SANDIA REPORT
SAND2004-2918
Unlimited Release
Printed June 2004

Mathematical and Algorithmic Issues in Multiphysics Coupling

Charles M. Stone, Xiuli Gai, Mary F. Wheeler

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy's National Nuclear Security Administration under Contract DE-AC04-94AL85000.

Sandia National Laboratories

TOTAL PAGES: 49

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

Available to DOE and DOE contractors from
U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone: (865)576-8401
Facsimile: (865)576-5728
E-Mail: reports@adonis.osti.gov
Online ordering: http://www.osti.gov/bridge

Available to the public from
U.S. Department of Commerce
National Technical Information Service
5285 Port Royal Rd
Springfield, VA 22161

Telephone: (800)553-6847
Facsimile: (703)605-6900
E-Mail: orders@ntis.fedworld.gov
Online order: http://www.ntis.gov/help/ordermethods.asp?loc=7-4-0#online

SAND2004-2918
Unlimited Release
Printed June 2004

Mathematical and Algorithmic Issues in Multiphysics Coupling

Charles M. Stone
Structural Mechanics Engineering Department
Sandia National Laboratories
P.O. Box 5800
Albuquerque, New Mexico 87185-0376

Xiuli Gai and Mary F. Wheeler
Center for Subsurface Modeling
The University of Texas at Austin
Austin, Texas 78712

Abstract

The modeling of fluid/structure interaction is of growing importance in both energy and environmental applications. Because of the inherent complexity, these problems must be simulated on parallel machines in order to achieve high resolution. The purpose of this research was to investigate techniques for coupling flow and geomechanics in porous media that are suitable for parallel computation. In particular, our main objective was to develop an iterative technique which can be as accurate as a fully coupled model but which allows for robust and efficient coupling of existing complex models (software).

A parallel linear elastic module was developed which was coupled to a three phase three-component black oil model in IPARS (Integrated Parallel Accurate Reservoir Simulator). An iterative de-coupling technique was introduced at each time step. The resulting nonlinear iteration involved solving for displacements and flow sequentially. Rock compressibility was used in the flow model to account for the effect of deformation on the pore volume. Convergence was achieved when the mass balance for each component satisfied a given tolerance. This approach was validated by comparison with a fully coupled approach implemented in the British Petroleum/Amoco ACRES simulator.

Another objective of this work was to develop an efficient parallel solver for the elasticity equations. A preconditioned conjugate gradient solver was implemented to solve the algebraic system arising from tensor product linear Galerkin approximations for the displacements. Three preconditioners were developed: LSOR (line successive over-relaxation), block Jacobi, and agglomeration multi-grid. The latter approach involved coarsening the 3D system to 2D and using LSOR as a smoother that is followed by applying geometric multi-grid with SOR (successive over-relaxation) as a smoother. Preliminary tests on a 64-node Beowulf cluster at CSM indicate that the agglomeration multi-grid approach is robust and efficient.

Intentionally Left Blank

Table of Contents

Intentionally Left Blank

Introduction

The modeling of fluid/structure interaction is of growing importance in both energy and environmental applications. Computational challenges in solving a coupled system of equations include: 1) high computational cost; 2) selection of a coupling method to optimize run times and accuracy for different type of physical applications; 3) solvability and stability of a discretized linear system. The purpose of this research was to investigate techniques for coupling flow and geomechanics in porous media that are suitable for parallel computation. In particular, our main objective was to develop an iterative technique which can be as accurate as a fully coupled model but which allows for robust and efficient coupling of existing complex models (software). Stability issues will be our primary goal in the future.

In coupled geomechanics and reservoir modeling, the finite element discretization of the force balance equation leads to very large linear systems, whose solution is both time and memory consuming. ICCG (Incomplete Cholesky Factorized Conjugate Gradient) is a popular technique for solving for displacements, but the technique is limited to about 60,000 nodal points on desktop machines. Most large 3D field scale problems will have to be run on parallel machines. In this project we use a reduced-communication, super coarsening multigrid method that can be used in a combinative way with other domain decomposition-based preconditioners to achieve faster convergence with high parallel scalability. A preliminary test case of 1.5 million grid blocks with up to 59 processors shows a parallel efficiency of above 90%.

The second computational issue we addressed is the efficiency and accuracy of different operator splitting techniques. We compare three methods for coupling multiphase porous flow and geomechanics. Sample simulations are constructed to highlight the similarities and differences in the techniques. One technique uses an explicit algorithm to couple porous flow and displacements where flow calculations are performed every time step and displacements are calculated only during selected time steps. A second technique uses an iteratively coupled algorithm where flow calculations and displacement calculations are performed sequentially for nonlinear iterations during time steps. The third technique uses a fully coupled approach where the program's linear solver must solve simultaneously for fluid flow variables and displacement variables. Comparison problems are run for both single-phase and three-phase flow problems involving poroelastic deformations.

Among these coupling schemes, the iterative approach is more attractive due to the fact that 1) it is accurate and stable; 2) it preserves software's modularity; 3) it provides a straightforward way to couple an existing porous flow simulator with an existing geomechanics simulator. The primary drawback to this method is that the calculations may display a first order convergence rate in the nonlinear iterations. However we reformulate the classical iterative method in a more general framework and show that the method can be viewed as performing one preconditioned Richardson iteration on a fully coupled system. The rock compressibility term used in a flow model is nothing but a preconditioner for the Schur complement pressure equation. We gain two folds by this interpretation. First an iterative coupling implementation can be more easily adapted to a fully coupled scheme. In this setup, solving the two coupled field equations simultaneously is from an implementation point of view not more complicated than using a classical operator splitting method repeatedly. This is worth noticing because people tend to implement and use the loosely coupled or iteratively coupled methods because they are numerically more explicit compared to a fully coupled scheme. Second it provides us the possibility to analyze convergence behavior of the iterative coupling method. The pressure mass matrix scaled by rock compressibility term can also be generalized as a preconditioner to other iterative linear solvers in a fully coupled scheme.

Two papers written in support of this work are presented the appendices. Appendix A is a paper published in the proceedings of the Society of Petroleum Engineers Reservoir Simulation Symposium that compares three techniques for coupling multiphase porous flow and geomechanics. Appendix B contains a second paper published in the same proceedings of the Society of Petroleum Engineers Reservoir Simulation Symposium that looks at issues related to the parallelization of software for coupled geomechanics and reservoir modeling.

Acknowledgments

Sandia National Laboratories is a multi-program laboratory operated by Sandia Corporation, a Lockheed Martin company, for the U.S. Department of Energy under contract No. DE-AC04-94AL8500. This work was supported by the Laboratory Directed Research and Development Program at Sandia from October 1, 2001 to September 30, 2003. The authors would also like to acknowledge the support of the National Science Foundation (NSF), Natural Gas and Oil Technology Partnership, and the Industrial Affiliates of the Center for Subsurface Modeling at the University of Texas at Austin.

Appendix A - Rick H. Dean, Xiuli Gai, Charles M. Stone, and Susan E. Minkoff, " A Comparison of Techniques for Coupling Porous Flow and Geomechanics", presented at the SPE Reservoir Simulation Symposium, Houston, Texas, U.S.A., 3-5 February 2003, paper SPE 79709.

Intentionally Left Blank

A Comparison of Techniques for Coupling Porous Flow and Geomechanics

Rick H. Dean, SPE, and Xiuli Gai, University of Texas at Austin, Charles M. Stone, Sandia National Laboratories, and Susan E. Minkoff, University of Maryland, Baltimore County

Abstract

This paper compares three techniques for coupling multiphase porous flow and geomechanics. Sample simulations are presented to highlight the similarities and differences in the techniques. One technique uses an explicit algorithm to couple porous flow and displacements where flow calculations are performed every time step and displacements are calculated only during selected time steps. A second technique uses an iteratively coupled algorithm where flow calculations and displacement calculations are performed sequentially for nonlinear iterations during time steps. The third technique uses a fully coupled approach where the program's linear solver must solve simultaneously for fluid flow variables and displacement variables. The techniques for coupling porous flow with displacements are described, and comparison problems are presented for single-phase and three-phase flow problems involving poroelastic deformations. All problems in this paper are described in detail so the results presented here may be used for comparison with other geomechanical/porous flow simulators.

Introduction

Many applications in the petroleum industry require both an understanding of the porous flow of reservoir fluids and an understanding of reservoir stresses and displacements. Examples of such processes include subsidence, compaction drive, wellbore stability, sand production, cavity generation, high-pressure breakdown, well surging, thermal fracturing, fault activation, and reservoir failure involving pore collapse or solids disposal. It would be useful to compare porous flow/geomechanics techniques for all of these processes, since some of these processes involve a stronger coupling between porous flow and geomechanics than others. However, this paper looks at a subset of these processes and compares three coupling techniques for problems involving subsidence and compaction drive. All of the sample problems presented in this paper assume that the reservoir absolute permeabilities are constant during a run. Displacements influence fluid flow through calculation of pore volumes and fluid pressures enter the displacement calculations through the poroelastic constitutive equations.

Several authors have presented formulations for modeling poroelastic, multiphase flow. Settari and Walters[1] discuss the different methods that have been used to combine poroelastic calculations with porous flow calculations. They categorize these different methods of coupling poroelastic calculations with porous flow calculations as decoupled,[1] explicitly coupled, iteratively coupled, and fully coupled. The techniques discussed in this paper are explicitly coupled, iteratively coupled, and fully coupled.

For an explicitly coupled approach,[2-4] a simulator performs computations for multiphase porous flow each time step and performs geomechanical calculations for displacements during selected time steps. The frequency of geomechanical updates is driven by the magnitude of the pore volume changes during the time steps. If the pore volumes change slowly during time steps then few geomechanical updates are required. The ability to perform geomechanical calculations for selected time steps is a very attractive feature of the explicitly coupled approach because a major portion of the computational time for a porous flow/geomechanics run is often spent in calculating displacements. Another attractive feature of the explicitly coupled approach is that it is very straightforward to use this technique to couple an existing porous flow simulator with an existing geomechanics simulator. One shortcoming of the explicitly coupled approach is that the explicit nature of the coupling can impose time step restrictions on runs because of concerns about stability and accuracy. However, for many subsidence problems the fluid flow calculations require time steps that are smaller than those imposed by the explicit coupling calculations.

For the iteratively coupled approach, multiphase porous flow and displacements are coupled through the nonlinear iterations for each time step. During each nonlinear iteration, a simulator performs computations sequentially for multiphase porous flow and for displacements. The flow and displacement calculations are then coupled through calculations of pore volumes at the end of each nonlinear iteration. An iteratively coupled approach will produce the same results as a fully coupled approach if both techniques use sufficiently tight convergence tolerances for iterations. Settari and Mourits,[5] and Fung, et al.[6] present examples of the iteratively coupled approach for multiphase flow. The primary attraction of the iteratively coupled approach is that it is very straightforward to couple an existing porous flow simulator with an existing geomechanics simulator. The primary drawback to the iteratively coupled approach is that the calculations may display a first order convergence rate in the nonlinear iterations and therefore may require a large number of iterations for difficult problems.

For the fully coupled approach, porous flow and displacement calculations are performed together, and the program's linear solver must handle both fluid flow variables and displacement variables. Tortike and Farouq Ali,[7] Li and Zienkiewicz,[8] and Lewis and Sukirman[9] have presented formulations of the fully coupled approach for poroelastic, multiphase flow. The primary attraction of the fully coupled approach is that it is the most stable approach of the three techniques and preserves second order convergence of nonlinear iterations. Drawbacks to the fully coupled approach are: it may be difficult to couple existing porous flow simulators and geomechanics simulators, it requires more code development than other techniques, and it can be slower than the explicit and iterative techniques on some problems.

The three techniques for coupling porous flow and geomechanics were incorporated into the same program so differences in the calculations could be attributed to the different techniques for coupling. If one were to compare three different programs each using a different technique for coupling, then it might be difficult to differentiate between differences due to coupling and differences due to basic algorithms in the separate programs. Comparison problems are presented for single-phase and three-phase flow problems involving poroelastic deformations. All techniques should produce the same results when using small time steps and tight convergence tolerances, so the choice between techniques is determined by ease of implementation, program availability, numerical stability, and computational efficiency.

A short review of the equations coupling porous flow and deformations is presented, followed by details of the algorithm for explicit coupling. Four problems are then presented and the results are compared using the three techniques. The first two problems are simple single-phase depletion problems that illustrate the role that stress and displacement boundary conditions play in porous flow calculations. The third problem is a single-phase depletion example where a soft reservoir is contained within a stiff nonpay region. The final problem is a three-phase, black-oil, five-spot pattern with a production well in one corner of the grid and a water injection well in the opposite corner. The coupling between geomechanics and fluid flow is fairly straightforward in problems 1, 2 and 4 and pressure histories for these runs can be reproduced by typical reservoir simulators with proper choices of compressibilities; however, problem 3 exhibits

geomechanical effects that cannot be seen in reservoir simulations that do not include geomechanical calculations.

