SPE 66359

A Parallel Multi-Block Black-Oil Model in Multi-Model Implementation

Qin Lu, Landmark Graphics Corp., and Malgorzata Peszynska, and Mary F. Wheeler, Univ. of Texas at Austin

This paper was prepared for presentation at the SPE Reservoir Simulation Symposium held in Houston, Texas, 11–14 February 2001.

Abstract

We discuss the multi-block algorithm for an implicit black-oil model as implemented in a multi-phase simulator framework IPARS. The multi-block algorithm consists of decomposition of the simulation domain into multiple non-overlapping subdomains or blocks according to the geometric, geological and physical/chemical properties, distribution of wells, etc. Each block can have its own grid system, and the grids of the neighboring blocks can be non-matching on the interface, which allows, for example, for local grid refinement, or discrete fault or fracture modeling. Adjacent blocks are coupled across the interface by a set of conditions imposing continuity of the primary variables and component mass fluxes, which is realized with the use of special interface mortar variables. The resulting system is solved by an interface Newton procedure. Regularization techniques and preconditioners are proposed to improve the performance of the solver. The multi-block technique is effective and scalable, as shown by our numerical experiments. In addition, we present how the multi-block black-oil model has been used in the coupling of different physical models.

Introduction

The main thrust of this paper is to investigate accurate and efficient numerical techniques for simulation of flow and transport phenomena in porous media, which are of major importance in the environmental and petroleum industries. We propose to emphasize a novel numerical methodology, which is called the multi-block algorithm. This algorithm is based on decomposing the simulation domain into multiple subdomains (blocks) according to their geological, geometric and physical / chemical properties. One then applies the most efficient grid, numerical scheme and physical model in each subdomain so that the computational cost is reduced and accuracy is preserved.

Multi-block (also known as macro-hybrid) formulations [1-7,23] provide numerical models consistent with the physical and engineering description of the underlying equations. That is, the equations hold with their usual meaning on the subdomains and physically meaningful conditions are imposed on interfaces between the subdomains. In particular it is possible to couple different discretizations on non-matching multi-block grids as well as to couple different physical models in different parts of the simulation domain. These two features make the multiblock approach one of great computational interests.

For instance, in many applications the geometry and physical properties of the domain or the behavior of the solution may require the use of different grids in different parts of the domain, which possibly do not match on the interface. For example, the geology of the subsurface may involve modeling of faults, pinchouts and other internal boundaries, where the discontinuities of coefficients (e.g., mobilities) reduce the accuracy of traditional single-block algorithm near discontinuities. By splitting the domain into multiple subdomains along the boundaries of discontinuities, solutions in each subdomain may have smooth properties and local convergence rates are regained. Furthermore, locally refined grids may be needed for accurate approximation of local phenomena such as high gradients around wells.

More generally, multi-block decomposition can be induced by differences in the physical processes or mathematical models or by differences in the numerical discretization models applied to different parts of the simulation domain[8,9,10]. The overall computational cost can be reduced by selecting the most appropriate model in a given part of the reservoir. For example, only a single-phase or a two-phase model is needed for the aquifer part of the reservoir, while a black-oil model or a compositional model is necessary if the gas phase is present in a subdomain.

In this paper, we discuss the formulation and implementation of multi-block algorithm for an implicit black-oil model. This work represents a nontrivial extension of the multi-block algorithm for two-phase oil-water model[1] as, in particular, it needs to address numerical regularization issues arising at phase transitions. Next we briefly describe how the multi-block black-oil model is used in the multi-physics

coupling with the two-phase oil-water model. We also address the issues that arose during implementation in the IPARS framework, in particular, we discuss the parallelism of multi-model problem with the MPI multi-communicator and model-based load balancing strategies. In the end, numerical experiments that demonstrate the scalability of our approach are presented.

Multi-Block Black-Oil Model

The 3-dimensional reservoir domain Ω is divided into a series of n_{bl} non-overlapping sundomains (blocks) Ω_k, k = 1, …, n_{bl}, due to geological faults [11], geometry irregularities, variations of rock properties and physical / chemical properties of flow, and distribution and types of wells, etc. Each block has its own smooth rectangular grid. The grids are constructed locally and could be non-matching on the interfaces between neighboring blocks. **Figure 1** illustrates a typical geometry of a two-dimensional domain decomposition. Note that the interfaces between blocks are filled with "mortars". These are elements of a finite element space called *mortar space*, which is constructed on the 2-dimensional interface.

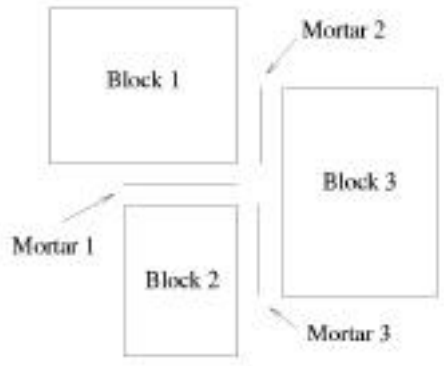

Figure 1. Two-dimensional multi-block domain decomposition.

Each block or subdomain Ω_k has an associated local physical model / solver which could be a single-phase model, a two-phase oil-water model, a black-oil model, a compositional model, etc. These models can be implicit, semi-implicit or explicit in time and they can use different linear and nonlinear solvers. In this paper we focus on the use of an implicit black-oil model as a subdomain solver. See Ref. 8 for the discussion related to various oil-water models coupled in the multi-block formulation.

Black-Oil Subdomain Formulation. The black-oil model is a three-phase (water, oil and gas) model describing the flow in a petroleum reservoir [12,22]. It is assumed that no mass transfer occurs between the water phase and the other two phases. In the hydrocarbon (oil-gas) system, only two components are considered. The *oil* component (stock-tank oil) is the residual liquid at atmospheric pressure left after differential vaporization, while the *gas* component is the remaining fluid.

One should distinguish the difference between *phase* and *component* [12,13]. For water, these two concepts are the same in the model since, by the above assumption, no mass transfer occurs between the water phase and the others. As in the above definition, oil and gas components refer to those hydrocarbon mixtures that would be in liquid (usually heavy hydrocarbons) and gaseous (usually light hydrocarbons) states, respectively, after surface separation. The corresponding pressure and temperature are normally called *standard conditions*, that is, usually (but not always) 14.7 psi and 60°F. The concept of *phase* would be meaningful only under a certain pressure and temperature.

To avoid confusion, we use capital letter subscripts to denote water, oil and gas component (W, O and G, respectively) and lower-case letter subscripts to denote water, oil and gas phases (w, o and g, respectively). Then mass conservation for water, oil and gas components are given by

$$\frac{\partial(\phi N_W)}{\partial t} = -\nabla \cdot U_W + q_W, \qquad \ldots\ldots(1)$$

$$\frac{\partial(\phi N_O)}{\partial t} = -\nabla \cdot U_O + q_O, \qquad \ldots\ldots(2)$$

$$\frac{\partial(\phi N_G)}{\partial t} = -\nabla \cdot (U_G + R_s U_O) + q_G. \qquad \ldots\ldots(3)$$

Note that the gas component flux consists of that in gas phase (free gas) and that dissolved in oil phase with gas-oil ratio denoted as R_o. At certain pressures (above bubble point) no free gas can exist and only two-phase (water-oil) conditions exist. In such conditions, N_G is proportional to N_O through the gas-oil ratio for the given pressure. On the other hand, if N_G is bigger than $R_{so}N_O$ at given pressure, then three phase conditions exist.

