

Mo Efe 11

## Parallel Fully Implicit Smoothed Particle Hydrodynamics Based Multiscale Method

A. Lukyanov\* (Schlumberger-Doll Research) & C. Vuik (Delft University of Technology)

### SUMMARY

---

Preconditioning can be used to damp slowly varying error modes in the linear solver residuals, corresponding to extreme eigenvalues. Existing multiscale solvers use a sequence of aggressive restriction, coarse-grid correction and prolongation operators to handle low-frequency modes on the coarse grid. High-frequency errors are then resolved by employing a smoother on fine grid.

In reservoir simulations, the Jacobian system is usually solved by FGMRES method with two-level Constrained Pressure Residual (CPR) preconditioner. In this paper, a parallel fully implicit smoothed particle hydrodynamics (SPH) based multiscale method for solving pressure system is presented. The prolongation and restriction operators in this method are based on a SPH gradient approximation (instead of solving localized flow problems) commonly used in the meshless community for thermal, viscous, and pressure projection problems.

This method has been prototyped in a commercially available simulator. This method does not require a coarse partition and can be applied to general unstructured topology of the fine scale. The SPH based multiscale method provides a reasonably good approximation to the pressure system and speeds up the convergence when used as a preconditioner for an iterative fine-scale solver. In addition, it exhibits expected good scalability during parallel simulations. Numerical results are presented and discussed.

## Introduction

Academia and the corporate sectors are moving towards exascale simulations with highly complex properties of the underlying numerical domain (e.g., geology, fracture network, micro-inclusions), where it is required to compute an accurate solution with the reasonable time. Hence, current computational problems deal with the high resolutions simulations, which contain billions of degrees of freedom (i.e., unknowns). Existing modern computational resources stimulate the development of novel solver techniques tailored to these problems. At the same time, specialized parallel and multicore architectures have driven the implementation of these new methods with unprecedented performance expectations.

Commercial reservoir simulators usually use the Newton-Raphson method to solve the non-linear governing equations for a given timestep. The corresponding Jacobian matrix and linear system are solved by the Flexible Generalized Minimum Residual Method (FGMRES) (Saad and Schultz (1986)) preconditioned by the Constrained Pressure Residual (CPR) preconditioner (see, e.g., Wallis (1983), Wallis (1985), Cao et al. (2005)). CPR decouples the linear system into two sets of equations, exploiting the specific properties of the pressure equation and transport equations. The former is solved with an Algebraic Multigrid (AMG) preconditioner (see, Ruge and Stuben (1987)), while the fully coupled system is solved using an ILU(k) type preconditioner. Cao et al. (2005) showed that a linear solver based on CPR-AMG preconditioning is extremely robust in terms of algorithmic efficiency. However, there are still challenges to overcome in implementing a near-ideal scalable AMG solver. Also, the second stage of CPR preconditioning often includes some variant of an incomplete LU factorization (e.g., ILU(0)), which again is non-trivial to parallelize. As a result, the linear solver is still far from near-ideal scalability.

To improve the AMG robustness, Klie et al. (2007) tried to consider deflation AMG solvers for highly ill-conditioned reservoir simulation problems. However, the manual construction of the deflation vectors is a time consuming process and will most likely lead to sub-optimal results (see, van der Linden (2013)). However, the deflation strategy (which has a lower algebraic complexity and is inexpensive to set up) may lead to a robust alternative of AMG in general. Furthermore, the deflation strategy also (strongly) scalable which can be combined with the robust preconditioners. Deflation was first proposed for symmetric systems and the CG method by Nicolaidis (1987) and Dostál (1988). Both construct a deflation subspace consisting of deflation vectors to deflate problematic eigenvalues from the linear system. A plethora of widely used deflation algorithms have been developed since, differing primarily in the application of the deflation operator (e.g. as a preconditioner or projection preconditioner) and the method to construct the deflation vectors (see, Frank and Vuik (2001), Vuik et al. (2002), Tang and Vuik (2007a), Tang and Vuik (2007c), Tang and Vuik (2007b), Tang (2008a), Jönsthövel et al. (2009), Jönsthövel et al. (2011), van der Linden et al. (2016)).

Also, as alternative to AMG solver for the pressure system, two-level multiscale solvers (MS) have received considerable attention in the literature over the past decades (see, e.g., Hou and Wu (1997), Jenny et al. (2003), Lunati and Jenny (2006), Hajibeygi et al. (2008), Efendiev and Hou (2009), Lunati et al. (2011), Zhou and Tchelepi (2012), Cortinovis and Jenny (2014), Wang et al. (2014), Tene et al. (2014), Lukyanov et al. (2014a), Cusini et al. (2014), Kozlova et al. (2015), Manea et al. (2015), Cusini et al. (2015), Møyner and Lie (2016)). Existing multiscale solvers use a sequence of aggressive restriction, coarse-grid correction and prolongation operators to handle low-frequency modes on the coarse grid. High-frequency errors are then resolved by employing a smoother on fine grid, which could be expensive if large number of smoothing iterations is required in the Newton-Raphson step. This is usually the case when multiscale solvers are used as a linear solver with the very small tolerance. However, both AMG and MS methods are used incompletely, i.e. 1 V cycle of AMG and one step of MS. In all cases, the goal is to construct a relatively good approximate pressure solution before computing remaining variables and applying a second stage preconditioner.

The multiscale solvers seem to be naturally scalable and can be used as a preconditioner. However, usage of the smoothers on the fine grid and solution of the coarse system form a bottleneck in constructing embarrassingly parallel algorithm. Hence, to reduce the impact of the smoothers on the fine grid, the

quality of the pressure solution after one step of MS should be of acceptable accuracy. Furthermore, the combination of a multiscale solver and the deflation method can also be constructed to improve the quality of the pressure solution Lukyanov et al. (2015).

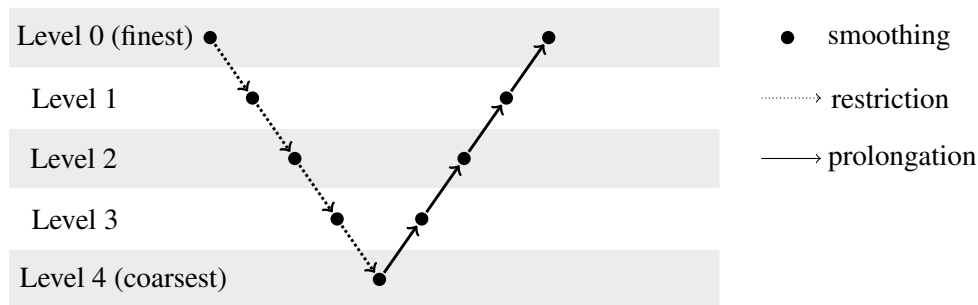
The outline of the paper is as follows. The second paragraph contains the description of the common features of the multiscale, multilevel, multigrid and deflation methods. In the following paragraph, we discuss the constructions of the deflation vectors and their connection to the multiscale methods. The fourth section presents the meshless deflation vectors and their construction in the case of two-level solution technique which can be generalized to multilevel applications. The section with numerical examples for serial and parallel runs follows this. The paper is concluded by the summary.