Coupled Flow and Deformation

For the problems in this paper, displacements enter the fluid flow equations through the calculation of reservoir pore volumes, and fluid pressures enter the displacement calculations through the stress/strain constitutive equations. A typical porous flow simulator expresses the pore volume for a grid block as

$$V_p = V_p^o\left[1 + c_r\left(p - p_o\right)\right] \quad \text{.....(1)}$$

where p is the fluid pressure and c_r is a compressibility-like term that must be entered by the user as part of the input data. However, for linear poroelastic calculations the pore volume for infinitesimal displacements may be expressed as

$$V_p = V_b^o\left[\varphi_o + \alpha\varepsilon_{kk} + \frac{1}{M}\left(p - p_o\right)\right] \quad \text{.....(2)}$$

where α and $1/M$ are Biot's parameters and Eq. 2 assumes expansion is positive. For the comparisons in this paper, α and $1/M$ are set equal to one and zero, respectively. For this choice of Biot's parameters, Eq. 1, a typical equation for flow simulators, expresses the pore volume in terms of the fluid pressure while Eq. 2 expresses the pore volume in terms of bulk strains, ε_{kk}. Flow simulators that are coupled to geomechanics programs may use an equation similar to Eq. 1 to approximate pore volume changes for the flow calculations and use an equation similar to Eq. 2 to calculate corrected pore volumes based upon reservoir deformations.

Logic that couples flow simulators to geomechanics programs must somehow account for the discrepancies between Eq. 1 and Eq. 2. Many coupling techniques will normally use a c_r term similar to that in Eq. 1 to enhance the coupling between flow calculations and displacement calculations. For explicitly coupled techniques, modified forms of Eq. 1 may be used to calculate pore volumes for those time steps where geomechanical updates are not performed. For iteratively coupled techniques, a c_r term may be included in the Jacobian for the flow equations, but Eq. 2 is always used to calculate pore volumes. For fully coupled techniques, a c_r term may be used in a preconditioning matrix for the flow equations when solving the linear system for flow variables and displacement variables.

The fluid pressure enters the deformation calculations through the linear poroelastic constitutive equation

$$\sigma_{ij} = \sigma_{ij}^o + \lambda\varepsilon_{kk}\delta_{ij} + 2\mu\varepsilon_{ij} - \alpha\left(p - p_o\right)\delta_{ij} \quad \text{.....(3)}$$

where tensile stresses are positive in Eq. 3. For the three-phase simulation included in this paper, the oil-phase pressure is used in Eqs. 1-3. For explicitly coupled and iteratively coupled techniques, the fluid pressure in Eq. 3 may be included in the equilibrium equation as a forcing function similar to the effects of a gravity head term. For a fully coupled technique, the fluid pressure in Eq. 3 generates a coefficient that must be included in the Jacobian for the system of flow variables and displacement variables.

Explicit Coupling

The coupling algorithm for the explicit technique is described in more detail here because the algorithm uses both Eq. 2 and a modified form of Eq. 1 to calculate pore volumes for grid blocks during simulations. The iteratively coupled and fully coupled techniques may use a c_r term for the Jacobian or in a preconditioner to accelerate iterative calculations, but never actually use Eq. 1 to calculate pore volumes.

The explicit coupling algorithm allows a program to perform geomechanical calculations on a time scale that is different from the time scale for the flow calculations. This is very useful for subsidence problems because a large portion of the computational time in a simulation can be spent in performing geomechanical calculations. For many problems, fluid fronts may propagate or well changes may occur

over very short time frames while subsidence may progress very slowly throughout the course of a simulation.

One can use Eqs. 1 and 2 to develop an algorithm for determining how often geomechanical calculations must be performed during a simulation. Let V_p^m be the pore volume for a grid block at time step m that was calculated using the geomechanical expression in Eq. 2. If the last geomechanical calculation was done for time step m, then for time step n > m, the pore volume in a grid block may be approximated by

$$\widetilde{V}_p^n = V_p^m + c_r^{est} V_p^o \left(p^n - p^m\right) \quad \text{.....(4)}$$

where p^n and p^m are the pressures for the grid block at time steps n and m, respectively. One may replace the V_p^o term in Eq. 4 by V_p^m; however, this does not change the accuracy of the approximation since terms at step m are constant in Eq. 4. Using V_p^m in place of V_p^o merely modifies the formula that one would develop for estimating values for the compressibility, c_r^{est}, at step m. Several techniques may be used to estimate compressibilities during a simulation. One approach derives analytical estimates of compressibilities using simple assumptions concerning stress and strain variations for a problem while a second approach uses pressure and pore volume changes between previous geomechanical updates during a simulation to estimate compressibilities. A third approach might calculate numerical estimates from the geomechanical equilibrium equations by calculating how variations in fluid pressures affect displacements. When estimating compressibilities, one may need to establish bounds for these estimates because values that are too large generate significant numerical errors and values that are too small give rise to oscillations or instabilities.

The explicitly coupled simulations in this paper use compressibilities in Eq. 4 that are derived from simple assumptions concerning stress and strain variations. For example, when a reservoir is deforming in the vertical direction and horizontal displacements are zero, uniaxial strain, then Eqs. 2 and 3 become approximately $\Delta V_p = V_b^o \Delta\varepsilon_{zz}$ and $\Delta\varepsilon_{zz} = \Delta p/(\lambda + 2\mu)$ when □=1 and $1/M = 0$. So Eq. 2 and 3 may be combined as $\Delta V_p = V_b^o \Delta p/(\lambda + 2\mu)$. But Eq. 4 may be written as $\Delta\widetilde{V}_p^n = c_r^{est} \varphi_o V_b^o \Delta p$, which produces an estimate for c_r^{est} that is $[(\lambda + 2\mu)\varphi_o]^{-1}$. This may also be written in terms of the elastic modulus and Poisson's ratio as $(1+\nu)(1-2\nu)/[(1-\nu)\varphi_o E]$.

If one uses the geomechanical expression in Eq. 2 to calculate the pore volume V_p^n at step n, then one can compare V_p^n with $\widetilde{V}_p^n$ to determine errors in using Eq. 4 in place of Eq. 2. The relative error in pore volume for step n may be written as

$$E_{rel} = abs\left(\frac{\widetilde{V}_p^n - V_p^n}{V_p^n}\right) \quad \text{.....(5)}$$

During an explicitly coupled simulation, one does not have a value of E_{rel} for every time step, but only has values for those steps where geomechanical calculations are performed. It is natural to assume that the error in Eq. 5 is related to the relative change in pore volume since the last geomechanical update at step m, where the relative change in pore volume between steps m and n is approximated by

$$\left(\Delta V_p\right)_{rel} = abs\left(\frac{\widetilde{V}_p^n - V_p^m}{V_p^m}\right) \quad \text{.....(6)}$$

If one assumes that E_{rel} is proportional to $(\Delta V_p)_{rel}$ for those time steps where geomechanical calculations are not performed, then one can implement an algorithm that determines when displacements must be updated. One may estimate the parameter □ in $E_{rel} \approx \beta(\Delta V_p)_{rel}$ as

$$\beta = \frac{E_{rel}}{(\Delta V_p)_{rel}} \quad \text{.....(7)}$$

where values of E_{rel} and $(\Delta V_p)_{rel}$ are determined from the two most recent time steps that included geomechanical updates. Prescribing a tolerance for E_{rel}, one may then use $\beta(\Delta V_p)_{rel}$ to determine when geomechanical updates need to be performed during subsequent time steps. The algorithm above is concerned with errors in pore volumes; however, similar logic may be applied to permeabilities if permeabilities change during a simulation. For the problems in this paper, the tolerance for E_{rel} is set to 0.001. The program also has options to specify updates for displacements after a prescribed number of time steps or after a presecribed pressure change since the last update, but neither option was used for the problems in this paper.

Program Description

The three techniques for coupling flow and geomechanics are available in the program ACRES[10] (ARCO's Comprehensive REservoir Simulator). The program uses masses and a fluid pressure as primary variables for the flow equations and displacements as primary variables for deformations. The program contains IMPEM (IMplicit Pressure Explicit Mass) and implicit time stepping algorithms; however, all coupled runs are currently restricted to using the IMPEM technique for the flow calculations. The program uses finite differences (mixed finite elements with piecewise constant pressures) for the flow variables and finite elements for deformation variables. The program is capable of performing poroelastic and poroplastic calculations for black-oil and fully compositional applications. The displacement calculations use trilinear basis functions with eight Gaussian integration nodes for forming the stiffness matrix and a single integration node for integrating the fluid pressure in the equilibrium equation.

Comparison Problems

Four problems are used to compare the three techniques for coupling porous flow and geomechanics. The first two problems are simple single-phase depletion problems that illustrate the role that stress and displacement boundary conditions play in porous flow calculations. The third problem is a single-phase depletion example where a soft reservoir is contained within a stiff nonpay region. The final problem is a three-phase, black-oil, five-spot pattern with a production well in one corner of the grid and a water injection well in the opposite corner. Biot's parameters □ and $1/M$ are set equal to one and zero, respectively, for all problems. All stresses described below are compressive and represent total stresses for the systems (include forces for fluid and solid).

All problems in this paper use a nonlinear convergence tolerance of 0.01 for volume errors, and a relative residual reduction tolerance of 0.01 for linear iterations, unless stated otherwise. The volume error is expressed as $(V_f-V_p)/V_p$ and the maximum is calculated for all grid blocks, where V_f and V_p are the fluid and pore volumes for a cell, respectively. All computing times presented in this paper are for a 700 Mhz Intel Mobile Pentium III.

Problems 1 and 2. Problems 1 and 2 are identical in description except problem 1 enforces zero displacement boundary conditions at the vertical faces of the grid and problem 2 applies constant horizontal stresses at the vertical faces of the grid. **Figs. 1a and 1b** show the stress and displacement boundary conditions for the two problems.

The grid is 11 x 11 x 10 with □x=□y=200 ft in the horizontal directions and □z=20 ft in the vertical direction. The top of the grid is at a depth of 6000 ft, the initial in situ reservoir porosity is 20%, and the reservoir permeabilities are 50 md and 5 md in the horizontal and vertical directions, respectively. The fluid

is single phase with a formation volume factor of 1.0, a viscosity of 1 cp, a fluid density of 62.4 lbm/ft^3, and zero fluid compressibility. The initial fluid pressure is 3000 psi at a depth of 6000 ft.

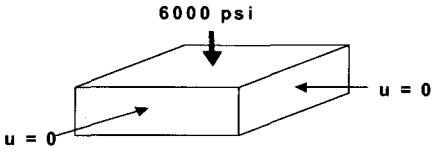

Fig. 1a – Constrained displacements for problem 1

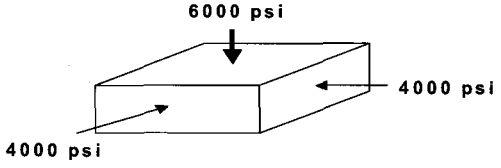

Fig. 1b – Unconstrained displacements for problem 2

The elastic modulus is 1 x 10^4 psi, Poisson's ratio is 0.3, and the initial in situ solid density (solid material without pores) is 2.7 gm/cm^3. Initial horizontal stresses are 4000 psi over the entire reservoir depth while the initial vertical stress is 6000 psi at 6000 ft with a vertical stress gradient of 1.0231 psi/ft throughout the reservoir. The bottom of the grid has a zero vertical displacement constraint and all faces of the grid have zero tangential stresses. Both problems apply a normal stress of 6000 psi at the top of the grid while problem 1 enforces zero normal displacements at the four vertical faces of the grid and problem 2 applies a normal stress of 4000 psi at these same faces. Assuming uniaxial strain behavior for problem 1 and constant total stresses for problem 2, the explicitly coupled simulations in this paper use constant values of 3.71 x 10^{-4} psi^{-1} and 6.00 x 10^{-4} psi^{-1} for the compressibility in Eq. 4.

A vertical well with a wellbore radius of 0.25 ft is completed in the center of the pattern in all ten layers of the grid, cells (6,6,1-10). The well is produced at a rate of 15,000 b/d for 500 days with a time step size of 10 days. No flow boundary conditions are assumed for the fluid at all faces of the grid.

Fig. 2 shows average pore-volume-weighted reservoir pressures for problems 1 and 2 using the three different techniques. All techniques produced nearly identical results for each problem. Fig. 2 shows how geomechanical stress or displacement boundary conditions influence the pressure response in the reservoir. Problem 2 shows much less pressure drop than problem 1 because of the support provided by the constant stress boundary conditions on the sides of the reservoir.

