Quasi-Newton Acceleration of ILU Preconditioners for Nonlinear Two-Phase Flow Equations in Porous Media

L. Bergamaschi[†], R. Bru,[*‡]A. Martínez[§], and M. Putti[¶]

January 30, 2011

Abstract

In this work, preconditioners for the iterative solution by Krylov methods of the linear systems arising at each Newton iteration are studied. The preconditioner is defined by means of a Broyden-type rank-one update of a given initial preconditioner, at each nonlinear iteration, as described in [5] where convergence properties of the scheme are theoretically proved. This acceleration is employed in the solution of the nonlinear system of algebraic equations arising from the Finite Element discretization of two-phase flow model in porous media. We report numerical results of the application of this approach when the initial preconditioner is chosen to be the incomplete LU decomposition of the Jacobian matrix at the initial nonlinear stage. It is shown that the proposed acceleration reduces the number of linear iterations needed to achieve convergence. Also, the cost of computing the preconditioner is reduced as this operation is made only once at the beginning of the Newton iteration.

Keywords: Quasi-Newton method, Krylov iterations, updating preconditioners, inexact Newton method, two phase flow, porous media.

[*]Corresponding author

[†]Department of Mathematical Methods and Models for Scientific Applications, University of Padova, Italy, (berga@dmsa.unipd.it).

[§]Department of Mathematical Methods and Models for Scientific Applications, University of Padova, Italy, (acalomar@dmsa.unipd.it).

[¶]Department of Mathematical Methods and Models for Scientific Applications, University of Padova, Italy, (putti@dmsa.unipd.it).

[‡]Instituto de Matemática Multidisciplinar, Departamento de Matemática Aplicada, Universidad Politecnica de Valencia, Spain, (rbru@mat.upv.es).

1 Introduction

Newton's method requires the solution of a number of linear systems with the Jacobian J as the coefficient matrix. When J is large and sparse, e.g. for problems arising from the discretization of a nonlinear PDE, preconditioned Krylov based iterative schemes can be employed for the solution of the linear system. As a result, two nested iterative procedures need to be implemented. To avoid oversolving, i.e. excessive and essentially useless iterations of the inner scheme, it is crucial to employ an "inexact" technique [10]. This approach tries to control the number of linear iterations by allowing the accuracy of the linear solver to vary across nonlinear iterations [12].

There are many papers in the literature, trying to combine the properties of the two nested iterative procedures. Among these we quote [8, 21] where multilevel preconditioners are proposed for accelerating the solution of the Jacobian linear system. In [20] the authors propose to exploit the underlying Krylov subspace information from the linear solver in order to accelerate the nonlinear convergence of Newton-Krylov methods. In [9] the authors try to accelerate the Newton convergence by making use of the information gathered from the Krylov subspace generated by the GMRES linear solver. Differently from their approach, we study how preconditioning efficiency can be enhanced as the nonlinear iteration progresses. In fact, another crucial issue for the reduction of total linear iterations is to use efficient preconditioning techniques. In general, ILU-type preconditioners [22, 25] can be employed and calculated at every nonlinear iteration. Techniques for selectively evaluating a preconditioner P may be developed to save on the cost of the calculation of the preconditioner. Note that the two phases where efficiency can be mostly improved are the cost of the linear system solution (thus including the number of iterations) and the cost of preconditioner evaluation.

In this paper we are mainly concerned with the efficient preconditioning of the linear system. The "optimal" preconditioner P is aimed at clustering eigenvalues of $PJ(\mathbf{x}_k)$. This can be accomplished for instance by minimizing the constant C of:

$$\|zI - PJ(\mathbf{x}_k)\| \leq C, \qquad z \in \mathcal{R}. \tag{1}$$

This requires that information from the nonlinear iterative scheme be taken into account in the evaluation of P.

The approach proposed in this paper is to solve the inner systems of the Newton method with an iterative Krylov subspace method, starting with ILU(0) [22] computed from the initial Jacobian and to update this preconditioner using a rank one sum. A sequence of preconditioners P_k can thus be defined by imposing the secant condition, as used in the implementation of quasi-Newton methods [11]. We choose to work with the Broyden update as described for instance in [19], and analyze the theoretical properties of the preconditioner and the numerical behavior of the resulting scheme. We are aware that the choice $P_0 \equiv ILU(0)$ is not the only possible alternative. Among the others we mention the class of approximate inverse preconditioners (see [6, 2, 3, 14, 15, 16]) which are particularly suited in a parallel environment. We choose to

work with the incomplete ILU factorization because of its wide-spread use in this type of problems and its simplicity. Our strategy is aimed to be independent of the initial preconditioner choice, i.e. the scope of this paper is to construct a sequence of preconditioners that improves the initial one.

We successfully try our approach on a number of nonlinear problems of large size arising from the Finite Element (FE) discretization of two phase flow equations in porous media. The Broyden acceleration provides in all the test cases a reduction of roughly 15% of the number of linear iterations and 10% of the total CPU time.

