Journal of Computational Physics 238 (2013) 141–153

Contents lists available at SciVerse ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

Auxiliary variables for 3D multiscale simulations in heterogeneous porous media

Andreas Sandvin [a,b], Eirik Keilegavlen [a,*], Jan M. Nordbotten [a]

[a] Department of Mathematics, University of Bergen, Johannes Brunsgate 12, 5008 Bergen, Norway
[b] Centre for Integrated Petroleum Research, University of Bergen, Allegaten 41, 5007 Bergen, Norway

ARTICLE INFO

Article history:
Received 9 November 2011
Received in revised form 21 November 2012
Accepted 4 December 2012
Available online 29 December 2012

Keywords:
Multiscale methods
Control volume methods
Porous media
Preconditioning
Domain decomposition

ABSTRACT

The multiscale control-volume methods for solving problems involving flow in porous media have gained much interest during the last decade. Recasting these methods in an algebraic framework allows one to consider them as preconditioners for iterative solvers. Despite intense research on the 2D formulation, few results have been shown for 3D, where indeed the performance of multiscale methods deteriorates. The interpretation of multiscale methods as vertex based domain decomposition methods, which are non-scalable for 3D domain decomposition problems, allows us to understand this loss of performance.

We propose a generalized framework based on auxiliary variables on the coarse scale. These are enrichments of the coarse scale, which can be selected to improve the interpolation onto the fine scale. Where the existing coarse scale basis functions are designed to capture local sub-scale heterogeneities, the auxiliary variables are aimed at better capturing non-local effects resulting from non-linear behavior of the pressure field. The auxiliary coarse nodes fits into the framework of mass-conservative domain-decomposition (MCDD) preconditioners, allowing us to construct, as special cases, both the traditional (vertex based) multiscale methods as well as their wire basket generalization.

1. Introduction

Geological porous media are typically characterized as heterogeneous at virtually every scale. This reflects the process by which geological formations are created, where natural sedimentation processes spanning kilometers horizontally and millennia in time lead to composite materials that are intrinsically complex in structure. Compounding the difficulties introduced by multiscale geological parameterizations are the strongly non-linear equations that describe multi-phase flow in porous media. These equations lead to challenges that are manifested in discontinuous solutions as well as both gravitational and viscous instabilities. Such phenomena are frequently best understood as multiscale in nature. As a consequence of the complexity in modeling multi-flow in geological porous media, virtually every text-book on the subject address issues of scale. We refer to [1] for classical examples.

Two main avenues are typically followed when confronting multiscale phenomena. The most classical approach, multiscale modeling, is to manipulate equations defined at a finest, verified scale, and attempt to derive effective equations valid on coarser scales. These equations are typically stated for derived variables. These derived variables broadly fall into three categories: conserved (extensive) quantities, auxiliary (intensive) state variables, and problem specific variables. This final category of variables may be unique to the problem, or to the coarser scales, and can be interpreted to represent emerging properties of the system. In some cases these emerging properties are parameterizations of what would otherwise be seen as

* Corresponding author.
E-mail addresses: Andreas.Sandvin@uni.no (A. Sandvin), Eirik.Keilegavlen@uib.no (E. Keilegavlen), Jan.Nordbotten@math.uib.no (J.M. Nordbotten).

0021-9991/$ - see front matter © 2012 Elsevier Inc. All rights reserved.
http://dx.doi.org/10.1016/j.jcp.2012.12.016

hysteretic, or non-unique, behavior. In the context of multi-phase flow in porous media, component masses are conserved at all scales, pressure is an intensive state variable at all scales, and finally various parameterizations of hysteresis or dynamical behavior are introduced to make the models appropriate in practice [2]. This classical approach has seen several formalizations in recent years, among the most instructive of which is the Heterogeneous Multiscale Method [3].

A more recent approach to handling multiscale characteristics is through adaption of the numerical methods themselves. Classically introduced as generalized finite elements by Babuska et al. [4], it was first made into a useful concept through the residual-free bubble methods [5], where multiscale features of the solution can be handled. Later, this concept was also applied to multiscale coefficients, in what is termed multiscale finite element and multiscale finite volume methods (see [6] for an introduction). While multiscale numerical methods have shown good properties on academic problems, they often fail to live up to their promise on real problems [7]. By exploiting the link between multiscale numerical methods and domain decomposition (DD), multiscale control volume methods can be framed in an iterative setting which greatly increases the potential for robust implementations [8]. However, an improved multiscale representation without iterations is still the ultimate goal.

In this paper, we propose to enhance the common understanding of multiscale numerical discretizations through an analogy to multiscale modeling. In particular, as multiscale control volume methods inherently discretize conserved quantities, it is natural to ask if the discrete approximation, like its modeling counterpart, can be enhanced through introducing problem-specific additional variables. We term these additional variables auxiliary, and the remainder of the paper is devoted to developing and verifying this concept. In particular we consider the issue of assigning boundary conditions to the local problems based on the state in the coarse variables. This poses challenges for multiscale numerical methods, especially in the presence of long correlation pathways that render non-local dependence of the solution. The problem is difficult already in two spatial dimensions, where the state in the coarse variables must be mapped onto a 1D boundary. Strategies proposed to remedy the situation include oversampling [9,10], utilizing global information [10,11] and using specialized boundary conditions [7,12]. The situation becomes worse in three spatial dimensions, since a mapping to a 2D boundary is needed. In this paper, auxiliary coarse variables are used to address these challenges. By exploiting links between multiscale control volume methods and domain decomposition, the auxiliary variables can easily be introduced in the linear solver. We consider grids with relatively few primary coarse variables (corresponding to aggressive coarsening), and enhance the coarse space by auxiliary variables. Thus the number of internal boundaries decreases, while there is enough degrees of freedom in the coarse space to capture details in the solution. Our numerical experiments involve model problems as well as industrial benchmark data. The results show that auxiliary coarse variables can improve the performance of the linear solver considerably.

The rest of the paper is structured as follows: in Section 2, multiscale methods for three-dimensional problems are discussed and difficulties are pointed out. A multiscale linear solver is introduced in Section 3, and the extension to coarse spaces is introduced in Section 4. Simulation results are presented in Section 5, and the paper is concluded in Section 6.

2. Challenges of 3D multiscale elliptic problems

In this study we consider the following elliptic problem for flow in three dimensional porous media,

$$-\nabla \cdot (\mathbf{K}\nabla u) = q, \tag{1}$$

where $\mathbf{K}$ is the permeability of the medium, u is the potential and q represents the source terms of the system. The heterogeneous structure of porous rocks is reflected in the permeability $\mathbf{K}$, which can vary by several orders of magnitude on different scales. It is the variation of this parameter which represents the major challenge, and has been the main focus of the multiscale methods for problems involving flow in porous media. Hou and Wu [13] showed that the sub-scale information of the elliptic operator can be captured within a few coarse-scale basis functions, which increases the accuracy of the recovered fine-scale solution. Several multiscale numerical methods have later been developed for the capturing of sub-scale information into local basis-functions. We refer to [6] for an overview of these methods.

