Multiscale gradient computation for multiphase flow in porous media

Moraes, R.; Rodrigues, J. R.P.; Hajibeygi, H.; Jansen, J. D.

DOI
10.2118/182625-MS

Publication date
2017

Document Version
Final published version

Published in
Society of Petroleum Engineers - SPE Reservoir Simulation Conference 2017

Society of Petroleum Engineers

SPE-182625-MS

Multiscale Gradient Computation for Multiphase Flow in Porous Media

R. Moraes, Department of Geoscience and Engineering, TU Delft; J. R. P. Rodrigues, Petrobras Research and Development Center; H. Hajibeygi and J. D. Jansen, Department of Geoscience and Engineering, TU Delft

This paper was prepared for presentation at the SPE Reservoir Simulation Conference held in Montgomery, TX, USA, 20–22 February 2017.

Abstract

A multiscale gradient computation method for multiphase flow in heterogeneous porous media is developed. The method constructs multiscale primal and dual coarse grids, imposed on the given fine-scale computational grid. Local multiscale basis functions are computed on (dual-) coarse blocks, constructing an accurate map (prolongation operator) between coarse- and fine-scale systems. While the expensive operations involved in computing the gradients are performed at the coarse scale, sensitivities with respect to uncertain parameters (e.g., grid block permeabilities) are expressed in the fine scale via the partial derivatives of the prolongation operator. Hence, the method allows for updating of the geological model, rather than the dynamic model only, avoiding upscaling and the inevitable loss of information. The formulation and implementation are based on automatic differentiation (AD), allowing for convenient extensions to complex physics. An IMPES coupling strategy for flow and transport is followed, in the forward simulation. The flow equation is computed using a multiscale finite volume (MSFV) formulation and the transport equation is computed at the fine scale, after reconstruction of mass conservative velocity field. To assess the performance of the method, a synthetic multiphase flow test case is considered. The multiscale gradients are compared against those obtained from a fine-scale reference strategy. Apart from its computational efficiency, the benefits of the method include flexibility to accommodate variables expressed at different scales, specially in multiscale data assimilation and reservoir management studies.

Introduction

History matching is an important reservoir management tool that aims for the calibration of the reservoir simulation model via assimilation of the production data. Computer-assisted history matching (CAHM) is a topic of active research due to its several challenges (Oliver and Chen, 2011). Among these challenges, the computational complexity and the dissimilarity in the scales between the simulation model and the production data stand out. The construction of lower fidelity, less computationally intensive models is a common practice to alleviate the computational complexity. Upscaling (Durlofsky, 2005), streamline simulation (Datta-Gupta and King, 2007) and reduced-order modeling (Kaleta et al., 2011) are examples of methods to construct such models. However, lower fidelity models might not accurately represent the system outputs, due to excessive simplifications of the rock-fluid physics and heterogeneous geological properties.

Multiscale (MS) methods, on the other hand, are capable of delivering the desirable computational efficiency with acceptable (controllable) level of accuracy (compared with fine-scale reference solutions) (Hou and Wu, 1997; Jenny et al., 2003). MS methods have considerably evolved in the past years. State-of-the-art methods include the effects of compositional (Lee et al., 2008; Hajibeygi and Tchelepi, 2014; Kozlova et al., 2015), fractures (Hajibeygi et al., 2011; Efendiev et al., 2015; Tene et al., 2016; Shah et al., 2016) and geo-mechanical deformations (Castelletto et al., 2016). A review of the main developments in the field is provided by Lie et al. (2016). MS methods have been originally developed for efficient and accurate solutions of elliptic (incompressible) and parabolic (compressible) flow equations. Even with their extensions for the solution of hyperbolic transport equations (see e.g. Aarnes et al. (2007); Zhou et al. (2012)), fine-grid resolution is still necessary to accurately capture the sharp, local saturation fronts (Cusini et al., 2016). As such, the scales at which the flow and transport equations are finally solved might be quite different. If a different resolution for transport is taken, a reconstruction of conservative velocity field at the transport scale is required (Jenny et al., 2005).

Adjoint-based CAHM has been reported to be the most efficient and accurate data assimilation technique (Oliver et al., 2008). The utilization of adjoint methods for CAHM dates back to the 1960's and 70's (Jacquard and Jain, 1965; Chavent et al., 1975; Chen et al., 1974). Important developments in adjoint gradient computation -in the scope of this work- are the algebraic description of the model equations (Kraaijevanger et al., 2007; Rodrigues, 2006) and the minimum intrusion to the simulator code (Sarma et al., 2005). However, the necessity of having access to the simulator source code and the implementation complexity are consistently reported as drawbacks of the technique.

Recently, many developments and successful applications of ensemble-based history matching techniques have been reported (Aanonsen et al., 2009). Note that both commercial (Hiebert et al., 2011; DeBaun et al., 2005) and research-oriented (Krogstad et al., 2015; Contributors, 2016; Voskov et al., 2009) simulators rely on modern software developments with convenient techniques to compute partial derivatives. One important technique is automatic differentiation (AD) (Corliss, 2002), which reduces the programming effort to obtain the partial derivative information required by the adjoint formulation. However, due its computational overhead, the development of efficient AD libraries is a topic of active research (Younis and Aziz, 2007; Li and Zhang, 2014; Zhou et al., 2011).

Also, although not a limitation, adjoint methods have been traditionally employed in fully implicit simulations. In this scenario, the adjoint equation is the transpose of the forward system Jacobian (Li et al., 2003). In contrast to this fact, MS simulations most commonly require a sequential coupling of flow and transport equations. Consequently, the adjoint equations for such a solution strategy must be devised. Rodrigues (2006) shows how the adjoint equation can generically account for different coupling strategies by properly considering the lagged-in-time primary variable dependencies.

In order to address the computational efficiency and dissimilarity of scales in data assimilation, the present work focuses on the computation of MS adjoint gradients applied to CAHM studies. In this context, Fu et al. (2011) show that a MSFV adjoint formulation can be efficiently applied to inverse problems of single-phase flow in heterogeneous porous media. Furthermore, Moraes et al. (2016) proposes a generic framework for the computation of MS gradient information for single-phase flow models. Both aforementioned developments show that approximated gradients computed by MS strategies are suitable for history matching. Also, although without any MS simulations, Grimstad et al. (2003) and Aanonsen et al. (2008) show how important it is to properly treat the scale difference of the data by employing upscaling and downscaling techniques.

In this paper, the MS gradient calculation method for multiphase flow in porous media is developed. Consistent with the main development line of MS methods, the presented method is built on a sequential coupling strategy for flow and transport. Two sets of (dual and primal) coarse grids are constructed, in order to formulate a finite-volume MS method (MSFV). Because the transport equation is solved at the fine-scale, the MSFV stage also involves the reconstruction of conservative velocity field. A proof-of-concept

test case is provided to illustrate the performance of the method. For this test case, the multiscale gradients are compared against those obtained from a fine-scale reference strategy.

