ALGEBRAIC MULTISCALE LINEAR SOLVER FOR HETEROGENEOUS ELLIPTIC PROBLEMS

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

Yixuan Wang
June 2012
I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and quality, as partial fulfillment of the degree of Master of Science in Petroleum Engineering.

(Hamdi Tchelepi)  Principal Adviser
Abstract

An Algebraic Multiscale Solver (AMS) for the pressure system of equations arising from incompressible flow in heterogeneous porous media is developed. The algorithm allows for several independent preconditioning stages to deal with the full spectrum of errors. In addition to the fine-scale system of equations, AMS requires information about the superimposed (dual) coarse grid, which is used for the construction of a wirebasket ordering. The primal coarse grid is used in the construction of a conservative coarse-scale operator and in the reconstruction of a conservative fine-scale velocity field as the last step of the solution process. The convergence properties of AMS are studied for various combinations including (1) the MultiScale Finite-Element (MSFE) method, (2) the MultiScale Finite-Volume (MSFV) approach, (3) Correction Functions (CF), (4) Block ILU with zero fill-in (BILU), and (5) point-wise ILU with zero fill-in (ILU). The reduced-problem boundary condition, which is used for localization, is investigated and improvements are proposed. For a wide range of test cases, including the highly heterogeneous (with over a million cells) SPE 10 permeability field, the performance of the different preconditioning options is analyzed. It is found that the best overall performance is obtained by combining MSFE and ILU as the global and local preconditioners, respectively. Comparison between AMS and the widely used SAMG solver illustrates that AMS and SAMG are comparable, especially for very large heterogeneous problems.
Acknowledgements

I would like to express my special thanks to my advisor, Professor Hamdi Tchelepi, for his constant support, patience and encouragement in my academic life. Without his profound insight and masterful guidance, this work would not be possible. I have learned a great deal from him not only in terms of knowledge, but also in the ways to perform research. I am so honored that I can continue my research under his supervision.

Also, my deep gratitude is owed to Dr. Hadi Hajibeygi, the post-doc in my research group, for providing me with a comprehensive understanding of multiscale simulation techniques. His encouragement and advice motivate me to resolve the challenges I encountered in this project. Dr. Hajibeygi has taught me a lot of things, such as presentation and paper writing skills. His great help and friendship are gratefully acknowledged.

I wish to thank Dr. Hui Zhou for providing the initial code and replying to my questions via emails and calls. Many thanks to Yifan Zhou for helping me set up the SAMG parameters used for comparison purposes in this project. In addition, I particularly appreciate Xiaochen Wang for her thoughtful discussions and kindly encouragement. Thanks to all colleagues and friends who make my life enjoyable at Stanford.

I would like to thank Dr. Seong Lee and Dr. Bradley Mallison of Cheveron ETC and Dr. Zhuoyi Li of Schlumberger for their constructive discussions.
Special thanks to my parents for their unconditional love and support throughout my life and encouraging me to do my best. I benefit greatly by their education from the very beginning.

Last but not least, financial support from the Stanford University Petroleum Research Institute (SUPRI-B) and the Department of Energy Resources Engineering are gratefully acknowledged.
Contents

Abstract iii

Acknowledgements v

1 Introduction 1

2 Algebraic Multiscale Solver 5
   2.1 Governing Equations ........................................... 5
   2.2 Algebraic Description .......................................... 8
   2.3 Fine-scale Velocity Reconstruction ............................ 15

3 Analysis of the Correction Function 19
   3.1 Independent Local Stage ........................................ 19
      3.1.1 Case 1: Homogeneous permeability ....................... 20
      3.1.2 Case 2: Heterogeneous permeability ..................... 21
   3.2 Spectral Analysis ................................................ 22
   3.3 Sensitivity to Large Permeability Contrasts ................. 25
   3.4 Computational Cost ............................................. 28
   3.5 Summary .......................................................... 29

4 Results 31
4.1 AMS Components .............................................. 31
  4.1.1 Restriction Scheme ...................................... 31
  4.1.2 Local Boundary Condition ............................... 33
  4.1.3 Local Preconditioner .................................... 33
4.2 AMS Performance .............................................. 35
  4.2.1 Comparison with SAMG .................................. 35
  4.2.2 Scalability Analysis ..................................... 39

5 Summary and Future Work ................................. 41

Bibliography ...................................................... 43
List of Tables

3.1 CPU time (sec) and iteration steps. .............................. 29

4.1 CPU time (sec) and iteration steps for SAMG. .................. 38

4.2 CPU time (sec) and iteration steps for AMS. ...................... 39
List of Figures

2.1 Primal (bold solid black) and dual (dashed blue) coarse grid cells. Fine-cells belonging to a coarse control volume and a dual coarse cell are highlighted in green and light orange, respectively. ................................................. 6

2.2 Multiscale grid with wirebasket ordering based on dual coarse cells. .................................................. 9

2.3 Permeability field, fine-scale reference and MSFV pressure solution. .................................................. 17

2.4 Velocity divergence before and after reconstruction. ............................................................................. 18

3.1 Comparison between the solutions obtained from fine-scale reference, MSFV, MSFV with CF and MSFV with BILU. Note that all solutions are conservative at the coarse-scale. .................................................. 21

3.2 Natural logarithm of layered permeability filed with $\psi_1 = 0.5$ and $\psi_2 = 0.02$. Fine grid size is $100 \times 100$ and coarse grid size is $10 \times 10$. .................................................. 23

3.3 Comparison between the solutions obtained from fine-scale reference, MSFV, MSFV-CF and MSFV-BILU. ......................................................................................... 23

3.4 Iteration histories for MS with and without CF. Note that BILU is used for all cases, since MS with CF is not convergent. .................................................. 24

3.5 Eigenvalues of MS and MS-BILU iteration matrices. ............................................................................. 25

3.6 Eigenvalues of MS-CF and MS-CF-BILU iteration matrices. ............................................................... 26

3.7 High contrast in mobility field (red) will deteriorate solvability of the correction functions. .................. 27
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.8</td>
<td>Iteration histories for MSFV-CF-BILU with modified strategy</td>
<td>28</td>
</tr>
<tr>
<td>3.9</td>
<td>Natural logarithm of the permeability used for analysing CPU time of</td>
<td></td>
</tr>
<tr>
<td></td>
<td>the iterative procedure with and without CF</td>
<td>29</td>
</tr>
<tr>
<td>4.1</td>
<td>Permeability and pressure solution for the SPE 10 bottom layer</td>
<td>32</td>
</tr>
<tr>
<td>4.2</td>
<td>Iteration histories for AMS with different restriction schemes</td>
<td>32</td>
</tr>
<tr>
<td>4.3</td>
<td>Permeability and pressure solution for the full SPE 10 case</td>
<td>33</td>
</tr>
<tr>
<td>4.4</td>
<td>Iteration histories for AMS with FV/FE restriction scheme and linear-</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ear/reduced boundary condition</td>
<td>34</td>
</tr>
<tr>
<td>4.5</td>
<td>Natural logarithm of the permeability for BILU and ILU comparison</td>
<td>36</td>
</tr>
<tr>
<td>4.6</td>
<td>The CPU time average and error bar plots for BILU and ILU comparison</td>
<td>36</td>
</tr>
<tr>
<td>4.7</td>
<td>The iteration steps average and error bar plots for BILU and ILU</td>
<td>37</td>
</tr>
<tr>
<td>4.8</td>
<td>Natural logarithm of heterogeneous permeability filed</td>
<td>38</td>
</tr>
<tr>
<td>4.9</td>
<td>CPU time with problem size</td>
<td>40</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

