Nonlinear Domain Decomposition Scheme for Sequential Fully Implicit Formulation of Compositional Multiphase Flow

O. Møyner (NTNU, SINTEF) and A. Moncorgé (TOTAL E&P UK)

March 4, 2020

Abstract

New Sequential Fully Implicit (SFI) methods for compositional flow simulation have been recently investigated. These SFI schemes decouple the fully coupled problem into separate pressure and transport problems and have convergence properties comparable to the Fully Implicit (FI) method. The pressure system is a parabolic problem with fixed overall-compositions and the transport system is a hyperbolic problem with fixed pressure and total-velocity. We discuss some aspects of how to design optimal SFI schemes for compositional flow with general equation-of-states by localizing the computations. The different systems are solved sequentially and the Fully Implicit solution is recovered by controlling the a-posteriori splitting errors due to the choice of decoupling. When the parabolic and the hyperbolic operators are separated, it is possible to design nonlinear domain decomposition schemes taking the advantage of the specific properties of each operator. Usually, for reservoir simulation models, most of the reservoir is converged with SFI methods in one outer-iteration. However, in some localized regions with strong coupling between the pressure and the compositions, the SFI algorithms may need several outer-iterations. Here we propose a domain decomposition method based on a predictor-corrector strategy. As a first step, the nonlinear parabolic pressure equation is solved on the whole domain with the Multiscale Restriction-Smooth Basis (MsRSB) method used as a linear domain decomposition solver. In a second step the compositions system is solved. At the end of this first outer-iteration, most of the reservoir is converged. Based on a-posteriori splitting-errors of the SFI scheme in volume and velocity, we define local regions where additional global outer-iterations would be required in the conventional SFI scheme. We then fix Dirichlet boundary conditions for the pressure and the compositions and solve local problems in these non-converged regions. After convergence of these smaller nonlinear problems, if the boundary conditions are changed by the updated regions, the global pressure problem is revisited. An additional post-processing of local transport iterations makes sure mass is conserved everywhere. The resulting algorithm converges to the same solution as the FI solver, with all simultaneous updates to composition and pressure in localized regions. We demonstrate the robustness of this nonlinear domain decomposition algorithm across a wide parameter range. Realistic compositional models with gas and water injection are presented and discussed.

1 Introduction

Sequential implicit (SI) methods for compositional flow have initially being developed by [1, 32, 27]. These methods are a two-step process: First, the pressure solution is found by solving a pressure equation, second, a transport step updates the compositions. In general, this splitting relies on fixing quantities and generally local saturation errors will be present. One approach is to keep these errors bounded in time using relaxation terms [1, 27]. Sequential methods for compositional flow have recently received renewed interest in the context of multiscale reservoir simulation [9, 10, 21] by the works of [8, 18, 22, 19]. [8] considered a Sequential Fully Implicit (SFI) method in order to converge to the Fully Implicit (FI) solution. In their work, the pressure is found by solving a total-mass conservation equation, which can result in many outer-iterations between the pressure and the composition systems before convergence to an acceptable tolerance. In [18], the same total-mass pressure equation is used to converge to any given accuracy by selecting areas of the domain as FI and other areas as SFI. This method converges with similar properties as the FI method applied to the entire domain but may be too expensive when the coupling regions become too large. [22] presents a SI method conserving mass after each outer-iteration. The pressure is computed by an IMPES-type algebraic reduction similar to [3]. In most cases, the SI solution is qualitatively very similar to the FI solution in terms of reservoir and well responses. [19] introduces the nonlinear volume-balance equation that is the generalization of the previous volume balance equation derived for immiscible [11] and black-oil [14, 20] fluid. The pressure equations from [22] and [19] converge to the same solution. In addition to the saturation splitting error, [19] describe the total-velocity splitting error resulting from the sequential treatment and show the importance of controlling these error terms in order to converge to the FI solution.

In this work, we are combining ideas from these previous works. We are aiming at having a scheme reproducing the FI solution at a cost as close as possible to the cost of a SI method (or a SFI method with a single outer-iteration). We propose a domain-decomposition strategy based on a predictor-corrector pair of models. For the predictor-model, we consider the SI method which is known to efficiently converge by fixing pressure and total velocity, but may in the process neglect important physical couplings between the flow and the transport. To correct this initial guess, we consider the FI method, but we could equally well have used a local SFI solver. The key idea is the FI solver is only applied in areas where significant large splitting errors are detected after a solve of the predictor-model. This approach is reminiscent of adaptive-implicit methods [7, 6, 33, 26], where cells are taken to be either implicit or explicit based on local CFL estimates [24] as a middle-ground between explicit and fully-implicit schemes. In this work, we do not need to estimate CFL as we couple different implicit solvers. As we are concerned with controlling splitting errors rather than stability, this can be considered a complimentary approach which is the middle-ground between a sequential-implicit and a fully-implicit scheme.

We are using the implementation and formulation from [22]. We have not made a comparison between the natural-variable formulations from [22] and [19] but the structures of the splitting errors are similar between the two approaches and the conclusions of this study are holding for the two formulations. As a first step, the pres-

sure equation is solved on the whole domain with the Multiscale Restriction-Smoothed Basis (MsRSB) method used as a preconditioner for a linear solver. In a second step, the transport system is solved to update the compositions. At the end of this first outer-iteration, most regions of the reservoir model are converged. Based on the splitting-errors of the SFI scheme, we define local regions where additional global outer-iterations would be required in the conventional SFI scheme. We then fix Dirichlet boundary conditions for the pressure and the compositions at the boundary of these local regions and solve the local problems with the FI method. After convergence of these smaller nonlinear problems, if the boundary conditions are impacted by the updated solution inside the regions, the global pressure and transport problems are revisited. In case where the boundary conditions are not changed by the interior updates, an additional post-processing of local transport is applied to make sure mass is conserved everywhere. The resulting algorithm converges to the same solution as the FI solver within prescribed tolerances for velocity and saturation errors.

2 Governing equations

2.1 Compositional flow

In this work, we consider the coupled flow-transport problem for m hydrocarbon components and a water component $m+1=n$. We assume up to three phases present: a liquid, a vapor and an aqueous phase. In terms of phase composition, we assume that the hydrocarbon component i is only present in the liquid and vapor phases,

$$R_i = \frac{\partial}{\partial t}(\phi\left[\rho_l S_l X_i + \rho_v S_v Y_i\right]) + \nabla\cdot(\rho_l X_i \vec{v}_l + \rho_v Y_i \vec{v}_v) - q_i \tag{1}$$

where ϕ is the porosity, ρ_α the phase mass-density, S_α the phase saturation, $\vec{v}_\alpha$ the phase velocity for phase α and q_i some component source term. We use X_i and Y_i to refer to the mass-fraction of component i in the liquid and vapor phases, respectively.

The mass conservation equation for the water component is written in residual form as

$$R_w = \frac{\partial}{\partial t}(\phi\rho_w S_w) + \nabla\cdot(\rho_w\vec{v}_w) - q_w. \tag{2}$$

The composition in the liquid and vapor phases are given by the isofugacity constraint when both phases are determined to be present from a stability test [17],

$$R_f = f_i^l - f_i^v \tag{3}$$

where f_i^α depends on the phase compositions and pressure. We require that the mass-fractions of each phase sum-up to unity

$$\begin{aligned} R_X =& 1 - \sum_i^m X_i, \quad X_i \in [0,1]\,\forall\, i \in \{1,...,m\} \\ R_Y =& 1 - \sum_i^m Y_i, \quad Y_i \in [0,1]\,\forall\, i \in \{1,...,m\}. \end{aligned} \tag{4}$$

In addition, the phase saturations sum-up to unity, meaning that the fluids fill the entire pore-volume

$$R_s = S_w + S_v + S_l - 1,\ S_\alpha \in [0,1]\,\forall\,\alpha. \tag{5}$$

To relate the phase pressures to each other, we let p_c^α denote the pressure difference between phase α and the liquid hydrocarbon phase,

$$p_\alpha = p_l - p_c^\alpha. \tag{6}$$

As we define the phase pressures relative to the liquid pressure, we will drop the subscript and let p refer to the liquid pressure in the following.

We find the phase-velocities using the multiphase extension of Darcy's law with gravity for some absolute permeability $\mathbf{K}$ and mobility λ_α defined as the ratio between the phase relative permeability $k_{r\alpha}$ and viscosity μ_α:

$$\vec{v}_\alpha = -\mathbf{K}\lambda_\alpha(\nabla p_\alpha - \rho_\alpha g \nabla z), \quad \lambda_\alpha = \frac{k_{r\alpha}}{\mu_\alpha} \tag{7}$$

with g the gravitational acceleration and z the depth.

Herein, we use a natural-variables [5] type formulation where phase equilibrium is solved simultaneously with the flow equations. This specific choice is not essential to the proposed domain-decomposition scheme. We can define $\mathbf{R}_{FI}$ as the fully-implicit residual made up of the conservation laws for each component, the isofugacity constraint and the saturation closure,

$$\mathbf{R}_{FI} = \left[R_w, R_1, \ldots, R_m, F_1, \ldots, F_m, R_s\right]^\intercal \tag{8}$$

where the primary variables are the pressure p, phase saturations S_l, S_v, S_w, liquid mole fractions $x_1, \ldots, x_{m-1}$ and vapor mole fractions $y_1, \ldots, y_{m-1}$ after eliminating the last mole fractions by setting $R_X = 0$ and $R_Y = 0$ and using (6) to parametrize all pressures relative to the liquid pressure.

