JOURNAL OF COMPUTATIONAL PHYSICS 62, 265–281 (1986)

Practical Considerations for Adaptive Implicit Methods in Reservoir Simulation

P. A. FORSYTH, JR.* AND P. H. SAMMON*

Computer Modelling Group, 3512 33 Street N.W., Calgary, Alberta, T2L 2A6, Canada

Received July 23, 1984; revised January 2, 1985

An adaptive implicit method utilizing different degrees of implicitness on a cell by cell basis is discussed. Techniques for solving the Jacobian of the system using an incomplete *LU* factorization are presented, and criteria for switching cells from explicit to implicit are investigated. Results for several sample problems show that substantial savings in execution time can be obtained with this technique.

1. INTRODUCTION

It is well known that explicit time differencing methods for flow problems result in algorithms which are computationally inexpensive per time step. However, the maximum stable time step may be extremely small. On the other hand, implicit methods allow large time steps but are much more computationally expensive per time step. In the following, we will describe some practical considerations arising in the application of the adaptive implicit method [1] to reservoir simulation. The adaptive implicit method attempts to minimize computation time by using an implicit method for a small fraction of the total number of finite difference cells, while using an explicit method for the remaining cells. In response to changing flow conditions, the degree of implicitness can change from time step to time step.

The basic equations used for black oil reservoir simulation consist of conservation equations for oil, gas and water. The phase velocity is given by Darcy's law [2]. The black oil approximation assumes that the water phase consists entirely of water, that the oil is non-volatile, and that gas can be present in both gas and oil phases [2]. This type of model is commonly used for primary depletion and secondary recovery (waterflooding). With the above assumptions, the following equations result:

conservation of oil:

$$\frac{\partial}{\partial t}\left(\frac{\phi S_0}{B_0}\right) + q_0 - \nabla \cdot \left[\frac{\lambda_0 k}{B_0}(\nabla p_0 - \rho_0 \bar{g})\right] = 0; \tag{1}$$

* Now with Dynamic Reservoir Systems.

0021-9991/86 $3.00

conservation of gas:

$$\frac{\partial}{\partial t}\left[\phi\left(\frac{R_s S_0}{B_0}+E_g S_g\right)\right]+R_s q_0+q_g$$
$$-\nabla\left[\frac{R_s\lambda_0}{B_0}k(\nabla p_0-\rho_0\bar{g})+E_g\lambda_g k(\nabla p_g-\rho_g\bar{g})\right]=0; \tag{2}$$

conservation of water:

$$\frac{\partial}{\partial t}\left(\frac{\phi S_w}{B_w}\right)+q_w-\nabla\cdot\left[\frac{\lambda_w k}{B_w}(\nabla p_w-\rho_w\bar{g})\right]=0; \tag{3}$$

where

$$\lambda_l = \text{phase mobility}$$
$$=\frac{k_{rl}}{\mu_l}, \qquad l=\text{oil, gas, water}$$

and where

B_l = formation volume factor for l = oil, water
E_g = gas expansion factor
$\bar{g}$ = gravity vector
k = absolute permeability
k_{rl} = relative permeability for l = oil, gas, water
p_l = pressure in phase l = oil, gas, water
q_l = well term for phase l = oil, gas, water
R_s = solution gas–oil ratio
S_l = saturation of phase l = oil, gas, water
t = time
ρ_l = density of phase l = oil, gas, water
ϕ = porosity.

Note that

$$B_o=\frac{[V_o+V_{dg}]_{RC}}{[V_o]_{STC}}$$
$$B_w=\frac{[V_w]_{RC}}{[V_w]_{STC}} \tag{4}$$
$$E_g=\frac{[V_g]_{STC}}{[V_g]_{RC}},$$

where $[V_l]_{RC}$ is the volume of a given mass of component l at reservoir conditions, and $[V_l]_{STC}$ is the volume occupied by the same mass at standard conditions. V_{dg} is the volume of gas dissolved in the oil. The solution gas:oil ratio is defined by

$$R_s = \left[\frac{V_{dg}}{V_o}\right]_{STC} \tag{5}$$

which is tabulated as an experimentally determined function of the oil phase pressure. The densities of the various phases are expressed at reservoir conditions as follows:

$$\begin{aligned} \rho_o &= ((\rho_o)_{STC} + R_s(\rho_g)_{STC})/B_o \\ \rho_w &= (\rho_w)_{STC}/B_w \\ \rho_g &= E_g(\rho_g)_{STC}. \end{aligned} \tag{6}$$

The porosity ϕ refers to the fraction of the volume of reservoir rock which is void space, and the saturation S_l denotes the fraction of the void space occupied by phase l. The relative permeability of phase l, k_{rl}, is an experimentally determined function of the saturations. A more detailed explanation of flow in porous media presented from a petroleum engineering standpoint can be found in Ref. [2].