The paper is organized as follows. The equations governing the two phase flow model are given in section 2. In section 3, the numerical treatment of the governing equations and a general approach of the obtained nonlinear system are studied. The Broyden-type rank-one update of the initial preconditioner is discussed in section 4, where the algorithm is described. In section 5, we report numerical results in 2D and 3D cases. Finally, some conclusions are given in section 6.

2 Two phase flow model: governing equations

Immiscible two phase flow in porous media in isothermal conditions is described by the mass conservation equation [1, 17]:

$$\frac{\partial(\phi\,\rho_\alpha\,S_\alpha)}{\partial t} = -\nabla\cdot[\rho_\alpha v_\alpha] + q_\alpha \qquad \alpha \in w, n \tag{2}$$

where subscript α refers to wetting (w) and non-wetting (n) phase, respectively (e.g. water and oil or water and gas). For each phase, S_α is the saturation, ρ_α the density, v_α the Darcy velocity, and q_α the mass source/sink rate. Finally, ϕ denotes the porous medium porosity. The phase velocity is given by extending Darcy law:

$$v_\alpha = -\lambda_\alpha\,\underline{\underline{k}}\,(\nabla p_\alpha - \rho_\alpha\,g) \tag{3}$$

where the mobility λ_α is defined as the ratio between relative permeability $k_{r\alpha}$ and dynamic viscosity μ_α, $\underline{\underline{k}}$ is the intrinsic permeability tensor, p_α the α-phase pressure, and g the gravity acceleration vector. Substitution of Equation (3) into the continuity equations (2) yields:

$$\frac{\partial(\phi\,\rho_\alpha\,S_\alpha)}{\partial t} = \nabla\cdot[\rho_\alpha\lambda_\alpha\,\underline{\underline{k}}\,(\nabla p_\alpha - \rho_\alpha\,g)] + q_\alpha \tag{4}$$

The solution of the PDE system (4) requires the following auxiliary relationships:

$$S_w + S_n = 1; \qquad p_c(S_w) = p_n - p_w \tag{5}$$

where p_c, the capillary pressure, is defined as the difference between the non-wetting and the wetting phase pressures. Substituting the auxiliary relationships (5) into the

PDEs (4), and solving for p_w and S_w, the system can be written as:

$$\begin{aligned} \frac{\partial(\phi\,\rho_w\,S_w)}{\partial t} &= \nabla\cdot[\rho_w\lambda_w\,\underline{\underline{k}}\,(\nabla p_w-\rho_w\,g)]+q_w \\ \frac{\partial[\phi\,\rho_n\,(1-S_w)]}{\partial t} &= \nabla\cdot[\rho_n\lambda_n\,\underline{\underline{k}}\,(\nabla p_w+\nabla p_c-\rho_n\,g)]+q_n \end{aligned} \tag{6}$$

Appropriate initial and boundary conditions complete the model formulation.

Capillary properties can be described using a number of constitutive laws, whose most widely used models are Brooks-Corey (BC) [7] and Van Genuchten (VG) [29]. Brooks-Corey capillary laws have the following representation:

$$\begin{aligned} k_{rw}(S_w) &= S_{we}^{(2+3\zeta)/\zeta}; \qquad k_{rn}(S_w)=(1-S_{we})^2\,(1-S_{we}^{(2+\zeta)/\zeta}) \\ p_c(S_w) &= p_d\,S_{we}^{-1/\zeta} \end{aligned}$$

where p_d is the pore entry pressure representing the lowest capillary pressure needed to displace the wetting phase by the non-wetting phase in a fully saturated medium, ζ the so called sorting factor or pore distribution index which is related to the medium pore size distribution. The sorting factor usually ranges between 0.2 (denoting a wide range of pore sizes) and 7 (for very uniform materials), $S_{we}=(S_w-S_{wr})/(1-S_{wr})$ is the effective water saturation, with S_{wr} the irreducible water saturation. Van Genuchten constitutive laws read as:

$$\begin{aligned} k_{rw}(S_w) &= S_{we}^{1/2}\,[1-(1-S_{we}^{1/m})^m]^2; \qquad k_{rn}(S_w)=(1-S_e)^{1/2}\,(1-S_{we}^{1/m})^{2\,m} \\ p_c(S_w) &= p_0\,(S_{we}^{-1/m}-1)^{1-m} \end{aligned}$$

where p_0 is the characteristic capillary pressure of the medium, and m is related to the pore distribution. Equations (6) represent a highly nonlinear system of PDEs and fluid densities and viscosities may also depend on the corresponding phase pressure:

$$\rho_w=\rho_w(p_w); \qquad \rho_n=\rho_n(p_n); \qquad \mu_w=\mu_w(p_w); \qquad \mu_n=\mu_n(p_n)$$