2.2 Pressure-transport system

We reformulate the system of conservation equations in (1) and (2) in an equivalent pressure-transport system. The pressure equation is formed by a weighted sum of the linearized mass conservation equations. It is equivalent to solving for the zero of the pressure residual,

$$\mathbf{R}_p = \beta_w R_w + \sum_{i=1}^{m} \beta_i R_i. \tag{9}$$

We solve $\mathbf{R}_p = 0$ keeping the overall composition of each component z_i and z_w fixed, with saturations and phase mole fractions changing based on the updated pressure according to (3) as predicted by the equation-of-state.

The transport system can be defined from the fully-implicit system in two steps. We first replace the expression for the phase flux $\vec{v}_\alpha$ (7) with the fractional-flow form

of the phase velocity $\vec{v}_\alpha$ where we assume a fixed total-velocity,

$$\vec{v}_\alpha = \frac{\lambda_\alpha}{\sum\limits_{\beta \in l,v,w} \lambda_\beta} \left(\vec{v}_t + \mathbf{K} \sum_{\beta \neq \alpha} \lambda_\beta (G_\beta - G_\alpha) \right), \tag{10}$$

$$G_\alpha = \nabla p_c^\alpha - \rho_\alpha g \nabla z, \tag{11}$$

where the fixed total-velocity $\vec{v}_t$ is defined from $\vec{v}_\alpha$ from the pressure system as $\vec{v}_t = \sum_\beta \vec{v}_\alpha$. We can now form the transport residual R_T analogously to R_{FI},

$$\mathbf{R}_T = \left[R_w, R_1, \ldots, R_m, F_1, \ldots, F_m \right]^\intercal. \tag{12}$$

The solution variables in this case are phase saturations S_l, S_v, S_w and mole fractions $x_1, \ldots, x_{m-1}, y_1, \ldots, y_{m-1}$. Comparing to the fully-implicit system, we still have exact mass-conservation as all mass-balance equations are present, but we have omitted the saturation closure $R_s = 0$. This means that we only guarantee that $S_\alpha \geq 0$ and the total-volume is relaxed. The reader is referred to [22] and [19] for the finer details of the pressure-transport strategy for compositional flow. In particular, for [22] for the computation of the weights β_w and β_i used in this work. In [19], the β_w and β_i are taken as the partial molar volume (or partial mass volume) of each component. In addition, the MRST implementation of both the FI and SI solvers used herein are open-source and available as a part of MRST [25].

3 Discretizations

We discretize (1) to (10) using the typical low-order implicit finite-volume schemes commonly used in subsurface simulation: We use a two-point flux approximation to discretize the permeability and pressure gradient and single-point upwinding for the mobility and density terms. We refer the reader for more details and on specific implementations used herein to [22] and [13].

As noted in the previous section, we let $\mathbf{R}_{FI} = 0$ refer to the converged fully-implicit solution. Similarly, $\mathbf{R}_{SI} = 0$ refers to the sequential-implicit solution, where the nonlinear pressure equation $\mathbf{R}_P = 0$ is first solved in order to obtain initial estimates of the pressure solution $\mathbf{x}^{n+1/2}$. This is followed by the solution of the transport system $\mathbf{R}_T = 0$ to get $\mathbf{x}^{n+1}$. This strategy is outlined in Figure 1 together with details on the choice of primary variables.

4 Domain-decomposition strategy

The strategy considered in this paper is a predictor-corrector approach where $\mathbf{R}_{pred}(\mathbf{x}, \mathbf{x}^0, \Delta t, \mathbf{q}, \Omega)$ represents the predictor and $\mathbf{R}_{corr}(\mathbf{x}, \mathbf{x}^0, \Delta t, \mathbf{q}, \Omega)$ represents the corrector residuals. Here, $\mathbf{x}$ is the state vector, which defines the solution state uniquely, $\mathbf{x}^0$ the state vector at the previous time-step, Δt the current time-step, $\boldsymbol{q}$ the boundary conditions and Ω the solution domain. We seek $\mathbf{x}$ such that

$$\|\mathbf{R}_{corr}(\mathbf{x}, \mathbf{x}^0, \Delta t, \mathbf{q}, \Omega)\| < \epsilon$$

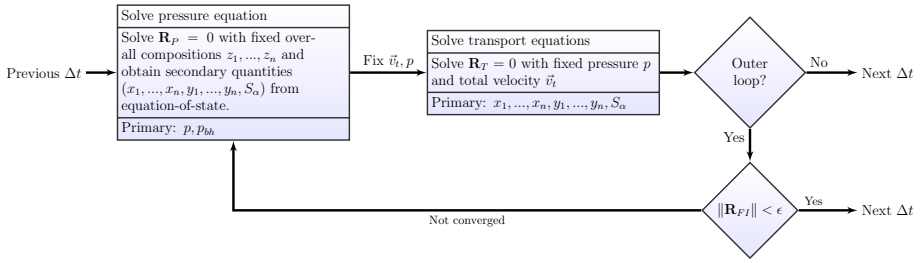


Figure 1: Outline of the sequential-implicit algorithm for compositional flow, with the optional sequential-fully-implicit outer-loop required for converging to the fully-implicit solution.

for some norm and tolerance, while we avoid solving

$$\mathbf{R}_{corr}(\mathbf{x}, \mathbf{x}^0, \Delta t, \mathbf{q}, \Omega) = 0$$

on the full domain Ω directly.

One pass of the proposed algorithm can be summarized as follows:

1. Solve $\mathbf{R}_{pred}(\mathbf{x}, \mathbf{x}^0, \Delta t, \mathbf{q}, \Omega) = 0$ to obtain initial estimate of $\mathbf{x}$.
2. Compute error indicators for estimate $\mathbf{x}$ and define local domain Ω_l.
3. Solve $\mathbf{R}_{corr}(\mathbf{x}_l, \mathbf{x}_l^0, \Delta t, \mathbf{q}, \Omega_l) = 0$ to obtain better values for $\mathbf{x}_l$, the values of $\mathbf{x}$ in local domain Ω_l.
4. Check if the corrected $\mathbf{x}$ is sufficiently close to the solution on the full domain $\mathbf{R}_{corr}(\mathbf{x}, \mathbf{x}^0, \Delta t, \mathbf{q}, \Omega) = 0$.

The central idea is that the predictor will produce a solution that is fairly accurate in most respects, with some *localized* error which can be resolved by the corrector on smaller sub-domains with the boundary conditions computed from the predictor solution.

4.1 Sequential implicit predictor-corrector

With the description of the general framework taken care of, we will now consider the specifics of applying it to a sequential implicit (SI) algorithm. Due to the splitting errors, one pass of the pressure and transport systems only leads to a fully converged FI residual in a few limited cases. SI approaches rely on time-step selection or smoothing to keep the splitting errors small. However, it is still challenging to obtain a-priori error estimates for multicomponent flows and smoothing may remove important properties of the solution. If we consider SFI approaches where a small FI residual is required, it will be necessary to perform multiple iterations of the outer loop, with each iteration consisting of pressure and transport solves. Moreover, repeated outer-loops can sometimes have unsatisfactory convergence for compositional problems. This motivates our approach, since compositional problems are both computationally intensive to solve

and has localized error terms due to compressibility, mass-exchange between phases and generally shorter time-steps than e.g. black-oil type descriptions. We let $\mathbf{R}_{SI}$ be the predictor $\mathbf{R}_{pred}$ and take $\mathbf{R}_{FI}$ as the corrector $\mathbf{R}_{corr}$.

4.1.1 Error indicators

The main challenges in applying our predictor-corrector strategy are the formulation of error-indicators and the choice of local boundary conditions. If we consider the expression for total velocity,

$$\vec{v}_t(\lambda_\alpha, p_\alpha, \rho_\alpha | \alpha \in \{l, v, w\}) = \sum_{\alpha \in \{l,v,w\}} -\lambda_\alpha \mathbf{K}(\nabla p_\alpha - \rho_\alpha g \Delta z),$$

we observe that changes in mobility and capillary pressure are major contributors. We note that there are two types of errors [19] present in the solution of our SI scheme: i) total-velocity errors, where the lagged value for inter-cell flow terms may lead to differences in flow field; ii) saturation or pore-volume errors, where phase mass-exchange or compressibility leads to differences between the pore volume and the fluid volumes predicted by thermodynamics in cells after the converged transport. The total-velocity or total-rate error is computed for a given cell i with cell neighbors $N(i)$ by

$$(\mathbf{e}_v)_i = \max_{j \in N(i)} \left| \mathbf{Q_t}_{ij}(\mathbf{x}^{n+1/2}) - \mathbf{Q_t}_{ij}(\mathbf{x}^{n+1}) \right|$$

which measures the difference between the discrete total-volumetric rate $\mathbf{Q_t}_{ij}(\mathbf{x}^{n+1/2})$ used in the transport, and the discrete total-volumetric rate $\mathbf{Q_t}_{ij}(\mathbf{x}^{n+1})$ defined by the updated mobilities after the transport. This is the fully-implicit expression for flux, and indicates if the assumption of fixed total-velocity during transport was reasonable. We require that the maximum total-rate error is smaller than 1% a reference value equal to the maximum of the total-volumetric rates and the pore volume $\mathbf{V}_i$ of cell i divided by the time-step

$$(\mathbf{e}_v)_i < 0.01 \max \left(\max_{j \in N(i)} |\mathbf{Q_t}_{ij}|, \frac{\mathbf{V}_i}{\Delta t} \right). \tag{13}$$

The ratio of pore-volume to time-steps is here included to avoid flagging cells with negligible flow, even if when the relative error may be larger than 1%. We also consider the discrete saturation error, which is naturally a per-cell value

$$(\mathbf{e}_s)_i = \left| 1 - \sum_{\beta = w,l,v} (\mathbf{S}_\beta)_i \right|$$

and we will flag all cells above a certain saturation error threshold,

$$(\mathbf{e}_s)_i < 0.001.$$

In addition, we flag cells that are perforated by wells which are not converged after transport.