Numerical simulation of multiphase flow in natural porous media involves the solution of large, heterogeneous, and anisotropic linear systems of equations. Multiscale methods have been developed to reduce the computational complexity required to solve such systems by combining information from fine and coarse scales, such that global operations, which are performed exclusively at coarse scales, are a small fraction of the total cost. The MultiScale Finite-Element (MSFE) method proposed by Hou and Wu [1] integrated the fine-scale information into a coarse-scale system by computing special basis functions, which are numerical solutions of localized boundary-value problems that capture the fine-scale effects. The major drawback of this method for reservoir simulation is that it cannot deliver a mass conservative velocity field. This is because having a conservative velocity field is crucial for accurate solutions of transport problems. In order to obtain approximate solutions that are strictly locally mass conservative at the fine scale, the Mixed MultiScale Finite-Element (MMSFE) [2–5] and the MultiScale Finite Volume (MSFV) [6] methods have been developed. Compared with the MMSFE formulation, the MSFV method yields mass conservative solutions with a smaller number of degrees of freedom.

In the MSFV method, similar to MSFE, local basis functions, which are solutions
of local problems, are employed to resolve the fine-scale coefficients. The basis functions are used to construct effective coarse-scale transmissibilities and to interpolate the coarse solution to the original fine scale. By construction, the MSFV approximate pressure solution guarantees mass conservation at the coarse scale. To obtain a locally conservative velocity field at the fine scale, additional local problems are solved subject to Neumann conditions obtained from the interpolated MSFV solution. The MSFV method has been extended to compressible multiphase flow \cite{7} and gravity effects \cite{8}. Furthermore, the efficiency of the MSFV method has been enhanced by adaptive computation of the basis functions for multiphase, time-dependent displacement problems \cite{9,11}. An algebraic description of the MSFV method was first introduced by developing an Operator-Based Multiscale Method (OBMM) by Zhou and Tchelepi \cite{12}. In OBMM, the construction of the MSFV coarse system and the interpolation of the coarse solution to find an approximate fine-scale solution are interpreted algebraically as restriction and prolongation steps, respectively. Applying these two operators to the original fine-scale system leads to the construction of the coarse-scale system, which can be solved in a variety of ways, since it is expected to be quite small. The MSFV approximate solution at the fine scale is then obtained by employing the prolongation operator. The advantage of OBMM is that it reduces the implementation complexity of the MSFV method, especially for reservoir models defined on unstructured grids.

The MSFV results have been shown to be in good agreement with the solutions obtained by running a classical expensive simulator at the fine scale for a wide range of heterogeneous test cases. Unfortunately, for extreme permeability contrasts and highly anisotropic problems, the MSFV method suffers from convergence and accuracy issues. Examples of these cases can be found for channelized permeability fields \cite{13} and problems with large grid aspect ratios (anisotropic permeability/transmissibility) \cite{14}. As for any multiscale method, the accuracy of the solutions
depends strongly on the quality of the local boundary conditions used. For very challenging problems (i.e., strong anisotropy with extreme permeability contrasts), no general local boundary condition has been devised. More importantly, no control on the level of error introduced by the localization assumption was possible in the original MSFV framework. To resolve these limitations, the iterative MSFV (i-MSFV) method was introduced [15], where the MSFV solution was improved with the help of locally computed ‘Correction Functions’ (CF) [8] and fine-scale smoothers. With the right components, the i-MSFV method converges to the fine-scale reference solution. With i-MSFV, if the iterative procedure is stopped before tight tolerances are achieved (e.g., on the residual and changes in pressure), one is guaranteed that the approximate velocity field is mass conservative. The i-MSFV method described in [15] overcomes the MSFV errors associated with the original single-pass MSFV method for many problems. However, for highly heterogeneous and anisotropic problems, the performance was not satisfactory. The deficiency was found to be related to the MSFV coarse scale operator. Hence, a symmetric MSFE based strategy was proposed [16] [17], and due to the importance of having a conservative solution at the end of the iterative process, the MSFV operator is employed as the last step [17]. The Two-stage Algebraic Multiscale Linear Solver (TAMS) [17] consists of local and global stages. In the global stage, low frequency errors are resolved by a multiscale preconditioner, while in the local stage high frequency errors are resolved using a local solver (smoother) such as the BILU [18]. By combining these two stages, all the errors are efficiently resolved. We note, however, that the specific role of the Correction Functions (CF) on the performance of TAMS was not studied.

In this report, a general iterative Algebraic Multiscale Solver (AMS) is described, which can use the MSFV or MSFE operators with or without correction functions. We show that the correction functions can be represented as an independent local preconditioning stage. Hence, i-MSFV or i-MSFE with correction functions, and
BILU can be represented as a three-stage process. We then study the effect of the CF stage on the convergence rate and the overall computational efficiency.

This report is organized as follows. In Chapter 2, we describe a general Algebraic Multiscale Solver (AMS) for the elliptic pressure equations. In Chapter 3, we analyse the effects of the CF on the MSFV framework. In Chapter 4, systematic tests are performed considering different local and global stages and also different local boundary conditions. Then, we compare the computational efficiency of AMS against SAMG [19], which is considered as the state-of-the-art linear solver. In the last chapter, we summarize our findings and suggest directions for future research.
Chapter 2

Algebraic Multiscale Solver

In this chapter, we develop an Algebraic Multiscale Solver (AMS) for the elliptic pressure equation. In the first section, the MSFV method is described. Then, AMS is described. Finally, the reconstruction of a fine-scale conservative velocity field is expressed in an algebraic manner.

2.1 Governing Equations

The pressure equation for single-phase incompressible flow can be written as

$$\nabla \cdot (\lambda \cdot \nabla p) = \nabla \cdot (\rho g \lambda \cdot \nabla z) + \tilde{q},$$

(2.1)

where $\lambda$ is the positive-definite mobility tensor, and $\tilde{q}$ represents source terms, such as injection and production wells. The MSFV method is proposed for the efficient solution of Eq. 2.1 with highly heterogeneous coefficients, $\lambda$, and complex RHS terms [6–8, 20].

The MSFV method superimposes two sets of coarse grids on the underlying fine grid, as illustrated in Figure 2.1. The first is a primal coarse grid, $\Omega^C_i (i \in \{1, \cdots, N\})$,
(bold solid black); the second is the dual coarse grid, \( \Omega^D_j \) \( (j \in \{1, \cdots, M\}) \), (dashed blue). Also shown in green and light orange are fine cells belonging to primal and dual coarse cells, respectively. These two superimposed grids can be much coarser than the underlying fine grid; therefore, the coarse-scale solution can be obtained cheaply. It is also important to note that the coarse grids (i.e., primal and dual) are not limited to the simple Cartesian structured grids shown in Figure 2.1.

