Iterative coupling reservoir simulation on high performance computers

Lu Bo[1]* and Wheeler Mary F.[2]

[1] BP America Inc., Houston, TX, 77079, US

[2] Department of Petroleum & Geosystems Engineering, The University of Texas at Austin, Austin, TX, 78712, USA

Abstract: In this paper, the iterative coupling approach is proposed for applications to solving multiphase flow equation systems in reservoir simulation, as it provides a more flexible time-stepping strategy than existing approaches. The iterative method decouples the whole equation systems into pressure and saturation/concentration equations, and then solves them in sequence, implicitly and semi-implicitly. At each time step, a series of iterations are computed, which involve solving linearized equations using specific tolerances that are iteration dependent. Following convergence of subproblems, material balance is checked. Convergence of time steps is based on material balance errors. Key components of the iterative method include phase scaling for deriving a pressure equation and use of several advanced numerical techniques. The iterative model is implemented for parallel computing platforms and shows high parallel efficiency and scalability.

Key words: Iterative coupling, reservoir simulation, multiphase flow, phase scaling, parallel scalability

1 Introduction

Reservoir simulation is one of the most useful tools in reservoir engineering. It accomplishes the following objectives: history matching and prediction of pressure and saturation in reservoirs; understanding fluid flow and oil recovery processes; and supporting optimal production plans. Reservoir simulation is based on the numerical approximation of solutions to the equation systems described by mass conservation and Darcy's Law. In these equation systems one encounters elliptic, parabolic, and near-hyperbolic equations with complicated nonlinear behavior arising from fluid and reservoir properties. Computational complexity arises from the high spatial heterogeneity of multi-scale porous media. This heterogeneity, together with measurement limitations, leads to uncertainties in simulation. As a result, numerically approximating subsurface phenomena is an intricate problem, which is critical to the industry for accurate predictions of costly projects. Accordingly, we propose the iterative method to address the time-stepping issues for the purpose of both minimizing computational cost (CPU time) and providing accurate solutions.

The fully implicit method (FIM) and the implicit pressure explicit saturation (IMPES) method are widely used as time-stepping approaches in reservoir simulation (Chen et al 2006; Coats, 1982; Fanchi, 2001; Kendall et al, 1983; Peaceman, 1977; Watts, 1986). The FIM solves reference pressure and saturation/concentration simultaneously within a given time step. The implicitness makes the FIM unconditionally stable

but computationally costly. The IMPES method is based on operator time-splitting that involves solving a reference pressure equation obtained by operating the mass balance and Darcy flow equations. Saturation/concentration equations are solved using total velocity and fractional mobility. The IMPES method, while computationally inexpensive, exhibits unacceptable oscillatory solutions unless very small time steps are employed.

Several other approaches, such as the sequential method and the adaptive implicit method (AIM), are focusing on combining the advantages of both the FIM and the IMPES to achieve better stability, accuracy, and efficiency (Fagin and Stewart, 1966; Fung et al, 1998; Spillette et al, 1973; Watts and Shaw, 2005). The sequential method solves pressure and saturation both implicitly, and in sequence. In general cases it improves the solution accuracy over the IMPES method. However, it requires additional computer memory and has problems in handling complicated capillary pressure curves. The AIM uses both the FIM and the IMPES in one simulation by applying the FIM to complicated subdomains and applying the IMPES to less heterogeneous subdomains. The assignment of different methods is based on error estimation at the beginning of each time step. The AIM shows advantages over other methods in simulation on serial computers, but it encounters obvious difficulties in parallel computing environments due to the load imbalance problem.

Parallel implementation is one of the standard functionalities of the next generation reservoir simulators (DeBaun et al, 2005; Fjerstad et al, 2005; Øian et al, 2005; Shiralkar et al, 2005). With parallel computation, one large problem is divided into several small sub-problems across multiple processors, and each processor solves its local

---

*Corresponding author. email: Bo.Lu@bp.com, utlubo@gmail.com

problem. Several domain decomposition methodologies, parallel programming libraries and linear solvers are applied by different reservoir simulators.

In this paper we select a new approach, iterative coupling, to solve the equation systems in reservoir simulation. It covers most of the problems mentioned above (Lu et al, 2007; Lu, 2008). In particular, effort will be expended on pursuing high parallel scalability and efficiency of the iterative method, under high performance computing (HPC) environments.

2 Model formulation

Iterative coupling is an operator-splitting technique that decouples the multiphase flow equation system into pressure and saturation/concentration equations, which are solved separately. One typical time step of the iterative model is shown in Fig. 1. Each time step starts with the extrapolation to calculate the initial values, and then a forcing function is used to tighten linear solver tolerance (Dawson et al, 1997; Klie, 1996; Lacroix et al, 2003). Phase saturations and material balances are checked to determine whether convergence is satisfied. If not, an additional iteration is conducted, in which nonlinear coefficients are updated and iteration tolerances are tightened. Sequential iterations are repeated until the specified target convergence condition is achieved.
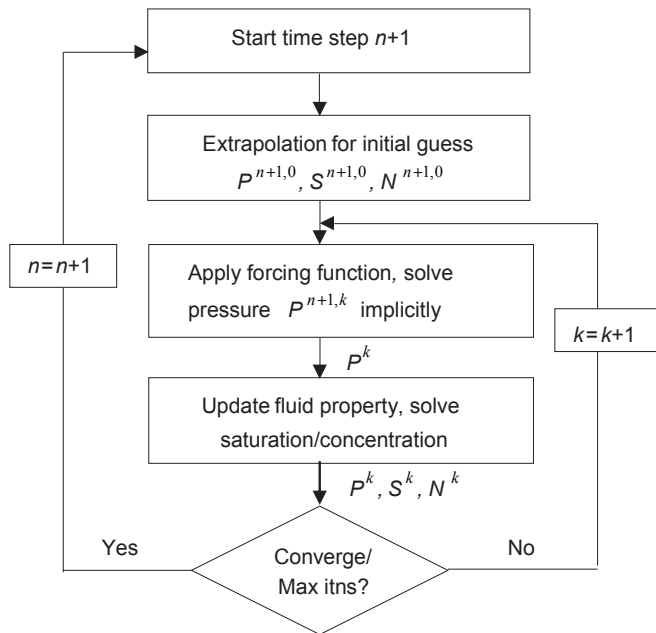

