EFFICIENCY STUDY OF THE MULTISCALE FINITE VOLUME FORMULATION FOR MULTIPHASE FLOW AND TRANSPORT

A REPORT
SUBMITTED TO THE DEPARTMENT OF ENERGY RESOURCES ENGINEERING OF STANFORD UNIVERSITY IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE

Xiaochen Wang
June 2009
I certify that I have read this report and that in my opinion it is fully adequate, in scope and in quality, as partial fulfillment of the degree of Master of Science in Petroleum Engineering.

Hamdi Tchelepi  Principal Adviser
Abstract

Multiscale methods have been developed to solve multiphase flow and transport problems in large-scale heterogeneous porous media accurately and efficiently. In this report, the computational efficiency of the multiscale finite-volume method (MSFV) is analyzed.

The power of MSFV lies in its ability to combine local basis functions with a global coarse-scale problem to solve highly detailed heterogeneous models. In the first part of this report, we compare MSFV with conventional sequential strategies for solving coupled multiphase flow and transport that employ state-of-the-art linear solvers. Specifically, conventional sequential implicit methods with algebraic multigrid (AMG) for the pressure equation and incomplete LU factorization (ILU) for the saturation equations are used as the reference. We developed modular object-oriented simulation codes for both the multiscale and fine-scale simulation methods. Our results indicate that the adaptivity in pressure (reuse of the basis functions), velocity, and saturation calculations employed in MSFV leads to more efficient computations compared with the conventional fine-scale sequential implicit method. In the two test cases described here, the MSFV simulations are, respectively, eight and two times faster than the sequential fine-scale simulation using AMG and ILU. The efficiency study in this part serves as a solid basis for further development of MSFV as a general algebraic approach for solving nonlinear flow and transport in highly detailed heterogeneous reservoir models.

In the second part, we employ an MSFV-based upscaling strategy for multiphase flow and transport, where the original MSFV algorithm is used to construct accurate
coarse-scale solutions. The computational domain is dynamically divided into ahead-of-the-front, front, and behind-the-front regions, according to the time evolution of local coarse-scale flow information. The fine-scale solution is reconstructed locally and adaptively in the front region, where the flow field changes rapidly so that very accurate fine-scale solutions are necessary. For the other two regions, which are also determined dynamically, only coarse-scale velocity and saturation fields are computed. The accuracy and efficiency of this MSFV-based upscaling method are tested using several numerical examples, including 2D and 3D models, for both incompressible and compressible flows. Our results indicate that MSFV-based upscaling provides coarse-scale solutions that are in excellent agreement with volume averaged fine-scale solutions, which we take as reference, at a cost comparable to conventional coarse-scale simulation of relatively small models. As the size of the coarse-scale model itself increases, the MSFV-based upscaling strategy is expected to be more accurate and much more efficient compared with schemes that work exclusively with coarse-scale operators for flow and transport.
# Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Subsection</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abstract</td>
<td></td>
<td>iii</td>
</tr>
<tr>
<td>Acknowledgments</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>1 Introduction</td>
<td>1.1 Development of Multiscale Methods</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>1.2 State-of-the-art Linear Solvers</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>1.3 Upscaling Techniques</td>
<td>6</td>
</tr>
<tr>
<td>2 Efficiency Study of the Multiscale Finite Volume Method</td>
<td>2.1 Adaptive Multiscale Finite Volume Method</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>2.1.1 Governing Equations</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>2.1.2 Adaptive MSFV Formulation</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>2.2 Numerical Schemes and Solvers</td>
<td>16</td>
</tr>
<tr>
<td></td>
<td>2.3 Numerical Examples</td>
<td>17</td>
</tr>
<tr>
<td></td>
<td>2.3.1 Case 1: 2D, incompressible, heterogeneous</td>
<td>17</td>
</tr>
<tr>
<td></td>
<td>2.3.2 Case 2: 2D, incompressible, SPE10 top layer</td>
<td>21</td>
</tr>
<tr>
<td>3 Multiscale Finite-Volume Based Upscaling</td>
<td>3.1 Adaptive Upscaling</td>
<td>27</td>
</tr>
<tr>
<td></td>
<td>3.2 Coarse-Scale Based Transition Criteria</td>
<td>28</td>
</tr>
<tr>
<td></td>
<td>3.3 Numerical Examples</td>
<td>29</td>
</tr>
<tr>
<td></td>
<td>3.3.1 Case 1: 2D, incompressible, heterogeneous</td>
<td>29</td>
</tr>
<tr>
<td></td>
<td>3.3.2 Case 2: 2D, incompressible, SPE10 top layer</td>
<td>34</td>
</tr>
</tbody>
</table>
3.3.3 Case 3: 3D, compressible, heterogeneous .......................... 39
3.4 Comparison with Original MSFV ........................................... 43

4 Summary and Future Work ...................................................... 48
# List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Error norms in case 1</td>
<td>33</td>
</tr>
<tr>
<td>3.2</td>
<td>Efficiency comparison in case 1</td>
<td>34</td>
</tr>
<tr>
<td>3.3</td>
<td>Error norms in case 2</td>
<td>39</td>
</tr>
<tr>
<td>3.4</td>
<td>Efficiency comparison in case 2</td>
<td>39</td>
</tr>
<tr>
<td>3.5</td>
<td>Error norms in Case 3</td>
<td>43</td>
</tr>
<tr>
<td>3.6</td>
<td>Efficiency comparison in Case 3</td>
<td>44</td>
</tr>
<tr>
<td>1</td>
<td>Functions for fine-scale simulation</td>
<td>55</td>
</tr>
<tr>
<td>2</td>
<td>Functions for both fine- and multi-scale simulations</td>
<td>55</td>
</tr>
<tr>
<td>3</td>
<td>Functions for multiscale simulation</td>
<td>56</td>
</tr>
</tbody>
</table>
List of Figures

2.1 Flow chart for sequential fully implicit scheme . . . . . . . . . . . . . 10
2.2 Primal and dual coarse grid . . . . . . . . . . . . . . . . . . . . . . 11
2.3 Permeability field and fine-scale saturation distribution in case 1 . . 18
2.4 Comparison of total simulation time between three scheme in case 1 . 19
2.5 Total simulation time between fine AMG and MSFV in case 1 . . . 19
2.6 Comparison of percentage of simulation time for each scheme in case 1 20
2.7 Comparison of pressure time in case 1 . . . . . . . . . . . . . . . . . 21
2.8 Comparison of saturation time in case 1 . . . . . . . . . . . . . . . 22
2.9 Permeability field in case 2 . . . . . . . . . . . . . . . . . . . . . . . 22
2.10 Fine-scale saturation distribution in case 2 . . . . . . . . . . . . . . 22
2.11 Comparison of total simulation time between three scheme in case 2 . 23
2.12 Total simulation time between fine AMG and MSFV in case 2 . . . 24
2.13 Comparison of pressure time in case 2 . . . . . . . . . . . . . . . . . 24
2.14 Comparison of saturation time in case 2 . . . . . . . . . . . . . . . . 25
3.1 Upscaling region and algorithm . . . . . . . . . . . . . . . . . . . . . 27
3.2 Fine-scale simulation result at t = 1.15 PVI in case 1 . . . . . . . . 30
3.3 Upscaled results in case 1 . . . . . . . . . . . . . . . . . . . . . . . 31
3.4 Cumulative oil recovery and oil fraction in production in case 1 . . 33
3.5 Permeability field in case 2 . . . . . . . . . . . . . . . . . . . . . . . 35
3.6 Fine-scale saturation at 1.8 τ₀ in case 2 . . . . . . . . . . . . . . . . 35
3.7 Fine-scale saturation at 5.9 τ₀ in case 2 . . . . . . . . . . . . . . . . 35
3.8 Volume averaged fine-scale saturation (reference) at 1.8 τ₀ . . . . 36
3.9 MSFV coarse-scale saturation at 1.8 τ₀ . . . . . . . . . . . . . . . . 36
3.10 Region and Algorithm at 1.8 $\tau_0$ ........................................... 36
3.11 Volume averaged fine-scale saturation (reference) at 5.9 $\tau_0$ ................. 37
3.12 MSFV Coarse-scale saturation at 5.9 $\tau_0$ ........................................... 37
3.13 Region and Algorithm at 5.9 $\tau_0$ ........................................... 37
3.14 Cumulative oil recovery and oil fraction in production in case 2 .................. 38
3.15 Permeability distribution in case 3 ........................................... 40
3.16 Fine-scale simulation result at 0.061 PVI in case 3 ........................................... 41
3.17 MSFV-based upscaling simulation result at 0.061 PVI in case 3 ......................... 41
3.18 Fine-scale simulation result at 0.728 PVI in case 3 ........................................... 42
3.19 MSFV-based upscaling simulation result at 0.728 PVI in case 3 ......................... 43
3.20 Trade-off between coarse-scale pressure error and efficiency ......................... 45
3.21 Trade-off between coarse-scale saturation error and efficiency ......................... 46

1 Code structure of fine-scale simulation ........................................... 53
2 Code structure of multiscale simulation ........................................... 54
Acknowledgments

First of all, I would like to give my sincere gratitude to my advisor Prof. Hamdi Tchelepi, who provided instruction and supervision related to both my research and course work during these two years. Thanks are due to his insightful suggestions, kindly encouragement, and helpful advice for my research. I do appreciate and benefit a lot from all the discussions and interactions with him.