The gas saturation is related to the water and oil saturations by

$$S_g = 1 - S_o - S_w. \tag{7}$$

The individual phase pressures are related through the capillary pressures P_{cl}:

$$\begin{aligned} p_w &= p_o - P_{cw} \\ p_g &= p_o + P_{cg}. \end{aligned}$$

These equations are discretized into finite difference form using a control volume approach [3], with central derivatives in space and upstream mobilities. The discretized equations can be written as follows:

oil equation:

$$\begin{aligned} &\frac{V_i}{\Delta t}\left[\left(\frac{\phi S_o}{B_o}\right)_i^{N+1} - \left(\frac{\phi S_o}{B_o}\right)_i^{N}\right] + q_{oi}^{N+1} - \frac{V_i}{\Delta x_i} \\ &\quad \times [T_{oi+1/2}^{M} k_{i+1/2} \psi_{oi+1/2} - T_{oi-1/2}^{M} k_{i-1/2} \psi_{oi-1/2}] \\ &\quad + (y \text{ and } z \text{ flow terms}) = 0; \end{aligned} \tag{8}$$

gas equation:

$$\frac{V_i}{\Delta t}\left[(\phi E_g S_g)_i^{N+1} - (\phi E_g S_g)_i^N + \left(\frac{\phi R_s S_o}{B_o}\right)_i^{N+1} - \left(\frac{\phi R_s S_o}{B_o}\right)_i^N\right]$$
$$+ q_{gi}^{N+1} + (R_s q_o)_i^{N+1}$$
$$- \frac{V_i}{\Delta x_i}\left[T_{g\,i+1/2}^M k_{i+1/2}\psi_{g\,i+1/2} - T_{g\,i-1/2}^M k_{i-1/2}\psi_{g\,i-1/2}\right]$$
$$- \frac{V_i}{\Delta t}\left[(R_s T_o)_{i+1/2}^M k_{i+1/2}\psi_{o\,i+1/2} - (R_s T_o)_{i-1/2}^M k_{i-1/2}\psi_{w\,i-1/2}\right]$$
$$+ (y \text{ and } z \text{ flow terms}) = 0; \tag{9}$$

water equation:

$$\frac{V_i}{\Delta t}\left[\left(\frac{\phi S_w}{B_w}\right)_i^{N+1} - \left(\frac{\phi S_w}{B_w}\right)_i^N\right] + q_{wi}^{N+1}$$
$$- \frac{V_i}{\Delta x_i}\left[T_{w\,i+1/2}^M k_{i+1/2}\psi_{w\,i+1/2} - T_{w\,i-1/2}^M k_{i-1/2}\psi_{w\,i-1/2}\right]$$
$$+ (y \text{ and } z \text{ flow terms}) = 0. \tag{10}$$

In the above, the subscript i refers to the ith grid block or cell, and the transmissibilities T_l, l = oil, gas, water, are defined as follows:

$$T_o = k_{ro}/(B_o \mu_o), \qquad T_g = E_g k_{rg}/\mu_g, \qquad T_w = k_{rw}/(B_w \mu_w),$$

where μ_l denotes the viscosity of phase l, $T_{l\,i+1/2}$ represents either T_{li} or $T_{l\,i+1}$ depending on which is the upstream point for the lth phase, and a similar definition holds for $(R_s T_o)_{i+1/2}$ based on the oil phase. V_i is the volume of the ith finite difference cell, and Δx_i is the cell width in the x-direction. The absolute permeability is defined using a harmonic mean [2]:

$$k_{i+1/2} = \frac{(\Delta x_i + \Delta x_{i+1})\, k_i k_{i+1}}{\Delta x_{i+1} k_i + \Delta x_i k_{i+1}}. \tag{11}$$

The superscripts N or $N+1$ refer to the time level. M can be either N or $N+1$ depending on whether cell i uses an IMPES formulation (to be defined later) or a fully implicit formulation.

The potential terms ψ are given by

$$\psi_{o\,i+1/2}$$
$$= \frac{p_{o\,i+1}^{N+1} - p_{o\,i}^{N+1}}{(\Delta x_i + \Delta x_{i+1})/2}$$
$$- \frac{\Delta x_i \rho_{o\,i+1}^M g_{x\,i+1} + \Delta x_{i+1} \rho_{o\,i}^M g_{x\,i}}{\Delta x_{i+1} + \Delta x_i}$$

$$\psi_{g\,i+1/2} = \frac{p^{N+1}_{o\,i+1} - p^{N+1}_{oi}}{(\Delta x_{i+1} + \Delta x_i)/2} + \frac{P^{M}_{cg\,i+1} - P^{M}_{cgi}}{(\Delta x_{i+1} + \Delta x_i)/2} - \frac{\Delta x_i \rho^{M}_{g\,i+1} g_{x\,i+1} + \Delta x_{i+1} \rho^{M}_{gi} g_{xi}}{\Delta x_{i+1} + \Delta x_i}$$

$$\psi_{w\,i+1/2} = \frac{p^{N+1}_{o\,i+1} - p^{N+1}_{oi}}{(\Delta x_{i+1} + \Delta x_i)/2} - \frac{P^{M}_{cw\,i+1} - P^{M}_{cw\,i}}{(\Delta x_{i+1} + \Delta x_i)/2} - \frac{\Delta x_i \rho^{M}_{w\,i+1} g_{x\,i+1} + \Delta x_{i+1} \rho^{M}_{wi} g_{xi}}{\Delta x_{i+1} + \Delta x_i}. \tag{12}$$

Upstream directions for each phase l are determined by the sign of the associated potential ψ_l. Note that g_x refers to the x component of the gravity vector. The above equations are discretized in a cell centered (or block centered) fashion [2]. A discussion of the convergence properties of such finite difference approximations defined on irregularly spaced cell centered grids is given elsewhere [4].

