Adjoint-based adaptive convergence control of the iterative finite volume multiscale method

de Zeeuw, W.; de Moraes, R. J.; Heemink, A. W.; Jansen, J. D.

DOI
10.2118/193812-MS

Publication date
2019

Document Version
Final published version

Published in
Society of Petroleum Engineers - SPE Reservoir Simulation Conference 2019, RSC 2019

SPE-193812-MS

Adjoint-Based Adaptive Convergence Control of the Iterative Finite Volume Multiscale Method

W. de Zeeuw, Department of Applied Mathematics, Delft University of Technology; R. J. de Moraes, Department of Geoscience and Engineering, Delft University of Technology; Petrobras Research and Development Center - CENPES; A. W. Heemink, Department of Applied Mathematics, Delft University of Technology; J. D. Jansen, Department of Geoscience and Engineering, Delft University of Technology

This paper was prepared for presentation at the SPE Reservoir Simulation Conference held in Galveston, Texas, USA, 10–11 April 2019.

Abstract

We propose a novel adaptive, adjoint-based, iterative multiscale finite volume (i-MSFV) method. The method aims to reduce the computational cost of the smoothing stage of the original i-MSFV method by selectively choosing fine-scale sub-domains (or sub-set of primary variables) to solve for. The selection of fine-scale primary variables is obtained from a goal-oriented adjoint model. An adjoint-based indicator is utilized as a criterion to select the primary variables having the largest errors. The Lagrange multipliers from the adjoint model can be interpreted as sensitivities of the objective function value with respect to deviations from the constraints. In case of adjoining the porous media flow equations with Lagrange multipliers, this implies that the multipliers are the sensitivities of the objective function with respect to the residuals of the flow equations, i.e., to the residual error that remains after approximately solving linear equations with the aid of an iterative solver. This allow us to recognize at which locations the solution contains more errors. More specifically, we propose a modification to the i-MSFV method to adaptively reduce the size of the fine-scale system that must be smoothed. The aim is to make the fine-scale smoothing stage less computationally demanding. To that end, we introduce a goal-oriented, adjoint-based fine-scale system reduction criterion. We demonstrate the performance of our method via single-phase, incompressible flow simulation models with challenging geological settings and using a history-matching like misfit objective function as the goal. The performance of the newly introduced method is compared to the original i-MSFV method. We investigate the adaptivity versus accuracy of the method and demonstrate how the solution accuracy varies by varying the number of unknowns selected to be smoothed. It is shown that the method can provide accurate solutions at reduced computational cost. The proof-of-concept applications indicate that the method deserves further investigations.

Introduction

Model-based reservoir management workflows (e.g. life-cycle optimization, history matching) rely on many evaluation of large-scale numerical simulation models. Each evaluation of the numerical model (forward

simulation) is, by itself, a computationally demanding task. This is, mainly, due to the solution of large linear systems of equations that arises from the numerical discretization of the equations that govern the physical phenomena involved in the exploitation of subsurface resources. Strategies like model order reduction (Jansen and Durlofsky 2017) and upscaling (Durlofsky 2005) aim to reduce this computation cost at the expense of simplifying/substituting the more rigorous geocellular models.

Multiscale (MS) modeling (Hou and Wu 1997; Jenny et al. 2003), a simulation strategy initially employed as an upscaling strategy, has been proven to be efficient to solve system of equations arising from the numerical discretization of elliptic and parabolic parial differential equations (Ţene et al. 2015; Zhou and Tchelepi 2008). Due to its mass conservation nature, the multiscale finite volume (MSFV) (Jenny et al. 2003) method is the preferred one in reservoir simulation (Li et al. 2015; Kozlova et al. 2015). However, the ability to accurately solve the flow problem in highly heterogeneous porous media depends on the fine-scale improvement of the localization conditions provided by the two-stage MS scheme (Zhou and Tchelepi 2012; Wang et al. 2014). The iterative MSFV (i-MSFV) method introduced by (Hajibeygi et al. 2008) smooths out the high-frequency errors resulting from the MS solution by successive fine-scale preconditioning operations. However, the fine-scale smoothing of large-scale numerical models can still be computationally demanding. For a superior efficiency, localization conditions can be adaptively improved where these are not acceptable depending to a certain criterion (e.g. the pressure equation residual) (Hajibeygi and Jenny 2011).

Gradient-based optimization techniques, regarded as the most efficient ones to address the aforementioned workflows (Oliver et al. 2008; Jansen 2011), requires efficient derivative computation methods. The adjoint method (Chavent et al. 1975; Rodrigues 2006; Kraaijevanger et al. 2007; Oliver et al. 2008) is regarded as the most efficient and accurate derivative computation strategy that has been widely used in reservoir simulation for different purposes, e.g. history matching and life-cycle optimization. However, adjoint models can provide more than just gradient information for optimization algorithms. In (Lien et al. 2008) an adjoint model is used to provide 'refinement indicators to adaptively refine or coarsen groups of wells and time steps for optimization. Exactly the same approach can be used for history matching (Chavent and Bissell 1998; Ameur et al. 2002) by refining/coarsening the history matching (parameter) grid.

Note that, in the aforementioned works, the adjoint model is utilized as, basically, a regularization strategy. However, the same strategy could be employed to refine/coarsen the parameter grid and/or the computational (state variable) grid. On the latter, as discussed in (Jansen 2016), in case of adjoining the forward model equations with Lagrange multipliers, this implies that the multipliers are the sensitivities of the objective function with respect to the residuals of the flow equations. In other words, the Lagrange multipliers can be interpreted as sensitivities of the objective function value with respect to deviations from the constraints.

Goal-oriented error estimation (Prudhomme and Oden 1999) has been extensively used to assess and control the accuracy of finite element numerical approximations. This class of error estimations requires the solution of the adjoint of the tangent linear forward model (the dual problem) and the computation of global error estimates. Such error estimates provide information, for instance, for the application of adaptive grid refinement (Alauzet and Loseille 2016).

In this work, we develop an adjoint-based, goal-oriented method and propose a modification to the i-MSFV method so that only the primary variables associated with high goal sensitivities are iteratively solved at the fine-scale smoothing stage. More specifically, we propose a modification to the i-MSFV method to adaptively reduce the size of the fine-scale system that must be smoothed. In the numerical results section, we present some early investigations on how the method can improve the forward simulation performance. The remainder of this paper is organized as follows. Next, we briefly recall the discretized system of equations we aim to solve, as well as the original i-MSFV method we aim to modify. Following the presentation of this background, we introduce an adjoint-based goal-oriented strategy which computes the sensitivities that will serve as a fine-scale system reduction criterion. A modification to the i-MSFV method

is then introduced, which, essentially, consists of an adaptive reduction of the fine-scale system of equations in the i-MSFV second stage. The performance of the i-MSFV goal-oriented method is then assessed based on a typical quadratic misfit objective function. Remarks about the method, potential improvements, as well as required further investigations are indicated just after. Finally, a summary of the developments, as well concluding remarks are presented.