Darcy's law for multi-phase flow is used to calculate the mass velocity of component *M* in its corresponding phase *m*

$$U_M = -\frac{k_{rm}}{B_m \mu_m}(\nabla p_m - \rho_m g \nabla D) \qquad \ldots\ldots(4)$$

Because of volume balance, the saturations must satisfy the constraint

$$S_w + S_o + S_g = 1. \qquad \ldots\ldots(5)$$

Capillary pressures are defined as functions of saturations in following way

$$p_{cow} = p_o - p_w, \qquad \ldots\ldots(6)$$

$$p_{cgo} = p_g - p_o. \qquad \ldots\ldots(7)$$

The above system of equations is discretized for numerical solution: the subdomain is covered by a tensor product grid

and the equations are discretized using cell-centered differencing in space and backward Euler differencing in time with P_w, N_O and N_G chosen as primary unknowns [10]. Upwinding is applied to cell edge values for numerical stability [12,14]. We recall that the cell-centered differencing in space is equivalent to the expanded mixed finite element method of the lowest order Raviart Thomas space (*RT0*) on a rectangular grid, provided certain quadrature rules are applied [15,3,4] . The system of non-linear algebraic equations is linearized by Newton's method. A *GMRES* solver with multi-level preconditioner [16, 17] or an *LSOR* solver are avalaible to solve the linear system at every Newtonian step.

Interface Conditions for Black-Oil Model. The above subdomain black-oil model equations hold in the interior of each of the blocks. These need to be complemented by conditions ensuring mass and momentum conservation across the interface.

In this paper for simplicity we assume that the rock type is the same for all subdomains, therefore the same capillary pressure data is used. As a consequence, the pressure of each phase is continuous. In the multiblock formulation, in order to impose the continuity of pressure and to ensure the mass conservation, we need to make the pressures and the mass fluxes corresponding to two adjacent subdomains match on the interface. Specifically, on each interface Γ_{kl}, $1 \le k<l \le n_{bl}$, we impose

$$p_m|_{\Omega_k} = p_m|_{\Omega_l}, \quad m = w, g, \qquad \text{……..(8)}$$

$$(U_M \cdot \nu_{kl}) = U_M|_{\Omega_k} \cdot \nu_k + U_M|_{\Omega_l} \cdot \nu_l = 0, \quad M = W, O, G. \qquad \text{……...……(9)}$$

This approach is based on a domain decomposition algorithm for single-phase flow developed originally for conforming grids [2], and later generalized to non-matching grids coupled with mortars [6,7] and used for two-phase oil-water models [1, 8]. Note that as a consequence of Equation (8) and of the assumed homogeneity of rock type, if pressures of one phase, for example, water phase, is continuous, then the saturations and densities of all phases are continuous. This means that, in order to express momentum conservation, instead of pressures as in Equation (8), we can match a different set of variables, for example, all values of primary unknowns of the black-oil model:

$$p_w|_{\Omega_k} = p_w|_{\Omega_l}, \qquad \text{……..……(10)}$$

$$N_O|_{\Omega_k} = N_O|_{\Omega_l}, \qquad \text{……..……(11)}$$

$$N_G|_{\Omega_k} = N_G|_{\Omega_l}. \qquad \text{……..……..(12)}$$

In fact, it is this set of conditions that is used in the discretization of (8) - (9). We note however, that Assumptions (11)-(12) are only used for simplicity and that the general case of different rock types is easily handled.

In discrete form, conditions (8) - (9) are imposed in the weak sense using the mortar spaces as follows. Let M_h denote the mortar space on the union of interfaces $\Gamma = \bigcup_{k<l} \Gamma_{kl}$, and let $\mu_h = (\mu_h)$. The mortar space is constructed over a mortar grid using piecewise polynomial finite element shape functions. It was shown theoretically and numerically [3,5,6] that, for matching grids, we may use the same space on the interface as the normal trace of the velocity space in subdomain; but in the case of non-matching grids, in order to preserve optimal convergence (and, in some cases, superconvergence), the mortar space must consist of piecewise polynomials of one degree higher than the normal trace of the velocity space in the subdomains. Since we use cell-centered finite differences in space in each subdomain and the normal trace of the velocity space is piecewise constant, we employ a piecewise linear space M_h on the interface.

The discrete equivalent of Equations (1) - (7), which are defined on unions of all blocks with Equations (9)-(12) imposed on the interface is written as a nonlinear interface problem to be solved by iteration for the values of interface primary unknowns $\psi = (\overline{P_w}, \overline{N_O}, \overline{N_G}) \in \mu_h$. The nonlinear equations evaluate the jump in fluxes U_h corresponding to ψ as

$$B(\psi, \mu) = \sum_{k<l} \int_{\Gamma_{kl}} \sum_{M} (U_{hM}(\psi) \cdot \nu_{kl}) \mu_{kl} \, ds. \qquad \text{…..(13)}$$

To achieve mass conservation (or flux continuity), we need to determine ψ such that the jump in the fluxes is

$$B(\psi, \mu) = 0 \quad \forall \mu \in \mu_h. \qquad \text{……(14)}$$

We can also write $B(\psi,\mu)=(\boldsymbol{B}(\psi),\mu)$ where the physical meaning of $\boldsymbol{B}(\psi)$ is the square of the jump of the fluxes across all the interfaces corresponding to the given value of ψ. Note that the value of U_h at each block k is delivered by a subdomain solver.

The solution to the interface problem (14) is equivalent to the solution on all subdomains with mass and momentum preserved across the interface.

Interface Solver. The system of nonlinear equations on the interface (14) is in our implementation in IPARS solved by an inexact Newton method [1,7,16]. Each Newtonian step solution is computed by a forward difference GMRES iteration. Each GMRES iteration involves solving subdomain problems with the Dirichlet boundary condition provided by the interface code. The subdomain problems in turn deliver values of fluxes across each interface. Additionally, two inexpensive projection steps are necessary: the first one projects ψ from interface mortar grid onto the local subdomain grids so it could be used as the Dirichlet value, and the second one

projects normal fluxes U_h computed by the subdomain solver from the local grids onto mortar grid. Details of the algorithm can be found in References 1,5,7,16.

The interface algorithm is an iterative procedure. It assumes an initial guess for the values of primary variables ψ on the interface to be used after projection as a Dirichlet boundary condition on subdomains. The subdomain problems are then solved and the jump in the component fluxes $\boldsymbol{B}(\psi)$ is calculated on the interface. If the jump is less than a given tolerance, the current time step is completed; otherwise, the interface unknowns ψ are updated and the procedure is repeated until convergence.

To facilitate the above steps in implementation, additional memory needs to be allocated for each block for storing boundary layer information such as values of primary variables, transmissiblities, component mass fluxes, etc. Analogous to the interior cells, upwinding is applied to the boundary values for flux calculations. One advantage of this approach is that the Dirichlet boundary conditions can be included in the Jacobian matrix and the residuals of the block without modifying the subdomain code [10].

Regularization of Gas-Phase Relative Permeability on Interface. One major issue that arises in modeling on the interface of the three-phase black-oil model is the treatment of the gas phase when it reaches the interface. The gas component consists of free gas and of gas dissolved in the oil phase. The gas phase can be either immobile, when gas saturation is less than the residual gas saturation when the gas is immovable even if the gas saturation is greater than zero, or, the gas phase can be mobile, when the gas saturation is greater than the residual value [13,18]. The former situation may arise when the interface is near production wells. In this case, it is assumed that no gas phase exists initially on or near the interface. But the gas phase may be created later near the production wells because of a pressure drop. The gas phase zone then expands as a function of time, and eventually may reach the interface. Note that the gas phase is created because of the pressure drop and not as a result of transport.