The remainder of this paper is organized as follows. First, the mathematical framework to analytically compute derivatives for the implicit pressure, explicit saturation (IMPES) solution method is developed. The framework is developed considering that flow is solved via a MS method and transport is solved on the fine-grid after the appropriate conservative velocity field reconstruction. Next, the newly developed method is validated against numerical differentiation. Also, the quality of the MS gradient is compared against results obtained with fine- scale adjoint. Finally, further opportunities on how a MS CAHM approach can improve the data assimilation exercise are discussed.

Forward Model Equations

Governing Equations and Multiscale Method

Two-phase, immiscible flow is described by mass conservation equation for each phase $\alpha \in \{o, w\}$

$$\frac{\partial}{\partial t}\left(\phi \rho_\alpha S_\alpha\right) + \nabla \cdot \left(\rho_\alpha \mathbf{u}_\alpha\right) = \rho_\alpha q_\alpha, \tag{1}$$

where ϕ is the porosity and S_α, ρ_α, u_α, and q_α are, respectively, saturation, density, and source for phase α. Neglecting gravity and capillary forces, the Darcy velocity for phase a can be expressed as

$$\mathbf{u}_\alpha = -\lambda_\alpha \mathbf{k} \cdot \nabla \mathbf{p}, \tag{2}$$

where λ_α is the mobility of phase α, i.e. the ratio between the phase relative permeability and viscosity. For incompressible fluid and porous-rock, each mass conservation equation can be divided by the respective (constant) density ρ_α and summed, which yields

$$-\nabla \cdot \left(\lambda_t \mathbf{k} \cdot \nabla p\right) = q_t, \tag{3}$$

where λ_t is the total mobility (the sum of all phase mobilities) and q_t is the sum of all phases' source terms. Finally, one transport equation can be written as

$$\phi \frac{\partial S_\alpha}{\partial t} + \nabla \cdot \left(f_\alpha \mathbf{u}_\alpha\right) = f_\alpha q_t. \tag{4}$$

The system is closed via the saturation constraint

$$\sum\nolimits_{\alpha=o,w} S_\alpha = 1. \tag{5}$$

Using a two-point flux finite volume discretization in space and an implicit-pressure explicit-saturation (IMPES) discretization in time (Aziz and Settari, 1979), the discretized form of Eq. (3) can be written as

$$\mathbf{A}^{n-1}\mathbf{p}^n = \mathbf{q}^n, \tag{6}$$

where p and q are pressure and source terms vectors, superscript n denotes the time-step, and A is the system matrix. Interfacial rock properties are computed by means of harmonic averages for the absolute permeabilities, whereas an upwind scheme is employed for interfacial fluid properties (i.e. fractional flows and mobilities). The dependency of the fluid mobilities on the saturation is treated lagged in time because we use an IMPES scheme. The discretization of Eq. (4) can be written as

$$\mathbf{V}\left(\mathbf{s}^n - \mathbf{s}^{n-1}\right) + \tilde{\mathbf{F}}^{n-1}\left(\mathbf{u}^n - \mathbf{q}^n\right) = 0, \tag{7}$$

where s and $\tilde{\mathbf{F}}$ are, respectively, the saturation vector and the fractional flow matrix, and $\mathbf{V} = \frac{\phi}{\Delta \mathbf{t}}\mathbf{I}$ (in which Δt is the time-step size and I is the identity matrix. Note that the non-linear dependency of the fractional flow on the saturation is evaluated at the previous time-step $n-1$. This allows S^n to be easily obtained from Eq. (7). The de-coupling of the equations allows Eq. (6) and Eq. (7) to be solved sequentially, with no dependency of the equation terms on s^{n-1}.

Multiscale (MS) methods, an efficient strategy to solve system of equations arising from the discretization of elliptic PDEs, can be employed to solve Eq. (6). The resulting MS system can be algebraically expressed as

$$(\mathbf{RAP})^{n-1}\breve{\mathbf{p}}^n = \mathbf{R}^{n-1}\mathbf{q}^n, \tag{8}$$

where R is the restriction operator, P the prolongation operator and $\breve{\mathbf{p}}$ the coarse pressure solution (Wang et al., 2014). The interpolated fine-scale pressure in obtained by means of the prolongation operator as

$$\mathbf{p}'^n = \mathbf{P}^{n-1}\breve{\mathbf{p}}^n. \tag{9}$$

Although the approximated pressure can provide a conservative velocity field at the coarse scale

$$\mathbf{u}' = -\lambda \cdot \nabla p', \tag{10}$$

the u' is not conservative at the fine-scale (Jenny et al., 2005; Hajibeygi and Jenny, 2011). Hence, an additional local problem

$$-\nabla \cdot (\lambda \cdot \nabla p''_k) = q'' \tag{11}$$

is solved on the primal coarse grid cell domain $\breve{\Omega}_k$ with the Neumann boundary conditions

$$\underbrace{(\lambda \cdot \nabla p''_k) \cdot \bar{\mathbf{n}}_k}_{\mathbf{u}''} = \underbrace{(\lambda \cdot \nabla p') \cdot \bar{\mathbf{n}}_k}_{\mathbf{u}'} \text{ at } \partial\breve{\Omega}_k. \tag{12}$$

Equations (11) and (12) can be written in discrete form as

$$\mathbf{A}''^{n-1}\mathbf{p}''^n = \mathbf{q}''^n - \mathbf{\Lambda}'^{n-1}\mathbf{p}'^n, \tag{13}$$

where A''^{n-1} results from assembling local discretizations of Eq. (11) into one global block diagonal matrix. Using the pressure calculated as solution of the problem defined by Eqs. (10) and (11) to calculate

$$\mathbf{u} = \begin{cases} -\lambda \cdot \nabla p''_k \text{ on } \breve{\Omega}_k, \\ -\lambda \cdot \nabla p' \text{ at } \partial\breve{\Omega}_k, \end{cases} \tag{14}$$

provides a conservative velocity field inside the $\breve{\Omega}_k$ domain. The velocity field computed via Eq. (14) is, therefore, suited to be used in the solution of Eq. (7). The discrete form of Eq. (14) can be expressed as

$$\mathbf{u}^n = \mathbf{\Lambda}'^{n-1}\mathbf{p}'^n + \mathbf{\Lambda}''^{n-1}\mathbf{p}''^n, \tag{15}$$

where $\Lambda' \neq 0$ only at the boundaries of the primal coarse cells, $\partial\breve{\Omega}_k$. Similarly, $\Lambda'' \neq 0$ only on the interior of the primal coarse cells and has a block diagonal structure.