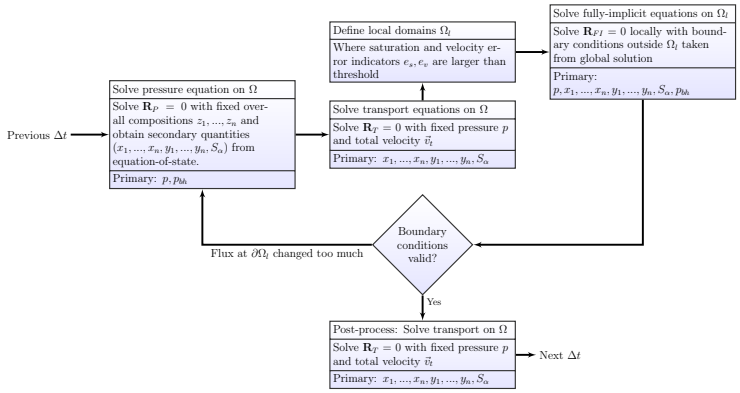


Figure 2: Algorithmic outline for the domain-decomposition sequential-implicit solver for compositional flow.

4.1.2 Corrector

As mentioned before, we intend to use the fully-implicit discretization to correct the sequential-implicit solution. After the local sub-domains Ω_l have been selected by the error indicators, we pose a local problem with boundary conditions taken from the global state. We use a Dirichlet boundary condition with fixed pressure, mobility and composition at the boundary, allowing the fluxes themselves to vary,

$$\{\vec{v}_\alpha(\mathbf{x})|\mathbf{x} \in \partial\Omega_l\} = -\lambda_\alpha \mathbf{K}(\nabla p_\alpha - \rho_\alpha g \nabla z).$$

We note that other choices are possible: Using for example total volumetric or mass fluxes as the inflow or outflow boundary conditions. There is little difference between the choices if the error indicators successfully captures part of the domain where significant errors are present. For the fully converged solution, taking either type of boundary conditions around some sub-region will obviously have the same local solution. Fixed pressure boundary conditions have the advantage of producing continuous pressure in incompressible regime where the solution becomes non-unique for pure Neumann boundary conditions, or when there is changes in mobility at the boundary. After the local problem has been solved to convergence, we now have a continuous pressure and regions with significant errors has been resolved by the corrector. In order to assess if the boundary conditions were accurate, we check the now updated fluxes at the boundary against their values before the local variables were updated. If this is below the threshold of (13), we can continue to the next time-step. While both our sequential implicit and fully-implicit solvers are exactly mass-conservative, these local boundary conditions may introduce small mass-errors unless $\|\mathbf{e}_v\|$ is exactly zero. For this reason, we apply a post-processing transport-solve with the updated velocity field on the global domain. This is simply the re-application of the transport-solver with essentially a converged solution as initial guess. The final algorithm is shown in Figure 2.

5 Numerical examples

We will consider a number of different test cases, demonstrating the behavior of our proposed algorithm in several scenarios. The `compositional` module of the Matlab Reservoir Simulation Toolbox (MRST) [25] is used in all examples, with Peng-Robinson as the equation-of-state for the hydrocarbon phases. Comparisons are made between several different solvers. The fully-implicit solver is referred to as FIM, and will be our reference solution. SFI1 refers to a pressure solve followed by a transport solve in the sequential-implicit solver. The solution from SFI1 is in most cases close to the fully-implicit solution but splitting errors due to mobility or mass-exchange effects may be present in regions with strong pressure-transport coupling. SFIO refers to the SFI solver with outer loops enabled: For each step, the solver iterates between solving the nonlinear pressure and transport equations until the FIM residual is sufficiently converged. Finally, SFIDD refers to our new predictor-corrector domain decomposition solver.

For each case, we report the pressure and transport iterations for all solvers, except for the fully-coupled FIM where we report Newton iterations of the coupled system. For SFIDD, the additional transport iterations of the post-processing of a completed outer loop is separated out and flagged as "SFIDD-post". We report the number of outer-loops for all sequential solvers and compare them with the Newton iterations of the FIM. For SFI1, when there is no time-step cut, there is always one outer-loop per time-step. We note that the computational cost of one outer loop varies between algorithms:

- For the fully-implicit solver, it amounts to one solution of a mixed-hyperbolic-parabolic linearized system with n degrees of freedom per cell, where the pressure leads to strong inter-block connections. Here, a CPR-type preconditioner is generally the most efficient strategy [31, 2].
- In the case of SFIO, each iteration will entail the solution of both a nonlinear parabolic system with one degree of freedom per cell for pressure which is amenable to e.g. multigrid or multiscale methods. In addition, the transport corresponds to a nonlinear system with n degrees of freedom. These variables are of the same hyperbolic [29, 28] type and do not require tailored preconditioners. Generally, the cost of the outer iterations becomes smaller as the initial guess for compositions improves during the iterations.
- SFIDD corresponds to the essentially the same solution process as SFIO, with the additional cost of solving a smaller fully-implicit system in the correction step.

For our prototype Matlab implementation, we do not report runtime results. Although we do observe speed-up for the SFIDD solver, it is difficult to reliably assess the efficiency when the solver is implemented in an interpreted language. We leave this comparison for future work. In each example, we also plot detailed information on the magnitude of the local domains taken as fully-implicit for SFIDD together with the fraction of the domain where CFL is larger than unity. While the solvers under consideration are not explicit, we also plot both the maximum CFL for saturations (CFL_S)

and compositions (CFL_X) using the form given by [24] to give some impression of the time-step length relative to the dynamics of the problem.

5.1 Immiscible one-dimensional displacement

In this first test case we are considering an immiscible test case from [30], which is also considered in [22] as a validation of the MRST compositional solvers with respect to a commercial simulator. We inject a gas mixture of Methane and Carbon dioxide in a 100-cell 1D reservoir initially in a pure liquid state containing n-Butane, n-Decane and Methane. The solution of this problem, shown in Figure 3 contains both a single-phase liquid region before the displacement front, a two-phase region where the front is moving and a pure gas phase behind the front. For this reason, it is a natural first example, demonstrating the proposed solver to a problem with multiple phase transitions. Figure

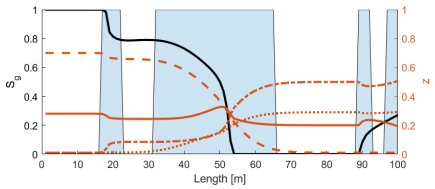
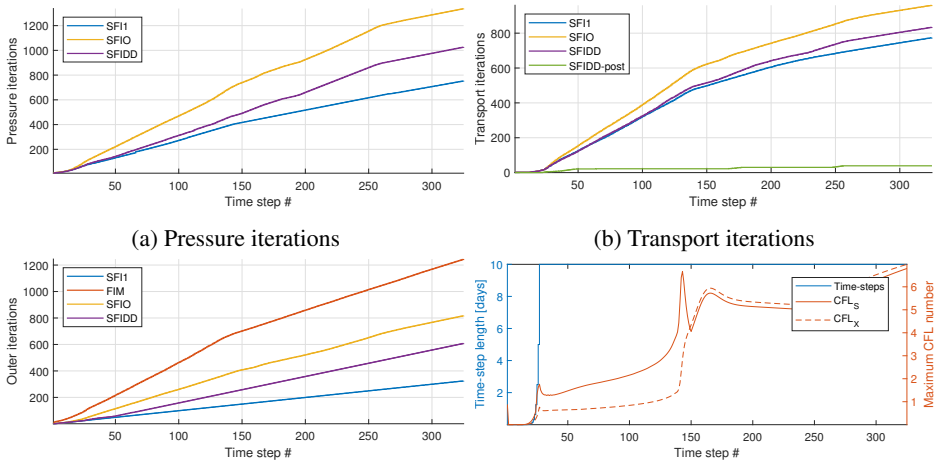


Figure 3: The displacement front for the immiscible test case part-way through the simulation. The gas saturation is plotted in black, with different components shown in red. Light blue indicates cells for which a local update was performed for this specific time-step.