3 Numerical model

3.1 Two-phase flow finite element equations

Equations (6) are discretized in space using linear finite elements (triangles in 2D and tetrahedra in 3D) yielding a system of first order differential equations that reads:

$$\begin{bmatrix} H_w & M_w \\ H_n & M_n \end{bmatrix}\begin{bmatrix} \boldsymbol{p}_w \\ \boldsymbol{S}_w \end{bmatrix}+\begin{bmatrix} 0 & M_w \\ 0 & M_n \end{bmatrix}\begin{bmatrix} \dot{\boldsymbol{p}}_w \\ \dot{\boldsymbol{S}}_w \end{bmatrix}+\begin{bmatrix} \boldsymbol{q}_w \\ \boldsymbol{q}_n \end{bmatrix}=0 \tag{7}$$

where H_w, H_n, M_w and M_n are wetting and non-wetting stiffness and mass matrices; $[\boldsymbol{q}_w,\boldsymbol{q}_w]^T$ incorporate source/sink terms and Neumann boundary conditions;

$[\boldsymbol{p}_w, \boldsymbol{S}_w]^T$ and $\left[\dot{\boldsymbol{p}}_w, \dot{\boldsymbol{S}}_w\right]^T$ are the vectors of the unknown nodal water pressure ($\boldsymbol{p}_w$) and saturation ($\boldsymbol{S}_w$), and the corresponding time derivatives. Mass matrices M_w and M_n are lumped for stability reasons, while in the stiffness matrices H_w and H_n hydraulic mobility is evaluated "fully upwind" [1, 17, 18,] to ensure convergence of the nonlinear scheme to the correct physical solution and to avoid undesirable oscillations when capillary forces become small. Stiffness matrices H_w and H_n are symmetric and positive definite and symmetric and positive semi-definite, respectively. Mass matrices M_w and M_n are diagonal matrices. System (7) can be written in a more compact form as:

$$H\,\boldsymbol{x} + M\,\dot{\boldsymbol{x}} + \boldsymbol{q} = 0 \tag{8}$$

where the meaning of the new symbols is derived by comparison of Equations (7) and (8). The time integration is implemented via Euler backward FD, giving the following nonlinear system of algebraic equations:

$$\left[H + \frac{M}{\Delta t}\right]^{(m+1)} \boldsymbol{x}^{(m+1)} = \left[\frac{M}{\Delta t}\right]^{(m+1)} \boldsymbol{x}^{(m)} - \boldsymbol{q}^{(m+1)} \tag{9}$$

where Δt is the time step size; (m) and $(m+1)$ indicate the previous and the current time level, respectively.

3.2 General approach for nonlinear systems

The nonlinear system (9), which has the same form for both two phase and unsaturated flow, is solved by Newton-like iterative methods. To this aim, equation (9) is rewritten as:

$$f(\boldsymbol{x}^{(m+1)}) = A\,\boldsymbol{x}^{(m+1)} - \bar{\boldsymbol{q}} = 0 \tag{10}$$

with:

$$A = \left[H + \frac{M}{\Delta t}\right]^{(m+1)}; \qquad \bar{\boldsymbol{q}} = \left[\frac{M}{\Delta t}\right]^{(m+1)} \boldsymbol{x}^{(m)} - \boldsymbol{q}^{(m+1)}$$

Let $\boldsymbol{s}_k = \boldsymbol{x}_{k+1} - \boldsymbol{x}_k$ be the current search direction, the Newton iteration is given by:

$$J(\boldsymbol{x}_k)\,\boldsymbol{s}_k = -f(\boldsymbol{x}_k); \qquad \boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \boldsymbol{s}_k \tag{11}$$

where the Jacobian matrix is given by $J = A + \dfrac{\partial A}{\partial \boldsymbol{x}}\,\boldsymbol{x} - \dfrac{\partial \bar{\boldsymbol{q}}}{\partial \boldsymbol{x}}$ and $\boldsymbol{s}_k$ is the current search direction. Nonlinear convergence is considered achieved when

$$\|J(\boldsymbol{x}_k)\boldsymbol{s}_k + F(\boldsymbol{x}_k)\| \leq \eta_k \|F(\boldsymbol{x}_k)\|. \tag{12}$$

where $\{\eta_k\}$ is a nonincreasing sequence. This is the stopping criterion used by Dembo, Eisenstat and Steihaug in [10]. Several variants of this criterion can be found in [12] and [13]. All of them try to choose η_k in order to avoid the so called oversolving of the linearized systems.

The iterative or direct solution of system (11) represents the main computational burden of the whole discretization procedure. In the case of two phase flow, Newton method leads to a non-symmetric linear system which cannot in general be solved by direct methods. For this reason, we choose a preconditioned Bi-CGSTAB approach [28]. Standard ILU factorization as the preconditioner, produces an acceleration of the Bi-CGSTAB method, but it does not take into account the nonlinear properties of the outer scheme and can thus be affected by poor convergence especially in presence of an ill-conditioned Jacobian (see [5]). The rank-one modification of the ILU(0) preconditioner proposed in this paper, which we will describe in the next section, is aimed at further accelerating the iterative solution of (11) especially when the Newton iteration, approaching the solution of (10), displays quadratic convergence.