A special focus of this paper will be on problems involving long correlation lengths of the parameter $\mathbf{K}$, e.g. fractures, faults and channels which occupy several coarse-scale grid blocks. The discretization of such problems are particularly difficult to upscale, and local iterations are required to guarantee accurate solutions [14]. Due to the difficulties involved, the primary focus of previous works has been the 2D problem. In the remainder of this section we will briefly discuss some of the existing challenges of multiscale numerical methods, and highlight some of the main challenges of extending these methods to 3D simulations.

2.1. Multiscale numerics

In general porous media there are rarely only two or a couple of distinct scales, but rather a continuum of physical scales which needs to be taken into account. However, for practical purposes we need to define a finest (geological) scale for the discretization of our problem. Usually the fine-scale discretization leads to a large coupled problem which is extensive and often too computationally expensive to solve. For multiscale methods, one or a couple of coarse scales are added to speed up the calculation of a fine-scale conservative solution.

The solution space for each coarse scale is spanned by local coarse-scale basis functions. This is referred to as the coarse space. As for upscaling, the coarse scale equations for the multiscale method are stated for conserved variables, thus the multiscale methods can be categorized as upscaling methods. In addition, through the construction of coarse-scale basis functions, the multiscale methods can be applied as multilevel DD-preconditioners. In this paper we will consider the multiscale control volume methods, which can be written as DD-preconditioners in the framework of mass conservative domain decomposition (MCDD) [8].

While the multiscale methods are designed to capture local heterogeneities within the sub-domains, one of the difficulties has been to include non-local and global information, which may influence the solution locally. An oversampling strategy was proposed by Hou et al. [9] to include non-local information into the calculation of the local basis-function, thus reducing the resonance effect of the intermediate scale heterogeneities. Adaptivity w.r.t the size and structure of these oversampling regions has further improved the accuracy and limited the number of additional computations [15]. The use of global information has also been considered [10,11]. Finally, local iterations seem to be inevitable when constructing robust implementations of the multiscale methods.

2.2. Extension to 3D

Despite intense research on challenging problems in 2D, few results have been reported for problems in 3D. The capturing of sub-scale information naturally becomes more challenging for three dimensional flow. However, the extension from 2D to 3D problems also introduces new challenges w.r.t. the construction of the coarse-scale basis functions.

Concerning upscaling, the third dimension results in a larger gap between the numerical scales, where the number of fine cells per coarse cell grows as n^{dim}. Any aggressive coarsening strategy is obviously more challenging for 3D problems. The boundary conditions for the local elliptic problems are difficult to approximate. The localization approximations embedded in the multiscale simulations are usually lower-dimensional approximations of the flow on the sub-domain faces and edges and the errors due to these approximations become more severe for three dimensional flow.

From a domain decomposition perspective, it is well known that the extension of the coarse-scale basis functions from 2D to 3D problems is non-trivial. The multiscale methods are in general vertex-based (VB) methods (Fig. 1(a)), meaning that the coarse-scale degrees of freedom (dof) are associated with the vertices of the local sub-domains. For these methods, the standard piecewise linear interpolation is not robust, and do not result in scalable preconditioners for 3D-problems [16]. While the 2D coarse scale solution is mapped directly onto the boundary, the 3D coarse-scale solution, defined on vertices, needs to be mapped in two stages. The solution is first interpolated onto the edge dof, and secondly onto the face dof. The harmonic basis functions of the multiscale methods have shown to be more robust than the usual piecewise linear basis functions, w.r.t. problems involving sub-scale heterogeneities in 2D [17], however, this has not been verified for 3D problems. Moreover, in the homogeneous limit, harmonic basis functions degenerate to the usual piecewise linear basis functions, which are not scalable with the size of the problem.

In domain decomposition, various extensions of the coarse space have been proposed to retain the approximation properties obtained in 2D simulations. Of special interest to this paper is the wire basket (WB) method [18], where both the vertex and edge dof belong to the coarse space, see Fig. 1(b). Thus, we only have one interpolation from the coarse-scale dof (wire basket) onto the boundary dof, which is similar to VB-methods for 2D-problems. Other extensions of the coarse space involves defining special interpolations on the sub-domain faces [16,19]. Such methods will not be discussed in this paper. Instead we will show how the traditional VB-methods with standard interpolations can be improved through the use of auxiliary coarse variables.

While the WB-method is attractive w.r.t. convergence of the fine-scale solution, it uses a large static coarse space resulting in a much larger and more dense numerical scheme on the coarse scale. In many cases, it is not necessary to use all these

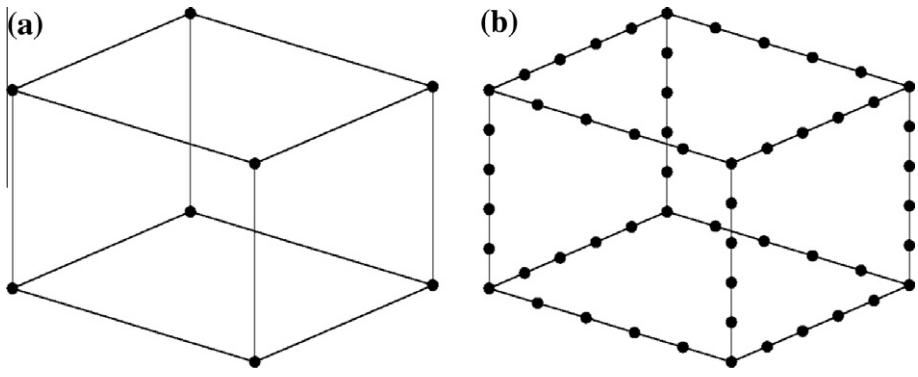

Fig. 1. Figure (a) and (b) shows the vertex cells and wire basket cells, respectively.

dof related to the wire basket. Thus, by discarding the less important dof on the coarse scale, we may save many unnecessary computations.

In this paper, we develop a generalized framework for multiscale simulation by introducing *auxiliary variables* to the coarse space. The auxiliary variables are strategic sampling points of the coarse scale and they are meant as a supplement to the existing conserved variables on the vertices. Thus, it can be seen as a generalization of the existing vertex-based multiscale control volume methods. Furthermore, we observe that the WB-distribution of coarse variables becomes a special choice of the auxiliary coarse variables. The auxiliary variables fit naturally into the framework of MCDD, in which case they represent an additional flexibility with respect to multiscale simulation and multiscale numerical modeling. However, before we can introduce the auxiliary coarse variables, we need to formulate the domain decomposition framework.