Fig. 1 Typical time step loop of the iterative model

2.1 Assumption

The following assumptions should be made before the development of the iterative model: an isothermal reservoir with instant thermodynamics equilibrium in all directions, slightly compressible formation and Darcy flow, a diagonal permeability matrix, no precipitation and absorption, and the well is treated as either a source or a sink point. In a black

oil fluid system, the gas component may exist in either the oil phase or the gas phase, but not in the water phase; the oil component exists only in oil phase; and no mass transfer occurs between the water phase and the other two phases.

2.2 Pressure equation

The governing equations for a black oil fluid system consist of the mass conservation equation for each component and the Darcy flow equation for each phase.

For three components (water, oil and gas), we have:

$$\frac{\partial (\phi N_W)}{\partial t} = -\nabla \cdot (\rho_W u_w) + q_W$$

$$\frac{\partial (\phi N_O)}{\partial t} = -\nabla \cdot (\rho_O u_o) + q_O$$

$$\frac{\partial (\phi N_G)}{\partial t} = -\nabla \cdot (\rho_G u_g + R_{so} \rho_G' u_o) + q_G$$

where subscripts W, O, and G denote water, oil, and gas components; subscripts w, o, and g denote water, oil, and gas phases; N is component concentration; ρ is density; ϕ is porosity; u is volumetric velocity; R_{so} is solution gas-oil ratio; and q is the mass rate of injection/production in wellbore, which is given by

$$q_W = \frac{\rho_{WS}}{B_w} Q_{WS}$$

$$q_O = \frac{\rho_{OS}}{B_o} Q_{OS} \tag{2}$$

$$q_G = \frac{\rho_{GS}}{B_g} Q_{GS} + R_{so} \frac{\rho_{GS}}{B_o} Q_{OS}$$

where Q_S is the volumetric flow rate of each component under standard conditions; B is formation volume factor.

Darcy's law for multiphase flow is applied in the form of velocity:

$$u_w = -K \frac{k_{rw}}{\mu_w} (\nabla P_w - \rho_w g \nabla D)$$

$$u_o = -K \frac{k_{ro}}{\mu_o} (\nabla P_o - \rho_o g \nabla D) \tag{3}$$

$$u_g = -K \frac{k_{rg}}{\mu_g} (\nabla P_g - \rho_g g \nabla D)$$

where K is reservoir permeability; k_{ra} is the relative permeability of phase α; μ is viscosity; P_α is the pressure of phase α; and D is the depth below the reference level. The mobility of phase α is defined as $\lambda_\alpha = k_{r\alpha}/\mu_\alpha$.

Substituting Eq. (3) into Eq. (1), and then summing all three mass equations give the pressure equation of the iterative black oil model. However, the differences of viscosity and density among the three phases may lead to phase imbalance. Compared with the oil and water phases, the gas phase is neglectable due to its small density and low viscosity, which reduces the accuracy of the solutions.

To avoid the phase imbalance, phase scaling is used in the iterative model. The mass equations for water, oil and gas phases are normalized by factors, $\dfrac{\mu_w^*}{\rho_{WS}}$, $\dfrac{\mu_o^*}{\rho_{OS}}$, $\dfrac{\mu_g^*}{\rho_{GS}}$, respectively, yielding

$$\frac{\partial}{\partial t}\left(\phi\mu_w^*\frac{S_w}{B_w}\right)$$
$$=\nabla\cdot\left[K\frac{\mu_w^*\lambda_w}{B_w}\left(\nabla P_w-\rho_w g\nabla D\right)\right]+\mu_w^*\frac{Q_{WS}}{B_w} \quad (4)$$

for water,

$$\frac{\partial}{\partial t}\left(\phi\mu_o^*\frac{S_o}{B_o}\right)=\nabla\cdot\left[K\frac{\mu_o^*\lambda_o}{B_o}\left(\nabla P_o-\rho_o g\nabla D\right)\right]+\mu_o^*\frac{Q_{OS}}{B_o} \quad (5)$$

for oil, and

$$\frac{\partial}{\partial t}\left[\phi\mu_g^*\left(\frac{S_g}{B_g}+R_{so}\frac{S_o}{B_o}\right)\right]$$
$$=\nabla\cdot\left[K\mu_g^*\frac{\lambda_g}{B_g}\left(\nabla P_g-\rho_g g\nabla D\right)\right]$$
$$+\nabla\cdot\left[KR_{so}\mu_g^*\frac{\lambda_o}{B_o}\left(\nabla P_o-\rho_o g\nabla D\right)\right]+\mu_g^*\frac{Q_{GS}}{B_g}+R_{so}\mu_g^*\frac{Q_{OS}}{B_o} \quad (6)$$

for gas, where μ^*, ρ_s are reference viscosity and reference density under standard conditions.