Note that all the above equations have the form

$$\text{(change in mass in a time step)} - [\text{(flow in)} - \text{(flow out)}] = 0. \tag{13}$$

The first term in Eq. (13) is referred to as the accumulation term while the second term is called the flow term. Note that the accumulation term for cell i contains only the variables from cell i, while the flow term contains cell i variables as well as variables from the nearest neighbors of cell i. This type of cell connectivity gives rise to a five point computational molecule in two dimensions, and a seven point molecule in three dimensions. The variables in the flow term are discretized in time by first order forward (explicit) or backward (implicit) differences, as shown in Eq. (8)–(12). The accumulation terms are always differenced in the mass conservative manner shown above. Further details of the discretization method can be found elsewhere [1–3].

Three primary variables are chosen in each finite difference cell. If there is free gas present, then the cell is said to be saturated or below the bubble point. The primary variables in this case are the oil pressure p_o, water saturation S_w and oil saturation S_o. When no free gas is present, the saturation pressure p_s replaces S_o as the primary variable. In the absence of free gas, the cell is said to be undersaturated or above the bubble point. All other variables are functions of these three primary variables. Further details concerning switching criteria for cells crossing the bubble point are given in reference [5].

The system of Eqs. (1)–(3) is of mixed hyperbolic–parabolic type. For example, if $S_w = S_g = 0$, the system reduces to the familiar parabolic diffusion equation with p_o

as the primary variable [2, 3]. Another extreme is given by the one dimensional incompressible oil–water system ($S_g = 0$). In this case the system reduces to a pure first order hyperbolic equation with S_w as the primary variable [2, 3].

If the flow terms are discretized explicitly in the single phase parabolic limit, the time step is limited by the compressibility of the system. For oil–water systems, this compressibility is normally very small [2]. Consequently, all methods always take the oil phase pressure p_o implicitly. This observation gives rise to the two widely used methods of time discretization noted above. In the IMPES method (Implicit Pressure, Explicit Saturation), the oil pressure is taken implicitly, while the saturations (or saturation pressures) are taken explicitly in the flow terms. Note that the saturations are hyperbolic-like variables. Consequently, the maximum stable time step for IMPES systems is limited by a throughput condition [2, 3]. This condition essentially requires that no more than one pore volume of fluid can pass through any cell during a time step [2, 3], and can be especially severe near wells.

Note that the source terms in Eqs. (1)–(3), which correspond to wells, are essentially line sources in three dimensions. Thus, the predominant behavior of the pressure field p_o near wells can be expressed as a logarithmic function of the distance to the well center [2]. Since the fluid velocity is proportional to the pressure gradient, flow rates will be very high near wells [2]. Single well studies are usually carried out in radial coordinates. These simulations are termed coning studies since they show the upward movement of water and downward movement of gas towards a producing well. Since very small cells are used near the wellbore to give high resolution, a severe throughput condition results. Moreover, if free gas is present, the small gas viscosity can lead to particularly high flow rates [2]. In all these cases the maximum stable IMPES time step is unacceptably small [2, 3]. To avoid these problems, a fully implicit method of time discretization is also widely used. This method requires that all three unknowns be evaluated at the new time level in the flow terms. Since it can be shown that simple iteration on the saturations diverges unless the time step is the order of the explicit time step [3], full Newton iteration is used to solve the discretized non-linear algebraic equations. Full Newton iteration has been found to have acceptable convergence properties for time steps several orders of magnitude larger than a typical explicit time step [6]. However, Newton iteration requires solution of an associated Jacobian system. For a fully implicit scheme, with three unknowns per cell, solution of this linear system can be very costly for large three dimensional problems.

Recently, Thomas and Thurnau [1] have suggested the use of an adaptive implicit method to reduce execution times for black oil problems. This method is based on the idea that at any given time during a simulation, only a few cells need to be solved fully implicitly, while the majority of cells can be solved in an IMPES manner.

This method had been demonstrated for some simple problems where high flow rates are confined to areas near the well, and Gaussian elimination is used to solve the Jacobian matrix. However, it is not clear that the same savings in execution

time can be expected if an efficient iterative technique is used to solve the Jacobian [7–9]. Also, there has been very little discussion of the criteria used to select implicit cells [1].

Consequently, in this paper we will develop an iterative method based on an incomplete LU factorization [10, 11] of the adaptive implicit Jacobian with ORTHOMIN [7] acceleration. This method uses minimal storage (only non-zeros stored) and so can result in a substantial storage for large problems. Two types of preconditioning have been developed, one based on a second degree, natural ordering and the other using red–black reduced system preconditioning [8, 9]. The resulting incomplete factorization code can solve completely general sparse systems with a variable number of equations per cell. Since the code is completely general, other orderings or degrees of factorization are trivial to implement. Solution of problems with multi-block well completions [12] or local mesh refinement [13] can also be solved with such a general code.

