

Copyright  
by  
Xiuli Gai  
2004

The Dissertation Committee for Xiuli Gai  
certifies that this is the approved version of the following dissertation:

**A Coupled Geomechanics and Reservoir Flow Model on  
Parallel Computers**

Committee:

---

Mary F. Wheeler, Supervisor

---

Rick H. Dean

---

Kamy Sepehrnoori

---

Mark E. Mear

---

A. Daniel Hill

**A Coupled Geomechanics and Reservoir Flow Model on  
Parallel Computers**

by

**Xiuli Gai, B.E., M.E.**

**DISSERTATION**

Presented to the Faculty of the Graduate School of  
The University of Texas at Austin  
in Partial Fulfillment  
of the Requirements  
for the Degree of

**DOCTOR OF PHILOSOPHY**

THE UNIVERSITY OF TEXAS AT AUSTIN

August 2004

To my mother, Guiqin,  
my husband, Yuping, and my son, Keith.

## Acknowledgments

First and foremost, I would like to thank my supervisor, Dr. Mary F. Wheeler, for her exceptional guidance and inspiration in this work. She led me into the fascinating world of numerics and taught me how to use mathematical tools to solve engineering problems. I am grateful to her not only for her encouragement and support for my academic research, but also for her forgiveness of my mistakes and her care for me personally.

I am also grateful to Dr. John Wheeler. I've never taken a class from him, but every day he teaches me through his code and through his integrity as a computer scientist, as a chemical and petroleum engineer and most of all as an individual. Words fail me to express my appreciation and respect for him.

I am indebted to Dr. Rick H. Dean for sharing his extensive knowledge and experiences in reservoir simulation, geomechanics and solid mechanics. I have benefited enormously from discussion and collaboration with him. My sincere gratitude also goes to the other members of my committee, Dr. Kamy Sepehrnoori, Dr. Mark E. Mear and Dr. A. Daniel Hill, for reading previous drafts of this dissertation and providing valuable suggestions and comments.

I am very thankful to my colleagues at CSM (the Center for Subsurface Modeling) and the department of PGE (Petroleum and Geosystems Engineer-

ing). In particular, I would like to thank Dr. Hector Klie, Dr. Wolfgang Bangerth and Connie Baxtor who have always been warmhearted and helpful during the most challenging time of my life, and I thank Andrew Pacetti for his patience and time in editing my thesis. Special thanks also go to Dr. Clint Dawson, Dr. Malgorzata Peszyńska, Dr. Yuri Vassilevski, Dr. Qin Lu, Dr. Shuyu Sun, Ruijie Liu, Dr. Vadym Aizinger and Owen Eslinger.

Last, but not least, I would like to thank my husband Yuping and my son Keith who made all of this possible with their unconditional love and sacrifice. Their arms are always open to me whether I succeed or fail. Words are not enough here to express the gratitude of a daughter to the greatest mother in the world, Guiqing Wang.

This work is supported by several funding agencies. Special acknowledgments are due to the Sandia National Laboratories, Department of Energy and the Industrial Affiliates of CSM. Dr. Mike Stone is highly acknowledged for initiating the collaboration with the Sandia National Laboratories.

# **A Coupled Geomechanics and Reservoir Flow Model on Parallel Computers**

Publication No. \_\_\_\_\_

Xiuli Gai, Ph.D.

The University of Texas at Austin, 2004

Supervisor: Mary F. Wheeler

Land subsidence due to the exploitation of groundwater and hydrocarbon fluids has triggered extensive studies in coupled fluid flow and geomechanics simulations. However, numerical modeling of coupled processes imposes great computational challenges. Coupled analysis for large scale full-field applications with millions of unknowns has been, historically, considered extremely complex and unfeasible. The purpose of this dissertation is to investigate accurate and efficient numerical techniques for coupled multiphase flow and geomechanics simulations on parallel computers.

We emphasize the iterative coupling approach in extending conventional fluid-flow modeling to coupled fluid-flow and geomechanics modeling. To overcome the slow convergence—a major drawback of this method—we propose new preconditioning schemes to achieve a faster convergence rate. Efficient and parallel scalable linear solvers are developed to reduce the computational overhead induced by the solution of discrete elasticity equations.

Special communications techniques are implemented to optimize parallel efficiency.

In this dissertation we first derive the mathematical model for multiphase flow in a deformable porous medium. We then present a new formulation of the iterative coupling scheme and prove the optimality of two physics-based preconditioners that are traditionally used in the petroleum industry. Practical strategies and new preconditioners are proposed to improve the numerical performance of the iteratively coupled approach. In addition, we develop two types of preconditioners for solving the linear elasticity system, namely, multi-level domain decomposition preconditioners using a super-coarsening multigrid algorithm and displacement decomposition preconditioners. Parallel implementation issues are also addressed. Numerical examples are presented to demonstrate the robustness, efficiency and parallel scalability of the proposed linear solution techniques.

# Table of Contents

<b>Acknowledgments</b>	<b>v</b>
<b>Abstract</b>	<b>vii</b>
<b>List of Tables</b>	<b>xiii</b>
<b>List of Figures</b>	<b>xvi</b>
<b>Chapter 1. Introduction</b>	<b>1</b>
1.1 Motivation . . . . .	1
1.1.1 Problem Statement . . . . .	1
1.1.2 Challenges in Coupled Simulations . . . . .	5
1.2 Objectives of This Work . . . . .	9
1.3 Literature Review . . . . .	10
1.3.1 Theory and Governing Equations . . . . .	10
1.3.2 Coupled Geomechanics and Reservoir Flow Modeling . .	12
1.3.3 Linear Solvers for the Coupled System . . . . .	14
1.4 Outline of the Thesis . . . . .	20
<b>Chapter 2. Mathematical Model</b>	<b>22</b>
2.1 Linear Porelasticity Theory . . . . .	22
2.2 Fluid-Flow Theory in Deformable Porous Media . . . . .	25
2.2.1 Coupled Black-oil Model Mass Conservation Equations .	26
2.2.2 Coupling Parameters . . . . .	35
2.3 Summary of Coupled Geomechanics and the Reservoir Flow Model	36
2.3.1 Black-oil Model . . . . .	37
2.3.2 Poroelasticity Model . . . . .	39
<b>Chapter 3. Finite Element Formulation</b>	<b>41</b>

<b>Chapter 4. Iteratively Coupled Technique</b>	<b>49</b>
4.1 Overview of Different Coupling Techniques . . . . .	50
4.2 Iteratively Coupled Technique . . . . .	54
4.2.1 Linearization of Nonlinear Systems . . . . .	54
4.2.2 Iterative Coupling by Operator Splitting . . . . .	55
4.2.3 Physics-based Preconditioners to $\mathbf{Q}\delta\mathbf{u}$ . . . . .	56
4.3 Iterative Coupling as One Iteration of a Preconditioned Richardson Method . . . . .	60
4.3.1 Preconditioners for $\mathbf{S}$ . . . . .	64
4.3.1.1 Optimal Preconditioners for $\mathbf{S}$ . . . . .	65
4.3.1.2 Preconditioners Based on Other Approximations of $\mathbf{Q}\mathbf{E}_u^{-1}\mathbf{Q}^T$ . . . . .	69
4.3.2 $\tilde{\mathbf{S}}_p$ as a Preconditioner for Krylov Subspace Methods . . . . .	71
4.4 Numerical Results . . . . .	72
4.4.1 Examples . . . . .	73
4.4.1.1 Example 1: Depletion in a Confined Reservoir . . . . .	73
4.4.1.2 Example 2: Depletion in an Unconfined Reservoir . . . . .	74
4.4.1.3 Example 3: A Confined Reservoir with Surrounding Non-pay Rocks . . . . .	74
4.4.1.4 Example 4: an Unconfined Reservoir with Stiff Non-pay Rocks . . . . .	76
4.4.1.5 Example 5: an Unconfined Reservoir with Soft Non-pay Rocks . . . . .	78
4.4.1.6 Example 6: Three-phase Flow in a Confined Reservoir . . . . .	78
4.4.2 Convergence with Respect to Fluid and Rock Properties . . . . .	79
4.4.3 Convergence with Respect to Coefficient Jump . . . . .	81
4.4.4 Convergence with Respect to Grid Refinement . . . . .	83
4.4.5 Convergence with Respect to Reservoir Heterogeneity . . . . .	84
4.4.6 Scaling Factor $\beta$ in $\tilde{\mathbf{S}}_{p5}$ . . . . .	89
4.4.7 A Fully Coupled Approach with Preconditioned Richardson Iterations . . . . .	92