Furthermore, Seong H. Lee of Chevron, ETC has been deeply involved with this work. Seong was my mentor during my summer internship at Chevron. His important ideas and helpful advice on this research are highly acknowledged.

What is more, I also would like to thank Hui Zhou, a senior PhD student in the department, for his significant help on this work.

I am grateful to my colleagues and friends at Stanford, who make my life here more colorful. Special thanks go to Yifan Zhou, for his great help within and outside of my academic life. Last but not least, this is for my parents, who support me in the past, present, and future.
Chapter 1

Introduction

Subsurface formation properties, like permeability and porosity, typically display heterogeneity with high levels of variability spanning a wide range of length scales. Small-scale heterogeneity may affect flow and transport at larger scales and have an impact on the modeled dynamic response of the reservoir. Therefore, accurate simulation of flow in subsurface porous media relies on being able to resolve the flow physics using high resolution reservoir characterizations.

1.1 Development of Multiscale Methods

To tackle this difficulty, a flourishing investigation on multiscale methods in reservoir simulation has developed. The objective of multiscale methods is to efficiently obtain accurate description of the flow at coarse scales that capture the effects of variations at the smaller scales. Many multiscale methods also focus on the reconstruction of high-fidelity fine-scale solutions that make full use of coarse-scale operators. Hou and Wu ([14]) proposed a multiscale finite-element method (MsFEM) for elliptic problems with special basis functions within each element. Localization was achieved by special conditions at the boundaries of coarse elements. However, the fine-scale velocities from this method are not mass conservative. Later, Chen and Hou ([13]) provided evidence that a conservative velocity field is necessary for accurate solution of transport, and they described a mixed finite-element multiscale method which
CHAPTER 1. INTRODUCTION

results in conservative fine-scale velocities. Another mixed finite-element multiscale method was presented by Arbogast ([11]) and later by Arbogast and Bryant ([12]) for simulating two-phase flow in heterogeneous porous media. The fine-scale influence was coupled with the coarse-scale solution using numerical Greens functions.

Finite-volume based formulations have emerged as an important branch of multiscale methods. Recently, Jenny, Lee, and Tchelepi ([16], [17], [18]) proposed the Multiscale Finite-Volume (MSFV) method. In the MSFV approach, two sets of basis functions, dual and primal, are used to solve the finite-scale elliptic problem efficiently followed by computation of a locally conservative fine-scale velocity field. Excellent performance of their MSFV method for incompressible flow, without gravity or capillarity, was demonstrated when compared with an established finite-volume simulator. In their work, adaptivity of basis-function calculation was introduced, which enhances the computational efficiency substantially. Because it is based on finite-volume discretization, MSFV is generally preferred for oil reservoir simulation practice, compared with finite-element multiscale methods.

The MSFV approach was later recast as an operator-based multiscale method (OBMM) and extended for compressible multiphase flow by Zhou and Tchelepi ([23]). OBMM is a general algebraic formulation, where additional physical processes can be included naturally. Moreover, OBMM provides a framework for extending MSFV from structured to unstructured grids relatively easily.

Adaptive prolongation and restriction operators for multiphase transport (i.e., saturation calculation) were recently developed by Lee, Zhou, and Tchelepi ([21], [24]). The calculations of the velocity and saturation fields using the MSFV method were extended to include dynamic adaptivity. Compared with the original MSFV, adaptivity in the solution of the transport equations, which complements the adaptivity in computing the flow field (pressure and total-velocity), leads to further improvements in the overall efficiency of the MSFV method for coupled flow and transport in highly detailed reservoir models.
1.2 State-of-the-art Linear Solvers

Just as in many other engineering research and application areas, in reservoir simulation powerful linear solvers for solving large-scale linear systems become more and more significant as the problem size and model complexity increase. For linear systems arising in practical models, which are usually sparse and with a size of $10^6$ to $10^7$, direct linear solvers like Gaussian elimination are not applicable due to their highly expensive computational costs (i.e., $O(N^3)$ for $N \times N$ matrix system). Instead, various iterative techniques, which take advantage of the matrix sparsity and thus have a smaller computational cost (i.e., $O(N^{1+\alpha})$ for $N \times N$ sparse matrix system, where $0 \leq \alpha \leq 1$), are developed and used for solving large-scale sparse linear systems.

Among general-purpose linear solvers, GMRES (Generalized Minimal Residual Method) is the most used Krylov subspace approach in reservoir simulation ([30]). As is well known ([29]), the performance of the iterative solver depends strongly on the preconditioner used. Here, we use two preconditioning methods to solve the coupled systems of algebraic equations, namely, Incomplete LU factorization (ILU) and algebraic multigrid (AMG).

- **ILU**: LU factorization is a process for representing a matrix $A$ as the product of a lower triangular matrix, $L$, and an upper triangular matrix, $U$:

$$A = LU.$$  

Such a factorization can be used to solve linear systems. However, in reservoir simulation problems, the $L$ and $U$ factors are not necessarily sparse. Instead, incomplete LU (ILU) factorization is used as a preconditioner for iterative linear solution. In the most widely used ILU(0) factorization, the lower and upper triangular matrices retain non-zero elements only in those positions that contain non-zero elements in the original matrix.

According to the type of matrices the preconditioner is applied to, ILU can be categorized into pointwise ILU and blockwise ILU (BILU) ([29]). Generally speaking, ILU (both pointwise and blockwise) is preferred for solving matrices
obtained from linearized hyperbolic conservation laws, which usually correspond to the transport equations in reservoir simulation. Here, pointwise ILU(0) is used for the pressure and saturation solution in MSFV, and for the saturation solution in the reference sequential fine-scale simulation. This is because when the sequential implicit scheme is applied to two-phase flow problems, each grid block contains only one equation and one variable for both the pressure and saturation solution stages, i.e., the block size is always one by one.

- **AMG:** Multigrid methods can effectively remove the low frequency errors in near-elliptic problems. The solution process, which involves relaxation, transfer of residuals from fine to coarse grids, and interpolation of corrections from coarse to fine levels, is very efficient in obtaining the solution on the finest grid ([27]). The high efficiency of multigrid methods lies in the fact that error that is only slightly affected by relaxation (smooth error) can be easily approximated on a coarser grid by solving the residual equation there, where it is cheaper to compute. This approximation of the error is then interpolated to the fine grid and used to correct the solution.

In geometric multigrid methods, coarse grids are determined based on geometry information (such as grid spacing). In contrast, algebraic multigrid (AMG) chooses coarse grids using the given fine-scale coefficient matrix only. This makes AMG an attractive “black box” solver, which greatly facilitates its application. The most preferable matrix for AMG to solve is the M matrix, which is irreducibly diagonally dominant with nonpositive off-diagonal elements and nonnegative eigenvalues. Ideal performance of AMG is achieved when solving large matrices of elliptic, or weakly parabolic character, which usually correspond to the linearized pressure system of equations in reservoir simulation.

The computational efficiency of MSFV is studied in the second chapter of the report. MSFV is compared with sequential simulation using AMG for pressure (elliptic system) and ILU for saturation (hyperbolic system).
1.3 Upscaling Techniques

In many applications of reservoir simulation, an efficient coarse-scale solution is needed due to two practical reasons:

- Infeasibility of direct simulation using the reference fine-scale geocellular model.
- The need for simulation runs using a large number of geostatistical realizations to study the uncertainty in the predictions due to uncertainty in the reservoir characterization model.

Given fine-scale field information (e.g., permeability, porosity), upscaling techniques are usually introduced to coarsen the fine-scale model to a more manageable resolution. It is important that these coarsened flow models account for the fine-scale heterogeneity in terms of key behaviors, such as the overall flow rate for given boundary conditions. Conventional upscaling techniques can be divided into single-phase upscaling and multiphase upscaling, according to the type of coarse-scale quantities that are computed. Specifically, the objective of single-phase upscaling is to obtain coarse-scale quantities characterizing single-phase flow, like effective transmissibility and porosity. As for multiphase upscaling, quantities such as pseudo-relative permeability, which characterizes multiphase flow, are computed. Single-phase upscaling is computationally more efficient than multiphase upscaling. In fact, the application of single-phase upscaling techniques is not limited to single phase flow problems, since reasonable accuracy can be obtained when applying single-phase upscaling to some multiphase flow systems. For comprehensive reviews of upscaling techniques for flow in porous media, see [3], [4] and [2].

For single-phase upscaling, one of the most representative works is the local-global upscaling method proposed by Chen and Durlofsky ([5]). In their work, effective (equivalent) transmissibilities for the coarse cells are computed before performing the full-physics simulations.

Upscaling for multiphase flow quite challenging because of the strong dependence upon flow and saturation histories and the nonlinear fractional flow function. Efendiev
and Durlofsky ([9], [10]) derived a generalized convection-diffusion equation for two-phase flow. Later, Chen and Durlofsky ([6]) combined the generalized convection-diffusion equation with local-global upscaling. They showed that this combination provides reasonable accuracy in many cases.

As evidenced in recent research on upscaling of multi-phase flow, reconstruction of the subcell velocity field constitutes an important step toward improving the accuracy of coarse-scale models. Efendiev and Durlofsky ([9], [10]) incorporated sub-scale effects that appear as a nonlinear, nonlocal term in the coarse-scale saturation equation. A mathematical framework of homogenized equations was recently proposed by Hou et al. ([15]). In their work, fluctuations of fine-scale quantities for two-phase flow are averaged along streamlines so that the limitation of small fluctuations in all quantities of the method by Efendiev and Durlofsky ([9]) can be overcome.