Algebraic Description of the Forward Model Equations

The discrete forward simulation equations for the IMPES coupling can be expressed via a purely algebraic formulation as

$$\begin{cases} \mathbf{g}_p^n\left(\mathbf{p}^n, \mathbf{s}^{n-1}, \boldsymbol{\theta}\right) = \mathbf{0}, \\ \mathbf{g}_s^n\left(\mathbf{p}^n, \mathbf{s}^{n-1}, \mathbf{s}^n, \boldsymbol{\theta}\right) = \mathbf{0}, \end{cases} \tag{16}$$

where $g_p{}^n$ and $g_S{}^n$ are vector-valued the equations describing flow (pressure) and transport (saturation) at time-step n and Θ is the vector of model parameters. Once again, note that the dependency of $g_p{}^n$ on s is lagged one step in time. The equations that determine the initial conditions are assumed to be

$$\begin{cases} \mathbf{g}_p^0\left(\mathbf{p}^0, \boldsymbol{\theta}\right) = \mathbf{0}, \\ \mathbf{g}_s^0\left(\mathbf{p}^0, \mathbf{s}^0, \boldsymbol{\theta}\right) = \mathbf{0}. \end{cases} . \tag{17}$$

As discussed in Moraes et al. (2016), the MS solution for the flow equation for time-step n can be algebraically expressed as

$$\mathbf{g}_p^n = \begin{bmatrix} \breve{\mathbf{g}}^n \\ \mathbf{g}'^n \end{bmatrix}, \tag{18}$$

where

$$\breve{\mathbf{g}}^n = (\mathbf{RAP})^{n-1}\breve{\mathbf{p}}^n - \mathbf{R}^{n-1}\mathbf{q}^n = \breve{\mathbf{A}}^{n-1}\breve{\mathbf{p}}^n - \breve{\mathbf{q}}^n = \breve{\mathbf{0}}, \tag{19}$$

and

$$\mathbf{g}'^n = \mathbf{p}'^n - \mathbf{P}^n\breve{\mathbf{p}}^n = \mathbf{0}. \tag{20}$$

Note that $A^{n-1} = A^{n-1}(s^{n-1})$ and $P^n = P^n(s^{n-1})$. Furthermore, although prolongation depends on time, it is only rebuilt infrequently, depending on how much the mobilities change in the corresponding coarse grid block (Jenny et al., 2005).

From Eq. (15) one can write

$$\mathbf{g}_u\left(\mathbf{u}^n, \mathbf{p}'^n, \mathbf{p}''^n, \mathbf{s}^{n-1}, \theta\right) = \mathbf{u}^n - \boldsymbol{\Lambda}'^{n-1}\mathbf{p}'^n - \boldsymbol{\Lambda}''^{\mathbf{n-1}}\mathbf{p}''^{\mathbf{n}} = \mathbf{0}. \tag{21}$$

Note that $\Lambda'^{n-1} = \Lambda'^{n-1}(s^{n-1})$ and $\Lambda''^{n-1} = \Lambda'^{n-1}(s^{n-1})$. Also, from Eq. (13), follows that

$$\mathbf{g}''^n\left(\mathbf{p}^n, \mathbf{p}'^n, \mathbf{s}^{n-1}, \theta\right) = \mathbf{A}''^n\mathbf{p}''^n - \mathbf{q}''^n - \boldsymbol{\Lambda}'^{n-1}\mathbf{p}'^n = \mathbf{0}. \tag{22}$$

The discrete transport equation (Eq. (7)) can be algebraically expressed as

$$\mathbf{g}_s^n\left(\mathbf{p}^n, \mathbf{s}^{n-1}, \mathbf{s}^n\right) = \mathbf{s}^n - \mathbf{s}^{n-1} - \mathbf{F}^{n-1}\mathbf{u}^n - \mathbf{F}^{n-1}\mathbf{q}^n = \mathbf{0} \tag{23}$$

where $\mathbf{F}^{n-1}\left(\mathbf{s}^{n-1}\right) = \frac{\tilde{\mathbf{F}}}{\mathbf{V}}$ is an upwind fractional flow operator.

A vector-valued function for the forward simulation Eqs. (18), (20), (22), (21), and (23), can now be expressed as

$$\mathbf{g}^n\left(\mathbf{x}^n, \mathbf{x}^{n-1}, \theta\right) = \begin{bmatrix} \breve{\mathbf{g}} \\ \mathbf{g}' \\ \mathbf{g}'' \\ \mathbf{g}_u \\ \mathbf{g}_s \end{bmatrix}^n = \mathbf{0}, \tag{24}$$

or

$$\mathbf{g}^n\left(\mathbf{x}^n,\mathbf{x}^{n-1},\theta\right)=\begin{bmatrix}\breve{\mathbf{A}}^{n-1} & & & & \\ -\mathbf{P}^{n-1} & \mathbf{I} & & & \\ & -\boldsymbol{\Lambda}'^{n-1} & \mathbf{A}''^{n-1} & & \\ & -\boldsymbol{\Lambda}''^{n-1} & -\boldsymbol{\Lambda}''^{n-1} & \mathbf{I} & \\ & & & -\mathbf{F}^{n-1} & \mathbf{I}\end{bmatrix}\begin{bmatrix}\breve{\mathbf{p}}\\ \mathbf{p}'\\ \mathbf{p}''\\ \mathbf{u}\\ \mathbf{s}\end{bmatrix}^n-\begin{bmatrix}\breve{\mathbf{q}}^n\\ \mathbf{0}\\ \mathbf{q}''\\ \mathbf{0}\\ \mathbf{s}^{n-1}+\mathbf{F}^{n-1}\mathbf{q}^n\end{bmatrix}=\mathbf{0}, \tag{25}$$

where the parameter-dependent state vector can be expressed as

$$\mathbf{x}^n=\mathbf{x}^n(\theta)=\begin{bmatrix}\breve{\mathbf{p}}\\ \mathbf{p}'\\ \mathbf{p}''\\ \mathbf{u}\\ \mathbf{s}\end{bmatrix}^n. \tag{26}$$

The functions defining the set of observable outputs for a time-step will be assumed to be functions of both p^n and s^{n-1}

$$\mathbf{y}^n=\mathbf{h}^n\left(\mathbf{p}^n,\mathbf{s}^{n-1},\theta\right), \tag{27}$$

which from Eq. (26) can be re-written as

$$\mathbf{y}^n=\mathbf{h}^n\left(\mathbf{x}^n,\mathbf{x}^{n-1},\theta\right). \tag{28}$$

Computation of Simulator Output Derivatives via Adjoint Method

The derivation of a generalized framework to compute the gradient information of the simulator output starts from the definition of the sensitivity matrix pre and post multiplied by arbitrary matrices V and W (as described in Moraes et al. (2016)):

$$\mathbf{WGV}=-\mathbf{W}\frac{\partial\mathbf{h}}{\partial\mathbf{x}}\left(\frac{\partial\mathbf{g}}{\partial\mathbf{x}}\right)^{-1}\frac{\partial\mathbf{g}}{\partial\boldsymbol{\theta}}\mathbf{V}+\mathbf{W}\frac{\partial\mathbf{h}}{\partial\boldsymbol{\theta}}\mathbf{V}. \tag{29}$$

The key aspect that defines the computational performance of the gradient information computation is the order of the operations involving $\left(\frac{\partial\mathbf{g}}{\partial\mathbf{x}}\right)^{-1}$. Based on that, both the Direct and Adjoint methods can be defined. Given that for most CAHM approaches the number of outputs is smaller than the number of parameters, only the adjoint method will be addressed in the scope of this work. However, as shown in Moraes et al. (2016), the framework here described can provide the Direct method by following a similar derivation.