<b>Chapter 5. Numerical Verification</b>	<b>95</b>
5.1 Mandel's Problem . . . . .	95
5.2 A 3D Confined Reservoir with Non-pay Rocks . . . . .	105
<b>Chapter 6. Linear Solvers for the Poroelastic Model</b>	<b>113</b>
6.1 Krylov Linear Iterative Solvers: GMRES and BiCG-STAB . .	114
6.2 Domain Decomposition Preconditioners . . . . .	115
6.2.1 One-level Overlapping DD Preconditioners . . . . .	116
6.2.2 Multilevel Domain Decomposition Preconditioners . . .	120
6.2.2.1 A Multigrid Introduction . . . . .	121
6.2.2.2 Super-coarsening Multigrid Preconditioner . . .	123
6.2.2.3 Multilevel Preconditioners . . . . .	128
6.3 Displacement Decomposition Preconditioner (DiD) . . . . .	130
6.3.1 Displacement Decomposition . . . . .	130
6.3.2 Preconditioners for DiD Subsystems . . . . .	135
6.3.2.1 Red-black Line Gauss-Seidel . . . . .	135
6.3.2.2 Algebraic Multigrid (AMG) . . . . .	136
6.3.2.3 Algebraic Multigrid for an Aggregated System .	141
6.4 Numerical Experiments . . . . .	143
6.4.1 Numerical examples . . . . .	144
6.4.1.1 Example 1 . . . . .	144
6.4.1.2 Example 2 . . . . .	144
6.4.1.3 Example 3 . . . . .	145
6.4.1.4 Example 4 . . . . .	145
6.4.2 Convergence with Respect to Grid Refinement . . . . .	146
6.4.3 Convergence with Respect to Poisson's Ratio . . . . .	152
6.4.4 Convergence with Respect to Coefficient Jump . . . . .	156
6.4.5 Comparison over Multiple Time Steps . . . . .	162

<b>Chapter 7. Parallel Implementation of the Poroelastic Model</b>	<b>165</b>
7.1 Grid Partitioning . . . . .	166
7.2 General Inter-processor Communications . . . . .	168
7.3 Parallel Super-coarsening Multigrid (SCMG) . . . . .	169
7.4 Communication Reduction Techniques . . . . .	172
7.5 Numerical Example 1 . . . . .	175
7.6 Numerical Example 2 . . . . .	180
<b>Chapter 8. Concluding Remarks and Future Work</b>	<b>192</b>
8.1 Conclusions . . . . .	192
8.2 Future Work . . . . .	197
<b>Bibliography</b>	<b>198</b>
<b>Vita</b>	<b>221</b>

## List of Tables

4.1	Preconditioners for the Schur complement of the pressure equation. . . . .	70
4.2	Example 3 with varying ratios of jump in Young's modulus: Total number of Newton iterations for the first two time steps. The symbol * indicates failure to converge. . . . .	82
4.3	Example 3 with $E_1 = 10^4$ psi and $E_2 = 10^6$ psi: Total number of Newton iterations for the entire simulation with a total of 38 time steps. . . . .	83
4.4	Example 6 with grid refinements: Total number of Newton iterations for the first two time steps. The symbol * indicates failure to converge. . . . .	84
4.5	Examples 4 & 5: Total number of Newton iterations and CPU time for the entire simulation with 38 time steps. Example 4: $E_1 = 10^4$ psi and $E_2 = 10^6$ psi; Example 5: $E_1 = 10^6$ psi and $E_2 = 10^4$ psi. . . . .	86
4.6	Example 1: Newton iteration counts using $\tilde{\mathbf{S}}_{p5}$ with varying scaling factors ( $\beta^o$ : optimal factor, $\beta^a$ : approximate factor). .	89
4.7	Example 2: Newton iteration counts using $\tilde{\mathbf{S}}_{p5}$ with varying scaling factors ( $\beta^o$ : optimal factor, $\beta^a$ : approximate factor). The symbol * indicates failure to converge. . . . .	89
4.8	Example 3: Newton iteration counts using $\tilde{\mathbf{S}}_{p5}$ with varying scaling factors ( $\beta^o$ : optimal factor, $\beta^a$ : approximate factor). .	90
4.9	Example 4: Newton iteration counts using $\tilde{\mathbf{S}}_{p5}$ with varying scaling factors ( $\beta^o$ : optimal factor, $\beta^a$ : approximate factor). .	90
4.10	Example 5: Newton iteration counts using $\tilde{\mathbf{S}}_{p5}$ with varying scaling factors ( $\beta^o$ : optimal factor, $\beta^a$ : approximate factor). The symbol * indicates failure to converge. . . . .	90
4.11	Rock compressibility predicted by different approximations to $\mathbf{Q}\mathbf{E}_u^{-1}\mathbf{Q}^T$ . . . . .	91
4.12	Comparison of Newton iteration count and CPU time in seconds for the iteratively coupled and the fully coupled approach. The discrete elasticity system is solved by a direct solver. . . . .	93

4.13	Comparison of Newton iteration count and CPU time in seconds for the iteratively coupled and the fully coupled approach. The discrete elasticity system is solved by ICCG. . . . .	93
5.1	Parameters in Mandel's problem. . . . .	97
5.2	Mandel's problem output time steps. . . . .	101
5.3	Results of the 3D problem at the end of 4000 days. . . . .	106
5.4	Parameters used in comparison runs for the 3D problem with surrounding non-pay rocks. . . . .	110
6.1	Total numbers of linear solver iterations and CPU times in seconds for GMRES and BiCG-STAB methods. . . . .	115
6.2	Total number of (GMRES) linear iterations for different domain decompositions. . . . .	121
6.3	Total number of linear solver iterations and CPU time for one-level and multilevel DD preconditioners. . . . .	129
6.4	Example 1 with varying grid refinements: Total number of linear solver iterations for one-level DD (DDa-IC) and multilevel DD (MLDD-IC/SCMG) preconditioners. . . . .	146
6.5	Example 1 with $64 \times 64 \times 32$ grid cells: Comparison of memory storage, linear solver initialization time and solution time per iteration for the incomplete Cholesky factorization with zero and one fill-in levels. . . . .	147
6.6	Example 1 with varying grid refinements: Total number of linear solver iterations using DiD-AMG preconditioners. $(n_x, n_y, n_z)$ indicates the dimension of a small subgrid block for aggregation. $k_1/k_2$ denotes the number of pre/post smoothings using multi-color line Gauss-Seidel. . . . .	149
6.7	Example 1 with $64 \times 64 \times 32$ grid cells: Comparison of memory storage, linear solver initialization time and solution time per iteration for DiD preconditioners. . . . .	149
6.8	Example 2 with $64 \times 64 \times 32$ grid cells with varying Poisson's ratios: Total number of linear solver iterations using SCMG with different SOR relaxation parameters $\omega$ . $(k_1/k_2)$ indicates the number of pre- and post- smoothings in SCMG. . . . .	153
6.9	Example 2 with $64 \times 64 \times 32$ grid cells and varying Poisson's ratios: Total number of linear solver iterations using multilevel DD (MLDD) preconditioners. . . . .	155