To meet the objective of an effective simulation method that computes accurate coarse-scale solutions and reconstructs local fine-scale properties dynamically when and where necessary, an upscaling technique for multiphase flow via adaptive reconstruction of fine-scale properties is designed and investigated in the third chapter of this report. For this technique, MSFV is applied to reconstruct the local fine-scale velocity and saturation fields. The approximate fine-scale flow information is then used to update the local coarse-scale solution in order to improve the accuracy of the coarse-scale model. After a detailed description of this MSFV-based upscaling, several numerical examples, including 2D and 3D fields, for both incompressible and compressible flow, are used to demonstrate the accuracy and efficiency of this method.
Chapter 2

Efficiency Study of the Multiscale Finite Volume Method

In this chapter, the multiscale finite volume method (MSFV) \(^{(16)}\) is described and analyzed. The focus is on the computational efficiency of MSFV compared with conventional sequential methods for fine-scale simulation. The simulators, for both fine-scale and multiscale simulations, have been implemented in C++ (Appendix A). The structure of this chapter is as follows. In the first section, the governing equations and the MSFV algorithm are described. Then, the numerical schemes and the different solvers for flow and transport are described in the second section. In Section 3, numerical examples of our efficiency study are shown.

2.1 Adaptive Multiscale Finite Volume Method

2.1.1 Governing Equations

We consider two-phase flow in porous media, without gravity or capillarity. The governing equations for reservoir simulation are the mass conservation equations of the two immiscible phases,
CHAPTER 2. EFFICIENCY STUDY OF THE MULTISCALE FINITE VOLUME METHOD

\[
\frac{\partial (\phi b_\alpha S_\alpha)}{\partial t} + \nabla \cdot (b_\alpha u_\alpha) = -q_\alpha, \tag{2.1}
\]

where subscript \( \alpha = w, o \) denotes the phase. \( S_\alpha, u_\alpha \), and \( q_\alpha \) are phase saturation, phase Darcy velocity, and phase source term, respectively. Here, \( q_\alpha \) is defined at standard conditions, and a positive value of \( q_\alpha \) corresponds to production. The \( b_\alpha \) in Equation 2.1 characterizes the phase compressibility and is a function of pressure \( p \). It is the inverse of the phase-formation volume-factor, which is defined as the ratio of the density at reservoir conditions to the density at standard conditions. The porosity, \( \phi \), is also considered a function of pressure.

Darcy’s velocity of phase \( \alpha \) is given by:

\[
u_\alpha = -\frac{k_{ra}}{\mu_\alpha} k \cdot \nabla p, \tag{2.2}\]

where \( k \) is the (absolute) permeability tensor, and \( k_{ra} \) and \( \mu_\alpha \) are the phase relative permeability and phase viscosity, respectively.

Another equivalent mathematical formulation of the conservation equations, which is used in the sequential pressure-saturation scheme, is to replace one of the conservation equations with a pressure equation. The expressions of Equation 2.1 are first divided by \( b_\alpha \) and then added together. After some manipulation, one obtains

\[
\nabla \cdot (\lambda \cdot \nabla p) = \sum_\alpha \frac{q_\alpha}{b_\alpha} + \frac{\partial \phi}{\partial t} + \sum_\alpha \frac{\phi S_\alpha}{b_\alpha} \frac{\partial b_\alpha}{\partial t} - \sum_\alpha \left( \frac{\lambda_\alpha \cdot \nabla p}{b_\alpha} \cdot \nabla b_\alpha \right). \tag{2.3}\]

This is the pressure equation. Here, \( \lambda_\alpha = \frac{k_{ra}}{\mu_\alpha} k \) is the \( \alpha \)-phase mobility tensor and \( \lambda = \sum_\alpha \lambda_\alpha \) is the total-mobility tensor. The pressure equation combined with one of the conservation equations (saturation equations), e.g., \( \alpha = w \), form the starting point for multiscale simulation.
2.1.2 Adaptive MSFV Formulation

The adaptive MSFV formulation employs an implicit operator splitting scheme, in which the pressure and saturation equations are solved sequentially ([17]). The outer loop contains a (1) pressure inner loop, (2) intermediate velocity calculation, and (3) a saturation inner loop. The flow chart of this sequential scheme is shown in Figure 2.1.

![Flow chart for sequential fully implicit scheme](image)

Figure 2.1: Flow chart for sequential fully implicit scheme

There are two coarse grids in the MSFV formulation, as illustrated in Figure 2.2. A conforming coarse grid (primal coarse grid) with $M$ nodes and $N$ cells is superimposed on the original fine grid. In Figure 2.2, the solid lines represent primal coarse cells, $\Omega_i^H$ ($i \in 1, \ldots, N$), and the dashed lines represent dual coarse cells, $\Omega_j^D$ ($j \in 1, \ldots, M$), which are constructed by connecting the centers of the primal coarse cells. The thin solid lines indicate the underlying fine grid.

Within the pressure inner loop, basis functions $\Theta^k_j$, one for each corner $k$ of each dual coarse grid cell $\Omega_j^D$, is computed by solving the equation
\[ \nabla \cdot (\lambda \nabla \Theta^k) = 0 \text{ on } \Omega^D, \]  
(2.4)

with reduced Neumann boundary conditions, which localize the basis function calculation. At the faces of \( \partial \Omega^D \), we specify

\[ \nabla_\perp \cdot (\lambda \cdot \nabla \Theta^k)_{\perp} = 0, \]  
(2.5)

where the subscript \( \perp \) denotes the vector normal to \( \partial \Omega^D \) and at any corner \( x_i \) of \( \Omega^D \), we have

\[ \Theta^k_j(x_i) = \delta_{ik}. \]  
(2.6)

Adaptivity in the computation of the basis functions is based on the time-change of the total mobility:

\[ \frac{1}{1 + \epsilon_\lambda} < \frac{\lambda^{n+1,\nu}}{\lambda^n} < 1 + \epsilon_\lambda, \]  
(2.7)

where the superscripts \((n + 1, \nu)\) and \(n\) denote the previous iteration of the current time step and the previous time step, respectively. The scalar \( \epsilon_\lambda > 0 \) is a user-defined
threshold. If the condition of Equation 2.7 is not fulfilled for any fine cell within a dual coarse cell, the basis function associated with that dual coarse cell must be recomputed in the current iteration. Otherwise, the basis functions computed from the last iteration are reused. From our numerical experience, about 5% of the basis functions need to be updated for a given iteration to achieve satisfactory accuracy of the simulation results.

Note that all the basis functions are local. For an interface between two adjacent primal coarse cells:

\[
\partial \Omega_{i_1}^H = \partial \Omega_{i_1}^H \cap \partial \Omega_{i_2}^H, \quad (2.8)
\]

the phase transmissibility is given by

\[
T_{\alpha,k}^{i_1i_2} = \int_{\partial \Omega_{i_1i_2}^H} \sum_{j=1}^{N} \left( \lambda_\alpha \cdot \nabla \Theta_j^k \right) \cdot n \, d\Gamma, \quad (2.9)
\]

where the vector \( n \) is the unit normal with respect to \( \partial \Omega_{i_1i_2}^H \) pointing in the direction from \( \Omega_{i_1}^H \) to \( \Omega_{i_2}^H \). Subscript \( \alpha \) represents the phase, while subscript \( k \) is the index of a corner for dual coarse cell \( j \).

Once the coarse-scale transmissibilities are computed, we can solve the global coarse-scale pressure equation:

\[
T p^H = -q^H, \quad \text{for } p^H \in \Omega^H, \quad (2.10)
\]

where \( T \) is the matrix of coarse-scale transmissibilities. Here, superscript \( H \) represents coarse-scale variables (e.g., pressure, velocity), while \( h \) denotes fine-scale variables. To obtain the fine-scale pressure field, a prolongation operator, \( P \), can be used, which is defined as:

\[
p^b = P p^H. \quad (2.11)
\]
In this equation, $\mathcal{P}$ is an $n \times N$ matrix, where $n$ is the number of global fine cells and $N$ is the number of global primal coarse cells. This prolongation operator is assembled from the basis functions and applied to the coarse-scale pressure, namely,

$$p^h_j(x) = \sum_k \Theta^k_j p^H_k, \quad \text{for } x \in \Omega^D_j. \quad (2.12)$$

The fine-scale pressures obtained using Equation (2.12) are not necessarily conservative on the fine scale, and we may need to reconstruct a conservative field when necessary.

The coarse-scale total velocity is given by

$$u^{H,\nu+1}_{i_1i_2} = T_{i_1i_2}(S^{H,\nu}_{i_1i_2})(p^H_{i_2} - p^H_{i_1}), \quad (2.13)$$

where superscript $\nu$ and $\nu + 1$ represent the previous and current iterations, respectively, for the current time step; subscripts $i_1$ and $i_2$ represent two adjacent primal coarse cells.

To reconstruct the fine-scale velocities, two kinds of operators are used. The first one is full reconstruction, which is a nonlinear operator. For this operator, we need to solve the elliptic pressure equation for each primal coarse grid with the flux boundary condition computed from the reconstructed fine-scale pressure from Equation (2.12)

$$\nabla \cdot \lambda \nabla p^h = q_o + q_w. \quad (2.14)$$

The pressures obtained from Equation (2.14) are conservative on the fine scale, and they can be used to compute conservative fine-scale velocities.