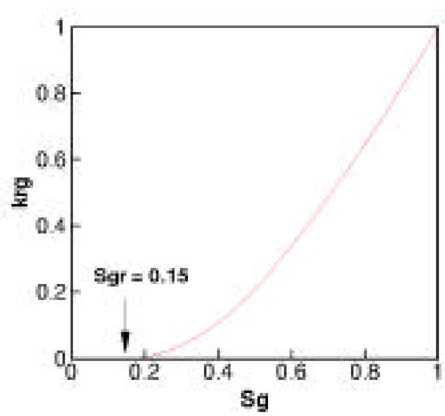

Figure 2. A typical gas relative permeability curve.

Because of the non-zero residual gas saturation, when the front of the gas phase zone first reaches the interface, the physical problem becomes degenerate and the Jacobian matrix on the interface is no longer diagonally dominant. Consequently the interface solver may not converge to a physical solution and the simulation may fail.

Figure 2 is a typical gas relative permeability curve, where S_{gr} is the residual gas saturation. As we can see from this figure, when the gas saturation S_g increases gradually from zero to the residual gas saturation S_{gr}, the concentration of the gas component increases on the interface, while the gas component mass flux (and therefore the jump in this flux across the interface) remains unchanged. Since neither the gas component mass flux with free gas (or gas phase) nor that with the dissolved gas changes (note that the amount of gas component dissolved in oil phase is unchanged since the pressure is unchanged), the change of the jump in the gas flux is zero on the interface. In this case, we have

$$\frac{\partial [U_G \cdot \nu]}{\partial \nu_G} \approx 0. \quad \ldots\ldots(15)$$

where $[U_G{\cdot}\nu]$ denotes the mass flux jump of gas component across the interface. Numerically, this partial derivative is a diagonal term in the interface Jacobian matrix. The zero diagonal term makes the matrix no longer diagonally dominant.

Our solution to this problem is to regularize the gas relative permeability curve [10]. We enforce a small slope dk_{rg}/dS_g between $S_g = 0$ and $S_g = S_{gr}$, as shown in the right plot of **Figure 3**, so that the derivative in Equation 12 will be greater than zero, and hence the interface Jacobian matrix will be better conditioned.

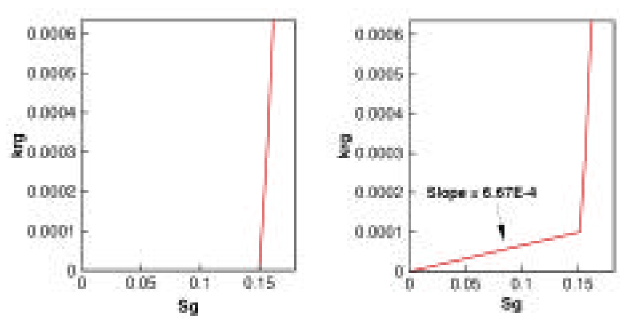

Figure 3. Gas relative permeability curve near residual gas saturation. Left: initial curve. Right: regularized curve.

The admissible sizes of the slope are case dependent. It is necessary to choose a slope sufficiently large for stabilization but not too large for changing the physics of the problem. Our experience indicates that the recommended range would be between 10^{-2} to 10^{-4}. **Figure 4** shows the comparison of the water pressure distribution of a one-dimensional case when the problem is simulated with a single block (no interface) and

with the multi-block (two blocks connected with an interface). **Figure 5** shows a comparison of gas saturation distributions. The dimensionless length of the case was 1, with grid size of 50 x 1 x 1. The initial gas saturation was set to zero and the gas relative permeability is shown in **Figure 2**, with a residual gas saturation S_{gr} = 0.15. A production well was set at x_d = 0.05, an injection well at x_d = 0.95, and the interface at x_d = 0.2. The results were after 100 days of simulation. As we see, the slopes ranging from 10^{-2} to 10^{-4} work well. But with slopes beyond this range, the simulations crashed on the interface flash calculations due to bad primary variable values [10].

Note that the regularization is only necessary on the interface. In the subdomains, the original gas relative permeability is used.

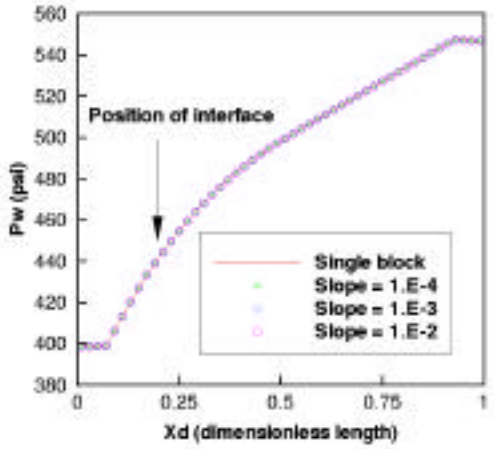

Figure 4. Effect of gas relative permeability regularization on water pressure distribution of a one-dimensional case with two blocks.

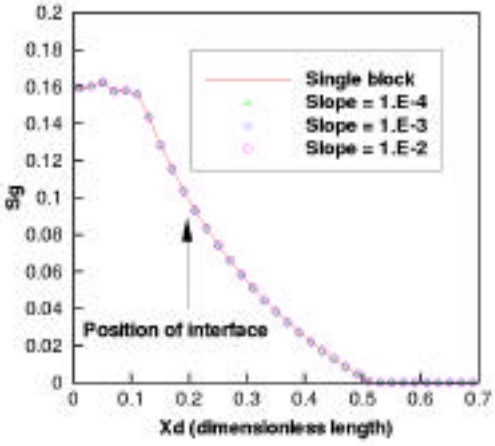

Figure 5. Effect of gas relative permeability regularization on gas saturation distribution of a one-dimensional case with two blocks.

Coupling Different Physical Models

In a given part of the reservoir, we may want to individually select the most appropriate physical model to be coupled to other parts and their assigned models through the multi-block algorithm. Each model should be as simple as possible yet it should include necessary elements to describe the relevant physical and chemical phenomena of the flow and transport in the associated subdomain. This strategy may lead to significant reduction in the overall computational especially for multi-million grid size cases while the accuracy of the simulation is retained [9,10].

For example, in the aquifer part of the reservoir, or in the part with water-oil two-phase system and no free gas created during the whole operation, a two-phase code is appropriate; while in the part with gas cap, or with production wells around which free gas may be created because of low bottom hole pressure, a black-oil or a compositional code is necessary. Here we discuss the coupling of the implicit black-oil model and an implicit two-phase oil-water model. In this coupling it is essential that conservation quantities be preserved across the interfaces. With the black-oil model discussed above, we briefly review the oil-water model.

Oil-Water Model. The immiscible two-phase oil-water model is in some sense a subset of the black-oil model in which gas phase and component are absent. In fact, it is governed by the equations

$$\frac{\partial(\phi S_w)}{\partial t} = -\nabla \cdot u_w = q_w, \qquad \ldots\ldots (16)$$

$$\frac{\partial(\phi S_o)}{\partial t} = -\nabla \cdot u_o = q_o. \qquad \ldots\ldots (17)$$

Darcy's law for multi-phase flow is used to calculate the mass velocity of phase *m*, note that the phase densities are assumed to have explicitely exponential character:

$$u_m = -\frac{k k_{rm}}{\mu_m} \rho_m (\nabla p_m - \rho_m g \nabla D). \qquad \ldots\ldots (18)$$

The saturations must satisfy the constraint

$$S_w + S_o = 1. \qquad \ldots\ldots (19)$$

Capillary pressure is defined as function of saturation in following way

$$p_{cow} = p_o - p_w. \qquad \ldots\ldots (20)$$

Oil-Phase Partitioning Across Interface. A natural question from the coupling of these two models arises: how to deal with the gas component on the interface, now that on one side of

the interface is a three-phase black-oil model, and on the other side a two-phase oil-water model, which assumes no gas phase? The natural strategy is as follows: select subdomains appropriately and partition the oil phase on interface into oil and gas components [10].