6.10	Example 2 with $64 \times 64 \times 32$ grid cells and varying Poisson's ratios: Total number of linear solver iterations for DiD-AMG preconditioners. $k_1/k_2$ indicates the number of pre- and post-smoothings. . . . .	155
6.11	Example 3 with varying magnitudes of jumps in Young's modulus: Total number of linear solver iterations for DiD-AMG and MLDD preconditioners. . . . .	158
7.1	Example 1: Comparison of SCMG with different number of coarse grid levels. . . . .	177
7.2	Example 2: Comparison of total linear iteration counts for different preconditioners. . . . .	186
7.3	Example 2: Comparison of total CPU time in seconds for different preconditioners. . . . .	187
7.4	Example 2: Comparison of parallel efficiency for different preconditioners. . . . .	188

## List of Figures

1.1	Integrated technologies for geomechanical modeling. . . . .	6
4.1	Example 1: Confined reservoir with no flow boundaries. . . . .	73
4.2	Example 2: Unconfined reservoir with no flow boundaries. . . . .	74
4.3	Example 3: Confined reservoir with surrounding non-pay rocks. . . . .	76
4.4	Example 3: Finite element discretization . . . . .	77
4.5	Example 1 with varying fluid compressibility and rock properties: Number of Newton iterations for the first two time steps. . . . .	80
4.6	Example 3 with $E_1 = 10^4$ psi and $E_2 = 10^6$ psi: Time evolution of Newton iteration counts per time step over the entire reservoir depletion. . . . .	84
4.7	Examples 4 & 5: Residual reduction histories with respect to the number of Newton iterations in the second time step. Upper: Example 4; Lower: Example 5. . . . .	87
4.8	Total numbers of grid points using $\mathbf{S}_{p2}(\mathbf{M}_{cr}^I)$ and $\mathbf{S}_{p3}(\mathbf{M}_{cr}^{II})$ respectively at different reservoir depletion time. Upper: Example 4; Lower: Example 5. . . . .	88
5.1	Mandel's problem. . . . .	96
5.2	Mandel's problem computation domain. . . . .	97
5.3	Mandel's problem pore pressure distribution along the x-axis. . . . .	102
5.4	Mandel's problem stress ( $\sigma_{yy}$ ) distribution along the x-axis. . . . .	102
5.5	Mandel's problem displacement $u_x$ along the x-axis. . . . .	103
5.6	Mandel's problem displacement $u_y$ along the x-axis. . . . .	103
5.7	Mandel's problem pore pressure history along the x-axis. . . . .	104
5.8	Pore pressure history at cell (6, 11, 6) for the 3D problem. . . . .	107
5.9	Pore pressure history at cell (11, 11, 6) for the 3D problem. . . . .	107
5.10	Pressure Mandel-Cryer zones at the end of 20 and 60 days for the 3D problem. . . . .	108
5.11	Pressure Mandel-Cryer zones at the end of 220 and 600 days for the 3D problem. . . . .	109

5.12	Comparison of average reservoir pressure history. . . . .	111
5.13	Comparison of vertical displacements from cell (11,11,1) to cell (11,11,12). . . . .	111
5.14	Comparison of pore pressure history at boundary cell (6,11,6). . . . .	112
6.1	Super-coarsening multigrid preconditioner. . . . .	124
6.2	Example 1 with $64 \times 64 \times 32$ grid cells: Convergence histories of multi-color line Gauss-Seidel preconditioners for the elasticity system and the displacement decomposed system. . . . .	137
6.3	Example 1 with $64 \times 64 \times 32$ grid cells: Convergence histories of one-level DD and multilevel DD (MLDD) preconditioners. . . . .	148
6.4	Example 1 with $64 \times 64 \times 32$ grid cells: Convergence histories of DiD preconditioners. DiD-AMG( $n_x, n_y, n_z$ )/ $k$ -LineGS indicates that AMG is applied to a coarse grid (generated by aggregations over each subgrid block of $n_x \times n_y \times n_z$ ) with $k$ sweeps of post-smoothing. . . . .	150
6.5	Example 1 with $64 \times 64 \times 32$ grid cells: Convergence histories of DiD preconditioners. DiD-AMG( $n_x, n_y, n_z$ )/ $k$ -LineGS indicates that AMG is applied to a coarse grid (generated by aggregations over each subgrid block of $n_x \times n_y \times n_z$ ) with $k$ sweeps of post-smoothing. . . . .	151
6.6	Example 2 with $64 \times 64 \times 32$ grid cells and $\nu = 0.44$ : Total linear solver solution time using SCMG with point Gauss-Seidel and SOR smoothers of varying smoothing steps. . . . .	154
6.7	Example 2 with $64 \times 64 \times 32$ grid cells and varying Poisson's ratios: Total linear solver solution time for multilevel DD and DiD preconditioners. . . . .	156
6.8	Example 3 with varying magnitudes of jumps in Young's modulus: Convergence histories of different space decomposition methods. Upper: DiD preconditioners; Lower: multilevel DD preconditioners. . . . .	159
6.9	Example 3 with varying magnitudes of jumps in Young's modulus: Linear solver solution times for DiD-AMG and MLDD preconditioners. . . . .	160
6.10	Example 3 with a refined grid ( $24 \times 48 \times 48$ ) and varying magnitudes of jumps in Young's modulus: Linear solver solution time for DiD-AMG and MLDD preconditioners. . . . .	161
6.11	Example 4: Residual reduction history vs. computer running time in the first Newton step. . . . .	163

6.12	Example 4: Total CPU time comparison. Upper: comparison of one-level DD and multilevel DD preconditioners; Lower: comparison of multilevel DD and DiD preconditioners. . . . .	164
7.1	Parallel scaling of the black-oil model on a 64-node Beowulf Pentium II 400 MHz PC cluster with a 1.28 gigabit/sec Myrinet network. . . . .	166
7.2	Grid partitioning based on domain decomposition. . . . .	167
7.3	Example 1: Comparison of DiD-DD-AMG(1,1,1) with and without SCMG corrections. . . . .	179
7.4	Example 1: Comparison of DiD-DD-AMG(2,2,2) with and without SCMG corrections. . . . .	181
7.5	Example 2: Reservoir porosity distribution. . . . .	183
7.6	Example 2: Reservoir surface depth. . . . .	184
7.7	Example 2: Oil saturation profile at $y = 2582.1$ ft. . . . .	184
7.8	Example 2: Poisson's ratio. . . . .	185
7.9	Example 2: Young's modulus. . . . .	185
7.10	Example 2: Well distribution in layer I. . . . .	186
7.11	Example 2: Well distribution in layer II. . . . .	187
7.12	Example 2: Domain decomposition with 16 processors. . . . .	188
7.13	Example 2: Comparison of DiD-DD-AMG(2,2,5) with and without SCMG corrections. . . . .	189

# Chapter 1

## Introduction

### 1.1 Motivation

Land subsidence due to the exploitation of subsurface resources, its damage to surface infrastructures, and its impact on our environment have triggered extensive studies in the subsurface modeling of fluid flow and geomechanics.