A systematic discussion of the methods used to select implicit cells is also presented. In many cases, the execution times are surprisingly insensitive to the choice of AIM (adaptive implicit) parameters.

2. Iterative Solution of the Aim Jacobian

The discretized equations of the adaptive implicit method (AIM) were presented in Eq. (8)–(12). In the following, we will assume that there are only two types of cells. These will be either fully implicit, in which case (p_o, S_o, S_w) or (p_o, p_s, S_w) are solved implicitly, or IMPES, in which case only p_o is solved implicitly in the flow terms. We do not consider the case where only two variables are taken implicitly, for reasons to be discussed later. In the case of an IMPES cell, the Jacobian for the three flow equations (oil, gas, water) is constructed as usual. Note that saturation (or saturation pressure) derivatives appear only in the diagonal submatrix. Suppose that an IMPES cell i is a neighbor of a fully implicit cell k. Then, it is easy to see that the equations for cell i contain derivatives with respect to $S_{o_i}^{N+1}$, $S_{w_i}^{N+1}$, only in the diagonal block. These equations do have off diagonal derivatives with respect to $S_{o_k}^{N+1}$, $S_{w_k}^{N+1}$, however, since cell k is an implicit cell. Turning attention to the equations of cell k, there are no derivatives with respect to $S_{o_i}^{N+1}$, $S_{w_i}^{N+1}$ (cell i is IMPES). There are derivatives with respect to $S_{o_k}^{N+1}$, $S_{w_k}^{N+1}$ coming from both the diagonal accumulation term, and the off-diagonal flow terms.

Another way to visualize the Jacobian is to imagine constructing the Jacobian as a block matrix by columns. Consider cell i. The material balance equations for cell i contain accumulation terms (viz. Eq. (13)) which contain all the primary variables evaluated at the $(N+1)$ time level (see Eqs. (8)–(12)). Consequently, the diagonal submatrix is in general full, since derivatives with respect to $(p_{o_i}^{N+1}, S_{o_i}^{N+1}, S_{w_i}^{N+1})$ are present. If cell i is IMPES, there are no additional derivatives in the diagonal submatrix with respect to $(S_{o_i}^{N+1}, S_{w_i}^{N+1})$. If cell i is fully implicit, there are additional derivatives with respect to $(S_{o_i}^{N+1}, S_{w_i}^{N+1})$ in the diagonal submatrix coming from the flow terms, since all variables evaluated at cell i are taken at the $M=$

$N+1$ time level. Now consider the Jacobian entries in the same ith column above and below the ith diagonal submatrix. The non-zero submatrix entries correspond to derivatives of the material balance equations for neighbors of cell i with respect to $(p_{o_i}^{N+1}, S_{o_i}^{N+1}, S_{w_i}^{N+1})$. If cell i is an IMPES cell, the corresponding submatrix will only contain a column corresponding to derivatives taken with respect to $p_{o_i}^{N+1}$, recalling that Eqs. (8)–(12) show that $p_{o_i}^{N+1}$ appears in the flow equations for all neighboring cells. All properties which are functions of (S_{w_i}, S_{o_i}) appear evaluated at the $M=N$ time level in all equations for cells which are neighbors of cell i, and do not generate Jacobian terms. Conversely, if cell i is a cell receiving a fully implicit treatment, the quantities which are functions of (S_{o_i}, S_{w_i}) in the equations for neighboring cells are all evaluated at time level $M=N+1$. Consequently, all submatrices lying above and below the diagonal submatrix are potentially full, as derivatives with respect to $p_{o_i}^{N+1}$, $S_{o_i}^{N+1}$, and $S_{w_i}^{N+1}$ could appear.

After construction of the Jacobian system, each block row of the matrix is scaled by the corresponding inverse of the diagonal submatrix. For implicit blocks, this procedure provides a scaling for the solution algorithm. For IMPES cells, this operation effectively eliminates the IMPES cell saturation variables from the system. This system reduction can be seen as follows. If cell i is IMPES, then after diagonal scaling, the only non-zero entries in the columns corresponding to $(S_{o_i}^{N+1}, S_{w_i}^{N+1})$ appear on the diagonal of the diagonal submatrix for cell i. This means that the two resulting equations containing $(S_{o_i}^{N+1}, S_{w_i}^{N+1})$ derivative dependence can simply be left out of the Jacobian system, as no other equations show dependence on $(S_{o_i}^{N+1}, S_{w_i}^{N+1})$ derivatives. After solving for the other primary variables, $(S_{o_i}^{N+1}, S_{w_i}^{N+1})$ can be determined by back substitution.

As a further illustration, consider the simple two dimensional grid in Fig. 1, with the pattern of implicit-IMPES cells as shown. After leaving out the equations containing IMPES cell saturation variables (as explained above) and performing diagonal submatrix diagonalizations, the Jacobian has the structure shown in Fig. 2.