4 show the cumulative number of iterations as well as the maximum CFL number [4] as a function of time. We observe in Figure 4c that SFIO uses on average 2.55 outer loops to converge. In this example, there are no time-step cuts. Consequently, the SFI1 solver will only perform a single outer loop per time-step and is the best case in terms of efficiency for any sequential solver. Our SFIDD has a number of outer-loops between the optimal number of SFI1, which always uses a single pass, and the upper limit of SFIO. On average, the SFIDD converges in 1.87 outer-iterations per time-step. Similar trends are observed when comparing the pressure iterations in Figure 4a and the transport iterations in 4b. While the fully-implicit system is difficult to compare directly to the pressure/transport system, we consider about one pressure and transport iteration per fully-implicit iteration for the same time-step to be a promising decoupling strategy: The cumulative number of iterations for the SFIDD method amounts to 82% of the FI method Newton iterations for the pressure and 70% of the FI method Newton iterations for the transport. Figure 5a shows that each local FI solution is converging in an average of 2.5 Newton iterations for the first half of the simulation and 1.5 Newton iterations for the second half of the simulation. Figure 5b shows that the areal extension of the FI domains for this challenging 1D test case can amount up to 50% of the simulation domain for the first outer-iterations and then never goes beyond 15% for the following outer-iterations. If we consider the blue regions in Figure 3, we observe that the FI updates are located near the producer, the leading rarefaction/shock

and the trailing shock. As the displacement is covering a significant part of the domain, the localized regions correspond to a fairly large fraction of the total domain. Note that the major contributor to the local regions in Figure 3 is the extent of the rarefaction wave from cells 40 to 55: We would expect that the localized regions would be much smaller for a stable displacement with more favorable viscosity ratio and consequently smaller multiphase region.



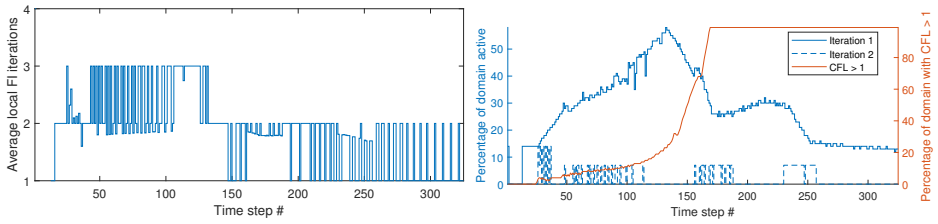
(a) Pressure iterations

(b) Transport iterations

(c) Outer iterations

(d) Time-steps and maximum CFL number

Figure 4: Iterations, time-steps and CFL numbers for the one-dimensional immiscible test case.



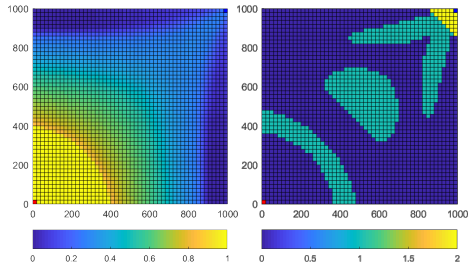
(a) Number of local iterations

(b) Average percentage of fully-implicit cells

Figure 5: Statistics for the SFIDD solver for the immiscible test case.

5.2 Quarter five-spot CO2 injection

In this second test case we are using the quarter five-spot test case where we inject a Carbon dioxide rich gas into a 2D reservoir with dimensions 1000 by 1000 meters and uniform permeability of 50 mD as shown in Figure 6a. The initial state of the reservoir contains a mixture of Methane, Carbon dioxide and n-Decane. Again, we plot the



(a) Gas saturation (b) Local updates

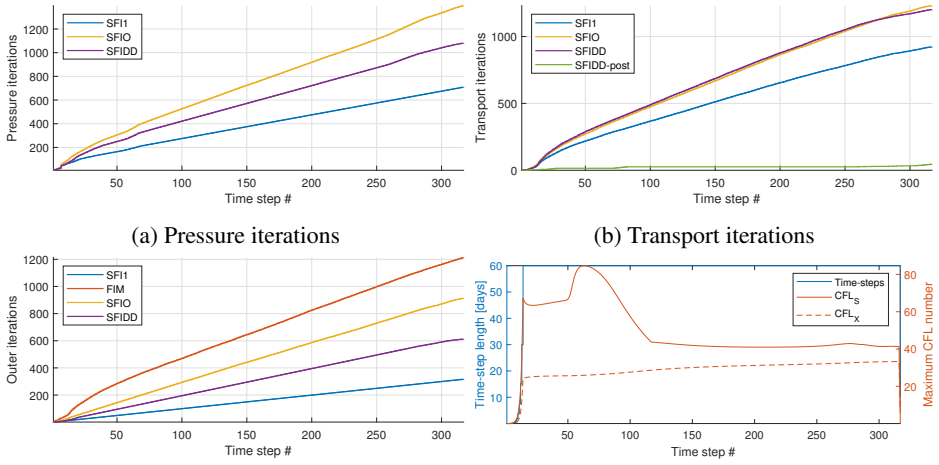
Figure 6: Gas saturation (left) and local update count (right) part-way during the quarter-five-spot test case. We observe that the localized updates for this step are centered around the phase transitions and near-producer regions.

cumulative iterations, time-step and maximum CFL in Figure 7. We again observe in Figure 7c that SFIO requires more than twice as many outer-iterations as SFI1 to converge. Our SFIDD has a number of outer-loops between the optimal number of SFI1 and the upper limit of SFIO. In average, the SFIDD converges in 1.85 outer-iteration, indicating that for later time-steps, the total-velocity decoupling is very accurate. We see a significant improvement in the number of pressure iterations shown in Figure 7a for the SFIDD solver. For the transport iterations shown in Figure 7b, SFIDD and SFIO gives comparable number of iterations. This is due to later time-steps using generally few transport iterations for either solver, and a few early time-steps where the transport solver in SFIDD uses additional iterations. The cumulative number of iterations for the SFIDD method corresponds to to 86% and 104% of the FIM method Newton iterations for the pressure and transport, respectively. The local FIM iterations seen in Figure 8a shows an average of 2 Newton iterations during the simulation, and the largest number of iterations is observed during the initial injection dynamics. Figure 8b shows that the areal extension of the FI domains for this challenging 2D test case can amount up to 45% of the simulation domain the first outer-iterations at the beginning of the simulation. When a second outer iteration is required, the localization is limited to a very small amount of cells.

Figure 6a shows the gas saturation after 60 time-steps and figure 6b shows the domains for the local update. We see a large distribution of the two-phase region, with localized updates limited to the trailing shock, the producer inflow regions and a segment of the front. We remark that as the problem is symmetric, the local domains are symmetric as well.

5.3 Three-phase displacement in a channelized medium

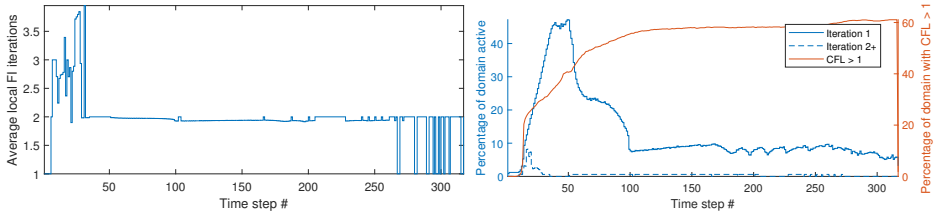
In this example, we consider layer 45 of the SPE 10 model 2 benchmark, which corresponds to the channelized fluvial Upper Ness formation. This specific layer was chosen due to the dual flow channels which meet near the end of the domain as can be seen in Figure 9. We place two injectors in the upper and lower channel, injecting gas and water respectively. The capillary pressure is taken from the Norne field model [23] and



(a) Pressure iterations (b) Transport iterations

(c) Outer iterations (d) Time-steps and maximum CFL number

Figure 7: Iterations, time-steps and CFL numbers for the quarter five-spot test case.

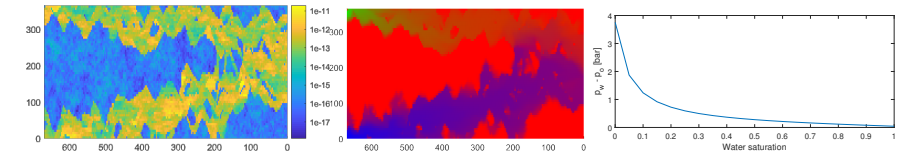


(a) Number of local iterations (b) Average percentage of fully-implicit cells

Figure 8: Statistics for the SFIDD solver for the quarter five-spot test case.