The basis functions in the MSFV and MSFE methods are obtained by solving

\[
\begin{cases}
\nabla \cdot (\lambda \cdot \nabla \phi^i_j) = 0 & \text{in } \Omega^D_j \\
\nabla \perp \cdot (\lambda \cdot \nabla \phi^i_j) \perp = 0 & \text{on } \partial \Omega^D_j \\
\phi^i_j(x_k) = \delta_{ik} & x_k \in \text{vertex}
\end{cases}
\]

where \( \phi^i_j \) denotes the basis function associated with coarse node \( i \) in dual coarse block \( \Omega^D_j \), and the subscript \( \perp \) indicates the vector or operator is projected along the normal direction pointing out of the dual cell boundary \( \partial \Omega^D_j \). The CF is used to capture the fine-scale RHS terms, which are represented as a local particular solution.

Figure 2.1: Primal (bold solid black) and dual (dashed blue) coarse grid cells. Fine-cells belonging to a coarse control volume and a dual coarse cell are highlighted in green and light orange, respectively.
of the following local problems

\[
\begin{aligned}
\nabla \cdot (\lambda \cdot \nabla \phi_j^*) &= \nabla \cdot (\rho g \lambda \cdot \nabla z) + \tilde{q} \quad \text{in } \Omega_j^D \\
\nabla_{\perp} \cdot (\lambda \cdot \nabla \phi_j^*)_{\perp} &= \nabla_{\perp} \cdot (\rho g \lambda \cdot \nabla z)_{\perp} \quad \text{on } \partial \Omega_j^D \\
\phi_j^* (x_k) &= 0 \quad x_k \in \text{vertex}
\end{aligned}
\]  

(2.3)

where \( \phi_j^* \) represents the CF solution in dual coarse block \( \Omega_j^D \). Then, the approximate solution is determined as the superposition of both basis and correction functions

\[
p \approx p' = \sum_{j=1}^{M} \left[ \sum_{i=1}^{N} \phi_{ij}^* p^c_i + \phi_j^* \right],
\]

(2.4)

where \( p^c_i \) is the coarse-scale solution at node \( i \). The coarse-scale system is constructed by first substituting Eq. (2.4) into Eq. (2.1) and integrating over the coarse control-volumes \( \Omega_i^C \), which can be written as

\[
\sum_{j \in \mathcal{N}_i} T_{ij} p^c_j = \sum_{d=1}^{M} \int_{\partial \Omega_i^C \cap \Omega_d} (\lambda \cdot \nabla \phi_d^*) \cdot \vec{n} d\Gamma - \int_{\Omega_i^C} r dV, \quad i \in [1, n_c],
\]

(2.5)

where the divergence theorem was also used. Here, \( \mathcal{N}_i \) denotes the neighboring coarse cells of cell \( i \), \( n_c \) is the number of coarse cells, \( d\Gamma \) is the coarse grid boundary element, and \( \vec{n} \) is the unit-normal vector pointing outward. The \( r \) term represents other source terms on the fine scale. The coarse-scale transmissibility is defined as

\[
T_{ij} = -\sum_{d=1}^{M} \int_{\partial \Omega_i^C \cap \Omega_d} (\lambda \cdot \nabla \phi_d^j) \cdot \vec{n} d\Gamma.
\]

(2.6)

After obtaining the coarse-scale solution \( p^c \) by solving Eq. (2.5), we use Eq. (2.4) to get the approximate fine-scale solution.
2.2 Algebraic Description

Eq. 2.1 discretized on the fine grid can be written algebraically as

\[ Ap = q. \] (2.7)

An algebraic interpretation of the multiscale framework was proposed by Zhou and Tchelepi [12, 17] based on introducing prolongation and restriction operators. Using a wirebasket ordering [21, 22], the fine-scale system is first reordered based on the grid partitioning provided by the dual coarse cells. For 2D problems with structured grids, the dual coarse grid divides the fine-scale cells into three categories: interior (white), edge (blue), and vertex (red) cells, as illustrated in Figure 2.2. As the figure indicates, vertices are the coarse-grid nodes, and the edge cells are the boundaries of the dual cells. For structured 3D problems, an additional category is face cells. Finally, internal cells are those that lie inside dual coarse cells. Note that such a wirebasket partitioning can be used for unstructured grids [21]. In this report, we use the two-point flux approximation (TPFA) scheme to discretize the fine-scale problem. For simplicity, the framework is described for 2D problems, although the implementation is 3D.

A wirebasket reordered fine-scale system can be expressed as

\[
\begin{bmatrix}
A_{II} & A_{IE} & 0 \\
A_{EI} & A_{EE} & A_{EV} \\
0 & A_{VE} & A_{VV}
\end{bmatrix}
\begin{bmatrix}
p_I \\
p_E \\
p_V
\end{bmatrix} =
\begin{bmatrix}
q_I \\
q_E \\
q_V
\end{bmatrix},
\] (2.8)

where the local matrix $A_{ij}$ represents the contribution of cell $j$ to the discrete mass conservation equation of cell $i$. $A_{IE}$ represents the contribution of edge cells to the mass conservation of interior cells. We note that the TAMS scheme [17] does not
account for RHS terms, i.e., $q_I = 0$ and $q_E = 0$. Here, we account for the presence of RHS terms by using the Correction Function (CF) [7, 8, 11, 15, 23]. The RHS vector is split into two parts: (1) gravitational source terms and (2) other source terms, i.e.

$$
\begin{bmatrix}
q_I \\
q_E \\
q_V
\end{bmatrix}
= \begin{bmatrix}
q_{I}^G \\
q_{E}^G \\
q_{V}^G
\end{bmatrix}
+ \begin{bmatrix}
\bar{q}_I \\
\bar{q}_E \\
\bar{q}_V
\end{bmatrix},
$$
(2.9)

where

$$
\begin{bmatrix}
q_{I}^G \\
q_{E}^G \\
q_{V}^G
\end{bmatrix}
= B \begin{bmatrix}
q_I \\
q_E \\
q_V
\end{bmatrix},
$$
(2.10)

and $B$ is a diagonal ratio matrix, i.e. $B_{ii} = q_i^G/q_i$ and $B_{ij} = 0$ if $i \neq j$.