#### 1.1.1 Problem Statement

Major problems caused by excessive withdrawal of underground fluids may be summarized as follows:

#### Land Subsidence

Compaction of unconsolidated aquifer systems that accompanies excessive groundwater pumping is by far the single largest cause of subsidence in the United States, and the increasing development of land and water resources threatens to exacerbate the existing subsidence problem while initiating new ones.

Surface subsidence may also be induced by hydrocarbon withdrawal

from weak formations and chalk reservoirs. While reservoir compaction itself has been widely recognized as an additional driving mechanism for increasing oil & gas recovery, its side effects are undesirable. The most obvious one is surface or seafloor settlement, which may create environmental problems and cause damage to oilfield structure and seabed pipelines. Some well-known fields which have experienced severe subsidence include the Wilmington field in California (Allen 1968 & 1972), the Ekofisk field in the North sea (Sulak & Danielsen 1989 and Hermansen et al. 1997), the South Belridge field in California (Hansen et al. 1993), and the westland loss in Gulf of Mexico region (White & Morton 1997), though this is by no means an exclusive list of fields that have experienced severe subsidence.

### **Wellbore Stability**

Loss of wells and/or production due to severe casing damage caused by formation subsidence can be very costly, particularly in shallow reservoirs with high porosity. For example, in the Lost Hills and Belridge fields (California) the major oil reservoir consists of diatomite sandstone with high porosity (0.45-0.7) and low permeability (Fredrich et al. 1998 and 2000). Extraction of large volumes of fluid, aided by hydro-fracturing in the low-strength diatomite formations at a shallow depth (about 700 m below the surface), have caused large pressure depletion that have resulted in significant reservoir compaction under the weight of overburden (Fielding 1998). The subsequent field-wide wellbore failures and casing damage have been expensive due to loss of pro-

duction and well replacements. By 1987, 10 to 15 ft of cumulative surface subsidence was estimated in some portions of the field, and more than 100 wells were being abandoned annually due to severe casing damage. While a water injection program started in the late 1980's somewhat alleviated the subsidence, the well failure rate is still economically significant at 2-6% of active wells per year (Fredrich et al. 1998 and 2000). Nearly 1000 wells have experienced severe casing damage during the past 20 years. The cost of well replacements still amounts to millions of dollars per year.

### **Sand Production**

For many decades, sand production from unconsolidated formations has been a worldwide challenge for the petroleum industry. The challenge is not merely to avoid or stop sand production but to be able to maintain commercial well productivity after efforts to control sand have been implemented. The selected control measure must be justified by a reasonable investment payback time.

Every year, the industry spends millions of dollars on cleaning out sand from wells and repairing damage associated with sand production. As a result, tremendous production quantities are lost or deferred. Sand production with its associated erosion and effects on equipment also represents a potential safety hazard. Consequently, huge investments have been made in many oil and gas fields worldwide to prevent sand from being produced to the surface. Also, there is an ever-growing demand for both analytical and numerical tools

to be able to monitor, analyze, and predict sand production, as well as evaluate different sand control methods and predict well performance under sand control. However, there is a common belief that the methods available to the industry today are, in spite of rigorous research, inadequate.

According to an estimate by the National Research Council in 1991 (National Research Council 1991), the annual cost in the United States from flooding and structural damage alone caused by land subsidence was more than \$125 million, not including the amounts spent to control and mitigate damage from subsidence. Due to difficulties in identifying and mapping the affected areas, establishing cause-and-effect relations, assigning economic value to environmental resources, and inherent conflicts in the legal system regarding these damages, the total costs of subsidence may be significantly larger than the current best estimates. More and more, some environmental agencies, the petroleum industry and the mining industry demand analytical and numerical tools for fundamental insights and a better understanding of the interactions between fluid flow, heat transfer and stress-strain behavior in porous media.

There is a growing trend for so called “integrated technologies for geomechanical modeling” that incorporates geological, porous flow and heat transfer modeling with or without chemical reactions, 4D time-lapse seismic analysis together with poromechanics modeling (see Figure 1.1). Objectives for such complex multi-field simulations are:

1. Better understanding of the dynamic fluid-structure interaction;

2. Accurately predicting underground deformation and surface subsidence due to the over-exploitation of underground natural resources and/or underground storage of energy residues;
3. Effectively monitoring small changes in land surface elevation with an unprecedented level of spatial detail;
4. Providing cost effective damage prevention or control.

In this thesis, investigations are focused on the coupling of subsurface flow (single phase and multiphase) with poroelasticity. We emphasize seeking the accurate numerical schemes for efficiently solving the coupled system on parallel computers. Complete integration of different numerical models as described in Figure 1.1 is a direction of our future research.

### **1.1.2 Challenges in Coupled Simulations**

While integrated geomechanical modeling has extensive energy and environmental applications, numerical modeling of such coupled physical processes has been, historically, considered extremely complex and unfeasible due to limited computing resources. In actual practice, either assumptions about part of the interaction process, which are not of primary interest, must be made, or a 3D coupled problem has to be approximated by a 1D or 2D model. For example, in conventional reservoir simulation the effects of rock compaction and porosity change are only partially accounted for by adding a rock compressibility term to the pressure equation. By doing so the porous

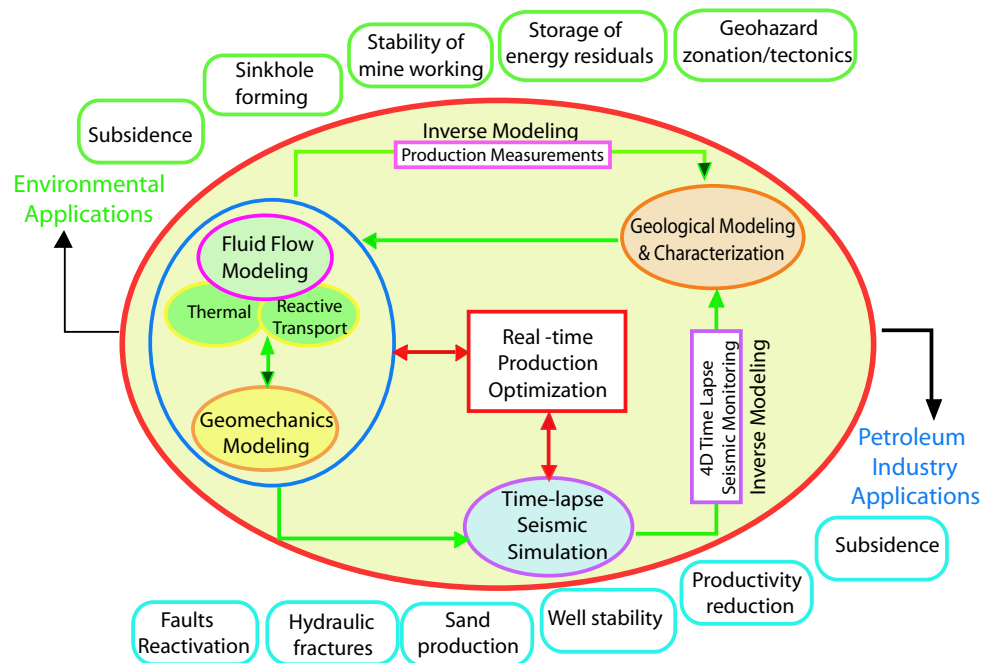


Figure 1.1: Integrated technologies for geomechanical modeling.

flow modeling is completely decoupled from solid mechanics calculations. Another example is groundwater modeling in which porous flow is modeled in 3D but compaction is typically simulated as a 1D process. While the poroelastic theory developed by Biot in 1941 provides the fundamental basis for 3D consolidation analysis, scientists and engineers commonly invoke the one-dimensional theory of hydrodynamic consolidation by Terzaghi in 1925.