Capillary pressure and saturation constraint equations are applied to Eqs. (4)-(6); the three equations are then summed together to yield the pressure equation of the iterative black oil model, as follows:

$$\nabla\cdot\left[K\left(\mu_w^*\frac{\lambda_w}{B_w}+\mu_o^*\frac{\lambda_o}{B_o}+\mu_g^*\frac{\lambda_g}{B_g}+R_{so}\mu_g^*\frac{\lambda_o}{B_o}\right)^k\nabla P_o^{k+1}\right]$$
$$-\frac{\left(C_{1w}+C_{1o}+C_{1g}\right)^k}{\Delta t^{n+1}}\delta P_o^{k+1}+\mu_w^*\frac{Q_{WS}}{B_w}+\mu_g^*\frac{Q_{GS}}{B_g}+\left(\mu_o^*+\mu_g^*R_{so}\right)\frac{Q_{OS}}{B_o}$$
$$=\left(D_{1w}+D_{1o}+D_{1g}\right)^k+D_{pcw}^k-D_{pcg}^k+\left(D_{2w}+D_{2o}+D_{2g}\right)^k \quad (7)$$

where coefficients C_{1w}, C_{1o}, and C_{1g} include derivatives of ϕ, R_{so}, and B with respect to pressure P_o as follows:

$$C_{1w}=\mu_w^*\left[\left(\frac{S_w}{B_w}\right)\left(\frac{d\phi}{dP_o}\right)+\left(\phi S_w\right)\left(\frac{dB_w^{-1}}{dP_o}\right)\right]^k$$
$$C_{1o}=\mu_o^*\left[\left(\frac{S_o}{B_o}\right)\left(\frac{d\phi}{dP_o}\right)+\left(\phi S_o\right)\left(\frac{dB_o^{-1}}{dP_o}\right)\right]^k \quad (8)$$
$$C_{1g}=\mu_g^*\left[\left(\frac{S_g}{B_g}+R_{so}\frac{S_o}{B_o}\right)\left(\frac{d\phi}{dP_o}\right)+\left(\phi S_g\right)\left(\frac{dB_g^{-1}}{dP_o}\right)\right]^k$$
$$+\mu_g^*\left[\left(\phi R_{so}S_o\right)\left(\frac{dB_o^{-1}}{dP_o}\right)+\left(\phi\frac{S_o}{B_o}\right)\left(\frac{dR_{so}}{dP_o}\right)\right]^k$$

All the other coefficients in Eq. (7) are listed below.

$$D_{pcw}^k=\nabla\cdot\left(K\mu_w^*\frac{\lambda_w}{B_w}\nabla P_{cow}\right)^k$$

$$D_{pcg}^k=\nabla\cdot\left(K\mu_g^*\frac{\lambda_g}{B_g}\nabla P_{cgo}\right)^k$$

$$D_{1w}^k=\mu_w^*\frac{\left(\phi\dfrac{S_w}{B_w}\right)^k-\left(\phi\dfrac{S_w}{B_w}\right)^n}{\Delta t^{n+1}}$$

$$D_{1o}^k=\mu_o^*\frac{\left(\phi\dfrac{S_o}{B_o}\right)^k-\left(\phi\dfrac{S_o}{B_o}\right)^n}{\Delta t^{n+1}}$$

$$D_{1g}^k=\mu_g^*\frac{\left[\phi\left(\dfrac{S_g}{B_g}+R_{so}\dfrac{S_o}{B_o}\right)\right]^k-\left[\phi\left(\dfrac{S_g}{B_g}+R_{so}\dfrac{S_o}{B_o}\right)\right]^n}{\Delta t^{n+1}}$$

$$D_{2w}^k=\nabla\cdot\left(K\mu_w^*\frac{\lambda_w}{B_w}\rho_w g\nabla D\right)^k$$

$$D_{2o}^k=\nabla\cdot\left(K\mu_o^*\frac{\lambda_o}{B_o}\rho_o g\nabla D\right)^k$$

$$D_{2g}^k=\nabla\cdot\left[K\left(\mu_g^*\frac{\lambda_g}{B_g}\rho_g+R_{so}\mu_g^*\frac{\lambda_o}{B_o}\right)g\nabla D\right]^k$$

The oil pressure is solved implicitly with Eq. (7), in which the well term is treated implicitly.

2.3 Saturation/concentration equation

Once the oil pressure is obtained, mass conservation equations are used to calculate oil and water concentrations, as follows:

$$N_O^{k+1}=\frac{\Delta t^{n+1}\left(-\nabla\cdot\left(\rho_O^{k+1}u_O^{k+1}\right)+q_O^{k+1}\right)+\phi^n N_O^n}{\phi^{k+1}}$$
$$N_W^{k+1}=\frac{\Delta t^{n+1}\left(-\nabla\cdot\left(\rho_W^{k+1}u_W^{k+1}\right)+q_W^{k+1}\right)+\phi^n N_W^n}{\phi^{k+1}} \quad (9)$$

In the concentration equation, all parameters related to pressure are updated with the new pressure, and all other parameters are upwinded from the latest iteration. Saturations are then calculated using their defined relationship to the concentrations (Wheeler, 1995).

