

**ADVANCED TECHNIQUES AND ALGORITHMS
FOR RESERVOIR SIMULATION, III:
MULTIPHYSICS COUPLING FOR TWO PHASE FLOW
IN DEGENERATE CONDITIONS**

MALGORZATA PESZYŃSKA*

Abstract. The multiphysics algorithm for coupling of two phase and single phase flow models is presented. The algorithm is built on a domain decomposition or multiblock formulation for multiphase flow and can be seen as a way to locally reduce the interface problem to a simpler one. Such an approach offers reduction in computational complexity. The key to implementation is flexibility in the choice of interface variables which may require application of a nonlinear boundary condition. Computational experiments demonstrate the strength of the method as well as its sensitivity to physical conditions.

Key words. multiphysics, domain decomposition, multiblock, two phase flow, primary variables, nonlinear boundary condition.

AMS(MOS) subject classifications. Primary 65M55; Secondary 35K65, 35K60, 76T99, 76S05.

1. Introduction. In this paper we discuss the multiphysics algorithm for coupling of multiphase flow models. We focus on two phase flow models in degenerate or *residual* conditions and discuss the *multiphysics* approach for coupling of two phase and single phase flow models. More details on the general approach will be presented elsewhere [9]. Discretization is discussed in [13, 16]. See [12] for examples of coupling of three phase, two phase and single phase models; see also [15] for an alternative multiblock multiphysics approach.

The multiblock with *multinumerics* and *multiphysics* concepts discussed here has proved to be a successful approach, and is now a part of IPARS (Integrated Parallel Accurate Reservoir Simulator) framework, see [13, 10, 14]. The implementation under IPARS allows currently for coupling of any set of the five models: *black oil* (implicit), *two phase oil-water* (implicit), *single phase* (implicit), *two phase oil-water* (explicit), *single phase* (explicit). One of the elements of the implementation is flexibility in the choice of interface variables. Below we illustrate the influence of this choice on the convergence of the multiblock algorithm. There is evidence that in some conditions, for two phase flow multiblock formulation, the use of pressures as primary variables with “natural” ordering of fluxes is favorable, where order of phases in pressures agrees with the order of fluxes [16]. However, in this paper we are interested in performance of the algorithm in conditions close to degenerate. Degenerate or residual conditions may

*TICAM, University of Texas at Austin, Austin, TX 78712. The work of the author was supported in part by NSF grant DMS 9873326, NPACI subcontract # 10152711, DOE grant DE-FG03-99ER25371, and by ATP/ARP grant 221.

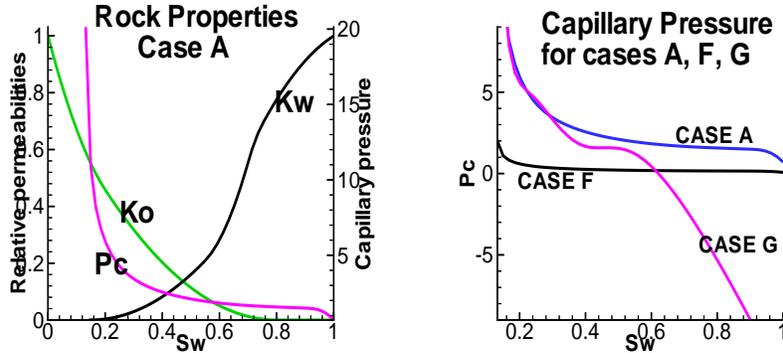


FIG. 1. Left: rock properties for case A. Right: Capillary pressure curves for cases A, F, G.

occur in some parts of the reservoir throughout the simulation, for example, in the aquifer part, or only after some time of production. It appears that a choice of one pressure and one concentration or saturation as primary variables may be preferable in such conditions. In fact, that choice leads naturally to the multiphysics approach which can be seen as a way to reduce the original two phase interface problem to a single phase problem, as the two phase formulation becomes degenerate. It turns out that the multiphysics approach not only gives accurate results (as compared to traditional approach), but that it also helps to reduce the computation time.

The plan of the paper is as follows. In Section 2 we review the two models: *two phase oil–water* and *single phase* and discuss conditions under which the two phase model becomes locally degenerate or in some sense “becomes” single phase. In Section 3 we recall the multiblock interface algorithm for the *two phase oil–water* model and discuss experiments and the behavior of the interface operator in different physical situations. In Section 4 we formulate the multiphysics interface algorithm and show results of the multiphysics coupling as well as discuss the efficiency of the multiphysics approach.

Throughout the paper we use the following notation. We use right subscripts α to denote the phase (component). The bar \bar{V} means that the variable V is defined on the interface (in the *mortar space*) and right superscript V^* means that it is used as a Dirichlet condition. Left superscripts and subscripts denote the domains in which a variable has a meaning. For example, ${}^2_{\mathcal{A}}P_w^*$ denotes boundary condition on water pressure in block \mathcal{A} where two phase flow model is used. Some of this notation is omitted when there is no danger of confusion.

TABLE 1
Definitions for two phase and single phase models.

model	two phase	single phase
definitions:		
pressure	${}^2P_w, {}^2P_o$	1P_w
compressibility	${}^2c_w, {}^2c_o$	1c_w
viscosity	${}^2\mu_w, {}^2\mu_o$	${}^1\mu_w$
density	${}^2\rho_w, {}^2\rho_o$	${}^1\rho_w$
potential	$\rho_\alpha = \rho_{\alpha,ref} \exp(c_\alpha P_\alpha)$ ${}^2\Psi_w, {}^2\Psi_o$	${}^1\Psi_w$
	$\nabla\Psi_\alpha = \nabla P_\alpha - \rho_\alpha G \nabla D$	
saturation	${}^2S_w, {}^2S_o$	(conversion factor) $({}^1S_w = 1 - S_{o,res})$
concentration	${}^2N_w, {}^2N_o$ ${}^2N_\alpha = {}^2S_\alpha^2 \rho_\alpha$	${}^1N_w = {}^1S_w^1 \rho_w$
injection/ production rate	${}^2q_w, {}^2q_o$	1q_w
mass fluxes	${}^2U_w, {}^2U_o$	1U_w

2. Models. Consider a reservoir Ω characterized by the values of porosity ϕ and permeability \mathbf{K} , both dependent on position $x \in \Omega$, and by the *rock properties* for the oil–water mixture expressed by relative permeabilities k_w, k_o and capillary pressure relationship P_c . The rock properties k_w, k_o, P_c are assumed to be known functions of the *wetting phase* (water) saturation S_w . In general, the simulator allows for different rock types or, in other words, for dependence of the rock properties on the position, but for simplicity this case is not discussed here. Also, let $D(x)$ denote the depth and G the gravity constant. The system is complemented by the boundary conditions: we use no–flow condition on the external boundary $\partial\Omega$ and we also use *well boundary conditions* for the source terms q_w, q_o implemented with the *Peaceman well model* [11]. Initial condition of the reservoir is one of hydrostatic equilibrium and it is made specific by using the value of water saturation $S_{w,init}$ and oil pressure $P_{o,init}$ at a prescribed depth. In a typical reservoir which is at equilibrium, with all cells of the same *rock type*, water as the heavier of the two fluid phases prevails at the bottom and oil prevails at the top.