It is now possible to construct a block incomplete LU factorization of the Jacobian. In the case of the fully implicit Jacobian, all blocks are 3×3 [6, 7]. However, for an AIM Jacobian, the blocks are either 3×3, 1×3, 3×1, or 1×1. As long as the incomplete factorization is carried out in a block sense, with due regard

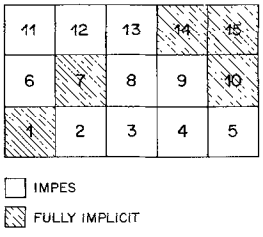

FIG. 1. Two dimensional example with a typical fully implicit-IMPES pattern.

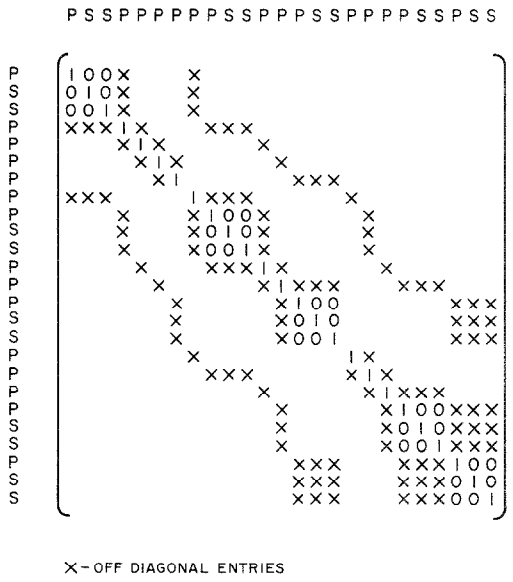

FIG. 2. Jacobian incidence matrix for the example of Fig. 1.

for matrix multiplication with non-square matrices, then the incomplete factorization can be constructed as described previously [7–9].

The solution algorithm essentially consists of three parts. The first part entails a symbolic factorization. Since the symbolic factorization is done in a block sense, it is independent of the degree of implicitness. Consequently, the symbolic factorization need only be done at the beginning of the simulation. The second part of the algorithm consists of pointer adjustment to account for the degree of implicitness (recall that only non-zeros are stored). This pointer adjustment is comparatively cheap once the symbolic factorization is carried out. Pointer adjustment must be carried out every time the degree of implicitness changes, which is quite frequently (every few time steps). The third part of the algorithm is the usual numeric incomplete factorization, followed by the repeated iterative cycle of forward and back solve and ORTHOMIN acceleration [7–9].

Note also that using a cell by cell approach for the incomplete factorization rather than a line by line approach requires less pointer storage. In general, for a fully implicit Jacobian, a line by line approach would require nine times (3×3) as much pointer storage as a cell by cell approach.

The types of incomplete factorization used in this study were a second degree naturally ordered factorization, and a reduced system precondition using red–black ordering. These ILU factorizations have been discussed in detail elsewhere [8, 9]. As mentioned previously, the ORTHOMIN algorithm [7] is used to accelerate the iterative method, since the Jacobian is in general non-symmetric.

3. Selection of Implicit Cells

Before discussing the criteria for selecting implicit cells, a short digression is

The first few time steps after a well opening require small time steps in order to resolve transients. Thereafter, the time steps can be increased quite rapidly until the next well change is encountered (assuming that no stability problems arise). The time step selector used in this paper was of the form

$$\Delta t^{N+1} = \Delta t^{N} \left[\frac{D_{\text{norm}}(I)}{\text{Del}(I)^{N}} \right]_{\text{min over variables, cells}}, \tag{14}$$

where $\text{Del}(I)^N$ is the maximum change in each basic variable over the previous time step [2]. The parameter $D_{\text{norm}}(I)$ is the desired value of a change in a variable over a time step. The ratio $\Delta t^{N+1}/\Delta t^N$ is bounded between 0.5 and 5. Typical values for the norms are

$D_{\text{norm}}(1) = 500$ psi for pressure
$D_{\text{norm}}(2) = 500$ psi for saturation pressure
$D_{\text{norm}}(3) = 0.10$ for oil, gas, water saturations.

These time step selection norms will be discussed in the following.

The criterion for selection of implicit cells described by Thomas and Thurnau [1] was based on a specified saturation or pressure change threshold from a previous iteration. It is immediately obvious that such criteria can only be used to switch cells from IMPES to fully implicit, and not vice versa. This is because a fully implicit cell can have an extremely large throughput, and yet the saturation changes can be small. It is clear that large instabilities would ensue if such a cell was switched to IMPES. However, the limitation of switching only from IMPES to fully implicit is not necessarily too restrictive for practical problems. In a typical black oil run, several small time steps are taken after an initial well opening, and thereafter the time steps increase rapidly until a well change is encountered, and the cycle of small time steps increasing to large time steps is repeated. Consequently, we will adopt the following strategy. After each well change (which requires small time steps) only a few cells are set fully implicit, such as well cells, nearest neighbors and/or next nearest neighbors. Once a cell is switched to fully implicit, it is not reset until the next non-trivial well change.