3 Implementation on HPC

The iterative model is designed to be parallel scalable on HPC. A Message Passing Interface (MPI) library is used to realize the parallel communication. A "ghost cell" bridges the neighboring processors for communication (Lu, 2000; Lu, 2008), as shown in Fig. 2. On processor 1, the layer close to processor 2 is designated as the ghost layer, whose values

are copied to processor 2 as the boundary layer. Similarly, the ghost layer on processor 2 is copied to processor 1 as the boundary. During parallel computation, values on the ghost

layers are exchanged between the neighboring processors. With more processors and irregular grids, the situation is more complicated but the same strategy is applied.
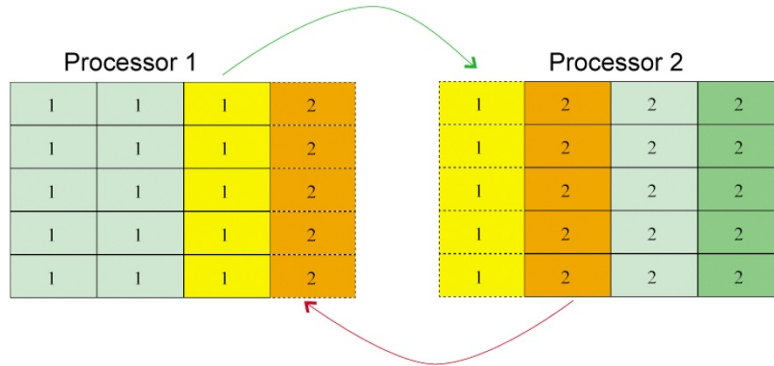

Fig. 2 Communication through ghost cells in 2D

In the iterative model, grid-based load balancing is used to distribute the reservoir grid blocks across all processors, by which the average number of grid columns is as equally as possible across all available processors. This method has been proven to be very efficient (Lu, 2000; Lu, 2008). Additionally, a pre-process strategy is adopted in the iterative model to optimize parallel performance. For a given problem, the following issues are considered before starting the simulation:

l Problem analysis: reservoir dimension, mesh size, structure, heterogeneity, wells, time step, included files;

l Numerical model analysis: number of unknowns, explicitness, functionality, fraction of parallel code;

l HPC facility: 64bit/32bit, CPU frequency, memory, network bandwidth, cache, peak performance;

l Linear solver: solver feature, preconditioner, convergence control, default parameters, parallel implementation;

l Communication estimation: elements at the interface, messages that must be passed, message size, C/C++/Fortran interface.

All of above estimations are wrapped in a size file, which is integrated into the executable during program compilation. This executable program is used to simulate just the given problem. In this way, the particular problem is solved with the corresponding executable designed for optimal performance. The effectiveness of this pre-processing is validated by following numerical examples.

4 Numerical example

All of the FIM, IMPES, and iterative models used in the following numerical examples are developed in the Integrated Parallel Accurate Reservoir Simulator (IPARS) framework (Wheeler, 1995; Lu, 2000). The FIM is based on a fully implicit cell-centered finite difference with implicit Euler discretization. It is well tested and produces good solution close to commercial simulators (Lu, 2000; Lu, 2008). The IMPES and iterative models are based on cell-centered finite difference too. The solution of the FIM is used as the standard solution in the research for purpose of comparison and validation. For all the models, an adaptive time step control approach is employed, in which a small time step is taken at the beginning and then increased by a factor until the maximum time step is achieved.

4.1 Example 1: Upscaled SPE 10

This reservoir is modified from the SPE 10 project in which only the upper part of the whole reservoir is used (Christie and Blunt, 2001). The reservoir dimension is 2,200ft ×1,200ft×100ft, with a structured mesh of 55×30×5=8,250. One water injector is located at the center of the reservoir; four producers are located at the four corners of the reservoir. All of these wells are bottom hole pressure specified. The total simulation time is 2,000 days. All three solution methods are used in simulation with various time step controls. The IMPES method does not complete the simulation due to an unexpected oscillation. Timing comparisons are shown in Table 1.

Table 1 Timing comparison for Example 1

Max time step day	Model	Oil balance	Water balance	CPU time sec
2	Iterative	0.99999343	1.00000196	1034.872
	FIM	1.00000003	1.00000004	2060.86
5	Iterative	1.00000301	1.00000817	1729.41
	FIM	1.00000003	1.00000003	3121.56

The solutions of the iterative model are very close to that of the FIM, as reflected by the production rate in Fig. 3. However, it only takes about half of the simulation time required by the FIM when we choose a time step of 2 days. The iterative method is stable and reflects the heterogeneity of the reservoir as clearly as FIM does in Figs. 4 and 5.
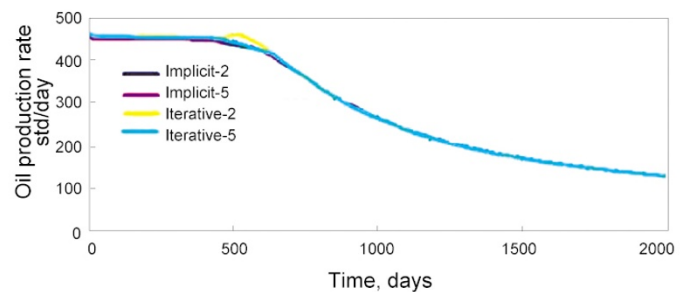