4 Preconditioner acceleration by Broyden update

The nonlinear algebraic problem (10) can be rewritten as:

$$F(x) = 0 \tag{13}$$

In Newton's method (the outer iteration), a number of linear systems of the form (11) are solved by an iterative Krylov subspace method (inner iteration). Starting from an initial preconditioner calculated from the first Jacobian matrix, i.e., at nonlinear step 0 ($J(\boldsymbol{x}_0)$), acceleration is obtained by building a sequence of preconditioners that will converge towards an "optimal" preconditioner.

Based on the Quasi-Newton scheme we construct as proposed in [5] the sequence of matrices:

$$B_{k+1} = B_k + \frac{(\boldsymbol{y}_k - B_k \boldsymbol{s}_k)\, \boldsymbol{s}_k^T}{\boldsymbol{s}_k^T \boldsymbol{s}_k}. \tag{14}$$

where the secant condition has been used. It can be easily proved that B_{k+1} is the closest matrix B satisfying the secant condition to B_k, the distance being measured in the Frobenius norm.

Thus, given an arbitrary initial preconditioner $P_0 = B_0^{-1}$, the sequence of preconditioners $\{P_k = B_k^{-1}\}$ are defined as

$$\begin{aligned} B_1^{-1} &= B_0^{-1} \\ B_{k+1}^{-1} &= B_k^{-1} - \frac{\left(B_k^{-1}\boldsymbol{y}_k - \boldsymbol{s}_k\right) \boldsymbol{s}_k^T B_k^{-1}}{\boldsymbol{s}_k^T B_k^{-1} \boldsymbol{y}_k}, \quad k = 1, 2, \ldots \end{aligned} \tag{15}$$

where each updated preconditioner is function of the previous preconditioner and the vectors $\boldsymbol{y}_k$ and $\boldsymbol{s}_k$ of the previous step in the Newton scheme. This update corresponds to a one rank correction.

4.1 Convergence properties of the sequence of preconditioners

We take into account the standard assumptions made in the analysis of convergence of the Newton method, i.e.:

1. Equation (13) has a solution $\boldsymbol{x}^*$.
2. $J(x) : \Omega \to \mathbb{R}^{n\times n}$ is Lipschitz continuous with Lipschitz constant γ.
3. $J(\boldsymbol{x}^*)$ is non-singular.

In [5] some convergence properties of the above sequence (15) are proved. In particular, the norm of all matrices of the sequence are bounded, as it is established in the following result.

Theorem 1 *Let the standard assumptions hold. Define* $\alpha = \|J(\boldsymbol{x}^*)^{-1}\|$. *For a fixed* $0 < \delta_1 < \dfrac{1}{\alpha}$ *there exist* δ *and* δ_B *such that if* $\|\boldsymbol{x}^* - \boldsymbol{x}_0\| < \delta$ *and* $\|B_0 - J(\boldsymbol{x}^*)\| < \delta_B$ *then*

$$\|B_k^{-1}\| < \frac{\alpha}{1-\delta_1\alpha}. \tag{16}$$

In addition, the constructed sequence of preconditioners (15) based on the Broyden secant update equation satisfies:

$$||I - B_k^{-1}J(\boldsymbol{x}_k)|| \leq K$$

where the constant K is fixed. In fact the following result can be proved:

Theorem 2 *Let the standard assumptions hold. Fixed* $0 < \delta_1 < \dfrac{1}{\alpha}, \delta_1' > 0$, *then there are* $\delta, \delta_B, \delta_B'$ *such that if* $\|\boldsymbol{x}^* - \boldsymbol{x}_0\| < \delta$, $\|B_0 - J(\boldsymbol{x}^*)\| < \delta_B$ *and* $\|B_0 - J(\boldsymbol{x}_0)\| < \delta_B'$ *then*

$$\|I - B_k^{-1}J_k\| < \frac{\delta_1'\alpha}{1-\delta_1\alpha}.$$

The above theorem states the important property that the sequence of the preconditioned matrices can be made as near as desired to the identity matrix depending on the initial Newton point $\boldsymbol{x}_0$ and the initial preconditioner B_0.

4.2 Implementation

The main "kernel" operation of our implementation of the preconditioner in the Krylov method is the computation of the matrix-vector product $B_k^{-1}\boldsymbol{z}_k^l$ in the k nonlinear iterations and in the l inner linear steps. Since B_k^{-1} can be written in terms of the initial preconditioner B_0^{-1}, the computation of $B_k^{-1}\boldsymbol{z}_k^l$ needs $3k-1$ dot products and $3k-1$ daxpy operations in addition to the computation of $B_0^{-1}\boldsymbol{z}_k^l$ (see [5] for details). At every nonlinear iteration, one has to compute and save $2k$ different vectors with a corresponding increase in CPU time and, more importantly, large memory occupation. To alleviate this problem we introduce the parameter k_{max} representing the maximum number of rank-one corrections allowed. Then, after k_{max} nonlinear iterations all computed vectors are discarded and a new initial preconditioner is computed. This idea

is implemented in our algorithm within the variant called restarted Newton-Broyden (RBN) algorithm.