A super-vector notation is used to capture the evolution in time. All instances of g^n as defined in Eq. (24) for all time-steps, can be collated in a function $\mathbf{g}:\mathbb{R}^{N_F}\times\mathbb{R}^{N_\Theta}\rightarrow\mathbb{R}^{N_F}$, such that the system of non-linear equations is represented as

$$\mathbf{g}(\mathbf{x},\boldsymbol{\theta})=\mathbf{0}. \tag{30}$$

Similarly, a function $\mathbf{h}:\mathbb{R}^{N_F}\times\mathbb{R}^{N_\Theta}\rightarrow\mathbb{R}^{N_Y}$ that collates all outputs y^n as defined in Eq. (28), for all time-steps, can be written as

$$\mathbf{y}=\mathbf{h}(\mathbf{x}(\boldsymbol{\theta}),\boldsymbol{\theta}). \tag{31}$$

If W is factored out in Eq. (29), it can be rewritten as

$$\mathbf{WG}=\mathbf{Z}\frac{\partial\mathbf{g}}{\partial\theta}+\mathbf{W}\frac{\partial\mathbf{h}}{\partial\theta}, \tag{32}$$

where

$$\mathbf{Z} = -\mathbf{W}\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\left(\frac{\partial \mathbf{g}}{\partial \mathbf{x}}\right)^{-1}. \tag{33}$$

The linear system described in Eq. (33) can be re-written in a block-wise fashion for each time-step *n:*

$$\left[\begin{array}{c|c|c|c} \mathbf{Z}^0 & \mathbf{Z}^1 & \cdots & \mathbf{Z}^n \end{array}\right] \left[\begin{array}{c|c|c|c} \frac{\partial \mathbf{g}^0}{\partial \mathbf{x}^0} & & & \\ \hline \frac{\partial \mathbf{g}^1}{\partial \mathbf{x}^0} & \frac{\partial \mathbf{g}^1}{\partial \mathbf{x}^1} & & \\ \hline & \ddots & \ddots & \\ \hline & & \frac{\partial \mathbf{g}^n}{\partial \mathbf{x}^{n-1}} & \frac{\partial \mathbf{g}^n}{\partial \mathbf{x}^n} \end{array}\right] = \left[\begin{array}{c|c|c|c} -\mathbf{W}^0\frac{\partial \mathbf{h}^0}{\partial \mathbf{x}^0} & -\mathbf{W}^1\frac{\partial \mathbf{h}^1}{\partial \mathbf{x}^1} & \cdots & -\mathbf{W}^n\frac{\partial \mathbf{h}^n}{\partial \mathbf{x}^n} \end{array}\right]. \tag{34}$$

By transposing Eq. (34), one obtains

$$\left[\begin{array}{c|c|c|c} \left(\frac{\partial \mathbf{g}^0}{\partial \mathbf{x}^0}\right)^T & \left(\frac{\partial \mathbf{g}^1}{\partial \mathbf{x}^0}\right)^T & & \\ \hline & \left(\frac{\partial \mathbf{g}^1}{\partial \mathbf{x}^1}\right)^T & \ddots & \\ \hline & & \ddots & \left(\frac{\partial \mathbf{g}^N}{\partial \mathbf{x}^{N-1}}\right)^T \\ \hline & & & \left(\frac{\partial \mathbf{g}^N}{\partial \mathbf{x}^N}\right)^T \end{array}\right] \left[\begin{array}{c} (\mathbf{Z}^0)^T \\ \hline (\mathbf{Z}^1)^T \\ \hline \vdots \\ \hline (\mathbf{Z}^N)^T \end{array}\right] = \left[\begin{array}{c} \left(-\mathbf{W}^0\frac{\partial \mathbf{h}^0}{\partial \mathbf{x}^0}\right)^T \\ \hline \left(-\mathbf{W}^1\frac{\partial \mathbf{h}^1}{\partial \mathbf{x}^1}\right)^T \\ \hline \vdots \\ \hline \left(-\mathbf{W}^N\frac{\partial \mathbf{h}^N}{\partial \mathbf{x}^N}\right)^T \end{array}\right]. \tag{35}$$

In order to better understand the partial derivative matrices that need to be computed from the forward simulation equations, the linear system of equations that must be solved for a given time-step *n* can be looked in more detail

$$\left(\frac{\partial \mathbf{g}^n}{\partial \mathbf{x}^n}\right)^T (\mathbf{Z}^n)^T = \left(-\mathbf{W}^n\frac{\partial \mathbf{h}^n}{\partial \mathbf{x}^n}\right)^T - \left(\frac{\partial \mathbf{g}^{n+1}}{\partial \mathbf{x}^n}\right)^T (\mathbf{Z}^{n+1})^T. \tag{36}$$

From Eq. (25) one obtains

$$\frac{\partial \mathbf{g}^n}{\partial \mathbf{x}^n} = \begin{bmatrix} \breve{\mathbf{A}}^{n-1} & & & & \\ -\mathbf{P}^{n-1} & \mathbf{I} & & & \\ & -\mathbf{\Lambda}'^{n-1} & \mathbf{A}''^{n-1} & & \\ & -\mathbf{\Lambda}'^{n-1} & -\mathbf{\Lambda}''^{n-1} & \mathbf{I} & \\ & & & -\mathbf{F}^{n-1} & \mathbf{I} \end{bmatrix}. \tag{37}$$

Transposing and substituting Eq. (37) into Eq. (36) then gives:

$$\begin{bmatrix} \left(\breve{\mathbf{A}}^{n-1}\right)^T & \left(-\mathbf{P}^{n-1}\right)^T & & & \\ \mathbf{I} & \left(-\mathbf{\Lambda}'^{n-1}\right)^T & \left(-\mathbf{\Lambda}'^{n-1}\right)^T & & \\ & \left(\mathbf{A}''^{n-1}\right)^T & \left(-\mathbf{\Lambda}''^{n-1}\right)^T & & \\ & & \mathbf{I} & \left(-\mathbf{F}^{n-1}\right)^T & \\ & & & \mathbf{I} & \end{bmatrix} \begin{bmatrix} \left(\breve{\mathbf{Z}}^{n}\right)^T \\ \left(\mathbf{Z}'^{n}\right)^T \\ \left(\mathbf{Z}''^{n}\right)^T \\ \left(\mathbf{Z}_u{}^{n}\right)^T \\ \left(\mathbf{Z}_s{}^{n}\right)^T \end{bmatrix} = \begin{bmatrix} \left(\mathbf{W}^n \frac{\partial \mathbf{h}^n}{\partial \breve{\mathbf{p}}^n}\right)^T \\ \left(\mathbf{W}^n \frac{\partial \mathbf{h}^n}{\partial \mathbf{p}'^n}\right)^T \\ \left(\mathbf{W}^n \frac{\partial \mathbf{h}^n}{\partial \mathbf{p}''^n}\right)^T \\ \left(\mathbf{W}^n \frac{\partial \mathbf{h}^n}{\partial \mathbf{u}^n}\right)^T \\ \left(\mathbf{W}^n \frac{\partial \mathbf{h}^n}{\partial \mathbf{s}^n}\right)^T \end{bmatrix} - \left(\frac{\partial \mathbf{g}^{n+1}}{\partial \mathbf{x}^n}\right)^T \begin{bmatrix} \left(\breve{\mathbf{Z}}^{n+1}\right)^T \\ \left(\mathbf{Z}'^{n+1}\right)^T \\ \left(\mathbf{Z}''^{n+1}\right)^T \\ \left(\mathbf{Z}_u{}^{n+1}\right)^T \\ \left(\mathbf{Z}_s{}^{n+1}\right)^T \end{bmatrix}. \tag{38}$$