Preliminaries

Discretized forward model equations

It is assumed that the flow-discretized system of equations can be described as (Aziz and Settari 1979)

$$\mathbf{A}(\theta)\mathbf{x} = \mathbf{q}(\theta). \tag{1}$$

Let N_F be the total number of fine grid blocks used in the fine-scale grid employed in the discretization of Eq. (1), so $\mathbf{A} \in \mathbb{R}^{N_F} \times \mathbb{R}^{N_F}$ is the system matrix, $\theta \in \mathbb{R}^{N_\theta}$ is the vector of parameters, $\mathbf{x} \in \mathbb{R}^{N_F}$ the vector of unknowns and $\mathbf{q} \in \mathbb{R}^{N_F}$ the vector of source terms. Equation (1) assumes an implicit dependency of x on θ,

$$\mathbf{x} = \mathbf{x}(\theta). \tag{2}$$

Two-stage iterative multiscale method

A two-stage multiscale (MS) solution strategy (Zhou and Tchelepi 2012; Wang et al. 2014) can be devised by first computing a coarse-scale solution as

$$(\mathbf{RAP})\,\breve{\mathbf{x}} = (\mathbf{Rq})\,, \tag{3}$$

or as

$$\breve{\mathbf{A}}\breve{\mathbf{x}} = \breve{\mathbf{q}}, \tag{4}$$

and then an approximated fine-scale solution as

$$\mathbf{x}' = \mathbf{P}\breve{\mathbf{x}}. \tag{5}$$

Let N_C be the number of coarse grid-blocks. The so-called prolongation operator $\mathbf{P} = \mathbf{P}(\theta)$ which is an $N_F \times N_C$ matrix that maps (interpolates) the coarse-scale solution to the fine-scale resolution. The so-called restriction operator $\mathbf{R} = \mathbf{R}(\theta)$ is defined as an $N_C \times N_F$ matrix which maps the fine scale to the coarse scale. More information about how these operators are constructed for the Multiscale Finite Volume (MSFV) method can be found in (Zhou and Tchelepi 2008; Wang et al. 2014). Let $\breve{\mathbf{x}} \in \mathbb{R}^{N_C}$ be the coarse-scale solution ($N_C \ll N_F$), $\mathbf{x}' \in \mathbb{R}^{N_F}$ the approximated fine-scale solution, $\breve{\mathbf{A}} \in \mathbb{R}^{N_C} \times \mathbb{R}^{N_C}$ the coarse-scale system and $\breve{\mathbf{q}} \in \mathbb{R}^{N_C}$ the coarse-scale system right-hand side (RHS).

The iterative multiscale strategy can be devised by considering versions of Eq. (1) and Eq. (3) written in residual form (Hajibeygi et al. 2008). Let $\mathrm{x}_{\nu-1}$ be an approximate solution to Eq. (1) at iteration $\nu - 1$ and

$$\mathbf{r}^{\nu-1} = \mathbf{q} - \mathbf{A}\mathbf{x}^{\nu-1} \tag{6}$$

be the corresponding residual. A multiscale improvement to this approximation can be devised by writing Eq. (3) in residual form as

$$\breve{\mathbf{A}}\breve{\mathbf{x}}^{\nu} = \breve{\mathbf{r}}^{\nu-1} \tag{7}$$

where

$$\breve{\mathbf{r}}^{\nu-1} = \mathbf{R}\mathbf{r}^{\nu-1}, \tag{8}$$

$\breve{\mathbf{r}}^{\nu-1} \in \mathbb{R}^{N_C}$. Here, $\breve{\mathbf{x}}^{\nu}$ is redefined as the coarse-scale correction. Redefining Eq. (5), we have

$$\mathbf{x}'^{\nu} = \mathbf{P}\breve{\mathbf{x}}^{\nu}, \tag{9}$$

such that x'^{ν} now represents the approximate fine-scale correction at iteration ν, i.e.,

$$\mathbf{x}^{\nu-1/2} = \mathbf{x}^{\nu-1} + \mathbf{x}'^{\nu}, \tag{10}$$

is an approximate solution of Eq. (1) augmented with the correction from the coarse-scale calculation.

The approximate solution provided by Eq. (10) can be improved if successive smoothing steps are employed (Hajibeygi et al. 2008).

Let

$$\mathbf{r}_{\sigma}^{\nu-1} = \mathbf{q} - \mathbf{A}\mathbf{x}^{\nu-1/2} = \mathbf{q} - \mathbf{A}\left(\mathbf{x}^{\nu-1} + \mathbf{x}'^{\nu}\right), \tag{11}$$

$\mathbf{r}_{\sigma}^{\nu-1} \in \mathbb{R}^{N_F}$, be the smoothed residual obtained from the approximation given by Eq. (10) and

$$\mathbf{A}\mathbf{x}_{\sigma}^{\nu} = \mathbf{r}_{\sigma}^{\nu-1}, \tag{12}$$

a version of Eq. (1) written in residual form. Here $\mathbf{x}_{\sigma}^{\nu} \in \mathbb{R}^{N_F}$ is the smoothed fine-scale correction at iteration ν. The solution smoothing is obtained by solving Eq. (29) using any iterative solver up to a prescribed (loose) tolerance or (small) maximum number of iterations (Hajibeygi et al. 2008; Ţene et al. 2015). The solution for a given iteration ν is, hence, obtained from

$$\mathbf{x}^{\nu} = \mathbf{x}^{\nu-1} + \mathbf{x}'^{\nu} + \mathbf{x}_{\sigma}^{\nu}. \tag{13}$$

Therefore, the iterative MS method is, essentially, an iterative coupling which solution is obtained by iterating between Eq. (7) and Eq. (29), through Eq. (9), with the dependency of Eq. (7) on x_{σ} lagged one iteration behind. One must note that $\mathbf{x}_{\sigma}^{\nu} = \mathbf{x}'^{\nu-1}$ after the smoothing stage stopping criteria are met, and $\mathrm{x}_F \approx \mathrm{x}$ after the iterative process stopping criteria are met. Let N_{σ} denote the total number of smoothing steps taken until the stopping criteria is met.

Goal-oriented Adjoint-based Model

Let

$$\mathbf{g}(\mathbf{x}, \theta) = \mathbf{A}(\theta)\mathbf{x} - \mathbf{q}(\theta) = \mathbf{0}, \tag{14}$$

be the set of algebraic model equations that describes the forward model given by Eq. (1), where $\mathbf{g} : \mathbb{R}^{N_F} \times \mathbb{R}^{N_\theta} \to \mathbb{R}^{N_F}$.

Let us suppose that the state vector x is used to evaluate a quadratic or integral goal $\mathcal{J}(\mathbf{x}) : \mathbb{R}^{N_F} \to \mathbb{R}$. Depending on the type of goal imposed on the problem, one can formulate a minimization problem to find the optimal goal value, with g_F used as constraint as

$$\begin{aligned} \underset{\theta}{\text{minimize}} \quad & \mathcal{J}(\mathbf{x}) \\ \text{subject to} \quad & \mathbf{g}_F(\mathbf{x}, \theta) = \mathbf{0}. \end{aligned} \tag{15}$$

In order to solve the minimization problem (15) efficiently, gradient information is required. In practice, solving for this information is considered to be computationally challenging when N_θ is high (Oliver et al. 2008). Derivative information can be efficiently obtained via the solution of the adjoint of the tangent linear approximation of the original model (Rodrigues 2006; Oliver et al. 2008). This adjoint model can be derived by implicit differentiation (Rodrigues 2006; Kraaijevanger et al. 2007; Oliver et al. 2008). The gradient of the goal with respect to the model parameters is given by

$$\frac{d\mathcal{J}(\mathbf{x})}{d\theta} = \frac{d\mathcal{J}(\mathbf{x}(\theta))}{d\theta} = \frac{\partial\mathcal{J}(\mathbf{x})}{\partial\mathbf{x}}\frac{d\mathbf{x}}{d\theta} \tag{16}$$

Differentiation of Eq. (14) with respect to θ leads to

$$\frac{d\mathbf{g}}{d\theta} = \frac{\partial \mathbf{g}}{\partial \mathbf{x}}\frac{d\mathbf{x}}{d\theta} + \frac{\partial \mathbf{g}}{\partial \theta} = \mathbf{0}, \tag{17}$$

or

$$\frac{d\mathbf{x}}{d\theta} = -\frac{\partial \mathbf{g}}{\partial \mathbf{x}}^{-1}\frac{\partial \mathbf{g}}{\partial \theta}. \tag{18}$$

Substituting Eq. (18) in Eq. (16) leads to

$$\frac{d\mathcal{J}(\mathbf{x})}{d\theta} = -\frac{\partial \mathcal{J}(\mathbf{x})}{\partial \mathbf{x}}\frac{\partial \mathbf{g}}{\partial \mathbf{x}}^{-1}\frac{\partial \mathbf{g}}{\partial \theta}. \tag{19}$$

The product $-\frac{\partial \mathcal{J}(\mathbf{x})}{\partial \mathbf{x}}\frac{\partial \mathbf{g}}{\partial \mathbf{x}}^{-1}$ in Eq. (19) can be obtained by solving

$$\frac{\partial \mathbf{g}}{\partial \mathbf{x}}^{\top}\lambda = -\frac{\partial \mathcal{J}(\mathbf{x})}{\partial \mathbf{x}}^{\top}. \tag{20}$$

Equation Eq. (20) is referred to as the *adjoint equation*, where $\lambda \in \mathbb{R}^{N_F}$ are the *adjoint variables*, or Lagrange multipliers. Note that, as shown by, e.g. Kraaijevanger et al. (2007); Jansen (2016); de Zeeuw (2018); de Moraes (2018), the implicit differentiation method provides the same adjoint equations as the more widely employed Lagrange multipliers method.