The runtime information for problems 1 and 2 are displayed in Tables 1 and 2. The column for iterations is the number of nonlinear iterations during a simulation. The explicitly coupled technique is faster than the other two techniques for this problem because it performs a small number of updates for the displacements when using an E_{rel} tolerance of 0.001. The explicitly coupled technique performs 18 and 15 updates for the displacements for problems 1 and 2, respectively.

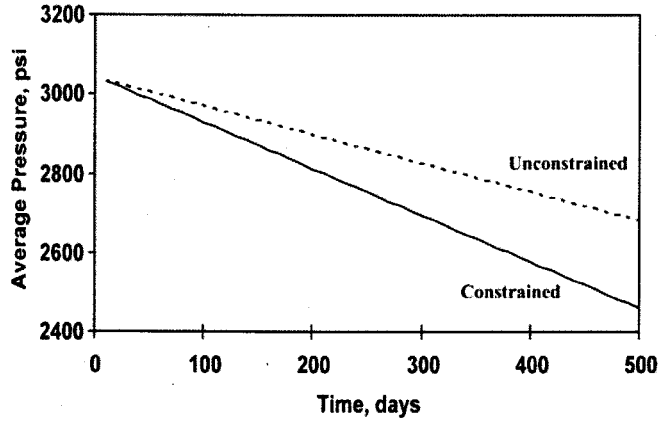

Fig. 2 – Average pressures for problems 1 and 2

Technique	*CPU Time*	*Time Steps*	*Iterations*
Explicit	**8.0 seconds**	**50**	**53**
Iterative	**10.7**	**50**	**51**
Full	**13.3**	**50**	**51**

Table 1 – Runtime information for problem 1

Technique	*CPU Time*	*Time Steps*	*Iterations*
Explicit	**7.8 seconds**	**50**	**53**
Iterative	**10.7**	**50**	**52**
Full	**12.4**	**50**	**51**

Table 2 – Runtime information for problem 2

Fig. 3 shows the subsidence at the top of the reservoir at the well for problems 1 and 2. The two problems produce similar displacements at early times, but the problems deviate substantially at later times. The change in subsidence in Fig. 3 is not a linear function of the average pressure drop in Fig. 2 until later in the run when a pseudo-steady state is reached for the pressure behavior.

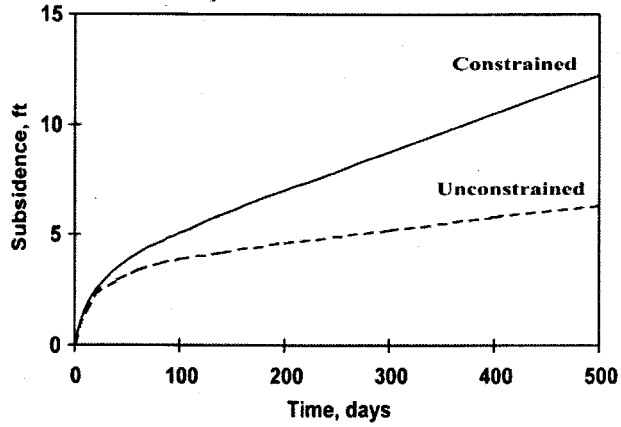

Fig. 3 – Subsidence for problems 1 and 2

The total subsidence for problem 1 is 12.2 ft after 500 days. This corresponds to an average vertical strain of 6.1%, which is very large considering that the calculations are based upon infinitesimal strain assumptions. Even though the pressures in Fig. 2 are based upon calculations using infinitesimal strains; it is expected that the results should not change substantially if the calculations are repeated using a finite

strain formulation. Based upon a simple uniaxial strain analysis, a finite strain simulation should predict a final average pressure for the constrained case that is about 10 psi larger than the result shown in Fig. 2.

Problem 3. Problem 3 is modeled after a problem presented by M. Gutierrez and R.W. Lewis.[11] Problem 3 includes a soft productive reservoir that is contained within a stiff nonpay region as shown in **Fig. 4**. Problem 3 displays a geomechanical effect at the boundary of the reservoir that cannot be seen in reservoir simulations that do not include geomechanical calculations. For this problem, geomechanical effects cause the fluid pressures to increase at the boundary of the reservoir during the initial stages of depletion.

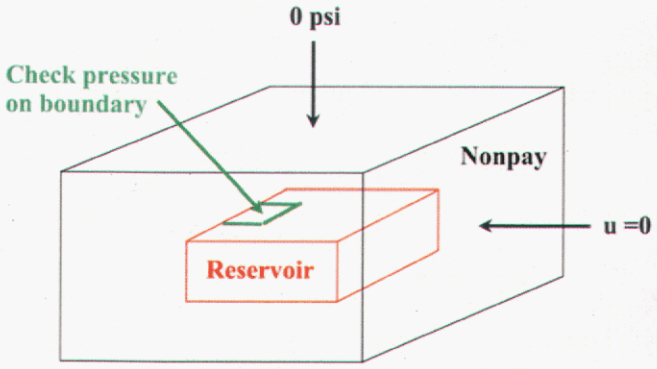

Fig. 4 – Reservoir and nonpay regions for problem 3

The grid is 21 x 21 x 12 and includes both the reservoir and nonpay regions. Grid block lengths in the x-direction are 4000 ft each for the first 5 grid blocks, 2000 ft each for the next 11 grid blocks, and 4000 ft each for the last 5 grid blocks. Grid block lengths for the y-direction are half the corresponding values in the x-direction. The top of the grid is at a depth of 0 ft and the thicknesses in the vertical direction are 4000, 3000, 2000, 800, and 200 ft for the first 5 layers that represent the overburden. The next five layers have thicknesses of 50 ft each and represent the reservoir. The last two layers have thicknesses of 100 ft each and represent the underburden. The horizontal and vertical permeabilities are 100 and 10 md, respectively, in the reservoir, cells (6-16,6-16,6-10). Permeabilities are zero in the nonpay region. The initial in situ porosity is 25% in both the reservoir and nonpay regions.

The fluid is single phase with a formation volume factor of 1.0 at 14.7 psi, a viscosity of 1 cp, a fluid density of 62.4 lbm/ft^3 at 14.7 psi, and fluid compressibility of 3 x 10^{-6} psi^{-1}. A nonzero fluid compressibility is used for this problem because a zero fluid compressibility makes the porous solid incompressible in the nonpay region. The initial fluid pressure is 14.7 psi at the surface.

The elastic moduli are 1 x 10^4 psi in the reservoir and 1 x 10^6 psi in the nonpay region, Poisson's ratio is 0.25 everywhere, and the initial in situ solid density (solid material without pores) is 2.7 gm/cm^3. The initial vertical stress is 0 psi at the surface with a vertical stress gradient of 0.9869 psi/ft throughout the grid, and initial horizontal stresses are equal to half of the vertical stress. The bottom and sides of the grid have zero normal displacement constraints and all faces of the grid have zero tangential stresses. Assuming uniaxial strain behavior for this problem, the explicitly coupled simulation uses values of 3.33 x 10^{-4} psi^{-1} and 3.33 x 10^{-6} psi^{-1} for the compressibility in Eq. 4 in the reservoir and nonpay regions, respectively.

A vertical well with a wellbore radius of 0.25 ft is completed in the center of the reservoir in all five layers, cells (11,11,6-10). The well is produced at a rate of 50,000 stb/d for 4000 days with a time step size of 20 days for the first 400 days, followed by time steps of 200 days stopping at 4000 days. Smaller time steps are taken at the beginning of the run to produce an accurate solution for the pressure increase at the reservoir boundary. Iteratively coupled and fully coupled techniques should be able to produce accurate results using the time steps specified for this problem, but explicitly coupled techniques may require time steps that are smaller than 20 days because of time discretization errors that arise due to the explicit coupling.

Fig. 5 shows average pore-volume-weighted pressures in the reservoir (excluding nonpay region) using the three different techniques. The three techniques produce significantly different results in Fig. 5 after the time step size increases from 20 days to 200 days.

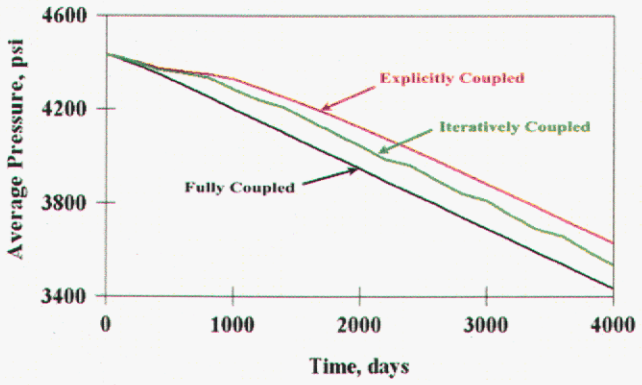

Fig. 5 – Average reservoir pressures for problem 3

The three techniques also predict large differences in pressures at the boundary of the reservoir at early times. The iteratively coupled technique requires a tighter tolerance on the nonlinear iterations and the explicitly coupled technique requires smaller time steps to reproduce the fully coupled results. For this problem, the iteratively coupled technique requires a nonlinear volume error tolerance of 0.0001 and the explicitly coupled technique requires a time step size of about one day. One can improve the explicitly coupled results by using a smaller value of estimated compressibility for this problem; however, values that are too small will produce oscillations in well pressures.

Fig. 6 shows the subsidence at the top of the reservoir and at the surface for all three techniques. The explicitly coupled results included in Fig. 6 use a time step size of one day for the simulation. The original explicitly coupled results using time step sizes of 20 days and 200 days did not agree well with the results in Fig. 6 predicting a final subsidence of 6.47 ft at the top of the reservoir. The final subsidence in Fig. 6 at the top of the reservoir is 7.76 ft.

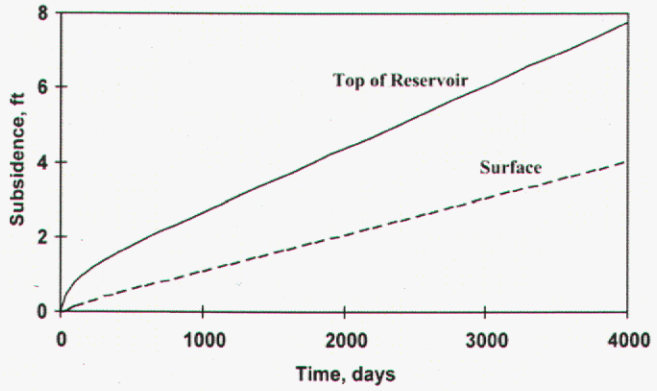

Fig. 6 – Subsidence for problem 3

Fig. 7 shows the pressure behavior at the boundary of the reservoir in cell (6,11,6). Initially the reservoir pressure increases as the reservoir is depleted because some of the vertical load that was supported at the center of the reservoir is transferred to the edges of the reservoir. This pressure increase cannot be observed in a reservoir depletion problem that does not include geomechanical calculations. The iteratively coupled results in Fig. 7 use a volume error tolerance of 0.0001, and the explicitly coupled results use a time step size of one day updating displacements every time step.

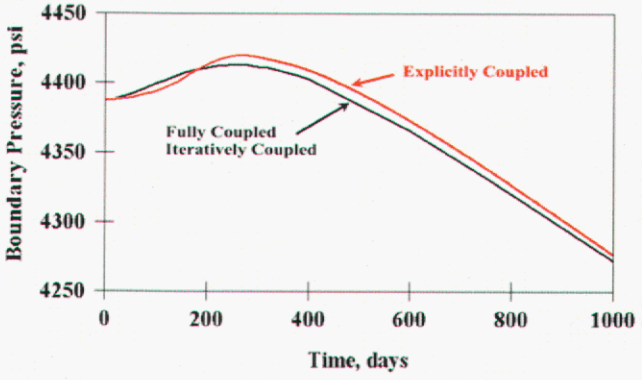

Fig. 7 – Pressure at boundary of reservoir for problem 3

The runtime results for problem 3 are shown in Table 3. The explicitly coupled technique is much slower than the other two techniques for this problem because it requires much smaller time step sizes. The fully coupled and iteratively coupled techniques also exhibit time discretization errors, but time discretization errors play a much larger role for the explicitly coupled technique. The iteratively coupled technique is slower than the fully coupled technique because the iteratively coupled technique requires a large number of nonlinear iterations for convergence. Also, the iteratively coupled technique exhibits only a first order rate of convergence for the nonlinear iterations because of the sequential nature of updating the flow and displacement equations.

Technique	*CPU Time*	*Time Steps*	*Iterations*
Explicit	**51.8 minutes**	**4000**	**4000**
Iterative	**6.9**	**38**	**449**
Full	**4.3**	**38**	**38**

Table 3 – Runtime information for problem 3

Problem 4. Problem 4 is a three-phase, five-spot with a water injection well in one corner of the grid and a production well in the diagonally opposite corner of the grid. The production rate is larger than the injection rate so the reservoir pressure decreases throughout the simulation.