In the decomposition into subdomains, we assume that in the time period being simulated no gas phase (free gas) appears on any interface. This can be achieved by a careful decomposition of the domain. In other words, the interfaces between black-oil models and two-phase oil-water models are selected to be below the gas cap and a sufficient distance away from the production wells in which free gas may be created because of pressure drops. This means that the reservoir pressure is above the *bubble-point pressure* [13] on and near the interfaces. However, the gas component dissolved in the oil phase is still allowed to exist on interfaces. The selection of domains can of course be adaptively changed to meet these two criteria.

The oil-water model describes the flow of water and oil *phase*, but does not distinguish between the *oil-phase* and the oil-component, while the black-oil model takes the oil component and the gas component in a liquid state as the oil-phase. Therefore our second strategy involves the splitting of the oil phase on the interfaces into oil and gas components. This is achieved by constructing appropriate correlations relating *component* with *phase*. No difficulties arise in the aqueous phase, since the water phase consists only of the water component in both models.

We define the oil phase in the oil-water model as saturated oil with a constant non-zero gas-oil ratio R_o, which can be chosen to be close to the solution gas-oil ratio R_{so}. Therefore, on the interface between these two models, we need to partition the oil phase concentration N_o of the oil-water model into the oil component concentration N_O and the gas component concentration N_G in order to match the value black-oil model, according to the mass fractions of these two components in the oil phase. Without considering unit conversions at this point, we define

$$N_W\big|^B_{\partial\Omega} = N_w\big|^H_{\partial\Omega}, \quad \ldots\ldots(21)$$

$$N_O\big|^B_{\partial\Omega} = N_o\big|^H_{\partial\Omega}\frac{\rho_{OS}}{\rho_{OS} + \rho_{GS} R_o}, \quad \ldots\ldots(22)$$

$$N_G\big|^B_{\partial\Omega} = N_O\big|^B_{\partial\Omega} R_o. \quad \ldots\ldots(23)$$

Multimodel Interface Coupling. The primary variables of the subdomain models, the implicit oil-water model and the implicit black-oil model, as well as those of the interface, are summarized in **Table 1**.

The primary variables on the interface are chosen to be something between these two models, which are water pressure P_w and oil phase concentration N_o. We only solve for two primary variables, since there are only two phases existing on the interface. The value of N_G on interface can be calculated from N_O of the black-oil model as the gas-oil ratio R_o is assumed to be constant on the interface. As stated in above, in order to match the *oil component* flux between the oil-water model and the black-oil model, we may assume R_o to be a non-zero constant on the interface, and the *oil phase* in the oil-water model is saturated oil with the same constant gas-oil ratio. We only match the oil component and the water component fluxes across interface. The gas component flux is matched automatically because the gas-oil ratio is constant on the interface.

Oil-water model	Black-oil model	Interface
P_o	P_w	$\bar{P}_w$
N_o	N_O	$\bar{N}_o$
	N_G	

Table 1. Primary variables for the subdomain models and interface coupling implicit oil-water model and black-oil model.

The interface formulation can be summarized as follows: Solve Equation (14), where for $\psi = (\bar{P}_w, \bar{N}_o) \in M_h$, the primary variables on the interface, we define

$$B_{M,\alpha}(\psi) = \sum_{k=1}^{n_B} \sum_{e_{kl} \subset \Gamma} \left(U_{\alpha M}(\psi) \cdot n_{kl}\, \mu_{kl} \right)_{\Gamma} \quad \ldots\ldots(24)$$

The system of the nonlinear equations on the interface can be solved by an inexact Newton method as stated previously.

Similar to the splitting of oil phase concentration on the interface, we need to split the oil phase flux of the oil-water model into oil component flux and the gas component flux to match the black-oil model, using the correlations analogous to Equations (21) - (23).

Implementation Issues

All implementation aspects and results presenteded in this paper are built upon the framework *IPARS* (Integrated Parallel Accurate Reservoir Simulator) [19,1,7,23], which has been developed at Center for Subsurface Modeling, University of Texas at Austin. It is a new generation framework for developing and running parallel reservoir simulators, solving problems involving a million or more grid elements, and supporting a variety of physical and numerical models plus simplified well management. This framework provides infrastructure common to most simulators, such as parallel message passing, memory management, grid generation, free-form keyword input, formated output for visualization, user units specification, general purpose 2-D function utilities, table lookup, etc., so that the developer only needs to code the physical model of interest.

The MPI portable message passing interface [20] is currently used to manage the message passing in parallel computations. The framework is portable, running on platforms such as IBM

and SGI workstations, Cray T3E, IBM SP, single PC under Linux, Windows and DOS, and PC clusters under Linux.

The framework can have multiple (fault) blocks (or subdomains), each of which may have associated its own physical model. The neighboring blocks are connected via an interface using mortar spaces approach presented here or *dual approach* discussed in Ref. 23. Unit conversion between different models may be necessary in the coupling of different models.

The linear solvers for different physical models can be either different or the same. A parallel GMRES solver[17,16] has been extended for solving multiple models simultaneously. The basic idea for the extension is to expand the work space (e.g., the basis for the Krylov space used in solving a linear system[16]) from a scalar to an array, so that each model has its own entry of work space.

The parallelism is a delicate and interesting issue to tackle. Different from the traditional single model simulator, the multi-model problem is actually a MIMD (multiple instruction multiple data) problem. We use multiple MPI communicators in the implementation [20,10]. The processors are split into multiple groups (or *communicators*) so that each physical model has its own communicator. Within a communicator, the adjacent processors can exchange boundary information that is necessary for parallel computation. The message passing between different communicators is also allowed.

Load balancing is important for parallel efficiency. The traditional (single model) load balancing strategy is that the grid cells are divided evenly between processors. In multi-model cases, a processor may be shared by more than one model. This may cause loss of parallel scalability because of synchronization problem between different models. Therefore we suggest using two criteria to improve load balancing (or the model-based load balancing) in multi-model implementation [10,8,9].

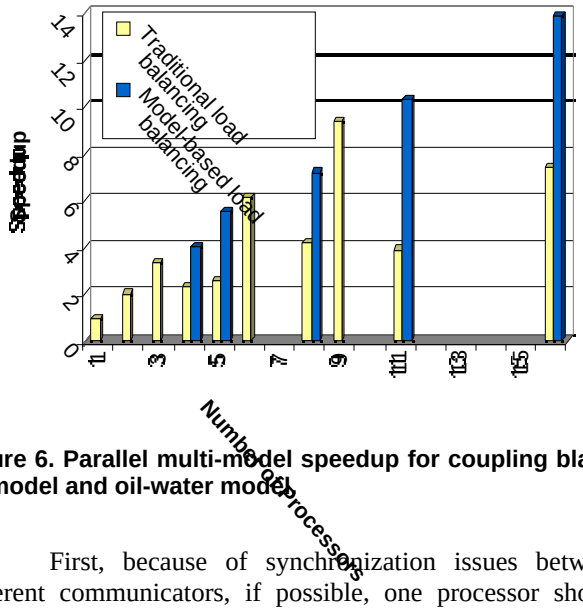

Figure 6. Parallel multi-model speedup for coupling black-oil model and oil-water model.