The Lagrange multipliers can be interpreted as sensitivities of the objective function value with respect to deviations from the constraints. In case of adjoining the (nonlinear) porous media flow equations with Lagrange multipliers, this implies that the multipliers are the sensitivities of the objective function with respect to the residuals of the flow equations. In other words, they may be interpreted as measures of the effect of perturbing the model equations, and thus the state, on the goal $\mathcal{J}(\mathbf{x})$ (Jansen 2016). The matrix $\frac{\partial \mathbf{g}}{\partial \theta}$ of size $N_F \times N_\theta$ gives, for each unknown, the rate of change in the model equations when the parameters of the systems are perturbed.

After transposing Eq. (20)

$$\frac{d\mathcal{J}(\mathbf{x})}{d\theta} = \lambda^{\top}\frac{\partial \mathbf{g}}{\partial \theta}, \tag{21}$$

one can find that the sensitivity of the goal with respect to a given parameter θ_j is given by

$$\frac{d\mathcal{J}(\mathbf{x})}{d\theta_j} = \lambda^{\top}\frac{\partial \mathbf{g}}{\partial \theta_j} = \left[\lambda \cdot \frac{\partial \mathbf{g}}{\partial \theta_1}, \cdots, \lambda \cdot \frac{\partial \mathbf{g}}{\partial \theta_{N_\theta}}\right]^{\top}, \tag{22}$$

where

$$\lambda \cdot \frac{\partial \mathbf{g}}{\partial \theta_j} = \sum_{i=1}^{N_F} \lambda_i \frac{\partial \mathbf{g}_i}{\partial \theta_j}. \tag{23}$$

is the inner product between λ and the *j*-th column of $\frac{\partial \mathbf{g}}{\partial \theta}$.

Let us define the point-wise product

$$s_{ij} = \lambda_i \frac{\partial \mathbf{g}_i}{\partial \theta_j}, \tag{24}$$

as the indicator for the importance of the results of the *i*-th model equation and this adjoint variable for the computation of the sensitivity of the goal with respect to parameter θ_j. In order to obtain an indicator for all parameter sensitivities we define

$$s_i = \sum_{j=1}^{N_\theta} |s_{ij}|. \tag{25}$$

Recalling that the gradient of the goal (Eq. (16)) measures the sensitivity of the goal with respect to variations of the parameters, let us now consider one of the parameters θ_j and the gradient of the goal with respect to this parameter; s_{ij} can then be interpreted as this sensitivity. Moreover, we define the total sensitivity indicator vector as

$$\mathbf{s} = \begin{bmatrix} s_1 & \dots & s_i & \dots & s_{N_F} \end{bmatrix}^\top. \tag{26}$$

If the *i*-th component of s is high, it means that perturbations in one or more parameters have a large effect on the *i*-th model equation. In contrast, if the value is relatively small, perturbing the parameters has a relatively small effect to our goal. Therefore, such information indicates which model equations have actually a significant effect on the goal $\mathcal{J}$. As a consequence, s, in the scope of this work, provides information that can be exploited in the forward simulation so that the solution of the model equations is focused on the just those equations that influence the goal. Based on that, we propose in the next section a linear system solver that exploits this information.

Adaptive Iterative Multiscale Finite Volume Method

As already mentioned, the i-MSFV smoothing stage has cost proportional to N_F. Our aim is to reduced the fine-scale linear system given by Eq. (29). Here, we devise a reduction method based on the idea that an accurate solution is only necessary for the unknowns/equations to which the goal functional $\mathcal{J}$ has a large sensitivity to.

Let $\mathcal{P}$ be a $\mathbb{R}^{N_F} \times \mathbb{R}^{N_F}$ permutation matrix (Golub and Van Loan 1996) that re-orders the original system of equations given by Eq. (1) in a block-wise fashion

$$\left(\mathcal{P}\mathbf{A}\mathcal{P}^T\right)\left(\mathcal{P}\mathbf{x}\right) = \begin{bmatrix} \mathbf{A}_{xx} & \mathbf{A}_{xp} \\ \mathbf{A}_{px} & \mathbf{A}_{pp} \end{bmatrix} \times \begin{bmatrix} \mathbf{x}_x \\ \mathbf{x}_p \end{bmatrix} = \begin{bmatrix} \mathbf{q}_x \\ \mathbf{q}_p \end{bmatrix} = \mathcal{P}\mathbf{q} \tag{27}$$

where A_x is a $\mathbb{R}^{N_x} \times \mathbb{R}^{N_x}$ matrix composed of equations regarding the N_x unknowns selected, A_p is a $\mathbb{R}^{N_p} \times \mathbb{R}^{N_p}$ matrix composed of equation regarding the equations which N_p variables are considered known according to a given criterion, and $\mathbf{A}_{xp} \in \mathbb{R}^{N_x} \times \mathbb{R}^{N_p}$ and $\mathbf{A}_{px} \in \mathbb{R}^{N_p} \times \mathbb{R}^{N_x}$ are matrices representing the couplings between known and unknown parts of the variables. $\mathbf{x}_x \in \mathbb{R}^{N_x}$ and $\mathbf{x}_p \in \mathbb{R}^{N_p}$ are, respectively, the known and unknown parts of the state vectors. Also, $\mathbf{q}_x \in \mathbb{R}^{N_x}$ and $\mathbf{q}_p \in \mathbb{R}^{N_p}$ are, respectively, the knowns and unknowns source vectors.

Assuming that certain variables are known enables us to remove these equations from the system Eq. (27). Hence, under the strong assumption that $A_{xp} = 0$, $A_{px} = 0$ and $A_{pp} = I$, it follows from Eq. (27), that

$$\mathbf{A}_{xx}\mathbf{x}_x = \mathbf{q}_x, \tag{28}$$

is the system that should be iterated over in the i-MSFV smoothing stage, meaning that Eq. (29) can be re-written as

$$\mathbf{A}_{\mathbf{xx}}\mathbf{x}_{x\sigma}^{\nu} = \mathbf{r}_{x\sigma}^{\nu-1}, \tag{29}$$

where $\mathbf{x}_{x\sigma} \in \mathbb{R}^{N_x}$ and $\mathbf{r}_{x\sigma} \in \mathbb{R}^{N_x}$ are, respectively, the reduced smoothed fine-scale correction and the reduced smoothed residual. This is equivalent to considering the known positions as Dirichlet boundary conditions and setting the right-hand-side residual associated with those prescribed values equal to zero.

We propose a system reduction based on the indicator vector s defined in Eq. (26) such that

$$\text{if } \mathbf{s}_i < \epsilon, \mathbf{x}_i \in \mathbf{x}_p, \text{ else }, \mathbf{x}_i \in \mathbf{x}_x, \tag{30}$$

where ϵ is a sensitivity indicator threshold. To this aim, we build the permutation matrix $\mathcal{P}$ using the rule given by Eq. (30) so that Eq. (28) is only comprised by model equations (and the corresponding adjoint variables) that have a major importance to the goal. We highlight that, in order to make the definition of ϵ independent of the problem setting, we apply a normalization of s. Therefore, the fine-scale smoothing system is adapted to approximately solve for the unknowns associated with the model equations with high goal impact. As a consequence, we are only reducing the residual associated with those equations, and neglecting the residual associated with the variables left out from the smoothing step. The algorithm for the goal-oriented adaptive i-MSFV method is presented in Algorithm 1. We refer to it as the iMSGo(ϵ) method.