In order to satisfy the reduced boundary condition, i.e. Eq. 2.3, we further divide
$q^G_E$ into two parts, tangential $q^G_{E\parallel}$ and normal $q^G_{E\perp}$, i.e.

$$q^G_E = q^G_{E\parallel} + q^G_{E\perp}. \tag{2.11}$$

Based on Eq. 2.3, the tangential part of the gravitational source term for boundary cells $q^G_{E\parallel}$ should be taken into account. So, before using the RHS vector to construct the CF, we need to extract the tangential part of $q^G_E$, which can be expressed as

$$
\begin{bmatrix}
q_I \\
q'_E \\
q_V
\end{bmatrix}
= \begin{bmatrix}
q_I \\
q^G_{E\parallel} + \tilde{q} \\
q_V
\end{bmatrix}
= \mathcal{E}
\begin{bmatrix}
q^G_I \\
q^G_{E\parallel} \\
q^G_V
\end{bmatrix}
+ \begin{bmatrix}
\tilde{q}_I \\
\tilde{q}_E \\
\tilde{q}_V
\end{bmatrix}
= (\mathcal{E}B + I - B)
\begin{bmatrix}
q_I \\
q_E \\
q_V
\end{bmatrix}, \tag{2.12}
$$

where $\mathcal{E}$ is a diagonal matrix

$$
\mathcal{E} =
\begin{bmatrix}
e_1 & & \\
& \ddots & \\
& & e_i \\
& & \\
& & \ddots \\
& & e_n
\end{bmatrix}, \tag{2.13}
$$

with

$$
e_i = \begin{cases}
|\vec{n}_{e,i} \cdot \vec{n}_g| & \text{if } i \in \mathbb{N}_{\text{edge}} \\
1 & \text{otherwise}
\end{cases} \tag{2.14}
$$

entries. Here, $\mathbb{N}_{\text{edge}}$ is the set of edge cells, $\vec{n}_{e,i}$ is the unit-vector tangent to the edge cells at cell $i$, and $\vec{n}_g$ is the vector parallel to gravitational acceleration. To simplify the overall algebraic description, matrix $E$ is introduced as

$$E = \mathcal{E}B + I - B, \tag{2.15}$$
2.2. **ALGEBRAIC DESCRIPTION**

which leads to

\[
\begin{bmatrix}
q_I \\
q'_E \\
q_V
\end{bmatrix} =
\begin{bmatrix}
q_I \\
q_E \\
q_V
\end{bmatrix}.
\] (2.16)

The matrix entries for interior cells are preserved in the approximate multiscale operator. The stencil for edge cells, however, is modified to reflect the localization assumption. In fact, the only source of error in the multiscale approximation is due to the localization assumption, e.g., the reduced problem boundary condition. The reduced problem condition is captured by setting \( A_{EI} \) to zero with the corresponding part associated with this effect accounted for in the diagonal term, \( A_{EE} \). Finally, the stencil for vertex cells is replaced by the multiscale coarse system. The multiscale system considering the correction function \( \tilde{M}_{mswe} p' = q' \) is solved for \( p' \), which is an approximation to the reference fine-scale solution of \( Ap = q \). Therefore, the multiscale approximate system is expressed as

\[
\begin{bmatrix}
A_{II} & A_{IE} & 0 \\
0 & \tilde{A}_{EE} & A_{EV} \\
0 & 0 & A_C
\end{bmatrix}
\begin{bmatrix}
p'_I \\
p'_E \\
p'_V
\end{bmatrix}
= \begin{bmatrix}
q_I \\
q'_E \\
R_C
\end{bmatrix}.
\] (2.17)

It is clear that the multiscale system is upper-triangular; hence it is easy invert. Once the coarse system \( A_C p'_V = R_C \) is solved, the pressures for the edges and the interiors are obtained using backward substitution, i.e.

\[
p'_E = -\tilde{A}_{EE}^{-1}(A_{EV} p'_V - q'_E)
\]
\[
p'_I = -A_{II}^{-1}(A_{IE} p'_E - q_I) = A_{II}^{-1}(A_{IE} \tilde{A}_{EE}^{-1}(A_{EV} p'_V - q'_E) + q_I)
\]. (2.18)
In matrix form, these can be expressed as

\[
\begin{bmatrix}
    p'_I \\
    p'_E \\
    p'_V
\end{bmatrix}
= \begin{bmatrix}
    A_{II}^{-1} A_{IE} \tilde{A}_{EE}^{-1} A_{EV} & A_{II}^{-1} - A_{II}^{-1} A_{IE} \tilde{A}_{EE}^{-1} & 0 \\
    -\tilde{A}_{EE}^{-1} A_{EV} & 0 & \tilde{A}_{EE}^{-1} \\
    I_{VV} & 0 & 0
\end{bmatrix}
\begin{bmatrix}
    q'_I \\
    q'_E \\
    q'_V
\end{bmatrix}
+ \begin{bmatrix}
    q_I \\
    q'_E \\
    0
\end{bmatrix}.
\tag{2.19}
\]

Here, \( I_{VV} \) is an identity matrix with dimension \( n_c \times n_c \), and \( p'_V \) is the coarse-scale solution. The prolongation operator \([17]\) is

\[
\mathcal{P} = G \begin{bmatrix}
    A_{II}^{-1} A_{IE} \tilde{A}_{EE}^{-1} A_{EV} & \\
    -\tilde{A}_{EE}^{-1} A_{EV} & I_{VV}
\end{bmatrix},
\tag{2.20}
\]

where \( G \) is the permutation matrix that transforms the elements from wirebasket ordering into natural ordering, first along the x, then y, then z directions. The CF pressure in natural ordering \( p^{corr} \) is

\[
p^{corr} = G \begin{bmatrix}
    A_{II}^{-1} - A_{II}^{-1} A_{IE} \tilde{A}_{EE}^{-1} & 0 \\
    0 & \tilde{A}_{EE}^{-1} \\
    0 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
    q_I \\
    q'_E \\
    q'_V
\end{bmatrix},
\tag{2.21}
\]

which indicates that CF solves 1D problems for edge cells using the same reduced boundary condition as the basis function. The last column is zero because the vertices are disconnected from edge and interior cells. Since the permutation matrix is orthogonal, i.e. \( G^T = G^{-1} \), we can write

\[
\begin{bmatrix}
    q_I \\
    q_E \\
    q_V
\end{bmatrix} = G^T \begin{bmatrix}
    q_I \\
    q'_E \\
    q'_V
\end{bmatrix}.
\tag{2.22}
\]
Finally, using Eq. 2.16, Eq. 2.21 and Eq. 2.22, one can relate the CF pressure to the original RHS vector as follows

\[
p_{\text{corr}} = G \begin{bmatrix} A_{II}^{-1} & -A_{II}^{-1}A_{IE}\tilde{A}_{EE}^{-1} & 0 \\ 0 & \tilde{A}_{EE}^{-1} & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} E \\ G^T \end{bmatrix} q, \tag{2.23}
\]

which can be further simplified by defining the correction operator \( C \) in natural order, as follows

\[
p_{\text{corr}} = C q, \tag{2.24}
\]

where

\[
C = G \begin{bmatrix} A_{II}^{-1} & -A_{II}^{-1}A_{IE}\tilde{A}_{EE}^{-1} & 0 \\ 0 & \tilde{A}_{EE}^{-1} & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} E \\ G^T \end{bmatrix}. \tag{2.25}
\]

Here, \( E \) is an extraction diagonal matrix with a size of \( n_f \times n_f \) (\( n_f \) is the number of fine cells), which employs the gravity term modifications for the edge cells. In the iterative procedure, the residual is treated as a source term, which needs to be fully honored. For edge cells, the gravity term is solved with the reduced boundary condition, i.e., only \( q_{E\parallel}^G \) is considered. This treatment is only for the first iteration. However, in the following iterations, all the residual is treated as a source term, which needs to be fully considered even for edge cells. Thus, there is no need to divide the residual into normal and tangential parts. Therefore,

\[
E = \begin{cases} (\mathcal{E} - I)B + I & \text{for the first iteration} \\ I & \text{otherwise} \end{cases}, \tag{2.26}
\]

where \( I \) is the \( n_f \times n_f \) identity matrix.