The above simplified models, either through decoupling or one dimensional approximation, are only appropriate and reasonably accurate under certain circumstances such as competent rocks. They are unacceptable, however, in the case that the underlying physics involves a strong coupling of fluid flow and solid deformation. In stress-sensitive reservoirs, rock deformation, porosity and permeability changes as well as rock failures can not be fully represented by the rock compressibility term alone. Moreover, the common assumption of 1D consolidation in groundwater modeling is motivated by an obvious truism, i.e., most aquifer-system compaction or reservoir compaction takes place in the vertical dimension. Nevertheless, the widespread occurrence of earth fissures indicates that horizontal deformation may be locally significant.

Since the advent of inexpensive high-speed digital computers, scientists and engineers have had the ability to simultaneously solve multiple field equations, such as thermoporoelastoplasticity, single phase or multiphase flow and heat transfer problems. The integrated analysis can be carried out in a loosely coupled fashion or with a tightly coupled scheme. However, challenges still ex-

ist in large scale, full-field 3D applications with a spatial resolution similar to the one in 3D seismic modelings. These challenges lie in the intensive demands for computational time and memory storage, which are attributed to:

1. Large coupled systems which include mass conservation equations for flow and force balance equations for elasticity. In the case of the black-oil model coupled with 3D poroelasticity, there are six primary unknowns. Thus, effective linear solvers need to be applied for solving the system efficiently and robustly.
2. The complex nonlinear behaviors of coupling multiphase flow and solid mechanics can result in slow nonlinear convergence.
3. Several coupling techniques have been proposed and widely used for solving multi-field equations with different time scales for each model. However, judging the trade-offs between accuracy and efficiency is difficult.
4. A practical and reliable scheme for applying adaptivity in time and space is still not available due to a lack of detailed theoretical analysis.
5. Field observations suggest that, while pressure depletion is a local process that only occurs inside aquifers or reservoirs, it triggers a redistribution of effective stress in a more extensive domain. In order for numerical solutions to be accurate, the computational domain needs to be as large as possible. Gutierrez and Lewis (1998), and Osorio (1997a

and 1997b) suggest that the domain should include overburdens, sideburdens and underburdens for a better representation of the changing reservoir boundary conditions.

In summary, we are solving a coupled system in the large physical domain that is preferred on a full-field scale with great vertical depth. Such a coupled analysis involves a large linear system with millions of unknowns, whose solution needs to be computed iteratively on parallel machines.

## 1.2 Objectives of This Work

In this work, we will adopt an iterative coupling technique to solve the single phase and the multiphase (water, oil and gas) flow equations in deformable porous media. The attraction of this method lies in that it is more stable and accurate as compared to a loosely coupled approach. Another attractive feature, which is favored by reservoir engineers, is the computational ease in coupling an existing porous flow simulator with an existing geomechanics simulator. However, a primary drawback to the iterative scheme is its slow convergence rate that may result in a large number of nonlinear iterations for difficult problems. Therefore, the first objective of this research is to examine the method that was originally proposed based on physical intuition. Using the similarity of the coupled system to the Stokes or generalized Stokes equations, an iterative scheme can be formulated as the Uzawa method. The reformulation sheds light on convergence analysis of the conventional iterative

approach, which uses rock compressibility to approximate volumetric strains. In doing so, new preconditioners may also be derived to improve the nonlinear convergence rate.

However, the convergence behavior of an iterative scheme depends crucially on the accuracy of the displacement solutions. Solving the discrete elasticity system by Galerkin finite element (FE) discretization uses more CPU time and memory than solving the flow system generated by an expanded mixed finite element (MFE) scheme. Thus, another major effort put forth in this work is the investigation of fast, efficient, and parallel scalable linear solvers and preconditioners for the solution of displacement.

## **1.3 Literature Review**

### **1.3.1 Theory and Governing Equations**

The first attempt to describe fluid-solid coupling in a deformable porous medium is attributed to the work of Terzaghi in 1925 (Terzaghi 1925). He introduces the concept of effective stress for incompressible solid grains. Until now his one-dimensional consolidation theory has been used extensively in subsidence studies and basin modelings. Based on Terzaghi's work, Biot (1941a, 1941b) establishes the general theory of three-dimensional consolidation in a framework consistent with the basic principles of continuum mechanics. In his subsequent works (Biot 1955, 1956a, 1956b, 1956c, 1957, 1962, 1973), Biot extends the poroelastic theory to anisotropic and nonlinear materials. While Biot's original theory assumes linear behavior for the solid matrix, it may

easily be generalized to complex models dealing with nonlinear problems and thermal effect (Small et al. 1976, Coussy 1989, Lewis and Schrefler 1998).

Several excellent reviews or re-interpretations of Biot’s consolidation equations have been presented by Geertsma (1957), Verruijt (1969), Ghaboussi & Wilson (1973), Rice and Cleary (1976), Jaeger and Cook (1979), Lewis & Schrefler (1987), Detournay and Cheng (1993), Bear and Bachmat (1990), Chen (1995), and Zienkiewicz et al. (1999). In particular, Ghaboussi & Wilson (1973) introduce fluid compressibility to the consolidation theory, and Rice & Cleary (1976) define several material coefficients for an undrained system that have been widely used in practice.

Several authors have presented the mathematical formulation for modeling poroelastic multiphase flow (Tortike & Farouq Ali 1987, Lewis & Sukirman 1993, Lewis & Schrefler 1998 and Li & Zienkiewicz 1992). It is worth mentioning that the re-interpretations of Biot’s poroelasticity theory by Verruijt (1969) and Bear & Bachmat (1990) are most pertinent to porous flow modeling coupled with geomechanical features. Both works, however, assume incompressible solid constituents. Adopting the same methodology, Chen et al. (1995) presents a mathematical model that takes into account the solid grain’s compressibility. In their formulation geomechanics is included in a conventional reservoir flow model as an additional module. They describe how conventional fluid-flow modeling can be extended to coupled fluid-flow and geomechanics modeling. Identification of the linkages and consistent interpretations between the flow and deformation fields are emphasized. While the

governing equations are based on single phase flow, they can be extended to the case of multiphase flow with consistency.

### **1.3.2 Coupled Geomechanics and Reservoir Flow Modeling**

Settari & Waters (1999) discuss different methods that have been used to combine poroelastic and multiphase flow calculations. Based on the degree of coupling, they categorize these methods into decoupled, explicitly coupled, iteratively coupled and fully coupled.

Early works in coupled analysis are primarily in a decoupled fashion (Chin & Boade 1990, Sulak et al. 1991 and Fredrich et al. 1996). Chin & Boade (1990) use an essentially decoupled reservoir simulator and a compaction model to study sea floor subsidence in the Ekofisk oil field. Sulak et al. (1991) compute a transient stress solution based on pressure histories from reservoir simulations. Their stress solution is then used to update porosity and permeability manually.

Minkoff et al. (1999) and Koutsabeloulis & Hope (1998) present a partially or explicitly coupled scheme for the multiphase flow with/without thermal effect. In Minkoff et al. (1999) a cell-centered finite difference scheme is used to discretize the reservoir flow equations while a finite element formulation is applied for the stress model. In the case that different grids are used for each model, projection algorithms are required for mapping the primary variables from one field to the other.