Algorithm 1: Goal-oriented iterative multiscale (iMSGo(ϵ)) method.

input : $\mathbf{A}$, $\mathbf{q}$, m_tol, ϵ
output: The state of the system $\mathbf{x}$

Set $\mathbf{r}^0 = \mathbf{q}$;
Set $\nu = 1$;
Set $\mathbf{x} = \mathbf{0}$;
Construct $\mathbf{s}$ given ϵ;
Construct $\mathcal{P}$ based on $\mathbf{s}$;
while $||\mathbf{r}||_2 > m_{tol}$ **do**
  Solve $\mathbf{A}\breve{\mathbf{x}}^\nu = \breve{\mathbf{r}}^{\nu-1}$;
  Compute $\mathbf{x}' = \mathbf{P}\breve{\mathbf{x}}^\nu$;
  Set $\mathbf{x} = \mathbf{x} + \mathbf{x}'^\nu$;
  Set $\mathbf{r}^\nu = \mathbf{r}^\nu - \mathbf{A}\mathbf{x}'^\nu$;
  if $\epsilon \neq 1$ **then**
    Permute matrix $\tilde{\mathbf{A}} = (\mathcal{P}\mathbf{A}\mathcal{P}^\top)(\mathcal{P}\mathbf{x}'^\nu)$;
    Permute residual $\tilde{\mathbf{r}} = \mathcal{P}\mathbf{x}'^\nu$;
    Smooth and solve for $\tilde{\mathbf{A}}_{\mathbf{xx}}\tilde{\mathbf{x}}^\nu_{\sigma\mathbf{x}} = \tilde{\mathbf{r}}$;
    Set $\tilde{\mathbf{r}} = \tilde{\mathbf{r}} - \tilde{\mathbf{A}}\mathbf{x}^\nu_\sigma$;
    Permute $\mathbf{r}^\nu = \mathbf{r}^\nu - \mathcal{P}^\top\tilde{\mathbf{r}}$ and $\mathbf{x}^\nu_{\sigma\mathbf{x}} = \mathcal{P}^\top\mathbf{x}^\nu_\sigma$;
    Set $\mathbf{x} = \mathbf{x} + \mathbf{x}^\nu_\sigma$;
    Set $\mathbf{r}^\nu = \mathbf{r}^\nu - \mathbf{A}\mathbf{x}^\nu_\sigma$;
    Compute $\bar{\mathbf{A}}$ and set $\mathbf{A} = \bar{\mathbf{A}}$;
  else
    Set $\mathbf{r}^\nu = \mathbf{0}$;
  $\nu = \nu + 1$;
Return $\mathbf{x}$;

In the algorithm, m_{tol} is the iMSGo(ϵ) residual tolerance. We highlight that, in the algorithm, if $\epsilon = 1$ the method is equivalent to the MSFV method. This is because, by construction, all positions are to be considered as known, hence no smoothing is applied. In a similar fashion we have that if $\epsilon = 0$ the method is equivalent to the i-MSFV method, because all positions are considered to be unknown, hence smoothing is applied across the entire solution domain.

Remarks about the method

Convergence. The iterative scheme proposed does not guarantee that the overall residual is reduced. Instead, only the residuals associated with the unknowns flagged by the goal are reduced. Moreover, there

is no guarantee that the residuals associated with the other unknowns will increase. However, if one is interested, the overall decrease of the residual could be addressed if extra iterations over the entire domain are introduced after the iMSGo(ϵ) converges to m_{tol}. However, we claim that this should not necessarily be of interest. For instance, in our studies, the goal provides us the indication which primary variables need to be accurately solved given given that the solution will be used in a history matching exercise.

Efficiency. The method (and consequentially the algorithm) requires the permutation of the fine-scale system of equations in order to construct the smoothing system. However, matrix permutation is not a scalable operation in a parallel setting as it implies heavy communication among processes. Algorithms with good parallelization properties are desirable in order to fully take advantage of modern hardware architectures. This is a potential issue for an efficient parallel implementation of the proposed method.

Numerical Results

The main goal of the numerical experiments is to investigate to computational gain when using the iMSGo(ϵ) method compared to the i-MSFV method. We define a typical misfit history matching objective function without regularization as our goal (Oliver et al. 2008)

$$\mathcal{J}(\mathbf{h}(\mathbf{x},\theta)) = \frac{1}{2}(\mathbf{h}(\mathbf{x},\theta)-\mathbf{d})^{\top}\mathbf{C}^{-1}(\mathbf{h}(\mathbf{x},\theta)-\mathbf{d}), \tag{31}$$

where we assume the observed quantities $\mathbf{d} \in \mathbb{R}^{N_d}$ to be the non-flowing pressure at observation wells, $\mathbf{h} \in \mathbb{R}^{N_d}$ the simulated grid-block pressure at the observation wells and $\mathbf{C} \in \mathbb{R}^{N_d \times N_d}$ a data co-variance matrix (here assumed to be diagonal with small measurement errors). There are N_d observations. In all experiments we have grid-block dimensions of $\Delta x = \Delta y = \Delta z = 1$ m. Furthermore we have a fluid viscosity of 1.0×10^{-3} Pa s. All the experiments are set up in terms of a non-dimensional pressure, i.e.,

$$p_D = \frac{p - p_{prod}}{p_{inj} - p_{prod}}, \tag{32}$$

where p_{inj} and p_{prod} are the injection and production pressures, respectively. In all the experiments, $p_{inj} = 1.0$ and $p_{prod} = 0.0$.

We note that because we are using a goal function as in equation Eq. (31), the algebraic formulation of the Lagrange multipliers can be written as

$$\lambda^{\top} = -\mathbf{w}\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\mathbf{A}^{-1}, \tag{33}$$

where $\mathbf{w} \in \mathbb{R}^{N_d}$ indicates the misfit at the observation wells (de Moraes et al. 2017). We can therefore see that, in this case, the Lagrange multipliers are directly dependent on the misfit goal functional, i.e. if the misfit is large, then so will be the Lagrange multipliers.

In the experiments, the construction of s is based on a previous fine-scale adjoint model solution. This implies the actual solution of a fine-scale system and its adjoint model. Even though this is an expensive operation, the indicators can be stored and utilized multiple times, e.g. in an optimization process. Furthermore, a more computationally efficient multiscale adjoint method (de Moraes et al. 2017) could also be employed in the construction of s. Even though the method is more efficient, it provides an approximate adjoint gradient. Early investigations on whether the MS adjoint provides indicators that are consistent with the fine-scale adjoint model can be found in de Zeeuw (2018).

We focus our analysis on two parameters: the total number of smoothing steps required for convergence to the outer tolerance m_{tol} and the accuracy of the method assessed via the pressure error 2-norm defined as

$$\gamma = ||\mathbf{x}_F - \mathbf{x}_M||_2, \tag{34}$$

where $x_F \in \mathbb{R}^{N_F}$ is the pressure solution obtained by using a fine-scale-only solution method and solving the linear system via an LU direct solver and $x_M \in \mathbb{R}^{N_F}$ is the pressure solution obtained by the iMSGO(ϵ) method.

Simple two-dimensional five-spot model

In this experiment the model dimensions are 21 × 21 fine grid cells. There are 1000 geological realizations in the ensemble. These fields are generated using the Principal Component Analysis (PCA) parametrization (Jansen 2013). Examples of these fields can be found in Fig. 1. We apply a coarsening ratio of 7 × 7, leading to a 3 × 3 coarse grid. We impose an inverted five-well spot as the well setting (see Table 1). The observations are generated from a twin-experiment. In this experiment, $m_{tol} = 1e-6$ and $\epsilon_\sigma = 1e-1$.

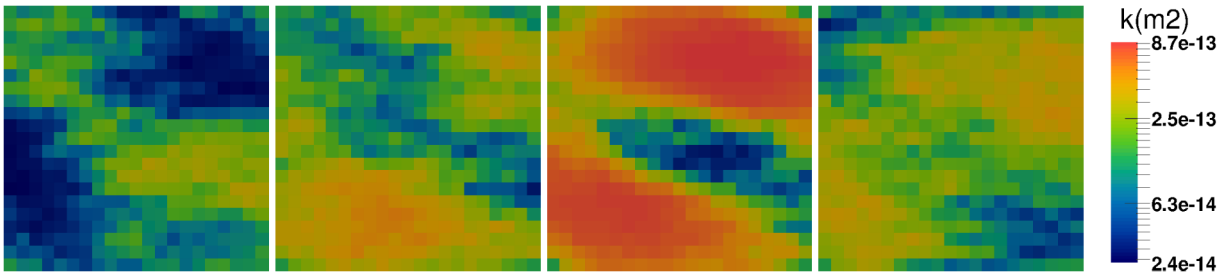


Figure 1—Four different permeability realizations from the ensemble of 1,000 members used in the 2D numerical experiments.