The fine-scale velocities can be also obtained by simple linear interpolation of the coarse-scale velocities, specifically,
CHAPTER 2. EFFICIENCY STUDY OF THE MULTISCALE FINITE VOLUME METHOD

\[ u_h^{\nu+1}(x) = u_h^{\nu}(x) + u_i^{H,\nu+1}(x_0) - u_i^{H,\nu}(x_0) + \frac{x - x_0}{|x_1 - x_0|} \]
\[ \otimes(u_i^{H,\nu+1}(x_1) - u_i^{H,\nu}(x_0)) - u_i^{H,\nu+1}(x_0) + u_i^{H,\nu}(x_0) \], \quad (2.15) \]

where \( \mathbf{u} \otimes \mathbf{v} = (u_1 v_1, u_2 v_2) \) and \( x_0 \) and \( x_1 \) are the coordinates of the coarse cell at the bottom-left and top-right corners, respectively. We can readily prove that if \( u_h^{\nu} \) and \( u_i^{H,\nu+1} \) are conservative on the fine and coarse scales, respectively, the interpolated fine-scale velocity, \( u_h^{\nu+1} \), is also conservative on the fine scale. One can expect that the simple linear operator is more computationally efficient, but less accurate than full reconstruction. The nonlinear full reconstruction operator and the linear interpolation operator can be chosen adaptively according to local flow information.

In the saturation inner loop, the coarse-scale saturations can be calculated by solving a global coarse-scale transport equation:

\[ \frac{V_i^H}{\Delta t} (S_i^{H,n+1} - S_i^{H,n}) = \sum_{\ell \in \partial \Omega_i^H} u_{i\ell}^H F_{i\ell}^H + q_i^H, \quad (2.16) \]

where \( V_i^H \) and \( q_i^H \) are the volume and total source term, respectively, for a primal coarse cell \( i \). \( \sum_{\ell \in \partial \Omega_i^H} u_{i\ell}^H F_{i\ell}^H \) provides all the fluxes into the given coarse cell \( i \). In Equation 2.16 the coarse cell total velocity, \( u_i^H \), is given by:

\[ u_i^H = \sum_{j \in \partial \Omega_i^H} u_j^h, \quad (2.17) \]

and the fractional flow in coarse cell, \( F_{i\ell}^H \), is given by:

\[ F_{i\ell}^H = \frac{1}{u_i^H} \sum_{j \in \partial \Omega_i^H} f(S_j^h) u_j^h. \quad (2.18) \]

To reconstruct fine-scale saturations, we employ two different operators. The
first one is a local nonlinear operator, in which an overlapping Schwarz method is used to reconstruct the fine-scale saturations. Specifically, we compute the fine-scale saturation by locally solving the fine-scale transport equations in each primal coarse cell with the saturation distribution available from the previous time step, or iteration, as a boundary condition. So, we have

\[
S_j^{h,\nu+1} = S_j^{h,\nu} + \frac{\Delta t}{V_j^h} \sum_l u_l^{h,\nu+1} f(S_j^{h,\nu+1}),
\]

(2.19)

where \(V_j^h\) is the volume of fine cell \(j\). The current and previous iterations are denoted by \(\nu + 1\) and \(\nu\), respectively. The index \(l\) represents the fine cells adjacent to fine cell \(j\).

The second operator is linear, in which the fine-scale saturation is calculated using linear interpolation of the change in the coarse-scale saturation during the time step, or iteration. Specifically, we write

\[
S_j^{h,\nu+1} = S_j^{h,\nu} + \xi_j \delta S_i^H,
\]

(2.20)

where \(\delta S_i^H\) is the saturation change in coarse cell \(i\) between the current iteration, \(\nu + 1\), and the previous iteration, \(\nu\). The coefficient \(\xi_i\) is the relative time-change of the fine-scale saturation with respect to the time-change of the coarse-scale saturation:

\[
\xi_i = \frac{\delta S_j^h}{\delta S_i^H}.
\]

(2.21)

Here, we assume that the relative saturation change, \(\xi_j\), does not vary much from the previous iteration. The nonlinear operator for saturation computation is conservative on the fine scale. However, the linear operator is conservative on the coarse-scale only. Again, the linear operator is less accurate but faster than the corresponding nonlinear operator.
2.2 Numerical Schemes and Solvers

We compare the efficiency of MSFV with that of conventional fine-scale simulation using state-of-the-art linear solvers. A sequential implicit scheme is used for both the MSFV and reference fine-scale simulations presented here. This allows us to compare the solver efficiency for the pressure and saturation solutions separately. For the pressure solution, the MSFV method is compared with fine-scale solutions obtained using AMG, which is widely used to solve elliptic, or weakly parabolic, systems like the pressure equation \([29]\). For the saturation solution, block-based Schwarz calculations are used for both MSFV and reference fine-scale simulation. Adaptive saturation calculation in MSFV is compared with corresponding fine-scale simulation using ILU as the preconditioner, which is quite effective for hyperbolic systems like the transport equation.

To study the efficiency of MSFV compared with other solvers, three schemes are compared. Namely,

- **fine\_ILU**: In the fine-scale simulation, ILU is used as the solver for both pressure and saturation solution;

- **fine\_AMG**: In the fine-scale simulation, AMG is used as the pressure solver, while ILU is used as the saturation solver;

- **MSFV**: In MSFV simulation, ILU is used as the solver for both the global coarse-scale pressure and adaptive local fine-scale solutions of both pressure and saturation.

The **fine\_ILU** scheme is not used any more in state-of-the-art reservoir simulators since ILU cannot provide an efficient pressure solution using a reasonable computational effort in most cases. Thus, it is included here only for comparison purposes. For the **fine\_AMG** scheme, the solvers used (AMG for pressure and ILU for saturation) are optimal choices for both steps of the sequential implicit scheme applied to fine-scale simulation. For the MSFV scheme, ILU is used for both the global coarse-scale and local fine-scale solutions. From some limited numerical tests, we found that
using AMG, instead of ILU, as the linear solver for the coarse-scale pressure system does not enhance the efficiency of MSFV further. This is possibly due to the fact that the global coarse-scale systems in MSFV are already quite small so that AMG does not lead to efficiency benefits when solving the system due to its higher setup cost compared with ILU. On the other hand, for the global pressure solution in fine-scale simulation, the system size is much larger than that of the coarse-scale pressure solution in MSFV, so that AMG can outperform ILU quite substantially.

2.3 Numerical Examples

In this section, we study the efficiency of MSFV using two numerical test cases. We studied two-phase flow and transport in two reservoir models:

- A heterogeneous reservoir model with moderate isotropic permeability correlation.
- A highly heterogeneous model with anisotropic permeability correlation lengths.

The permeability distribution was adapted from the top layer of the 10th SPE comparative solution project [7].

The fluids are assumed to be incompressible and the quadratic permeability model is used ($k_{ro} = S^2_o, \alpha = w, o$). The viscosity ratio between water and oil is 1:5 (unfavorable displacement).

2.3.1 Case 1: 2D, incompressible, heterogeneous

The first numerical example is for incompressible flow in a two dimensional (70 ft x 70 ft) heterogeneous reservoir. The permeability field is log-normal with a mean value of log-permeability of 4 and a variance of 2, and the spatial dimensionless correlation length is 0.2. The permeability is generated by the Sequential Gaussian Simulation Method ([8]), and the resulting field is depicted in Figure 2.3 (a). There is no-flow at all boundaries with an injector at the top left and a producer at the bottom right. Both wells are under flow-rate control (0.5 STB/day). In this case, a fine grid of
70 × 70, and a coarse grid of 10 × 10 are used. The coarsening ratio is 49 as each coarse cell contains 7 × 7 fine cells. Using a sequential fully implicit scheme, fine-scale (including fine_ILU and fine_AMG) and MSFV simulations are performed. The simulation results using the two fine-scale schemes are identical and are considered as the reference solution. The saturation distribution at 1.15 PVI from the reference fine-scale simulation results is shown in Figure 2.3 (b).

![Permeability field and fine-scale saturation distribution in case 1](image)

(a) Permeability  (b) Fine-scale saturation (true)

Figure 2.3: Permeability field and fine-scale saturation distribution in case 1

For the three simulation methods described in the last section, we first compare the total simulation time of these schemes in Figure 2.4. We can see that, the fine_ILU is the most time-consuming scheme. It takes fine_ILU more than one order of magnitude more compute time compared with the other two schemes. Furthermore, if we zoom in to compare the efficiency of fine_AMG and MSFV (in Figure 2.5), we see that the MSFV method is about 8 times faster than fine_AMG in terms of the total simulation time, which is quite encouraging.

We are also interested to see the proportions of time spent on computing the pressure, velocity, and saturation solutions in each simulation scheme. The results
Figure 2.4: Comparison of total simulation time between three scheme in case 1

Figure 2.5: Comparison of total simulation time between fine AMG and MSFV in case 1
are shown in Figure 2.6. We can see that for the pressure solution, the proportion of time in \textit{fine}_ILU is much larger than that in \textit{fine}_AMG. In fact, the saturation solutions using \textit{fine}_ILU and \textit{fine}_AMG consume similar efforts. The efficiency gain of \textit{fine}_AMG compared with \textit{fine}_ILU is due to AMG, which solves the pressure equation much more efficiently than ILU. What is more, for the MSFV method, the pressure solution is not the dominant part, and more simulation time is needed for computing the mass-conservative velocity field.

