iterations for variably-saturated flow equations has also been considered in [5]. The basic accelerated method, termed the *Picard–Broyden* method in [5], is obtained by applying Broyden's method (specifically, the *first Broyden* method [3]) to the Picard system $\mathbf{h} - \mathbf{G}(\mathbf{h}) = 0$. In numerical experiments in [5], the Picard–Broyden method showed faster convergence than Picard iteration. Algorithm AA has a certain relationship to the *second Broyden* update [3]; however, it is not a straightforward application of the second Broyden method to the Picard system (see [12]).

As a last note on the methods, we point out that a rule of thumb has been that each Newton iteration takes about twice the run time of each Picard iteration due to the symmetry in the linear system giving rise to lower memory requirements and faster linear solves. In addition, the cost of computing derivatives of the relative permeabilities adds to the expense of Newton relative to Picard. The Anderson-accelerated Picard method considered here will cost more per iteration than Picard, depending on m, but we expect that each iteration will still be considerably faster than an iteration of Newton's method.

3 NUMERICAL TESTS

In our numerical tests, we examine solver performance on problems incorporating the van Genuchten parameterization for relative permeability, given as

$$k_r(h) = \begin{cases} 1, & h \geq 0, \\ (1 + (\alpha|h|)^n)^{-\frac{m}{2}} \left(1 - \frac{(\alpha|h|)^{n-1}}{[1+(\alpha|h|)^n]^m}\right)^2, & h < 0, \end{cases}$$

where $m = 1 - 1/n$. These problems generally become more difficult as n becomes smaller and α becomes larger. In particular, when $n < 2$, the derivative of k_r is no longer Lipschitz continuous at $h = 0$, and thus the function no longer satisfies conditions needed for provably fast convergence of Newton solvers.

We consider two test problems. The first is a steady-state problem designed specifically to examine performance of solvers on the nonlinearities particular to the van Genuchten

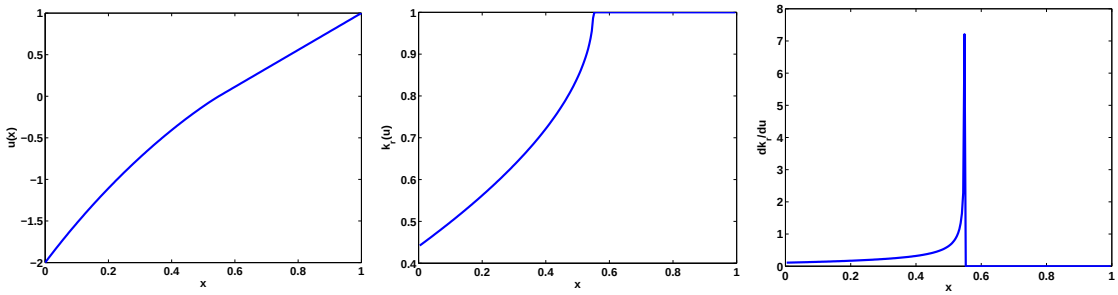

Figure 1: Solution (left), relative permeability (middle), and derivative of relative permeability (right) for slightly loamy sand as a function of space.

parameterization of relative permeability. The second is a RE test case from [4] in which the relative permeabilities and their derivatives are smooth.

3.1 Case 1: Steady-state elliptic problem

This is a test case from [13], posed on the domain $x \in (0, 1)$ and given as,

$$-(k_r(u)u')' = 0, \quad u(0) = -2, \quad u(1) = 1, \tag{8}$$

where u is the unknown, and k_r is given in (3).

We discretize (1) using (2) with arithmetic averages of $k_r(u_i)$ and $k_r(u_{i+1})$ at the half points, $i + 1/2$. We consider the cases given in Table 1. The "New Mexico" values are

Soil Class	α	n
New Mexico	0.0335	2
Slightly loamy sand	0.0599	1.51
Loam	0.206	1.09
Perturbation	0.207	1.1

Table 1: Parameter values for tests in Case 1.

from [4], and values for the "Slightly loamy sand" and "Loam" are from Table 4 in [2]. "Perturbation" values are included to test solver robustness. We use 256 grid points and take an initial solution estimate to be linear from -2 at $x = 0$ to 1 at $x = 1$.