The grid for problem 4 is displayed in **Fig. 8** showing water saturations at the end of 25 years. The grid is 21 x 21 x 11 with $\Delta x=\Delta y=60$ ft in the horizontal directions and $\Delta z=20$ ft in the vertical direction. The top of the grid is at a depth of 4000 ft, and the initial in situ reservoir porosity is 30%. Reservoir permeabilities vary by layer with horizontal permeabilities equal to 5, 100, 20, 20, 20, 100, 20, 20, 100, 20, and 20 md, respectively. Vertical permeabilities are 0.01 times horizontal permeabilities. Two–phase relative permeabilities and capillary pressures are listed in Tables 4 and 5, and Stone 2 is used for three-phase relative permeabilities.[12]

Water has a formation volume factor of 1.0 at 14.7 psi, a viscosity of 1 cp, a fluid density of 62.4 lbm/ft^3 at 14.7 psi, and fluid compressibility of 3 x 10^{-6} psi^{-1}. The oil and gas densities at the surface are 56.0 lbm/ft^3 and 57.0 lbm/mcf, respectively. Pressure-dependent oil and gas properties are listed in Table 6.

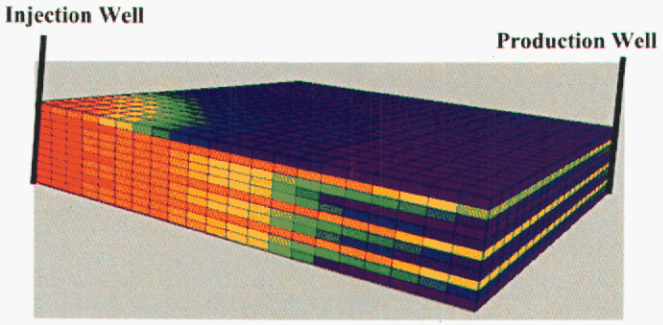

Fig. 8 –Water saturations after 25 years for problem 4

The initial reservoir pressure is 3010 psi at 4010 ft and initial fluid saturations are 20%, 80% and 0% for water, oil and gas, respectively. The oil is initially undersaturated with a bubble-point pressure of 3000 psi and an oil compressibility of 10^{-5} psi^{-1} in all layers.

The elastic modulus is 5 x 10^4 psi, Poisson's ratio is 0.35, and the initial in situ solid density (solid material without pores) is 2.7 gm/cm^3. The initial vertical stress is 4000 psi at the top of the reservoir with a vertical stress gradient of 0.9256 psi/ft throughout the grid and initial horizontal stresses are equal to half of the vertical stress. The bottom and sides of the grid have zero normal displacement constraints and all faces of the grid have zero tangential stresses. Assuming uniaxial strain behavior for this problem, the explicitly coupled simulation uses a value of 4.15 x 10^{-5} psi^{-1} for compressibility in Eq. 4.

Vertical wells are completed in diagonally opposite corners of the grid in all 11 layers. The water injector has a prescribed rate of 500 stb/d (¼ of the well's total rate), and the production well has a prescribed liquid rate of 750 stb/d (¼ of the well's total rate) with a limiting bottomhole pressure of 500 psi. Wellbore radii of 0.069 ft (instead of 0.25 ft) are used to represent wells of radii 0.25 ft that are at the corners of the grid blocks,[13] and a multiplying factor of 0.25 is used for the wellbore constants since only ¼ of a well's production is being simulated in the pattern. Simulations are performed for 25 years using time step sizes that are controlled by stability considerations for the IMPEM technique.

The three techniques produce nearly identical results for problem 4. **Fig. 9** shows average pore-volume-weighted, oil-phase pressures and subsidence in the center of the pattern at the top of the reservoir. **Fig. 10** shows the wellbore pressure, gas/oil ratio, and water/oil ratio at the production well.

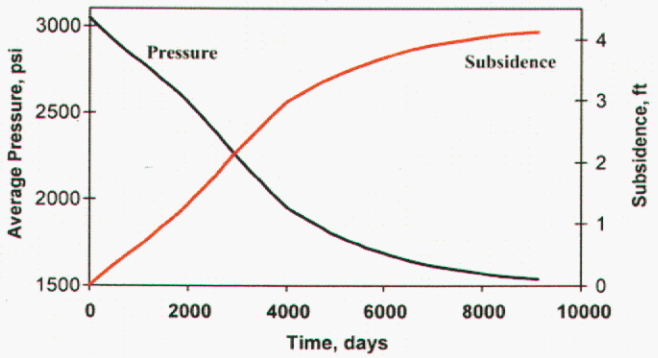

Fig. 9 – Average pressure and subsidence for problem 4

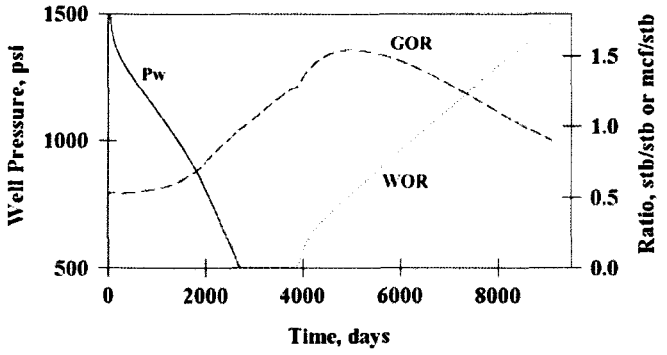

Fig. 10 – Production history for problem 4

The runtime information for problem 4 is displayed in Table 7. The explicitly coupled technique is much faster than the other two techniques for this problem because it performs a small number of geomechanical updates during the simulation. The explicitly coupled technique requires only 33 updates for displacements throughout the simulation. A minimum of 25 updates are required because displacements are printed each year during the simulation. The iteratively coupled and fully coupled techniques would perform better for this problem if they were combined with the implicit time stepping option in the program, rather than with the IMPEM option, but it is expected that the explicitly coupled option would still be the best option because few geomechanical updates are required.

Technique	*CPU Time*	*Time Steps*	*Iterations*
Explicit	**9.0 minutes**	**3324**	**3325**
Iterative	**40.6**	**3326**	**3326**
Full	**47.5**	**3326**	**3326**

Table 7 – Runtime information for problem 4

A run was performed without geomechanical calculations using a value of 4.15×10^{-5} psi^{-1} for c_r in Eq. 1 and the results reproduced the pressure and fluid histories in Figs. 9 and 10. The simulation without geomechanical calculations took 7.2 minutes; so for this problem, geomechanical calculations add only 25% to the overall computational time for the model when using the explicitly coupled technique.

Conclusions

Explicitly coupled, iteratively coupled, and fully coupled techniques have been applied to four sample problems. The three techniques produce nearly identical results on problems 1, 2, and 4 using the same time step sizes and the same convergence tolerances. Problem 3 involves geomechanical effects that are not present in the other three problems and the three techniques initially produced different results for this problem; however, all three techniques produced similar results when a tight tolerance was used for the nonlinear iterations for the iteratively coupled technique, and when small time steps were used for the explicitly coupled technique. All problems in this paper are described in detail so the results presented here may be used for comparison with other geomechanical/porous flow simulators.

The three coupling techniques produce similar results and one's selection of a technique is determined by ease of implementation, program availability, numerical stability, and computational efficiency. No technique worked best on all four problems presented in this paper. The fully coupled technique worked best for problem 3 running twelve times faster than the explicitly coupled technique, and the explicitly coupled technique worked best for problem 4 running five times faster than the fully coupled technique.

Acknowledgements

BP provided a copy of ACRES to the University of Texas at Austin for running the comparisons in this paper. The authors would like to acknowledge the support of DOE, NSF, NGOTP, and the Industrial

Affiliates of the Center for Subsurface Modeling at the University of Texas at Austin. Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy under contract DE-AC04-94AL85000.

Nomenclature

c_r = reservoir compressibility, Lt^2/m, psi^{-1}
E = elastic modulus, m/Lt^2, psi
E_{rel} = relative difference in pore volumes
M = Biot's poroelastic parameter, m/Lt^2, psi
p = fluid pressure, m/Lt^2, psi
p_o = initial fluid pressure, m/Lt^2, psi
V_b^o = initial grid block volume, L^3, ft^3
V_p = pore volume, L^3, ft^3
$\tilde{V}_p$ = pore volume estimate from pressure equation, L^3, ft^3
V_p^o = initial pore volume, L^3, ft^3
α = Biot's poroelastic parameter, dimensionless
δ_{ij} = Kronecker delta, dimensionless
Δ = change in a variable, dimensionless
ε_{ij} = strain, expansion is positive, dimensionless
ε_{kk} = volumetric strain, dimensionless
λ = Lame constant, m/Lt^2, psi
μ = Lame constant, m/Lt^2, psi
ν = Poisson's ratio, dimensionless
σ_{ij} = total stress, tension is positive, m/Lt^2, psi
σ_{ij}^o = initial total stress, m/Lt^2, psi
φ_o = initial porosity, dimensionless

References

1. Minkoff, S., Stone, C., Arguello, J., Bryant, S., Eaton, J., Peszynska, M. and Wheeler, M., "Staggered in Time Coupling of Reservoir Flow Simulation and Geomechanical Deformation: Step 1 --- One-Way Coupling," paper SPE 51920 presented at the 1999 SPE Reservoir Simulation Symposium, Houston, Feb. 14-17.
2. Settari, A. and Walters, D.A., "Advances in Coupled Geomechanical and Reservoir Modeling With Applications to Reservoir Compaction," paper SPE 51927 presented at the 1999 SPE Reservoir Simulation Symposium, Houston, Feb.14-17.
3. Minkoff, S., Stone, C., Arguello, J., Bryant, S., Eaton, J., Peszynska, M. and Wheeler, M., "Coupled Geomechanics and Flow Simulation for Time-Lapse Seismic Modeling," 69th Annual International Meeting of Soc. of Expl. Geophys. (1999) 1667-1670.
4. Minkoff, S., Stone, C., Bryant, S., Peszynska, M. and Wheeler, M., "A Two-way, Staggered-in-time, Loose Coupling Algorithm for Sophisticated Fluid Flow and Geomechanical Deformation Modeling," submitted in 2002 to Journal of Petroleum Science and Engineering.
5. Settari, A. and Mourits, F.M., "Coupling of geomechanics and reservoir simulation models," Comp. Methods and Advances in Geomech., Siriwardane and Zeman, (eds.), Balkema, Rotterdam (1994) 2151-2158.
6. Fung, L.S.K., Buchanan, L. and Wan, R.G., "Coupled Geomechanical-thermal Simulation for Deforming Heavy-oil Reservoirs," J. Cdn. Pet. Tech. (April 1994) 22.
7. Tortike, W.S., and Farouq Ali, S.M., "A Framework for Multiphase Nonisothermal Fluid Flow in a Deforming Heavy Oil Reservoir," paper SPE 16030 presented at the 1987 SPE Reservoir Simulation Symposium, San Antonio, Feb. 1-4.
8. Li, X. and Zienkiewicz, O.C., "Multiphase Flow in Deforming Porous Media and Finite Element Solutions," Computers and Structures (1992) 45, 211.

9. Lewis, R.W., and Sukirman, Y., "Finite Element Modelling of Three-Phase Flow in Deforming Saturated Oil Reservoirs," Intl. J. for Num. and Anal. Methods in Geomech.(1993) 17, 577.
10. ARCO Reservoir Simulator Development, "Appendix 30: Poroelastic Calculations," ACRES Reference Manual, Internal ARCO Document, ARCO, Plano, TX, 1999.
11. Gutierrez, M. and Lewis, R.W., "The Role of Geomechanics in Reservoir Simulation," paper SPE/ISRM 47392 presented at the 1998 SPE/ISRM Eurock Conference, Trondheim, July 8-10.
12. Stone, H.L., "Estimation of Three-Phase Relative Permeability and Residual Oil Data," J. Can. Pet. Tech. (1973) 12, 53.
13. Kuniansky, J. and Hillestad, J.G., "Reservoir Simulation Using Bottomhole Pressure Boundary Conditions," SPEJ (Dec. 1980) 473.

Metric Conversion Factors

bbl x 1.589 874	E-01	=	m^3
cp x 1.0*	E-03	=	Pa s
ft x 3.048*	E-01	=	m
lbm x 4.535924	E-01	=	kg
mcf x 2.831 685	E+01	=	m^3
md x 9.869 233	E-04	=	□m^2
psi x 6.894 757	E+00	=	kPa
psi^{-1} x 1.450 377	E-01	=	kPa^{-1}

*Conversion factor is exact.