![Bar chart showing percentage of simulation time for each scheme in case 1](chart.png)

Figure 2.6: Comparison of percentage of simulation time for each scheme in case 1

In Figure 2.7, from the comparison of simulation time for the pressure solution, it is observed that the MSFV approach leads to significant efficiency gains over AMG (in \textit{fine}_AMG). This is rather encouraging since AMG is considered as a powerful linear solver for the pressure solution in general-purpose simulators. The efficiency advantage of MSFV over ILU for pressure solution is huge, which is too large to be shown here.

From the comparison of the saturation solution time in Figure 2.8, we can see that the computational efficiency in performing the saturation calculations in MSFV also
outperforms ILU (in \textit{fine\_AMG}), which is primarily due to adaptivity. Therefore, the efficiency gains in the total simulation of MSFV over \textit{fine\_AMG} are due to both the pressure and saturation solutions.

![Figure 2.7: Comparison of pressure time in case 1](image)

### 2.3.2 Case 2: 2D, incompressible, SPE10 top layer

For this numerical case, the permeability field (shown in Figure 2.9, \(2200ft \times 540ft\)) is from the top layer of the SPE 10 Problem \([7]\) with a fine grid of \(220 \times 54\) and a coarse grid of \(20 \times 6\). The coarsening factor is \(11 \times 9\). The boundary condition is no-flow at the top and bottom boundaries. Flow is driven under a fixed pressure difference between the left and right boundaries. Due to changes in the total-mobility field as water is injected, the production and injection rates also change with time. The saturation distribution obtained from the reference fine-scale simulation is shown in Figure 2.10.

MSFV and reference fine-scale simulations, including \textit{fine\_AMG} and \textit{fine\_ILU}, have been performed for this case. From the comparison of total simulation time
CHAPTER 2. EFFICIENCY STUDY OF THE MULTISCALE FINITE VOLUME METHOD

Figure 2.8: Comparison of saturation time in case 1

Figure 2.9: Permeability field in case 2

Figure 2.10: Fine-scale saturation distribution in case 2
plotted in Figure 2.11, it can be seen that fine_ILU is far less efficient than MSFV and fine_AMG. Figure 2.12 compares MSFV with fine-scale; MSFV is at least 2 times faster than fine_AMG in terms of total simulation time for this relatively challenging, but small, case.

![Comparison of total simulation time between three schemes in case 2](image)

Figure 2.11: Comparison of total simulation time between three schemes in case 2

The simulation times for the pressure and saturation solutions are shown in Figure 2.13 and 2.14 from which it is observed that MSFV is much more efficient than AMG (in fine_AMG) for the pressure solution, and MSFV is also more efficient than ILU (in fine_AMG) for the saturation solution.

From the efficiency comparisons of these numerical examples, we have shown that for different problem size and different boundary conditions (source/sink and Dirichlet boundary conditions), the MSFV model provides efficiency gains over conventional sequential fully-implicit fine-scale simulation.
CHAPTER 2. EFFICIENCY STUDY OF THE MULTISCALE FINITE VOLUME METHOD

Figure 2.12: Comparison of total simulation time between fine AMG and MSFV in case 2

Figure 2.13: Comparison of pressure time in case 2
Figure 2.14: Comparison of saturation time in case 2
Chapter 3

Multiscale Finite-Volume Based Upscaling

We describe an MSFV-based upscaling strategy for multiphase flow, and we compare this upscaling approach with other methods including the original MSFV. This work is motivated by the observation that the MSFV coarse-scale operator for the flow problem (pressure) can be seen as an upscaled transmissibility field that captures the essential fine-scale effects on the coarse pressure field quite accurately. Another motivation is that transport (i.e., solving the saturation equation) is highly nonlinear but is dominated by local behaviors. For this upscaling strategy, fine-scale velocities and saturations are reconstructed locally and adaptively, only when and where it is necessary (i.e., in the front region) via the MSFV method described in Chapter 2. The organization of this chapter is as follows. In Section 1, the approach and algorithms of this MSFV-based upscaling strategy are introduced. Numerical examples that demonstrate the accuracy and efficiency of this upscaling method for coupled flow and transport are presented in Section 2. We then compare the upscaling MSFV with the original MSFV in Section 3.
### 3.1 Adaptive Upscaling

In developing an upscaling strategy, the focus is on the quality of the coarse-scale solution. The adaptive strategy can be motivated using a typical Buckley-Leverett saturation profile at two times (see Figure 3.1).

![Upscaling region and algorithm](image)

**Figure 3.1: Upscaling region and algorithm**

The domain is divided into three regions.

- **Region (1):** ahead-of-the-front region that has not been invaded by the injected fluid during the $t_2 - t_1$ period
- **Region (2):** front region that has experienced significant saturation changes during the $t_2 - t_1$ period
- **Region (3):** behind-the-front region that have been swept by the injected fluid during the $t_2 - t_1$ period

In region (1) (ahead-of-the-front region), we do not solve for the fine-scale velocities and saturations. Only coarse-scale velocity and saturation solutions are calculated in this region. In region (2) (front region), the velocities and saturations show significant change as a function of time; therefore, the fine-scale velocity and saturation solutions are needed to obtain high-quality coarse-scale velocity and saturation solutions. In this region, after solving for the coarse-scale velocities and saturations, MSFV is
applied to reconstruct the fine-scale velocities and saturations using the prolongation operators we described in the previous chapter. Then, accurate predictions of the coarse-scale velocities and saturations can be obtained by averaging this approximate fine-scale information. In region (3) (behind-the-front region), an accurate and computationally expensive solution (like that in region (2)) is not necessary, since saturation changes during the time interval $t_2 - t_1$ are small. So, only coarse-scale velocities and saturations are calculated.

Compared with the original MSFV described in Chapter 2, if accurate coarse-scale solutions are the target, this adaptive upscaling strategy is expected to further enhance the overall efficiency by solving for fine-scale velocities and saturations in regions (1) and (3) only.

### 3.2 Coarse-Scale Based Transition Criteria

As we described earlier, different algorithms are used in different regions. These regions correspond to (ahead-of-the-front, front, and behind-the-front regions). The designation of which region a coarse cell belongs to is determined dynamically according to the evolution of the flow field and the local saturation information. A set of coarse-scale-based criteria is designed to capture the transition between the three regions. Specifically,

- Region (1) to Region (2): $\Delta t S_j^H > \triangle_1$,
- Region (2) to Region (3): $\Delta t S_j^H < \triangle_2$ and $\Delta t S_j^H < (\Delta t S_j^H)_{prev}$,

where $\Delta t S_j^H$ is the difference of saturation in coarse block $j$ between the previous and current time steps, and $\triangle_i$ ($i = 1, 2$) are specified tolerances. These tolerances are case dependent and are chosen based on computational experience with the particular problem of interest. The specific choice is attempts to balance accuracy requirements with computational cost.
3.3 Numerical Examples

In this section, numerical examples are explored to investigate the accuracy and efficiency of the MSFV-based upscaling strategy. Three reservoir models are studied here:

- 2D heterogeneous reservoir model with moderate isotropic permeability correlation.
- 2D highly heterogeneous media with a large anisotropic permeability correlation length; the permeability distribution was adapted from the top layer of 10th SPE Comparative Solution Project Model in [7].
- 3D heterogeneous reservoir model with moderate isotropic permeability correlation lengths.

3.3.1 Case 1: 2D, incompressible, heterogeneous

The field setting and fluid properties for the first numerical case here are exactly identical to the first case discussed in the efficiency study of Chapter 2. The permeability field is shown in Figure 3.2 (a). The reservoir is initially fully saturated with oil, and water is injected to displace the oil. There is an injector at the top left, and a producer at the bottom right, both under flow-rate control with no-flow conditions at all the domain boundaries. The fine-scale grid, $70 \times 70$, is uniformly coarsened into a $10 \times 10$ coarse-scale grid, so the upscaling factor is 49. The flow rates for the injector and producer are uniformly distributed in the coarse cells, i.e., injection in the top-left coarse cell and production in the bottom-right coarse cell. We simulate this case using both MSFV-based upscaling and reference fine-scale simulation. In the fine-scale simulation, AMG is used for the pressure solution, and ILU is used for the saturation solution (same as the fine-AMG scheme of Chapter 2).

Figure 3.2 (b) is the fine-scale saturation map from a reference fine-scale simulation result at 1.15 PVI. Figure 3.2 (c) is the coarse-scale saturation map obtained by volume averaging of the saturations in all the fine cells contained in a coarse block,
and this coarse-scale solution is taken as the reference to which we compare the MSFV-based upscaling method. Note that the reference solution, in an average sense, captures the fine-scale effects and reflects the complex structure of the saturation distribution in the heterogeneous domain.

Figure 3.2: Fine-scale simulation result at \( t = 1.15 \) PVI in case 1

Figure 3.3 shows the results from the MSFV-based upscaling approach at 0.11 and 1.15 PVI. The first two columns in Figure 3.3 represent the areas where fine-scale variables are reconstructed dynamically in order to update (improve) the upscaled model. Specifically, the first column is for the reconstruction of velocity, while the second column is for the reconstruction of saturation. The fine-scale velocities and saturations are obtained using full reconstruction or linear interpolation.