3. MCDD

The discretization of Eq. (1) results in a sparse system of linear equations,

$$Au = b, \tag{2}$$

where the system matrix A is in general large. Hence, a direct calculation of Eq. (2) is typically very time-consuming. In reservoir simulations, several linear systems on this form must be solved within each time-step, in which case inexact solvers are always used. We seek a fast inexact solver for approximating the solution of the fine-scale problem (2).

3.1. Mass conservative coarse-scale operator

Mass conservation is an essential property for reservoir simulation and fluid flow. In our case the equation is discretised into a set a fine-scale control-volumes ω_i, resulting in a linear system of equations which is mass conservative on the fine-scale. Each row of the linear system (2) will represent the mass balance equation over ω_i.

For the multiscale methods, the coarse model is defined on a primal coarse grid Ω, where each cell Ω_i is a collection of fine-scale control volumes and the coarse-scale interfaces coincide with interfaces on the fine-scale as seen in Fig. 2. If the fluxes on the coarse scale are mass conservative, it is possible to recalculate a mass conservative flux field on the fine-scale by a pre-processing of the fine-scale system (2) [20].

If all the equations corresponding to the fine-cells contained within a primal coarse cell Ω_i are added, the resulting equation will represent mass balance for Ω_i. This coarse scale equation is substituted for the row of the coarse cell (centermost cell of Ω_i). More precisely, we can write the pre-processed fine-scale system of equations as

$$Cu = p, \tag{3}$$

where

$$C = \sum_{\Omega_i} (R_i)^T M_i R_i A \quad \text{and} \quad p = \sum_{\Omega_i} (R_i)^T M_i R_i b.$$

Here, M_i is the integration matrix, while R_i represents the usual restriction matrix, consisting of zeros and ones, such that, acting on the global dof, R_i picks out the degrees of freedom corresponding to Ω_i. The integration matrix M_i is written as

$$M_i = I + e_{iV}(\underline{1} - e_{iV})^T, \tag{4}$$

where I is the identity matrix, e_{iV} is the unit vector identifying the row of the vertex cell and $\underline{1}$ is the vector entirely filled with ones.

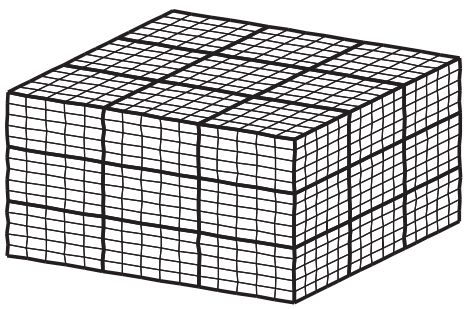

Fig. 2. The figure shows the multiscale grid. Here, the bold faces indicate the primal coarse grid, constructed on top of an underlying fine-scale grid.

3.2. Schur complement system

Several domain decomposition techniques coupled with various iterative schemes have been developed. For the majority of cases, the domain decomposition techniques are applied as preconditioners, where the iterations are accelerated by the use of Krylov subspace iterative methods. In this paper we will focus on the two-level additive Schwartz MCDD preconditioner, applied to Eq. (3). For information about other DD preconditioners, see e.g. [16,21,22]. The preconditioner will be accelerated using GMRES [23].

To formulate the MCDD preconditioner, we need to introduce the dual coarse grid Ω', consisting of dual coarse cells Ω'_i. The dual coarse grid is defined such that the centers of the coarse cells Ω_i are located at the vertices of Ω'. Thus the coarse scale conservation of mass is associated with the vertices of Ω'. The fine-scale cells joining the vertex cells belong to the boundary of the dual-grid. The dual coarse-grid Ω' is a corner-cell grid and represents a non-overlapping decomposition of the global domain. For simplicity, we will refer to Ω'_i as a sub-domain. For each sub-domain, the fine-scale dof corresponding to internal cells are denoted as internal cells, while the fine-scale dof located on the sub-domain boundaries are denoted as boundary cells. The boundary cells are further sub-divided into face cells, edge cells and vertex cells, see Fig. 3. While the internal cells are localized on each sub-domain, the boundary cells are shared by neighboring sub-domains and are globally coupled.

By rearranging the system of equations into those dof corresponding to internal and boundary unknowns we can rewrite Eq. (3) as

$$\begin{bmatrix} C_{II} & C_{IB} \\ C_{BI} & C_{BB} \end{bmatrix} \begin{bmatrix} u_I \\ u_B \end{bmatrix} = \begin{bmatrix} p_I \\ p_B \end{bmatrix}. \tag{5}$$

The internal cells belonging to different sub-domains are now decoupled, and the block-diagonal matrix C_{II} can formally be inverted directly on each sub-domain Ω'_i. The Schur complement system is found by eliminating these internal cells, resulting in a system for the boundary cells only:

$$Su_B = g, \tag{6}$$

where

$$S = C_{BB} - C_{BI}C_{II}^{-1}C_{IB}, \tag{7}$$

and

$$g = p_B - C_{BI}C_{II}^{-1}p_I.$$

We can further split the Schur complement matrix into cells corresponding to the face (F), edge (E) and vertex (V):

$$S = \begin{bmatrix} S_{FF} & S_{FE} & S_{FV} \\ S_{EF} & S_{EE} & S_{EV} \\ S_{VF} & S_{VE} & S_{VV} \end{bmatrix}. \tag{8}$$

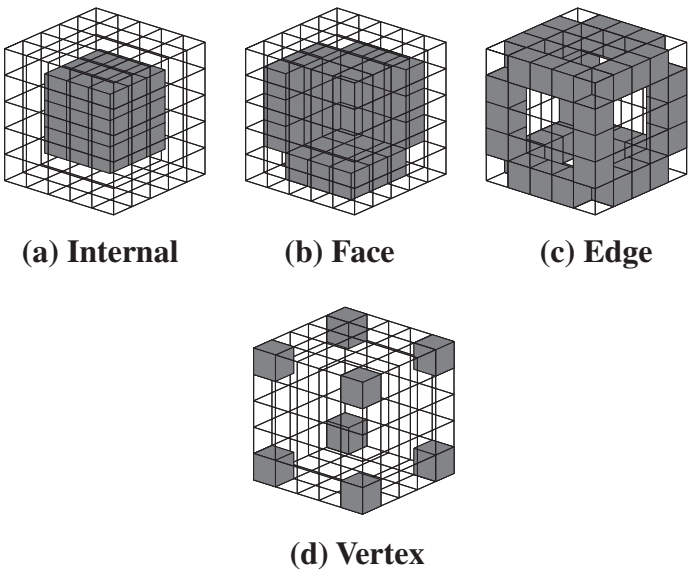

Fig. 3. An illustration of the internal, face, edge and vertex cells used in the domain decomposition method. For clarity of the visualization, only some of the face cells are shown.