Representative works of iterative coupling can be found in Settari &

Mourtis (1994 and 1998), Tortike & Farouq Ali (1992), and Fung et al. (1994). Thomas et al. (2002) and Gai et al. (2003) describe an iteratively coupled model that employs parallel computing.

Lewis & Sukirman (1993) demonstrate their numerical studies of three-dimensional three-phase flow in a deforming hydrocarbon reservoir. A finite element method is applied to obtain simultaneous solutions of displacement and fluid pressure. Based on Lewis & Sukirman's formulation, Gutierrez & Lewis (1998) describe a fully coupled scheme to examine the role of geomechanics in reservoir simulations. They point out that one attractive feature of the fully coupled scheme lies in its ability to use the same discretization for both flow and mechanics calculations. Based on Chen et al.'s formulation for coupled single phase flow, Osorio et al. (1997 and 1999) develop a 3D finite difference, fully implicit model to study the effect of rock compaction on reservoir productivity. The numerical procedure used in their program is a Picard-like block Gauss-Seidel method. Therefore, the fully coupled scheme is essentially an iterative method incorporated with outer loops. Stone et al. (2003) integrate a 3D elastoplastic model into a commercial reservoir simulator. They apply a finite difference discretization to the stress equations on staggered grids. Chin et al. (1998) develop a fully coupled model (single phase flow with FE approximation) for pressure transient analysis in stress-sensitive reservoirs. Prevost (1997) presents a partitioned solution procedure for simultaneous integration of coupled-field problems. The decoupling of the multi-field equations is achieved in the linear solution phase. Previous efforts

to seek simultaneous solutions of pressure and displacement may also be found in Li & Zienkiewicz (1992) and Tortike & Farouq Ali (1992).

Dean et al. (2003) incorporate the three coupling techniques—explicitly coupled, iteratively coupled and fully coupled—into the same program and compare the performance of each method in terms of stability, accuracy and computational efficiency.

Wan (2002) addresses stability issues in the case that unstable finite element spaces are used for pressure and displacement fields. A stabilized Galerkin finite element method is used to alleviate pressure oscillations. For the same purpose, Liu (2004) adopts a discontinuous Galerkin (DG) FE scheme to discretize the equilibrium equations. One attractive feature of Liu’s scheme is that it allows local mass conservation for flow.

### **1.3.3 Linear Solvers for the Coupled System**

Due to the complexities involved in solving two-field equations, extending a conventional reservoir model to a coupled fluid-flow and geomechanics model is not trivial, even though considerable success has been achieved in recent years. As mentioned above, mathematical and numerical formulations have been developed; different coupling techniques have been investigated; and there is a considerable quantity of literature covering field applications of coupled geomechanics and reservoir flow modeling. Recently, stability issues have received more attention with regard to non-physical pressure oscillations in low permeability zones (Wan 2002 and Liu 2004). Coupled simulations are

employing increasingly sophisticated individual models and more accurate and stable numerical schemes. However, the computational bottleneck of solving a large coupled system still remains. Few works mentioned above investigate the linear solution techniques in coupled analysis. Frequently, a direct method is applied, but the solution cost and storage requirements increase dramatically for three-dimensional problems. Thus, the total number of DOF is highly limited in practice.

Dean (2000) describes a three-step GCR (generalized conjugate residuals) scheme for solving a linear system involving pressure and displacements. The pressure block is solved by a PCG-like iterative solver while the displacement block is computed by a direct or ICCG (incomplete Choleski conjugate gradient) method. Prevost (1997) proposes a partitioned solution procedure for the simultaneous integration of transient coupled field problems. An iterative partitioned conjugate gradient method is used. It allows the use of existing single-field analysis software modules as preconditioners. But the success of the iterative scheme requires that both the flow and displacement matrices be symmetric and positive definite. Thus, it is not feasible to apply the method directly to fully implicit multiphase flows. In Lipnikov (2002), the Lanczos method is used to solve a linear system arising from the discretization of Biot's poroelasticity equations. As with Prevost's scheme, his block diagonal preconditioners involve the solution of two totally decoupled systems, namely, the discrete Lamé and diffusion operators. Problems arise in the case of incompressible flow with pure Neumann boundary conditions

since the pressure solution is only unique up to a constant. Thus special treatments are required for the pressure solutions. Phoon et al. (2003) presents the symmetric quasi-minimal residual method for Biot’s coupled system. A simple diagonal preconditioner is implemented and investigated. Instead of completely neglecting the coupling matrix that corresponds to the volumetric strain, they approximate it and add it to the pressure equations. By doing so, the convergence rate is improved.

A unique feature of the iteratively coupled scheme is that it obviates the need for developing special linear solvers for the global system (flow and displacement). The building blocks are the inversion, or approximate inversion, of the matrices for flow and displacement, respectively. Usually, linear solvers for multiphase flows are well developed in most commercial reservoir simulators. Thus, efficient, robust and parallel scalable linear solvers and preconditioners need to be developed for the discrete elasticity system.

There is an abundance of literature dealing with preconditioning iterative solution methods for stress analysis, especially in the context of linear elasticity. Some work covers specific topics, such as thin domain structure in 3D (plates and shells) and mixed/penalty methods for incompressible problems. Here we briefly comment on some of the approaches that are well suited for general 3D pure displacement problems on parallel computers.

Domain decomposition (DD) is well-known as a flexible method for the solution of linear or nonlinear partial differential equations. In particular, the multigrid methods offer the prospect of optimal scaling with problem

size (Stüben 1983), either as stand-alone methods, or as accelerators for the Krylov subspace methods. Several authors have suggested the use of multigrid for stress analysis, e.g., Bulgakov & Belyi (1992) and Braess (2001). Successful applications of multigrid methods to linear and nonlinear solid mechanics problems can be found in Fish et al. (1993), Chan & Smith (1994) and Feng et al. (1997 and 1998) for single processors, and Kacau & Parsons (1993), Lang et al. (2000) and Adams (2000) for multiple processors. However, multigrid methods require a hierarchical grid structure that is not readily available in unstructured grids. To overcome this problem, algebraic multigrid (AMG) has been introduced. Recently, great success has been achieved in applying the AMG algorithm directly to systems of PDEs (Ruge 1986, Vanek et al. 1995, Bulgakov & Kuhn 1995 and Stüben 2000 on single processors, and Adams 2002 on parallel machines). Note that all the methods mentioned above are specifically designed for unstructured grids.

FETI (finite element tearing and interconnecting) (Farhat 1991) represents the class of non-overlapping domain decomposition (DD) approaches that are introduced for the parallel finite element solution of equilibrium equations. The idea is to partition the spatial domain into a set of totally disconnected subdomains, each assigned to an individual processor. Lagrange multipliers are introduced to enforce compatibility at the interface nodes. A parallel conjugate projected gradient algorithm is developed for the solution of the coupled system. Other variants of the FETI algorithm include the two-level FETI (Farhat & Mandel 1998) and the dual-primal FETI (Farhat et al. 2000

and 2001).

Another family of preconditioners frequently used in structural mechanics is the so-called displacement decomposition (DiD) method. In DiD preconditioners are constructed by block partitioning of a stiffness matrix based on a SDC (separate displacement component) ordering of displacement variables. The decomposed systems have essentially the same properties as scalar PDEs, for which a variety of efficient preconditioning and solution procedures exist. Up to now, the DiD method remains as one of the most robust approaches available.