### Multiscale, Multilevel, Multigrid and Deflation Methods

There are a number of common and fundamental features in currently used state-of-the-art methods. Given the fine-scale system of the governing equations, (two-level and multi-level) multiscale, multilevel, (algebraic and geometric) multigrid and deflation solvers construct a coarse-scale system by applying (one or several depending on the number of coarsing levels) restriction  $\mathbf{R}$  and prolongation  $\mathbf{P}$  operators, solve the solution at coarse level, and then prolong (interpolate) it back to the original fine-scale system by using (one or several depending on the number of coarsing levels) prolongation operators  $\mathbf{P}$ . This can be applied a pre-defined number of times as

$$\mathbf{A}^{k+1} = \mathbf{R}^k \mathbf{A}^k \mathbf{P}^k \quad (1)$$

where  $k$  is the level of the appropriate step of multiscale, multilevel, multigrid, and deflation methods,  $\mathbf{A}^{k+1}$  is the matrix on the next level (for the pressure system),  $\mathbf{R}^k$ ,  $\mathbf{P}^k$  are the restriction and prolongation operators at the level  $k$ . Hence, the level  $k$  has a different meaning depending on the selected method. For example, for deflation method this corresponds to a different deflated eigenspace. Typical V-cycle scheme of five levels for (multilevel-) multiscale, multigrid, and multilevel deflated method is shown in Figure 1.



**Figure 1** Typical V-cycle scheme of five levels for (multilevel-) multiscale, multigrid, and multilevel deflated method.

Naturally, some generalization (multilevel-) multiscale, multigrid, and multilevel deflated method can be constructed using a W-cycle. It is obvious how this works in case of (multilevel-) multiscale and multigrid methods. The non-trivial multilevel deflated method can be constructed as follows. Let  $\mathbf{A}^k \in \mathbb{R}^{n \times n}$  be the non-symmetric linear system matrix at the level  $k$ . Then we define the matrix at the next level (coarse matrix). Similar to the (multilevel-) multiscale, multigrid methods, the matrix  $\mathbf{A}^{k+1} \in \mathbb{R}^{d \times d}$  is defined using either Galerkin (note that non-Galerkin projection can also be applied, see Falgout and Schroeder (2014), Lukyanov et al. (2014a)) as

$$\mathbf{A}^{k+1} = \mathbf{R}^k \mathbf{A}^k \mathbf{P}^k, \quad (2)$$

and the deflation matrices at the level  $k$  (i.e.,  $\mathbf{D}_1^k$  and  $\mathbf{D}_2^k$ ) are defined as

$$\begin{aligned}\mathbf{D}_1^k &= \mathbf{I} - \mathbf{A}^k \mathbf{P}^k \left( \mathbf{A}^{k+1} \right)^{-1} \mathbf{R}^k, \\ \mathbf{D}_2^k &= \mathbf{I} - \mathbf{P}^k \left( \mathbf{A}^{k+1} \right)^{-1} \mathbf{R}^k \mathbf{A}^k.\end{aligned}\quad (3)$$

where  $\mathbf{A}^1 = \mathbf{A}$  is the original fine scale matrix. The  $\left( \mathbf{A}^{k+1} \right)^{-1}$  is constructed to avoid a direct inversion of the matrix and expensive matrix-matrix product in the expression  $\left( \mathbf{A}^{k+1} \right)^{-1} \mathbf{R}^k$ . This is achieved by doing an user defined number pre- and a post-smoothing (e.g., Gauss-Seidel (GS) or ILU(0)) steps similar to Multigrid Solver (see, Lukyanov et al. (2014b)). Finally, the solution to the original linear system can be found using the relation:

$$x = \sum_{i=1}^k \mathbf{P}^i \left( \mathbf{A}^{i+1} \right)^{-1} \left( \mathbf{R}^i \right)^T b + \prod_{i=1}^k \mathbf{D}_2^i \hat{x} \quad (4)$$

where  $\hat{x}$  is a solution of the 'deflated system':

$$\prod_{i=1}^k \mathbf{D}_1^i \mathbf{A} \hat{x} = \prod_{i=1}^k \mathbf{D}_1^i b \quad (5)$$

The prolongation  $\mathbf{P}^i$  and restriction  $\mathbf{R}^i$  operators in deflation method are defined using the deflation matrix,  $\mathbf{Z}^i$  (by no limiting example) as follows

$$\mathbf{P}^i = \mathbf{Z}^i, \quad \mathbf{R}^i = \left( \mathbf{Z}^i \right)^T \quad (6)$$

where  $\mathbf{Z}^i$  is the the deflation matrix between levels  $i$  and  $i+1$ . Let us recall that the prolongation operator  $\mathbf{P}$  in multiscale methods such as the multiscale finite element (MSFE) and multiscale finite volume (MSFV) are constructed based on local solutions (basis functions) of the fine-scale problem with different boundary conditions. The local support for the basis functions are obtained by first imposing a coarse grid ( $\bar{\Omega}_k$ ) on the given fine-grid cells. The coarse grid operator is then constructed using Galerkin condition (1). The restriction operator, i.e., mapping fine scale to coarse scale, can be obtained by using either a finite element (MSFE) or finite volume (MSFV) methods as per terminology of the reservoir simulation community. The former employs a transpose of the prolongation operator, i.e.,

$$\mathbf{R}_{\text{FE}} = \mathbf{P}^T, \quad (7)$$

There is a clear similarity in constructing a coarse system between multiscale (MSFE, MSFV) and deflation method. The high-frequency errors at the fine scale in multiscale (MSFE, MSFV) are then resolved by employing a smoother on fine grid. The equivalent step in deflation method is the solution of the 'deflated system' defined by the equation (5). It is clear that direct solution step of the 'deflated system' can be substituted by performing smoothing iterations using either the original matrix  $\mathbf{A}$  or deflated matrix  $\left( \prod_{i=1}^k \mathbf{D}_1^i \mathbf{A} \right)$ . Hence, the primary difference between (two-level or multi-level) multiscale and (two-level or multi-level) deflation method is the selection of the restriction and prolongation operators. Multiscale methods have been developed to describe local heterogeneities within the sub-domains in the coarse system by using purely locally-supported basis functions. The deflation methods use a different strategy in constructing the restriction and prolongation operators based on the detailed analysis of the underlying spectrum of the preconditioned matrix.