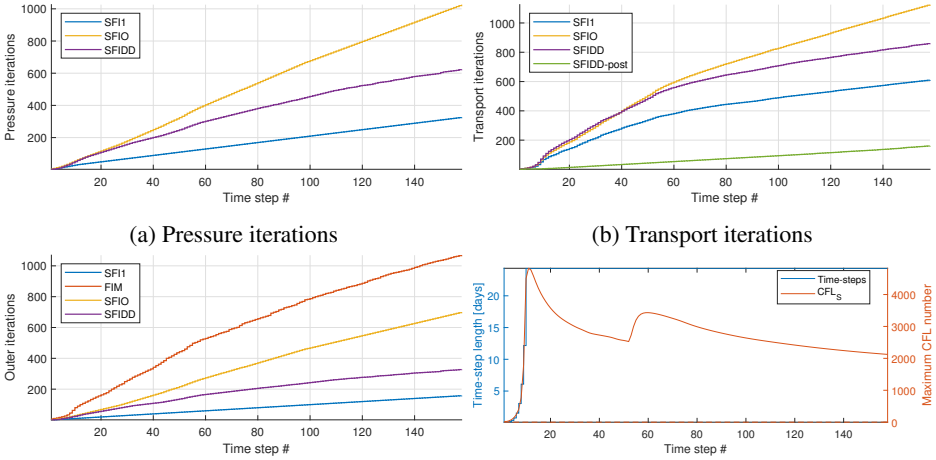
is plotted in Figure 9c. The phases are assumed to be immiscible, with viscosities of 1, 5 and 0.1 cP for water, oil and gas. The corresponding Corey exponents are 2, 3 and 1, with compressibility factors of 10^{-6}, 10^{-4} and 10^{-3} bar^{-1}, respectively. We inject one pore-volume in total over 10 years divided into 150 time-steps. The resulting CFL maximum CFL numbers in Figure 10d are large in magnitude as a consequence of the channelized structure.

When we examine the iterations in Figure 10 we observe that SFIDD again falls somewhere inbetween SFIO and SFI1 in terms of the number of iterations. SFIDD is closer to SFI 1 when considering the outer iterations in Figure 10c and observes a significant reduction in the number of pressure iterations when compared to SFIO. We note that a significant amount of cells (up to 38%) are flagged as fully-implicit before breakthrough in Figure 11b, but this number drops sharply after breakthrough even as 50% of the domain has CFL larger than unity.



(a) Permeability (b) Saturations (c) Water-oil capillary pressure

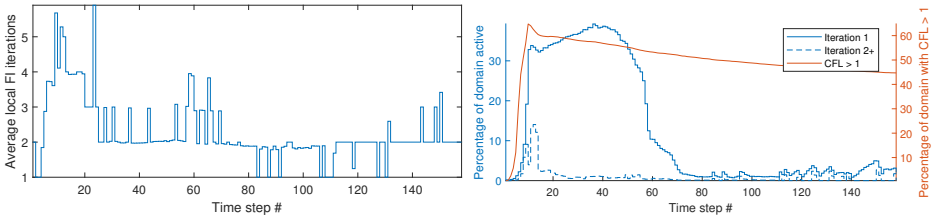
Figure 9: Permeability, phase saturations and capillary pressure function for the channelized SPE10 test case.



(a) Pressure iterations (b) Transport iterations

(c) Outer iterations (d) Time-steps and maximum CFL number

Figure 10: Iterations, time-steps and CFL numbers for the channelized SPE10 case.

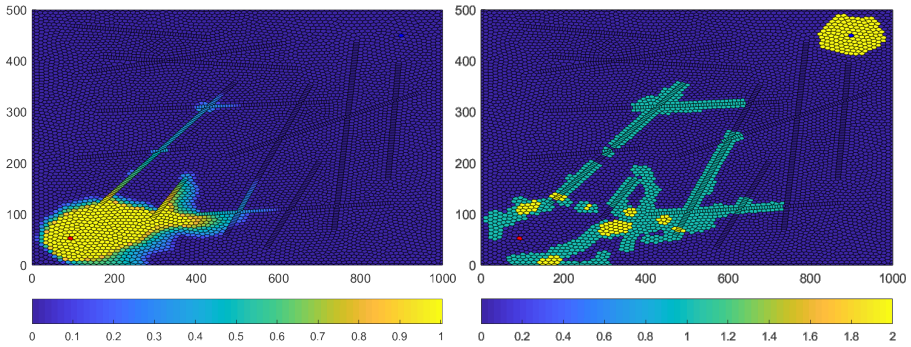


(a) Number of local iterations (b) Average percentage of fully-implicit cells

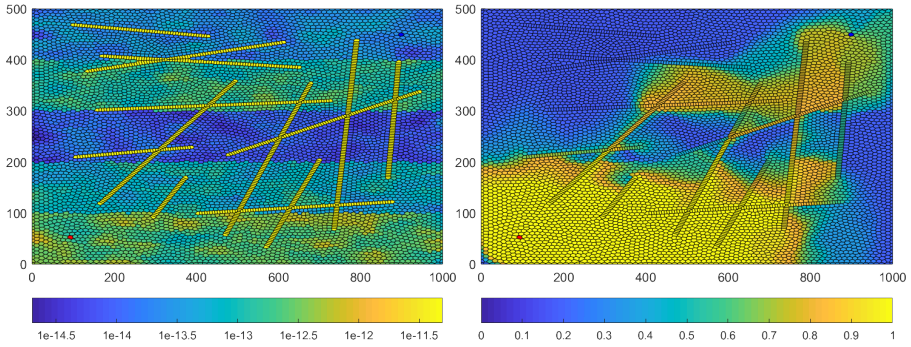
Figure 11: Statistics for the SFIDD solver for the three-phase displacement in a channelized layer of SPE10.

5.4 Unstructured grid

In this unstructured test case from [22] we consider an unstructured 2D reservoir with fractures, where the fluid model and resident/injection compositions are taken from the SPE 5 benchmark [12]. The permeability and final gas saturation are plotted in



(a) Gas saturation (b) Local updates

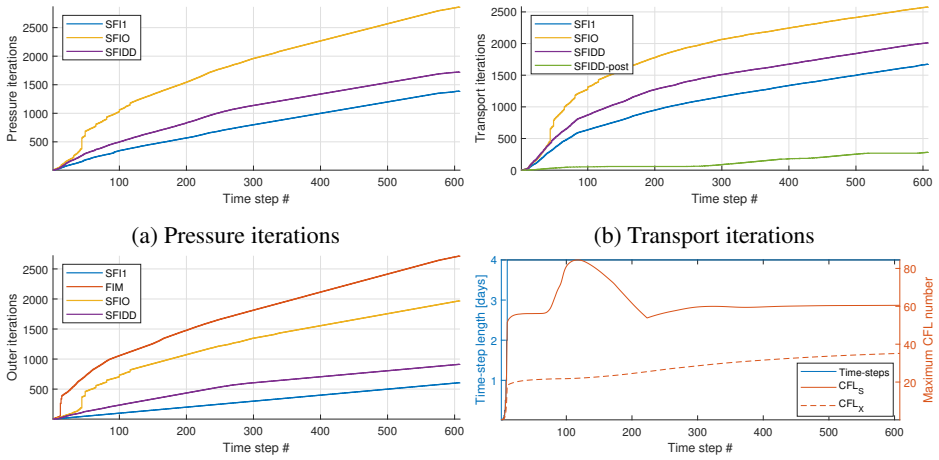


(c) Permeability (d) Final gas saturation

Figure 12: Permeability, gas saturation at different times and local updates for the unstructured fractured test case.

Figure 12, indicating that the permeability contrasts significantly impact the flow field. The different nonlinear iterations are plotted in Figure 13. We observe that the CFL numbers after the gas reaches the fractures rises to 60. We observe in Figure 13c that SFIO needs more than three times as many outer-iterations as SFI1 before convergence is achieved. On the other hand our SFIDD needs only 30% more outer-iterations than the optimal number of SFI1. In average, the SFIDD converges the time-step in 1.50 outer-iteration. Generally, the reduced number of iterations for SFIDD is also observed for the pressure and the transport shown in Figures 13a and 13b, only requiring a few more iterations than the optimal SFI1. When comparing the total number of FIM iterations to the pressure/transport systems, SFIDD uses 63% iterations for the pressure and 85% of the total FIM iterations.

When we consider the localized solves plotted in Figure 14b, we note that the local updates are limited to a smaller region than in the previous test-cases. As the two previous examples were homogeneous 1D and 2D displacements, it is not surprising that the large permeability variations leads to a more complex flow field where the largest errors are localized near fracture-matrix interactions. The largest local domain during the initial injection is about 17% of the global domain, with later steps taking less than 5% of the cells as local. When a second loop is required, the local cells is always a small fraction of the domain. If we consider the early-time gas front shown in Figure 12a together with the corresponding local updates in Figure 12b, we can verify that the error is localized around the fractures and near the producer. Apart from a single step in the early injection period, the number of local Newton iterations always around 2, as shown in Figure 14a. It is clear that in this heterogeneous case, SFIDD is much more efficient than SFIO. We expect this kind of localized behavior as a result of heterogeneities to be more typical of realistic heterogeneous simulations than the previous two test-cases.



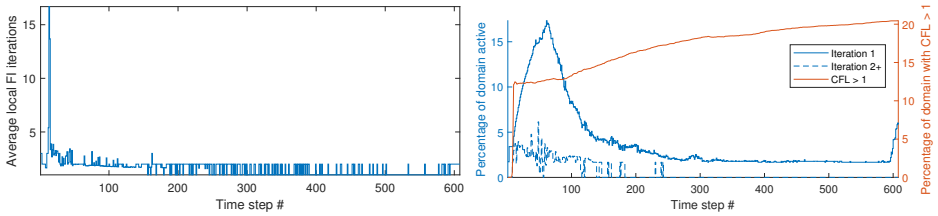
(a) Pressure iterations

(b) Transport iterations

(c) Outer iterations

(d) Time-steps and maximum CFL number

Figure 13: Iterations, time-steps and CFL numbers for the unstructured test case.