Axelsson and Gustafsson (1978) implement their preconditioners based on the point-ILU factorization of the DiD system. Using Korn's inequality, they demonstrate that the preconditioned matrix has a conditioner number that is independent of mesh size  $h$  but dependent on the Poisson's ratio. Similar results can be found in Blaheta (1994). Several block-ILU factorization techniques for the DiD system are proposed in Axelsson (1983), Axelsson et al. (1984) and Axelsson & Polman (1986). Their robustness and parallel vectorizability are discussed in Axelsson (1983), Axelsson & Polman (1986), and Axelsson & Eijkhout (1987).

Gustafsson & Lindskog (1998) analyze the block diagonal preconditioners based on the SDC part of the elasticity equations. Each subproblem is solved by the PCG method with a modified incomplete factorization MIC(0) preconditioner. In Gustafsson & Lindskog (2002), full block incomplete factorization preconditioners are presented and analyzed. To avoid inner/outer

iterations they replace the inner PCG iterations by their MIC(0)-factors. Special finite element discretization and node numbering lead to partial parallelism for the MIC(0) preconditioners. However, in order to guarantee the existence of MIC(0) factorization, the preconditioning matrices need to satisfy the *maximum principle* (to be M-matrices). Chan et al. (1997) present block-ILU factorization preconditioners based on block-size reduction for 2D elasticity systems. Their factorization exists for symmetric and positive definite block-tridiagonal matrices that are not necessarily M-matrices.

Padiy (1999) suggests a DiD (block diagonal) preconditioner for the discretized linear elasticity problems on a tensor product of two-dimensional and one-dimensional meshes. The author considers an approximation of the diagonal blocks by the additive AMLI (algebraic multilevel iteration) method (Axelsson & Vassilevski 1989 and Axelsson & Vassilevski 1990). Mihajlović & Mijalković (2002) adopt a scalar algebraic multigrid (AMG) solver to obtain the approximate solution of each subproblem. They demonstrate the superiority of DiD-based AMG preconditioners over the standard ILU preconditioners. Similar to the analysis in Axelsson (1978), the spectrum of their DiD/AMG preconditioners is bounded independently of the mesh parameter ( $h$ ) though not independently of the problem parameter (Poisson's ratio).

While the realization of DiD-based preconditioners can be vectorized, they only allow partial parallelism. For massively parallel, or at least medium-sized computer systems (fewer than one hundred processors), DiD methods need to be used together with DD methods. The idea is to consider the com-

ponents of the displacement vector as a base for additional space decomposition. Domain decomposition is performed in such a way that local subspace problems are solved for the material displacement along a single Cartesian coordinate. Blaheta et al. (2003) address the use of DiD and overlapping DD preconditioners for linear elasticity problems. In their two level schemes, each subdomain problem is approximately solved by incomplete factorization (DiD-MIC(0)) while the coarse grid problem is solved by inner PCG iterations preconditioned by DiD-MIC(0). To handle nonlinear preconditioning by the inner PCG iterations, a GPCG (generalized preconditioned conjugate gradient) method is used for outer iterations.

Another variant of DiD preconditioners for parallel computing is presented by Lirkov (2003). After constructing the decoupled block-diagonal part of the original stiffness matrix, a circulant block factorization is used for solving each displacement component.

## 1.4 Outline of the Thesis

Discussions in this thesis are organized as follows:

1. In Chapter 2, a mathematical model for a three-phase, three-component black-oil system in deformable porous media is derived. Mass conservation and equilibrium equations, as well as boundary conditions, for a coupled problem are presented.
2. In Chapter 3, a finite element formulation for the coupled system is

described.

3. In Chapter 4, the iteratively coupling technique is reformulated. The optimality of incorporating rock compressibility terms as preconditioners is discussed. Several other preconditioners are also proposed, and their performance is compared with respect to grid refinement, jumping coefficients and the changes of fluid and rock properties.
4. In Chapter 5, we present two numerical examples to verify our numerical model for the coupled system.
5. Chapter 6 is devoted to the discussion of linear solvers and preconditioners for the discrete elasticity block.
6. In Chapter 7, parallel implementation issues for the poroelastic model are addressed. In particular, the parallel aspects of a super-coarsening multigrid method are illustrated. Performance of the proposed linear solver and preconditioners is evaluated in terms of parallel scalability and efficiency.
7. Chapter 8 concludes the thesis with a summary of accomplishments and recommendations for future research.

## Chapter 2

### Mathematical Model

In this chapter we present mathematical models for describing coupled multiphase flow with geomechanical calculations in a deformable hydrocarbon reservoir. The coupled-field problems are characterized by the interaction of essentially different physical phenomena within the same material domain. Biot's self-consistent theory is used to develop the governing equations. Rock mechanical behavior in deformable porous media will be also investigated.

#### 2.1 Linear Porelasticity Theory

Many sedimentary rocks display instantaneous and, above all, reversible elastic behavior. Biot's poroelasticity theory is derived by extending the general linear elasticity theory for continuum media to fully saturated porous media. There are numerous volumes that deal exhaustively with linear elasticity, such as Timoshenko and Goodier (1969). In all subsequent discussion, stress, strain and pore pressure are defined relative to an initial state unless stated otherwise, and they follow a sign convention wherein the tensile stress and strain are taken as positive. Repeated subscript indexes imply summation.

The first attempt to describe fluid-solid coupling in a deformable porous

medium is found in Terzaghi (1925). He introduced the concept of effective stress for incompressible solid grains in a strictly one-dimensional framework. His one-dimensional consolidation theory has been widely used in practice to study landsurface subsidence problems. Later, Biot (1941a, 1941b) generalized the theory to the three-dimensional case in a framework consistent with the basic principles of continuum mechanics. In his subsequent works (Biot 1955, 1956a, 1956b, 1956c, 1957, 1962, 1973), Biot extended the poroelastic theory to anisotropic and nonlinear material.

Biot's consolidation equations consist of equilibrium equations for an element of the solid frame, stress-strain relations for the solid skeleton, and a continuity equation for the pore fluid. The theory is based upon the following assumptions:

1. The solid phase is assumed to comprise a porous skeleton of particles surrounded by one or more fluids (liquid or gaseous hydrocarbons and water);
2. The shear stresses in the fluid phases are small while a surrounding pressure is exerted on the solid phase;
3. Solid material is isotropic with respect to rock mechanical properties;
4. The equilibrium equation accounts for only quasi-static behavior, and inertial effects are neglected;

5. The small-strain deformation is assumed so that linear elasticity theory applies;
6. An isothermal condition is assumed.

The poroelastic equations are formulated in terms of total stresses, bulk strains and pore pressure. The three basic principles in the poroelastic theory (Biot 1941) are described as below.