Lee et al. [21] proposed an algorithm and associated labeling criteria for adaptive calculation of fine-scale velocities and saturations. The third-column maps show the coarse-scale saturation obtained using the MSFV-based upscaling method at 0.11 PVI and 1.15 PVI. In the first two columns, dark blue represents region (1) (ahead-of-the-front region), yellow and light blue represent region (2) (front region), and dark red represents region (3) (behind-the-front region). In the yellow region, full reconstruction (nonlinear operator) is applied for velocity/saturation, while in the light-blue region, linear interpolation (linear operator) is used for velocity/saturation.

It is observed that the domain with fine-scale reconstruction of variables increases
Figure 3.3: Upscaled results in case 1
as the saturation front spreads. As the front moves into a new area, nonlinear reconstruction of the fine-scale variables is generally needed. Once the history of saturation development is well-established and the time changes of the variable become small, linear reconstruction of the fine-scale solution is usually sufficient. In addition, when the area is completely swept by the invading fluid (e.g., the trailing expansion wave in the Buckley-Leverett profile), an accurate coarse-scale representation can be obtained without reconstruction of fine-scale variables. Furthermore, the shape and extent of different regions (ahead-of-the-front, front, and behind-the-front) are consistent with the corresponding coarse-scale saturation profile. Efficiency gains can be obtained in the ahead-of-the-front (dark blue) and behind-the-front (dark red) regions compared with the original MSFV, since the fine-scale velocities and saturations are not reconstructed in those regions.

If we compare the coarse-scale saturation map from the MSFV-based upscaling strategy at 1.15 PVI (bottom-right map in Figure 3.3) with the reference solution in Figure 3.2 (c), it is clear that the two solutions are very close. The upscaled result captures the detail of the front shape reasonably well. The oil recovery and oil cut for this case are shown in Figure 3.4. The solution from the MSFV-based upscaling method is in excellent agreement with the fine-scale results.

The error norms of the coarse-scale pressure, $e_{pc}$, and coarse-scale saturation, $e_{satc}$, are listed in Table 3.1. These error norms (L-2 norms) are defined as:

$$e_{pc} = \frac{\|p_{ms}^H - p_f^H\|_2}{\|p_{init}\|_2},$$

$$e_{satc} = \|S_{ms}^H - S_f^H\|_2.$$  (3.2)

From Table 3.1 it can be observed that the errors in the coarse-scale pressure are only on the order of $10^{-3}$, while those for the coarse saturation are only on the order of $10^{-2}$. Note that the error norms are sensitive to the criteria (tolerance values) for the transition between the different regions. The efficiency comparison for this case is shown in Table 3.2, which indicates that the MSFV-based upscaling treatment is at least one order of magnitude faster than the fine-scale simulation. Note that in the
Figure 3.4: Cumulative oil recovery and oil fraction in production in case 1

Table 3.1: Error norms in case 1

<table>
<thead>
<tr>
<th>( t(\text{PVI}) )</th>
<th>( e_{pc} )</th>
<th>( e_{satc} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.11</td>
<td>0.00216</td>
<td>0.0145</td>
</tr>
<tr>
<td>0.29</td>
<td>0.00227</td>
<td>0.0326</td>
</tr>
<tr>
<td>1.15</td>
<td>0.00153</td>
<td>0.0162</td>
</tr>
</tbody>
</table>
fine-scale simulations presented here, AMG and ILU are used to solve the pressure and saturation equations, respectively.

<table>
<thead>
<tr>
<th>t(PVI)</th>
<th>fine-scale (s)</th>
<th>upscaling</th>
<th>MSFV (s)</th>
<th>times faster</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.11</td>
<td>19.0</td>
<td>0.67</td>
<td>24.2</td>
<td></td>
</tr>
<tr>
<td>0.29</td>
<td>31.4</td>
<td>2.12</td>
<td>14.8</td>
<td></td>
</tr>
<tr>
<td>1.15</td>
<td>92.9</td>
<td>7.45</td>
<td>12.5</td>
<td></td>
</tr>
</tbody>
</table>

### 3.3.2 Case 2: 2D, incompressible, SPE10 top layer

The setting of the second numerical case is also identical to that used in the efficiency study of Chapter 2. The permeability field is adapted from the top layer of SPE 10 with a fine grid of $220 \times 54$ and a coarse grid of $20 \times 6$. The initial pressure is 4000 psi. The left boundary is kept at a constant pressure of 4000 psi with water injection, and the right boundary is kept at a constant pressure of 1000 psi with reservoir fluid production. Flow is driven by the pressure difference between the left and right boundaries. For this case, a characteristic time is defined as:

$$\tau_0 = \frac{\phi \mu_o L_x^2}{||p_{left} - p_{right}|| k},$$

where $k$ is the geometric mean of the permeability field, $\mu_o$ is the viscosity of oil phase, and $L_x$ is the model size in the x-direction.

The permeability field for this case is shown in Figure 3.5. The fine-scale saturation maps from reference fine-scale simulation results at $1.8\tau_0$ and $5.9\tau_0$ are presented in Figures 3.6 and 3.7, respectively. The saturation distribution clearly indicates that the underlying distribution of heterogeneity results in a complex saturation distribution.

The MSFV-based upscaling results at $t = 1.8\tau_0$ are shown in Figure 3.9. Compared with the reference coarse-scale solution (volume averaged from the reference fine-scale solution) in Figure 3.8 the coarse-scale saturation distribution from the MSFV-based upscaling method is quite close. The domains where a saturation reconstruction
CHAPTER 3. MULTISCALE FINITE-VOLUME BASED UPSCALING

Figure 3.5: Permeability field in case 2

Figure 3.6: Fine-scale saturation at $1.8 \tau_0$ in case 2

Figure 3.7: Fine-scale saturation at $5.9 \tau_0$ in case 2
strategy is used in the upscaled model are plotted in Figure 3.10. Due to the strong heterogeneity, the reconstruction of the fine-scale saturation entails the nonlinear operator in MSFV for most of the domain behind the front.

Figure 3.8: Volume averaged fine-scale saturation (reference) at 1.8 $\tau_0$

Figure 3.9: MSFV coarse-scale saturation at 1.8 $\tau_0$

Figure 3.10: Region and Algorithm at 1.8 $\tau_0$

In addition, the upscaled results at $t = 5.9\tau_0$ are depicted in Figures 3.11, 3.12, and 3.13. In Figure 6.12, it is observed that more of the behind-the-front region (dark red) appears than at $t = 1.8\tau_0$.

A comparison of cumulative oil recovery and oil fraction is presented in Figure 3.14. The results demonstrate the accuracy of the MSFV-based upscaling method for this case. Clearly the upscaled model yields an accurate prediction of the global integral performance of the reservoir model.
CHAPTER 3. MULTISCALE FINITE-VOLUME BASED UPSCALING

Figure 3.11: Volume averaged fine-scale saturation (reference) at $5.9 \tau_0$

Figure 3.12: MSFV Coarse-scale saturation at $5.9 \tau_0$

Figure 3.13: Region and Algorithm at $5.9 \tau_0$
Figure 3.14: Cumulative oil recovery and oil fraction in production in case 2
The error norms of the coarse-scale pressure and saturation fields in Table 3.3 show that for this relatively challenging case, the errors are on the order of $10^{-3}$ for pressure and $10^{-2}$ for saturation. The accuracy is quite satisfactory given the high degree of heterogeneity and complex correlation structures.

<table>
<thead>
<tr>
<th>$t(\tau_0)$</th>
<th>$\epsilon_{pc}$</th>
<th>$\epsilon_{suc}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.80</td>
<td>0.00654</td>
<td>0.0425</td>
</tr>
<tr>
<td>3.84</td>
<td>0.00623</td>
<td>0.0270</td>
</tr>
<tr>
<td>5.88</td>
<td>0.00625</td>
<td>0.0231</td>
</tr>
</tbody>
</table>

The efficiency comparison for this case is shown in Table 3.4, which indicates that the upscaled MSFV method is about 2.5 times more efficient than the original (fine-scale) MSFV approach for this relatively challenging case.

<table>
<thead>
<tr>
<th>$t(\tau_0)$</th>
<th>fine-scale (s)</th>
<th>upscaling MSFV (s)</th>
<th>times faster</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.80</td>
<td>379</td>
<td>113</td>
<td>3.35</td>
</tr>
<tr>
<td>3.84</td>
<td>758</td>
<td>271</td>
<td>2.78</td>
</tr>
<tr>
<td>5.88</td>
<td>1108</td>
<td>435</td>
<td>2.55</td>
</tr>
</tbody>
</table>

### 3.3.3 Case 3: 3D, compressible, heterogeneous

For the last numerical case, we test the MSFV upscaling strategy for compressible flow in a 3D domain. The permeability field ($210 \text{ ft} \times 210 \text{ ft} \times 210 \text{ ft}$) is log-normal with a mean of log-permeability of 4 and a variance of 2. The permeability is generated by the Sequential Gaussian Simulation Method ([8]), and the resulting distribution is shown in Figure 3.15. There is an injector and a producer at opposite corners of the domain, both under flow-rate control (200 STB/day). The reservoir is initially saturated with oil, and water is injected. Gravity is neglected. In this case, the fine
CHAPTER 3. MULTISCALE FINITE-VOLUME BASED UPSCALING

grid is $30 \times 30 \times 30$, and the coarse grid is $6 \times 6 \times 6$. The upscaling factor is $5 \times 5 \times 5$. We simulate this model using both MSFV-based upscaling and reference fine-scale simulation. AMG and ILU are used, respectively, to solve the pressure and saturation equations in the fine-scale simulation.