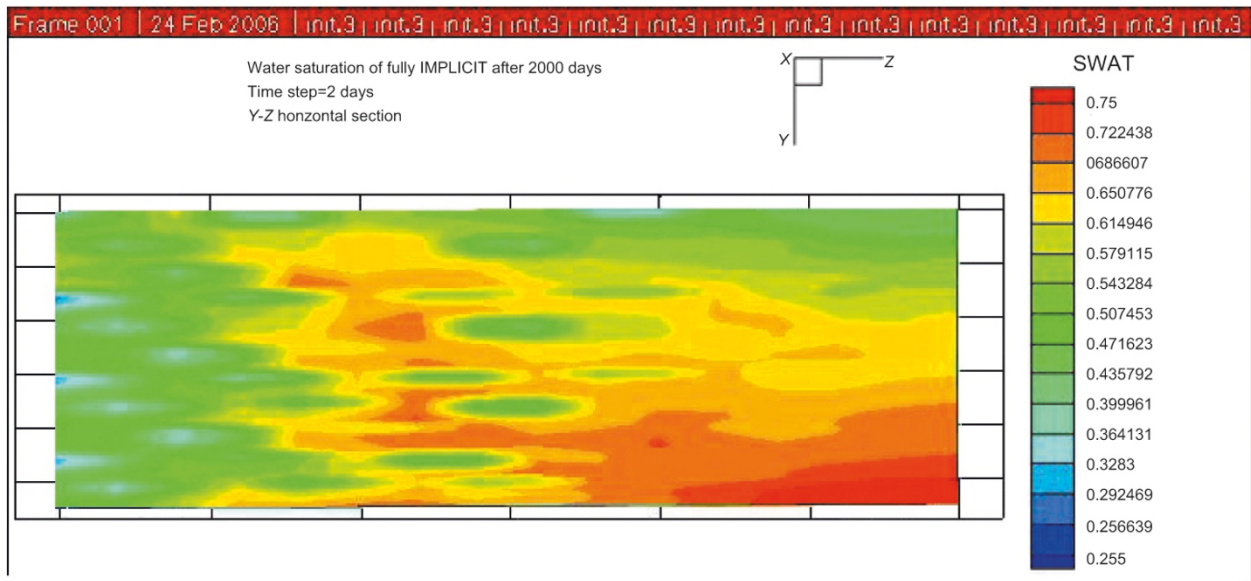**Fig. 3** Oil production rate in Producer 3 of Example 1

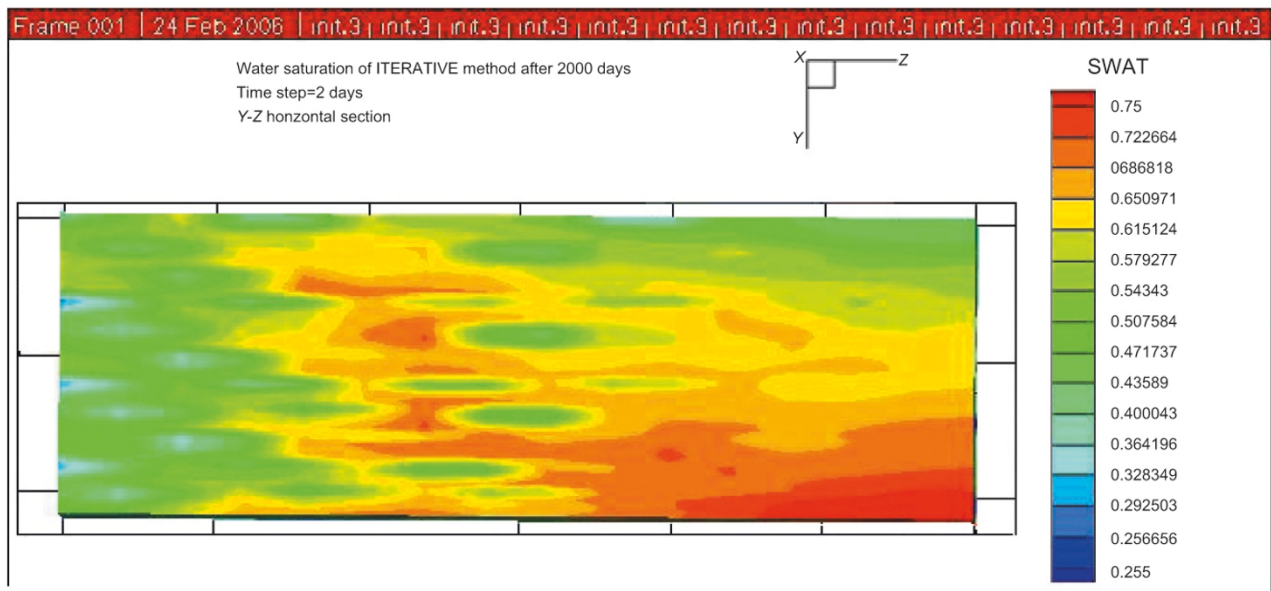**Fig. 4** Saturation profile after 2000 days with the FIM

Fig. 5 Saturation profile after 2000 days with the iterative method

4.2 Example 2: SPE 9 black oil problem

This example comes from the 9th SPE comparative project (Killough, 1995). The reservoir has 9,000 grid elements with high heterogeneity. High permeability zones are distributed in a few layers with poor connection. The PVT properties and a reservoir description can be found in the reference. The reservoir hosts 26 wells, including 1 injector, and all of them are bottom hole pressure specified.