RESTARTED NEWTON-BROYDEN (RNB) ALGORITHM
Input: $\boldsymbol{x}_0, F, k_{\max}, nlmax$, tol

- Compute $B_0(B_0^{-1})$ approximating $J_0(J_0^{-1})$; $k = 0$, restart=TRUE
- WHILE $\|F(\boldsymbol{x}_k)\| >$ tol AND $k < nlmax$ DO
  1. IF (NOT restart) THEN update B_k^{-1} from B_{k-1}^{-1}; restart = FALSE.
  2. Solve $J(\boldsymbol{x}_k)\boldsymbol{s}_k = -F(\boldsymbol{x}_k)$ by a Krylov method with preconditioner B_k^{-1} and tolerance η_k.
  3. $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \boldsymbol{s}_k$
  4. $k = k + 1$
  5. IF k MOD $k_{\max}$ = 0 THEN
     - restart = TRUE; compute $B_k(B_k^{-1})$ approximating $J_k(J_k^{-1})$
- END WHILE

The advantage of the restarted algorithm is twofold: on one hand it considerably reduces the cost of a single iteration which is proportional to the number of updates. Secondly, it allows to take into account the power of Theorem 2. After a restart, and if the outer Newton iteration is converging, the quantities $\|\boldsymbol{e}_0\| = \|\boldsymbol{x}^* - \boldsymbol{x}_0\|$ and $\|B_0 - J(\boldsymbol{x}^*)\|$ will be reduced with a consequent improvement of the preconditioner quality due to the reduction of $\|I - B_k^{-1}J_k\|$.

5 Numerical tests and discussion

5.1 Description of the test problems

We verify the performance of the proposed algorithm on a 2-dimensional example taken from [18] and its extension to 3D. In a heterogeneous sandy box of dimensions 0.9×0.9 m wide and 0.65 m high, initially saturated with water, a Trichloroethylene (TCE) infiltration takes place over a 0.1×0.1 m area located at the center of the upper surface (see Figure 1). A permanent TCE saturation of 0.25 is maintained for 20000 s, which represents the simulated time. The remaining part of the upper and the bottom boundary are impermeable, whereas along the lateral sides hydrostatic water pressure is assumed. For symmetry reasons only a quarter of the sandy sample is discretized. The 2D mesh is made up of 1961 nodes and 3744 triangles whereas the 3D mesh

		sand #1	sand #2
intrinsic permeability k	[m^2]	$6.64 \cdot 10^{-11}$	$7.15 \cdot 10^{-12}$
porosity ϕ	[-]	0.40	0.39
(BC) sorting factor ζ	[-]	2.7	2.0
(BC) pore entry pressure p_d	[Pa]	755	2060
irreducible water saturation S_{wr}	[-]	0.09	0.12
Fluid properties		water	TCE
density ρ	[kg/m^3]	1000	1462
viscosity μ	[Pa s]	0.001	0.00057

Table 1: Soil and fluid properties for the test cases.

consists of 72557 nodes and 404352 tetrahedra. The nonlinear problems therefore have $n = 3922$ and $n = 145114$ degrees of freedom, respectively. Soil and fluid properties are summarized in Table 1. The simulations were carried on up to $t = 200$ seconds for the 2d case and $t = 300$ seconds for the 3d test.

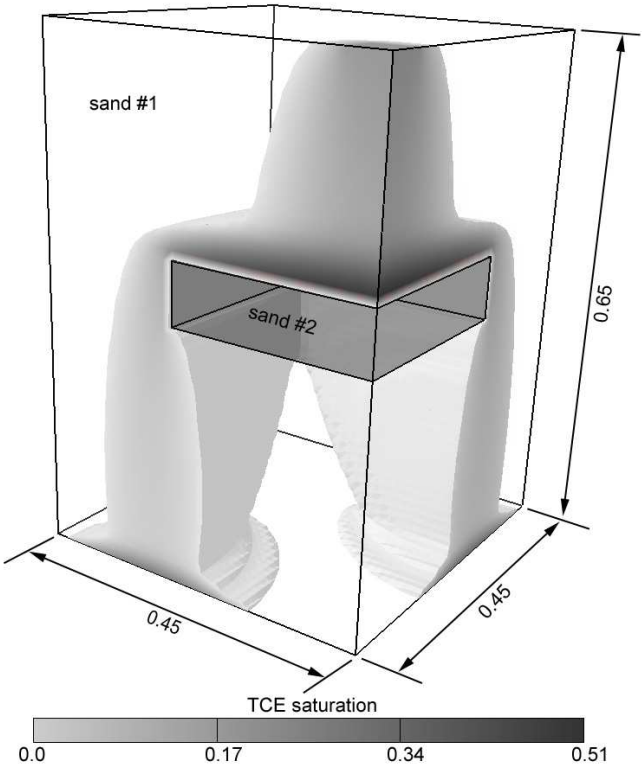

Figure 1: Three-dimensional test case configuration and saturation profile after 20000 s of TCE infiltration.