We simulate this model using both MSFV-based upscaling and reference fine-scale simulation. AMG and ILU are used, respectively, to solve the pressure and saturation equations in the fine-scale simulation.

The fine-scale saturation profile at 0.061 PVI, as well as, the corresponding averaged coarse-scale saturation profile (reference solution) are represented in Figure 3.16. The upscaled result from the MSFV-based upscaling strategy is shown in Figure 3.17. In Figure 3.17, plot (a) is the upscaled coarse-scale saturation map, and plot (b) shows the domain, where the fine-scale saturations are reconstructed to improve the coarse-scale model.

From the comparison between Figure 3.16 (b) and Figure 3.17 (a), it can be seen that the coarse-scale saturation map from the upscaling method agrees with the reference solution quite well. So, the upscaling strategy captures the complex saturation structure and flow history on the coarser scale quite well for compressible flow in this 3D reservoir model. In addition, from the regions where reconstruction of the fine-scale saturations is used (3.17 (b)), we can see that some regions with sharp saturation gradients (yellow) appear around the injector as water is injected into the

Figure 3.15: Permeability distribution in case 3
CHAPTER 3. MULTISCALE FINITE-VOLUME BASED UPSCALING

Figure 3.16: Fine-scale simulation result at 0.061 PVI in case 3

Figure 3.17: MSFV-based upscaling simulation result at 0.061 PVI in case 3
reservoir. The rest of the domain is in the ahead-of-the-front region (dark blue). Note that in the ahead-of-the-front region, fine-scale saturations are not reconstructed, because the obtained coarse-scale prediction with only coarse-scale operators using MSFV is accurate enough. So, compared with the original MSFV, further efficiency gains can be obtained in the dark blue region, which is dominant in the flow scenario in this case.

Results from the reference fine-scale simulation at $t = 0.728 \text{PVI}$ are presented in Figure 3.18 while the solution from the MSFV-based upscaling method is shown in Figure 3.19. Again, the agreement between the coarse-scale saturation map and the reference solution shows that the upscaling strategy is quite effective for compressible flow in 3D, at least for this case. At this late time, water has broken through, and some behind-the-front regions (dark red in Figure 3.19 (b)) appear around the injector.

Error norms in Table 3.5 provide evidence that for this relatively larger problem in 3D, the MSFV-based upscaling method yields coarse-scale solutions that are quite accurate. Table 3.6 shows that compared with reference fine-scale simulation, the MSFV-based upscaling strategy is at least six times faster for this case. This speedup
is rather encouraging, since powerful solvers as AMG and ILU are used in the fine-scale simulation.

Table 3.5: Error norms in Case 3

<table>
<thead>
<tr>
<th>t(PVI)</th>
<th>$e_{pc}$</th>
<th>$e_{satc}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.061</td>
<td>0.00047</td>
<td>0.0131</td>
</tr>
<tr>
<td>0.243</td>
<td>0.00381</td>
<td>0.0232</td>
</tr>
<tr>
<td>0.728</td>
<td>0.04578</td>
<td>0.0325</td>
</tr>
</tbody>
</table>

### 3.4 Comparison with Original MSFV

If coarse-scale solutions are the target, compared with original MSFV, the upscaling MSFV leads to further efficiency gains while sacrificing some accuracy. The question is whether it is worth using the MSFV framework as an upscaling strategy to construct high-quality coarse solutions.

To investigate this problem, three different schemes are performed using a large
Table 3.6: Efficiency comparison in Case 3

<table>
<thead>
<tr>
<th>t(PVI) fine-scale (s)</th>
<th>upscaling MSFV (s)</th>
<th>times faster</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.061 222</td>
<td>19.1</td>
<td>11.6</td>
</tr>
<tr>
<td>0.243 743</td>
<td>97.3</td>
<td>7.63</td>
</tr>
<tr>
<td>0.728 2203</td>
<td>328</td>
<td>6.72</td>
</tr>
</tbody>
</table>

homogeneous case with a fine grid of $140 \times 140$ and a coarse grid of $20 \times 20$, namely, original, upscaling, and coarse MSFV. Their definitions are as follows:

- **Original MSFV**: The whole domain is considered as part of the front region, where fine-scale variables (velocities and saturations) are reconstructed in addition to the coarse-scale variables. Therefore, the fine-scale solution is reconstructed everywhere in the domain.

- **Upscaling MSFV**: The domain is divided into ahead-of-the-front, front, and behind-the-front regions. Different transition criteria between these regions would result in different accuracy and computational efforts.

- **Coarse MSFV**: The whole domain is set as a behind-the-front region, where only coarse-scale variables (velocities and saturations) are calculated. No fine-scale solutions are reconstructed.

In Figures 3.20 and 3.21, the error norms for the coarse-scale pressures and saturations are shown versus the computational time for the simulations. The definitions of these error norms, $e_{pc}$ and $e_{satc}$, are the same as those used earlier (see Equation 3.1 and 3.2). The green dot represents the result from coarse MSFV, which has the highest efficiency. However, the error norm is very large (nearly 70% in pressure and 25% in saturation), which is totally unacceptable. The red dot represents the result of the original MSFV method, which has the highest accuracy, but also uses the largest simulation time. The straight line in each figure connects the simulation results of the original and coarse MSFV methods. It can be observed in both figures that all the simulation results of upscaling MSFV are below the straight line. Therefore, we claim...
Figure 3.20: Trade-off between coarse-scale pressure error and efficiency
Figure 3.21: Trade-off between coarse-scale saturation error and efficiency
that the MSFV-based upscaling achieves a reasonable balance between accuracy and efficiency. Further work is needed to quantify the errors based on theoretical analysis, or perhaps using dynamic a-posteriori error estimates during a simulation run.
Chapter 4

Summary and Future Work

Summary

In this report, we studied the computational efficiency of the multiscale finite-volume (MSFV) method. We studied two variants of MSFV. The first is the adaptive MSFV scheme aimed at solving the flow and transport problems on the reference fine-scale model. The second is the MSFV-based upscaling approach, where the focus is on obtaining high-quality coarse-scale solutions of coupled flow and transport.

The MSFV method is computationally more efficient than conventional fine-scale sequential simulation with state-of-the-art solvers. Adaptivity in the computation of the basis functions leads to significant efficiency gains over AMG (Algebraic Multi-Grid), which is a powerful solver for the pressure solution. AMG solves for the pressure in every grid block for each time step, while in MSFV, most of the basis functions are reused from the last time step, and the pressure solution is only updated in a few limited regions during the current time step.

Adaptive saturation calculation in MSFV is also more efficient than ILU in the sequential fine-scale approach, which is widely used for solving the saturation equations. This gain is not as large as that for the pressure solution in the numerical cases presented here. However, we expect the efficiency gain in the saturation solution to grow for large-scale problems and become quite significant for compositional models, where the transport problems dominate the overall effort.
The MSFV-based upscaling strategy is considered as an extension of the original MSFV. Instead of insisting on computing the fine-scale solution, coarse-scale approximations are the target. This upscaling strategy is based on decomposing the domain into different regions based on the change of the saturation field over a time step (or iteration), namely, (1) ahead-of-the-front, (2) front, and (3) behind-the-front regions. For the first and third regions, where the flow field and saturations change slowly, only coarse-scale velocities and saturations are calculated. As for the front region, where the changes are large, local fine-scale velocities and saturations are reconstructed using special prolongation operators, so that better accuracy can be achieved for the updated coarse-scale model. A set of coarse-scale criteria is designed to capture the transitions between the different regions.

The accuracy and efficiency of this upscaling method are demonstrated using several numerical examples. It is observed that the MSFV upscaling strategy provides coarse-scale predictions that are in excellent agreement with reference solutions. Moreover, the MSFV-based upscaling simulations are obtained with significant efficiency gains compared with fine-scale simulations. This MSFV upscaling method is more efficient than the original MSFV and achieves a reasonable balance between accuracy and efficiency.

**Future Work**

For future work, one of the most promising directions is to develop a general linear-solver methodology based on MSFV ideas. Chapter 2 of this report provided a preliminary investigation of the potential and possibility of this future work through a comparison between MSFV and linear solvers in state-of-the-art simulators.

For the MSFV framework itself, there are also several possible future directions, including extensions to unstructured grids, compositional models, and thermal problems. Multiscale modeling of strong capillarity in heterogeneous reservoirs is also an interesting and challenging area. Furthermore, theoretical analysis of the errors associated with the MSFV approach for coupled flow and transport is desirable.
Appendix: MSFV C++ Simulator

In this appendix, the pseudo code of the C++ MSFV research simulator and the code structures for both the reference fine-scale and MSFV simulations are described.