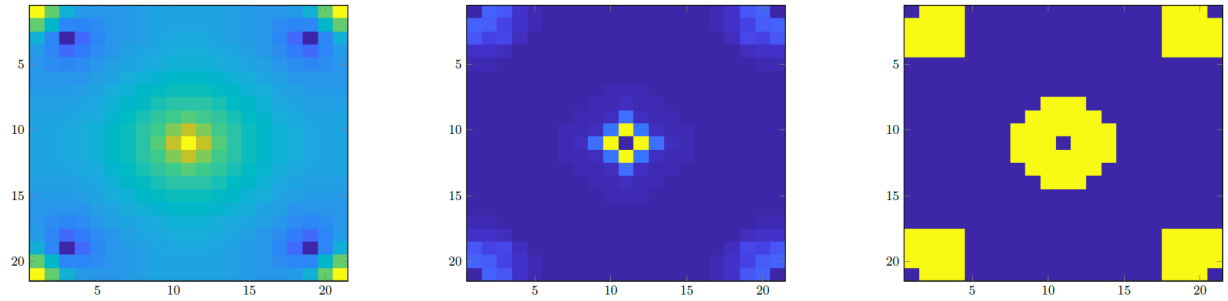
For illustration of the method, we first consider two twin experiments on a simple five-spot model with a homogenous permeability field with different reference fields. One homogeneous reference field and one heterogeneous reference field are randomly chosen from the ensemble. We use $\epsilon = 1e-2$ to build the sensitivity indicator. Fig. 2 shows the results for the experiment where a homogeneous permeability field and a homogeneous twin experiment are used. Because the pressure solution is symmetric, the misfit will therefore be symmetric as well. This also means that the adjoint variables have the same radial symmetry. Consequently, the sensitivity is also symmetric.

For the twin experiment with the heterogeneous reference field, we can see that the adjoint variables indeed are related to the misfit at these positions. Furthermore, we clearly see that the sensitivity region is not symmetric and that the sensitivity region seems to select more positions in the neighbourhood of the largest misfit. This can be easily explained, as the largest misfit leads to high values of our goal, where we want to accurately calculate the solution. Therefore, more positions in this neighbourhood are selected to be smoothed. These results indicate the usefulness of the computation of the adjoint variables and the sensitivity indicators.

Next we investigate how the selected number of unknowns varies as a function of ϵ for the homogeneous test case. Four different sensitivity regions and their corresponding tolerances e are shown in Fig. 4. The yellow positions correspond to the goal positions used in the smoothing step. The blue positions correspond to the non-goal positions. We clearly see that, the tighter the tolerance, the largest the number of positions that require smoothing.

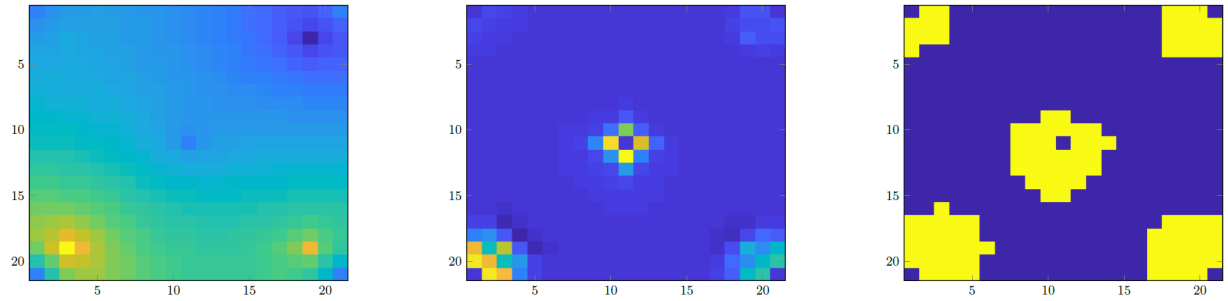
Table 1—Well configuration for the simple two-dimensional five-spot model.

Well	Fine scale position (I, J)	Well type
INJE	(11, 11)	Injection
PROD1	(1, 1)	Production
PROD2	(21, 1)	Production
PROD3	(1, 21)	Production
PROD4	(21, 21)	Production
OBSWELL1	(3, 3)	Observation
OBSWELL2	(19, 3)	Observation
OBSWELL3	(3, 19)	Observation
OBSWELL4	(19, 19)	Observation



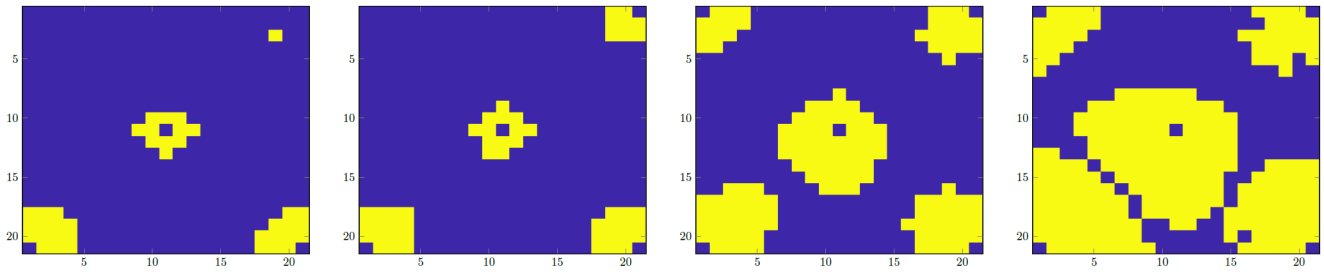
(a) Adjoint state (b) Sensitivity region (c) Selected positions for smoothing

Figure 2—The adjoint states (a), sensitivity region s (b) and the selected positions for smoothing (c) for the simple 2D test case with homogeneous twin experiment. Here, the goal tolerance $\epsilon = 1e-2$.



(a) Adjoint state (b) Sensitivity region (c) Selected positions for smoothing

Figure 3—The adjoint states (a), sensitivity region s (b) and the selected positions for smoothing (c) for the simple 2D test case with heterogeneous twin experiment. Here, the goal tolerance $\epsilon = 1e-2$.



(a) $\epsilon = 1e-1$ (b) $\epsilon = 5e-2$ (c) $\epsilon = 5e-3$ (d) $\epsilon = 1e-3$

Figure 4—Sensitivity regions for the 2D homogeneous test case when different goal tolerances (ϵ) are employed. Yellow means unknowns which will be taken into account in the iMSFV smoothing step. Blue represents unknowns that will be left out.

The pressure error norm and total number of smoothing iterations as a function of the tolerance ϵ are shown in Fig. 5. From the plots, together with the sensitivity regions in Fig. 4, we conclude that the behaviour of the iterative pressure solution is as expected. If the goal tolerance equals 1, the method is equivalent to the MSFV method and has the largest pressure error norm. Gradually decreasing the goal tolerance means increasing the number of positions that we smooth, hence gradually decreasing the error norm. The error norm is the smallest when a goal tolerance of 0 is used, because then the entire model is smoothed. From this figure, we can also see that the total number of smoothing steps decreases as we tighten up the goal tolerance.