Total simulation time is 900 days, and the maximum time step is set as 5 days. As shown in Table 2, with the iterative method, the total CPU time is 6,114 sec, reduced from 8,017 sec using the FIM. The iterative method obtains very good matches for both the oil production rate and the gas/oil ratio reflected in Figs. 6 and 7, respectively.
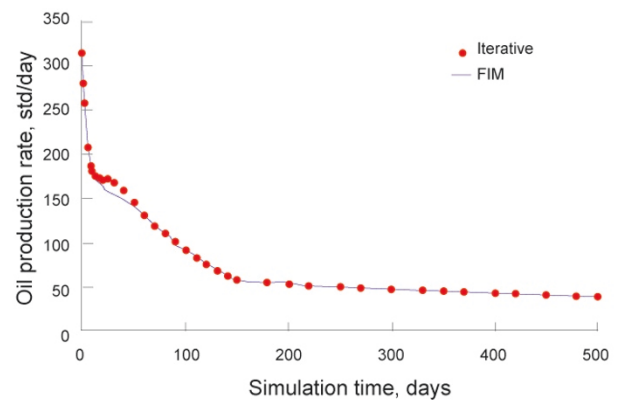

Fig. 6 Oil rate in Producer No.20

Table 2 Timing comparison for Example 2

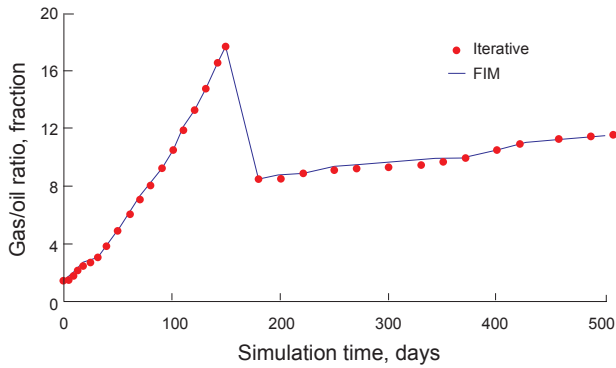| Model	Oil balance	Water balance	Gas balance	CPU time sec
Iterative	1.0003205	0.9992816	1.0006002	6114
FIM	1.0000012	0.9999990	1.0000047	8017

Fig. 7 Gas/oil ratios in Producer No.20

4.3 Example 3: AMV problem on Lonestar

This example is modified from a real reservoir, and all reservoir properties are transformed from a geological model (Lu, 2008). The reservoir has dimensions of 111,952.94 ft long, 130,806 ft wide, and 2,400 ft thick. The grid mesh is 584×550×126=40,471,200. This is a heavy oil reservoir, in which the oil has API as low as 8.5 and very high viscosity. High heterogeneity is observed with wide range of permeability from 10 mD to 10 D. More than 100 wells have been drilled, with total production rate about 150,000 BD. In this simulation, 16 wells, consisting of one injector and 15 producers, are used to represent all the wells by matching the production

rate. All wells are bottom hole pressure specified. Total simulation time is 2,000 days.

This simulation is conducted on the Lonestar cluster from TACC (Texas Advanced Computing Center, Lonestar User's Guide, 2007. www.tacc.utexas.edu/services/userguides/lonestar), with the number of processors ranging from 30 to 500, and the speedup is illustrated in Fig. 8. The red dashed line is the ideal parallel speedup. For this problem with more than 40 million elements, with 50 processors, speedup is reflected by a factor of 47; with 500 processors, speedup is reflected by a factor of 349, which is a 70% parallel efficiency.

The distribution of the total CPU time in these simulations is displayed in Fig. 9. As expected, the linear solver occupies most of the CPU time in all simulations, up to 92% with 50 processors. Parallel communication, initialization, and others occupy the rest of the CPU time. As the number of processors increases, parallel communication takes more and more CPU time, from 2% with 30 processors to 15% with 500 processors.
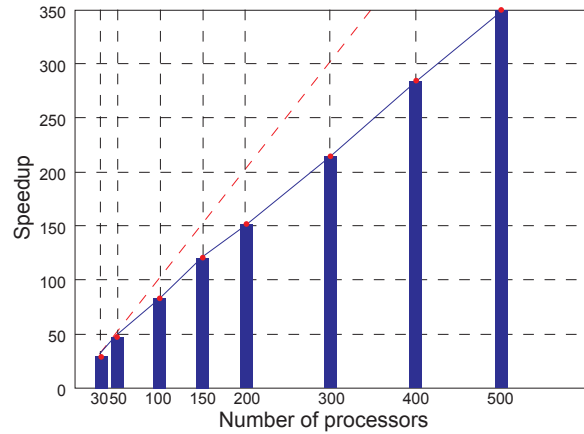

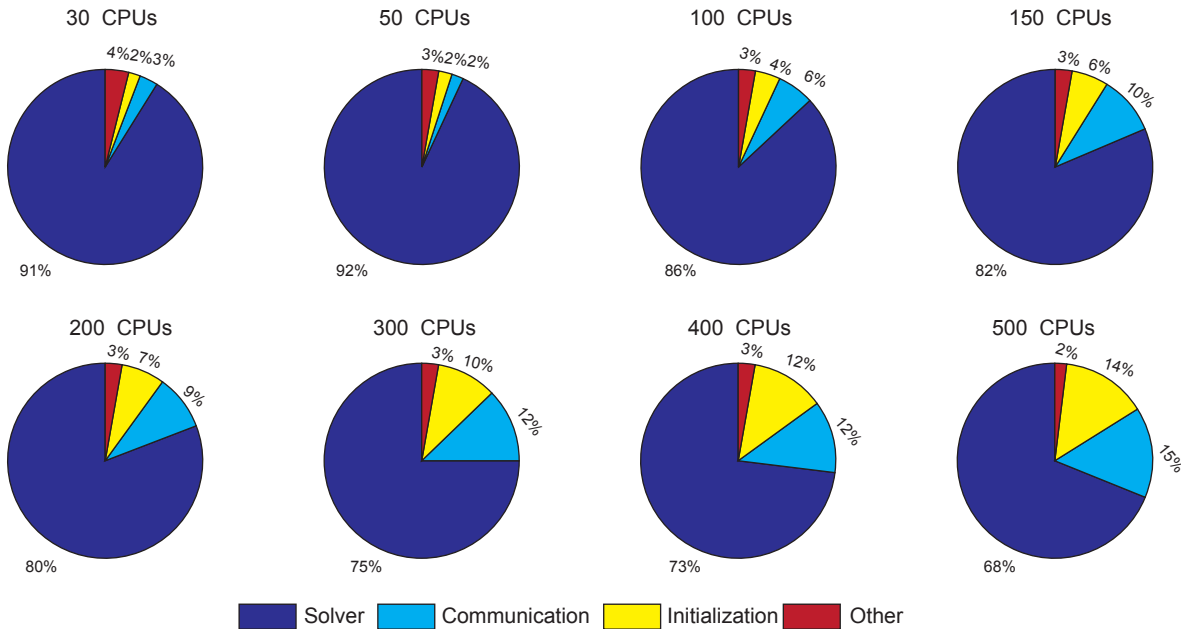**Fig. 8** Parallel scalability on Lonestar with AMV problem