In recent years, the multiscale restriction smoothed basis (MsRSB) method was introduced (see, Møyner and Lie (2016)) to overcome the geological complexities and the use of complex grid geometries required to compute compute the underlying basis functions. Below, it will be shown that the basis functions employed by MsRSB method have been extensively used in the deflation methods for various applications Tang and Vuik (2007a), Tang and Vuik (2007c), Tang and Vuik (2007b), Frank and Vuik (2001), Vuik et al. (2002), van der Linden et al. (2016)

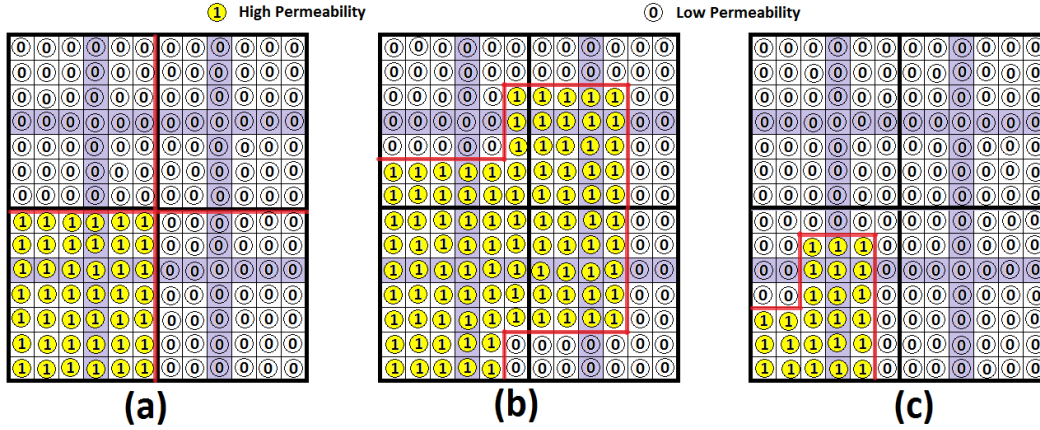


Figure 2 Subdomain (a), levelset (b) and subdomain-levelset deflation (c).

### Multiscale Restriction Smoothed Basis and Deflation Vectors

In both methods (i.e., MsRSB and Deflation), the discretized computational domain  $\Omega$  is first decomposed into  $d$  non-overlapping subdomains  $\bar{\Omega}_j$  with  $j \in \{1, \dots, d\}$ . The deflation vector  $\bar{Z}_j$  forms  $j$ -th column of the deflation operator  $\mathbf{Z}$  or initial basis functions  $\mathcal{P}^0_j$  of MsRSB method, corresponding to  $\bar{\Omega}_j$  and it is defined as (see, Frank and Vuik (2001), Tang (2008b), van der Linden (2013), van der Linden et al. (2016), Møyner and Lie (2016), Shah et al. (2016))

$$(\mathcal{P}^0_j)_i = (\bar{Z}_j)_i = \begin{cases} 1, & x_i \in \bar{\Omega}_j \\ 0, & x_i \in \Omega \setminus \bar{\Omega}_j, \end{cases} \quad (8)$$

where  $x_j$  is a fine-scale grid cell center. Based on the above definition,  $\bar{Z}_j$  and  $\mathcal{P}^0_j$  are piecewise-constant vectors or functions (equal to a constant value of one inside the corresponding coarse domain  $\bar{\Omega}_j$ ), disjoint and orthogonal. For this choice of the deflation subspace, the deflation projectors  $\mathbf{P}_1$  and  $\mathbf{P}_2$  essentially agglomerate each subdomain in a single cell. Hence, the subdomain deflation and MsRSB are closely related to domain-decomposition methods and multigrid (see, Frank and Vuik (2001)). For problems in the bubbly flow, the span of the deflation vectors (8) approximates the span of the eigenvectors corresponding to the smallest eigenvalues Tang and Vuik (2007c). In our case, as studied by Vuik et al. (2002), the subdomains can be defined using the underlying heterogeneity, e.g., a low-permeable region can be separated from the high-permeable regions and form  $d$  decompositions. As it was noted in Vermolen et al. (2004), it is better to use no overlap subdomains. This fact leads to the weighted overlap method, which mimics average and no overlap in the case of no contrasts and large contrasts, respectively. It is shown that the overlap is crucial in approximating the eigenvectors corresponding to the extreme eigenvalues.

In highly heterogeneous computational domain with large jumps in the permeability field, the subdomain-levelset deflation can be used (see, Tang and Vuik (2007c)). In this case where subdomain deflation does not take jumps into account, the subdomain-levelset deflation identifies different regions in the domain with similar properties. A simple example is given in Figure 2. The fine grid is  $12 \times 14$  with the coarse grid  $2 \times 2$  are shown in Figure 2. In each case, the values shown on the fine cells correspond to the values in the first deflation vector. In the middle and right figure, the border between the yellow circles (high permeability) and white circles (low permeability) exemplifies a sharp contrast in the matrix coefficient. The figures shows the following:

- In the left figure, subdomain deflation is used. The red solid line divides the domain into the four subdomains  $\bar{\Omega}_1, \bar{\Omega}_2, \bar{\Omega}_3$  and  $\bar{\Omega}_4$ . Each subdomain corresponds to a unique deflation vector.
- In the middle figure, levelset deflation is used. This time, the red solid line coincides with the contrast in the matrix coefficient. As a result, we get the two domains  $\bar{\Omega}_1$  and  $\bar{\Omega}_2$ .

- In the right figure, subdomain-levelset deflation is used. The subdomain division is determined using certain criteria, which in this example leads to the division red solid line between  $\bar{\Omega}_1$  and  $\bar{\Omega}_2$ . Within each subdomain, levelset deflation uses the jump between the high permeability and low permeability cells to obtain the subdomains  $\bar{\Omega}_1, \bar{\Omega}_2, \bar{\Omega}_3$  and  $\bar{\Omega}_4$ .

In this paper, the subdomain-levelset deflation concept is used. We highlight the fact that the algorithm is particularly suitable for a parallel implementation. We can apply the levelset deflation method to each subdomain (coarse cells), and append the deflation vectors with zeros for all cells outside the neighboring subdomains. Furthermore, it is important to note that deflation vectors can be constructed based on the jump in the PVT data (e.g., bubbly flow Tang and Vuik (2007c)). For example, during the polymer flooding the aqueous viscosity changes drastically in the presence of polymer. Hence, the deflation vectors can be constructed based on the location of the polymer within the reservoir.