First, because of synchronization issues between different communicators, if possible, one processor should never handle more than one model / code; second, the number of processors assigned to each model should be weighted by its simulation speed, provided the first criterion is satisfied. **Figure 6** demonstrates the comparison of speedup with these two strategies for a three-block case coupling the black-oil mode and the oil-water model. With model-based load balancing strategies, ideal or even superlinear speedup, can be obtained.

Computational Examples

Example 1: The First Multi-Block Black-Oil Case. Our first example was a horizontal reservoir with geometry shown in **Figure 7**. One injection well and one production well were completed at the two ends of the reservoir, with piecewise linearly changed bottom hole pressure with respect to the simulation time. The size of the reservoir was 1600ft x 500ft x 15ft, in two horizontal directions and one vertical direction, respectively (we will follow the same order in later discussions). The original water pressure was 2000 psi at the top of the reservoir. The permeability field was layered, with 100 md and 50 md in the horizontal and vertical directions, respectively, at the top and the bottom layers, while the horizontal permeability was 50 md at the middle layer. The rock at the northeast and southwest corners was impermeable. The reservoir is split into three blocks according to the geometry and the location of wells. At the blocks (Block 1 and Block 3) containing wells, which were at southeast and northwest corners, we applied fine grid in each block, with grid size 10 x 10 x 3 and the size of each grid cell 20ft x 20ft x 5ft. Block 2, the largest of the three blocks, contained no wells and the flow in this block was smooth, therefore we used a coarse grid of size 10 x 10 x 3. Here each grid cell was of size 160ft x 50ft x 5ft.

Two interfaces were set between the neighboring blocks with grid size of the mortar space to be 1 x 3 (horizontal direction and vertical direction, respectively); continuous piecewise linear basis was applied in the mortar space on the interface.

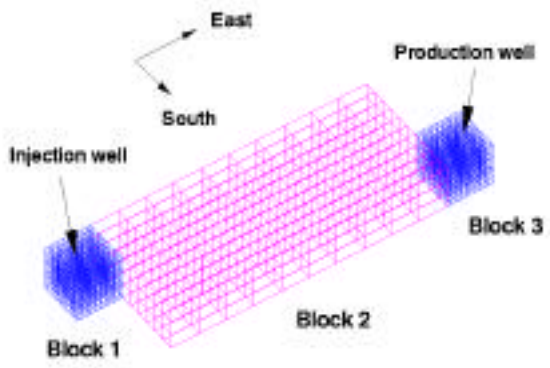

Figure 7. The geometry and grid system of the reservoir used by Example 1.

We ran this case with the multi-block black-oil code. It normally took two Newtonian steps on interface, and eighteen subdomain evaluations on average per time step. We observed that the jump in fluxes across the interface dropped quadratically with the interface Newtonian iterations, which is optimal for Newton's method.

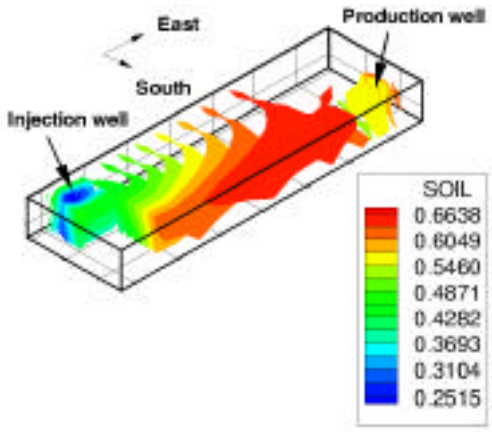

Figure 8. Oil saturation contour for the horizontal reservoir at the 1600th day given by multi-block black-oil model.

For the comparison purposes, we also ran this case using the single block black-oil model (no interface) with fine grid (20ft x 20ft x 5ft) over the entire reservoir, with some of the elements keyoud out at the corners to match the original geometry. **Figure 8** and **Figure 9** show the oil saturation contours obtained from these two approaches. **Figure 10** and **Figure 11** compare the gas saturation profiles between these two apporaches.

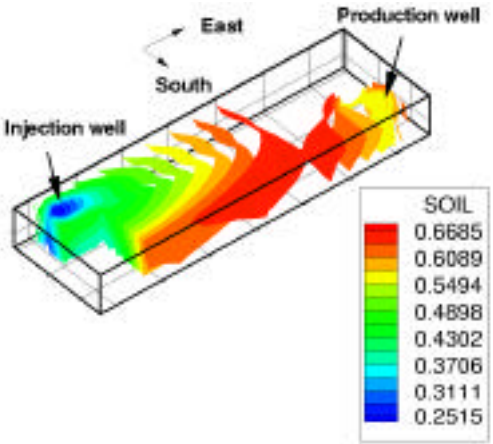

Figure 9. Oil saturation contour for the horizontal reservoir at the 1600th day given by single-block black-oil model.

We observed that both the oil and the gas fronts cross the interfaces without any difficulties ¾ gas relative permeability regularization is necessary here. The contours from these two approaches look very similar although they are not exactly the same. The small difference arises from two sources. First, there is a small local discrepancy between the numerical results given by the interface algorithm and those for the single block. This is because the flux and pressure continuity is imposed in a weak sense only and also since the interface Newtonian iteration stops when some non-zero tolerance criterium is satisfied. Second, since the grid sizes used by the two approaches are different, the graphic tool (Tecplot [21]) has to provide different results because of the interpolation procedure used in plotting. This is a typical difficulty in showing multi-block results even when computed results are continuous as the graphic tool plots the contours separately for each block.

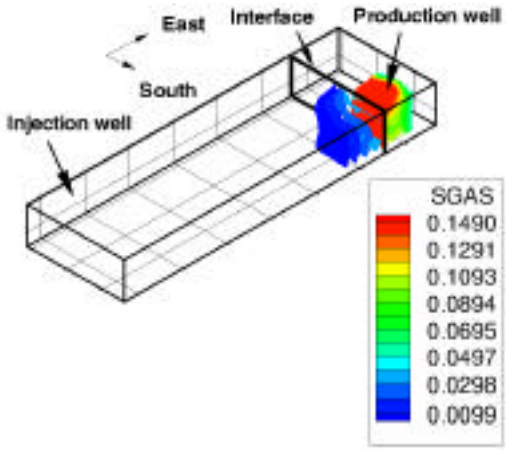

Figure 10. Gas saturation contour for the horizontal reservoir at the 1600th day given by multi-block black-oil model.

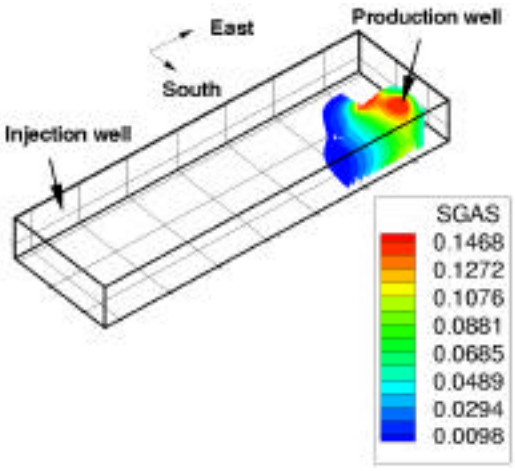

Figure 11. Gas saturation contour for the horizontal reservoir at the 1600th day given by single-block black-oil model.

Another way of comparing the results of different simulations is presented in **Figure 12,** which shows the injection and production flow rates obtained from the multi-block and from the single block simulators; they match each other perfectly.