Figure 1 shows the solution u, k_r, and the derivative of k_r over the spatial domain for the slightly loamy sand. We see that where the solution crosses 0 (near $x = 0.55$) the value of k_r rapidly approaches 1 and its derivative steeply increases. Note that as $n \to 1$, the relative permeability approximates a step function, and the derivative becomes infinite at the point $u = 0$. Note the sharp rise in the derivative seen in Figure 1.

Figure 2 shows convergence plots for the cases of Table 1. For the New Mexico data, all solvers converge quickly. The relative permeability and its derivative are smooth, and

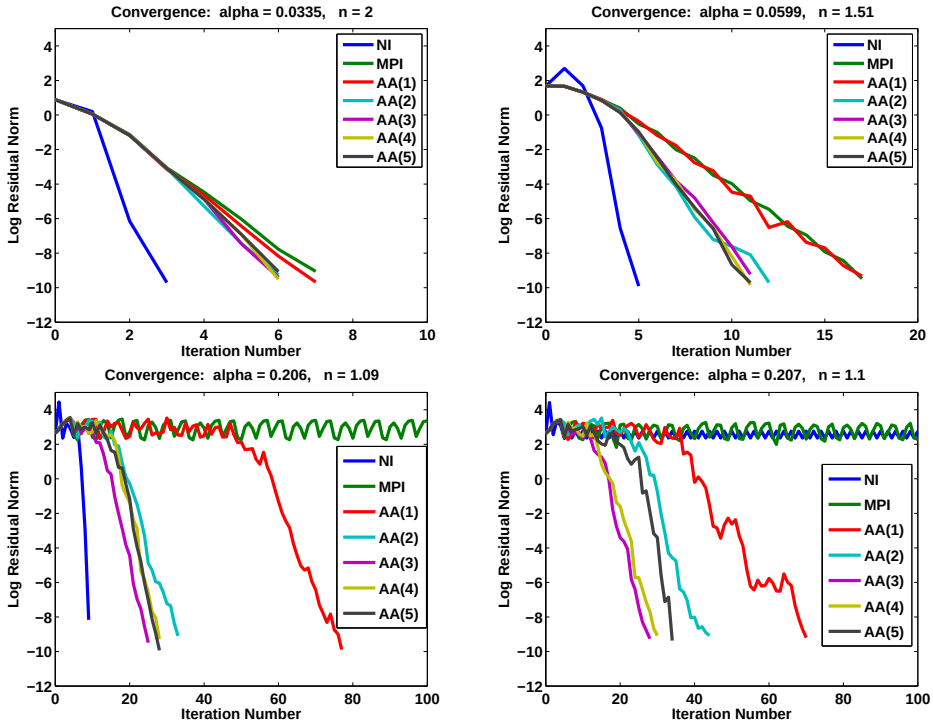

Figure 2: Nonlinear residual versus iteration number for the cases in Table 1: New Mexico (upper left), slightly loamy sand (upper right), loam (lower left), and the slight perturbation (lower right).

the MPI and AA methods (with values of m shown in parentheses) take about twice the iterations of NI. For the slightly loamy sand, although the derivative of k_r is not smooth, we see convergence of all methods with NI as the fastest. NI shows trouble initially with residual increases after the first couple of iterations. NI quickly recovers and is able to solve efficiently afterwards. AA gives reasonable convergence (about twice the iterations of NI) once $m > 1$. For the loam soil, the derivative of relative permeability is sharper. We see clear failure of the MPI method, with recovery of robustness with the AA scheme. NI still converges fastest, and now even the accelerated methods take more than twice the number of iterations of NI. The last case is a small perturbation of the loam soil. We see failure of both NI and MPI with convergence of the AA methods. We note the convergence numbers for the AA methods are very similar to those for the loam soil.

3.2 Case 2: Richards' Equation

This case is based on parameters in [4]. Here $K = 0.00922$ cm/s, $h(z, 0) = -1000$ cm, $h(0, t) = -1000$ cm, and $h(60 \text{ cm}, t) = -75$ cm. We take $\Delta t = 60$ min, $\Delta z = 2.5$ cm, and run for one day. The water content is given by $\theta(h) = (\theta_s - \theta_r)\left(1 + (\alpha|h|)^n\right)^{-m} + \theta_r$. Data is "New Mexico" from Table 1, with $\theta_s = 0.368$ and $\theta_r = 0.102$.