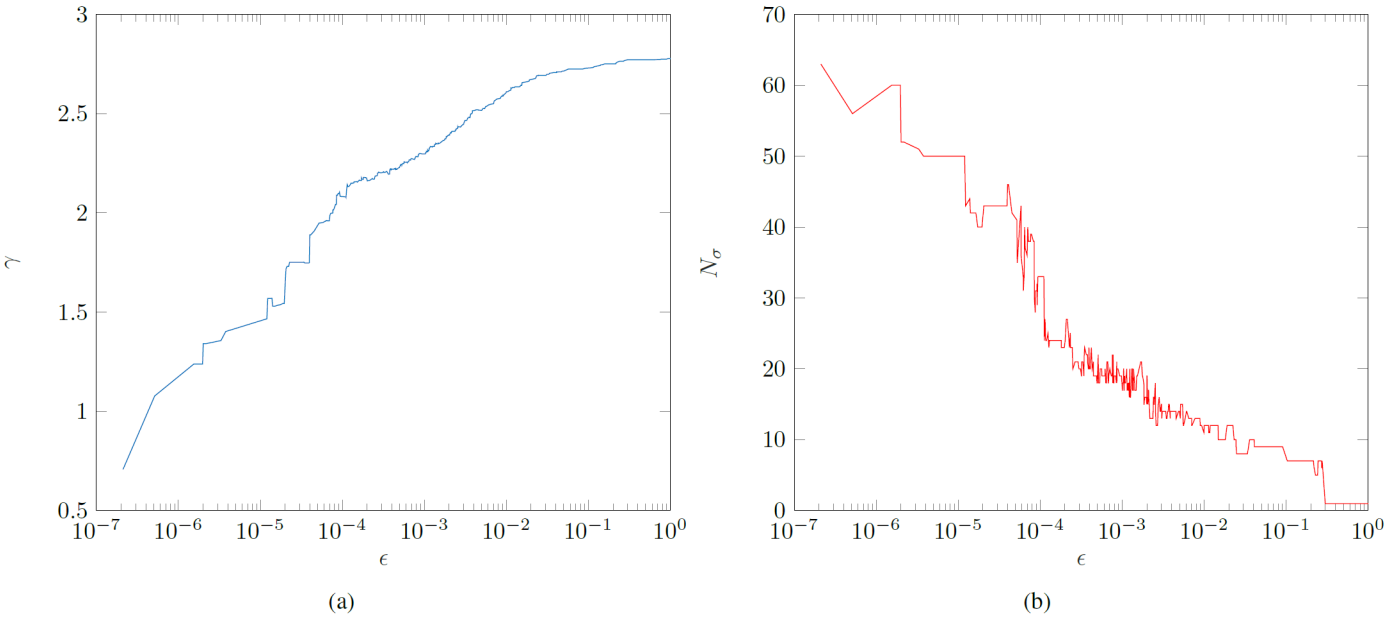


Figure 5—The pressure error norm (a) and the total number of smoothing steps (b) as a function of the goal tolerance for the simple 2D five-spot.

Next, we evaluate the robustness of the iMSGO(ϵ) method by employing the method over the entire ensemble of permeability realizations. For this, we note that, instead of directly specifying a goal tolerance, we calculate the required goal tolerance to smooth over a number of x_σ positions, where $0 \leq x_\sigma \leq 441$ in this experiment. This is done because the goal tolerances can result in the selection of a different number of unknowns to solve for at each different realization. Therefore, by doing this we ensure that each test case is equivalent from the computational point of view. Fig. 6 shows the behaviour of the method with respect to the total number of iterations and the pressure error norm.

The results with respect to the pressure error norm indicate the usefulness of the method. Gradually the error norm decreases when the goal tolerance is increased. However, it can be noted that the decrease of the pressure error norm is slow. When smoothing 92.5% of the complete domain, the error norm decreases on average with only 1% compared to the MSFV method. As it regards the number of smoothing steps, we find that the results are comparable to those for the homogeneous case. However there is a strong increase in the total number of smoothing iterations when comparing the i-MSFV method to the iMSGo(ϵ) method with a low goal tolerance. Tightening up the goal tolerances over the 1000 test cases eventually leads to a substantial decrease in the amount of smoothing steps. Also we can see that the variance in the convergence behaviour decreases when the goal tolerance is increased.

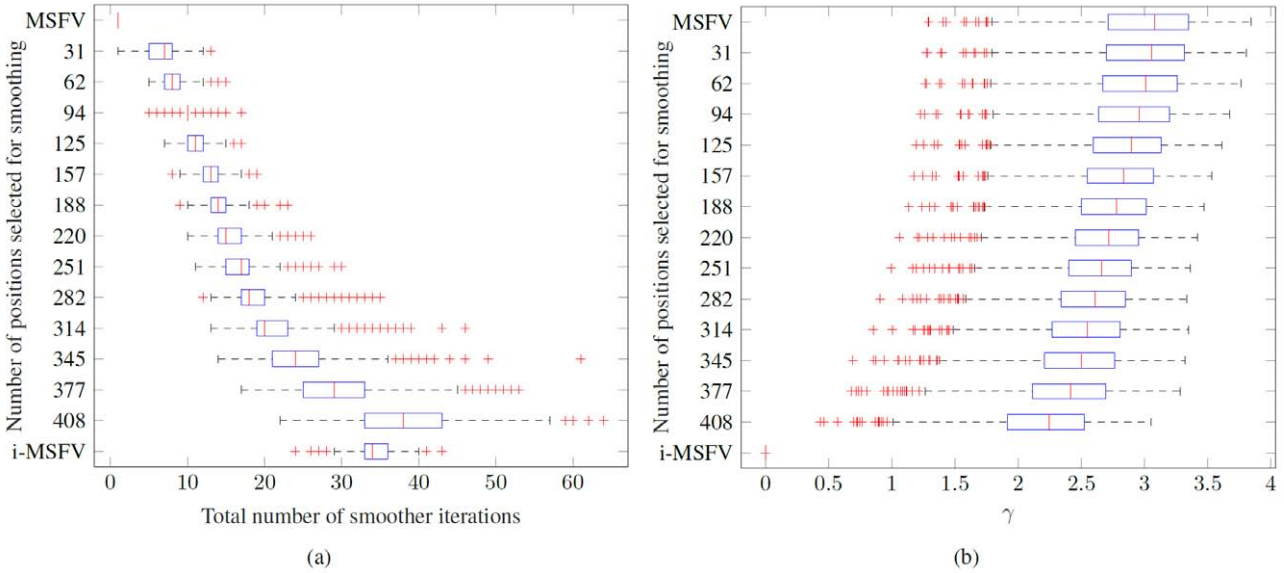


Figure 6—(a) Total number of smoothing steps required for convergence for the entire ensemble of permeability realizations for different goal tolerances. (b) Pressure error norm for different goal tolerances the entire ensemble of permeability realizations.

SPE-10 Comparative Test Case

We investigate the performance of the method in the SPE-10 comparative case Christie and Blunt (2001). In our experiments, we consider flow through the top and bottom layers of this model. The corresponding permeability fields can be found in Fig. 7.

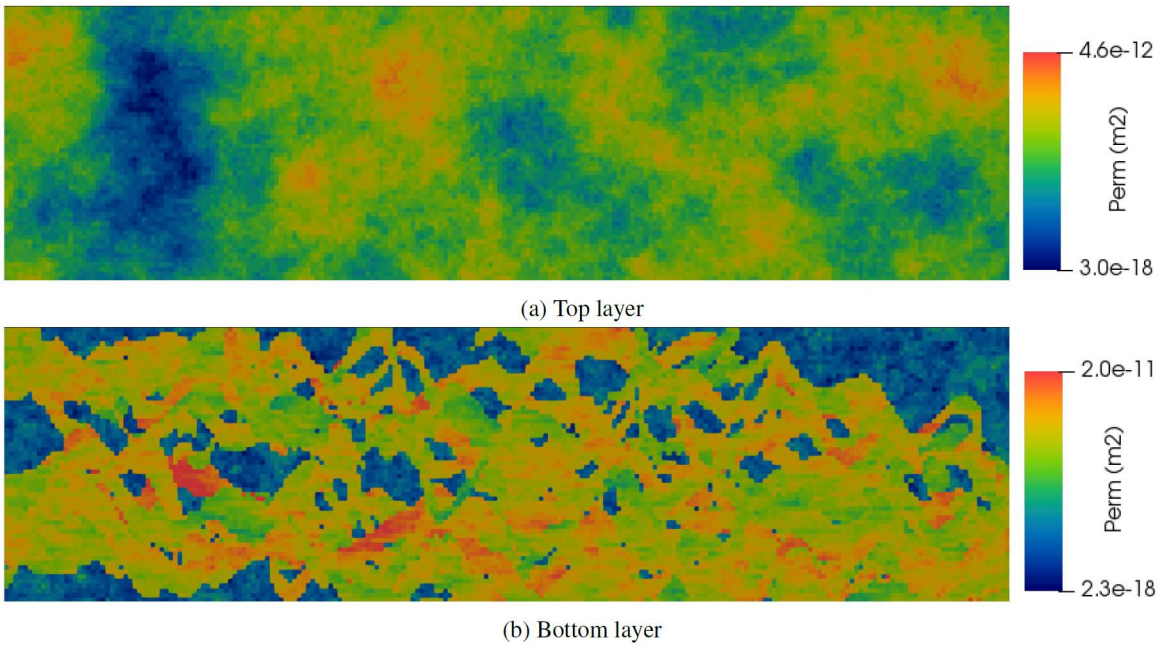


Figure 7—SPE-10 comparative test case permeability fields for the top (figure (a)) and bottom (figure (b)) layer.