Fig. 9 CPU time distributions on the Lonestar cluster

Fig. 9 also indicates several hints to improve the parallel simulation performance, such as using better networking to cut communication time and adopting better solver to shorten linear solving time.

4.4 Example 4: SPE 10 on Blue Gene

In this case, the SPE 10 problem is solved with the iterative black oil model on IBM's Blue Gene cluster. The reservoir has a mesh of 220×60×85=1,122,000. Again, the PVT properties are modified from the SPE 9 reservoir. Various numbers of processors, up to 2,048, are used in this simulation. As shown in Fig. 10, better parallel efficiency has been achieved with fewer processors. However, even with 2,048 processors, speedup is achieved by a factor of 1,541, which has a parallel efficiency of greater than 75%. The CPU time distribution is shown in Fig. 11. As expected, the parallel communication time increases significantly with increasing number of processors, up to 28% with more than 1,500 processors, which is the main factor that slows down the HPC simulation.
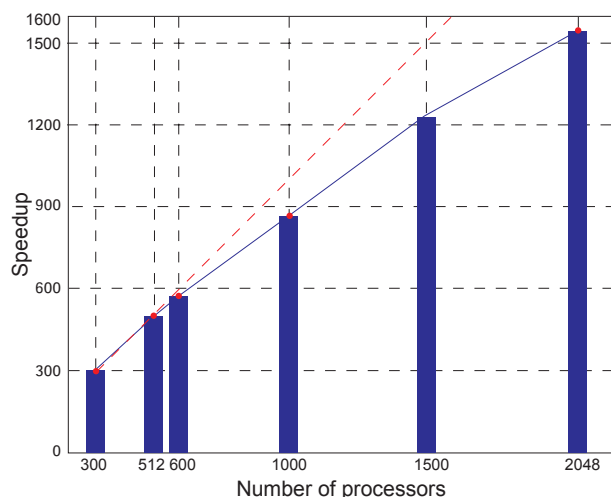

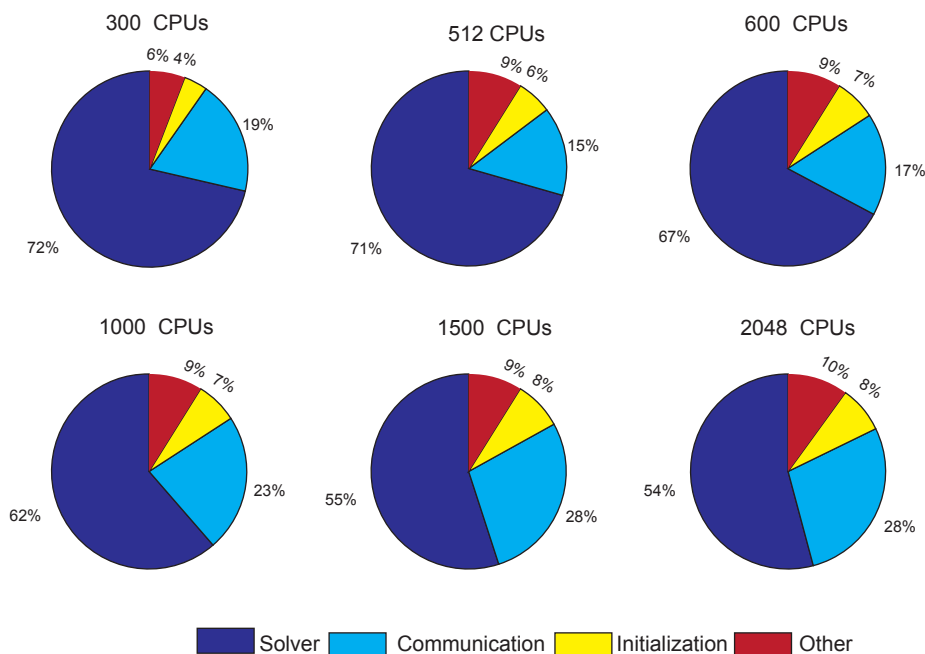**Fig. 10** CPU time distributions on the Blue Gene cluster

Fig. 11 Parallel scalability on Blue Gene with fine grid SPE 10

5 Conclusions

The iterative model is developed with the objective of combining the advantages of the FIM and the IMPES method. The iterative model is faster than the FIM because it decouples the equation system into pressure and concentration equations, and solves them separately. It is more accurate than the IMPES method due to its increasing implicitness. The advantages of the iterative method are shown with several numerical examples, especially its high efficiency on HPC. This study can be summarized as follows:

1) According to examples, the iterative method is more than 30% faster than FIM with acceptable material balance errors. Even better results have been obtained with particular problems.