Written on this form, the Schur complement S still has a block diagonal structure, but all the blocks are now full. An explicit construction of S will in fact require the same amount of work as solving the entire fine-scale problem (5). Hence, the various Schur complement matrices must be approximated by sparse matrices, such that we can decompose the Schur problem. This gives us a MCDD or Schur complement preconditioner. A special requirement for the MCDD preconditioner is that the equation for the vertices (conserved variables on the coarse scale) must be solved exactly, providing a mass conservative coarse scale solution.

3.3. Interface approximations

The accuracy of the multiscale methods depends highly on the choice of boundary conditions for the local elliptic problems. A successful choice of boundary conditions may result in an accurate representation of the coarse scale problem and a good first approximation to the fine-scale solution. However, within an iterative setting, an accurate representation of the boundary conditions only affects the first iteration; after the first iteration, the residual does not represent any actual physics but should be regarded as noise. Thus, the best multiscale approximation does not necessarily give the most efficient preconditioner [7].

The most common interface approximation for the multiscale control volume methods is similar to the tangential component (TC) approximation for DD-preconditioners. The TC-approximation of the Schur matrix S (see Eq. (7)), can be written as

$$\hat{S} = C_{BB}^T, \tag{9}$$

where C_{BB}^T is denoted the tangential part of the C_{BB}. While C_{BB} is a diagonally dominant matrix, C_{BB}^T has reduced diagonal elements such that the row-sum equals to zero, i.e. $C_{BB}^T \cdot \underline{1} = 0$. The tangential part of C_{BB} arises in the multiscale methods as a lower dimensional discretization (in the "tangential" direction), also called the reduced boundary condition. This is in general only possible for regular Cartesian grids, however the TC-approximation is also valid for general grids [7]. We can write the TC-approximation for the Schur complement (8) on matrix form,

$$\hat{S} = \begin{bmatrix} C_{FF}^T & C_{FE} & 0 \\ 0 & C_{EE}^T & C_{EV} \\ S_{VF} & S_{VE} & S_{VV} \end{bmatrix}. \tag{10}$$

While S in general is a dense and globally coupled matrix, $\hat{S}$ is sparse and locally decoupled.

3.4. The coarse scale operator

For multiscale methods, the coarse cells represent integrated and conserved quantities on Ω. We will denote the coarse cells as C. Normally they are spanned by the vertex cells, in which case we write $C = V$. By using Eq. (6) and (10) we can write up the equations for the face, edge and vertex cells:

$$\begin{aligned} C_{FF}^T u_F &= g_F - C_{FE} u_E, \\ C_{EE}^T u_E &= g_E - C_{EV} u_V, \\ A_V u_C &= b_V, \end{aligned} \tag{11}$$

where

$$A_V = [\, S_{VF} \quad S_{VE} \quad S_{VV} \,] \begin{bmatrix} \left(C_{FF}^T\right)^{-1} C_{FE} \left(C_{EE}^T\right)^{-1} C_{EV} \\ -\left(C_{EE}^T\right)^{-1} C_{EV} \\ I \end{bmatrix} \tag{12}$$

is the coarse-scale operator and

$$b_V = g_V + (S_{VF}(C_{FF}^T)^{-1} C_{FE} - S_{VE}) g_E - S_{VF} g_F \tag{13}$$

is the corresponding right-hand side vector on the coarse scale.

3.5. Remark on implementation

The terminology used to describe the linear solver is based on domain decomposition literature. This point of view convenient, since it highlights the similarity between the auxiliary variables introduced in the next section and traditional wire basket domain decomposition methods. The domain decomposition framework can be viewed as a bottom-up approach in that the coarse linear system is constructed by a series of approximated Schur-complement reductions of the fine-scale equations. An alternative top-down approach is offered by the family of multiscale linear solvers, wherein the link between

the coarse and fine linear system represented by basis functions that are associated with each coarse degree of freedom [6]. The coarse approximation can be formulated in terms of both approaches. We have found the top-down approach most convenient both in terms of implementation and in obtaining a unified framework for linear solvers and upscaling, where the coarse system can be considered an independent, coarse discretization of the continuous problem [24].

4. Auxiliary coarse variables

Due to the more aggressive coarsening that necessarily follows from considering higher-dimensional problems, the multiscale methods become increasingly sensitive to the approximations introduced to the Schur complements. This is analogous to the situation that arises for traditional static condensation type methods, where in three spatial dimensions a discussion between vertex-based and wire basket methods arises. In this section, we introduce a framework for addressing these problems for multiscale preconditioners, in which the coarse space can be enriched by what we term auxiliary coarse variables.

4.1. Generalized preconditioner

Originally, the multiscale grid has been structured in a hierarchical way, where the coarse cells are fixed and represent coarse scale volumes or elements, on which mass conservation is satisfied. Indirectly, the coarse scale equation represents a special mass conservative discretization on a coarse scale grid.

The MCDD preconditioner satisfies the same hierarchical structure, where the vertex cells preserve the coarse-scale mass conservation, however, the MCDD framework is not restricted to those coarse scale degrees of freedom. An extension of the coarse-scale degrees of freedom has been studied in domain decomposition, where certain configurations of the coarse space can improve the interpolation of the coarse scale solution and give scalable and more robust preconditioners; see e.g. [25,26] for more information. This is specially important in 3D, where the usual vertex-based domain decomposition method, like most multiscale numerical methods, fail to be scalable for linear interpolations on the homogeneous problem. To address this shortcoming we generalize the multiscale iterative methods by introducing auxiliary variables on the coarse scale. We define the coarse cells as $C = [X \quad V]$, where X represents the non-conserved auxiliary variables and V is the conserved variables on the vertices. Any fine-scale cell can in general be chosen as an auxiliary coarse variable; we have found placement along the dual edges to be a robust choice, see Fig. 4 for illustrations. If $X = E$ we have the wire basket distribution of the coarse cells, which can be used to build preconditioners that are provably scalable [18]. On the other hand, if X is empty, we retain the VB-method.