The two models in the IPARS framework discussed here are the two phase oil–water model and the single phase flow model for water. The definitions and equations for both models are summarized for convenience in Table 1 and Table 2.

TABLE 2
Equations for two phase and single phase models.

model	two phase	single phase
conservation:		
momentum (Darcy's law)	${}^2U_w = {}^2\rho_w \mathbf{K} \frac{k_w}{\mu_w} \nabla({}^2\Psi_w)$ ${}^2U_o = {}^2\rho_o \mathbf{K} \frac{k_o}{\mu_o} \nabla({}^2\Psi_o)$	${}^1U_w = {}^1\rho_w \mathbf{K} \frac{1}{\mu_w} \nabla({}^1\Psi_w)$
mass	$\frac{\partial(\phi^2 N_w)}{\partial t} - \nabla \cdot {}^2U_w = {}^2q_w$ $\frac{\partial(\phi^1 N_o)}{\partial t} - \nabla \cdot {}^1U_o = {}^1q_o$	$\frac{\partial(\phi^1 \rho_w)}{\partial t} - \nabla \cdot {}^1U_w = {}^1q_w$
closure:		
volume balance	${}^2S_w + {}^2S_o = 1$	
capillary pressure	$P_c = P_o - P_w$	
primary variables	P_o, N_o	P_w

2.1. Degeneracy. Whether at initial conditions or after some time of production, some parts of the reservoir or some cells of the computational grid may contain very little of the oil phase, so that individual oil “globules” become disconnected and oil phase cannot flow. How much of the *residual* oil phase remains, depends on the rock properties. If rocks are *water-wet*, then a typical set of rock properties may look like those of Figure 1. In this case, for water saturations $S_w \geq 1 - S_{o,res} = 0.8$, the value of oil relative permeability k_o is close to zero, thereby making the oil phase immobile or $U_o = 0$. It is then natural to identify the cells described by the two phase flow equations with cells described by the single phase equations. More precisely, we can assume the same constitutive properties ${}^2c_w = {}^1c_w$, ${}^2\mu_w = {}^1\mu_w$, ${}^2\rho_{w,ref} = {}^1\rho_{w,ref}$ etc. From these follows the identification ${}^2P_w = {}^1P_w$. As concerns mass, we can assume that in each (two phase) cell with ${}^2S_w \geq 1 - S_{o,res}$, the value of 2S_w is automatically switched for the purposes of simulation to ${}^2S_w = 1$ and we write symbolically ${}^1S_w = 1$ (but note that saturation variable has no “real” meaning in single phase flow model). Alternatively, we can use conversion factor ${}^2S_w = {}^1S_w = 1 - S_{o,res}$. Either way, there is a natural identification ${}^1N_w = {}^1S_w^1\rho_w = {}^2N_w$. Both of these options assure the mass conservation. In the current implementation we use the former option.

The remaining issue is the identification of fluxes 2U_w and 1U_w crossing the interface that is crossing boundaries of cells adjacent to the interface. The flux 2U_w depends on the value of $k_w({}^2S_w)$. In the cell adjacent to interface and belonging to the two phase model block, the value of saturation 2S_w describing the residual conditions is determined as discussed

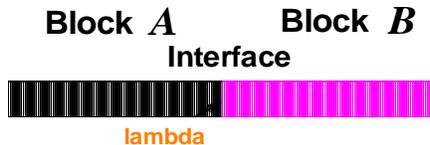


FIG. 2. Domain decomposition for 1D case.

above. Now, the value of $k_w(^2S_w)$ itself can be one (like on the sample $k_w(S_w)$ curve shown in Figure 1) or to some smaller number, depending on the available measurements and scaling, in particular on the wettability of the rock. If $k_w(^2S_w) = 1$, then in a natural way we identify ${}^2U_w = {}^1U_w$. In a more general case, we write ${}^2U_w = k_w(^1S_w){}^1U_w$, and a conversion factor is necessary.

We remark that the above reduction of the two phase system to the single phase is only appropriate for some rock types and physical conditions. For example, for *oil-wet* rocks, the oil phase stays in the pores as it “sticks” to their surfaces, the residual oil saturation is higher, oil phase is not disconnected but water phase may be, and S_w never reaches high values. In intermediate cases called *mixed-wet*, a variety of possibilities may occur depending on the history of the reservoir [4, 5]. Also, when *hysteresis* in capillary pressure is important, then the *imbibition* rather than *drainage* P_c curves should be used in waterflood simulations. An example of *mixed-wet* or *imbibition* P_c curve is given in Figure 1, case G. Note that most of the theoretical results available for well-posedness of the oil-water systems [1, 3] assume that the value of $k_o P'_c \rightarrow 0$ or that it is at least bounded. This assumption, while justifiable for *water-wet* cases, will not be valid for *oil-wet* cases or many mixed-wet cases.

3. The interface algorithm in degenerate conditions. In this section we recall a multiblock algorithm for a two phase model, and discuss its behavior in degenerate conditions. We first present results for the first time step for some typical *water-wet* cases, then discuss sensitivity of the algorithm to capillary pressure, and finally discuss its long-term behavior.

3.1. Multiblock algorithm in 1D. Consider the reservoir as in Figure 2 which is decomposed into two blocks \mathcal{A} , \mathcal{B} , or *subdomains*, see Figure 2, with interface Γ . The domain decomposition algorithm which is an extension of [6] is used to (weakly) impose conservation of mass and momentum across the interface using mortar spaces [13, 16, 2, 14]. For simplicity, assume a 1D two phase case where there is only one degree of freedom in the mortar space composed of piecewise constants only. In general, a given mortar grid has a set of degrees of freedom associated with each interface (primary) variable. In the 1D case the phase pressures $\mathbf{\Lambda}$ on the interface

(or other variables, see dicussion below) are constants and the mortar projections are identities, hence $\mathbf{\Lambda} = \mathbf{\Lambda}^*$. The value of $\mathbf{\Lambda}^*$ is transformed to deliver Dirichlet conditions imposed on the primary variables (P_o, N_o) of the two phase subdomain problems

$$(3.1) \quad {}^{\mathcal{A}}P_o|_{\Gamma} = P_o^*, \quad {}^{\mathcal{A}}N_o|_{\Gamma} = N_o^*$$

$$(3.2) \quad {}^{\mathcal{B}}P_o|_{\Gamma} = P_o^*, \quad {}^{\mathcal{B}}N_o|_{\Gamma} = N_o^*.$$