5.2 Numerical Results

For our numerical results we have used as initial preconditioners (B_0) the incomplete LU factorization with no fill-in ILU(0) [22]. In this case the application of B_0^{-1} results in two triangular sparse linear system solutions. The value of the stopping criterion parameter $\eta_k \equiv \eta$ (equation (12)) was set to 10^{-4}. The numerical experiments were performed on a 32-bit PC-workstation equipped with a 1526 MHz AMD processor, 2000 Mbyte of core memory, and 256 Kbyte of secondary cache. The CPU times are measured in seconds.

	kmax	timesteps	NL it.	linear it.	CPU [s]	solver [s]
ILU(0) fixed		42	194	10269	17.34	10.66
ILU(0) (J_k)		42	196	8436	15.65	9.02
RNB-ILU(0)	2	42	196	6732	14.12	7.45
RNB-ILU(0)	3	42	196	7368	14.65	8.01
RNB-ILU(0)	4	42	196	7692	15.06	8.42
RNB-ILU(0)	5	42	196	7716	15.15	8.42

Table 2: Timing results of the 2D case.

	kmax	timesteps	NL it.	linear it.	CPU [s]	solver [s]
ILU(0) fixed		27	107	5470	1611.63	670.21
ILU(0) (J_k)		27	107	5264	1608.86	667.76
RNB-ILU(0)	2	27	107	4501	1504.36	563.29
RNB-ILU(0)	3	27	107	4453	1512.86	571.76
RNB-ILU(0)	4	27	107	4316	1513.67	572.53

Table 3: Timing results of the 3D case.

In Tables 2 and 3 we report the results of the 2D and 3D simulations, respectively, with different preconditioners resulting from different choices of the $kmax$ value. We display the number of time steps, the total number of nonlinear iterations (NL it.), the total number of linear iterations, the overall CPU time and the CPU time needed by the linear solver. In the first row we keep the ILU(0) preconditioner fixed during the nonlinear iteration while in the second row we report the results obtained by computing the incomplete LU factorization of each Jacobian matrix J_k. It is shown that the Broyden rank-one update with restart (RBN-ILU(0)) produces an improvement with respect to the ILU preconditioner in both number of linear iterations and CPU total time. We note that the optimal values of $kmax$ parameters are the smallest ones, suggesting that it is more efficient to compute the preconditioner of choice (ILU(0) in our tests) every 2/3 iterations and then enriching it with a small number of rank-one updates. This will ensure acceleration of the iterative procedure due to the decreased number of iterations and with a negligible additional cost per iteration. The optimal

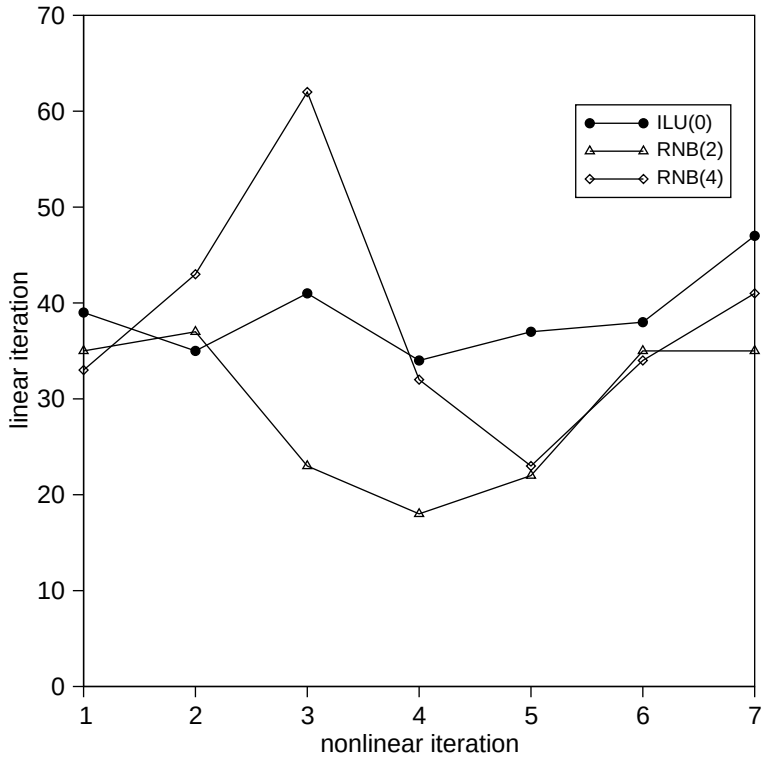

Figure 2: 2D test case. Number of linear iteration vs. nonlinear iteration for a fixed time step.