Nevertheless, it is still not obvious that simply switching cells to fully implicit when large saturation changes are observed will capture all instabilities. Clearly, extreme instabilities with large changes are easily discerned with such a strategy. However, slowly growing instabilities may not be detected. Consequently, in order to have a reasonable chance of detecting slowly growing instabilities, the saturation change thresholds must be significantly smaller than the norms which control time step selection (Eq. (14)). This strategy is essentially conservative since saturation

changes of this magnitude are necessary for instability but not sufficient. In line with this conservative strategy, an IMPES cell is switched to fully implicit in all three variables. This avoids problems which could occur, for example, with large saturation pressure changes in a cell just before a gas switch is encountered [5].

The above method was developed after numerous runs were made with a fixed implicit-IMPES pattern and the results compared with those from a fully implicit run. Even though cumulative variables such as oil production and water production were in good agreement in many cases, small instabilities were observed if not enough fully implicit cells were used. These instabilities sometimes caused repeated time steps later in the run. Consequently, we have attempted to use the most conservative strategy possible which will still allow a significant reduction in run time.

Thus, our strategy uses mostly IMPES cells for the small time steps after a well change, and as the time steps are increased, cells are switched to fully implicit as saturation and saturation pressure changes start to exceed a small preset fraction of the time step selection norms. A fraction of 20 to 25% was found to be suitable for most applications.

The above strategy was augmented by a facility that caused neighbors of switched cells to also switch if their changes exceeded a slightly smaller threshold value. This facility helped to prevent the appearance of isolated explicit cells in otherwise fully implicit regions.

This method can also be extended to solve the fully implicit equations in all cells at reduced cost. In this case the residual is evaluated using the fully implicit discretized equations. However, the saturation derivatives of the flow terms are ignored (for the purposes of Jacobian construction) in cells where saturation changes are small. This approximate Jacobian will have the same structure as the AIM Jacobian, and hence will be less expensive to solve than the full Jacobian. Note that this is not full Newton iteration. This is in contrast to the true AIM method which uses explicit variables in IMPES cells, and results in full Newton iteration.

The iterative solution method described previously was incorporated in a black oil reservoir simulator that provides the mechanism for equation formation using a variable implicitness formulation. As noted previously, the three conservation equations for each cell were either solved fully implicitly, or in an IMPES fashion, using full Newtonian iteration. A variable bubble point formulation, using saturation pressure, was incorporated. The model was used for a variety of simulation problems, including the problems presented by Odeh [14] and Chappelear and Nolen [15] in the first and second SPE comparative solution projects. The results of these computations are presented here, including CPU (central processing unit) times for a Honeywell DPS-8. These times are only accurate to $\pm 5\%$ and include time spent for data input and output. One set of times is also given for a CRAY-1S.

It was generally found that a 40% reduction in computing time could be realized by the adaptive implicit approach, as compared to simulations performed with all cells initialized to the implicit state (fully implicit), when using the new iterative

matrix solution algorithm. Generally, the computing times, and the results were not sensitive to the choice of the switching threshold. Also, if the adaptive implicit Jacobian was viewed as an approximate replacement for the matrix generated by the fully implicit equations, and the fully implicit solution was solved for using this approximation, it was found that a solution was obtained in times comparable to the adaptive implicit approach. There was again little difference between the fully implicit solution and the adaptive implicit solution.

The results for the first SPE comparative solution project are given in Table I. This problem was a gas injection problem on a $10 \times 10 \times 3$ grid. Complete details of the description of this problem can be found in Ref. [14]. Neighbors and next-nearest neighbors of well cells were set implicit initially, as were the well cells themselves. A maximum time step of 250 days was used. Timing results are shown for various levels of the threshold parameters, as well as for a fully implicit simulation. Also, timings are shown for two cases where the fully implicit equations were solved using an adaptive implicit Jacobian. Material balance calculations were performed for each component at the end of each time step as follows: (initial mass + injection − production − calculated mass)/max (injection, production). Material balance errors were always less than 8×10^{-3} and final cumulative productions differed by less than 4% for all cases. A 40% reduction in CPU time for the adaptive implicit cases, as compared to the fully implicit case 4, and the insensitivity of timings to the threshold size, can be seen. Also, it does not take appreciability longer to solve the fully implicit equations by use of the adaptive implicit Jacobian than it does to obtain the adaptive implicit solution.

It should be noted that approximately two-thirds of the cells were implicit at the end of each adaptive implicit simulation. However, the time weighted fraction of cells implicit did not exceed 40% for case 1 and 33% for cases 2 and 3. Thus, even

TABLE I

Results for the First SPE Comparative Solution Project

	Time step selection norm:	pressure	1000.0 psi
		saturation pressure	1000.0 psi
		saturations	0.20

Case	Saturation pressure threshold (psi)	Saturation threshold	CPU time (sec) (Honeywell DPS-8)
1	125.0	0.025	1285 (1336[a])
2	250.0	0.050	1239 (1358[a])
3	600.0	0.150	1334
4	Fully implicit throughout		2178

[a] Fully implicit solution found using adaptive implicit Jacobian.

though the top layers of the reservoir contain mostly mobile gas, and are made up of mostly implicit cells at the end of the simulations, the simulations required only one-third of the cells implicit, on average.