The interface algorithm uses an iteration to find the value of $\mathbf{\Lambda}$ such that the jump in the phase fluxes across the interface ${}^2\mathbf{B}(\mathbf{\Lambda}) = (B_o(\mathbf{\Lambda}), B_w(\mathbf{\Lambda})) = \mathbf{0}$. In a multidimensional case, the computation of $\mathbf{B}(\mathbf{\Lambda}) = {}^2\mathbf{B}(\mathbf{\Lambda})$ involves an integral over Γ of a product with test functions from an appropriate mortar space. In the 1D case discussed here, the matching of fluxes across Γ , with $\eta_{\mathcal{A}}, \eta_{\mathcal{B}}$ denoting the unit normal to Γ outward to blocks \mathcal{A}, \mathcal{B} , respectively, is expressed by

$$(3.3) \quad B_{\alpha}(\mathbf{\Lambda}) := {}_{\mathcal{A}}U_{\alpha} \cdot \eta_{\mathcal{A}} + {}_{\mathcal{B}}U_{\alpha} \cdot \eta_{\mathcal{B}} = 0, \quad \forall \alpha = o, w.$$

In general, $\mathbf{\Lambda}$ can represent a set of pressures, but it can also be any set of variables which uniquely determines the conservation of momentum as well as leads to a well-posed subdomain problem. Which variables are appropriate depends on the problem. For example, the choice of $\mathbf{\Lambda} = (\overline{P_o}, \overline{N_o})$ appears natural as it coincides with primary variables as in Table 2 and therefore does not require any extra map beside the mortar projections $\mathbf{\Lambda} \mapsto \mathbf{\Lambda}^*$ to deliver Dirichlet values P_o^*, N_o^* .

The schematic representation of the map $\mathbf{\Lambda} \mapsto B_{\alpha}(\mathbf{\Lambda})$ is

$$\mathbf{\Lambda} \xrightarrow{\text{proj.}} \begin{array}{l} {}_{\mathcal{A}}\mathbf{\Lambda}^* \\ {}_{\mathcal{B}}\mathbf{\Lambda}^* \end{array} \mapsto \begin{array}{l} ({}_{\mathcal{A}}P_o^*, {}_{\mathcal{A}}N_o^*) \\ ({}_{\mathcal{B}}P_o^*, {}_{\mathcal{B}}N_o^*) \end{array} \xrightarrow{\begin{array}{l} \text{solve}(\mathcal{A}) \\ \text{solve}(\mathcal{B}) \end{array}} \begin{array}{l} {}_{\mathcal{A}}U_{\alpha}|_{\Gamma} \cdot \eta_{\mathcal{A}} \\ {}_{\mathcal{B}}U_{\alpha}|_{\Gamma} \cdot \eta_{\mathcal{B}} \end{array} \mapsto B_{\alpha}(\mathbf{\Lambda}).$$

Note that the map $\mathbf{\Lambda} \mapsto \mathbf{B}(\mathbf{\Lambda})$ is a composition of computation (3.3) following the Dirichlet-to-Neumann map $({}_{\mathcal{A}}P_o^*, {}_{\mathcal{A}}N_o^*) \mapsto ({}_{\mathcal{A}}U_{\alpha}|_{\Gamma} \cdot \eta_{\mathcal{A}}, {}_{\mathcal{B}}U_{\alpha}|_{\Gamma} \cdot \eta_{\mathcal{B}})$, the transformation maps $\mathbf{\Lambda}^* \mapsto (P_o^*, N_o^*)$ for each block, and of mortar projections $\mathbf{\Lambda} \mapsto \mathbf{\Lambda}^*$.

The problem $\mathbf{B}(\mathbf{\Lambda}) = \mathbf{0}$ is solved by a version of the *inexact Newton-GMRES* algorithm [13, 16]. In fact, we use a finite difference approximation to the directional derivative

$$(3.4) \quad \mathbf{B}'(\mathbf{\Lambda}) \triangle \mathbf{S} \approx D_{\delta} \mathbf{B}(\mathbf{\Lambda}) \triangle \mathbf{S} = \frac{\mathbf{B}(\mathbf{\Lambda} + \delta \triangle \mathbf{S}) - \mathbf{B}(\mathbf{\Lambda})}{\delta},$$

or some other version according to [7, 14] with δ chosen appropriately relative to the subdomain solver tolerance ζ . In each iteration, a new guess for the values of interface primary variables $\mathbf{\Lambda}$ is computed in the Newton step $\mathbf{B}'(\mathbf{\Lambda}) \triangle \mathbf{S} = -\mathbf{B}(\mathbf{\Lambda})$. If the norm $b(\mathbf{\Lambda}) = \|\mathbf{B}(\mathbf{\Lambda})\|$ is small enough, then the iteration stops. Otherwise, a new guess is sought. The

Newton step is solved with GMRES and an efficient preconditioner for the GMRES solver is necessary for efficiency. However, in the 1D or small dimensional experiments cases discussed below, GMRES can be assumed to work almost exactly and preconditioning plays no role. This way we can associate any convergence results with only the nonlinear properties of the map $\Lambda \mapsto \mathbf{B}(\Lambda)$ and with the choice of primary variables.

TABLE 3
Choices of primary variables and of order of fluxes $\mathbf{B}^j(\Lambda_k)$.

	$\Lambda^0 = (\overline{P_o}, \overline{N_o})$	$\Lambda^1 = (\overline{P_o}, \overline{P_w})$	$\Lambda^2 = (\overline{P_w}, \overline{N_o})$
$\mathbf{B}^0 = (B_o, B_w)$	00	10	20
$\mathbf{B}^1 = (B_w, B_o)$	01	11	21

3.2. Choices of interface variables. We discuss three choices of interface primary variables and two different arrangements of the jump of the fluxes. We refer to these as $\Lambda^0 = (\overline{P_o}, \overline{N_o})$, $\Lambda^1 = (\overline{P_o}, \overline{P_w})$, $\Lambda^2 = (\overline{P_w}, \overline{N_o})$. The value of the jump in the fluxes is referred to as $\mathbf{B}^0 = (B_o, B_w)$ or $\mathbf{B}^1 = (B_w, B_o)$. The choices and the mnemonic symbols for their combinations are summarized in Table 3. Note that the order of fluxes may be very significant as it can change the character of Jacobian from being (positive) definite to being indefinite or saddle point system like.