Sw	Krw	Krow	Pwc
0.2	0.0	0.5102	6.4
0.25	0.0039	0.4133	5.6
0.3	0.0156	0.3266	4.9
0.35	0.0352	0.2500	4.2
0.4	0.0625	0.1837	3.6
0.45	0.0977	0.1276	3.0
0.5	0.1406	0.0816	2.5
0.55	0.1914	0.0459	2.0
0.6	0.2500	0.0204	1.6
0.65	0.3164	0.0051	1.2
0.7	0.3906	0.0	0.9
0.8	0.5625	0.0	0.4
0.9	0.7656	0.0	0.1
1.0	1.0	0.0	0.0

Table 4 – Water/oil data for problem 4

Sw+So	Krog	Krg	Pgc
0.2	0.0	0.6303	3.2
0.25	0.0	0.5511	2.8
0.3	0.0	0.4772	2.5
0.35	0.0026	0.4086	2.1
0.4	0.0104	0.3454	1.8
0.45	0.0234	0.2874	1.5
0.5	0.0416	0.2348	1.3
0.55	0.0651	0.1875	1.0
0.6	0.0937	0.1455	0.8
0.65	0.1275	0.1089	0.6
0.7	0.1666	0.0775	0.5
0.75	0.2108	0.0514	0.3
0.8	0.2709	0.0307	0.2
0.85	0.3149	0.0153	0.1
0.9	0.3748	0.0052	0.0
0.95	0.4398	0.0004	0.0
0.97	0.4673	0.0	0.0
1.0	0.5102	0.0	0.0

Table 5 – Gas/oil data for problem 4

Pressure psi	B_o rvb/stb	B_g rvb/mcf	R_s mcf/stb	μ_o cp	μ_g cp
300.00	1.0663	10.2582	.0610	1.5	.02
600.00	1.0931	4.9878	.1161	1.5	.02
900.00	1.1173	3.2461	.1681	1.5	.02
1200.00	1.1408	2.3855	.2197	1.5	.02
1600.00	1.1718	1.7522	.2894	1.5	.02
2000.00	1.2030	1.3838	.3608	1.5	.02
2400.00	1.2346	1.1479	.4342	1.5	.02
2800.00	1.2667	.9876	.5102	1.5	.02
3000.00	1.2843	.9221	.5521	1.5	.02
3200.00	1.2996	.8743	.5889	1.5	.02
3600.00	1.3334	.7921	.6708	1.5	.02
4000.00	1.3683	.7312	.7561	1.5	.02
4500.00	1.4137	.6763	.8685	1.5	.02

Table 6 – Pressure dependent oil and gas data for problem 4

Intentionally Left Blank

Appendix B – Xiuli Gai, Rick H. Dean, Mary F. Wheeler, and Ruijie Liu, "Coupled Geomechanical and Reservoir Modeling on Parallel Computers", presented at the SPE Reservoir Simulation Symposium, Houston, Texas, U.S.A., 3-5 February 2003, paper number SPE 79700.

Intentionally Left Blank

Coupled Geomechanical and Reservoir Modeling on Parallel Computers

Xiuli Gai, University of Texas at Austin, Rick H. Dean, SPE, Mary F. Wheeler and Ruijie Liu, University of Texas at Austin

Abstract

In coupled geomechanics and reservoir modeling, the finite element discretization of the force balance equation leads to very large linear systems, whose solution is both time and memory consuming. ICCG (Incomplete Cholesky Factorized Conjugate Gradient) is a popular technique for solving for displacements, but the technique is limited to about 60,000 nodal points on desktop machines. Most large 3D field scale problems will have to be run on parallel machines. In this paper, we present a reduced-communication, super coarsening multigrid method that can be combined with other domain decomposition-based preconditioner to achieve faster convergence with high parallel scalability. A preliminary test case of 1.5 million grid blocks with up to 59 processors shows a parallel efficiency of above 90%.

Introduction

The modeling of fluid-structure interactions is of growing importance for both energy and environmental applications. Due to complexities, nonlinearities, phase behavior, and the number of partial differential equations required to describe a coupled system of poro-elasticity and/or poro-plasticity with multiphase flow, extending a conventional reservoir model to a coupled fluid-flow and geomechanical model is not trivial, though considerable success has been achieved in recent years.

Several authors[1,2,3,4,5] have presented mathematical and numerical formulations for the coupled equations. Different coupling techniques have been investigated[6,7] that are applicable to existing reservoir flow models and one's choice normally depends on speed, accuracy, or ease of implementation. Comparisons have been done between coupled and uncoupled simulations, and between different coupling techniques for accuracy and efficiency. Recently, stability issues have also been discussed concerning oscillations in low permeability zones. Individual models are becoming more sophisticated, and coupling methods are becoming more accurate and more stable. However, the computational bottleneck of solving the large linear system still remains. The finite element formulation of the geomechanics model has a 27-point stencil compared to the standard 7-point stencil in the finite difference formulation of multiphase flow equations. More than 70 percent of the total CPU time for a coupled simulation is spent in the geomechanics model solving the system of linear equations. So it is very important to develop efficient linear solvers to reduce the computational cost without losing scalability on parallel computers, so that coupled analyses can be economically and numerically feasible for practical field applications. Thomas *et al.*[8] describe a coupled procedure running on multiple processors, but the parallel efficiency with 16 processors is less than 50 percent.

In this work, we present a parallel poroelastic model combined with a multiphase flow model using the iteratively coupled technique. The linear solver we employ is PCG preconditioned by Incomplete Cholesky Factorization (IC) or block Jacobi. Faster convergence and higher scalability is achieved by calling a super coarsening multigrid routine to make further global corrections after each preconditioning step.

The rest of the paper is organized as follows. First the governing flow and deformation equations are described briefly. Instability issues are also briefly discussed. Then the super coarsening multigird preconditioner and its parallel implementation are presented. Results of numerical experiments conducted

on a 64-node Beowulf PC cluster are shown to demonstrate the efficiency and parallel scalability of the combined IC (or block Jacobi) and multigrid preconditioned BiCGSTAB (Biconjugate Gradient Stablized) method.

Geomechanics and Multiphase Flow Equations

Terzaghi first analyzed the fluid-flow-stress coupling equations in 1925 as a 1D consolidation problem. Later, Biot[9] extended the theory to a more generalized three-dimensional case, based on a linear stress-strain relation and a single-phase fluid flow. Here we present an extension of Biot's equations for three-phase immiscible and isothermal flow.

Static-equilibrium equation:

$$\nabla \cdot \boldsymbol{\sigma} + \mathbf{f} = 0 \qquad (1)$$

Incremental isotropic constitutive equations relating the total stresses, bulk strains, and fluid pressure:

$$d\boldsymbol{\sigma} = \mathbf{D}\, d\boldsymbol{\varepsilon} - \alpha \mathbf{m}\, dp \qquad (2)$$

Incremental strain-displacement relation:

$$d\varepsilon_{ij} = \frac{1}{2}\left(du_{i,j} + du_{j,i}\right) \qquad (3)$$

The force balance Eq. 1 is formulated in terms of displacements $\boldsymbol{u}$ using relations of Eq. 2 and Eq. 3. A linear Galerkin finite element method is used to discretize Eq 1 to 3. Multiphase flow calculations enter the stress calculation through the fluid pressure p in Eq. 2, which is taken to be the water phase pressure in our formulation.

Multiphase flow mass conservation equations:

$$\begin{aligned} &\frac{\partial}{\partial t}\left[\phi^* N_{o,w}\right] + \nabla \cdot \mathbf{U}_{o,w} = q_{o,w} \\ &\frac{\partial}{\partial t}\left[\phi^* N_g\right] + \nabla \cdot \left(\mathbf{U}_g + R_o \mathbf{U}_o\right) = q_g \end{aligned} \qquad (4)$$

where the first equation above represents the water and oil phases. Here, ϕ^* is called the fluid fraction to differentiate it from the true porosity ϕ for deformable rocks, and is calculated as

$$\phi^* = \frac{V_p}{V_b^0} = \phi^0 + \alpha\left(\nabla \cdot \mathbf{u} - \nabla \cdot \mathbf{u}^0\right) + \frac{1}{M}\left(p - p^0\right) \qquad (5)$$

Like in most commercial reservoir simulators, Eq. 4 is formulated using a cell centered finite difference method. An iterative procedure[6,7,10] is taken to integrate the reservoir flow model Eq. 4 to the geomechanics model Eq 1 to 3. The former operates as a primary control model and calls the geomechanics routines for new porosity values whenever its pressure and saturation information are updated. Convergence is checked based on criteria in both the reservoir model and geomechanics model.

One should be aware that the above finite element formulation, a standard procedure employed in the oil industry using piece-wise constant pressure and continuous piece-wise linear displacements, is not always numerically stable. Oscillations may occur in low permeability regions. The stability condition requires that pressure and displacement spaces cannot be chosen independently when the discretization is based on the "Galerkin" variational form. To remove the pressure oscillation one may adopt a pressure stabilization method (Hughes *et al.*[11] and Wan[12]). However, the method does not conserve mass exactly. Another local mass conservative approach is to apply a coupled Galerkin and NIPG scheme in the low permeability

regions. Detail discussion on the theory and numerical results of this method will appear soon in another paper.

Super coarsening Multigrid Preconditioner

In iterative coupling, rock deformation and reservoir flow equations are solved independently. The advantage is that fast and efficient linear solvers can be developed for each individual model. Even in a fully coupled scheme, a fast and efficient solver for the elasticity equations could be useful. It can produce good estimates for the displacements before an updated solution for the total system is computed. Finite element discretizations of elasticity equations lead to large systems of linear equations with sparse matrices. A typical numerical method for such a system is the preconditioned conjugate gradient method (PCG). The PCG method is valued in that it is very suitable for parallel computing and even an ill-conditioned problem can be easily solved with the help of a good preconditioner. The multigrid method is also a well-known scalable method whose convergence rate is independent of problem size and the number of iterations remains fairly constant. Several authors have investigated different combinations of multigrid and the conjugate gradient methods. Kettler *et al*[13, 14] used multigrid as a preconditioner for the PCG method while Bank and Douglas[15] treated PCG as a relaxation method of the multigrid solver. Braess[16] considered these two combinations and reported that the conjugate gradient method with multigrid preconditioning is effective for elasticity problems. Here, we propose a new combination that uses super coarsening multigrid together with incomplete Cholesky factorization or block Jacobi as preconditioners for the BiCGSTAB (biconjugate gradient stabilized) method. Using additional multigrid preconditioning at low computation cost, an efficient method with high parallelism and fast convergence is obtained.

Super Coarsening Multigrid V-cycle

Super coarsening multigrid method was developed by Wheeler[17,18] who took advantage of the fact that most reservoir problems have a vertical dimension that is much smaller than the horizontal dimensions. In his multigrid implementation, a 3D fine grid is collapsed to a coarser 2D grid after the first coarsening step (Fig. 1). He has applied this method to reservoir multiphase flow equations, and it appears to be very fast and efficient. In this work, we adopt the same idea for the elasticity equations.

The basic idea of multigrid method is to solve linear systems based on restricting and extrapolating solutions between a series of nested grids. Simple iterative methods (such as Gauss-Seidel) tend to damp out high frequency components of the error fastest, but low frequency error is left. Multigrid can eliminate low frequency errors efficiently on coarse grids. Recursive application of this idea to each consecutive system of coarse-grid equations leads to a multigrid V-cycle (Fig. 1). If the components of the V-cycle are defined properly, the result is a method that uniformly damps all error frequencies with a computational cost that depends only linearly on the problem size. In other words, multigrid algorithms are scalable. For more details and discussions, we refer to available introductory publications on multigrid methods; in particular, ref. 19 and ref. 20. The four basic components employed for the multigrid method in our implementation are listed as follows:

- Full weighting restriction operator maps *h-grid* defects to *2h-grid*. Its stencil reads

$$I_h^{2h} = \frac{1}{16}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}_h^{2h}$$

- Bilinear interpolation prolongation operator maps *2h-grid* corrections to *h-grid* corrections. Its stencil notation reads

$$I_{2h}^{h} = \frac{1}{4}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}_{2h}^{h}$$

- Galerkin coarse grid operator A_{2h} is used and defined as

$$A_{2h} = I_h^{2h} A_h I_{2h}^h$$

- 4-color Gauss-Seidel relaxation performs both pre-smoothing and post-smoothing on the coarse grids.

Preliminary tests show that the above super coarsening multigrid method does not work well for elasticity equations if used directly as a preconditioner for a PCG solver, since it solves only for 2D problems. However, if we can apply it as an extra correction step for the generally used PCG preconditoners, the resulting algorithm is robust, efficient and scalable. To account for the unsymmetric preconditioning matrix, BiCGSTAB method is used instead of conjugate gradient method.