The general Schur complement preconditioner can now be written

$$B = \hat{S}^{-1} = \left\{ \begin{bmatrix} C_{FF}^T & C_{FE} & C_{FC}|_{(C=E)} \\ 0 & C_{EE}^T & C_{EC} \\ S_{CF} & S_{CE} & S_{CC} \end{bmatrix} \right\}^{-1}, \tag{14}$$

where the approximate Schur complement matrix $\hat{S}$, is on the same form as (10). We observe that if X contains any edge cells, these will no longer be represented in the set E and connections to these cells will be stored in the third block column of (14). In the case of X being empty, the preconditioner (14) is similar to the multiscale control-volume preconditioner using vertex dof on the coarse scale (see Eq. (10)). In the special case of $X = E$, we can simplify the Schur complement preconditioner, writing

$$B_W = \left\{ \begin{bmatrix} C_{FF}^T & [C_{FE} \quad 0] \\ S_{WF} & S_{WW} \end{bmatrix} \right\}^{-1},$$

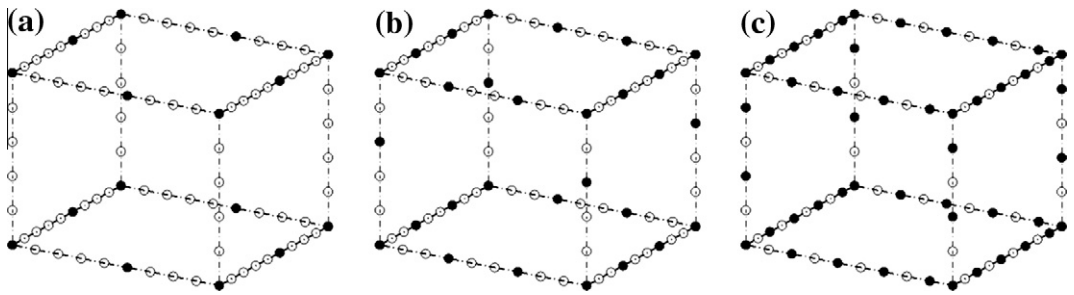

Fig. 4. Figure (a)–(c) shows a coarse cell with three sets of auxiliary coarse cells.

where $W = [E \quad V]$ is the wire basket. The auxiliary coarse variables (ACVs) X, as applied in the framework of the Schur complement preconditioners, thus generalizes multiscale methods to include enriched coarse spaces. In general, adaptivity with respect to the selection of coarse cells is not new. In algebraic multigrid methods, the coarse scale dof are algebraically chosen within the iterative method. Several multilevel methods monitor the residual, and chooses the most inaccurate cells to be corrected for on a "coarse" scale. In this setting, the coarse scale is nothing more than a bounded set of the fine-scale. The idea of extending the coarse space are also considered in [27,28] in the setting of multiscale finite element methods.

4.2. Properties of the auxiliary coarse variables

The auxiliary variables give us the flexibility of including additional fine-scale information, which might have large influence on the coarse scale. In principle, we now have the freedom of designing the coarse scale that we want, like in the case of numerical multiscale modeling. In this framework, the coarse cells represent distributed sampling points of the fine-scale problem, and the distribution of these points are important for the accuracy of the method.

We envision at least three strategies for the use of auxiliary coarse variables:

- *Parameter based*. The auxiliary coarse variables can be applied by means of refinement of the coarse space. In particular, this strategy should be considered in regions with strong contrasts in permeability.
- *Source based*. The inclusion of auxiliary coarse cells could also be motivated by the right hand side of the problem. In particular, auxiliary cells may be placed according to source terms of the system, normally represented by non-zero right-hand side elements. By placing an auxiliary coarse variable on the position of a Dirichlet well, one basis function will capture the fine-scale pressure in the vicinity of that well. This is analogous to constructing additional well-basis functions [29], which is normally performed for the MSFV method. The support for the additional basis function corresponding to the well is here limited to the corresponding sub-domain of the well.
- *Algebraic construction*. A third approach is to choose the auxiliary variables algebraically within an iterative procedure based on error estimates. The auxiliary variables are dynamic, in the sense that we can include or discard these variables at any stage of an iterative procedure. In any case, we only include or discard single basis-functions, which by construction are independent. This will introduce another adaptivity with respect to solving large linear systems effectively.

The placement of the additional coarse cells on the sub-domain boundaries is also analogous to multiscale methods with polygonal coarse elements. For each additional coarse cell on the sub-domain, we get one additional basis function. In the simplified case of linear basis functions, we observe that the extra auxiliary cells on the boundary will result in piecewise linear functions on the boundary. In the limit, as all the boundary cells are filled with auxiliary coarse variables, every cell is treated exactly and the true solution is captured. Even though this strategy is aimed at capturing non-linear and higher order functions of the solution, this is not similar to higher order multiscale methods, in which case we would need to incorporate higher order interpolations between the added coarse cells. In this paper we want to move the discussion away from only focusing on improved interpolations or boundary conditions. Here we apply the same boundary conditions everywhere, but in areas where the boundary conditions fail to capture the fine-scale physics, we show that enrichment of the coarse space may be applied directly or within an iterative procedure to improve the sub-scale capturing and convergence of the fine-scale solution.

4.3. Computational cost

The motivation for introducing auxiliary variables is to enrich the coarse space such that the preconditioner becomes more efficient in terms of the number of iterations needed for a sufficient reduction of the residual. The significant parameter to be measured is however the overall computational cost of the linear solver. This will be highly dependent on issues such as the degree of parallelization and available memory; for instance, for problems where many iterations are needed it may be more effective e.g. to factorize the coarse scale matrix if one can afford to store the factorization. The final cost comparison between refinement of the coarse space and addition of auxiliary variables therefore comes down to implementation and hardware; we chose to use the total number of coarse variables as a proxy for computational cost in the result section presented next. Nevertheless, some theoretical considerations on the overall cost is presented here.

The multiscale method has three components with significant computational cost: the construction of restriction and prolongation operators (basis functions), solving local subproblems on edges, faces and the interior of subdomains and solving the coarse system. The first component is computed prior to the iterations, while the two latter must be performed for all iterations. We will discuss the cost of each component in some detail.

Let the number of primary and auxiliary coarse variables be denoted by N_C and N_A, respectively. The main cost of solving local problems is related to the interior of the subdomains (represented by C_{II}^{-1} in Eq. (7)), and we denote the cost of this by M. Then, computing basis functions for the primary coarse cells on a 3D Cartesian coarse grid is dominated by the term $8MN_C$, since the basis function will have support in 8 subdomains. If the auxiliary coarse variables are located on edges with four adjacent subdomains, the added cost of computing basis functions is $4MN_A$. There will be as many subdomains as there are coarse nodes, thus solving local subproblems within the iterations have a cost of MN_c for each iteration. The cost of an individual iteration is not affected by the additional coarse degrees of freedom, but the overall cost of this component is

reduced by $MN_c\Delta N_{iter}$, where ΔN_{iter} is the decrease in the number of iterations due to a richer coarse space. Finally, the size of the coarse system is increased form N_C to $N_C + N_A$ when auxiliary coarse variables are added, with an associated increase in cost. In this context, it should also be mentioned that it is not necessary to solve the coarse system exactly in each iteration, in particular the auxiliary coarse variables can be approximated without loosing the conservation property on the coarse scale.

Finally, if time-dependent problems such as multi-phase flow in porous media is considered, the total number of iterations needed throughout the simulation increases, while the coarse operator must only be updated in small parts of the domain for each time step. In this case, the benefit of auxiliary variables should increase.