For each Λ^k , $k = 0, 1, 2$, one needs to determine a map $\Lambda^* \mapsto (P_o^*, N_o^*)$ which is in general nonlinear and which follows from the set of conservation and constitutive equations in Table 2. In fact, the main value to be determined is one of S_w^* . All the other values follow, once S_w^* is known. For convenience, the maps are summarized below (asterisks are omitted).

$$(3.5) \quad S_w(\Lambda^0) = S_w(P_o, N_o) = 1 - \frac{N_o}{\rho_o(P_o)}$$

$$(3.6) \quad S_w(\Lambda^1) = S_w(P_o, P_w) = P_c^{-1}(P_o - P_w)$$

$$(3.7) \quad S_w(\Lambda^2) \stackrel{\text{implicit by}}{\vdots} S_w(P_w, N_o) = 1 - \frac{N_o}{\rho_o(P_w + P_c(S_w))}.$$

In the above equations, calculation (3.5) is straightforward, while equation (3.6) requires inverting of P_c which may present difficulties when P_c' is close to zero. On the other hand, inspection of the last equation (3.7) shows that, using simple fixed point iteration, the value of $S_w(\Lambda^2)$ can be found faster for smaller P_c' values and that the iteration may have difficulties where P_c' is large, as the derivative of the fixed point map $S_w = F(S_w)$ is $F'(S_w) = N_o c_o \frac{P_c'}{\rho_o}$. Also, for cases with small compressibility of oil c_o , the iteration may converge faster. In other words, the choice Λ^2 may be more

favorable than Λ^1 where P'_c is small, and vice-versa. In general, in some cases, depending on values of P_c and P'_c , the map $B'(\Lambda)$ may not be well conditioned, thereby making it more difficult to solve the Newton step. In addition, the finite difference approximation to the Jacobian is sensitive to singularities. See the results of computational experiments presented below.

In the remainder of this section we test the convergence of the Newton iteration on the interface in degenerate (residual oil) conditions. To put these results in perspective, we discuss the convergence over a large set of initial saturations when S_w varies from values around $S_w = 0.2$ (mainly oil with water at residual conditions) to around $S_w = 0.75$ (mainly water only with oil at residual conditions). In the latter case we show how the original two phase interface problem can be reduced to the single phase using the multiphysics approach.

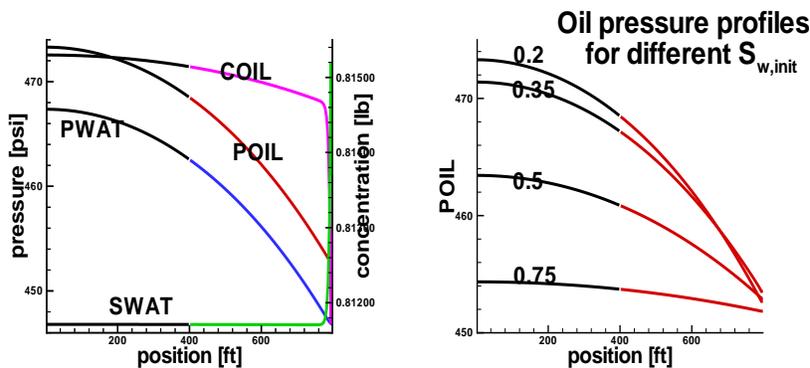


FIG. 3. Left: profiles of solution of case A for $S_{w,init} = 0.2$. Right: pressure solution for $S_{w,init} = 0.2, 0.35, 0.5, 0.75$. Curves in different blocks have different colors. Note the “gap” at 400’ which corresponds to the interface.

3.3. Computational experiments for 1D problem. Consider a 1D reservoir of length 800’ initially at equilibrium with $P_{o,init} = 500$ psi and a case dependent $S_{w,init}$. The reservoir is decomposed into two blocks or subdomains, each of 100 cells \times 4’. The constants in constitutive relationships are selected so that the relative properties of oil and water are conserved, yet the concentrations are very close in values to saturations. Specifically, $\rho_{o,ref} = 1.0$, $\rho_{w,ref} = 1.2[\frac{lb}{ft^3}]$. The compressibility constants are $c_o = .4 \times 10^{-4}$, $c_w = .33 \times 10^{-5}[/math> /psi]. Relative permeability and capillary pressure functions used in this case are those in Figure 1, case A. At initialization the reservoir is at equilibrium. There is a production well located at the end of one block where the pressure after the beginning of simulation is dropped to 450psi. In all cases discussed here the stopping criterium for the interface iteration was chosen to maintain mass balance$

up to 8 significant figures. All cases are run without preconditioner, as explained above.

Figure 3 shows the profiles of solution at 10 days which corresponds to the first time step. Such an unusually large time step is used intentionally, in order to amplify the difficulties. The solutions are plotted separately in the interior of each block and so there is a “gap” between the plots at 400’. However, note the apparent continuity of the solution across the interface. The value at the “gap” or at 400’ is the solution of the problem on the interface or the solution of $b(\mathbf{\Lambda}) = 0$ or $\mathbf{B}(\mathbf{\Lambda}) = \mathbf{0}$.

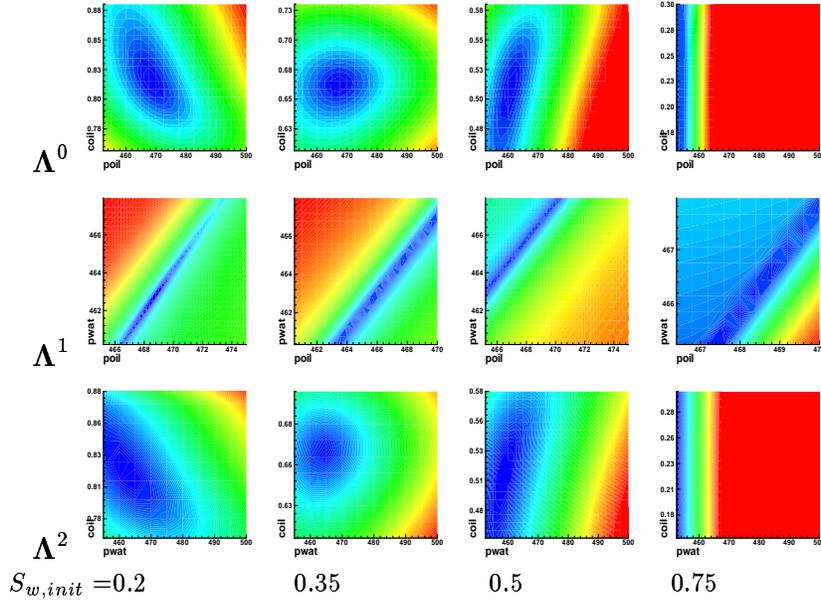


FIG. 4. Profiles of $b(\mathbf{\Lambda}^k)$ for $k = 0$ (top), $k = 1$ (middle), $k = 2$ (bottom) for $S_{w,init} = 0.2, 0.35, 0.5, 0.75$ (from left to right). Note that dark colors correspond to the solution of $\mathbf{B}(\mathbf{\Lambda}) = \mathbf{0}$.