Table 2 shows the total number of iterations cumulative for the RE test case for each

solver. NI again requires the fewest iterations for convergence. As the tolerance decreases, requiring a smaller final nonlinear residual, we clearly see the difference between the quadratic convergence of NI and the linear convergence of MPI. Once the solution estimate is close to the true solution of the nonlinear problem, NI will reduce the error rapidly and give very significant improvements in the solution at each iteration. In contrast, MPI will not make as much progress with each iteration. This situation was seen in Figure 2 as well. Table 2 also shows a significant benefit in using algorithm AA over MPI alone. In fact, the number of iterations required for many of the AA variants is less than twice the number for NI, suggesting that further computational testing should be done to determine the viability of AA methods as an alternative to NI once the full computational cost is measured. The advantage of applying AA to MPI over NI decreases with decreasing tolerances since NI requires very few additional iterations to meet the tighter convergence requirements.

Solver	10^{-6}	10^{-8}	10^{-10}
NI	173	180	189
MPI	412	589	766
AA(1)	316	398	498
AA(2)	345	425	501
AA(3)	292	368	454
AA(4)	280	354	427
AA(5)	285	352	422

Table 2: Cumulative iteration counts for the RE test. Tolerances are shown at the top of each column.

4 CONCLUSIONS

Our results show that Anderson acceleration for the modified Picard method provides a robust nonlinear solver for the types of nonlinearities common in RE. The number of iterations required for solving with Anderson acceleration is roughly twice that of using Newton's method. Anderson-accelerated modified Picard does not require derivatives of the relative permeability and only requires solving symmetric systems. We thus expect each iteration of Anderson-accelerated modified Picard to be considerably faster than an iteration of Newton's method. Future work will include assessing the time for solution and for each iteration of these methods. We will also examine the relative performance of the Anderson-accelerated modified Picard method and the Picard-Broyden method of [5].

ACKNOWLEDGEMENTS

Walker's work was supported in part by NSF grant DMS 0915183. Woodward and Yang's work was performed under the auspices of the U.S. Department of Energy by

Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

REFERENCES

[1] D. G. Anderson, *Iterative procedures for nonlinear integral equations*, J. Assoc. Comput. Machinery, 12 (1965), pp. 547–560.

[2] K. Bohne, C. Nitsche, and F. J. Leij, *Requirements and use of indirect methods for estimating the hydraulic functions of unsaturated soils*, in Indirect Meth. for Estimating the Hydraulic Props. of Unsat. Soils, Univ. CA, Riverside, 1989, pp. 359–368.

[3] C. G. Broyden, *A class of methods for solving nonlinear simultaneous equations*, Math. Comp., 19 (1965), pp. 577–593.

[4] M. A. Celia, E. T. Bouloutas, and R. L. Zarba, *A general mass-conservative numerical solution for the unsaturated flow equation*, Water Resour. Res., 26 (1990), pp. 1483–1496.

[5] C. Fassino and G. Manzini, *Fast-secant algorithms for the non-linear richards equation*, Comm. in Num. Methods in Eng., 14 (1998), pp. 921–930.

[6] E. Isaacson and H. B. Keller, *Analysis of Numerical Methods*, Dover, 1994.

[7] C. T. Kelley, *Iterative Methods for Linear and Nonlinear Equations*, vol. 16 of Frontiers in Applied Mathematics, SIAM, Philadelphia, 1995.

[8] F. Lehmann and P. Ackerer, *Comparison of iterative methods for improved solutions of the fluid equation in partially saturated porous media*, Transp. in Porous Media, 31 (1998), pp. 275–292.

[9] C. Paniconi and M. Putti, *A comparison of Picard and Newton iteration in the numerical solution of multidimensional variably saturated flow problems*, Water Resour. Res., 30 (1994), pp. 3357–3374.

[10] L. A. Richards, *Capillary conduction of liquids through porous mediums*, Physics, 1 (1931), pp. 318–333.

[11] M. T. van Genuchten, *A closed form equation for predicting the hydraulic conductivity of unsaturated soils*, Soil Sci. Soc. Am. J., 44 (1980), pp. 892–898.

[12] H. F. Walker and P. Ni, *Anderson acceleration for fixed-point iterations*, Tech. Report MS-9-21-45, WPI Math. Sci. Dept, Sept. 2009. Rev. Jan., 2010. Submitted.

[13] I. Yavneh and G. Dardyk, *A multilevel nonlinear method*, SIAM J. Sci. Comput., 28 (2006), pp. 24–46.