(a) Number of local iterations

(b) Average percentage of fully-implicit cells

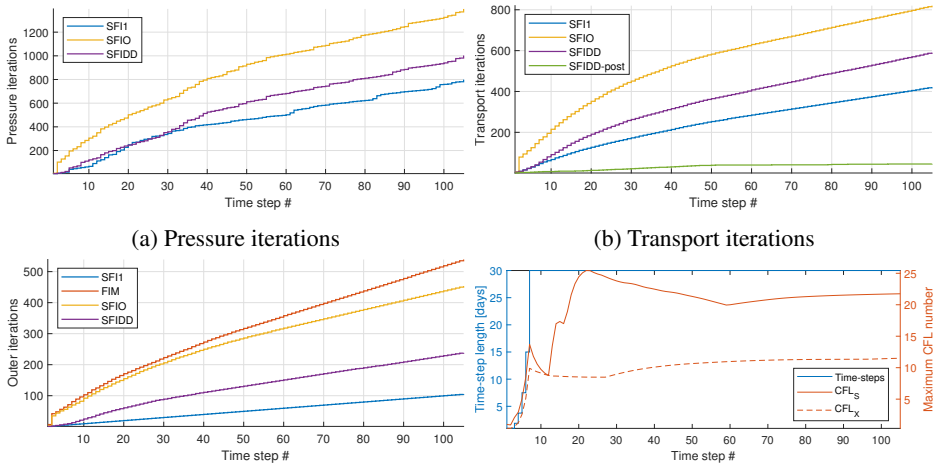
Figure 14: Statistics for the SFIDD solver for the unstructured test case.

5.5 Gas injection in a corner-point model

For this case, the model is a synthetic, but realistic shallow-marine reservoir taken from the SAIGUP project [16] and upscaled to 6980 grid cells. The fluid model is taken from [15], and is a system made up of six lumped pseudo-components. The reservoir is initialized at 100 bar pressure at the shallowest point, with a temperature of 387 degrees K. A gas-cap is present, containing mostly light hydrocarbon components, initially segregated due to gravity. Three producers operating at 80 bar are present in the four upper layers of the model – one in the gas-zone, and two in the oil-zone near the gas. Four injectors in the lower four layers inject a fixed rate of CO_2 near the producers over 3000 days, resulting in significant pressure build-up and free gas in the lower layers. We use just over 100 time-steps in total. We have plotted the permeability field, the final gas saturation and the well configuration in Figure 17

The iteration results are reported in Figure 15. As in the previous cases, we observe a significant improvement in the number of pressure and transport iterations for SFIDD compared to SFIO. The wells are in this case limited to a smaller part of the domain, typical of many real-field simulation models. Figure 16b shows the fraction of cells that are solved with the local FI solver. The maximum is 17% of the active cells. We expect this number to be smaller on larger field-scale models. Apart from a few of the early time-steps, the local Newton iterations shown in Figure 16a maintain values below 5 and going as low as 1 from the middle of the simulation. For this case, SFIDD uses 177% and 108% of the fully-implicit iterations in the pressure and transport, respectively. The reason for the generally high number of pressure iterations in the sequential solvers may be a combination of the injection scenario itself and the relatively challenging fluid model. A possible reason could be that the decoupling of the pressure system may not take into account all the derivatives as described in [19]. This will be investigated as a part of future work.

Figure 18a shows the BHP profiles of the 4 injection wells for SFI1, FIM and SFIDD and Figure 18b shows the gas volume fraction of the 3 producing wells for SFI1, FIM and SFIDD. We observe a significant discrepancy between SFI1 and FIM for both the early injector bottom-hole pressures and gas cut in the producers. This difference is not present for SFIDD, which accurately reproduces the FIM solution.



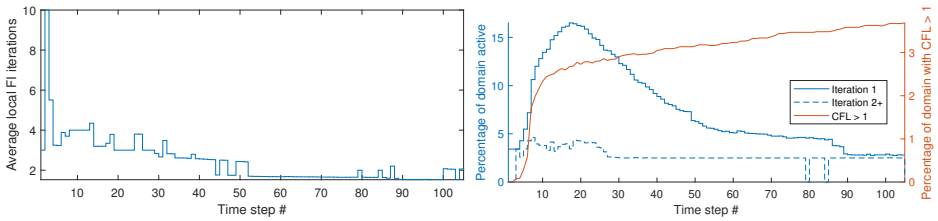
(a) Pressure iterations

(b) Transport iterations

(c) Outer iterations

(d) Time-steps and maximum CFL number

Figure 15: Iterations, time-steps and CFL numbers for the SAIGUP test case.



(a) Number of local iterations

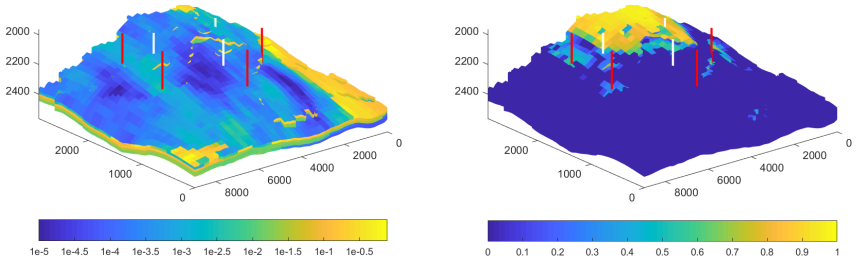
(b) Average percentage of fully-implicit cells

Figure 16: Statistics for the SFIDD solver for the SAIGUP test case

5.6 Alternate gas-water-gas 2D case

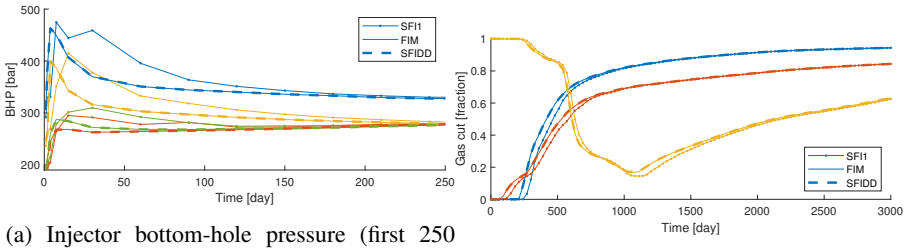
For our final test case, we consider the alternate gas-water-gas injection problem from [19]. The final saturation distribution is plotted together with the permeability field sampled from SPE 10, model 2 in Figure 19. We inject gas for the first period of 5000 days, resulting in CFL values up to 140. For the second injection period of additional 5000 days, inject water in the reservoir from the same injector. As the viscosity of water is higher than the previously injected gas, the displacement is now stable and is dominated by a self-sharpening shock between the gas phase and the water phase. The resulting well responses are shown in Figure 20. The time-steps are initially quite small and increase until a CFL of around 150 is reached. Finally for the last period, gas is again re-injected for 5000 days. CFL for this period is around 80.

Figure 21 contains the plots of iterations and time-steps. We observe in Figure 21c that SFIO needs more than four times as much outer-iterations than SFI1 to converge. On the other hand our SFIDD needs twice as much outer-iterations than the optimal number of SFI1. In average, SFIDD converges the time-step in 2.01 outer-iteration.



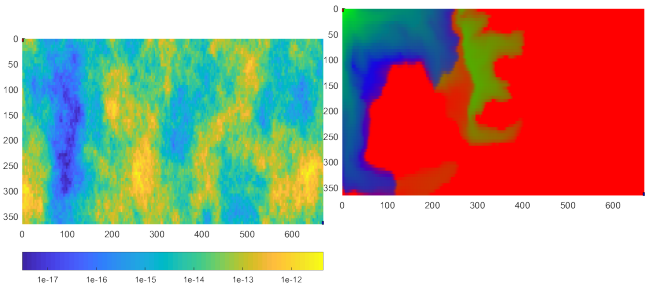
(a) Vertical permeability (b) Final gas saturation

Figure 17: Vertical permeability and final gas saturation for the SAIGUP corner-point model with CO_2 injection.



(a) Injector bottom-hole pressure (first 250 days) (b) Gas volume fraction

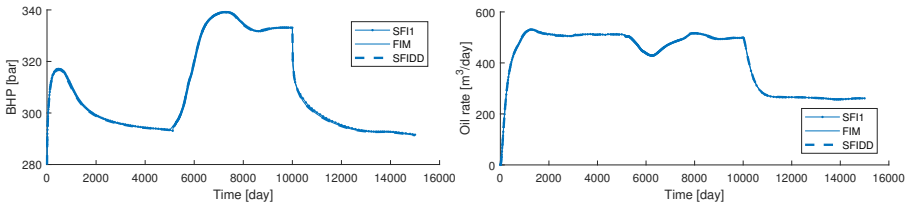
Figure 18: The bottom-hole pressure in the injectors for the first 250 days and gas-cut for the injectors for the SAIGUP corner-point model.



(a) Permeability (b) Final saturation distribution

Figure 19: Permeability and final saturations for the SPE 10 subset with gas-water-gas injection. Note that green, blue and red correspond to vapor, aqueous and liquid phases respectively.