The model case A is used in the following experiment. First, we examine the values of $b(\mathbf{\Lambda}) = \|\mathbf{B}(\mathbf{\Lambda})\|$ for $S_{w,init} = 0.2, 0.35, 0.5, 0.75$. To this aim, we set up the code to deliver the value of $b(\mathbf{\Lambda})$ for requested values of $\mathbf{\Lambda}$ at the first interface iteration, and we display these values as a contour plot in the plane spanned by interface primary variables; see Figure 4. The choice of range of the requested values of $\mathbf{\Lambda}$ is such that $b(\mathbf{\Lambda})$ is well defined. Also, the step between the adjacent values of primary variables does not take into account any special scaling : step between pressures is similar to step between concentrations which is how Newton/GMRES would operate anyway. Next, we investigate the convergence and sensitivity of the interface operation for the first time step.

TABLE 4

Function evaluations for cases A, B. “-” indicates no convergence, “*” indicates problem with the transformation map $\mathbf{\Lambda}^* \mapsto (P_o^*, N_o^*)$.

$S_{w,init}$	Case A (small δ)						Case B (larger δ)					
	00	01	10	11	20	21	00	01	10	11	20	21
.20	10	10	10	10	10	10	9	9	7	7	16	19
.35	7	7	7	7	7	7	7	10	7	7	7	10
.50	10	7	10	7	7	7	10	10	10	7	9	10
.60	7	10	10	7	10	10	10	10	13	10	10	10
.65	-	7	-	-	10	10	10	10	16	10	10	10
.70	-	7	10	7	7	12	7	7	19	10	7	7
.72	*	9	10	7	*	6	7	7	22	7	7	7
.75	-	-	-	-	-	-	7	12	19	7	7	12

The comparison of the profiles of $b(\mathbf{\Lambda})$ is very interesting. First observe that profiles for $\mathbf{\Lambda}^0$ and for $\mathbf{\Lambda}^2$ appear similar which may lead to similar behavior of the interface solver, a fact that is later confirmed by iteration counts. In particular, at $S_{w,init} = 0.75$ (last column), both choices of $\mathbf{\Lambda} = \mathbf{\Lambda}^0, \mathbf{\Lambda}^2$ show the degenerate behavior of the code because there appears to be little dependence of the plotted value $b(\mathbf{\Lambda})$ on the second variable \overline{N}_o . This shows that one can reduce the interface problem using the multiphysics approach to a single phase problem. Another observation is that in case of $\mathbf{\Lambda}^1$, the solution lies inside of a very thin ellipse. One of the axes of this ellipse is the line $\overline{P}_o - \overline{P}_w = P_c(S_w)$ where S_w corresponds to the solution of $\mathbf{B}(\mathbf{\Lambda}) = \mathbf{0}$ located in the center of the ellipse. The ratio of lengths of the two ellipses indicates difficulties in solving the problem $\mathbf{B}(\mathbf{\Lambda}) = \mathbf{0}$.

Now we discuss the number of interface iterations for the first time step with the stopping criterium as specified above. The interface iterations are also called function evaluations, as each iteration (Newton or GMRES) delivers a new value of $\mathbf{B}(\mathbf{\Lambda})$ for current guess $\mathbf{\Lambda}$, see Equation 3.4. The number of function evaluations are presented in Table 4 and they are directly related to the number of Newton iterations (not shown). For example, 7 evaluations correspond to 2 Newton steps since each Newton step requires two evaluations in GMRES, and one evaluation is required per step to check for convergence and one extra is used for postprocessing.

In general, it appears that the easiest to solve of all cases is the case of $S_{w,init} = 0.35$. This corresponds to the most regular of all plots for this case and is reflected by a lack of sensitivity to the parameter δ, ζ used in the difference approximation to Jacobian $\mathbf{B}'(\mathbf{\Lambda})$. In general, for cases which converged, the smaller value of δ corresponded to fewer iterations, as then the approximation to the Jacobian $\mathbf{B}'(\mathbf{\Lambda})$ is better. However, for some cases, for $S_w \geq 0.5$ there is significant sensitivity of the interface algorithm to the choice of parameters δ, ζ . It turns out that for $S_{w,init} = 0.75$ the

TABLE 5

Summary of cases for computational experiments. M denotes number of mortar degrees of freedom. Only horizontal permeability shown with ratio $\frac{h}{v} = 10$ assumed. Size and dimensions are shown for blocks \mathcal{A} and \mathcal{B} or only for block \mathcal{A} .

	dim	size(ft) (1 blk) #cells	ϕ	K (md)	time step (days)	ζ, δ	M
A	1D	400' 100	0.2	100	10	$10^{-8}, 10^{-10}$	1
B						$10^{-6}, 10^{-8}$	
C	3D	400'×400'×24' 8×8×2 (\mathcal{A}) 10×10×2 (\mathcal{B})	0.2	100	1	$10^{-8}, 10^{-10}$	4
D				100(\mathcal{A}) 500(\mathcal{B})	1 1		
E		dip of 11 deg		100			

iteration will break down for small δ, ζ (case A), but it will converge for somewhat larger values of these parameters (case B).

3.4. Experiments for 3D case. In order to understand better how the choice of $\mathbf{B}^j(\mathbf{A}_k)$ influences convergence on the interface, we set up a couple of 3D cases labelled C, D, E. All cases are summarized in Table 5. In these cases, $S_{w,init}$ is set at top of the reservoir and is used to compute saturations along the interface which are higher than $S_{w,init}$. This explains the shift in the results between cases A and C. Also, the runs with more mortar degrees of freedom, and/or with more wells and/or with higher heterogeneity (not shown here) show overall qualitative behavior similar to cases A and B while they differ in the number of evaluations.

In general, the runs show substantial influence of the convergence on the choice of $\mathbf{B}^j(\mathbf{A}^k)$; see Table 6. From the results presented it appears that the two most robust choices throughout the values $S_w = 0.2 \dots 0.75$ were those labelled as 01 and 21. Surprisingly, in cases C and D, for small saturation values $S_{w,init} < 0.5$, the “exchange” of the order of fluxes does not result in dramatic improvement or deterioration of convergence. On the other hand, the influence is dramatic for large saturation values as well as for case E in which flow is more gravity-influenced than in cases C, D. The (poor) performance of the interface iteration in the large saturation cases $0.5 \leq S_w \leq 0.75$ is apparently related to the fact that at these saturations, the oil flow is small or very small relative to water flow and so the order of fluxes in \mathbf{B}^0 (oil flux first) appears inappropriate.

In general, it appears natural, in the conditions close to degenerate, to drop the oil coordinate of $\mathbf{B}(\mathbf{A})$ and to drop one variable in \mathbf{A} and thereby to reduce the original system from 2×2 to 1×1 (per mortar variable). The

TABLE 6
Function evaluation count for cases C, D, E.