Although we have identified a full analogy between deflated vectors  $\mathbf{Z}$  and initial basis functions  $\mathcal{P}_j^0$  of MsRSB method, the final restriction-smoothed basis functions are computed by employing a modified form of the damped-Jacobi smoothing approach:

$$\delta \mathcal{P}_k^\eta = -\omega \mathbf{D}^{-1} \mathbf{A} \mathcal{P}_k^\eta \quad (9)$$

where  $\mathbf{A}$  is the fine-scale matrix,  $\mathbf{D} = \text{diag}(\mathbf{A})$  is the diagonal part of the matrix  $\mathbf{A}$ . The final update for prolongation operator is defined as

$$\mathcal{P}_k^{\eta+1} = \mathcal{P}_k^\eta + \delta \hat{\mathcal{P}}_k^\eta \quad (10)$$

where  $\delta \hat{\mathcal{P}}_k^\eta$  is the restricted iterative increments (see, Møyner and Lie (2016), Shah et al. (2016)). To ensure that basis functions have local support, the increments  $\delta \mathcal{P}_k^\eta$  is restricted to have nonzero values only inside  $\bar{\Omega}_k$  leading to the definition of the  $\delta \hat{\mathcal{P}}_k$ . Finally, the basis functions (9)-(10) of the MsRSB method can be written in the abstract form as:

$$\mathcal{P}_k = \mathbf{M}_{MsRBS}^{\frac{1}{2}} \mathcal{P}_k^0 \quad (11)$$

where  $\mathbf{M}_{MsRBS}^{\frac{1}{2}}$  is the predefined smoothing matrix of MsRBS method defined by (9)-(10) and restriction expression (see, Møyner and Lie (2016), Shah et al. (2016)).

## Remarks

- The proposed method is similar to smoothed aggregation-based multigrid methods (see, Vanek (1992), Vanek et al. (1996))
- A deflation technique applied to a preconditioned system with preconditioner  $\mathbf{M}^{-1}$  at different level  $k$

$$\tilde{\mathbf{D}}_1^k = \mathbf{I} - \tilde{\mathbf{A}}^k \tilde{\mathbf{P}}^k \left( \tilde{\mathbf{A}}^{k+1} \right)^{-1} \tilde{\mathbf{R}}^k, \quad \tilde{\mathbf{D}}_2^k = \mathbf{I} - \tilde{\mathbf{P}}^k \left( \tilde{\mathbf{A}}^{k+1} \right)^{-1} \tilde{\mathbf{R}}^k \tilde{\mathbf{A}}^k, \quad (12)$$

$$\tilde{\mathbf{A}}^{i+1} = \tilde{\mathbf{R}}^i \tilde{\mathbf{A}}^i \tilde{\mathbf{P}}^i, \quad \tilde{\mathbf{P}}^i = \tilde{\mathbf{Z}}^i, \quad \tilde{\mathbf{R}}^i = \left( \tilde{\mathbf{Z}}^i \right)^T, \quad \mathbf{Z} = \mathbf{M}^{-\frac{1}{2}} \tilde{\mathbf{Z}}$$

is equivalent to preconditioning of a deflated system

$$\prod_{i=1}^k \mathbf{D}_1^i \mathbf{M}^{-1} \mathbf{A} \hat{x} = \prod_{i=1}^k \mathbf{D}_1^i \mathbf{M}^{-1} b, \quad x = \sum_{i=1}^k \mathbf{P}^i \left( \hat{\mathbf{A}}^{i+1} \right)^{-1} \left( \mathbf{R}^i \right)^T \left( \mathbf{M}^{-1} b \right) + \prod_{i=1}^k \mathbf{D}_2^i \hat{x} \quad (13)$$

Here  $\hat{\mathbf{A}}^{i+1} = \prod_{l=1}^i \mathbf{R}^l \left( \mathbf{M}^{-1} \mathbf{A} \right) \mathbf{P}^l$ ,  $\hat{x}$  is a solution of the preconditioned 'deflated system'. The matrix

$\tilde{\mathbf{Z}}$  is interpreted as a preconditioned deflation-subspace matrix (see, Tang (2008b) who formulated this for a two-level deflation method applied to symmetric positive definite matrix):

$$\tilde{\mathbf{Z}} = \mathbf{M}^{\frac{1}{2}} \mathbf{Z} \quad (14)$$



where  $\mathbf{M}$  is the specified preconditioner. Comparison of the equation (14) with the equation (11) and taking into account that  $\mathcal{P}_j^0 = \bar{Z}_j$  leads to the fact that two-level (preconditioned with the matrix  $\mathbf{M}_{MsRBS}^{\frac{1}{2}}$ ) deflation projection with the post smoothing used in MsRBS should give identical results with MsRBS. Commonly, the deflation matrix (matrix of basis functions)  $\mathbf{Z}$  is applied to preconditioned matrix  $\mathbf{M}^{-1}\mathbf{A}$  and, hence, the smoothing (14) is not applied since it is equivalent to have twice preconditioning of the matrix  $\mathbf{A}$ .

- Localization can be constructed using any partition of unity (PU) functions  $\{\varphi_i\}$  with  $\Omega_i = \text{supp}(\varphi_i)$  at any level  $k$  as follows

$$p = \sum_{i=1}^N \varphi_i p_i = \sum_{i=1}^N \left( \varphi_i p_{\text{smooth}}^i + \varphi_i p_{\text{jump}}^i + \varphi_i p_{\text{singular}}^i \right) \quad (15)$$

where  $p_{\text{smooth}}^i$  is the smooth part of the solution,  $p_{\text{singular}}^i$  is the part of the solution with singularity,  $p_{\text{jump}}^i$  is the part of the solution with the jumps, and functions  $\varphi_i$  play the role of “glue”.

### Meshless Deflation Vectors

The meshless deflation vectors (basis functions) utilize some advantages of the existing multiscale schemes and meshless methods. In light of this, the meshless deflation vectors (basis functions) are constructed using two sets of computational points (see, Lukyanov et al. (2014b)):

1. Fine points set  $S_F$  (e.g., cell centers of underlying mesh), i.e.  $\Omega = \text{span} \{ \bar{\Omega}_{\mathbf{X}_I, \bar{h}} / I = 1, \dots, N_F \}$  consisting of  $N_F$  patches which are interior to the support of the kernel  $\bar{W}(\mathbf{X} - \mathbf{X}_I, \bar{h})$ , i.e.  $\Omega_{\mathbf{X}, \bar{h}} = \text{supp} \bar{W}(\mathbf{X} - \xi, \bar{h})$ ,  $N_F$  is the number of points,  $h_F$  is the fine scale diameter (or smoothing length);
2. Coarse points set  $S_C$  (e.g., user defined points), i.e.  $\Omega = \text{span} \{ \bar{\Omega}_{\mathbf{X}_J, h} / J = 1, \dots, N_C \}$  consisting of  $N_C$  patches which are interior to the support of the kernel  $\bar{W}(\mathbf{X}_J - \mathbf{X}, h)$ , i.e.  $\Omega_{\mathbf{X}, h} = \text{supp} \bar{W}(\mathbf{X} - \xi, h)$ ,  $N_C < N_F$  is the number of points,  $h_C$  is the coarse scale diameter (or smoothing length);