Model Problem

We now consider the above method for the solution of a three-dimensional, three-phase waterflooding problem in a quarter of a five spot pattern. The size of reservoir is 1056x1056x160 ft. A uniform grid, 16x16x8 is used. Initial reservoir pressure is 3020 psi at a depth of 4650 ft. Initial in-situ porosity is 0.3. Initial oil saturation is 0.8. No free gas is present. The permeability field varies by layer. With the same reservoir model, two different sets of boundary conditions are enforced for the stress model. While both cases apply a load of 6000 psi on top of the reservoir, the first case runs with rigid side and bottom boundaries (Fig. 2) and second case applies a normal stress of 2600 psi on the side boundaries and keeps the bottom boundary fixed (Fig.3). Both cases are run with block Jacobi and incomplete Cholesky factorization preconditioner. Total number of iterations and running time with multigrid correction are compared with the results running without multigrid correction. Linear solver tolerance is set to be 1.0^{-5}.

The first case, constrained displacement case, runs for 20 time steps and takes 86 Newton iterations, which means that the linear elasticity system is solved 86 times during the simulation. With block Jacobi preconditioner, the computation domain is divided into several subdomains in horizontal directions. Each subdomain problem is solved directly. Comparison results are show in Table 1. The first column in Table 1 shows the numbers of rows and columns in each subdomain. The extreme case is 1 x 1 subdomain case or line Jacobi, by which a vertical line is solved simultaneously. The second and third columns show the total number of iterations and elapsed time using block Jacobi preconditioned PCG method. The last two columns show the total number of iterations and elapsed time using block Jacobi preconditioned BiCGSTAB method with multigrid corrections. Corresponding results for CG solver and direct solver are also given in Table 1. The direct method is fastest for this linear elastic problem because the stiffness matrix is only factored once. The direct method would be much slower for plastic problems where the stiffness matrix must be reformulated and factored at every time step. Our first observation is that no matter how the problem domain is divided, multigrid corrections can greatly reduce the number of iterations and cut the total running time. The second observation is that the total number of subdomains and the shape of each subdomain affect the convergence behavior, but the degradation can be reduced by the multigrid method.

Similar runs using IC preconditioner with different fill levels are shown in Table 2. The IC preconditioner works better than block Jacobi. However there is the multigrid method still improves the results, no matter how many fill levels are used

Block Jacobi (N_x, N_y)	PCG	BiCGSTAB (Multigrid)

	Num. of iterations	Time (sec.)	Num. of iterations	Time (sec.)
1 x 1	4336	118.4	746	68.6
2 x 2	3448	121.1	602	68.4
2 x 4	3539	137.5	596	70.4
4 x 2	3277	152.1	580	87.1
4 x 4	2035	121.8	531	83.3
8 x 8	1520	172.9	454	112.4
CG	Takes 12891 iterations and 249.1 seconds			
Direct solver	Takes 34.4 seconds			

Table 1: Total number of iterations and elapsed time using block Jacobi preconditioner with / without multigrid corrections for the model problem with constrained displacement B.C.

IC (Fill level)	ICCG		BiCGSTAB (Multigrid)	
	Num. of iterations	Time (sec.)	Num. of iterations	Time (sec.)
0	2382	75.0	561	57.0
1	1599	77.7	439	59.0
2	534	88.4	303	62.5
CG	Takes 12891 iterations and 249.1 seconds			
Direct solver	Takes 34.4 seconds			

Table 2: Total number of iterations and elapsed time using IC preconditioner with / without multigrid corrections for the model problem with constrained displacement B.C.

Unlike the first case, which is close to a uniaxial strain problem because of the fixed horizontal and bottom boundaries, the unconstrained case is a little bit harder to solve. It takes more iterations for the linear solver to converge even with the incomplete Cholesky factorization method. From Table 3 and Table 4, it can be seen that the multigrid corrections dramatically reduce the total number of iterations for both block Jacobi and IC preconditioner. The resulting BiCGSTAB solver is at least two times faster than the PCG method without multigrid corrections. It is also interesting to see that ICCG method with two fill levels is faster than zero and one fill levels. However, after combining with multigrid method, IC with zero fill is the fastest one, which is good because the zero fill level takes much less memory space.

Block Jacobi (N_x, N_y)	PCG		BiCGSTAB (Multigrid)	
	Num. of iterations	Time (sec.)	Num. of iterations	Time (sec.)
1 x 1	15292	411.5	1372	154.9
2 x 2	11378	388.7	1085	136.8
2 x 4	10186	429.5	1073	142.7
4 x 2	9491	441.8	1062	156.8
4 x 4	7137	403.2	937	159.9
8 x 8	4424	459.4	806	213.6
CG	Takes 57330 iterations, 1140.9 seconds			
Direct solver	Takes 49.8 seconds			

Table 3: Total number of iterations and elapsed time using block Jacobi preconditioner with / without multigrid corrections for the model problem with unconstrained displacement B.C.

IC (Fill level)	ICCG		BiCGSTAB (Multigrid)	
	Num. of iterations	Time (sec.)	Num. of iterations	Time (sec.)
0	8831	258.8	893	108.3
1	5236	248.1	780	119.6
2	2897	247.8	667	152.1
CG	Takes 57330 iterations, 1140.9 seconds			
Direct solver	Takes 49.8 seconds			

Table 4: Total number of iterations and elapsed time using IC preconditioner with / without multigrid corrections for the model problem with unconstrained displacement B.C.

Parallel Multigrid Preconditioner

A parallel version of the coupled reservoir and geomechanics model was developed under the framework called IPARS[21] (Integrated Parallel Accurate Reservoir Simulator), based on the message-passing interface (MPI). The parallel work for this reservoir flow model has already been done. Parallel performance analysis for a three-dimensional, three-phase example with one million grid blocks shows an almost linear speedup (Fig 4). In the following sections, we focus on parallel implementation and performance evaluation for the geomechanics model. In particular, we will deal with the question of how multigrid algorithm is implemented on parallel computers.

Grid Partitioning

The IPARS framework applies the domain decomposition method to divide the original grid into as many subgrids as processors employed. We chose the partition in such a way that every grid point belongs to only one subgrid or subdomain. Each processor is then assigned to one subgrid. To handle the dependency of a grid point on its adjacent interface points, one overlapping boundary (ghost point) is placed around each subgrid. To eliminate the necessity of data exchange between the flow and geomechanics model both models are assigned to the same subgrid and share the same memory slots for pressure, saturation and porosity variables. The solution proceeds by solving the original problem in each subdomain followed by data exchanges of the overlapping boundary conditions on the artificial interfaces.

The mapping of the subgrids on different multigrid levels to individual processors is based on the decomposition of the fine grid levels. Since the coarse grid problem is a direct analog of the fine grid problem, we can perform grid partitioning on the coarse grid accordingly. In general, there is no reason to change the partitioning of the subdomains or the mapping to individual processors. In other words, the same geometric points, with the exception of boundary points, on different grid levels are allocated to the same processor. Otherwise, additional communication among processors would be required during the intergrid transfer.

In terms of data communication, two types of overlapping are required by the multigrid components (Lang *et al.*[22]).

- *Horizontal Overlap (HGhost)*: After each coarse grid level is divided into subdomains, one overlapping boundary needs to be placed around each subdomain on each level. Of course, the overlapping concept has to be adapted according to the different grid levels. On each grid, at least one overlap region is needed in order to be able to perform the parallel smoothing step and defect

calculations. Since the distance between two adjacent grid points increases on coarse grids, the "geometric size of overlap regions" will be different on different grid levels.

- *Vertical Overlap (VGhost)*: Between two multigrid levels, whenever a coarse grid element requires information of its neighboring fine grid elements, which belongs to other processors, this coarse grid element must have local copies of those fine grid elements. For example, in Fig. 5, the coarse grid element in processor 0 has dependency on the three fine grid elements, which are allocated to processors 1, 2 and 3 respectively.

Parallel Multigrid Communications

There are five algorithmic steps on each multigrid level that may require data communications. They are smoothing, defect calculation, the fine-to-coarse grid transfer, coarse-to-fine grid transfer and coarse grid operator generation.

- *Relaxation*
  Assume that some initial guess is given and the HGhost nodes are updated. Then, with some smoothers (like Gauss-Seidel) each processor may independently perform one relaxation sweep over its subgrid. After all processors have completed this sweep, the information at HGhost nodes must be updated either for the next sweep or for the defect calculation.

- *Defect calculation*
  The pre-smoothing steps are followed in the coarse grid correction scheme by defect calculation. Since we have already updated the HGhost nodes after each relaxation sweep, there is no need for any update prior to defect calculation, and the processors may complete this task independently.

- *Restriction*
  If full weighting is used as a restriction operator each coarse grid point must know the defect values of all neighboring fine grid points. For example, in a 2D problem, a coarse grid point has totally eight neighboring fine grid points. Local copies are required if some of those grid points belong to other processors. To solve this problem one could simply transfer the missing defect values from the fine grids processor to coarse grid processor and store them at the VGhost nodes. However, a more efficient method allows each processor to calculate its share of the restricted defects; and then send it to the master processor, where all the defects collected from other processors are summed together. In Fig. 6, the defect calculations of a coarse grid point in processor 3 (squared black point) need the defect data of eight neighboring fine grid points, five of which are allocated to other processors, including one point to processor 0, two points to processor 2, and 2 points to processor 1. Instead of sending the data of these five grid points to processor 3, all processors involved will calculate their own share of the restricted defects at the coarse point. Later, processors 0, processor 1, and processor 2 send their computed defects to the master processor 3, where all the values are summed together.

- *Prolongation*
  With bilinear interpolation, if the HGhost nodes on the coarse grid have already been updated after the last smoothing step on that level, there is no need for additional communication to calculate the correction in each subdomain on the fine grid. If the HGhost nodes on the fine grid have not been changed since the last smoothing step we can perform the correction to the HGhost nodes on the fine grids without data transfer between processors.

- *Coarse Grid Operator Generation*
  If a Galerkin coarse grid operator is employed, a coarse grid point needs the coefficient data of its neighboring fine grid points. It is quite similar to the full weighting restriction. The stencil here, however, is larger than a 9-point full weighting stencil. For symmetric matrices a 12-point stencil is employed. Again, a better way to handle this is to use a coarse-to-coarse grid data transfer scheme instead of a fine-to-coarse grid scheme. Passing stiffness matrix coefficients among processors seems unconventional, but with half stored symmetric matrices it is necessary.

It should be noted that all the communication required in the defect calculation and coarse grid operator generation may be totally avoided by using the non-overlapping domain decomposition method.

Communication Reduction Techniques

Parallel efficiency of an algorithm depends on the ratio of communication time to computation time; it is directly proportional to the ratio of volume to surface area of each subdomain. In a multigid cycle, as the grid becomes coarser and coarser, the volume to surface area ratio decreases. In addition, as fewer and fewer grid points are mapped to multiple processors, more and more processors are left without any grid point to compute on the very coarse grids. Finally, only one or a few processors have one (or a few) grid point. Whereas the idling processors on the very coarse grids appear to be the main problem at first sight, experience and theoretical considerations show that the large communication overhead on the very coarse grids is usually of greater concern than the idling processors. Special techniques developed to reduce the coarse level communication include coarse grid agglomeration, employing multiple coarse grids, and using different cycle schemes. However, as long as one demands that the results of the parallel algorithm be identical with those of the sequential algorithm, a substantial reduction of the communication cost will not be achieved.

Generally, the time needed to send a message of length L in one packet is modeled by the formula

$$t_{comm} = \alpha + \beta L \qquad (6)$$

Here α is the so-called start-up time, which is required for each message. β is the time required to transfer one word. It is determined by the bandwidth of the respective communication channel. So if α is large, the number of messages should be minimized; if β is large, the communication volume is of primary concern. Here we discuss some simple and easily implemented techniques based on reducing both the number of messages and the volume of each message.

- Intergrid-level communications from fine grid to coarse grid are replaced by intra-grid-level communications in the defect and coarse grid operator calculations. Here the total number of messages stays the same, but the amount of data transferred is reduced.
- On very coarse grids, as the number of grid points decreases, the start-up time α dominates the communication overhead. To reduce the number of messages, one can pack several variables in the same packet instead of sending one variable at a time.

- If RB Gauss-Seidel method is used for smoothing the data volume L in Eq. 6 may be reduced by RB updates after each sweep.

- The communication overhead consists of message handling, as described in Eq. 6, and network latency. Overlapping the communication with computation can hide the network latency. For example, the Gauss-Seidel relaxation can be separated into two parts: local computation that involves only inner points and non-local computation that requests ghost point's information. The first part can be done while a processor is waiting for the incoming data. After receiving all the ghost node data, the processor may continue with the second part of the computation. If remote data reaches a processor during the process of the local computation, network latency can be hidden.

Parallel Performance Evaluation

In this section, we will show two examples to illustrate the parallel performance of our coupled geomechanical and reservoir flow models. The first one is a single-phase ground water pumping problem. The second one is a three-phase waterflooding problem. All the parallel runs are performed on a 64-node Beowulf PC cluster on a 1.28 gigabit/sec Myrinet network. Each node is a Pentium II 300 MHz processor.

The cluster has 32 GB of aggregate RAM and over 200 GB of collective storage.