	Case C (3D, homog.)						Case D (3D, nonhom.)					
$S_{w,init}$	00	01	10	11	20	21	00	01	10	11	20	21
.20	15	16	13	16	15	15	16	17	15	26	25	24
.35	26	25	19	25	25	23	26	21	21	24	22	21
.50	-	50	34	33	-	52	-	44	38	48	-	45
.75	74	10	42	-	6	10	-	9	45	-	-	9

	Case E (3D, dip)					
$S_{w,init}$	00	01	10	11	20	21
.20	15	15	14	21	14	14
.35	163	64	40	119	-	95
.50	90	11	80	*	-	11
.75	203	11	81	*	*	11

TABLE 7
Function evaluation count for cases F, G, H.

	Case F						Case G					
$S_{w,init}$	00	01	10	11	20	21	00	01	10	11	20	21
.20	9	10	18	10	10	10	91	115	38	37	-	-
.35	7	10	9	11	7	10	-	9	67	*	-	9
.50	10	10	19	11	10	10	-	13	79	*	-	13
.75	7	11	-	16	7	11	-	9	74	*	-	9

	Case H					
$S_{w,init}$	00	01	10	11	20	21
.20	15	15	13	16	24	25
.30	17	15	*	*	17	15
.35	25	24	*	*	26	23
.50	25	18	23	25	25	19
.75	22	19	23	24	33	19

question is however, which primary variable to drop. In the above case, it appears natural to drop N_o in the choice of $\mathbf{\Lambda}^0$, $\mathbf{\Lambda}^2$ and to drop B_o in \mathbf{B}^1 . This reduction leads to the multiphysics setting discussed in Section 4.

Of the two choices, $\mathbf{\Lambda}^2$ is easier to implement, since the variable P_w has a natural counterpart in the single phase flow part while P_o does not. On the other hand, the choice $\mathbf{\Lambda}^1$ does not offer that “natural transition”. In fact, as the evolution of profiles of $B(\mathbf{\Lambda})$ over $S_w \in (0.2, 0.75)$ suggests, the operator $\mathbf{B}(\mathbf{\Lambda}^1)$ becomes more singular as S_w increases.

3.5. Sensitivity to capillary pressure. With the evidence above, it is still necessary to test different physical conditions. While comprehensive study is beyond the scope of this paper, it appears that the behavior of capillary pressure curves is the main factor determining the convergence. In this direction, we study model cases which use a different capillary pressure relationship than case A.

Consider the capillary pressure relationships shown on Figure 1B, with curve F in which capillary pressure effects are very small and curve G in which the inflection point of the P_c curve is moved towards smaller saturations and where a negative capillary pressure values occur for larger saturations. Curve G can be considered as an example of a *mixed-wet* system [8, 4] or of an imbibition P_c curve. Simulation cases F and G use the grid from case A and capillary curves F, G from Figure 1B, respectively. Additionally, case H uses grid from case C and capillary pressure curve F.

The function evaluation count for cases F, G, H is shown in Table 7. It is not surprising to see that the choice of $\mathbf{\Lambda}^1$ appears to be the least stable in case F. This is because the values of P_c as well as of P'_c are small, therefore making it harder to get S_w from equation (3.6). The same effect is amplified in case H. Another observation is that the use of “natural” ordering of fluxes seems inappropriate for $\mathbf{\Lambda}^0, \mathbf{\Lambda}^2$. For case G, the results shown are as anticipated: the largest number of iterations occurs near singularity of P'_c . This means that reduction to multiphysics cases is not appropriate for case G.

3.6. Long time behavior of the interface algorithm. It is important to discuss the same issues as above over production lifetime of a reservoir (up to the time well after breakthrough). For these purposes, we consider case \mathcal{C} geometrically congruent to C and scaled down by a factor of $10 \times 10 \times 10$ in all directions. The purpose of the scaling is to make the reservoir so small that the breakthrough occurs after merely a couple of months and the computations complete faster. Still, the case \mathcal{C} is very similar to C in that the iteration counts for first time step (not shown) for case \mathcal{C} are similar to those for case C.

We run several simulation with case \mathcal{C} using different combinations of primary variables and ordering of fluxes. The case(s) are run over sufficiently long time so that the waterflood front actually moves through the whole reservoir and the water saturation close to the interface Γ ranges from .2 to .7 during simulation. For example, for the top case in Table 8, the saturation at 500 days (197 time steps) near the block interface is close to .7. We report that qualitative results of simulations (not shown here) for different choices $\mathbf{B}^j(\mathbf{\Lambda}^k)$ exhibit only minor difference up to 0.3% in the well rates.

Table 8 contains the average number of evaluations and timings for case \mathcal{C} . These demonstrate the longterm behavior of the multiblock algorithm which essentially follows the trends observed for the first time step only.

TABLE 8

Long term effects of use of different primary variables for case C scaled by $10 \times 10 \times 10$. FEV=average number of evaluations per time step. Time is computation time.

$S_{w,init}$	00	01	10	11	20	21
	case with $S_{w,init} = .35$, 500 days (197 tsteps), coarse mortars					
FEV	11.96	10.28	9.75	10.32	11.68	10.21
time	307.12	276.64	299.35	305.60	302.21	271.94
	case with $S_{w,init} = .2$, 1000 days (367 tsteps), coarse mortars					
FEV	10.68	9.05	9.82	9.87	10.38	8.95
time	552.51	493.16	599.58	596.34	553.20	481.11

In summary, choice 21 appears as the most efficient for all runs over the production lifetime of the reservoir.

4. Multiphysics solution of degenerate cases. In the *water-wet* conditions it is natural to consider the oil-phase as immobile and practically absent from the problem in the residual conditions. The appropriate algebraic reduction is realized by the multiphysics algorithm.

In the multiphysics couplings, we replace one of the subdomain solvers in the original multiblock two phase oil-water subdomain solver by the single phase solver. This requires that the boundary conditions that we impose on the subdomain problem are now modified. In particular, the single phase flow model needs a boundary condition imposed on only one variable, specifically, on its subdomain solver primary variable 1P_w . Also, the single phase subdomain solver delivers values of flux for one phase only. The studies presented in the previous section show that the natural and efficient choice of the (mortar) interface variable for which we derive the Dirichlet boundary condition values P_w^* is water pressure $\overline{P_w}$.