The multiscale approximate solution expressed in Eq. 2.4 is stated algebraically
as
\[ p' = P p'_{V} + p^{\text{corr}}, \quad (2.27) \]

where \( P \) is the same prolongation operator used by TAMS [17]. Once the coarse-scale pressure, \( p'_{V} \), is obtained, Eq. 2.18 provides the edge and interior pressures. To compute \( p'_{V} \), the following coarse-scale system is constructed and solved

\[ A_{C} p'_{V} = R_{C}, \quad (2.28) \]

where
\[ A_{C} = R A P, \quad (2.29) \]

and
\[ R_{C} = R q - R A p^{\text{corr}}. \quad (2.30) \]

This coarse-scale system is consistent with Eq. 2.5. The restriction operator \( R \) can be obtained based on either a finite-volume, or a finite-element, scheme. For finite-volume type restriction (MSFV), the fine-scale equations in a coarse cell are simply summed up. Therefore, the MSFV restriction operator is expressed as

\[ R_{K,i} = \begin{cases} 
1 & \text{if } \Omega_{i}^{f} \subset \Omega_{K}^{C} \quad \forall K \in \{1, \ldots, n_{c}\}; \forall i \in \{1, \ldots, n_{f}\} \\
0 & \text{otherwise}
\end{cases}. \quad (2.31) \]

The finite-element based restriction operator (MSFE) is the transpose of the prolongation operator, i.e.
\[ R = P^{T}. \quad (2.32) \]

With the prolongation and restriction operators fully defined, one can solve the MSFV, or MSFE, coarse-scale system for \( p'_{V} \). Then, Eq. 2.27 is used to prolong
2.3. Fine-scale Velocity Reconstruction

Since the approximate multiscale solution, \( p' \), is obtained using basis functions and correction functions (Eq. 2.4) that are constructed by solving local elliptic problems on dual coarse blocks, the fine-scale velocity calculated directly from \( p' \) is not continuous across dual coarse block boundaries. As a result, this velocity field cannot be used to solve (nonlinear) transport problems (i.e., saturation equations). Since the boundaries of primal coarse blocks are in the interior of dual coarse blocks (Figure 2.1), the computed velocity field is continuous across those boundaries. Therefore, these continuous fluxes can be used to impose a Neumann boundary condition for local problems, whose solution is locally conservative in each primal coarse block [6].

The only prerequisite is that mass conservation is satisfied on the primal coarse scale, which can be achieved by the application of the proposed AMS with the finite-volume restriction operator as the last step [17]. As explained in [23], the reconstruction step is obtained algebraically by first reordering the fine-scale system based on the primal coarse cells partitions, i.e.

\[
\bar{A}p = \bar{q}.
\]
For structured 2D problems, $\bar{A}$ is a block penta-diagonal matrix, which can be split into block diagonal ($D$), upper ($U$) and lower ($L$) parts, i.e.

$$\bar{A} = L + D + U. \quad (2.36)$$

Each diagonal block of $D$ represents the transmissibilities between the fine cells within coarse cell $\Omega^C_i$. The corresponding off-diagonal blocks in $L$ and $U$ include the connections with fine cells in the neighborhood of coarse cells $\Omega^C_j$, $j \in \mathbb{N}_i$. Then, the local problem with the Neumann boundary condition can be written as

$$D' \bar{p}'' = \bar{q} - (\bar{A} - D')\bar{p}', \quad (2.37)$$

where $D' = D + E$, and $E$ is a diagonal matrix, defined as $E_{ii} = \sum_{j=1}^{n_f} (L_{ij} + U_{ij})$. Also, $(\bar{A} - D')\bar{p}'$ represents the flux across primal coarse cell boundaries, which is the Neumann boundary condition we want to impose. Note that the local elliptic problems with Neumann boundary condition are singular. To solve this problem, pressure is fixed at one fine cell in each coarse block $\Omega^C_i$. Once $\bar{p}''$ is obtained, it is transformed into natural ordering, $p''$. Then, a fine-scale conservative velocity field can be reconstructed as

$$u = \begin{cases} 
-\lambda \cdot \nabla p'' & \text{on } \Omega^C_i \\
-\lambda \cdot \nabla p' & \text{on } \partial \Omega^C_i 
\end{cases} \quad (2.38)$$

In order to demonstrate the conservative reconstruction, the top layer of the SPE 10 $[24]$ case is used. The fine problem has $220 \times 60$. We use a coarse grid of $22 \times 6$. Pressure is fixed in the upper left cell (1,1) and the lower right cell (220,60) with values of 1 and 0, respectively. The absolute value of the velocity divergence $\|\nabla \cdot u\|$ for each fine-scale cell is computed before and after the reconstruction step. Figure $2.4$. 
shows that the velocity divergence is non-zero on the boundaries of the dual coarse blocks before reconstruction, and that it is zero for every fine-scale cell-except for the source cell ((1,1) and (220,60))-after reconstruction.

Figure 2.3: Permeability field, fine-scale reference and MSFV pressure solution.
(a) Before reconstruction

(b) After reconstruction

Figure 2.4: Velocity divergence before and after reconstruction.
Chapter 3

Analysis of the Correction Function

The correction function (CF) has been proposed to honor the fine scale RHS terms [7, 8, 11, 15, 23]. In this chapter, we analyse the effects of CF on the iteration procedure from a general algebraic point of view.

3.1 Independent Local Stage

After some mathematical manipulation, Eq. 2.34 can be written as

\[ M^{-1}_{mswe} = \mathcal{P}(\mathcal{RAP})^{-1}\mathcal{R} + \mathcal{C} - \mathcal{P}(\mathcal{RAP})^{-1}\mathcal{RAC} \]

\[ = M^{-1}_{ms} + \mathcal{C} - M^{-1}_{ms}\mathcal{AC} \]  

(3.1)

In other words, the iterative procedure

\[ p^{\nu+1} = p^{\nu} + M^{-1}_{mswe}(q - Ap^{\nu}) \]

(3.2)
is equivalent to
\[ p^{\nu+1/2} = p^{\nu} + \mathcal{C}(q - A p^{\nu}) \] (3.3)
\[ p^{\nu+1} = p^{\nu+1/2} + M_{ms}^{-1}(q - A p^{\nu+1/2}). \] (3.4)

That is, this iterative procedure is equivalent to two steps: (1) first update the solution with the CF operator; (2) then update with the multiscale preconditioner \( M_{ms}^{-1} = \mathcal{P}(R A \mathcal{P})^{-1} \mathcal{R} \), which does not involve correction functions. Therefore, the operator \( \mathcal{C} \) can be represented as a totally independent stage, which does not affect the MS preconditioner at all. This has not been shown previously, and it helps us to quantify the impact of CF on the iterative multiscale solution strategy. The following two cases show that CF is quite similar to other standard (local) preconditioners and that CF honors the fine-scale RHS to improve the approximate multiscale solution.