Single Phase Example

The total reservoir dimensions are 113,600 x 113,600 x 1016 ft. The domain is discretized with uniform 128 x 128 x 11 rectangular meshes. The maximum interval length in horizontal directions is 1875 ft, while the minimum one is 200 ft. Vertical layers have varying thicknesses, ranging from 154 ft in the overburdens and underburdens to 6 ft in the seal layers. No flow occurs on any of the reservoir boundaries. With the exception of a free surface on top, all the other reservoir boundaries are rigid. Horizontal permeability is 2000 md throughout the reservoir. Vertical permeability is 10 percent of the horizontal one. Porosity values vary from 0.43 to 0.05 by layers. The initial pressure is 14.7 psi at the ground surface. Sixteen production wells are located at the reservoir center with a variant pumping rate from 339 stb/day to 15562 stb/day. The simulation was originally set up to run for 50 years with a constant time step of 365 days; but in this parallel scaling study, it runs for only 20 time steps and takes 39 Newton iterations. Linear solver tolerance is 10^{-4}.

First, we run the problem with BiCGSTAB solver preconditioned by line Jacobi combined with multigrid corrections to see whether the communication reduction techniques can improve the parallel efficiency. The multigrid cycle consists of six grid levels. The coarsest one has only 2 x 2 elements. Fig. 7 and Fig. 8 show that parallel performance is dramatically improved using our communication reduction techniques. Parallel efficiency is greatly increased from 66 percent to 90 percent with 32 processors.

In the next couple of runs, we use the incomplete Cholesky factorization as a subdomain solver, and compare the total running time between ICCG method without multigrid preconditioning and BiCGSTAB method with the multigrid preconditioning. At least three processors are required to provide enough memory for the incomplete the Cholesky factorization. The total running time with different number of processors is shown in Fig. 9 and Fig. 10. As seen in the former, single processor test case, the multigrid method greatly improves the convergence rate of the BiCGSTAB solver for both zero and one fill levels. Fig. 11 shows good speedups for both ICCG method and BiCGSTAB method. From Table 5, it can be seen that the influence of grid partitioning on the linear solver behavior is not significant for this problem. Fig. 12 shows that the IC preconditioner with zero fill level performs even better than IC (1), when used together with the multigrid method.

Number of Processors	Total Number of Iterations			
	ICCG (0)	BiCGSTAB (Multigrid)	ICCG (1)	BiCGSTAB (Multigrid)
3	12215	309	8576	264
4	11575	313	7745	284
8	12089	326	8311	273
16	12185	342	8004	305
24	12025	343	8180	300
32	12233	362	8576	345

Table 5: Preconditioner comparisons for single-phase example

Three-phase Black-oil Example

The main purpose of running this example is to show that using the parallel version of the coupled reservoir and geomechanics model, simulation of large scale, full-field 3D problems with millions of unknowns is now feasible.

This is a waterflooding problem. The grid configuration is 256 x 256 x 22 with □x=□y=300 ft in the horizontal directions and varying thicknesses in the vertical direction. In total, there are 1.5 million grid points with 9 million unknowns. Both porosity and permeability fields vary in layers. Initial reservoir pressure is 3500 psi and initial oil saturation is 0.8. There is no free gas. A water injection well is located in one corner of the reservoir. Twenty-five production wells are spread out through the entire domain. The

injection rate is 4000 stb/day. Production rates are 2000 stb/day. The simulation is run with up to 59 processors for 12 time steps and 36 Newton iterations. Linear solver tolerance for the geomechanics model is 10^{-3}. For this example, we use IC (0) combined with the multigrid method. The multigrid V-cycle consists of seven grid levels, and the coarsest one has only 2 x 2 grids. Fig. 13 and Fig. 14 show a good parallel speedup with 90 percent parallel efficiency. The parallel efficiency and speedup are normalized by the total running time of 16 processors. The little bumps in Fig. 13 are caused by grid partitioning for different number of processors, including different number of subdomain and the different shape of each subdomain.

Discussion

The convergence rate of the super coarsening multigrid method may be deteriorated when the material property is anisotropic or a reservoir has great heterogeneity in the vertical direction. However, in our implementation, since the method does not work as a stand alone preconditioner, the deterioration may not cause significant convergence problem.

For nonlinear geomechanical problems, the stiffness matrix needs to be reformulated at each nonlinear iteration. One may concern that the data communications required in generating the Galerkin coarse grid operators may cause unbearable overhead. We rerun the three-phase problem assuming a nonlinear behavior with a changing stiffness matrix, so that all the matrix related operations will be conducted at each nonlinear iteration. The parallel speedup shown in Fig. 15 is even better than the linear elasticity case (Fig. 14). Fig. 16 shows that above 90 percent of parallel efficiency is achieved. It appears that the additional communication overhead has been hidden up by the high computation intensity in the incomplete Cholesky factorizations and coarse grid matrix calculations.

Conclusion

1. A parallel version of a coupled reservoir flow and geomechanics model is developed using overlapping domain decomposition and MPI.
2. A fast, efficient and scalable parallel linear solver is implemented to solve the large, sparse system of linear equations arising from Galerkin finite element formulation of the force balance equation.
3. A reduced-communication, super coarsening multigrid procedure is applied to the elasticity equations. It can be combined with other domain decomposition-based preconditioners to achieve a faster convergence rate with high parallel scalability.
4. A field scale, three-phase example with 1.5 million grid points, and total 9 million unknowns is run with up to 59 processors. Above 90 percent of parallel efficiency is achieved.
5. The parallel solution presented in this paper is based on the formulation of a continuous linear displacement and piece-wise constant pressure. This finite element scheme may result in pressure oscillations in the low permeability regions. We recently coupled the continuous Galerkin with a Non-Symmetric Interior Penalty method to successfully remove the pressure oscillations. The theory and numerical results of this new formulation scheme will appear in another paper soon. Special considerations for the parallel implementation of this new formulation will also be investigated.

Nomenclature

$\boldsymbol{\sigma}$ = stress tensor
$\square$ = strain tensor, expansion is positive
$\boldsymbol{f}$ = gravity force
$\mathbf{D}$ = stress-strain relationship matrix
$\mathbf{u}$ = displacement
$\mathbf{u}^0$ = initial displacement vector
$\square$ = Biot's poroelastic constant
M = Biot's poroelastic constant

p = fluid pressure
p^o = initial fluid pressure
ϕ^* = fluid fraction
V_b^o = initial grid block volume
V_p = pore volume
N_o = oil concentration
N_w = water concentration
N_g = gas concentration
$\boldsymbol{U}_o$ = oil phase velocity
$\boldsymbol{U}_w$ = water phase velocity
$\boldsymbol{U}_g$ = gas phase velocity
R_o = solution gas oil ratio
q_o = well oil flow rate ver unit volume
q_w = well water rate ver unit volume
q_g = well gas rate ver unit volume

References

1. Li, X. and Zienkiewicz, O.C.: "Multiphase Flow in Deforming Porous Media and Finite Element Solutions," Computers and Structures (1992) 45, 211.
2. Gutierrez, M. and Lewis, R.W.: "The Role of Geomechanics in Reservoir Simulation," paper SPE/ISRM 47392 presented at the 1998 SPE/ISRM Eurock Conference, Trondheim, Norway, 8-10 July.
3. Lewis, R.W. and Sukirman, Y.: "Finite Element Modelling of Three-Phase Flow in Deforming Saturated Oil Reservoirs," Intl. J. Num. and Anal. Meth. In Geomech., (1993), 17, 577-598.
4. Osorio, J.G., Chen, H.Y. and Teufel, L.W.: "Numerical Simulation of the Impact of Flow-Induced Geomechanical Response on the Productivity of Stress-Sensitive Reservoirs," paper SPE 51929 presented at 1999 SPE Reservoir Simulation Symposium, Houston, Texas, 14-17 February.
5. Chen, H.Y., Teufel, L.W. and Lee, R.L.: "Coupled Fluid Flow and Geomechanics in Reservoir Study – I. Theory and Geoverning Equations," paper SPE 30752 presented at the SPE Annual Technical Conference & Exhibition held in Dallas, U.S.A. 22-25 October, 1995
6. Settari, A. and Walters, D.A.: "Advances in Coupled Geomechanical and Reservoir Modeling With Applications to Reservoir Compaction," paper SPE 51927 presented at the 1999 SPE Reservoir Simulation Symposium, Houston, 14-17 February.
7. Settari, A. and Mourits, F.M.: "Coupling of geomechanics and reservoir simulation models," Comp. Methods and Advances in Geomech., Siriwardane and Zeman, (eds.), Balkema, Rotterdam (1994) 2151-2158.
8. Thomas, L.K., Chin, L.Y., Pierson, R.G., and Sylte, J.E.: "Coupled Geomechanics and Reservoir Simulation," paper SPE 77723 presented at SPE Annual Technical Conference & Exhibition, San Antonio, Texas, 29 September - 2 October, 2002.
9. Biot, M.A.: "General Theory of Three-Dimensional Consolidation," J. Appl. Phys, (1941), 12, 155.
10. Tran, D., Settari, A. and Nghiem, L.: "New Iterative Coupling Between a Reservoir Simulator and a Geomechanics Module," paper SPE/ISRM 78192 presented at the SPE/ISRM Rock Mechanics Conference, Irving, Texas, 20-23 October, 2002.
11. Hughes, T.J.R., Franca, L.P. and Balestra M.: "A new finite element formulation for computational fluid dynamics: V. Circumventing the Babuška-Brezzi condition: A stable Petrov-Galerkin formulation of the Stokes problem accommodating equal-order interpolations", Comp. Meth. Appl. Mech. Engrg., (1986) 59, 85-99.
12. Wan. J.: *Stabilized Finite Element Methods for Coupled Geomechanics and Multiphase Flow, Ph.D. thesis, Stanford University, 2002.*
13. Kettler, R., and Meijerink J.A.: "A multigrid method and a combined multigrid-conjugated gradient method for elliptic problems with strongly discontinuous coefficients in general domains," Tech. Rep. 604, Shell Oil Company, 1981.

14. Kettler, R.: "Analysis and Comparison of Relaxation Schemes in Robust Multigrid and Preconditioned Conjugate Gradient Methods," in *Multigrid Methods* (W. Hackbusch and U. Trotenberg, eds.), vol. 960 of Lecture Notes in Mathematics, pp. 502-534, Springer-Verlag, 1982.
15. Bank, R.E. and Douglas, C.C.: "Sharp estimates for multigrid rates of convergence with general smoothing and acceleration," SIAM J. Numer. Anal., vol. 22, no. 4, pp. 617-633, August 1985.
16. Braess, D.: "On the Combination of Multigrid Methods and Conjugate Gradients," in *Multigrid Methods II* (W. Hackbusch and U. Trotenberg, eds.), vol. 1228 of Lecture Notes in Mathematics, pp. 52-64, Springer-Verlag, 1986.
17. Wheeler, J. and Smith, R.: "Reservoir simulation on a hypercube," paper SPE 19804 presented at 1989 SPE Annual Technical Conference & Exhibition, San Antonio, 8-11 October, 1989.
18. Lacroix, S., Vassukevski, Yu., Wheeler, J. and Wheeler, M.: "Iterative Solution Methods for Modeling Multiphase Flow in Porous Media Fully Implicity," under revision for publication in SISC.
19. Trottenberg, U., Oosterlee, C.W. and Schüller, A.: *Multigrid*, Academic Press, San Deigo, 2001.
20. Briggs. W.L., Henson, V.E. and McCormick, S.F.: *A Multigrid Tutorial Second Edition*, SIAM, 2000.
21. Wheeler, J.: "Integrated Parallel Accurate Reservoir Simulator (IPARS)," presented at The 8th Annunal Industrial Affiliates Meeting, Center for Subsurface Modeling, The University of Texas at Austin, 27-28, October 1998.
22. Lang, S., Wieners, C. and Wittum, G.: "The application of adaptive parallel multigrid methods to problems in nonlinear solid mechanics," Error-Controlled Adaptive Finite Element Methods in Solid Mechanics, Editor Erwin, Stein, 2000, John Wiley & Sons Ltd.