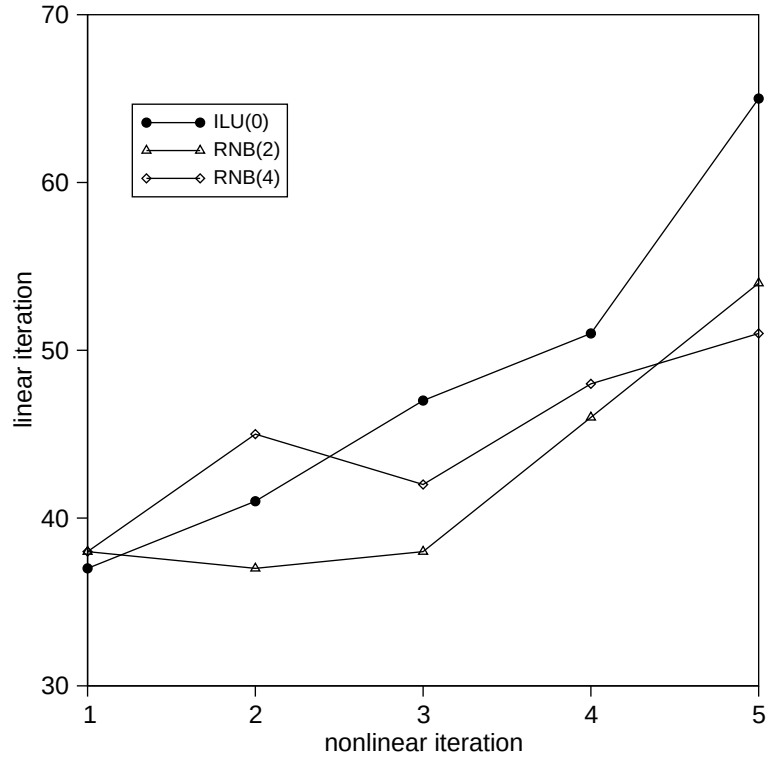

Figure 3: 3D test case. Number of linear iteration vs. nonlinear iteration for a fixed timestep.

value of the restart parameter is found to be $kmax = 2$, which consists in computing the preconditioner of choice every two nonlinear iteration and correct it with only 1 rank-one update.

Figures 2 and 3 show the total number of linear iterations versus the nonlinear iteration index for a fixed time step which has been chosen near the end of the simulation, for the 2D and the 3D problems, respectively. From the figures it is clear that the proposed acceleration is particularly convenient with respect to the ILU(0) preconditioner for the last nonlinear iterations. This is in agreement with the theoretical findings which assure that the distance of the preconditioned matrix from identity is smaller if the starting Newton point is closer to the solution.

5.3 Keeping the Jacobian fixed throughout the Newton iteration

We report in table 4 the results obtained by computing the Jacobian matrix only at the first two Newton iterations, keeping J_2 constant until convergence. We note that the number of time steps is higher than using the full Newton approach since the loss of quadratic convergence forces the reduction of time steps a number of times during the simulation. The number of nonlinear iterations is, as expected, much higher again because of slower convergence. The CPU time due to matrix assembly is reduced but this does not counterbalance the increased solver CPU time. It is interesting to note that also in this case the Restarted Newton Broyden variant produces an important reduction of the overall linear iterations.

	kmax	timesteps	NL it.	linear it.	CPU [s]	solver [s]
ILU(0)		84	514	20988	29.12	22.51
RNB-ILU(0)	2	84	513	15965	24.12	17.88

Table 4: Timing results of the 2D case with constant Jacobian $J_k \equiv J_2$, $k \geq 2$.

6 Conclusion and future developments

A family of preconditioners based on the Broyden secant update formula is proposed to accelerate the convergence properties of a given preconditioner during the linear solver phase of a nonlinear iterative procedure. During Newton iteration, starting from a preconditioner B_0^{-1} approximating the inverse of the initial Jacobian matrix, a sequence of preconditioners B_k^{-1} is defined, by taking into account information of the previous nonlinear iterations. The developed theoretical analysis, proves that the sequence satisfies $||I - B_k^{-1} J_k(\boldsymbol{x}_k)|| \leq C$, with C that can be made arbitrarily small, depending on $\boldsymbol{x}_0$ and B_0.

The performance of the algorithm described in this paper has been tested onto 2D and 3D problems arising from FE discretization of two phase flow in porous media. The numerical results show an improvement, both in terms of linear iteration number and CPU time, with respect to the classical ILU(0) preconditioner. We stress that our theoretical developments make our procedure free from the initial preconditioner choice. If a better performing initial P_0 than ILU(0) is devised, than the rank one update formula will improve it, may be even better than it is shown in this paper.

Based on the recent work in [4] where a positive definite sequence of preconditioners is proposed by means of a rank-two BFGS update, we are planning to accelerate also the Picard iteration to solve the nonlinear two-phase flow as well as the Richard's equation in porous media. Starting from a positive definite approximation of the Jacobian, it is possible to define a symmetric positive definite sequence of preconditioners which allows the solution of the linearized system by the preconditioned conjugate gradient method.

Acknowledgments

This work was partially supported by the Italian MIUR project "Numerical Models for Multiphase Flow and Deformation in Porous Media and by Spanish DGI grant MTM2007–64477.

References

[1] K. Aziz and A. Settari, "Petroleum Reservoir Simulation", Applied Science Publishers, London, 1979.

[2] M. Benzi, Preconditioning techniques for large linear systems: a survey, J. Comput. Phys., 182 (2) (2002) 418–477.

[3] M. Benzi, J. K. Cullum, and M. Tůma, "Robust approximate inverse preconditioning for the conjugate gradient method", SIAM J. Sci. Comput., 22 (2000), pp. 1318–1332.