### 3.1.1 Case 1: Homogeneous permeability

We consider a quarter five-spot problem, in which the permeability is isotropic and homogeneous, \( k_x = k_y = 1 \), with a fine grid of 40 \times 40 and a coarse grid of 4 \times 4. The BILU block size is the same as the size of the dual coarse cells, in order to provide the same support as the correction functions. The finite-volume restriction operator is used to construct the coarse-scale system. The pressure is fixed at (2,2) and (39,39) with values of 10 and 1, respectively. These two points are interior cells; therefore, the Dirichlet boundary condition serves as a source term on the fine-scale system. Figure 3.1 shows the solution after one iteration, with and without CF, along with BILU. The reference fine-scale solution is also shown. These results indicate that original multiscale approximate solution cannot represent the fine-scale solution accurately; however, the source terms are captured well by adding a local stage, such as either CF, or BILU. We note that the solutions obtained by MSFV-CF and MSFV-BILU are virtually the same. This indicates that the correction-function is just one
possible local stage, and we are not restricted in using it to capture the effects of fine-scale RHS terms. Moreover, we note that as long as MSFV is employed as the final (clean-up) step, local mass conservation on the primal coarse grid is achieved no matter which local preconditioner is used.

![Comparison between the solutions obtained from fine-scale reference, MSFV, MSFV with CF and MSFV with BILU. Note that all solutions are conservative at the coarse-scale.](image)

**Figure 3.1**

### 3.1.2 Case 2: Heterogeneous permeability

In this case, $100 \times 100$ fine and $10 \times 10$ coarse grids are employed. The log-normally distributed permeability field with spherical variogram and dimensionless correlation lengths of $\psi_1 = 0.5$ and $\psi_2 = 0.02$ is used. Also, the variance and mean of $\ln(k)$ are 2 and 3, respectively. As depicted in Figure 3.2, the angle between the long correlation...
length and the vertical domain boundaries is $45^\circ$. The pressure is fixed at (1,1) and (100,100) with values 10 and 1, respectively. In this case, we consider gravity with constant $\rho g = 1$. Gravity is in the y direction. Define $\varepsilon = \|r_k\|_2/\|r_0\|_2$ as the relative residual $l^2$ norm reduction, where $r_k$ is the residual at the $k^{th}$ iteration, and $r_0$ is the original RHS. In this report, we calculate $\varepsilon$ after each iteration to monitor the convergence history. Figure 3.3 shows that the CF improves the original MSFV solution significantly because it honors the gravitational source terms. Also, the MSFV-CF solution is better than the MSFV-BILU solution. The main difference between CF and BILU in this case is the local boundary condition. Since the MSFV-CF solution is better, it indicates that the reduced boundary condition is more appropriate to capture the gravity effects if one makes a single pass. However, the benefit of CF is not as significant for the iterative procedure. The choice of the local stage preconditioner is a trade-off between accuracy and computational effort. Moreover, Figure 3.4 indicates that CF improves the convergence in terms of iteration steps. In other words, since CF serves as an additional local stage, less iteration steps are required for both FV- and FE-based multiscale solvers. However, the most important criterion is the overall computational efficiency measured in CPU time. Finally, we note that, the FE-based multiscale approach outperforms FV-based method for this case. More detailed comparisons about these two preconditioning schemes are discussed in the next chapter.

### 3.2 Spectral Analysis

**Lemma 1.** The iteration matrix $I_{ms} \equiv I - M_{ms}^{-1}A$ of multiscale system only has eigenvalue of 1 and 0

**Proof.** $(M_{ms}^{-1}A)^2 = P(RAP)^{-1}RAP(RAP)^{-1}RA = M_{ms}^{-1}A$, then any eigenvalue $\lambda$ of the matrix $M_{ms}^{-1}A$ must satisfy $\lambda^2 = \lambda$. Therefore, $\lambda = 0, 1$ and the iteration matrix
3.2. SPECTRAL ANALYSIS

Figure 3.2: Natural logarithm of layered permeability filed with $\psi_1 = 0.5$ and $\psi_2 = 0.02$. Fine grid size is $100 \times 100$ and coarse grid size is $10 \times 10$.

Figure 3.3: Comparison between the solutions obtained from fine-scale reference, MSFV, MSFV-CF and MSFV-BILU.
Figure 3.4: Iteration histories for MS with and without CF. Note that BILU is used for all cases, since MS with CF is not convergent.

\[ I_{ms} \] only has eigenvalues of 1 and 0.

Lemma 1 indicates that the multiscale stage on its own cannot converge because only low frequency errors are resolved [17]. A local stage is required to remove the high frequency errors. However, if we just employ CF as the local preconditioner (i.e. MS-CF), it still cannot converge. We now show an example of the full eigenvalue structures of MS system alone and MS system with local stages such as the CF and the BILU for a simple 2D problem. The setup is as in Case 1, except that the pressure is fixed at (1,1) and (40,40). Figure 3.5 shows that multiscale strategy does not converge alone, however, when it is employed in combination of a local preconditioner such as BILU, all the eigenvalues are reduced into the unit circle. However, Figure 3.6 shows if the CF used instead of the BILU, i.e. MS-CF, we find that even though some eigenvalues are moved deeper into the unit circle, some eigenvalues are in fact larger than unity. This is because the CF shares the same reduce boundary condition assumption with the MS system. So that the local errors on the dual-coarse block boundary cannot be removed by the CF. Therefore we still need some other local preconditioning stage to reduce the local errors and achieve convergence. If we use
BILU as an additional local stage, i.e. MS-CF-BILU, then all the eigenvalues are less than unity. Also, we note that the maximum eigenvalue is smaller than that for the MS-BILU case, which explains why the three stage MS-CF-BILU preconditioner converges faster. To summarize, if we just use the CF, we still need some other local preconditioner to converge.

![Eigenvalues of MS and MS-BILU iteration matrices.](image)

Figure 3.5: Eigenvalues of MS and MS-BILU iteration matrices.

### 3.3 Sensitivity to Large Permeability Contrasts

Another drawback of the CF is that it is very sensitive to large contrasts in the transmissibility field. This is because CF is the solution of a 1D problem on the edge with source terms. For example, if there exist non-zero source terms between two impermeable regions crossing the boundary, as illustrated in Figure 3.7, the numerical solution may be difficult to compute, or is completely wrong. This can slow things down, or even lead to divergence. Here, we propose to modify the residual instead of clipping the transmissibility field and only for the CF calculation. The residual is
modified by a local scaling factor before it is used to update the CF, i.e.

\[
\begin{bmatrix}
q_I^m \\
q_E^m \\
q_V^m
\end{bmatrix} = E' \begin{bmatrix}
q_I \\
q_E \\
q_V
\end{bmatrix},
\]  

(3.5)

where \( E' \) is a diagonal matrix,

\[
E' = \begin{bmatrix}
\beta_1 & & \\
& \ddots & \\
& & \beta_i \\
& & & \ddots \\
& & & & \beta_n
\end{bmatrix}
\]  

(3.6)
3.3. SENSITIVITY TO LARGE PERMEABILITY CONTRASTS

\[ \beta_i = \begin{cases} 
\frac{T_{e,\min}}{T_{e,\max}} & \text{if } i \in \mathbb{N}_{\text{edge}} \\
1 & \text{otherwise}
\end{cases} \] \hspace{1cm} (3.7)

where \( T_{e,\min} \) and \( T_{e,\max} \) are the minimum and maximum values of the effective transmissibility at the interfaces along edge \( e \). This approach is purely local, and \( E' \) can be calculated automatically based on the fine-scale transmissibility field and grid information. It is important to note that in this approach, there is no need to manually set up a threshold to clip the transmissibility values for CF calculations.