Finally, we consider the computational efficiency of the multi-block approach. As we can see in **Table 2**, the multi-block simulation ran faster by a factor of 1.43 than the single-block black-oil model simulation. Concluding this example we note that by using the multiblock algorithm to handle local grid refinement and coarsening we reduced the computational without sacrificing accuracy.

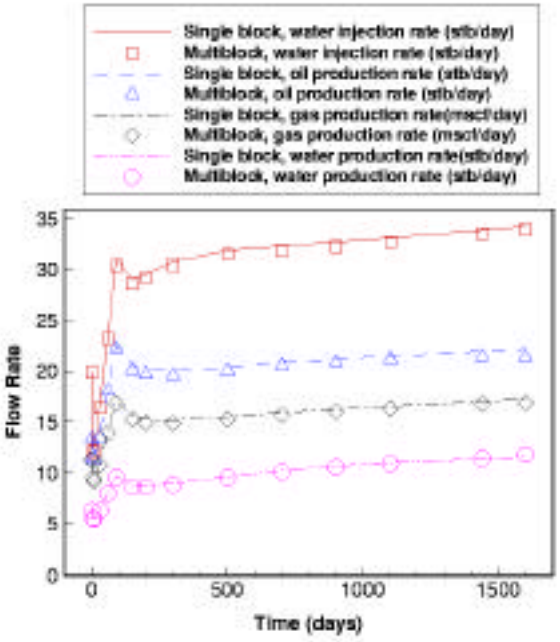

Figure 12. Comparison of injection and production flow rates for Example 1 given by multi-block and single-block black-oil model.

	Multi-block Black-oil	Single-block Black-oil
Total CPU time (sec)	2438	3486

Table 2. Comparison of CPU time for the case used in Example 1 for simulation time of 1600 days.

Example 2: The Second Multi-Block Black-Oil Case. This was a dipping rectangular reservoir, whose geometry and permeability distributions are shown in **Figure 13**.

The size of of the reservoir is 880ft x 880ft x 3ft. It has a dip angle $\alpha = 2.9°$ from a horizontal direction. It is characterized by a very high permeability contrast; the permeability of most of the domain was 2 md in all directions, while there was a high permeability streak along two sides of the reservoir with permeability of 500 md, except for a layer at the middle of the streak, which had permeability of 100 md. An injection well and a production well were located at the two opposite corners of the high-permeability streak. The initial water pressure and water saturation at the top of the reservoir were 2300 psi and 0.22, respectively.

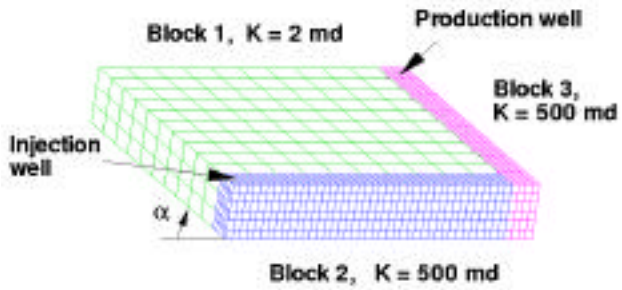

Figure 13. Geometry, permeability distribution, wells location, multi-block decomposition and gridding of the dipping reservoir used by Example 2.

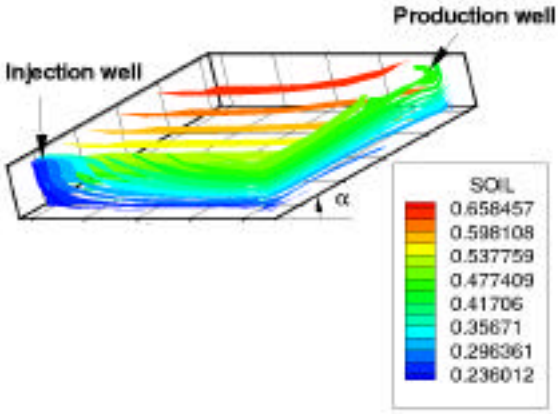

Figure 14. Oil saturation profile at the 1000th day given by multi-block black-oil model.

For this case, it was desirable to apply fine grid in regions with high permeability and wells, and use coarse grid in the remaining part of the reservoir. Therefore, we split the reservoir into three blocks, with three interfaces set between neighboring blocks, see **Figure 13**. The size of a fine grid cell was 20ft x 20ft x 5ft, and that for a coarse grid cell was 80ft x 80ft x 10ft.

The mortar grid between Block 1 and Block 2, and that between Block 1 and Black 3 were 3 x 1 (horizontal direction and vertical direction, respectively), the grid size of the mortar space between Block 2 and Block 3 was 1 x 1. The basis for the mortar spaces was taken to be continuous piecewise linear. The oil saturation profiles at the 1000th day are shown as **Figure 14**.

Again, we simulated the same problem with the single-block black-oil model for comparison. We had to use fine grid all over the reservoir. As a result, the multi-block approach was almost twice as fast as the single-block approach, as shown in **Table 3** while the well flow rates showed excellent matching shown in **Figure 15**.

	Multi-block Black-oil	Single-block Black-oil
Total CPU time (sec)	2773	5467

Table 3. Comparison of CPU time for the case used in Example 2 for simulation time of 1000 days.

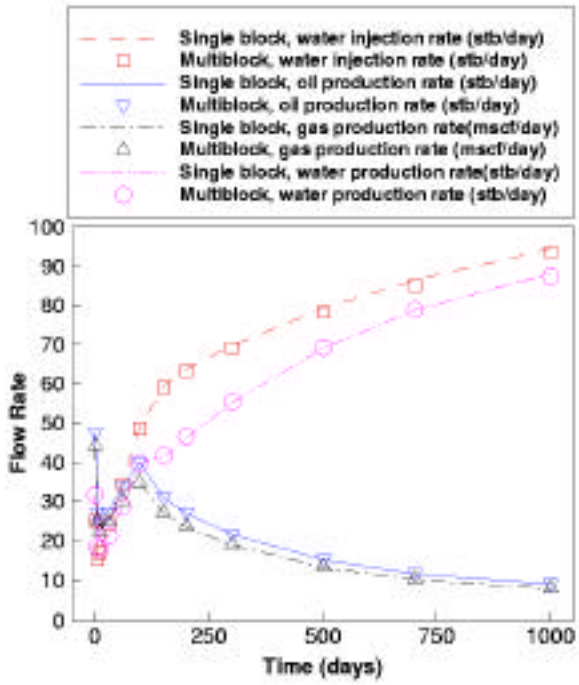

Figure 15. Comparison of injection and production flow rates given by multi-block and single-block black-oil model for the case used by Example 2.

Example 3: Multi-Model Case. This is a reservoir with the geometry suitable to split into three blocks, with two interfaces between neighboring blocks. The domain decomposition, gridding, permeability distribution, well locations and assignment of physical models are shown in **Figure 16**. The two interfaces were between Block 1 and Block 2, and between Block 1 and Block 3, respectively. The fault between Block 2 and Block 3 was impermeable. The size of block 1 was 800ft x 800ft x 48ft (two horizontal directions and one vertical direction, respectively), the size of Block 2 was 9600ft x 6400ft x 48ft, and 6400ft x 9600ft x 48ft for Block 3. Grid sizes of the three blocks were 20 x 20 x 8, 18 x 13 x 4 and 13 x 19 x 4, respectively. The grid cell size in Block 1 was chosen as relatively small (40ft x 40ft x 8ft per cell), since this block represented the production zone with seven wells. In contrast, coarse grid (160ft x 160ft x 16ft per cell) was used in most of Block 2 and 3 except around four injection wells (80ft x 80ft x 16ft per cell). The total number of grid cells was 5124.