In both cases (fine and coarse), the same cubic spline for kernel function  $\bar{W}$  was used (see, Lukyanov et al. (2014b)). Following the conventional multiscale approach, the approximation of the fine scale pressure distribution  $p_F$  using coarse set values  $p_C$  is constructed as

$$p_F(\mathbf{X}) \approx \sum_{J=1}^{N_C} V_{\xi_J} \cdot \bar{W}(\mathbf{X} - \xi_J, h_C) \cdot p_C(\xi_J) \quad (16)$$

or

$$\begin{aligned} p_F(\mathbf{X}_I) \approx & \mathbf{X}_I \cdot \left( \sum_{J=1}^{N_C} V_{\xi_J} \cdot \bar{W}(\mathbf{X}_I - \xi_J, h_C) \left[ \sum_{K=1}^{N_C} [p_C(\xi_K) - p_C(\xi_J)] \nabla^* \bar{W}(\xi_J - \xi_K, h_C) \right] \right) + \\ & + \sum_{J=1}^{N_C} \left[ V_{\xi_J} \cdot \bar{W}(\mathbf{X}_I - \xi_J, h_C) \cdot \left( p_C(\xi_J) - \xi_J \cdot \left( \sum_{K=1}^{N_C} [p_C(\xi_K) - p_C(\xi_J)] \nabla^* \bar{W}(\xi_J - \xi_K, h_C) \right) \right) \right] \end{aligned} \quad (17)$$

The scheme (17) provides a first-order consistent prolongation from a coarse set of particles into a fine set of particles, i.e. it provides an exact solution for a linear pressure distribution at both scales. In case where coarse level points do not form a subset of fine level points, the aggressive restriction is also applied following relations (zero and first-order consistent operators) such as:

$$p_C(\mathbf{X}) \approx \sum_{J=1}^{N_F} V_{\xi_J} \cdot \bar{W}(\mathbf{X} - \xi_J, h_F) \cdot p_F(\xi_J) \quad (18)$$

or

$$\begin{aligned}
p_C(\mathbf{X}_I) \approx & \mathbf{X}_I \cdot \left( \sum_{J=1}^{N_F} V_{\xi_J} \cdot \bar{\mathbf{W}}(\mathbf{X}_I - \xi_J, h_F) \left[ \sum_{K=1}^{N_F} [p_F(\xi_K) - p_F(\xi_J)] \nabla^* \bar{\mathbf{W}}(\xi_J - \xi_K, h_F) \right] \right) + \\
& + \sum_{J=1}^{N_F} \left[ V_{\xi_J} \cdot \bar{\mathbf{W}}(\mathbf{X}_I - \xi_J, h_F) \cdot \left( p_F(\xi_J) - \xi_J \cdot \left( \sum_{K=1}^{N_F} [p_F(\xi_K) - p_F(\xi_J)] \nabla^* \bar{\mathbf{W}}(\xi_J - \xi_K, h_F) \right) \right) \right] \quad (19)
\end{aligned}$$

It is important to note that relations (16), (17), (18), (19) can be written using matrix notation:

$$\begin{aligned}
\mathbf{u}_F &= \bar{\mathbf{W}}^P \cdot \mathbf{u}_C, \quad \bar{\mathbf{W}}^P : S_C \rightarrow S_F, \quad \mathbf{u}_C = \bar{\mathbf{W}}^R \cdot \mathbf{u}_F, \quad \bar{\mathbf{W}}^R : S_F \rightarrow S_C \\
\tilde{\mathbf{W}}^R &= B_W^{-1} \bar{\mathbf{W}}^R, \quad B_W = \bar{\mathbf{W}}^R \bar{\mathbf{W}}^P, \quad \tilde{\mathbf{W}}^R \bar{\mathbf{W}}^P = \mathbf{I}, \quad \tilde{\mathbf{W}}^R : S_F \rightarrow S_C \\
\tilde{\mathbf{W}}^P &= \bar{\mathbf{W}}^P B_W^{-1}, \quad B_W = \bar{\mathbf{W}}^R \bar{\mathbf{W}}^P, \quad \bar{\mathbf{W}}^R \tilde{\mathbf{W}}^P = \mathbf{I}, \quad \tilde{\mathbf{W}}^P : S_C \rightarrow S_F
\end{aligned} \quad (20)$$

where  $\bar{\mathbf{W}}^P$  and  $\tilde{\mathbf{W}}^P$  are the prolongation operators,  $\bar{\mathbf{W}}^R$  and  $\tilde{\mathbf{W}}^R$  are the restriction operators. In general, it is clear that  $\bar{\mathbf{W}}^R \neq (\bar{\mathbf{W}}^P)^T$ . As a result, the following options are available to construct deflation vectors for restriction and prolongation operators:

$$\begin{aligned}
(I) \quad \mathbf{P} &= \bar{\mathbf{W}}^P, \quad \mathbf{R} = (\bar{\mathbf{W}}^P)^T, \quad (II) \quad \mathbf{P} = \tilde{\mathbf{W}}^P, \quad \mathbf{R} = (\tilde{\mathbf{W}}^P)^T \\
(III) \quad \mathbf{P} &= \bar{\mathbf{W}}^P, \quad \mathbf{R} = \bar{\mathbf{W}}^R, \quad (IV) \quad \mathbf{P} = \bar{\mathbf{W}}^P, \quad \mathbf{R} = \tilde{\mathbf{W}}^R, \quad (V) \quad \mathbf{P} = \tilde{\mathbf{W}}^P, \quad \mathbf{R} = \bar{\mathbf{W}}^R \\
(VI) \quad \mathbf{P} &= (\bar{\mathbf{W}}^R)^T, \quad \mathbf{R} = \bar{\mathbf{W}}^R, \quad (VII) \quad \mathbf{P} = (\tilde{\mathbf{W}}^R)^T, \quad \mathbf{R} = \tilde{\mathbf{W}}^R
\end{aligned}$$

The option (I) is used in this paper. The prolongation and restriction operators in this method are based on a SPH gradient approximation (instead of solving localized flow problems) commonly used in the meshless community for thermal, viscous, and pressure projection problems. Furthermore, the smoothing length  $h_C$  can be selected to maintain only the host particle in the compact support resulting in the basis functions defined by the equation (8). Hence, the meshless basis functions provide a more flexible framework for constructing deflation vectors (or basis functions).