Figure 3.7: High contrast in mobility field (red) will deteriorate solvability of the correction functions.

The following example demonstrates the effectiveness of this modification strategy. The SPE 10 top layer is considered with \( 220 \times 60 \) and \( 22 \times 6 \) fine and coarse cells, respectively, and BILU is used with \( 22 \times 6 \) blocks. The pressure is fixed at \((1,1)\) and \((220,60)\) with values of 0 and 1, respectively. Figure 3.8 shows that MSFV-CF-BILU improves the performance of MSFV-BILU to some extent. With the modified CF as just described, MSFV-CF-BILU accelerates the iterative process further. For 3D
cases, this modification needs to be performed for the cells located on faces, which makes it more complicated.

![Figure 3.8: Iteration histories for MSFV-CF-BILU with modified strategy.](image)

### 3.4 Computational Cost

Although the CF can help the convergence rate, this benefit should more than offset the additional computational cost for CF to be useful in an iterative process. The criterion to evaluate a preconditioner is not the number of iterations, but the overall computational cost. In order to examine the computational cost of the CF stage, we consider a 3D problem with a normally distributed log-permeability field generated by sequential Gaussian simulation [25], shown in Figure 3.9. The mean and variance are 4 and the correlation length is one eighth (of the total length) for each direction. The fine-scale grid is $128 \times 128 \times 64$, and the coarse grid is $16 \times 16 \times 8$. For BILU, we chose a block size of $4 \times 4 \times 4$. The pressure is fixed at the left and right faces with values of 1 and 0, respectively. The stopping criterion is a reduction in the relative $l^2$
norm of the residual by five orders of magnitude (i.e., $\|r_k\|_2/\|r_0\|_2 \leq 10^{-5}$). Table 3.1 indicates that although the number of iterations is reduced by 23\% when the CF is used, the computational cost of the solution phase increased by 37\%.

![Image of permeability](image)

Figure 3.9: Natural logarithm of the permeability used for analysing CPU time of the iterative procedure with and without CF.

<table>
<thead>
<tr>
<th></th>
<th>MSFE-BILU</th>
<th>MSFE-CF-BILU</th>
</tr>
</thead>
<tbody>
<tr>
<td>Iteration steps</td>
<td>43</td>
<td>33</td>
</tr>
<tr>
<td>Solve time (sec)</td>
<td>19.63</td>
<td>26.93</td>
</tr>
</tbody>
</table>

Table 3.1: CPU time (sec) and iteration steps.

### 3.5 Summary

In this chapter, we showed that the CF can be represented as a purely local stage, and that it does help convergence. However, the CF cannot be used alone and additional efforts are required to make the iterative procedure convergent, such as other local stages, or GMRES. Also, the CF is sensitive to high contrasts in the transmissibility,
which complicates the solution process. More importantly, the additional computational cost associated with the CF can become expensive, since the update may be required for each iteration. One may argue that the CF can capture the fine-scale source term, but since there are many possible local preconditioners, we can choose other operators as the local method to honor the fine-scale source term in a more efficient and robust solver framework than the CF.
Chapter 4

Results

In this chapter, we perform systematic tests to find the ‘best’ overall strategy for the algebraic multiscale solvers (AMS). In our framework, we have various options, for different strategies, such as finite-volume or finite-element restriction, linear or reduced local boundary conditions, and different local preconditioners. Based on the numerical experiments, we propose the ‘best’ overall combination of steps in the iterative procedure.

4.1 AMS Components

4.1.1 Restriction Scheme

AMS can employ two schemes to restrict the fine-scale information to the coarse-scale system. One approach is based on a finite-volume (FV) type restriction operator; the other one is based on a finite-element (FE) type. In this section, we test the performance of the two different restriction schemes. We consider the SPE 10 bottom layer permeability field with channelized structures (Figure [4.1]), which is a challenging problem for MSFV. The fine grid is $220 \times 60$ and the coarse grid is $22 \times 6$, and each
BILU block contains $10 \times 10$ fine cells. Also, both FV and FE restriction operators are used, with and without correction functions. The pressure is fixed at the left and right sides with values of 1 and 0, respectively. For this case, if we use FV, the residual does not decrease for many iterations; however, the residual decreases very quickly if we use FE (Figure 4.2). This is because the finite-volume coarse-scale operator has weak preconditioning properties. This case also demonstrates that the CF cannot fix the MSFV deficiency for highly heterogeneous problems.

Figure 4.1: Permeability and pressure solution for the SPE 10 bottom layer.

Figure 4.2: Iteration histories for AMS with different restriction schemes.
4.1.2 Local Boundary Condition

In this section, we study the effects of different local boundary conditions. Here we consider the full SPE 10 3D case (Figure 4.3), which has a $60 \times 220 \times 85$ fine cells. Here, we use a $6 \times 22 \times 17$ coarse grid. Each BILU block size is set to $4 \times 4 \times 5$. The pressure is fixed at the left and right faces with values of 1 and 0, respectively. GMRES is also employed in this case. We compare the reduced boundary condition for FV, FE, the linear boundary condition for FV and FE. Figure 4.4 shows that the reduced boundary condition with FE is still the best choice. For FV, the linear boundary condition is better than the reduced boundary condition for this particular problem. For the linear boundary condition, FE is comparable to FV. Moreover, the reduced boundary condition is significantly better than the linear boundary condition for FE.

![Permeability and pressure solution for the full SPE 10 case.](image)

Figure 4.3: Permeability and pressure solution for the full SPE 10 case.

4.1.3 Local Preconditioner

Overall, we have found that FE with the reduced boundary condition is the best overall choice. Next, we need to choose an efficient local preconditioner. Block ILU(0) is used as the second preconditioner in TAMS [26]. Here, we choose the
Figure 4.4: Iteration histories for AMS with FV/FE restriction scheme and linear/reduced boundary condition.
point-wise Incomplete LU factorization (ILU) as the local preconditioner. Based on our experiments, the CPU time is comparable to BILU; however, ILU can save almost the entire set up time of BILU. We perform a comparison for a normally distributed log-permeability field, as shown in Figure 4.5 with both mean and variance of 4 and correlation lengths of one eighth for each direction. The fine-scale grid is 128×128×64, and the coarse grid is 16×16×8. For BILU, we choose a block size of 4×4×4. The pressure is fixed at the left and right faces with values of 1 and 0, respectively. We consider both isotropic \( \Delta x = \Delta y = \Delta z \) and anisotropic \( \Delta x = 5\Delta y = 50\Delta z \) cases with 20 different realizations, which are generated using sequential Gaussian simulation [25]. Figure 4.6 and Figure 4.7 show the average values and the ranges for iteration steps and CPU time. For all cases, even though ILU requires more iterations, the overall CPU time is lower than the other strategy. Therefore, ILU is more efficient than BILU. For the isotropic problems, the difference between FE and FV is not very significant, but for anisotropic case, FE is more efficient than FV.