5. Numerical results

In this section we demonstrate the method of auxiliary variables through numerical experiments of increasing complexity. Our aim is to demonstrate the capability of auxiliary coarse variables to stabilize the multiscale methods for challenging problems in 3D. The simulations are conducted on parameters describing porous media exhibiting both short and long correlation lengths, as well as large jumps in the permeability coefficients. As for the geometry, only uniform Cartesian grids are considered. For all the numerical tests we apply auxiliary coarse variables restricted to either the wire basket or Dirichlet sources (fixed pressure wells).

When adding ACVs on an edge, the coarse space on the edge goes from representing linear (or reduced operators) functions for the vertex based method to a piecewise linear (or piecewise reduced operators) functions. Thus ACVs can therefore be considered a refinement of the coarse space in the form of a piecewise linear polynomial and increasing number of degrees of freedom, bearing some similarity with traditional *p*-refinement for finite elements. Therefore we chose to compare ACV refinement of the coarse operator to classical *h*-refinement of a vertex based method, which is realized by reducing the disparity between the fine and coarse scales. For simplicity, the additional ACVs are placed only on the subdomain edges. This keeps the refinement strategy relatively simple, and the wire basket scheme provides an upper limit of the possible improvement compared to the vertex based scheme.

5.1. Log-normal test case

Our first test case is a 3D grid with 108 cells in each direction, having in total 1.259.712 fine scale cells of unit size. The permeability has a log-normal distribution, as illustrated in Fig. 5. A Dirichlet injection well is placed in a corner, while as Neumann producer is located in the middle of the domain. For simplicity, periodic boundary conditions are applied.

The grid size facilitates vertex grids with a coarsening of 27, 9 and 3 cells in each direction, and we refer to these grids a V1, V2 and V3, respectively. Thus the coarse grids V2 and V3 can be seen as vertex refinements of grid V1. We will also refine grid V1 by placing 2 and 8 ACVs on each edge, rendering coarse grids denoted ACV1 and ACV2, respectively. Similarly, coarse grid ACV3 is created by adding 2 ACVs to each edge of grid V2. Finally, grid WB1 and WB2 are wire basket refinements of grid V1 and V2, respectively. A wire basket refinement of V3 was not considered, due to the large number of coarse nodes. The coarse grid configurations, together with the number of coarse unknowns and the number of GMRES iterations needed are summarized in Table 1.

Our first observation is that the vertex-based multiscale preconditioner is highly sensitive to the degree of coarsening, and indeed performs very poorly for the coarsest cases. This is consistent with our notion of low robustness for the vertex-based method in 3D. We observe that in this case, the numerical experiments indicate that the performance of the preconditioner can be improved by adding ACVs instead of vertex nodes. This is clearly seen by comparing grid V2 and ACV2: both coarse grids are refinements of grid V1, with grid V2 having somewhat more coarse unknowns than grid ACV2. Nevertheless, the ACV strategy needs fewer iterations to reach the desired residual.

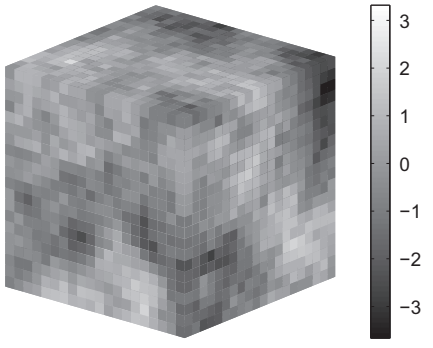

Fig. 5. A log-normal permeability field. The base-10 logarithm of the permeability is plotted.

Table 1
The number of primary (PCV) and auxiliary coarse variables for the log-normal grid. The last column shows the number of GMRES iterations needed to achieve a relative residual of 10^{-8}.

Grid	Type	# PCV	# ACV	# (Coarse nodes)	# it
V1	Vertex	64	1	65	212
V2	Vertex	1728	1	1729	151
V3	Vertex	46,656	1	46,657	45
ACV1	ACV	64	385	449	170
ACV2	ACV	64	1537	1601	107
ACV3	ACV	1728	10,369	12,079	74
WB1	WB	64	4993	5057	65
WB2	WB	1728	41,473	43,201	42

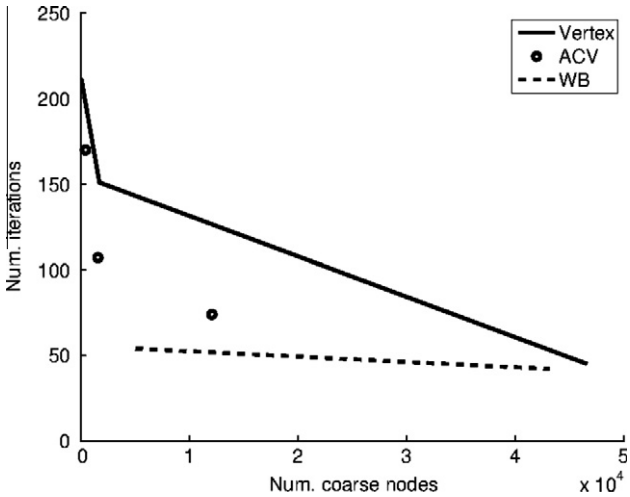

Fig. 6. Number of iterations vs. coarse unknowns for vertex and ACV grids for the log-normal permeability.

For convenience, Fig. 6 plots the iteration count vs. the number of coarse unknowns. The figure illustrates that the ACV scheme is a middle course between the vertex and wire basket schemes, and that the preconditioner can be significantly improved compared to the vertex based method by adding relatively few coarse variables.

5.2. SPE 10

As our second test case, we consider the permeability field from the tenth SPE comparative benchmark problem [30]. The grid consists of $60 \times 220 \times 85$ cells. Of these, the permeability in the upper 35 layers in the z-direction resembles a log-normal distribution, while the lower 50 layers are characterized by highly permeable channels and high permeability contrasts, see Fig. 7. Again, periodic boundary conditions are assumed in all tests. We note that to our knowledge, this represents the first systematic investigation of the multi-scale finite volume type methods applied as a preconditioner reported for problems of this size and complexity.

To investigate the impact of auxiliary coarse variables on this test case, we consider both the full model, as well as sub-models from the upper and lower part of the formation. In all test cases, we will place an injection well in one corner, and a producer in the middle of the domain. A fixed pressure is applied in the injection well, and the injection cell is modeled as an ACV for all simulations.