The same applies for the pressure and the transport iterations of SFIDD in Figures 21a and 21b that converge in a few more iterations than SFI1, which has best possible outer



(a) Injector bottom-hole pressure

(b) Producer oil-rate

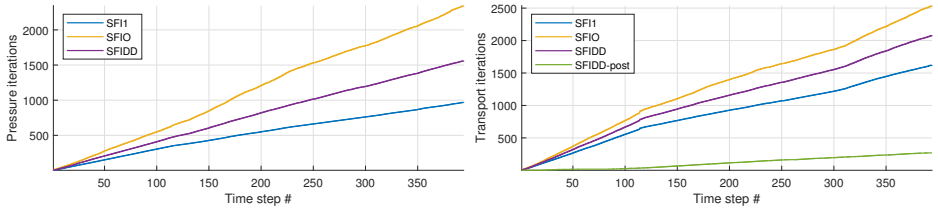
Figure 20: Injector bottom-hole pressure and producer oil-rate for the SPE 10, model 2 subset.

convergence rate, albeit without any control on the solution error relative to FIM. We observe that the cumulative number of iterations for the SFIDD method corresponds to 72% of the FIM iterations for pressure and 108% of the FIM iterations the transport. Figure 22a shows that each local FIM solution is converging in an average of 2.5 Newton iterations for each time-step. Figure 22b shows that the areal extent of the local domains for this highly heterogeneous 2D test case can amount up to 15-20% of the simulation domain for the periods with gas injection. For the water injection, only a few percent of the domain are flagged as local, reinforcing the conclusion that errors are localized around the high-flow segment of the multiphase region. The errors are more localized for stable displacements where a more viscous fluid displaces a less viscous fluid than for unstable displacements where a less viscous fluid displaces a more viscous fluid. This occurs as the displacement is dominated by the leading shock, and the transition between gas and water is sharp. The assumption of fixed total-velocity is very accurate in single-phase regions, occurs on either side of the shock. We observe that after a first local FI processing has been performed during the first outer-loop, the additional local FI solves only involve a few cells. In most time-steps, the cost of the FI corrector algorithm is equivalent to a small fraction of SFI1 solver, making SFIDD method competitive with respect to SFIO and FIM.

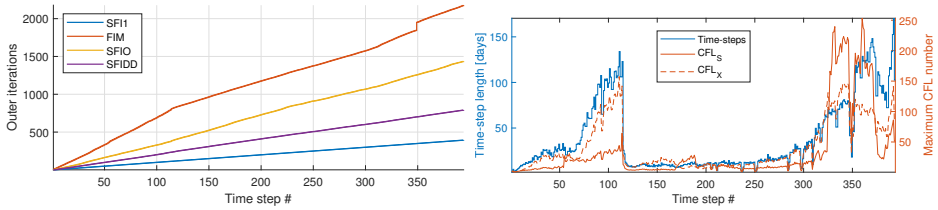
5.7 Displacement and pressure error

In the preceding we have measured error with the well responses. This is sufficient for many purposes, as the well-responses such as rates and/or bottom-hole-pressures correspond to the externally observed effect of injection or production when applied to a subsurface reservoir. Consequently, in an engineering context, these are often the most important outputs from a reservoir simulator. We generally observe perfect agreement when SFIO, SFIDD and FIM converge to similar tolerances, with some differences seen relative to SFI. These errors, while not always significant, are usually a delayed breakthrough when the mobility of the system is increasing, leading to an under-estimation of total velocity in each pressure step.

In order to assess the error between the solvers more systematically, we will consider two types of error. The last step in either of the four methods is always the solution of the mass-balance equations and they are therefore exactly mass-conservative to within the same tolerance. The mass-distribution in the domain, however, differs be-

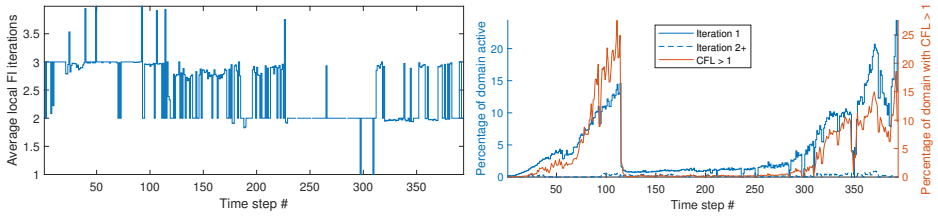


(a) Pressure iterations (b) Transport iterations



(c) Outer iterations (d) Time-steps and maximum CFL number

Figure 21: Pressure, transport and outer iterations as well as time-steps and estimated maximum CFL for the different solvers for the SPE 10, model 2 gas-water-gas case.



(a) Number of local iterations (b) Average percentage of fully-implicit cells

Figure 22: Number of local iterations and average number of fully-implicit cells in each local solve for the SFIDD solver for the 10, model 2 gas-water-gas case.

tween the solvers. We let the mass of component c in cell i after time-step k be defined as

$$M_{ic}^k = \begin{cases} \phi_i \rho_w S_w & \text{at } t_k \text{ if } c = m+1 \\ \phi_i (X_c \rho_l S_l + Y_j \rho_v S_v)_i & \text{at } t_k \text{ otherwise.} \end{cases} \tag{14}$$

In the following we let $\tilde{Z}$ refer to the value of a function Z which is to be compared with the reference. The total mass-discrepancy error is then the sum of mass-discrepancy for each component, integrated over all n_t time-steps and n_c cells, relative to the component mass at each time-step,

$$E_m = \frac{1}{t_{end}} \sum_k^{n_t} \Delta t_k \sum_c^{m+1} \frac{\sum_i^{n_c} |\tilde{M}_{ic}^k - M_{ic}|}{\sum_i^{n_c} M_{ic}^k}. \tag{15}$$

Example	SFI1	SFIO	SFIDD
1) Immiscible 1D	6.037e-03	2.367e-03	3.404e-03
2) Quarter-five-spot	1.454e-03	9.504e-05	1.475e-04
3) Channelized SPE10	8.071e-04	2.401e-06	4.546e-05
4) Unstructured fracture	1.856e-03	2.844e-04	3.108e-04
5) SAIGUP	3.496e-03	2.302e-05	4.041e-05
6) Alternate gas-water-gas	2.455e-03	7.099e-05	1.032e-04

Table 1: Total mass-distribution error E_m relative to FIM for the different examples

Example	SFI1	SFIO	SFIDD
1) Immiscible 1D	2.562e-03	7.349e-04	7.754e-04
2) Quarter-five-spot	8.797e-04	1.258e-05	5.861e-05
3) Channelized SPE10	4.523e-03	5.710e-06	4.901e-04
4) Unstructured fracture	6.548e-04	1.056e-04	9.883e-05
5) SAIGUP	2.551e-03	1.734e-05	4.159e-05
6) Alternate gas-water-gas	1.333e-04	2.523e-06	5.222e-06

Table 2: Pressure error E_p relative to FIM for the different examples

Although it is not likely that a correct mass distribution is obtained with the wrong pressure, we also compute the pore-volume weighted pressure error,

$$E_p = \frac{1}{t_{end}} \sum_k^{n_t} \Delta t_k \frac{\sum_i^{n_c} |(\tilde{\phi}\tilde{p})_i^k - (\phi p)_i^k|}{\sum_i^{n_c} (\phi p)_i^k}. \tag{16}$$

Table 1 contains E_m for SFI1, SFIO and SFIDD. We generally observe that the error of SFIDD is around one order of magnitude smaller than for the SFI1 solver and comparable to SFIO. SFIO generally has slightly lower error than SFIDD as it exactly checks the relatively expensive fully-implicit residual rather than the indirect error indicators, but both are within the acceptable ranges where they are visually indistinguishable from the fully-implicit solution. For the pressure error in Table 2, we see a larger difference between SFI and SFIO and SFIDD. Here, the pressures of SFIO and SFIDD are comparable, indicating that we observe no ill-effects from our chosen local boundary conditions.

6 Conclusion

A full compositional description is a computationally intensive approach for reservoir simulation, but may be required in order to model reliably important processes like miscible flooding and CO_2 sequestration. Sequential-implicit methods are one class of proposed solution strategies which allows a decoupling of the parabolic pressure from the (nearly) hyperbolic composition variables. By separating the equations according

to their mathematical structure, it is possible to design specialized solvers which target specific problems, increasing accuracy and efficiency. However, the decoupling strategies employed to design separate pressure and transport systems may introduce errors in the solutions which must be resolved in order to produce reliable predictions for engineering decisions. In this work, our goal is to develop a scheme which reproduces the FI solution at a cost as close as possible to the cost of a SI method or a SFI method with a single outer-iteration. We have proposed a domain-decomposition strategy based on a predictor-corrector pair of models together with problem-specific error indicators.