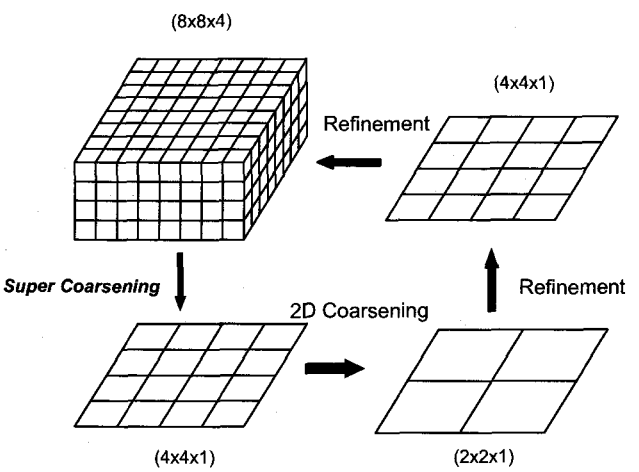

Figure 1: A super coarsening multigrid V-cycle

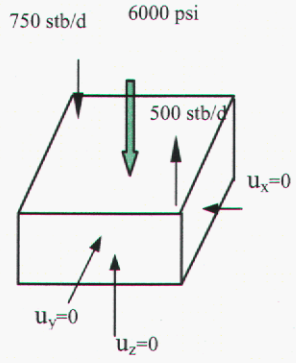

Figure 2: Constrained displacement boundary conditions for the model problem

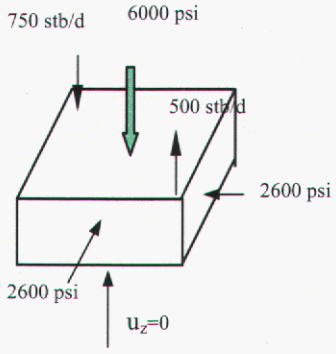

Figure 3: Unconstrained displacements boundary conditions for the model problem

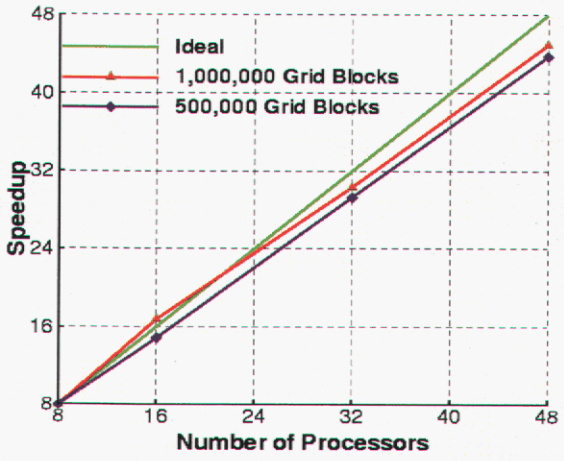

Figure 4: IPARS black-oil model parallel speedup

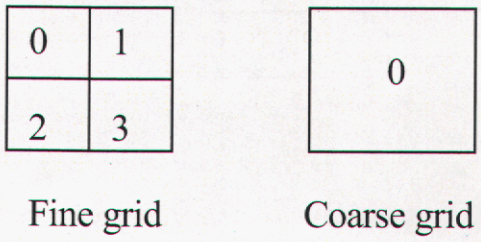

Figure 5: Coarse grid assignment in a parallel multigrid shame

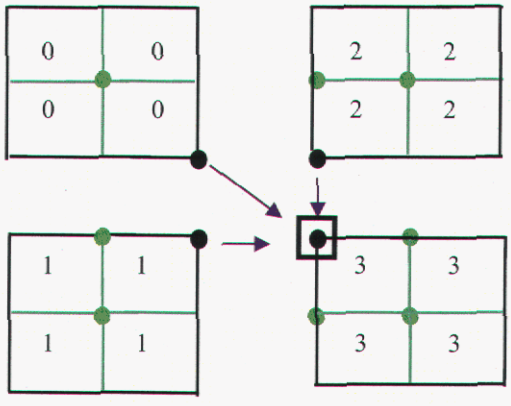

Figure 6: Special intra-grid level communications in the defect calculation

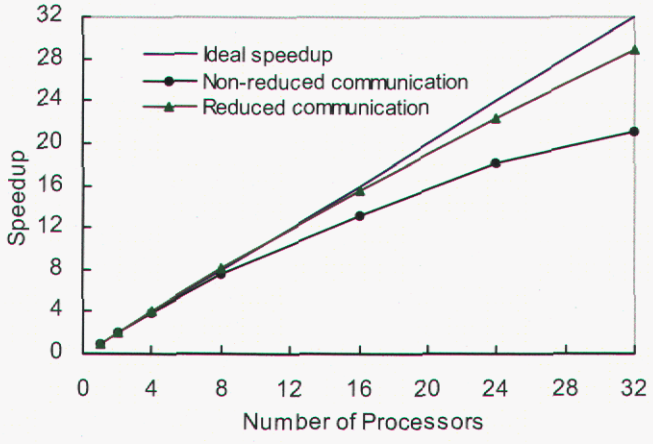

Figure 7: Parallel speedup with non-reduced and reduced communications for the single-phase problem

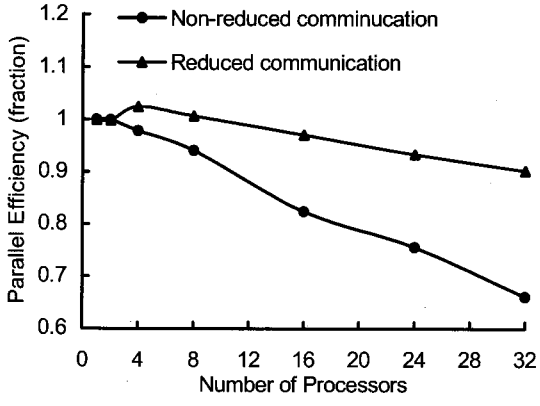

Figure 8: Parallel efficiency with non-reduced and reduced communications for the single-phase

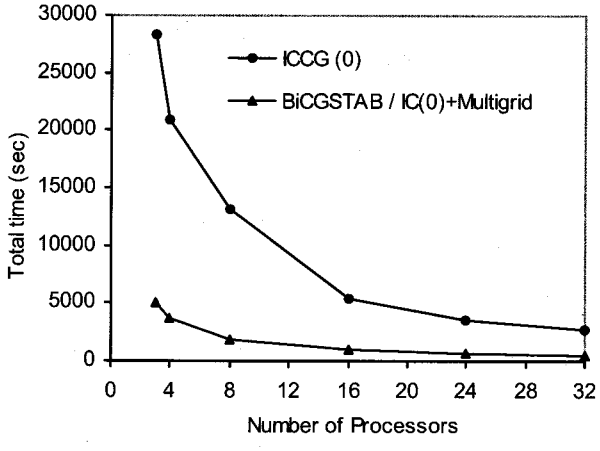

Figure 9: Total running time using IC (0) with / without multigrid cycles for the single-phase problem

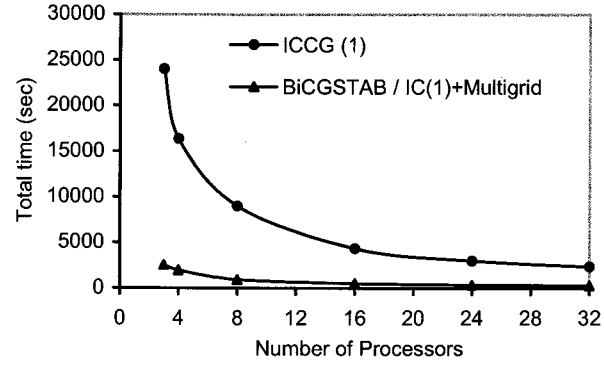

Figure 10: Total running time using IC (1) with / without multigrid cycles for the single-phase problem

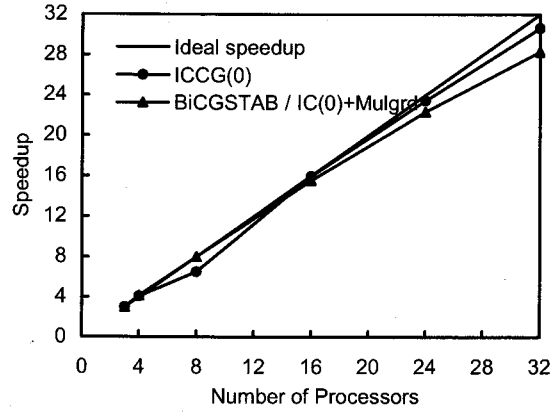

Figure 11: Parallel speedup for the single-phase problem

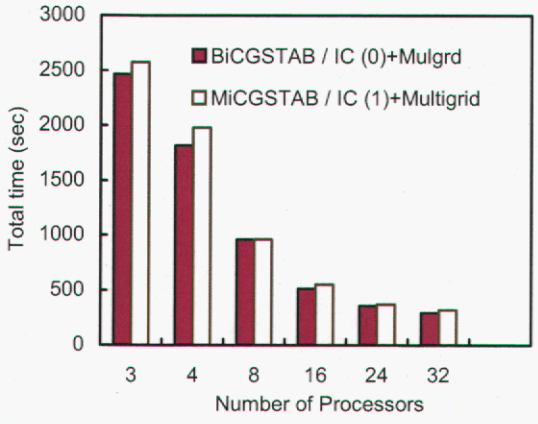

Figure 12: Total running time using IC (0) and IC (1) with multigrid cycles for the single-phase problem

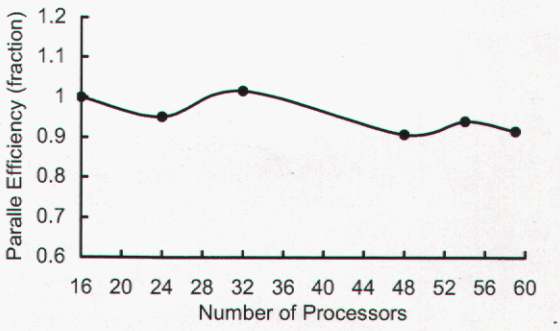

Figure 13: Parallel efficiency for the three-phase problem with 1.5 million grid blocks

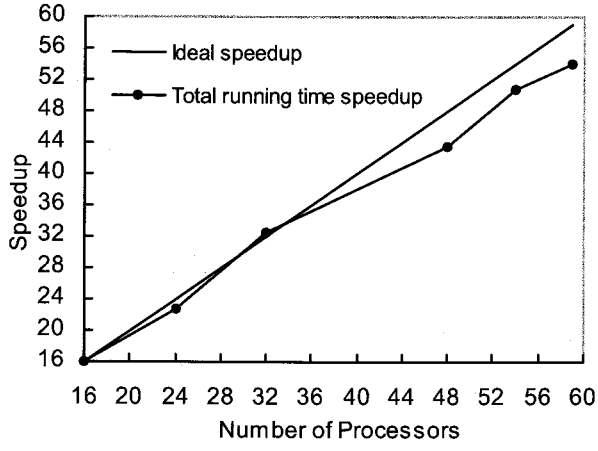

Figure 14: Parallel speedup for the three-phase problem with 1.5 million grid blocks

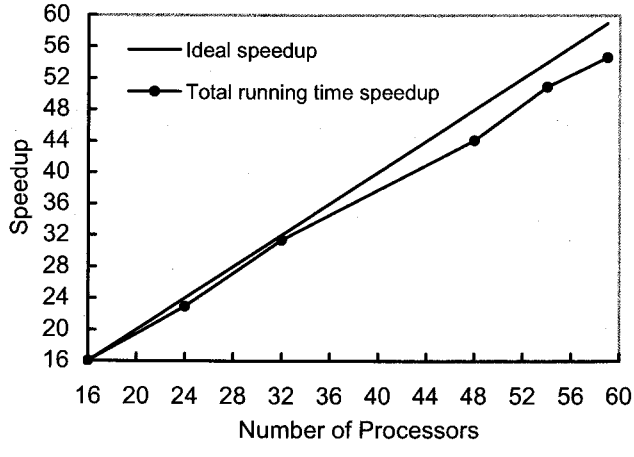

Figure 15: Parallel speedup for the three-phase problem with 1.5 million grid blocks, assuming stiffness matrix changes at each nonlinear iteration.

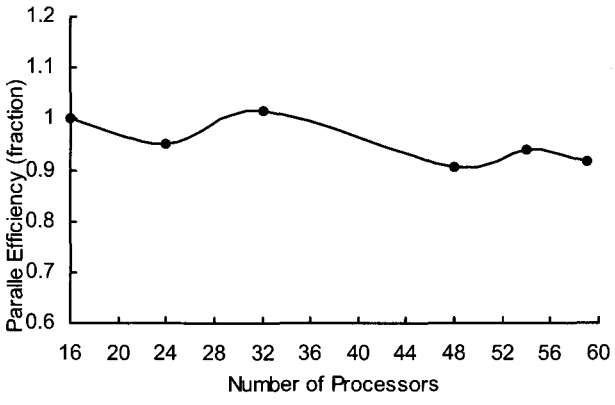

Figure 16: Parallel efficiency for the three-phase problem with 1.5 million gird blocks, assuming stiffness matrix changes at each nonlinear iteration

Distribution:

1	MS0323	D. L. Chavez	1011
1	MS0372	J. Jung	9127
10	MS0376	C. M. Stone	9127
1	MS9018	Central Technical File	8945-1
2	MS0899	Technical Library	9616
1	MS0612	Review & Approval Desk for DOE/OSTI	9612