5.2.1. Refinement of the coarse space

As a first test, we investigate how adding ACVs to a vertex grid impacts the number of iterations. We define three cases, the two first being subsets of the full model, corresponding respectively to a log-normal and channelized permeability field. The third case represents the full benchmark dataset:

Case A: a grid of $60 \times 220 \times 35$, permeability from SPE10 layers 1–35
Case B: a grid of $60 \times 220 \times 35$, permeability from SPE10 layers 36–70
Case C: a grid of $60 \times 220 \times 85$, permeability from SPE10 layers 1–85

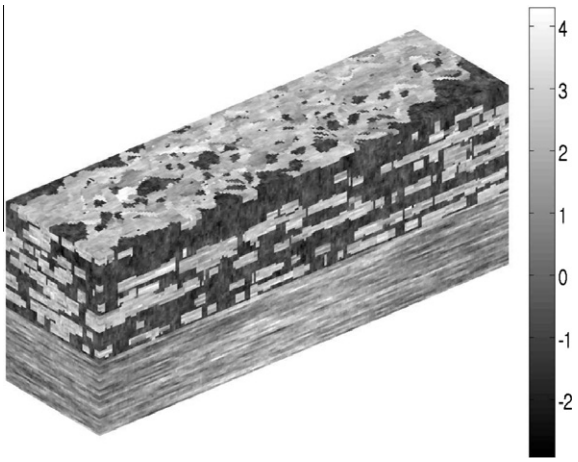

Fig. 7. The base-10 logarithm of the SPE10 permeability field. The formation is viewed upside down to emphasize the channelized features in the lower part of the reservoir.

Table 2
The number of GMRES iterations needed to reach a relative residual of 10^{-6} for the SPE10 test cases.

	# ACV	# (Coarse nodes)	# it, Case A	# it, Case B
Cases A and B				
Vertex	1	561	193	1207
ACV 1	2341	2801	99	876
ACV 2	5041	5601	96	931
ACV 3	8401	8961	62	618
WB	14,001	14,561	63	600
	# ACV	#(Coarse nodes)	# it, Case C	
Case C				
Vertex	1	1361	1516	
ACV 1	6801	8161	1082	
ACV 2	13,601	14,961	1091	
ACV 3	20,401	21,761	734	
WB	34,001	35,361	780	

On all these cases, we apply a coarse grid of $15 \times 11 \times 5$. We test vertex and wire basket preconditioners, as well as three preconditioners with an increasing number of ACVs denoted ACV 1, 2 and 3. The ACV refinement is the same for each edge, that is, permeability variations, etc. are not taken into account when defining the extra coarse nodes. The GMRES iterations are halted when the relative residual reaches 10^{-6}.

The number of coarse scale nodes and the results are summarized in Table 2. We see that the number of iterations needed to reach the desired tolerance decrease as auxiliary coarse variables are added. With the ACVs placed on the edges, the wire basket method is the limiting case with the densest coarse space, and thus a reduction factor of 2–3 compared to the vertex scheme is optimal for these tests. A large part of that improvement is achieved already when going to ACV 1. Also, the extra computational cost stemming from adding the extra coarse variables needed to go from ACV 3 to a full wire basket scheme does not seem to be justifiable.

The reduction in the iteration count is highest in the upper layers, where there are few abrupt changes in the permeability field, and thus the pressure solution has highest regularity. In the lower layers, the solution is erratic, and to best capture this behavior by ACVs, the auxiliary nodes should likely be placed on strategic locations in the reservoir, such as highly permeable channels. Despite that no adaptivity was applied to position the ACVs in the current test, the number of iterations needed is still reduced significantly, even when a relatively small number of ACVs are added.

5.2.2. Comparison of refinement strategies

It is worth comparing vertex and ACV refinement for the SPE 10 case, building on the analogy between h and p refinement. In this test, only Cases A and B are considered, to emphasize the role of the different structure in the permeability fields.

We consider three vertex grids with an increasing number of coarse unknowns. Furthermore, we consider two ACV schemes, created by refining the coarsest vertex grid in such a way that the ACV grids and refined vertex grids have a similar

Table 3
The number of GMRES iterations needed to reach a relative residual of 10^{-6} for the SPE10 test cases. Three vertex preconditioners were tested, as well as two ACV schemes.

Type	# PCV	# ACV	# (Coarse nodes)	# it, Case A	# it, Case B
Cases A and B					
Vertex	400	1	401	249	1481
Vertex	1680	1	1681	391	1984
Vertex	3698	1	3699	832	1732
ACV	400	1201	1601	171	1220
ACV	400	3201	3601	106	832

number of coarse unknowns. Thus the simulations illustrate the performance of vertex and ACV refinement for a fixed number of coarse cells.

The results for both Cases A and B are shown in Table 3, together with the number of coarse unknowns for the different schemes. For the log-normal-type permeability in Case A, vertex refinement renders a higher number of iterations needed compared to the strategy of adding auxiliary coarse variables. However, for the channelized Case B, the picture is less clear. This qualitative behavior is consistent with the notion of *h*-refinement for multiscale methods [6], where we note that for the case with high regularity (the upper layers) it is beneficial to go refine the coarse grid, while for the case of less regularity (the lower layers), the resonance effect between the characteristic length scales in the parameter field may influence the optimal grid spacing in a way where it is not always beneficial to reduce the spacing on the coarse grid.

In contrast, for both upper and lower layers, adding ACVs to the coarsest vertex grid renders fewer iterations. Compared to the vertex grids with similar number of coarse unknowns, the ACV scheme for most cases has about half the number of iterations; in one case the reduction factor is almost 8. These results can be seen in the context of ACV providing a bridge between the vertex-based approximation and the relatively more robust wire basket approximation, leading to a consistently better approximation of the Schur complement systems.

5.3. Standalone multiscale methods

As discussed in Section 3.3, the multiscale preconditioners derived herein can be seen as standalone multiscale approximations, which indeed gives the classical multiscale finite volume method [20]. As such, it is of interest to also discuss the performance of the multiscale preconditioner as an approximate solution by itself, not just its convergence properties.

While for 2D cross sections of the testcases discussed above, the multiscale methods can be made reasonably robust [7], the same is not true in 3D. In particular, for the regular Cartesian grids considered herein, the Multiscale preconditioner as a stand-alone solver gives unacceptable approximations even on the log-normal type permeability fields, and completely fails to give reasonable results for channelized problems such as the lower layers of SPE10. For this reason, we have chosen to emphasize the utility of multiscale methods in the iterative framework.

In practical implementations, the great advantage of the mass conservative types of preconditioners discussed herein is that converged solutions to the linear system of equations are not needed in order to have a locally conservative flow-field. Thus, in the iterative framework the tolerance of the linear solver can be chosen well above the tolerance used both when applying traditional preconditioners, but also above the tolerance used in our examples. This allows for considerable computational savings.

6. Conclusion

We have developed a new way of constructing coarse spaces for 3D multiscale simulations. While the existing coarse space for multiscale control volume methods based on vertex variables becomes unstable for three dimensional problems, we propose a generalized framework for including additional variables enriching the coarse space. We have denoted the additional coarse variables as auxiliary. The extended coarse space, based on auxiliary variables enables us to more accurately and more directly transfer fine-scale heterogeneous information onto the coarse scale. Moreover, it gives us the flexibility of constructing suitable coarse scale systems based on the complexity of the problem.