4.2 AMS Performance

Based on the results of the last section, we choose FE restriction in combination with reduced local boundary condition and ILU as the local preconditioner for our AMS strategy. In this section, we examine the performance of AMS by comparing it with the state-of-the-art linear solver SAMG [19], and we also investigate the scalability potential of AMS.

4.2.1 Comparison with SAMG

We first generate a 32×32×32 grid with the normally distributed log-permeability generated using sequential Gaussian simulation [25], with a mean of 3 and a variance of 4. The correlation length is one eighth of the domain length in each dimension.
CHAPTER 4. RESULTS

Figure 4.5: Natural logarithm of the permeability for BILU and ILU comparison.

Figure 4.6: The CPU time average and error bar plots for BILU and ILU comparison.
Figure 4.7: The iteration steps average and error bar plots for BILU and ILU comparison.

The permeability field is shown in Figure 4.8. Then, we perform piecewise constant refinement of the grid by a factor of two in each direction. Therefore, four grids are generated with $32^3$, $64^3$, $128^3$ and $256^3$ cells, respectively. For these four different problems, the number of fine cells varies from ten thousand to ten million. We examine the isotropic problem here and fix the coarsening ratio at $8 \times 8 \times 8$. In this case, Dirichlet boundary conditions are employed on the left and right faces with values of 1 and 0, respectively; no-flow boundary conditions are applied on all other boundaries. The stopping criterion is a reduction in the relative $l^2$ norm of the residual by five orders of magnitude. The finite-element type restriction scheme and ILU are used for AMS. The SAMG library is from Fraunhofer Institute SCAI, release version 25a1, December 2010 [19]. It employs a single stand-alone V-cycle, a convergence tolerance of 0.1 in relative residual reduction for the SAMG iteration and one Gauss-Seidel C-relaxation sweep as pre- and post- smoothing steps on each level. Also, the coarsest level solution is by a direct solver (sparse Gauss elimination). The CPU time and iteration steps for SAMG and AMS are shown in Tables 4.1 and
Based on this comparison, AMS is quite comparable to SAMG, in terms of CPU time for various problem sizes, which makes AMS a promising solver. Table 4.2 indicates the convergence rate is insensitive to the problem size. In fact, AMS can be even faster by enhancing the coarsening strategy; moreover, the benefits from AMS will be significant when it is used in an adaptive sequential strategy [27] or with parallel computing.

![Natural logarithm of heterogeneous permeability field.](image)

**Figure 4.8:** Natural logarithm of heterogeneous permeability field.

<table>
<thead>
<tr>
<th>Problem size</th>
<th>32 × 32 × 32</th>
<th>64 × 64 × 64</th>
<th>128 × 128 × 128</th>
<th>256 × 256 × 256</th>
</tr>
</thead>
<tbody>
<tr>
<td>Setup phase</td>
<td>0.12</td>
<td>1.14</td>
<td>12.33</td>
<td>157.41</td>
</tr>
<tr>
<td>Solution phase</td>
<td>0.14</td>
<td>0.76</td>
<td>6.33</td>
<td>73.02</td>
</tr>
<tr>
<td>Total</td>
<td>0.26</td>
<td>1.90</td>
<td>18.66</td>
<td>230.43</td>
</tr>
<tr>
<td>GMRES iterations</td>
<td>6</td>
<td>7</td>
<td>9</td>
<td>10</td>
</tr>
</tbody>
</table>

**Table 4.1:** CPU time (sec) and iteration steps for SAMG.
4.2. AMS PERFORMANCE

<table>
<thead>
<tr>
<th>Problem size</th>
<th>32 × 32 × 32</th>
<th>64 × 64 × 64</th>
<th>128 × 128 × 128</th>
<th>256 × 256 × 256</th>
</tr>
</thead>
<tbody>
<tr>
<td>Setup phase</td>
<td>0.29</td>
<td>2.12</td>
<td>17.00</td>
<td>147.96</td>
</tr>
<tr>
<td>Solution phase</td>
<td>0.10</td>
<td>0.87</td>
<td>8.01</td>
<td>55.91</td>
</tr>
<tr>
<td>Total</td>
<td>0.39</td>
<td>2.99</td>
<td>25.01</td>
<td>203.87</td>
</tr>
<tr>
<td>GMRES iterations</td>
<td>22</td>
<td>21</td>
<td>21</td>
<td>18</td>
</tr>
</tbody>
</table>

Table 4.2: CPU time (sec) and iteration steps for AMS.

4.2.2 Scalability Analysis

In order to study the scalability of AMS, we normalize all the four problem with the 32³ case, and we record the CPU time for the computational parts, including the local linear systems construction, multiscale operators (prolongation, restriction and coarse scale system) construction, total setup phase (including the first two parts and ILU setup) and solution phase. Figure 4.9 shows that the CPU time increases linearly with problem size for each computational component. All the results are close to the unit-slope linear reference, which indicates that AMS is scalable.
Figure 4.9: CPU time with problem size.
Chapter 5

Summary and Future Work

Summary

In this report, a general Algebraic Multiscale Solver (AMS) for the pressure equation was developed. We incorporate the Correction functions (CF) into our framework and show that it can be represented as an independent local stage, which can be entirely separated from the multiscale preconditioning stage. As a local preconditioner, the CF helps to capture the fine scale RHS and accelerates the overall convergence rate, but the gain in convergence rate does not offset the additional cost. Also, CF must be combined with other local solvers, or smoothers, to guarantee convergence. Therefore, we conclude that CF is not necessary in the AMS framework, and we propose to use a more efficient local preconditioner such as ILU. Then, in a manner similar to [23], we described how AMS can allow reconstruction of a conservative velocity field after any iteration level, which is crucial for the hyperbolic transport problem. In addition, we systematically tested the performance of AMS and found that the finite-element restriction scheme is more robust and efficient than the finite-volume method, because the finite-volume framework fails to provide good solutions in the presence of highly heterogeneous and anisotropic (with high aspect ratios) fields. For
the finite-element restriction, the reduced local boundary condition outperforms the linear boundary condition. Therefore, the best overall strategy for AMS is to use a finite-element restriction scheme, an ILU local preconditioner, and the reduced local boundary condition. The performance of AMS is quite comparable to the state-of-the-art algebraic multigrid (SAMG) preconditioner, and AMS shows great scalability properties.

**Ongoing and Future Work**

We are incorporating a well model with multiple completions into AMS. The prolongation and restriction operators are extended by employing well basis function [20, 28]. Another research direction is to adapt AMS to generally unstructured grids, since AMS only requires the fine-scale transmissibility and a wirebasket reordering, it is well suited for unstructured grids. Finally, extending multiscale methods to the fully implicit scheme is a challenging direction, which will broaden the application of multiscale methods to general-purpose reservoir simulation.
Bibliography