Table 2—Well configuration for the simple two-dimensional five-spot model.

Well	Fine scale position (I, J)	Well type
INJE	(1,220)	Injection
PROD	(60,1)	Production
OBSWELL1	(33,5)	Observation
OBSWELL2	(28,50)	Observation
OBSWELL3	(28,83)	Observation
OSBWELL4	(43, 204)	Observation

We employ a constant coarsening ratio of 5×11, hence generating a coarse grid of 12 × 20 grid blocks. The observations are generated from a homogeneous twin-experiment with a permeability value of $k = 1 \cdot 10^{-13}\text{m}^2$. The specifications of the well configuration can be found in Table 2. In this experiment we fix $\epsilon_\sigma = 1.0 \cdot 10^{-2}$ and we set the outer residual with a value of $1.0 \cdot 10^{-6}$.

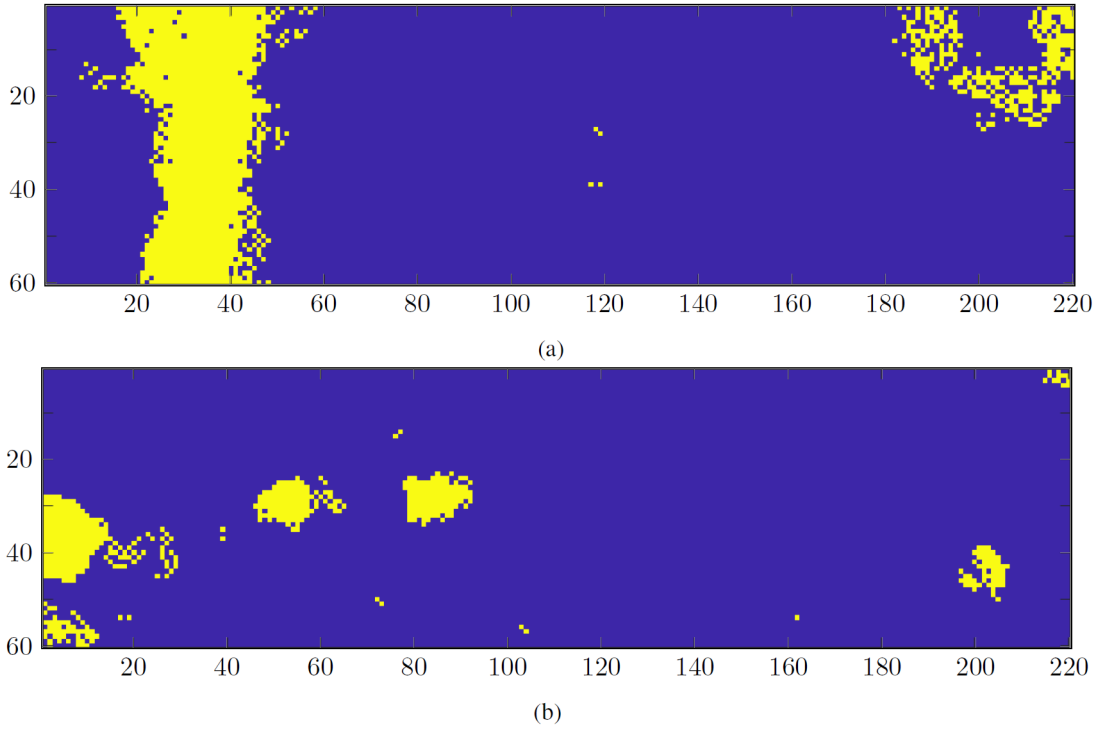


Figure 8—Sensitivity regions for the SPE-10 test case. Top layer (a) and bottom layer (b). Here $\epsilon = 1e - 5$.

Since the computational domain is very large, and a large decrease of the smoothing step would be very attractive, we investigate the behaviour of the number of total smoothing steps when smoothing covers 25% of the domain or more. This is equivalent to smoothing 3300 or more unknowns. Fig. 9 illustrates the results for the top layer and Fig. 10 those for the bottom layer.

Overall, the same behaviours as discussed in the simple 2D test case is observed, indicating that the method is robust even when applied to cases with high permeability contrast.

Concluding Remarks and Future Work

We have introduced an adjoint-based goal-oriented i-MSFV method in which the computational effort in the smoothing stage is alleviated by an adaptive reduction of the fine-scale smoothing linear system. The adaptive reduction is achieved by means of a novel indicator for the importance of the model equations and the associated adjoint variables to a pre-specified goal. We highlight that the goal-oriented sensitivity

indicator here developed could be used to reduced the forward simulation computational cost in different ways; for instance, as a localization strategy in non-linear solvers or as an indicator for grid refinement/coarsening in adaptive gridding strategies. Also, the adaptation to the i-MSFV algorithm is cast in such way that different smoothing linear system reduction strategies could be employed.

It was observed from the numerical experiments that the computational cost of the linear system solution is reduced both by a decreased number of smoothing steps and by a decreased size of the smoothing step linear system. However, the error norm between the fine-scale and the iMSGo(ϵ) method solutions illustrated that, as expected, there is no guarantee that the method provides accurate solutions with respect to all primary variables. We claim that this is, actually, not desired though. In an optimization-based workflow, we should be only interested in accurate solutions of variables which have most influence on the goal. This work develops the fundamental mathematical framework to support such a strategy, but its efficiency remains to be investigated in optimization studies.

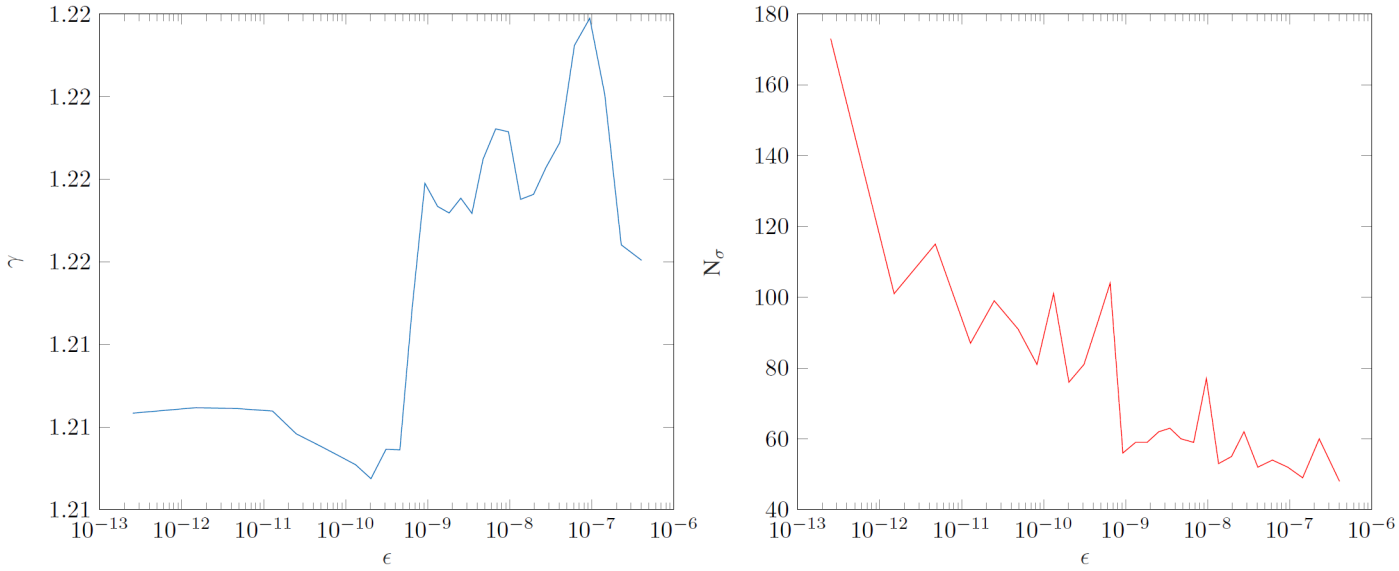


Figure 9—The pressure error norm (left) and the total number of smoothing steps (right) as a function of the goal tolerancel for SPE-10 top layer.