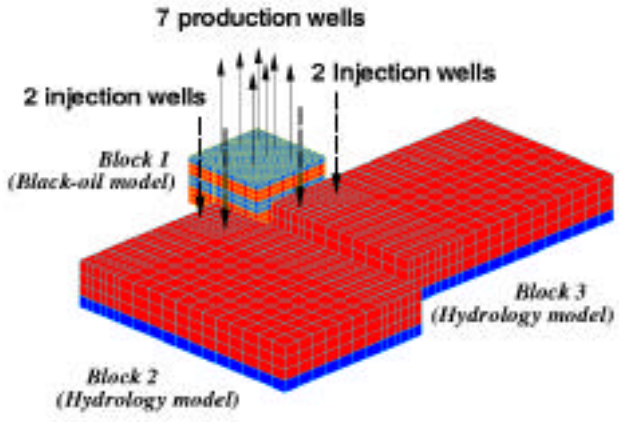

Figure 16. Geometry, domain decomposition, gridding, permeability distribution, wells locations and assignment of models of the reservoir used by Example 3.

There were total eleven wells, including two oil rate specified production wells, five bottom hole pressure specified production wells, and four bottom hole pressure specified water injection wells used to maintain the pressure. The permeability was layered as shown in Figure 16 with light color denoting high permeabilities (200 md) and dark color marking low permeabilities (20 md or 50 md). Block 1 had high initial oil saturations, ranging from 0.5 to 0.8, with a gas cap dominating the top 8 ft, while the most parts of Block 2 and 3 were aquifers. The initial oil pressure at the top of the reservoir was 2000 psi.

According to the geological and geometric characteristics of the reservoir, and the well types and locations, we assigned the black-oil model in Block 1, and the oil-water model in Block 2 and Block 3.

The oil phase in the oil-water model and on interface was assumed to be saturated with constant gas-oil ratio, which was about 90% of the solution gas-oil ratio under the average pressure on interface.

We used minimum mortar grid which is 1 x 1. It usually took two Newtonian iterations on interface to converge to the

given tolerance, with the jump in fluxes decreased quadratically, which is the optimal convergence rate of Newton's method when a good enough initial guess is given. The initial time step was 0.5 day and it quickly increased to 20 days.

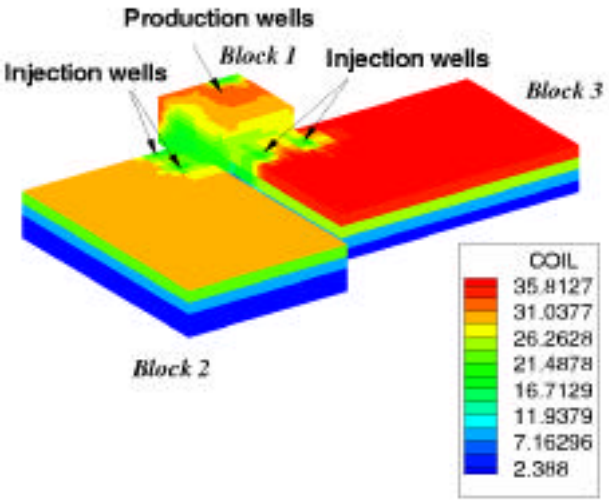

Figure 17. Oil component concentration profile of the reservoir with three blocks at t = 500 days given by multi-model.

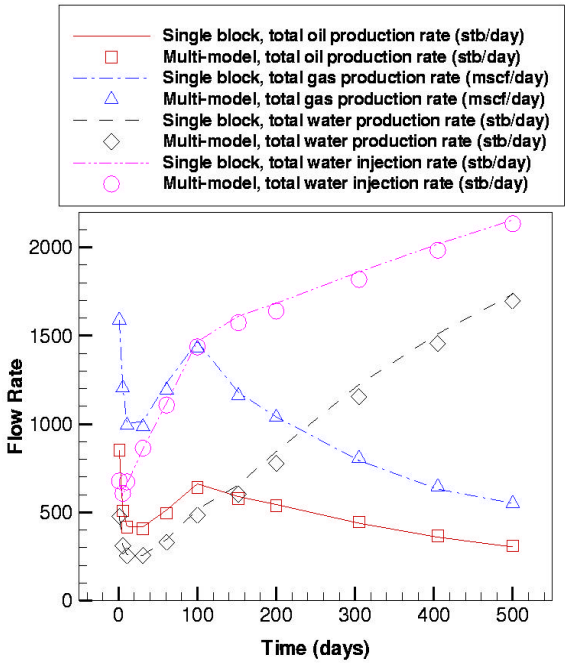

Figure 18. Comparison of total well flow rates of the reservoir in Example 3 given by multi-model and single-block black-oil model.

The oil component concentration profiles at the 500th day obtained from the multi-model are shown in **Figure 17**. The effect of production as well as of waterflooding is obvious in that the water front has passed through the interfaces and the gas cap shrank back to the far corner of Block 1 away from the other two blocks.

For comparison purposes, we simulated the case with the single-block black-oil model with uniform fine grid (40ft x 40ft x 8ft for each cell) in the entire reservoir. The total number of grid cells was 41600 in this case. The well flow rates are compared at **Figure 18**. The total oil production rate and total gas production rate got excellent agreements between the multi-model and the single-block black-oil model. However, the total water production rate and total water injection rate had about 5% difference between the two approaches, which was expected, since the size of the grid cells around the injection wells were different between these two simulations. The fact that the injection wells were handled by the oil-water model in the multi-model approach, and that they were handled by the black-oil model in the single-block black-oil model approach may have contributed to part of the differences.

As shown in **Table 4**, the multi-model achieved dramatic reduction in computational time by a factor of 7.07 compared to the single-block black-oil model. This is clearly an effect of using the faster model (oil-water model) in two larger blocks combined with grid coarsening there. Without the latter, the cost reduction could be less dramatic.

	Multi-model	Single-block Black-oil
Total CPU time (sec)	4053	28639

Table 4. Comparison of CPU time between multi-model and single-block black-oil model for simulation time of 500 days in Example 3.

Conclusions

1. The multi-block algorithm is a promising method to couple the appropriate grids and physical models, based on decomposing the simulation domain into multiple subdomains (blocks) according to the geological, geometric and physical / chemical properties. The flux continuity across the interface between adjacent blocks is achieved accurately. Through a comparison study between the multi-block / multi-model and the traditional single-block / single-model, it has been demonstrated that the computational cost is reduced and accuracy is preserved when multi-block couplings are used.

2. In the multi-block black-oil model, the regularization of the gas relative permeability is applied on the interface to avoid the degeneration of the physical problem which may occur when the gas phase saturation on the interface is greater than zero but less than the residual value.

3. In the coupling of the implicit black-oil model and the implicit oil-water model, proper domain decomposition is needed to ensure that the gas phase never reaches the interface in the time period being simulated; oil phase partitionning is

applied to match the oil component and gas component respectively across the interface between the two models.

4. The MPI multi-communicators have been used in parallel implementation of the multi-model problem. The processors are divided into multiple groups or communicators, so that different groups can run different models simultaneously. This implementation allows an arbitrary number and combination of models, and is independent of the interface code. Model-based load balancing strategies have been applied to achieve ideal speedup; in some cases, even superlinear speedup can be obtained.