[4] L. Bergamaschi, R. Bru, and Á. Martinez. Low-rank update of preconditioners for the inexact Newton method with SPD Jacobian. 2009. submitted.

[5] L. Bergamaschi, R. Bru, A. Martínez and M. Putti, "Quasi-Newton preconditioners for the inexact Newton method", Electronic Transactions of Numerical Analysis, 23, 76–87, 2006.

[6] L. Bergamaschi and A. Martínez, "Parallel acceleration of Krylov solvers by factorized approximate inverse preconditioners", in VECPAR 2004, M. Daydè et al., ed., vol. 3402 of Lecture Notes in Computer Sciences, Heidelberg, 2005, Springer-Verlag, pp. 623–636.

[7] R. H. Brooks and A. T. Corey, "Hydraulic properties of porous media", Hydrology Paper 3, Colorado State University, Fort Collins, CO, 1964.

[8] H. Class, R. Helmig, and P. Bastian. Numerical simulation of non-isothermal multiphase multicomponent processes in porous media. 1. An efficient solution technique. *Advances In Water Resources*, 25(5):533–550, 2002.

[9] C. N. Dawson, H. Klie, M. F. Wheeler, and C. Woodward, A parallel, implicit, cell-centered method for two-phase flow with a preconditioned Newton-Krylov solver, Comput. Geos., pp. 215–249, 1997.

[10] R. S. Dembo, S. C. Eisenstat, and T. Steihaug, "Inexact Newton methods", SIAM J. Num. Anal., 19, 400–408, 1982.

[11] J. E. Dennis, Jr. and J. J. Moré, "Quasi-Newton methods, motivation and theory", SIAM Rev., 19, 46–89, 1977.

[12] S. C. Eisenstat and H. F. Walker, "Choosing the forcing terms in an Inexact Newton method", SIAM J. Sci. Comput., 17, 16–32, 1996.

[13] M. A. Gomes-Ruggiero, V. L. Lopes and J. V. Toledo-Benavides, "A globally convergent inexact Newton method with a new choice for the forcing term", Annals of Operations Research, 157(1), 193–205, 2008. Comput. Phys. 182 (2) (2002) 418–477.

[14] G. A. Gravvanis, Generalized approximate inverse finite element matrix techniques, Neural Parallel Sci. Comput. 7 (4) (1999) 487–500.

[15] G. A. Gravvanis, K. M. Giannoutakis, Parallel approximate finite element inverse preconditioning on distributed systems., in: ISPDC/HeteroPar, 2004, pp. 277–283.

[16] M. J. Grote, T. Huckle, Parallel preconditioning with sparse approximate inverses, SIAM J. Sci. Comput. 18 (3) (1997) 838–853.

[17] R. Helmig, "Multiphase flow and transport processes in the subsurface. A contribution to the modeling of hydrosystem", Springer-Verlag, Berlin, 1997. 367 pp.

[18] R. Huber and R. Helmig, "Node-centered finite volume discretizations for the numerical simulation of multiphase flow in heterogeneous porous media", Computat. Geosci., 4 (2000), pp. 141–164.

[19] C. T. Kelley, "Iterative Methods for Linear and Nonlinear Equations", SIAM, Philadelphia, 1995.

[20] H. Klie and M. F. Wheeler, Krylov-Secant Methods for Accelerating the Solution of Fully Implicit Formulations, SPE Reservoir simulation symposium, The Woodlands, TX, 2005.

[21] S. Lacroix, Y. Vassilevski, J. Wheeler, and M. Wheeler. Iterative solution methods for modeling multiphase flow in porous media fully implicitly. *SIAM Journal On Scientific Computing*, 25(3):905–926, 2003.

[22] J. A. Meijerink and H. A. van der Vorst, "An iterative solution method for linear systems of which the coefficient matrix is a symmetric M-matrix", Math. Comp., 31, 148–162, 1977.

[23] J. L. Morales and J. Nocedal, "Automatic preconditioning by limited memory Quasi-Newton updating", SIAM J. on Optimization, 10, 1079–1096, 2000.

[24] D. W. Peaceman, "Fundamentals of Numerical Reservoir Simulation", Elsevier, Amsterdam, 1977.

[25] Y. Saad, “ILUT: A dual threshold incomplete LU factorization”, Numer. Linear Alg. Appl., 1, 387–402, 1994.

[26] Y. Saad and M. H. Schultz, “GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems”, SIAM J. Sci. Stat. Comput., 7, 856–869, 1986.

[27] J. D. Tebbens and M. Tůma, “Preconditioner updates for solving sequences of large and sparse nonsymmetric linear systems”, SIAM Journal on Scientific Computing, 29, 1918-1941, 2007.

[28] H. A. van der Vorst, “Bi-CGSTAB: A fast and smoothly converging variant of BI-CG for the solution of nonsymmetric linear systems”, SIAM J. Sci. Stat. Comput., 13, 631–644, 1992.

[29] M. T. van Genuchten, “A closed-form equation for predicting the hydraulic conductivity of unsaturated soils”, Soil Science Society of America Journal, 44 (1980), pp. 892–898.