The structure of Eq. (38) shows that it can be solved via a back substitution, i.e. through solving the contributions from the saturation equation all the way back to the pressure equation. Hence, the solution of the gradient information is not only backward in time, but it is also backward in terms of the order in which the equations are solved in the forward simulation.

Computation of partial derivative matrices and Automatic Differentiation

One key aspect of analytical gradient computation is the need for some additional partial derivative matrices that are not required by standard forward simulation procedures. The partial derivatives of the model equations with respect to the primary variables lagged in time, $\frac{\partial \mathbf{g}^{n+1}}{\partial \mathbf{x}^n}$, can be fully determined from dependencies of Eqs. (18), (20), (22), (21) and (23) on x^{n-1}, and more specifically on s^{n-1} in the scope of the present work. Also, the dependencies of the model equations on the parameters must be taken into account in order to fully determine $\frac{\partial \mathbf{g}}{\partial \theta}$. Ultimately, a reservoir simulation code is an algorithm comprised of many sub-algorithms, each of them dependent on a given pre-determined set of parameters. The output of each (sub-algorithm) will be consumed by another (sub-) algorithm and, finally, by the (sub-) algorithms responsible for the computation of model equations. A convenient way for analytically computing the partial derivatives of the model equations w.r.t. their dependent parameters/variables is Automatic Differentiation (AD). In brief, AD is a computational technique that computes the partial derivatives of a given function w.r.t. its arguments at the same time the function value is evaluated. More information about AD can be found in the literature (Corliss, 2002; Younis and Aziz, 2007; Zhou et al., 2011; Li and Zhang, 2014). The same discussion holds when it comes to the computation of the partial derivatives of the model output, namely $\frac{\partial \mathbf{h}}{\partial \mathbf{x}}$ and $\frac{\partial \mathbf{h}}{\partial \theta}$.

Not only the computation of the partial derivatives requires proper treatment, in order to efficiently evaluate the expressions determining the products, special attention must be also payed to the order of the operations. This is important to determine the adjoint method, and it is also important in some other steps of the computation when perform in operations involving the prolongation operator (in those cases where it depends on the parameters). These operations are discussed in the next section.

Partial derivatives with respect to the parameters. Utilizing the definition of g (Eq. (24)) to expand Eq. (32), one can define the update of WG for the time-step *n* as

$$\mathbf{WG}^n = \left(\breve{\mathbf{Z}}\frac{\partial \breve{\mathbf{g}}}{\partial \theta} + \mathbf{Z}'\frac{\partial \mathbf{g}'}{\partial \theta} + \mathbf{Z}''\frac{\partial \mathbf{g}''}{\partial \theta} + \mathbf{Z}_u\frac{\partial \mathbf{g}_u}{\partial \theta} + \mathbf{Z}_s\frac{\partial \mathbf{g}_s}{\partial \theta} + \mathbf{W}\frac{\partial \mathbf{h}}{\partial \theta}\right)^n, \tag{39}$$

where the partial derivatives of g″, g_u, and g_s with respect to Θ can be obtained from, respectively, Eqs. (22), (21), and (23) as

$$\frac{\partial \mathbf{g}''^n_u}{\partial \theta} = \frac{\partial \mathbf{A}''^{n-1}}{\partial \theta}\mathbf{p}''^n - \frac{\partial \mathbf{q}''^n}{\partial \theta} - \frac{\partial \mathbf{\Lambda}'^{n-1}}{\partial \theta}\mathbf{p}'^n, \tag{40}$$

$$\frac{\partial \mathbf{g}^n_u}{\partial \theta} = -\frac{\partial \mathbf{\Lambda}'^{n-1}}{\partial \theta}\mathbf{p}'^n - \frac{\partial \mathbf{\Lambda}''^{n-1}}{\partial \theta}\mathbf{p}''^n, \tag{41}$$

and

$$\frac{\partial \mathbf{g}^n_s}{\partial \theta} = -\frac{\partial \mathbf{F}^{n-1}}{\partial \theta}\mathbf{u}^n - \frac{\partial \mathbf{F}^{n-1}}{\partial \theta}\mathbf{q}^n - \mathbf{F}^{n-1}\frac{\partial \mathbf{q}^n}{\partial \theta}. \tag{42}$$

The $\frac{\partial \mathbf{A}''^{n-1}}{\partial \theta}$, $\frac{\partial \mathbf{\Lambda}'^{n-1}}{\partial \theta}$, and $\frac{\partial \mathbf{\Lambda}''^{n-1}}{\partial \theta}$ partial derivatives result in third-order tensors. The interpreation of the operara- tions involving such terms can be found in Moraes et al. (2016). The first two terms of Eq. (39) involve operations with the partial derivative of the prolongation operator with respect to the parameter vector. This is discussed next.

Partial derivatives of the prolongation operator with respect to the parameters

The operations with the prolongation operator appear in the first two terms of Eq. (39). From the partial derivatives of $\breve{\mathbf{g}}$ and g' with respect to Θ, one finds (Moraes et al., 2016)

$$\left(\breve{\mathbf{Z}}\frac{\partial \breve{\mathbf{g}}}{\partial \theta} + \mathbf{Z}'\frac{\partial \mathbf{g}'}{\partial \theta}\right)^n = \left[\breve{\mathbf{Z}}\left(\left[\frac{\partial \mathbf{R}}{\partial \theta}(\mathbf{AP}) + \mathbf{R}\frac{\partial \mathbf{A}}{\partial \theta}\mathbf{P} + (\mathbf{RA})\frac{\partial \mathbf{P}}{\partial \theta}\right]\breve{\mathbf{x}} - \frac{\partial \mathbf{R}}{\partial \theta}\mathbf{q} - \mathbf{R}\frac{\partial \mathbf{q}}{\partial \theta}\right) - \mathbf{Z}'\left(\frac{\partial \mathbf{P}}{\partial \theta}\breve{\mathbf{x}}\right)\right]^n. \tag{43}$$

It is described in Moraes et al. (2016) (Algorihm 4 in that paper) how to efficiently compute $\mathrm{m}^T(\partial \mathbf{P}/\partial\theta\breve{\mathbf{x}})$, in an adjoint-like fashion, at computation cost proportionally to the number of dual-coarse grid cells and independent of the number of parameters. Since the present work deals with time-stepping problems, this computation must be performed at all time-steps. But, because basis functions are only infrequently re-constructed, an efficient implementation can take advantage of this fact. For instance, a direct solver can be utilized for the computation of the prolongation partial derivatives, such that the system matrix is only factorized once and solved multiple times for different right-hand sides.