Nomenclature

B_m = *formation volume factor of phase m, bbl/stb*
$B(\psi,\mu)$ = *the jump in fluxes across the interface, ft/sec*
D = *depth, ft*
g = *gravity magnitude, psi-ft^2/lb*
k_{rm} = *relative permeability of phase m*
K = *absolute permeability, darcy*
n_{bl} = *number of subdomains*
n_c = *number of components*
M_h = *discrete space of mortar scalar functions on all interface*
N_m = *concentration of phase m, lb/ft^3*
N_M = *concentration of component M, or stock tank volume of component M per unit pore volume, stb/bbl*
$\overline{N}$ = *interface (mortar space) component concentration, stb/bbl*
P_{cgo} = *capillary pressure between gas and oil phases*
P_{cow} = *capillary pressure between oil and water phases*
P_m = *pressure of phase m, psi*
$\overline{P}$ = *interface (martor space) pressure, psi*
q_M = *mass rate of component M as source term, ft^3/sec*
R_o = *gas-oil ratio, mscf/scf*
R_{so} = *solution gas-oil ratio, mscf/scf*
S_m = *saturation of phase m*
t = *time, sec*
U_m = *flux of phase m, lb/sec*
U_M = *flux of component M, ft/sec*

α = *angle, degree*
ψ = *primary unknowns on the interface*
Φ = *porosity*
ρ_m = *density of phase m, lb/ft^3*
ν = *outer unit normal vector*
ν_l = *outer unit normal vector on face of l'th block*
Γ_{kl} = *the interface between the kth and the lth blocks*
μ = *discrete mortar space test function*
μ_m = *viscosity of phase m, cp*
∂D = *Dirichlet boundary of subdomain*
Ω = *the reservoir domain*
Ω_k = *the kth block or subdomain*
$\partial\Omega$ = *boundary of* Ω

$d\sigma$ = *differential of surface area*

$[\cdot]_{kl}$ = *jump (difference in values on two sides of an interface between the kth block and the lth block)*

Subscripts

g = gas phase
G = gas component
h = approximated quantity
m = phase subscript (w, o, or g)
M = component subscript (W, O, or G)
o = oil phase
O = oil component
w = water phase
W = water component

Superscripts

B = properties of black-oil model
H = properties of oil-water model

Acknowledgments

We would like to acknowledge several contributors to the multi-block IPARS project, in particular Ivan Yotov, Manish Parashar and John Wheeler. DOE, NSF (KDI grand DMS 9873326), NGOTP, and the Industrial Affiliates of the Center of Subsurface Modeling supported this work.

References

1. Wheeler, M. F., Arbogast, T., Bryant, S., Eaton, J., Lu, Q., Peszynska, M., and Yotov, I.: "A Parallel Multiblock / Multidomain Approach for Reservoir Simulation," paper SPE 51884 presented at the 1999 SPE Symposium on Reservoir Simulation, Houston, Texas (Feb. 1999).

2. Glowinski, R. and Wheeler, M. F.: "Domain Decomposition and Mixed Finite Element Methods for Elliptic Problems", *Domain Decomposition Methods for Partial Differential Equations*, SIAM, Philadelphia (1988) 144.

3. Arbogast, T., Wheeler, M. F., and Yotov, I.: "Logically Rectangular Mixed Methods for Flow in Irregular, Heterogeneous Domains", *Computational Methods in Water Resources XI*, SIAM, A. A. Aldama and others (eds.), Computational Mech. Publ., Southampton (1996) 621.

4. Arbogast, T., Wheeler, M. F., and Yotov, I: "Mixed Finite Elements for Elliptic Problems with Tensor Coefficients as Cell-Centered Finite Differences," *SIAM J. Numer. Anal.* (1997) **34**, 828.

5. Yotov, I: "Mortar Mixed Finite Element Methods on Irregular Multiblock Domains," *Iterative Methods on Scientific Computation*, IMACS series Comp. Appl. Math., J. Wang et al (eds.), IMACS (1998) 239.

6. Yotov, I.: *Mixed Finite Element Methods for Flow in Porous Media*, PhD dissertation, Rice University, Houston, Texas (April 1996).

7. Wang, P., Yotov, I., Wheeler, M. F., Arbogast, T., Dawson, C., Parashar, M., and Sephernoor K.: "A New Generation EOS Compositional Reservoir Simulator: Part I -- Formulation and

Discretization," paper SPE 37979 presented at the 1997 SPE Reservoir Simulation Symposium, Houston, Texas (1997).

8. Peszynska, M., Lu, Q., and Wheeler, M. F.: "Coupling Different Numerical Algorithms for Two Phase Fluid Flow," Mathematics of Finite Elements and Applications, *MAFELAP 1999*, Elsevier, 2000, p.205-214.

9. Peszynska, M., Lu, Q., and Wheeler, M. F.: "Multiphysics Coupling of Codes," XIII International Conference on Computational Methods in Water Resources, Calgary, Alberta, Canada, Bentley, L.R. and Sykes, J.F. and Brebbia, C.A. and Gray, W.G. and Pinder, G.F., eds., p. 175-182, Balkema, 2000

10. Lu, Q.: *A Parallel Multi-Block / Multi-Physics Approach for Multi-Phase Flow in Porous Media*, PhD dissertation, University of Texas at Austin, Austin, Texas (May 2000).

11. Champman, R. E.: *Petroleum geology*, Elsevier Scientfic Publishing Company (1983).

12. Peaceman, D. W.: *Fundamentals of Numerical Reservoir Simulation*, first edition, Elsevier Scientfic Publishing Company, Amsterdam-Oxford-New York (1977).

13. Dake, L. P.: *Fundamentals of Reservoir Engineering*, Elsevier Scientific Publishing Company, Oxford-New York (1978).

14. Aziz K. and Settari, A.: *Petroleum reservoir simulation*, Applied Science Publishers, Ltd, London (1977).

15. Russell, T. F. and Wheeler, M. F.: "Finite Element and Finite Difference Methods for Continuous Flows in Porous Media," *The Mathematics of Reservoir Simulation*, R. E. Ewing (eds.), SIAM, Philadelphia (1983) 35.

16. Kelly, C. T.: *Iterative Methods for Linear and Nonlinear Equations*, SIAM, Philadelphia (1995).

17. Edwards, H. C.: "A Parallel Multilevel-Preconditioned GMRES Solver for Multiphase Flow Models in the Implicit Parallel Accurate Reservoir Simulator," *TICAM report*, Texas instit. for comp. and appl. math., Austin, Texas (Feb. 1998).

18. Honarpour H., Kedritz, L., and A. H. Harvey A. H.: *Relative Permeability of Petroleum Reservoir*, CRC Press, Inc., Boca Raton, Florida (1986).

19. Parashar, M., Wheeler, J. A, Browne, J. C., Pope, G. and Wang, K., and Wang, P.: "A New Generation EOS Compositional Reservoir Simulator: Part II -- Framework and Multiprocessing," paper SPE 37977 presented at the 1997 SPE Reservoir Simulation Symposium, Houston, Texas (1997).

20. Snir, M., Otto, S., Huss-Lederman, S., Walker, D., and Dongarra, J.: *MPI: the CompleteReference*, The MIT Press (1996).

21. *Tecplot, User's Manual*, version 7.5, Amtec Engineering, Inc., Bellevue, Washington (1998).

22. Lake, L. W., *Enhanced Oil Recovery*. Prentice Hall, Englewood Cliffs, New Jersey (1989)

23. Wheeler, M. F., Wheeler, J. A. and Peszynska, M.: *A Distributed Computing Portal for Coupling Multi-Physics and Multiple Domains in Porous Media*, XIII International Conference on Computational Methods in Water Resources, Calgary, Alberta, Canada, Bentley, L.R. and Sykes, J.F. and Brebbia, C.A. and Gray, W.G. and Pinder, G.F., eds., p. 167-174, Balkema, 2000