### Multiscale Meshless Based Method

The linear system of equations (primary focus is the pressure equations) with the matrix  $\mathbf{A}_F$  and right hand side  $b_F$  can be difficult to solve, if its condition number  $k = \|\mathbf{A}_F\| \|\mathbf{A}_F^{-1}\|$  is large. This motivates the concept of preconditioning. Let  $\mathbf{V}$  be an approximation of  $\mathbf{A}^{-1}$ , which is easy to construct and to solve for a given input. Hence, we can write a simple Richardson iteration scheme with the predefined number of application of the multilevel multiscale meshless based preconditioner:

$$[p_F]^{m+1} = [p_F]^m + \mathbf{V} \cdot (b_F - \mathbf{A}_F [p_F]^m) \quad (21)$$

where  $m$  is the iteration index,  $(p_F)^m$  is the pressure vector at the iteration  $m$ ,  $\mathbf{V}$  is the left multiscale meshless based preconditioner defined as an operational object. A meshless multiscale preconditioner is constructed as follows for a given input  $v_F$  and output  $z_F$  using a two-level case by no limiting example:

$$[z_F] = \mathbf{P} (\mathbf{A}_C)^{-1} \mathbf{R} [v_F], \quad \mathbf{A}_C = \mathbf{R} \mathbf{A}_F \mathbf{P} \quad (22)$$

$$[w_F] = [z_F] + \mathbf{S}_\gamma^{-1} \cdot ([v_F] - \mathbf{A}_F [z_F]) \quad (23)$$

where  $\mathbf{S}_\gamma^{-1}$  is the smoothing operator (e.g., Gauss-Seidel (GS) or ILU(k) or BILU(k) smoothing method or Krylov-space accelerator) applied  $\gamma$  times. The multiscale meshless based method involves setting up the restriction and prolongation operators, which can be constructed at the beginning of the simulations.

The coarse pressure system  $\mathbf{A}_C v = w$  can be solved with different strategies depending on the size of the coarse pressure matrix. The size of the coarse pressure matrix is defined by the size of the model and the choice of the coarsening factors. If the size of the coarse system is small a direct solver can be used. The iterative methods are used for the large coarse system.



## Numerical Experiments

In this section, the performance results of the fully implicit smoothed particle hydrodynamics (meshless) based multiscale method (MsMBM) are presented. The method has been prototyped in a commercial reservoir simulator and the method was compared against the two stage CPR-AMG-ILU(0) solver, which is the default solution strategy in majority of commercial simulators.

The test cases are classified by the number of grid cells, fluid models and time discretization. The following characteristics were compared between the default and proposed solutions.

- *Time steps*: the total number of time steps required to complete the simulation.
- *Non-linear iterations*: the number of non-linear iterations in the simulation.
- *Linear iterations*: the number of iterations used to solve the linear systems generated by the non-linear solver.
- *Linear solver time*: the amount of time needed to solve the linear systems.
- *CPU time*: the overall computational time required to complete the simulation.

### Test Cases

Table 1 describes all the running five test cases ranging from relatively small (not too small) models to relatively large (not too large) real life test cases. The black oil, iso-thermal and thermal compositional

Active cells	Dimensions	Fluid model	Implicitness	Number of phases	Number of active components
389557	$154 \times 90 \times 34$	Compositional Isothermal	AIM IMPES	3	13
348807	$238 \times 192 \times 114$	Black Oil Isothermal	Fully Implicit	3	3
348812	$238 \times 192 \times 114$	Compositional Isothermal	AIM IMPES	3	8
1722781	$18 \times 1126 \times 85$	Compositional Thermal with steam permitted	Fully Implicit	3	3
164945	not a 'box'	Compositional Thermal with steam permitted	Fully Implicit	3	3

**Table 1** Basic properties of the simulated test cases.

models with varying degree of heterogeneity in the reservoir grid properties are considered in this paper to test the performance of the Multiscale Meshless Based Method (MsMBM). The names of the test cases correspond to the total number of cells in the models. Table 1 contains several examples where the number of active cells is lower than the number of cells suggested by the dimensions. In those cases, the domain includes a number of inactive cells. The settings for the simulation test cases can be found in Table 2, where the parameter  $k$  represents the number of the top level GMRES accelerations with MsMBM preconditioner. The following tables and figures shows the performance results for all experimental tests including reference performance results (CPR-AMG-ILU(0)).

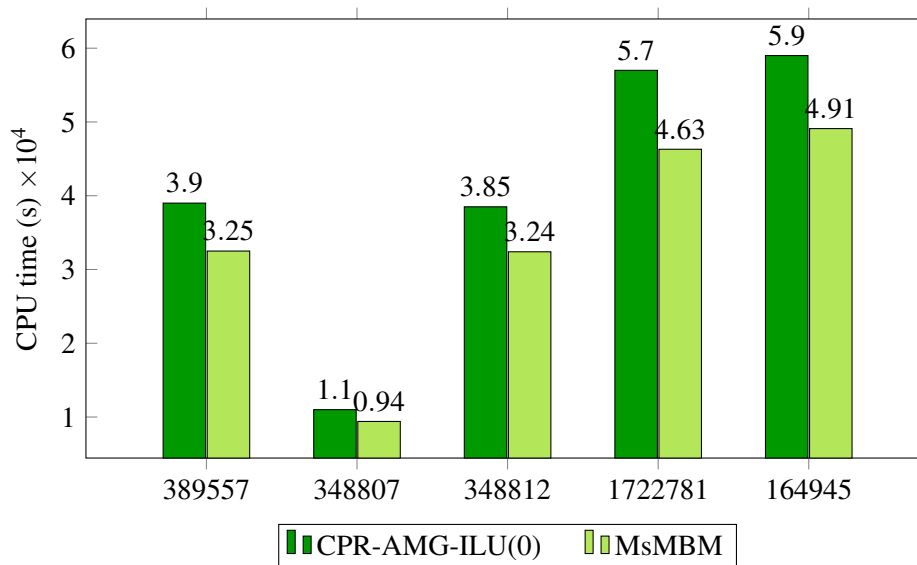
In general, Multiscale Meshless Based Method (MsMBM) does not change the number of time steps and non-linear iterations compared to the CPR-AMG-ILU(0) solution strategy in the considered simulations. This is due to second stage preconditioner ILU(0), which eliminates all the differences in the pressure solver, where AMG and MsMBM are applied. However, in all cases there is a significant effect on the number of linear iterations. This is a very common observations for multiscale solutions strategies (see,

	Pre-smoothing	Post-smoothing	MsMBM-GMRES(k)	Coarse Solver
389557	GS	GS	k = 3	AMG
348807	GS	GS	k = 3	AMG
348812	GS	GS	k = 3	AMG
1722781	GS	GS	k = 3	AMG
164945	GS	GS	k = 3	AMG

**Table 2** Experimental settings.

Lukyanov et al. (2014b), Cusini et al. (2014), Cusini et al. (2015)) which leads to a larger number of linear iterations in general. This is, of course, a subject to a number of smoothing iterations on the fine scale. This indicates that the convergence rate of the multiscale method is worse than default AMG as other components of the linear solver have not changed. However, the increase in linear iterations is acceptable given the fact that the MsMBM has a more favorable CPUs in mapping information from fine to coarse level and solving this at the coarse level. The localized construction of the multiscale basis functions has strong affect on parallel scalability of multiscale method in general. Below, the CPUs time of different cases are presented.