Schematically, the algorithm is as follows. In every iteration, the interface algorithm comes up with values of interface degrees of freedom corresponding to one variable $\Lambda = (\overline{P_w})$ only. This value is then projected to deliver values of pressure ${}^1_{\mathcal{A}}P_w^*$ in the single phase model in block \mathcal{A} and values of water pressure ${}^2_{\mathcal{B}}P_w^*$ in block \mathcal{B} . In the two phase flow model assigned to block \mathcal{B} , we additionally need the values of oil concentration ${}^2_{\mathcal{B}}N_o^*$. These must correspond to the residual conditions and are taken from initialization step at the beginning of simulation. Both values, (${}^2_{\mathcal{B}}P_w^*$ and ${}^2_{\mathcal{B}}N_o^*$), are then transformed to deliver values of ${}^2_{\mathcal{B}}S_w^*$ and in what follows values of ${}^2_{\mathcal{B}}P_o^*$. Finally, the set (${}^2_{\mathcal{B}}P_o^*$, ${}^2_{\mathcal{B}}N_o^*$) is used by the subdomain (two phase) solver. The algorithm can be represented schematically as

$$\Lambda \xrightarrow{\text{proj.}} \begin{array}{l} \mathcal{A}\Lambda^* \\ \mathcal{B}\Lambda^* \end{array} \mapsto \begin{array}{l} ({}^1_{\mathcal{A}}P_w^*) \\ ({}^2_{\mathcal{B}}P_w^*) \end{array} \xrightarrow{\begin{array}{l} \text{solve}(\mathbf{1}, \mathcal{A}) \\ \text{solve}(\mathbf{2}, \mathcal{B}) \end{array}} \begin{array}{l} {}^1_{\mathcal{A}}U_w|_{\Gamma} \cdot \eta_{\mathcal{A}} \\ {}^2_{\mathcal{B}}U_w|_{\Gamma} \cdot \eta_{\mathcal{B}} \end{array} \mapsto B_w(\Lambda).$$

Note that algebraically, this algorithm describes the reduction of the original two phase interface problem to a single phase problem which in the Jacobian / residual formulation can be seen as

$$\begin{aligned} \begin{bmatrix} \frac{\partial B_w}{\partial P_w} & \frac{\partial B_w}{\partial N_o} \\ \frac{\partial B_o}{\partial P_w} & \frac{\partial B_o}{\partial N_o} \end{bmatrix} \begin{bmatrix} \Delta P_w \\ \Delta N_o \end{bmatrix} \\ = - \begin{bmatrix} B_w \\ B_o \end{bmatrix} \xrightarrow{\text{reduce}} \left[\frac{\partial B_w}{\partial P_w} \right] [\Delta P_w] = - [B_w]. \end{aligned}$$

In the implementation, this reduction can be done globally for all interfaces, or locally only for some selected interfaces. The choice between these options depends on the physical conditions which exist near a particular interface: whether they are of *residual* type or not. Furthermore, this reduction or “switch” to multiphysics can be done adaptively in time. For example, in case \mathcal{C} discussed in section 3.6, the results show that after 500 days of simulation the oil phase near block interface is practically immobile. From that time on, one can continue the original two phase multiblock simulation in the multiphysics mode which could dramatically increase the efficiency of computation.

To demonstrate the effectiveness of this approach consider a 2D dipping reservoir example $800' \times 20'$ (200×10 grid) for which, at equilibrium, the water–oil contact (WOC) is located as shown in Figure 5. There is an injection well alongside the bottom side of the domain as well as a production well alongside the top side. The reservoir is decomposed into two domains, with the interface Γ lying below WOC. Specifically, the interface is at a distance of $200'$ or 50 cells down from the upper corner. This decomposition ensures that single phase or two phase degenerate conditions exist in a neighborhood of the interface on the two phase (top) side. The top block is assigned the two phase model, and the bottom block is assigned the single phase model. The mortar grid is 5×1 (12 degrees of freedom) and convergence tolerance is set to reduce the initial interface residual by about a factor of 10^{-3} .

The results of simulation (pressure and saturation profiles) after 40 days are displayed in Figure 5. The first question concerning these results is how they are related to the solution of the problem posed on the whole domain (without domain decomposition) or, in other words, whether the interface iteration converged to a physical solution. To answer this question, we set up two simulations. The first one, (a), does not use domain decomposition and the solution is obtained with only the two phase flow code. The second, (b), uses multiblock multiphysics approach described above. The comparison of the two is discussed below.

One of the ways to compare these results is to look at the pointwise values of pressures and concentrations. These are not identical because the continuity across interface is only imposed in the weak sense, recall

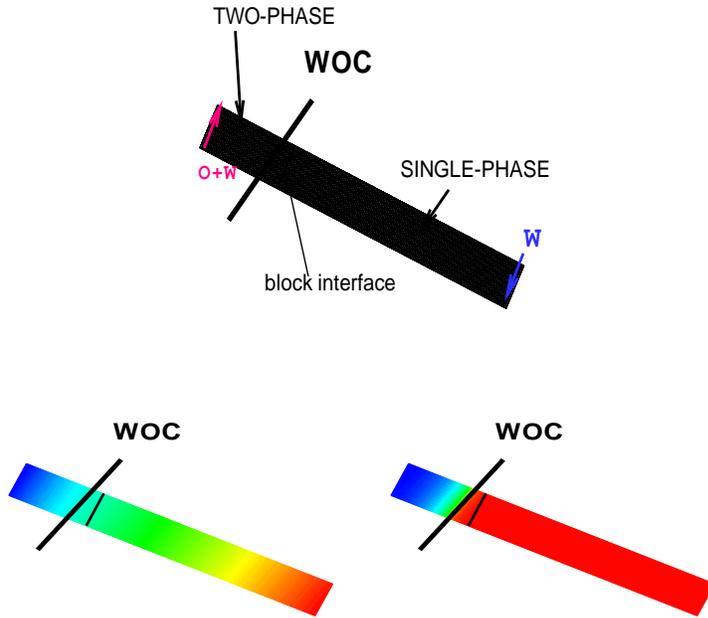


FIG. 5. Grid and decomposition of multiphysics case (top). Profiles of pressure (left bottom) and saturation (right bottom) after 40 days of simulation.

formulation in [13, 16]. However, the well rates (injection and production) obtained for runs (a) and (b) as in Figure 6 show these well rates to be very close. A closer quantitative inspection of differences between them shows differences less than .1% for oil production and less than 1% for water injection and production. Given the uncertainty of reservoir description, these differences are negligible in comparison to the ones caused by the use of different geostatistical permeability models or by different discretizations [15]. In conclusion, the multiphysics approach gives satisfactory results.

The second question to ask is how efficient is the use of the multiphysics approach. In other words, are the runs using multiphysics more cost-effective than those with a traditional (single block) approach using one two phase flow code over the whole domain. It is obvious that the most significant factor in the comparison and in the efficiency of multiphysics is the relative cost of the single phase and two phase models for the given problem. These depend on the cost factors for the models which usually scale like CN^β where C is a model dependent constant, N denotes the number of grid cells associated to a model subdomain, and β is a solver and model dependent exponent typically satisfying $1 \leq \beta \leq 3$. Note that $\beta = 1$ may correspond to a single phase flow model with a multigrid solver and

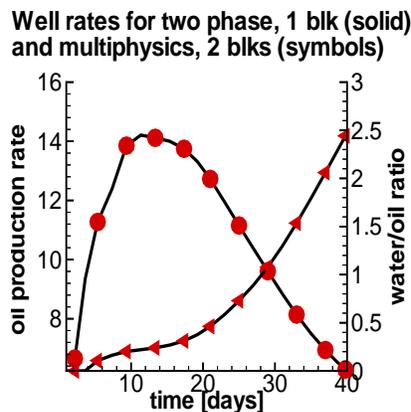


FIG. 6. Comparison of well rates for two phase single block and multiphysics runs.