The conservative velocity field reconstruction step in the backward simulation. Along with the inexpensive (as shown by Moraes et al. (2016)) solution of a linear system of equations arising from $\breve{\mathbf{g}}$, the model equation g″ also requires the solution of a linear system. This can be noted by the non-identity entry in the main diagonal of Eq. (38), more specifically, the third equation in this system of equations. By looking at this specific equation, for a given time-step *n* and a given set of outputs *i* of the same time-step, one can write

$$\left(\mathbf{Z}''_{i,\cdot}{}^n\right)^T = \left(\mathbf{A}''^{n-1}\right)^{-T}\left(\left(\mathbf{\Lambda}''^{n-1}\right)^T\left(\mathbf{Z}_{u_{i,\cdot}}{}^{n-1}\right)^T\right). \tag{44}$$

Because of the block structure of the operands of Eq. (44), g″ can be determined in similar fashion via the solution of local systems corresponding to the primal-coarse grid cell domains. The solution of this equation can be also devised via the solution of local-system domains.

Numerical Experiments

The following numerical experiments are presented to first validate and then assess the accuracy of the gradient information computed by the method presented in this work. For this purpose, a misfit objective function with no regularization term

$$O(\theta) = \frac{1}{2}(\mathbf{h}(\mathbf{x},\theta) - \mathbf{d}_{obs})^T \mathbf{C}_D^{-1}(\mathbf{h}(\mathbf{x},\theta) - \mathbf{d}_{obs}), \tag{45}$$

with a gradient

$$\nabla_\theta O = \mathbf{G}^T \mathbf{C}_D^{-1}(\mathbf{h}(\mathbf{x},\theta) - \mathbf{d}_{obs}), \tag{46}$$

is considered (Oliver et al., 2008). Here, d_{obs} are the measured data and C_D is the measurement covariance matrix. In the experiments, the history matching parameters are cell-centered permeabilities and the observed quantity is the oil rate at the production wells from all time-steps of the reference simulation. The injection/production wells are controlled by maximum/minimum bottom-hole pressures (BHP).

Validation

The MS-gradient method is validated against the numerical differentiation method using a two-sided Taylor approximation:

$$\begin{aligned}\nabla_\theta O_i = &\frac{1}{2\delta\theta_i}\left(O(\theta_i,\ldots,\theta_{i-1},\theta_i+\delta\theta_i,\theta_{i+i},\ldots,\theta_{N_\theta}) - \right.\\ &\left. O(\theta_i,\ldots,\theta_{i-1},\theta_i-\delta\theta_i,\theta_{i+i},\ldots,\theta_{N_\theta})\right),\end{aligned} \tag{47}$$

where δ is a multiplicative parameter perturbation. The relative error can be defined as

$$\varepsilon = \frac{\|\nabla_\theta O_{FD} - \nabla_\theta O_{AN}\|_2}{\|\nabla_\theta O_{AN}\|_2}, \tag{48}$$

where $\nabla_\Theta O_{FD}$ is obtained by performing the appropriate number of multiscale reservoir simulations required to evaluate Eq. (47). Also, $\nabla_\Theta O_{AN}$ is obtained by employing the adjoint MS gradient computation (to evaluate Eq. (46)). Note that the presented framework can be used by setting

$$\mathbf{W}^T = \mathbf{C}_D^{-1}(\mathbf{h}(\mathbf{x},\theta) - \mathbf{d}_{obs}), \tag{49}$$

so that the gradient of O can be written as $\nabla_\Theta O = (\mathrm{WG})^T$. For all cases, for simplicity, it is assumed that $C_D = I$.

In order to validate the proposed derivative calculation method, as well as the implementation, it is investigated the linear decrease of the error ε by decreasing the perturbation value δ (Heath, 2002) from 10^{-1} to 10^{-4} (the range within which discretization errors dominate).

The first case is a one-dimensional, homogeneous medium with 45 grid blocks. Fine grid block dimensions are $\Delta_x = \Delta_y = \Delta_z = 1.0$ m. A primal coarse grid of just 3 grid blocks is employed (coarsening ratio of 15). Injection and production wells are located at the first and last vertices of the primal grid blocks, respectively. Fig. 1 illustrates the setup for this experiment. Reservoir porosity is constant and equal to 0-2. The oil/water mobility ration is equal to 1.0, so that the total mobility does not change during the simulation. The observed oil-rate is generated from a reference, randomly distributed permeability field. The simulation time is 30 days and the time-step size is 1 day. The so-called Peaceman well model (Peaceman, 1978), with well diameter equal to 0^ 114 m. In the second case the reference permeability field is a homogeneous field and the gradients are computed based on the randomly distributed permeability field. This allows to assess the proper capture of fine- scale heterogeneities by the prolongation operator partial derivative with respect to the grid-block permeabilities.

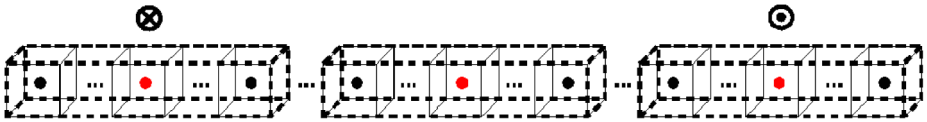

Figure 1—Fine and coarse grids and wells setup for 1D numerical experiments. The solid thin lines represent the fine grid-blocks. The bold dashed lines represent the primal-coarse grid blocks. Vertex cells are identified via red circles. The crossed circle represents the injection well and the dotted circle the production well.

The expected behavior of linearly decreasing error values as the perturbation size decreases is observed in the experiments, as shown in Fig. 2.

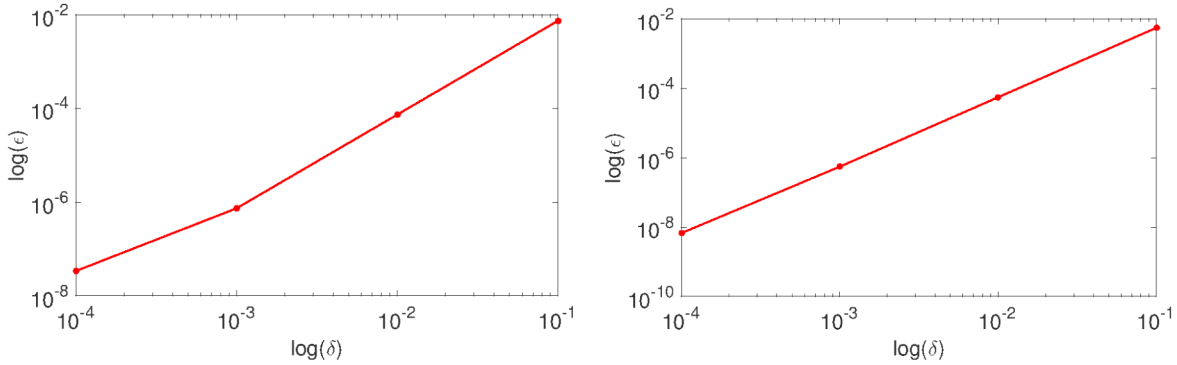

Figure 2—Validation of MS gradient computation method via comparison with numerical differentiation for the one-dimensional, (a) homogeneous and (b) heterogenous validation cases.