The results for the second SPE comparative solution project are given in Table II. This problem was a single well cross-sectional coning study in radial coordinates on a 10×15 grid. Further details can be found in Ref. [15]. Neighbors and next-nearest neighbors of the producing well blocks were set implicit initially, as were the well cells. All cells reverted to IMPES at the 720.0 day well change, which represented the major well change in this problem, in accordance with the strategy described before. (It should be noted that this resetting had little effect on pressure control had no effect on this coning problem.) Also, timings are shown for two cases where the fully implicit equations were solved using an adaptive implicit Jacobian. Material balance errors were always less then 5×10^{-4} and final cumulative productions differed by less than 4% for all cases. A 40% reduction of CPU time for the adaptive implicit case 1, as compared to the fully implicit case 4, is again evident, and a 30% reduction holds for case 2. The saturation threshold of 0.01 (20% of the associated norm) yields the best performance for this problem (case 1).

Case I had a time weighted fraction of cells implicit value of 20% for this coning problem, with no more than half the cells fully implicit at any time. A total of 43 time steps were required, involving 145 Newton iterations. It should be again noted that the additional time required to solve the fully implicit equations using an adaptive implicit Jacobian is small.

TABLE II

Results for the Second SPE Comparative Solution Project

Time step selection norms:	pressure	250 psi
	saturation pressure	250 psi
	saturations	0.05

Case	Saturation threshold	CPU time (sec) (Honeywell DPS-8)
1	0.010	313[b] (349[a])
2	0.025	373 (388[a])
3	0.050	418 (418[a])
4	Fully implicit throughout	513

[a] Fully implicit solution found using adaptive implicit Jacobian.
[b] This run required 19.2 sec on the CRAY-1S.

The CRAY-1S time of 19.2 seconds for this problem compares favourably with times reported in [15]. It should be noted that the coding of this model makes no special provision for vector machine architecture, however, and that full Newtonian iterations and a complete variable bubble point formulation were always used. The latter considerations can lead to greater computing times for this model when compared to times for codes that are vectorized, or that take special advantage of the stability of the water zone and the saturated nature of the problem.

The final problem involves a water injection scheme during which bubble point transitions occur. The geometry selected is as for the first SPE comparative solution project data, as are most of the fluid properties. It is assumed now that the reservoir is quite undersaturated ($p_s = 3000$ psi), while the initial reservoir pressure is retained at $p_o = 4800.0$ psi. Quadratic water–oil relative permeability curves are used and water is injected into cell (10, 10, 3) at a rate of 20,000 bbl/day. The same production well is used. The simulation covers a period of 3650.0 days, during which time free gas appears near the production well. A constraint violation near 900 days causes the production well to be operated as a constant bottom hole pressure well after this time.

Some timing results for this problem are shown in Table III. Case I began with the usual neighbors and next-nearest neighbors of well cells implicit (20 implicit cells) and case 2 began with 54 cells implicit in a neighborhood of the wells. The timings show that switching thresholds and initial setting strategy have little effect for this problem. A 45–50% reduction in time is observed as compared to a simulation performed with all cells implicit (case 3). The time weighted fraction of cells implicit for case 1 did not exceed 20%. Material balance errors never exceeded 3×10^{-3} for all cases, and total production values differed by less than 1%.

Case 4 describes results obtained by fixing only the production cell implicit and otherwise using an IMPES approach. The maximum time step was reduced to a

TABLE III

Results for a Water Injection Scheme

	Time step selection norms:	pressure	1000.0 psi
		saturation pressure	1000.0 psi
		saturations	0.20

Case	Saturation pressure threshold (psi)	Saturation threshold	CPU time (sec.) (Honeywell DPS-8)
1	200.0	0.050	359
2	300.0	0.100	331
3	Fully implicit throughout		653
4	IMPES throughout[a]		1077

[a] Max time step 30 days.

size suitable for an IMPES simulator. It should be noted that if the production cell is not taken implicit (essentially defining a fully implicit well formulation), the simulator could not complete the problem in a reasonable time.

In order to verify that the maximum stable time step was used for the IMPES run, the maximum time step was doubled to 60 days. A large number of repeat time steps were observed, indicating that the maximum stable time step was exceeded.

The advantages of using an implicit–explicit approach, or even a fully implicit model, over IMPES, can be seen. In fact, the IMPES approach required three times as much time as the adaptive approach, and half again as much time as the fully implicit simulation.

Table IV summarizes some further information concerning the adaptive implicit, fully implicit and IMPES cases. Even though the IMPES method only uses 2.4 iterations per time step and its time per iteration is least, the large number of time steps required lead to large cost. The fully implicit method's advantage derives from its ability to take large time steps, when compared to the IMPES approach. Finnally, the adaptive implicit method exhibits a cost per iteration near that of the IMPES appraoch, but still can take fully implicit time steps, leading to a particularly low overall cost.