$\beta = 3$ to a multiphase flow model with a direct solver. Other factors which influence the cost of the multiphysics approach include the overhead of the domain decomposition algorithm involving the outer iteration, projections, etc. In addition, in runs on a parallel machine, one needs to use proper load balancing between blocks and models.

To illustrate some of these issues, we present timings of some runs on a single processor machine in Table 9. The last column of Table 9 contains results for cases (a), (b) discussed above. It shows that for the problem with fine grid and with single phase region three times as large as the two phase region, the multiphysics simulation ran faster by more than a factor of 2 than the single block code. For comparison, data in first and second columns show timings for problems of different sizes and discretizations. For example, the first two rows of the first column correspond to the case where single phase domain was as large as the two phase domain, in which case multiphysics was slower than single block approach. Also, the last two rows of the first column correspond to the original case of size 800' with coarse discretization. These results prove that the number of cells assigned to each model is an essential factor in the cost-effectiveness of multiphysics.

5. Conclusions and acknowledgments. In this paper we discussed the multiphysics approach viewed as an efficient domain decomposition technique to solve two phase problems with residual conditions present locally. The focus was on *water-wet* reservoirs and we indicated sensitivity of the numerical algorithm to the physical conditions like wettability and capillary pressure. Our studies are intended as a first step to adaptivity of the multiphysics approach. However, more in-depth studies are necessary in order to determine the best choice of primary variables and order of fluxes for general multiphase multicomponent flow and transport problems.

TABLE 9
Comparison of timings for single block and multiphysics runs.

	Discretization in lateral direction: (# of cells) \times (cell dimensions) Vertical discretization is $10 \times 2'$.		
Influence of size	$(50+50) \times 4'$	$(50+100) \times 4'$	$(50+150) \times 4'$
a) single block (two ph.)	66.58	257.41	533.10
b) multiphysics	267.20	250.45	245.67
Influence of discretization	$(10+30) \times 20'$	$(20+60) \times 10'$	$(50+150) \times 4'$
a) single block (two ph.)	7.12	33.79	533.10
b) multiphysics	9.59	32.65	245.67

In particular, in the case of the black oil model [13, 12, 15], the choice of only pressures as boundary conditions does not always lead to a well-posed problem, as the amount of gas component in (undersaturated) oil phase may be completely arbitrary at residual gas phase saturation.

In the end, we would like to thank our colleagues: Qin Lu for his work on implementation of multiphysics, Manish Parashar for development of multiblock libraries and Bahareh Momken for her contribution to the single phase code. Discussions with John Wheeler, Yuri Vassilevski, and collaboration with coauthors of [13, 16] Mary Wheeler and Ivan Yotov are gratefully acknowledged.

REFERENCES

- [1] H.W. ALT AND E. DI BENEDETTO. Nonsteady flow of water and oil through inhomogeneous porous media. *Ann. Scuola Norm. Sup. Pisa Cl. Sci. (4)*, **12**(3):335–392, 1985.
- [2] T. ARBOGAST, L.C. COWSAR, M.F. WHEELER, AND I. YOTOV. Mixed finite element methods on non-matching multiblock grids. *SIAM J. Numer. Anal.*, **37**(4):1295–1315, 2000.
- [3] T. ARBOGAST, M.F. WHEELER, AND NAI-YING ZHANG. A nonlinear mixed finite element method for a degenerate parabolic equation arising in flow in porous media. *SIAM J. Numer. Anal.*, **33**:1669–1687, 1996.
- [4] FORREST F. CRAIG. *The reservoir engineering aspects of waterflooding*. Monograph (Society of Petroleum Engineers of AIME) Vol. **3**, 1971.
- [5] M. DELSHAD, R.J. LENHARD, MART OOSTROM, G.A. POPE, AND S. YANG. A mixed-wet hysteretic relative permeability and capillary pressure model in a chemical compositional reservoir simulator. In *1999 SPE Symposium on Reservoir Simulation*, pages 135–146, Houston, Texas, 1999. SPE 51884.
- [6] R. GLOWINSKI AND M.F. WHEELER. Domain decomposition and mixed finite element methods for elliptic problems. In *Domain Decomposition Methods for Partial Differential Equations*, pages 144–172, Philadelphia, 1988. SIAM.
- [7] C.T. KELLY. *Iterative methods for linear and nonlinear equations*. SIAM, Philadelphia, 1995.
- [8] L.W. LAKE. *Enhanced oil recovery*. Prentice Hall, Englewood Cliffs, New Jersey, 1989.
- [9] Q. LU, M. PESZYŃSKA, M.F. WHEELER, AND I. YOTOV. Multiphysics and multi-numeric couplings for multiphase flow in porous media. *in preparation*.

- [10] M. PARASHAR, J.A. WHEELER, J.C. BROWNE, G. POPE, K. WANG, AND P. WANG. A new generation eos compositional reservoir simulator: Part II – framework and multiprocessing. In *1997 SPE Reservoir Simulation Symposium*, Houston, Texas, 1997. SPE 37977.
- [11] D.W. PEACEMAN. Interpretation of well-block pressures in numerical reservoir simulation with non-square grid blocks and anisotropic permeability. *Trans. AIME*, **275**:10–22, 1983.
- [12] M. PESZYNSKA, Q. LU, AND M.F. WHEELER. Multiphysics coupling of codes. In L.R. Bentley, J.F. Sykes, C.A. Brebbia, W.G. Gray, and G.F. Pinder, editors, *XIII International Conference on Computational Methods in Water Resources, Calgary, Alberta, Canada*, pages 175–182. Balkema, 2000.
- [13] M.F. WHEELER. Advanced techniques and algorithms for reservoir simulation, ii: The multiblock approach in the integrated parallel accurate reservoir simulator (ipars). In *same*.
- [14] M.F. WHEELER, T. ARBOGAST, S. BRYANT, J. EATON, Q. LU, M. PESZYNSKA, AND I. YOTOV. A parallel multiblock/multidomain approach for reservoir simulation. In *1999 SPE Symposium on Reservoir Simulation*, pages 51–62, Houston, Texas, 1999. SPE 51884.
- [15] M.F. WHEELER, J.A. WHEELER, AND M. PESZYNSKA. A distributed computing portal for coupling multi-physics and multiple domains in porous media. In L.R. Bentley, J.F. Sykes, C.A. Brebbia, W.G. Gray, and G.F. Pinder, editors, *XIII International Conference on Computational Methods in Water Resources, Calgary, Alberta, Canada*, pages 167–174. Balkema, 2000.
- [16] I. YOTOV. Advanced techniques and algorithms for reservoir simulation, iv: Multiblock solvers and preconditioners. In *same*.