Gradient Accuracy

The metric used to evaluate the quality of the MS gradient is the angle between fine-scale and MS normalized gradients

$$\alpha = \cos^{-1}\left(\nabla_\theta^T \hat{O}_{FS}\, \nabla_\theta \hat{O}_{MS}\right), \tag{50}$$

where

$$\nabla_\theta \hat{O}_{FS} = \frac{\nabla_\theta O_{FS}}{\|\nabla_\theta O_{FS}\|_2}, \tag{51}$$

and

$$\nabla_\theta \hat{O}_{MS} = \frac{\nabla_\theta O_{MS}}{\|\nabla_\theta O_{MS}\|_2}. \tag{52}$$

Here, $\nabla_\Theta O_{FS}$ is the gradient computed via a fine-scale IMPES adjoint and $\nabla_\Theta O_{MS}$ denotes the MS adjoint gradient computed via the method developed in the present work. As a minimum requirement, acceptable MS gradients are obtained if α is much smaller than 90° (Fonseca et al., 2015).

Both homogeneous and heterogeneous cases result in $\alpha = 0^o$, indicating that the fine-scale and MS gradients are perfectly aligned in this case. This is due to the fact that, in 1D, no approximations (due to localization) are made in the MS solution, and thus, in the MS gradient computation. As explained in Moraes et al. (2016), the same behavior should not be expected in higher dimensional problems, where basis functions localization assumptions impact the quality of the gradient.

Summary and Concluding Remarks

A framework to efficiently compute gradient information required by computer-assisted history matching studies via an adjoint multiscale formulation is presented. No assumptions regarding the type of HM parameters or observed data are made during the derivation of the method. Following an IMPES coupling strategy, it is shown how to derive the adjoint formulation of sequentially coupled simulation strategies. In the formulation of the forward model equations, the flow equation is solved by a MS strategy, while the transport equation is solved in the fine scale. A conservative velocity field is reconstructed from the approximate MS pressure field via the solution of Neumann local problems based on the mass-conservative fluxes at the coarse-grid block boundaries. The equations associated with the aforementioned steps are incorporated in the adjoint formulation by algebraically keeping them in the set of forward model equations. Only relatively small linear systems of equations must be solved, given that they arise from either the independent local basis function equations, or the (also independent) local computations of the conservative velocity field. This fact grants the method efficiency for high-performance computing environments (Manea et al., 2015). The partial derivative matrices required by the adjoint formulation are computed via automatic differentiation. Preliminary 1D validation experiments show that all the complexities involved in the computation of MS gradients for multiphase flow simulations are appropriately captured by the formulation. The expected accuracy of the method is demonstrated when compared to fine-scale gradient. Further numerical experiments are required to verify the accuracy and efficiency of the formulation in higher dimensions.

Acknowledgments

Rafael Moraes PhD project is sponsored by Petrobras S.A. The authors would like to thank the Delft Advanced Reservoir Simulation (DARSim) research team for the insightful discussions during the development of this work.

References

Aanonsen, S. I. et al. (2008). Efficient history matching using a multiscale technique. *SPE Reservoir Evaluation & Engineering*, **11**(01):154–164.

Aanonsen, S. I., Nævdal, G., Oliver, D. S., Reynolds, A. C., and Valles, B. (2009). The ensemble Kalman filter in reservoir engineering-a review. *SPEJ*, **14**(03):393–412.

Aarnes, J. E., Hauge, V. L., and Efendiev, Y. (2007). Coarsening of three-dimensional structured and unstructured grids for subsurface flow. *Advances in Water Resources*, **30**(11):2177–2193.

Aziz, K. and Settari, A. (1979). *Petroleum reservoir simulation*. Chapman & Hall.

Castelletto, N., Hajibeygi, H., and Tchelepi, H. A. (2016). Multiscale finite-element method for linear elastic geomechanics. *Journal of Computational Physics, under review*.

Chavent, G., Dupuy, M., and Lemmonier, P. (1975). History matching by use of optimal theory. *SPEJ*, **15**(01):74–86.

Chen, W. H., Gavalas, G. R., Seinfeld, J. H., and Wasserman, M. L. (1974). A new algorithm for automatic history matching. *SPEJ*, **14**(6):593–608.

Contributors, O. (2016). The open porous media initiative. http://http://opm-project.org/.

Corliss, G. (2002). Automatic differentiation of algorithms: from simulation to optimization, volume **1**. Springer Science & Business Media.

Cusini, M., van Kruijsdijk, C., and Hajibeygi, H. (2016). Algebraic dynamic multilevel (adm) method for fully implicit simulations of multiphase flow in porous media. *Journal of Computational Physics*, **314**:60–79.

Datta-Gupta, A. and King, M. (2007). *Streamline simulation: theory and practice*. SPE textbook series. *Society of Petroleum Engineers*.

DeBaun, D., Byer, T., Childs, P., Chen, J., Saaf, F., Wells, M., Liu, J., Cao, H., Pianelo, L., Tilakraj, V., et al. (2005). An extensible architecture for next generation scalable parallel reservoir simulation. In SPE Reservoir Simulation Symposium, The Woodlands, Texas, USA, 31 January-2 Feburary. Society of Petroleum Engineers.

Durlofsky, L. J. (2005). Upscaling and gridding of fine scale geological models for flow simulation. In 8th International Forum on Reservoir Simulation Iles Borromees, Stresa, Italy, volume **2024**.

Efendiev, Y., Lee, S., Li, G., Yao, J., and Zhang, N. (2015). Hierarchical multiscale modeling for flows in fractured media using generalized multiscale finite element method. *GEM-International Journal on Geomathematics*, **6**:141–162.

Fonseca, R. M., Kahrobaei, S. S., Van Gastel, L. J. T., Leeuwenburgh, O., and Jansen, J. D. (2015). Quantification of the impact of ensemble size on the quality of an ensemble gradient using principles of hypothesis testing. In SPE Reservoir Simulation Symposium, Houston, Texas, USA, 23-25 February. Society of Petroleum Engineers.

Fu, J., Caers, J., and Tchelepi, H. A. (2011). A multiscale method for subsurface inverse modeling: Single-phase transient flow. *Advances in Water Resources*, **34**(8):967–979.