Numerical results show that the proposed framework can be used to construct efficient numerical methods for flow in 3D porous media; specially for problems involving long heterogeneous structures. In particular, the robustness of the multiscale framework in 3D is significantly enhanced with the novel formulation.

The wire basket multiscale method is a special case of the auxiliary coarse space, in which all edge cells are sampled on the coarse scale. This method degenerates to a robust preconditioner for problems involving homogeneous permeability on each sub-domain. While the WB-method usually gives the lowest number of iterations, much of the captured fine-scale information is often unnecessary. Numerical experiments show that much of the non-linear information may be captured within the local basis-functions by including a few auxiliary variables on the sub-domain boundaries. As such, the proposed methodology forms a flexibel and reasonable compromise between computational cost and iterative efficiency, which cannot be obtained by classical multiscale methods.

References

[1] D. Das, S.M. Hassanizadeh (Eds.), Upscaling Multiphase Flow in Porous Media: From Pore to Core and Beyond, Springer, Dordrecht, 2005.
[2] J. Bear, Dynamics of Fluids in Porous Media, Elsevier, New York, 1972.
[3] W. E, B. Engquist, The heterogeneous multiscale methods, Commun. Math. Sci. 1 (1) (2003) 87–132.
[4] I. Babuska, G. Caloz, J.E. Osborn, Special finite-element methods for a class of 2nd-order elliptic problems with rough coefficients, SIAM J. Numer. Anal. 31 (4) (1994) 945–981.
[5] T.J.R. Hughes, G.R. Feijóo, L. Mazzei, J.-B. Quincy, The variational multiscale method – a paradigm for computational mechanics, Comput. Methods Appl. Mech. Eng. 166 (1–2) (1998) 3–24.
[6] Y. Efendiev, T.Y. Hou, Multiscale Finite Element Methods: Theory and Applications, Surveys and Tutorials in the Applied Mathematical Sciences, vol. 4, Springer, New York, 2009.
[7] A. Sandvin, J.M. Nordbotten, I. Aavatsmark, Multiscale mass conservative domain decomposition preconditioners for elliptic problems on irregular grids, Comput. Geosci. 15 (3) (2011) 587–602.
[8] J.M. Nordbotten, P.E. Bjrstad, On the relationship between the multiscale finite-volume method and domain decomposition preconditioners, Comput. Geosci. 12 (3) (2008) 367–376.
[9] T.Y. Hou, X.-H. Wu, Z. Cai, Convergence of a multiscale finite element method for elliptic problems with rapidly oscillating coefficients, Math. Comput. 68 (227) (1999) 913–943.
[10] Y. Efendiev, V. Ginting, T. Hou, R. Ewing, Accurate multiscale finite element methods for two-phase flow simulations, J. Comput. Phys. 220 (1) (2006) 155–174.
[11] L. Durlofsky, Y. Efendiev, V. Ginting, An adaptive local-global multiscale finite volume element method for two-phase flow simulations, Adv. Water Resour. 30 (3) (2007) 576–588.
[12] I. Lunati, P. Jenny, Treating highly anisotropic subsurface flow with the multiscale finite-volume method, Multiscale Model. Simul. 6 (1) (2007) 308–318 (electronic).
[13] T.Y. Hou, X.-H. Wu, A multiscale finite element method for elliptic problems in composite materials and porous media, J. Comput. Phys. 134 (1) (1997) 169–189.
[14] H. Hajibeygi, G. Bonfigli, M.A. Hesse, P. Jenny, Iterative multiscale finite-volume method, J. Comput. Phys. 227 (19) (2008) 8604–8621.
[15] J.M. Nordbotten, Adaptive variational multiscale methods for multiphase flow in porous media, Multiscale Model. Simul. 7 (3) (2008) 1455–1473.
[16] A. Toselli, O. Widlund, Domain Decomposition Methods – Algorithms and Theory, Series in Computational Mathematics, vol. 34, Springer-Verlag, Berlin, 2005.
[17] I.G. Graham, P.O. Lechner, R. Scheichl, Domain decomposition for multiscale PDEs, Numer. Math. 106 (4) (2007) 589–626.
[18] B.F. Smith, A domain decomposition algorithm for elliptic problems in three dimensions, Numer. Math. 60 (1) (1991) 219–234.
[19] C. Farhat, F.-X. Roux, A method of finite element tearing and interconnecting and its parallel solution algorithm, Int. J. Numer. Methods Eng. 32 (1991) 1205–1227.
[20] P. Jenny, S.H. Lee, H.A. Tchelepi, Multi-scale finite-volume method for elliptic problems in subsurface flow simulation, J. Comput. Phys. 187 (1) (2003) 47–67.
[21] A. Quarteroni, A. Valli, Domain decomposition methods for partial differential equations, in: Numerical Mathematics and Scientific Computation, The Clarendon Press Oxford University Press, New York, 1999. Oxford Science Publications.
[22] B.F. Smith, P.E. Bjørstad, W.D. Gropp, Domain Decomposition, Cambridge University Press, Cambridge, 1996.
[23] Y. Saad, M.H. Schultz, GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems, SIAM J. Sci. Statist. Comput. 7 (3) (1986) 856–869.
[24] J. Nordbotten, E. Keilegavlen, Inexact linear solvers for control volume discretizations, SIAM J. Sci. Comput., submitted for publication.
[25] J. Mandel, B. Sousedík, Coarse spaces over the ages, in: Domain Decomposition Methods in Science and Engineering XIX, Lecture Notes in Computational Science and Engineering, vol. 78, Springer-Verlag, Berlin, Heidelberg, 2011.
[26] O. Widlund, The development of coarse spaces for domain decomposition algorithms, in: Domain Decomposition Methods in Science and Engineering XVIII, Lecture Notes in Computational Science and Engineering, vol. 70, Springer-Verlag, Berlin Heidelberg, 2009.
[27] J. Galvis, Y. Efendiev, Domain decomposition preconditioners for multiscale flows in high-contrast media, Multiscale Model. Simul. 8 (4) (2010) 1461–1483.
[28] J. Galvis, Y. Efendiev, Domain decomposition preconditioners for multiscale flows in high contrast media: reduced dimension coarse spaces, Multiscale Model. Simul. 8 (5) (2010) 1621–1644.
[29] C. Wolfsteiner, S.H. Lee, H.A. Tchelepi, Well modeling in the multiscale finite volume method for subsurface flow simulation, Multiscale Model. Simul. 5 (3) (2006) 900–917 (electronic).
[30] M. Christie, M. Blunt, Tenth SPE comparative solution project: a comparison of upscaling techniques, SPEREE 4 (4) (2001) 308–317.