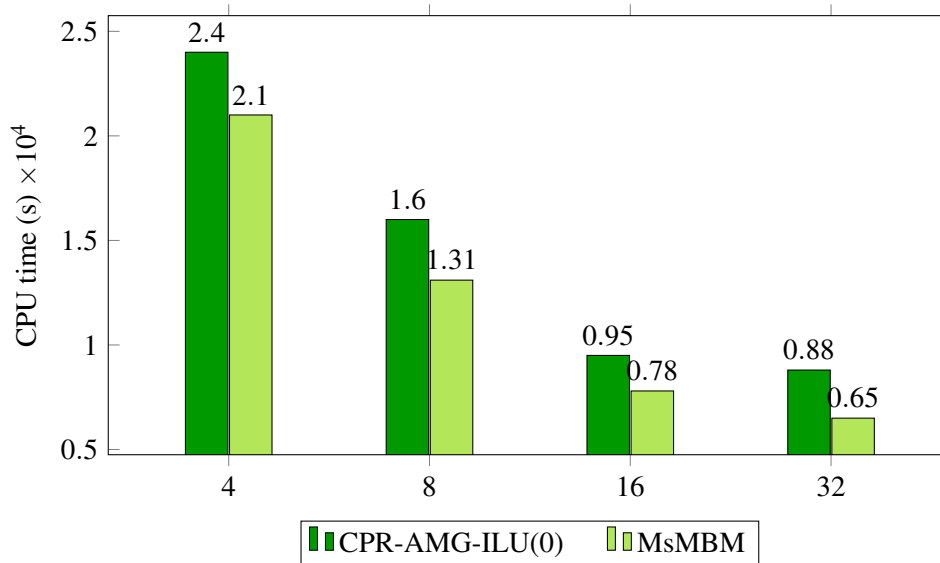
Figure 3 shows the total time (CPUs) of the linear solver stage for different cases in serial runs. The MsMDM algorithm improves the results of the default method CPR-AMG-ILU(0) method in terms of total simulation time. The total simulation time is significantly reduced in all cases. Comparing the performance results of different simulations, it is clear that moderate number of cheap smoothing iterations reduces considerably the number of linear iterations and it allows a substantial speedup. If the correct settings are chosen CPR-MsMDM-ILU(0) results to be more than 20 % faster than CPR-AMG-ILU(0) in serial runs. The simulation tests clearly shows that MsMBM leads to a substantial speedup



**Figure 3** Total time of the linear solver stage for CPR-AMG-ILU(0) and MsMDM solution strategies.

around 20% in general for serial runs. If the correct settings are chosen then the multiscale meshless based method preconditioner in the Krylov subspace iterative solvers results in a speedup relative to the reference settings (i.e., CPR-AMG-ILU(0)).

The size and complexity of modern reservoir simulation problems require vast computational resources and parallel computation. Hence, the state-of-the-art solvers, for example, the AMG preconditioner is used for parallel computing of large problems. Figure 4 shows preliminary results of the parallel scalability of the proposed multiscale method versus the default solver (i.e., CPR-AMG-ILU(0))



**Figure 4** Scalability of the total time of the simulation runs for CPR-AMG-ILU(0) and MsMDM solution strategies in the case 389557.

It is clear that further analysis of the method and its scalability is required to obtain full picture about capabilities of the proposed method.

## Conclusions

A preconditioned Krylov subspace method such as the preconditioned FGMRES method can significantly improve the convergence and robustness of a numerical simulator. This paper considers the preconditioned GMRES method with multilevel multiscale meshless based method for solving such pressure system. The solution technique proposed in this paper uses a meshless approximation method to construct a priori the deflation space (or basis functions space). The analysis of the common fundamental features of different multiscale strategies are presented.

In this paper, a parallel fully implicit smoothed particle hydrodynamics (SPH) based multiscale method for solving pressure system is presented. The prolongation and restriction operators in this method are based on a SPH gradient approximation (instead of solving localized flow problems) commonly used in the meshless community for thermal, viscous, and pressure projection problems. This method can reduce to the recently proposed MsRSB. In general, it gives more flexibility in constructing various restriction and prolongation operators.

This method has been prototyped in a commercially available simulator. This method does not require a coarse partition and, hence, can be applied to general unstructured topology of the fine scale. The SPH (or meshless) based multiscale method provides a reasonably good approximation to the pressure system and speeds up the convergence when used as a preconditioner for an iterative fine-scale solver. In addition, it exhibits expected good scalability during parallel simulations. Presented numerical results support theoretical and practical expectations from this method. Additional analysis of this method is required to identify the optimal parameters of this method.