Grimstad, A.-A., Mannseth, T., Nævdal, G., and Urkedal, H. (2003). Adaptive multiscale permeability estimation. *Computational Geosciences*, **7**(1):1–25.

Hajibeygi, H. and Jenny, P. (2011). Adaptive iterative multiscale finite volume method. *Journal of Computational Physics*, **230**(3):628–643.

Hajibeygi, H., Karvounis, D., and Jenny, P. (2011). A hierarchical fracture model for the iterative multiscale finite volume method. *Journal of Computational Physics*, **230**(24):8729–8743.

Hajibeygi, H. and Tchelepi, H. A. (2014). Compositional multiscale finite-volume formulation. *SPEJ*, **19**(02):316–326.

Heath, M. T. (2002). *Scientific computing: an introductory survey*. The McGraw-Hill Companies Inc.: New York, NY, USA.

Hiebert, A. D., Khoshkbarchi, M., Sammon, P. H., Alves, I. N., Rodrigues, J., Belien, A. J., Howell, B., Saaf, F. E., Valvatne, P., et al. (2011). An advanced framework for simulating connected reservoirs, wells and production facilities. In SPE Reservoir Simulation Symposium, The Woodlands, Texas, USA, 21-23 February. Society of Petroleum Engineers.

Hou, T. Y. and Wu, X.-H. (1997). A multiscale finite element method for elliptic problems in composite materials and porous media. *Journal of computational physics*, **134**(1):169–189.

Jacquard, P. and Jain, C. (1965). Permeability distribution from field pressure data. *SPEJ*, **5**(04):281–294.

Jenny, P., Lee, S., and Tchelepi, H. (2005). Adaptive multiscale finite-volume method for multiphase flow and transport in porous media. *Multiscale Modeling & Simulation*, **3**(1):50–64.

Jenny, P., Lee, S. H., and Tchelepi, H. A. (2003). Multi-scale finite-volume method for elliptic problems in subsurface flow simulation. *Journal of Computational Physics*, **187**(1):47–67.

Kaleta, M. P., Hanea, R. G., Heemink, A. W., and Jansen, J.-D. (2011). Model-reduced gradient-based history matching. *Computational Geosciences*, **15**(1):135–153.

Kozlova, A., Li, Z., Natvig, J. R., Watanabe, S., Zhou, Y., Bratvedt, K., and Lee, S. (2015). A real-field multiscale black-oil reservoir simulator. In SPE Reservoir Simulation Symposium, Houston, Texas, USA, 23-25 February. Society of Petroleum Engineers.

Kraaijevanger, J. F. B. M., Egberts, P. J. P., Valstar, J. R., and Buurman, H. W. (2007). Optimal waterflood design using the adjoint method. In SPE Reservoir Simulation Symposium, Houston, Texas, U.S.A., 26-28 February. Society of Petroleum Engineers.

Krogstad, S., Lie, K.-A., Møyner, O., Nilsen, H. M., Raynaud, X., Skaflestad, B., et al. (2015). MRST-AD -an open-source framework for rapid prototyping and evaluation of reservoir simulation problems. In SPE reservoir simulation symposium, Houston, Texas, USA, 23-25 February. Society of Petroleum Engineers.

Lee, S. H., Wolfsteiner, C., and Tchelepi, H. A. (2008). Multiscale finite-volume formulation for multiphase flow in porous media: black oil formulation of compressible, three-phase flow with gravity. *Comput. Geosci.*, **12**(3):351–366.

Li, R., Reynolds, A., Oliver, D., et al. (2003). History matching of three-phase flow production data. *SPEJ*, **8**(04):328–340.

Li, X. and Zhang, D. (2014). A backward automatic differentiation framework for reservoir simulation. *Computational Geosciences*, **18**(6):1009–1022.

Lie, K., Møyner, O., Natvig, J., Kozlova, A., Bratvedt, K., Watanabe, S., and Li, Z. (2016). Successful application of multiscale methods in a real reservoir simulator environment. In ECMOR XIV-15th European Conference on the Mathematics of Oil Recovery, Amsterdam, The Netherlands, 29 August-1 September.

Manea, A. M., Sewall, J., and Tchelepi., H. A. (2015). Parallel multiscale linear solver for highly detailed reservoir. *SPE Reservoir Simulation Symposium*, pages 1168–1193.

Moraes, R., Rodrigues, J., Hajibeygi, H., and Jansen, J. D. (2016). Multiscale gradient computation for subsurface flow models. *Journal of Computational Physics*, (submitted 13 July 2016).

Oliver, D. S. and Chen, Y. (2011). Recent progress on reservoir history matching: a review. *Computational Geosciences*, **15**(1):185–221.

Oliver, D. S., Reynolds, A. C., and Liu, N. (2008). *Inverse theory for petroleum reservoir characterization and history matching*. Cambridge University Press.

Peaceman, D. W. (1978). Interpretation of well-block pressures in numerical reservoir simulation. *SPEJ*, **18** (3):183–194.

Rodrigues, J. R. P. (2006). Calculating derivatives for automatic history matching. *Computational Geosciences*, **10**(1):119–136.

Sarma, P., Aziz, K., Durlofsky, L. J., et al. (2005). Implementation of adjoint solution for optimal control of smart wells. In SPE Reservoir Simulation Symposium, The Woodlands, Texas, USA, 31 January-2 Feburary. Society of Petroleum Engineers.

Shah, S., Møyner, O., Tene, M., Lie, K.-A., and Hajibeygi, H. (2016). The multiscale restriction smoothed basis method for fractured porous media (f-msrsb). *Journal of Computational Physics*, **318**:36–57.

Tene, M., Al Kobaisi, M. S., and Hajibeygi, H. (2016). Algebraic multiscale method for flow in heterogeneous porous media with embedded discrete fractures (f-ams). *Journal of Computational Physics*, **321**:819845.

Voskov, D. V., Tchelepi, H. A., Younis, R., et al. (2009). General nonlinear solution strategies for multiphase multicomponent eos based simulation. In SPE Reservoir Simulation Symposium, The Woodlands, Texas, USA, 2-4 February. Society of Petroleum Engineers.

Wang, Y., Hajibeygi, H., and Tchelepi, H. A. (2014). Algebraic multiscale solver for flow in heterogeneous porous media. *Journal of Computational Physics*, **259**:284–303.

Younis, R. and Aziz, K. (2007). Parallel automatically differentiable data-types for next-generation simulator development. In SPE Reservoir Simulation Symposium, Houston, Texas, U.S.A., 26-28 February. Society of Petroleum Engineers.

Zhou, H., Lee, S. H., Tchelepi, H. A., et al. (2012). Multiscale finite-volume formulation for saturation equations. *SPEJ*, **17**(01):198–211.

Zhou, Y., Tchelepi, H. A., Mallison, B. T., et al. (2011). Automatic differentiation framework for compositional simulation on unstructured grids with multi-point discretization schemes. In SPE Reservoir Simulation Symposium, The Woodlands, Texas, USA, 21-23 February. Society of Petroleum Engineers.