The first SPE problem [14] was rerun using the same parameters as Case 1, Table I, with the reduced system red–black ordered preconditioning. This preconditioning reduced the run time by approximately 5%. Although the reduced system ILU [8, 9] was more efficient on some difficult time steps, there were many relatively easy time steps where the increased set-up costs of the reduced system ILU did not pay off. We have found that for practical problems, which have orders of magnitude differences in time steps, the second degree natural ILU [7–9] seems to be the most efficient.

The test problems presented above are quite difficult in the sense that a reasonable fraction of the cells become implicit at the end of the run. However, we have run many practical simulations on field scale problems, in which only the well cell and nearest neighbors become implicit. In this case, the CPU savings are much larger than the 40–50% indicated above. With the AIM technique, it has become possible to run fairly large problems in a reasonable time on small machines with limited memory.

TABLE IV

Results for a Water Injection Scheme

	Implicit–explicit (Case 1)	Fully implicit (Case 3)	IMPES (Case 4)
Total time steps	20	19	128
Total iterations	77	65	302
Iterations/time step	3.8	3.4	2.4
Total time (sec)	359	653	1077
Time/iteration (sec)	4.7	10.0	3.6

5. Conclusions

An iterative matrix solving technique has been developed for adaptive implicit Jacobian matrices. This technique is based on an incomplete LU factorization and is related to methods that have already proven robust and efficient for fully implicit reservoir simulation and for IMPES simulators [7–9]. This method uses a cell by cell approach which minimizes the work required for symbolic factorization. A second degree natural and reduced system preconditioning were used. The technique uses minimal storage. The new method consistently shows at least 40% improvement in total computing time when compared to fully implicit simulations, for a variety of black oil problems. It has also been shown that the new method can be used to generate an approximate Jacobian. This approximate Jacobian can be used to solve the fully implicit reservoir equations using less time than with the fully implicit Jacobian.

The conservative switching strategy described above, which is based on the observations of Thomas and Thurnau [1], provides stable simulations. However, it is still capable of attaining the significant computing time reductions noted above. The choice of parameters proved to be not particularly critical for typical black oil problems.

As is well known, and reiterated by the water injection problem described earlier, fully implicit simulators can outperform IMPES simulators on many problems due to their ability to take large time steps. Alternately, the low cost of the IMPES approach makes this technique useful for simple problems or small time steps. The adaptive implicit method combines these approaches and compensates for inappropriate time step norm selection if the switching strategy described above is used. If norms are chosen small, but the problem is not particularly difficult, only a few cells should switch implicit and the simulation can take advantage of many cells where an IMPES formulation is used. If norms are chosen large for a somewhat difficult problem, the model will revert to a fully implicit formulation and thereby will take the largest time steps possible.

The adaptive implicit method using the new iterative solver, with its inherent benefits for larger problems, gives a stable simulation procedure that offers significant computing cost reductions over other techniques.

Acknowledgments

This research was supported by the General Members of CMG. The authors are indebted to A. Rabie

References

1. G. W. Thomas and D. H. Thurnau, *Soc. Pet. Engrg. J.* **23** (1983), 760–768.
2. K. Aziz and A Settari, "Petroleum Reservoir Simulation," Applied Science, London, 1979.

3. D. W. PEACEMAN, "Fundamentals of Numerical Reservoir Simulation," Elsevier, Amsterdam, 1977.
4. P. A. FORSYTH, JR., AND P. H. SAMMON, submitted for publication.
5. P. A. FORSYTH AND P. H. SAMMON, *Soc. Pet. Engrg. J.* **24** (1984), 505–507.
6. A. D. K. AU, A. BEHIE, B. RUBIN, AND P. K. W. VINSOME, "Techniques for Fully Implicit Reservoir Simulation," paper SPE 9302, presented at the Fall Meeting of SPE, Dallas, 1980.
7. A. BEHIE AND P. K. W. VINSOME, *Soc. Pet. Engrg. J.* **22** (1982), 659–668.
8. A. BEHIE AND P. FORSYTH, *SIAM J. Sci. Statist. Comput.* **5** (1984), 543–561.
9. A. BEHIE, D. COLLINS, AND P. FORSYTH, *J. Comput. Methods Appl. Mech. Engrg.* **42** (1984), 287–299.
10. J. A. MEIJERINK AND H. A. VAN DER VORST, *Math. Comp.* **31** (1977), 148–162.
11. D. S. KERSHAW, *J. Comput. Phys.* **26** (1978), 43–65.
12. R. H. TRIMBLE AND A. E. MCDONALD, *Soc. Pet. Engrg. J.* **21** (1981), 454–458.
13. P. QUANDALLE AND P. BESSET, "The Use of Flexible Gridding for Improved Reservoir Modelling," paper SPE 12239, presented at the Seventh SPE Symposium on Reservoir Simulation, San Francisco, 1983.
14. A. ODEH, *J. Pet. Tech.* **33** (1981), 13–25.
15. J. E. CHAPPELEAR AND J. S. NOLEN, "Second Comparative Solution Project: A Three Phase Coning Study," paper SPE 10489, presented at the Sixt SPE Symposium on Reservoir Simulation, New Orleans, 1982.