2) The iterative method is more stable and more accurate than IMPES method, and it allows larger time steps, with very small or no oscillations.

3) Several numerical techniques, such as phase scaling, extrapolation, and a forcing function, have been applied to the iterative model, and they have been proven very efficient in improving the simulation performance.

4) The iterative method has been shown to have very good parallel scalability with high efficiency on different platforms. More than 75% parallel efficiency has been achieved with 2,048 processors. With its parallel scalability, the iterative model can handle very large reservoir problems with millions of elements.

5) The feature of the iterative approach allows coupling of multi-model and multi-physics simulations into one

framework. It is also feasible to use different discretizations for the pressure (e.g., Mimetic or MPFA methods with full tensors) and saturation (e.g., Discontinuous Galerkin) in reservoir simulation.

Acknowledgements

References

Chen Z, Huan G and Ma Y. Computational Methods for Multiphase Flow in Porous Media. Philadelphia: SIAM Comp. Sci. and Eng. 2006

Christie M a and Blunt M J. Tenth SPE comparative solution project: a comparison of upscaling techniques. SPE Reservoir Evaluation & Engineering. 2001. 4(4): 308-317 (SPE 72649)

Coats K H. Reservoir simulation: state of art. JPT. 1982. 34(8): 1633-1642 (SPE 10200)

Dawson C N, Klíe H, Wheeler M F, et al. A parallel, implicit, cell-centered method for two-phase flow with a preconditioned Newton-Krylov solver. Computational Geosciences. 1997. 1(3-4): 215-249

DeBaun D, Byer T, Child P, et al. An extensible architecture for next generation scalable parallel reservoir simulation. SPE Reservoir Simulation Symposium, 31 January-2 February 2005, The Woodlands, Texas (SPE 93274)

Fagin R G and Stewart Jr. C H. A new approach to the two-dimensional multiphase reservoir simulator. SPE Journal. 1966. 6(2): 175-82 (SPE 1188)

Fanchi R J. Principles of Applied Reservoir Simulation (2nd Edition). Gulf Professional Publishing. 2001

Fjerstad P A, Sikandar A S, Cao H, et al. Next generation parallel computing for large-scale reservoir simulation. SPE International Improved Oil Recovery Conference in Asia Pacific, 5-6 December 2005, Kuala Lumpur, Malaysia (SPE 97358)

Fung L S-K, Collins D A, Nghiem L X, et al. An adaptive implicit switching criterion based on numerical stability analysis. SPE Reservoir Engineering. 1998. 4(1): 45-52 (SPE 16003)

IBM Rochester Blue Gene Capacity on Demand Center. User Guide, 2005

Kendall R P, Morrell G O, Peaceman D W, et al. Development of a multiple application reservoir simulator for use on a vector computer. SPE Middle East oil Technical Conference and Exhibition, 14-17 March 1983, Bahrain (SPE 11483)

Killough J E. Ninth SPE comparative solution project: a reexamination of black-oil simulation. SPE Reservoir Simulation Symposium, 12-15 February 1995, San Antonio, Texas (SPE 29110)

Klie H. Krylov-secant Methods for Solving Large Scale Systems of Coupled Nonlinear Parabolic Equations. Ph.D Thesis. Rice University. 1996

Lacroix S, Vassilevski Y, Wheeler J, et al. Iterative solution methods for modeling multiphase flow in porous media fully implicitly. SIAM Journal of Scientific Computing. 2003. 25(3): 905-926

Lu B, Alshaalan T M and Wheeler M F. Iterative coupled reservoir simulation for multiphase flow. SPE Annual Technical Conference and Exhibition, 11-14 November 2007, Anaheim, California, U.S.A. (SPE 110114)

Lu B. Iteratively Coupled Reservoir Simulation for Multiphase Flow in Porous Media. Ph.D Thesis. The University of Texas at Austin. 2008

Lu Q. A Parallel Multiblock/Multiphysics Approach for Multiphase Flow in Porous Media. Ph.D Thesis. The University of Texas at Austin. 2000

Øian M S, Espedal MS, Garrido I, et al. Parallel simulation of multiphase/multicomponent flow models. Lecture Notes in Computational Science and Engineering. 2005. 99-114

Peaceman D W. Fundamentals of Numerical Reservoir Simulation. Elsevier Scientific Publishing Co., Amsterdam. 1977

Shiralkar G S, Fleming G C, Watts J W, et al. Development and field application of a high performance, unstructured simulator with parallel capability. SPE Reservoir Simulation Symposium, 31 January-2 February 2005, The Woodlands, Texas (SPE 93080)

Spillette A G, Hillestad J G and Stone H L. A high-stability sequential solution approach to reservoir simulation. Fall Meeting of the Society of Petroleum Engineers of AIME, 30 September-3 October 1973, Las Vegas, Nevada (SPE 4542)

TACC. Lonestar User's Guide. www.tacc.utexas.edu/services/userguides/lonestar/ 2007

Watts J W. A compositional formulation of the pressure and saturation equations. SPE Reservoir Engineering. 1986. 1(3): 243-252 (SPE 12244)

Watts J W and Shaw J S. A new method for solving the implicit reservoir simulation matrix equation. SPE Reservoir Simulation Symposium, 31 January-2 February 2005, The Woodlands, Texas (SPE 93068)

Wheeler J. IPARS User's Manual. The University of Texas at Austin, 1995

(Edited by Sun Yanhua)