A sequential fully implicit scheme for MSFV simulation is shown in the pseudo code. For each time step, there are several loops. Both fine-scale and coarse-scale pressures are solved in the pressure inner loop; then the total velocities are computed; finally saturations are solved both on coarse and fine scales, in the saturation inner loop. The implicitness of this scheme guarantees the stability of the results, while the sequential coupling strategy takes advantage of the excellent performance of the prolongation and restriction operators for pressure and saturation, respectively, in MSFV.
/* START OF SIMULATION */

n = 1

do

/** TIME STEP LOOP **/
iter = 1: \((S^H)^{iter} = (S^H)^n\), \((S^h)^{iter} = (S^h)^n\), \((p^H)^{iter} = (p^H)^n\), \((p^h)^{iter} = (p^h)^n\)

do

/** SYSTEM OUTER LOOP **/
iterp = 1: \((p^H)^{iterp} = (p^H)^{iter}\), \((p^h)^{iterp} = (p^h)^{iter}\)

do

/** PRESSURE INNER LOOP **/
update basis functions
calculate coarse-scale transmissibilities
solve linearized coarse-scale pressure equation \(\Rightarrow (p^H)^{iterp+1}\)
pressure prolongation operator \(\Rightarrow (p^h)^{iterp+1}\)
iterp = iterp + 1

until (convergence of nonlinear pressure equation)
\((p^H)^{iter+1} = (p^H)^{iterp}\), \((p^h)^{iter+1} = (p^h)^{iterp}\)
solve coarse-scale velocity \(\Rightarrow (u^H)^{iter+1}\)
adaptive prolongation operator for fine-scale velocity \(\Rightarrow (u^h)^{iter+1}\)

iters = 1: \((S^H)^{iters} = (S^H)^{iter}\), \((S^h)^{iters} = (S^h)^{iter}\)

do

/** SATURATION INNER LOOP **/
solve coarse-scale transport equations \(\Rightarrow (S^H)^{iters+1}\)
adaptive prolongation operator for fine-scale saturation \(\Rightarrow (S^h)^{iters+1}\)

iters = iters + 1

until (convergence of nonlinear transport equations)
\((S^H)^{iter+1} = (S^H)^{iters}\), \((S^h)^{iter+1} = (S^h)^{iters}\)
iter = iter + 1

until (convergence of system of pressure and transport equations)
\((p^H)^{n+1} = (p^H)^{iter}\), \((p^h)^{n+1} = (p^h)^{iter}\), \((S^H)^{n+1} = (S^H)^{iter}\), \((S^h)^{n+1} = (S^h)^{iter}\)
n = n + 1

until (Simulation done)
The function call sequence is presented in Figure 1-2. Figure A.1 shows the code structure for the fine-scale simulation, where \texttt{nl\_fine\_cal} is the main function. It calls many other functions as listed in the figure. Some of these functions (in pink boxes) are shared with the multiscale simulation scheme. In Figure 2, the code structure of the multiscale simulation is shown. For MSFV simulation, \texttt{nl\_ms\_coarse} is the main function, which calls many more functions compared with the fine-scale simulation. These functions may also call more functions, i.e., there are several call levels. Some functions in the MSFV simulation (in pink boxes) are the same as those in the fine-scale simulation. The reason for sharing some common functions between the fine-scale and multiscale simulators is that we are trying to compare the two simulation methods in a fair and meaningful manner. We would like to make sure that the efficiency differences between the two types of simulation are due to the algorithms and specific linear solvers rather than the implementation techniques. So some functions, including initialization and property calculations are shared by both simulation methods.
Functions only for fine-scale simulation

Functions for both fine- and multi-scale simulations

Figure 1: Code structure of fine-scale simulation
Functions only for multiscale simulation

Functions for both fine- and multi-scale simulations

Figure 2: Code structure of multiscale simulation
To clearly explain the usage of these functions, descriptions of all the functions appearing in these code structures are listed in Table 1-3. Specifically, Table 1 shows the functions for fine-scale simulation; Table 2 shows the functions shared by fine-scale and multiscale simulations; Table 3 shows the functions for multiscale simulation.

Table 1: Functions for fine-scale simulation

<table>
<thead>
<tr>
<th>Functions</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CalFineOperator</td>
<td>Generate fine-scale pressure system</td>
</tr>
<tr>
<td>CalFineVelo</td>
<td>Calculate fine-scale total velocity</td>
</tr>
<tr>
<td>CalPhaseVelo</td>
<td>Calculate fine-scale phase velocity</td>
</tr>
<tr>
<td>CheckConverge1</td>
<td>Check system convergence based on residual</td>
</tr>
<tr>
<td>nL_fine_cal</td>
<td>Primary function to perform fine-scale simulation</td>
</tr>
<tr>
<td>SolveLocalT rpt2</td>
<td>Solve local transport problem</td>
</tr>
<tr>
<td>SolveSchwT rpt</td>
<td>Solve transport system using Schwaz overlapping method</td>
</tr>
</tbody>
</table>

Table 2: Functions for both fine- and multi-scale simulations

<table>
<thead>
<tr>
<th>Functions</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CalFracFlow_o</td>
<td>Calculate the fine-scale fractional flow for each cell</td>
</tr>
<tr>
<td>GetRelPerm</td>
<td>Calculate relative permeability and its derivative for a cell</td>
</tr>
<tr>
<td>InitField</td>
<td>Field initialization (pressure and saturation)</td>
</tr>
<tr>
<td>UpdateMob</td>
<td>Update mobility with upstream weighting</td>
</tr>
<tr>
<td>UpdateMobChk</td>
<td>Calculate the total transmissibility for each cell</td>
</tr>
<tr>
<td>UpdatePProp2</td>
<td>Update the pressure dependent properties b, phi and their derivatives</td>
</tr>
<tr>
<td>UpdateTrans</td>
<td>Update transmissibility with upstream weighting</td>
</tr>
<tr>
<td>UpdateUpstream</td>
<td>Update upstream direction of a phase (without tolerance) with gravity</td>
</tr>
<tr>
<td>UpdateUpstream2</td>
<td>Update upstream direction of a phase (with tolerance) with gravity</td>
</tr>
</tbody>
</table>
### Table 3: Functions for multiscale simulation

<table>
<thead>
<tr>
<th>Functions</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CalCrsSat</td>
<td>Calculate coarse-scale saturations by volume average fine-scale saturations</td>
</tr>
<tr>
<td>CalDualPressure</td>
<td>Reconstruct fine-scale pressure within each dual cell using basis functions</td>
</tr>
<tr>
<td>CalFineOperator_ms1</td>
<td>Generate fine-scale pressure system</td>
</tr>
<tr>
<td>CalFracFlow</td>
<td>Calculate fine-scale fractional flow within each coarse cell</td>
</tr>
<tr>
<td>CalLFPhaseVelo</td>
<td>Calculate fine-scale phase velocities and their derivatives within each coarse cell</td>
</tr>
<tr>
<td>CalLFVelo</td>
<td>Calculate fine-scale velocities within each coarse cell</td>
</tr>
<tr>
<td>CalSatXi</td>
<td>Calculate Xi</td>
</tr>
<tr>
<td>CalTcCcRc</td>
<td>Calculate Tc, Cc and Rc</td>
</tr>
<tr>
<td>CalTfCfRf</td>
<td>Calculate Tf, Cf and Rf</td>
</tr>
<tr>
<td>CalVc</td>
<td>Calculate coarse-scale velocities</td>
</tr>
<tr>
<td>FixCrsPrs</td>
<td>Fix coarse-scale pressure at some cell to meet boundary condition</td>
</tr>
<tr>
<td>Fraction_Coarse</td>
<td>Calculate coarse-scale fractional flow</td>
</tr>
<tr>
<td>MapFine2Dual</td>
<td>Map fine properties to dual grid properties</td>
</tr>
<tr>
<td>nl_ms_coarse</td>
<td>Primary function to perform multiscale simulation</td>
</tr>
<tr>
<td>ProlongSat</td>
<td>Reconstruct fine-scale saturations via prolongation operators</td>
</tr>
<tr>
<td>ProlongVel</td>
<td>Reconstruct fine-scale velocities via prolongation operators</td>
</tr>
<tr>
<td>ReconstructVelo</td>
<td>Reconstruct fine-scale velocities with linear interpolation (linear operator)</td>
</tr>
<tr>
<td>SetUpdFlag</td>
<td>Set flag at dual coarse-cell for updating basis functions</td>
</tr>
<tr>
<td>SetupLFPhaseVelo</td>
<td>Set up the local fine-scale and boundary phase velocity for each coarse cell</td>
</tr>
<tr>
<td>SetupLFVelo</td>
<td>Set up the local fine-scale and boundary total velocity for each coarse cell</td>
</tr>
<tr>
<td>SolveCrsTrpt1</td>
<td>Solve global coarse-scale transport equation</td>
</tr>
<tr>
<td>SolveDualBasis_M</td>
<td>Solve dual basis function based on reduced boundary condition and elliptical PDE</td>
</tr>
<tr>
<td>SolveGravity</td>
<td>Solve gravity pressure with the same assumption as basis function</td>
</tr>
<tr>
<td>SolveLocalGravityPrs</td>
<td>Solve gravity pressure based on reduced boundary condition and elliptical PDE</td>
</tr>
<tr>
<td>SolveLocalPrs_v</td>
<td>Solve local fine-scale conservative pressure within each coarse cell</td>
</tr>
<tr>
<td>SolveMsLocalTrpt</td>
<td>Implicitly solve local transport equation for each coarse cell</td>
</tr>
<tr>
<td>SolveMultiDual</td>
<td>Solve dual basis functions</td>
</tr>
<tr>
<td>UpdateGravChk</td>
<td>Update the divergence of gravity in each fine cell</td>
</tr>
<tr>
<td>Upstream_ms_ving</td>
<td>Update upstream direction of a phase (with tolerance) without gravity</td>
</tr>
</tbody>
</table>
Bibliography