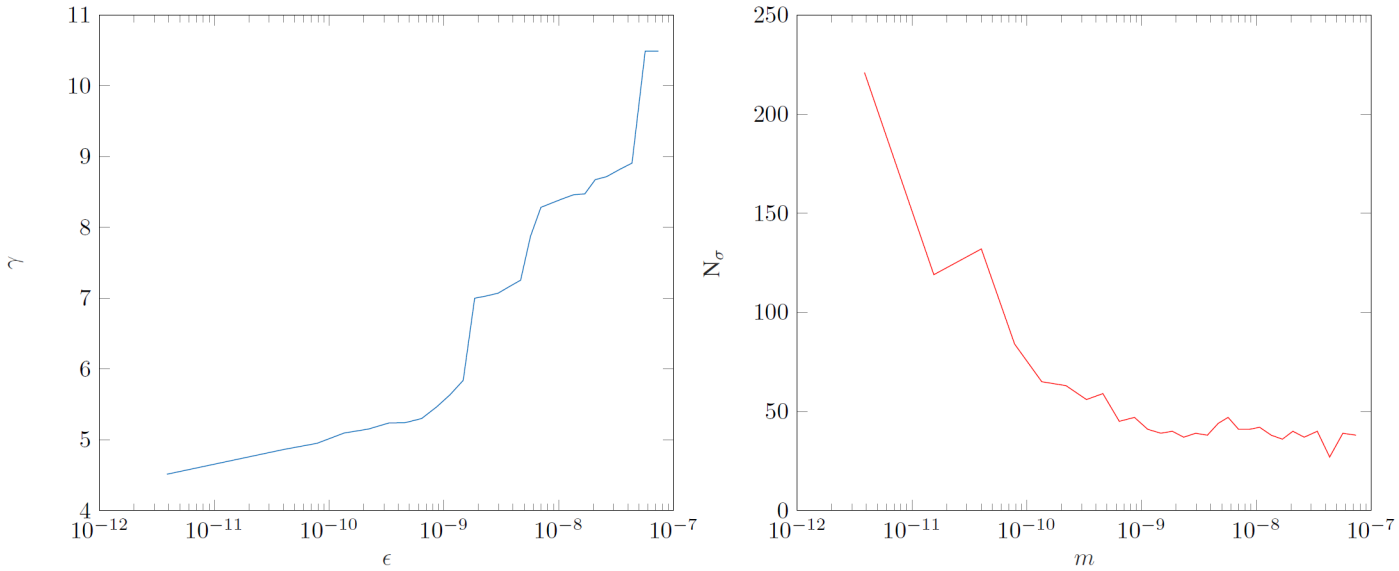


Figure 10—The pressure error norm (left) and the total number of smoothing steps (right) as a function of the goal tolerancel for the SPE-10 bottom layer.

How small the goal tolerance should be set remains a topic for further investigation. The optimal threshold could be set as the one that provides a forward model solution that leads to a gradient that is still in agreement with the fine-scale gradient. A first experiment investigation has already been carried out by de Zeeuw (2018).

Acknowledgements

Part of this research has been carried out in the context of Rafael Moraes' PhD project, which was sponsored by Petrobras.

References

Alauzet, F. and Loseille, A. (2016). A decade of progress on anisotropic mesh adaptation for computational fluid dynamics. *Computer-Aided Design*, **72**:13–39.

Ameur, H. B., Chavent, G., and Jaffré, J. (2002). Refinement and coarsening indicators for adaptive parametrization: application to the estimation of hydraulic transmissivities. *Inverse problems*, **18**(3):775.

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

Chavent, G. and Bissell, R. (1998). Indicator for the refinement of parameterization. In *Inverse problems in engineering mechanics*, pages 309–314. Elsevier.

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

Christie, M. and Blunt, M. (2001). Tenth spe comparative solution project: *A comparison of upscaling techniques*. volume **4**.

de Moraes, R. J. (2018). Multiscale Analytical Derivative Formulations for Improved Reservoir Management. PhD thesis, Delft University of Technology.

de Moraes, R. J., Rodrigues, J. R., Hajibeygi, H., and Jansen, J. D. (2017). Multiscale gradient computation for flow in heterogeneous porous media. *Journal of Computational Physics*, **336**:644–663.

de Zeeuw, W. (2018). Derivative computation using an adjoint-based goal-oriented iterative multiscale lagrangian framework: Application to reservoir simulation. Master's thesis, Delft University of Technology.

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**.

Golub, G. H. and Van Loan, C. F. (1996). Matrix Computations (3rd Ed.). Johns Hopkins University Press, Baltimore, MD, USA.

Hajibeygi, H., Bonfigli, G., Hesse, M. A., and Jenny, P. (2008). Iterative multiscale finite-volume method. *J. Comput. Phys.*, **227**(19):8604–8621.

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

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

Jansen, J. D. (2011). Adjoint-based optimization of multi-phase flow through porous media-a review. *Computers & Fluids*, **46**(1):40–51.

Jansen, J. D. (2013). A simple algorithm to generate small geostatistical ensembles for subsurface flow simulation. *Research note. Dept. of Geoscience and Engineering, Delft University of Technology, The Netherlands*.

Jansen, J. D. (2016). Gradient-based optimization of flow through porous media, v.**3**.

Jansen, J. D. and Durlofsky, L. J. (2017). Use of reduced-order models in well control optimization. *Optimization and Engineering*, **18**(1):105–132.

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

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. 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. Society of Petroleum Engineers.

Li, Z., McWilliams, J. C., Ide, K., and Farrara, J. D. (2015). A multiscale variational data assimilation scheme: formulation and illustration. *Monthly Weather Review*, **143**(9):3804–3822.

Lien, M. E., Brouwer, D. R., Mannseth, T., Jansen, J.-D., et al (2008). Multiscale regularization of flooding optimization for smart field management. *SPE Journal*, **13**(02):195–204.

Oliver, D. S., Reynolds, A. C., and Liu, N. (2008). Inverse Theory for Petroleum Reservoir Characterization and History Matching. Cambridge University Press.

Prudhomme, S. and Oden, J. T. (1999). On goal-oriented error estimation for elliptic problems: application to the control of pointwise errors. *Computer Methods in Applied Mechanics and Engineering*, **176**(1-4):313–331.

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

Ţene, M., Wang, Y., and Hajibeygi, H. (2015). Adaptive algebraic multiscale solver for compressible flow in heterogeneous porous media. *J. Comput. Phys.*, **300**:679–694.

Wang, Y., Hajibeygi, H., and Tchelepi, H. A. (2014). Algebraic multiscale solver for flow in heterogeneous porous media. *J. Comput. Phys.*, **259**:284–303.

Zhou, H. and Tchelepi, H. A. (2008). Operator-based multiscale method for compressible flow. *SPE J.*, **13**(02):523–539.

Zhou, H. and Tchelepi, H. A. (2012). Two-stage algebraic multiscale linear solver for highly heterogeneous reservoir models. *SPE J.*, **17**(02):523–539.