## References

- Cao, H., Tchelepi, H., Wallis, J. and Yardumian, H. [2005] Constrained Residual Acceleration of Conjugate Residual Methods. *SPE Annual Technical Conference and Exhibition*.
- Cortinovis, D. and Jenny, P. [2014] Iterative Galerkin-Enriched Multiscale Finite-Volume Method. *J. Comput. Phys.*, **277**, 248–267.
- Cusini, M., Lukyanov, A., Natvig, J. and Hajibeygi, H. [2014] A Constrained Pressure Residual Multiscale (CPR-MS) Compositional Solver. *ECMOR XIV - 14th European Conference on the Mathematics of Oil Recovery*, EAGE.
- Cusini, M., Lukyanov, A., Natvig, J. and Hajibeygi, H. [2015] Constrained Pressure Residual Multiscale (CPR-MS) Method. *Journal of Computational Physics*, **299**, 472–486.
- Dostál, Z. [1988] Conjugate gradient method with preconditioning by projector. *International Journal of Computer Mathematics*, **23**(3-4), 315–323.
- Efendiev, Y. and Hou, T. [2009] *Multiscale Finite Element Methods: Theory and Applications*. Springer.
- Falgout, R. and Schroeder, J. [2014] Non-Galerkin Coarse Grids for Algebraic Multigrid. *SIAM Journal on Scientific Computing*, **36**(3), 309–334.
- Frank, J. and Vuik, C. [2001] On the construction of deflation-based preconditioners. *SIAM J. Sci. Comput.*, **23**(2), 442–462.
- Hajibeygi, H., Bonfigli, G., Hesse, M. and Jenny, P. [2008] Iterative multiscale finite-volume method. *J. Comput. Phys.*, **227**, 8604–8621.
- Hou, T. and Wu, X.H. [1997] A multiscale finite element method for elliptic problems in composite materials and porous media. *J. Comput. Phys.*, **134**, 169–189.
- Jenny, P., Lee, S. and Tchelepi, H. [2003] Multiscale finite-volume method for elliptic problems in subsurface flow simulation. *J. Comput. Phys.*, **187**, 47–67.
- Jönsthövel, T., van Gijzen, M., MacLachlan, S., Scarpas, A. and Vuik, C. [2011] Comparison of the deflated preconditioned conjugate gradient method and algebraic multigrid for composite materials. *Computational Mechanics*, 1–13.
- Jönsthövel, T., van Gijzen, M., Vuik, C., Kasbergen, C. and Scarpas, A. [2009] Preconditioned conjugate gradient method enhanced by deflation of rigid body modes applied to composite materials. *Computer Modeling in Engineering and Sciences*, **47**, 97–118.
- Klie, H., Wheeler, M., Stueben, K. and Clees, T. [2007] Deflation AMG solvers for highly ill-conditioned reservoir simulation problems. *SPE Reservoir Simulation Symposium*, doi:10.2118/105820-MS.
- Kozlova, A., Li, Z., Natvig, J., Watanabe, S., Zhou, Y., Bratvedt, K., and Lee, S. [2015] A real-field Multiscale Black-Oil Reservoir Simulator. *SPE Reservoir Simulation Symposium*, Society of Petroleum Engineers, Houston, Texas, USA.
- van der Linden, J. [2013] *Development of a deflation-based linear solver in petroleum reservoir simulation*. Msc Thesis TU Delft.
- van der Linden, J., Jönsthövel, T., Lukyanov, A. and Vuik, C. [2016] The parallel subdomain-levelset deflation method in reservoir simulation. *Journal of Computational Physics*, **304**, 340–358.
- Lukyanov, A., Hajibeygi, H., Natvig, J. and Bratvedt, K. [2014a] A multilevel Galerkin and Non-Galerkin, conditionally and Unconditionally Monotone Constrained Pressure Residual Multiscale Methods. *US Patent Docket No.: IS14.8805*.
- Lukyanov, A., Hajibeygi, H., Vuik, C. and Bratvedt, K. [2015] Adaptive Deflated Multiscale Solver. *US Patent Docket No. IS14.9739-US-PSP*.
- Lukyanov, A., van der Linden, J., Jonsthovel, T. and Vuik, C. [2014b] Meshless Subdomain Deflation Vectors in the Preconditioned Krylov Subspace Iterative Solvers. *ECMOR XIV - 14th European Conference on the Mathematics of Oil Recovery*, EAGE.
- Lunati, I. and Jenny, P. [2006] Multiscale finite-volume method for compressible multiphase flow in porous media. *Journal of Computational Physics*, **216**(2), 616–636.
- Lunati, I., Tyagi, M. and Lee, S. [2011] An iterative multiscale finite volume algorithm converging to the exact solution. *Journal of Computational Physics*, **230**(5), 1849–1864.
- Manea, A., Sewall, J. and Tchelepi, H. [2015] Parallel multiscale Linear Solver for Highly Detailed Reservoir Models. *SPE Reservoir Simulation Symposium*, Society of Petroleum Engineers, Houston, Texas, USA.

- Møyner, O. and Lie, K. [2016] A Multiscale Restriction-Smoothed Basis Method for High Contrast Porous Media Represented on Unstructured Grids. *Journal of Computational Physics*, **304**, 46–71.
- Nicolaidis, R. [1987] Deflation of Conjugate Gradients with Applications to Boundary Value Problems. *SIAM J. Numer. Anal.*, **24**(2), 355–365.
- Ruge, J. and Stuben, K. [1987] Algebraic multigrid (AMG). In: McCormick, S. (Ed.) *Multigrid Methods*, 3, SIAM, Philadelphia, PA, 73–130.
- Saad, Y. and Schultz, M. [1986] A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM J. Sci. Stat. Comput.*, **7**(3), 856–869.
- Shah, S., MÅyner, 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.
- Tang, J. [2008a] *Two-Level preconditioned conjugate gradient methods with applications to bubbly flow problems*. PhD Thesis TU Delft.
- Tang, J. [2008b] Two-Level Preconditioned Conjugate Gradient Methods with Applications to Bubbly Flow Problems. *PhD Thesis TU Delft*.
- Tang, J. and Vuik, C. [2007a] Acceleration of Preconditioned Krylov Solvers for Bubbly Flow Problems. In: Shi, Y., Albada, G., Dongarra, J. and Sloot, P. (Eds.) *Computational Science - ICCS 2007, Lecture Notes in Computer Science*, 4487, Springer Berlin Heidelberg, 874–881.
- Tang, J. and Vuik, C. [2007b] Efficient deflation methods applied to 3-D bubbly flow problems. *Electronic Transactions on Numerical Analysis*, **26**, 330–349.
- Tang, J. and Vuik, C. [2007c] New variants of deflation techniques for pressure correction in bubbly flow problems. *Journal of Numerical Analysis, Industrial and Applied Mathematics*, **2**, 227–249.
- Tene, M., Hajibeygi, H. and Y. Wang, H.T. [2014] Compressible Algebraic Multiscale Solver (CAMS). *Proceedings of the 14th European Conference on the Mathematics of Oil Recovery (ECMOR), Catania, Sicily, Italy*.
- Vanek, P. [1992] Acceleration of convergence of a two-level algorithm by smoothing transfer operator. *Appl. Math.*, **37**, 265–274.
- Vanek, P., Mandel, J. and Brezina, M. [1996] Algebraic multigrid by smoothed aggregation for second and fourth order elliptic problems. *Computing*, **56**(2), 179–196.
- Vermolen, F., Vuik, C. and Segal, A. [2004] Deflation in preconditioned conjugate gradient methods for finite element problems. In M. Křížek, P. Neittaanmäki, R. Glowinski, and S. Korotov, editors, *Conjugate Gradient and Finite Element Methods*. Springer, Berlin, 2004, 103–129.
- Vuik, C., Segal, A., el Yaakoubi, L. and Dufour, E. [2002] A Comparison of Various Deflation Vectors Applied to Elliptic Problems with Discontinuous Coefficients. *Appl. Numer. Math.*, **41**(1), 219–233.
- Wallis, J. [1983] Incomplete Gaussian Elimination as a Preconditioning for Generalized Conjugate Gradient Acceleration. *SPE Reservoir Simulation Symposium*, Society of Petroleum Engineers, Houston, Texas, USA.
- Wallis, J. [1985] Constrained residual acceleration of conjugate residual methods. *SPE Reservoir Simulation Symposium*, (1), doi:10.2118/13536-MS.
- Wang, Y., Hajibeygi, H. and Tchelepi, H. [2014] Algebraic multiscale solver for flow in heterogeneous porous media. *Journal of Computational Physics*, **259**, 284–303.
- Zhou, H. and Tchelepi, H. [2012] Two-stage algebraic multiscale linear solver for highly heterogeneous reservoir models. *SPE Journal*, **17**(2), 523–539.