The predictor model is applied globally and corresponds to the regular SI method. The local corrector model is the FI method, where the local domains are chosen according to both saturation and velocity error indicators with a clear physical interpretation. After convergence is achieved, an additional post-processing with a local transport solve is followed to ensure strict local mass-conservation in all cells. In several challenging two and three-phase test cases, including stratigraphic corner-point and fully unstructured grids, we observe that the proposed algorithm generally converges in one or two outer-iterations where the SFI would need up to four iterations to converge. The areal extension of the local FI domains varies from negligible to up to 50% of the simulation domain for the different test cases, depending on the dimension and physical effects. We note that the errors are generally very local for stable displacements, while the invaded two-phase regions and the associated errors are larger in extent for unstable displacements. Generally large local domains are only observed in a few time-steps throughout a longer simulation. They correspond to specific events such as start of water or gas injection and the domains for local solves are small or non-existent for most of the time-steps.

We consider the described method to be a promising approach for field-scale compositional simulation: Typical models often have smaller interaction domains between wells where nonlinear dynamics is present for a given time-step, while most of the domain can efficiently be solved by a single pass of the SI method. We believe that the SFIDD solver will be an ideal strategy for problems where SFI is not sufficiently accurate and SFIDD may outperform its constitutive SI and FI parts.

7 Acknowledgements

Olav Møyner is funded by VISTA, which is a basic research program funded by Equinor and conducted in close collaboration with The Norwegian Academy of Science and Letters. Arthur Moncorgé would like to thank TOTAL management for permission to publish this work.

References

[1] Acs, G., Doleschall, S., Farkas, E.: General purpose compositional model. Society of Petroleum Engineers Journal **25**(04), 543–553 (1985). DOI 10.2118/10515-pa. URL http://dx.doi.org/10.2118/10515-PA

[2] Cao, H., Tchelepi, H.A., Wallis, J.R., Yardumian, H.E., et al.: Parallel scalable unstructured cpr-type linear solver for reservoir simulation. In: SPE Annual Technical Conference and Exhibition. Society of Petroleum Engineers (2005)

[3] Coats, K.: A note on IMPES and some IMPES-based simulation models. SPE Journal **5**(03), 245–251 (2000). DOI 10.2118/65092-pa. URL http://dx.doi.org/10.2118/65092-PA

[4] Coats, K.: IMPES stability: Selection of stable timesteps. SPE Journal **8**(02), 181–187 (2003). DOI 10.2118/84924-pa. URL http://dx.doi.org/10.2118/84924-PA

[5] Coats, K.H.: An equation of state compositional model. SPE Journal **20**(05), 363–376 (1980)

[6] Collins, D., Nghiem, L., Li, Y., Grabonstotter, J.: An efficient approach to adaptive-implicit compositional simulation with an equation of state. SPE reservoir engineering **7**(02), 259–264 (1992)

[7] Forsyth Jr, P., Sammon, P.: Practical considerations for adaptive implicit methods in reservoir simulation. Journal of Computational Physics **62**(2), 265–281 (1986)

[8] Hajibeygi, H., Tchelepi, H.: Compositional multiscale finite-volume formulation. SPE Journal **19**(02), 316–326 (2014). DOI 10.2118/163664-pa. URL http://dx.doi.org/10.2118/163664-PA

[9] Hou, T., Wu, X.H.: A multiscale finite element method for elliptic problems in composite materials and porous media. Journal of Computational Physics **134**(1), 169 – 189 (1997). DOI http://dx.doi.org/10.1006/jcph.1997.5682. URL http://www.sciencedirect.com/science/article/pii/S0021999197956825

[10] Jenny, P., Lee, S., Tchelepi, H.: Multi-scale finite-volume method for elliptic problems in subsurface flow simulation. Journal of Computational Physics **187**(1), 47 – 67 (2003). DOI http://dx.doi.org/10.1016/S0021-9991(03)00075-5. URL http://www.sciencedirect.com/science/article/pii/S0021999103000755

[11] Jenny, P., Lee, S., Tchelepi, H.: Adaptive fully implicit multi-scale finite-volume method for multi-phase flow and transport in heterogeneous porous media. Journal of Computational Physics **217**(2), 627 – 641 (2006). DOI http://dx.doi.org/10.1016/j.jcp.2006.01.028. URL http://www.sciencedirect.com/science/article/pii/S002199910600026X

[12] Killough, J., Kossack, C.: Fifth comparative solution project: evaluation of miscible flood simulators. In: SPE Symposium on Reservoir Simulation. Society of Petroleum Engineers (1987)

[13] Krogstad, S., Lie, K.A., Møyner, O., Nilsen, H.M., Raynaud, X., Skaflestad, B.: Mrst-ad–an open-source framework for rapid prototyping and evaluation of reservoir simulation problems. In: SPE reservoir simulation symposium. Society of Petroleum Engineers (2015)

[14] Lee, S., Wolfsteiner, C., Tchelepi, H.: Multiscale finite-volume formulation for multiphase flow in porous media: black oil formulation of compressible, three-phase flow with gravity. Computational Geosciences **12**(3), 351–366 (2008). DOI 10.1007/s10596-007-9069-3. URL http://dx.doi.org/10.1007/s10596-007-9069-3

[15] Mallison, B.T., Gerritsen, M.G., Jessen, K., Orr, F.M.: High order upwind schemes for two-phase, multicomponent flow. SPE Journal **10**(03), 297–311 (2005)

[16] Manzocchi, T., Carter, J.N., Skorstad, A., Fjellvoll, B., Stephen, K.D., Howell, J.A., Matthews, J.D., Walsh, J.J., Nepveu, M., Bos, C., Cole, J., Egberts, P., Flint, S., Hern, C., Holden, L., Hovland, H., Jackson, H., Kolbjørnsen, O., MacDonald, A., Nell, P.A.R., Onyeagoro, K., Strand, J., Syversveen, A.R., Tchistiakov, A., Yang, C., Yielding, G., Zimmerman, R.W.: Sensitivity of the impact of geological uncertainty on production from faulted and unfaulted shallow-marine oil reservoirs: objectives and methods. Petroleum Geoscience **14**(1), 3–15 (2008)

[17] Michelsen, M.L.: The isothermal flash problem. part i. stability. Fluid phase equilibria **9**(1), 1–19 (1982)

[18] Moncorgé, A., Tchelepi, H., Jenny, P.: Modified sequential fully implicit scheme for compositional flow simulation. Journal of Computational Physics **337**, 98 – 115 (2017). DOI 10.1016/j.jcp.2017.02.032. URL https://doi.org/10.1016/j.jcp.2017.02.032

[19] Moncorgé, A., Tchelepi, H., Jenny, P.: Sequential fully implicit formulation for compositional simulation using natural variables. Journal of Computational Physics **371**, 690 – 711 (2018). DOI 10.1016/j.jcp.2018.05.048. URL https://doi.org/10.1016/j.jcp.2018.05.048

[20] Møyner, O., Lie, K.A.: A multiscale restriction-smoothed basis method for compressible black-oil models. SPE Journal **21**(06), 2079–2096 (2016). DOI 10.2118/173265-PA. URL http://dx.doi.org/10.2118/173265-PA

[21] Møyner, O., Lie, K.A.: A multiscale restriction-smoothed basis method for high contrast porous media represented on unstructured grids. Journal of Computational Physics **304**, 46 – 71 (2016). DOI http://dx.doi.org/10.1016/j.jcp.2015.10.010. URL http://www.sciencedirect.com/science/article/pii/S0021999115006725

[22] Møyner, O., Tchelepi, H.: A mass-conservative sequential implicit multiscale method for isothermal equation of state compositional problems. SPE Journal (To appear) (2018)

[23] OPM Data repository: The Norne benchmark case. url: http://www.github.com/opm/opm-data (2015)

[24] Russell, T.: Stability analysis and switching criteria for adaptive implicit methods based on the cfl condition. In: SPE Symposium on Reservoir Simulation. Society of Petroleum Engineers (1989)

[25] SINTEF Digital: The MATLAB Reservoir Simulation Toolbox. www.sintef.no/MRST (2018)

[26] Thomas, G., Thurnau, D.: Reservoir simulation using an adaptive implicit method. SPE Journal **23**(05), 759–768 (1983)

[27] Trangenstein, J., Bell, J.: Mathematical structure of compositional reservoir simulation. SIAM Journal on Scientific and Statistical Computing **10**(5), 817–845 (1989). DOI 10.1137/0910049. URL https://doi.org/10.1137/0910049

[28] Trangenstein, J.A., Bell, J.B.: Mathematical structure of compositional reservoir simulation. SIAM journal on scientific and statistical computing **10**(5), 817–845 (1989)

[29] Trangenstein, J.A., Bell, J.B.: Mathematical structure of the black-oil model for petroleum reservoir simulation. SIAM Journal on Applied Mathematics **49**(3), 749–783 (1989)

[30] Voskov, D.: An extended natural variable formulation for compositional simulation based on tie-line parameterization. Transport in porous media **92**(3), 541–557 (2012)

[31] Wallis, J.R., Kendall, R., Little, T., et al.: Constrained residual acceleration of conjugate residual methods. In: SPE Reservoir Simulation Symposium. Society of Petroleum Engineers (1985)

[32] Watts, J.: A compositional formulation of the pressure and saturation equations. SPE Reservoir Engineering **1**(03), 243–252 (1986). DOI 10.2118/12244-pa. URL http://dx.doi.org/10.2118/12244-PA

[33] Young, L., Russell, T., et al.: Implementation of an adaptive implicit method. In: SPE Symposium on Reservoir Simulation. Society of Petroleum Engineers (1993)