**Stress equilibrium equations:**

$$-\nabla \cdot \boldsymbol{\sigma} = \mathbf{f}, \quad (2.1)$$

where  $\boldsymbol{\sigma}$  is the total stress,  $\boldsymbol{\sigma} = (\sigma_{11}, \sigma_{22}, \sigma_{33}, \sigma_{12}, \sigma_{23}, \sigma_{13})^T$ . The gravity term  $\mathbf{f}$  is a function of porosity ( $\phi$ ), fluid saturations ( $S_o, S_w, S_g$ ), fluid densities ( $\rho_o, \rho_w, \rho_g$ ) and solid density ( $\rho_s$ ), i.e.,

$$\mathbf{f} = [\rho_s(1 - \phi) + \phi(\rho_o S_o + \rho_w S_w + \rho_g S_g)] \mathbf{g}. \quad (2.2)$$

**Isotropic constitutive equations:**

$$\boldsymbol{\sigma} = \frac{E}{1 + \nu} \left( 1 + \frac{\nu}{1 - 2\nu} \mathbf{m} \mathbf{m}^T \right) \boldsymbol{\varepsilon} - \alpha \mathbf{m} p, \quad (2.3)$$

$$\boldsymbol{\varepsilon} = \left( \frac{1 + \nu}{E} - \frac{\nu}{E} \mathbf{m} \mathbf{m}^T \right) \boldsymbol{\sigma} + \alpha \frac{1 - 2\nu}{E} \mathbf{m} p. \quad (2.4)$$

Here,  $\boldsymbol{\varepsilon}$  is the strain vector,  $\boldsymbol{\varepsilon} = (\varepsilon_{11}, \varepsilon_{22}, \varepsilon_{33}, \varepsilon_{12}, \varepsilon_{23}, \varepsilon_{13})^T$ ;  $p$  is the pore pressure;  $\alpha$  is one of Biot's constants;  $\mathbf{m} = (1, 1, 1, 0, 0, 0)^T$ ;  $E$  and  $\nu$  are

Young's modulus and Poisson's ratio respectively. Often, it is more convenient to express the above stress-strain relations in terms of Lamé's constants, i.e.,

$$\boldsymbol{\sigma} = (\lambda \mathbf{m} \mathbf{m}^T + 2\mu) \boldsymbol{\varepsilon} - \alpha \mathbf{m} p, \quad (2.5)$$

$$\boldsymbol{\varepsilon} = \frac{1}{2\mu} \left( 1 - \frac{\lambda}{3\lambda + 2\mu} \mathbf{m} \mathbf{m}^T \right) \boldsymbol{\sigma} + \frac{2\mu}{3\lambda + 2\mu} \alpha \mathbf{m} p, \quad (2.6)$$

where  $\lambda$  and  $\mu$  are related to  $E$  and  $\nu$  by

$$\begin{aligned} \lambda &= \frac{\nu E}{(1 + \nu)(1 - 2\nu)}, \\ \mu &= \frac{E}{2(1 + \nu)}. \end{aligned}$$

**Strain-displacement relation:**

$$\varepsilon_{ij} = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right). \quad (2.7)$$

Numerical simulation of the poroelastic model involves solving the above equations with displacements as primary variables. Either traction or displacement needs to be specified on the boundary.

## 2.2 Fluid-Flow Theory in Deformable Porous Media

Several excellent reviews or re-interpretations of Biot's consolidation have been presented, e.g. Geertsma (1957), Jaeger and Cook (1979), Verruijt (1969), Rice and Cleary (1976), Detournay and Cheng (1993), and Bear and Bachmat (1990). In particular, the works by Verruijt and Bear are the most pertinent to the porous flow modeling coupled with poroelasticity theory.

Their coupled equations are based on the mass conservation law for both fluid and solid phases. Both works, however, assume incompressible solid grains ( $\alpha = 1$ ). Following the same approach, Chen et al. (1995) present a mathematical model for single phase flow with the solid grain’s compressibility taken into account. In this section, we shall derive mass conservation equations for a three-phase black-oil system in deformable porous media. The difference between our derivation and Lewis & Schrefler’s derivation (Lewis and Schrefler 1998) lies in the handling of solid grain’s compressibility.

### **2.2.1 Coupled Black-oil Model Mass Conservation Equations**

The black-oil model is a simplified compositional model for describing multiphase flow with mass interchange between phases. The standard black-oil model consists of three distinct fluid phases: gas, water, and oil, and two pseudo-hydrocarbon components: oil and gas. The oil component is defined as produced oil at stock tank conditions, and the gas component is defined as produced separator gas. The black-oil model (Lu 2000 and Lu et al. 2001) under the framework of IPARS (Integrated Parallel Accurate Reservoir Simulator) (Parashar & Wheeler et al. 1997, Wheeler 1998) is designed to be able to predict fluid compressibility and simple mass transfer effects between oil and gas phases. The underlying assumptions are (1) gas may dissolve in the oil phase, but oil will not dissolve in gas; and (2) no mass transfer occurs between the water phase and the other two phases. Other assumptions include:

1. The reservoir is isothermal;

2. No chemical reaction, precipitation or adsorption occurs;
3. The mass fluxes due to dispersion and diffusion are much smaller than the advective mass flux, and may, therefore, be neglected;
4. Fluid flow is characterized by Darcy's law for a Newtonian fluid;
5. Well injection and production are treated as source or sink terms;
6. The permeability tensor is diagonal;
7. The viscosity of each phase is constant;
8. Reservoir formation is slightly compressible;
9. The reservoir is surrounded by an impermeable formation so that no flow occurs on the external boundaries.

Under these assumptions, the mass continuity equations for water, oil, gas and solid components are given by the following equation. **Fluid phase mass conservation equations:**

$$\frac{\partial(\phi N_w)}{\partial t} = -\nabla \cdot \left( \phi \frac{S_w}{B_w} \mathbf{v}_{fw} \right) + q_w, \quad (2.8)$$

$$\frac{\partial(\phi N_o)}{\partial t} = -\nabla \cdot \left( \phi \frac{S_o}{B_o} \mathbf{v}_{fo} \right) + q_o, \quad (2.9)$$

$$\frac{\partial(\phi N_g)}{\partial t} = -\nabla \cdot \left( \phi \frac{S_g}{B_g} \mathbf{v}_{fg} + \phi \frac{S_o}{B_o} R_o \mathbf{v}_{fo} \right) + q_g. \quad (2.10)$$

**Solid phase mass conservation equations:**

$$\frac{\partial(1 - \phi)\rho_s}{\partial t} = -\nabla \cdot [(1 - \phi)\rho_s \mathbf{v}_s]. \quad (2.11)$$

**Darcy's law:**

$$\bar{\mathbf{v}}_\alpha = -\frac{\mathbf{k}k_{r\alpha}}{\mu_\alpha B_\alpha} (\nabla p_\alpha - g\nabla D). \quad (2.12)$$

In the above equations,  $S_\alpha$  ( $\alpha = o, w, g$ ) is the phase saturation;  $B_\alpha$  is the phase formation volume factor;  $R_o$  is the solution gas-oil ratio;  $\mathbf{v}_{f\alpha}$  is the interstitial phase velocity while  $\bar{\mathbf{v}}_\alpha$  is the Darcy velocity;  $\mathbf{k}$  is a diagonal absolute permeability tensor;  $k_{r\alpha}$  is the relative permeability for phase  $\alpha$ ;  $p_\alpha$  is the phase pressure;  $N_o$ ,  $N_w$  and  $N_g$  are the so-called concentrations for oil, water and gas components, respectively. The relations between concentrations and phase saturations are defined as

$$N_w = \frac{S_w}{B_w}, \quad (2.13)$$

$$N_o = \frac{S_o}{B_o}, \quad (2.14)$$

$$N_g = \frac{S_g}{B_g} + R_o \frac{S_o}{B_o}. \quad (2.15)$$

Note that our fluid mass conservation equations are really volume conservation equations. The basis for all mass balance is the volume of each phase at stock-tank conditions. Once the stock tank conditions are specified, the volume balances on the oil, water and gas components may be converted to mass balances through the phase densities at stock-tank conditions.

The velocity terms that appear in (2.8), (2.9) and (2.10) are the interstitial velocities. Darcy velocities and solid phase velocity enter the continuity equations through their relations to the interstitial velocities. Considering

the solid phase as the reference phase, the macroscopic Darcy velocity can be written as

$$\bar{\mathbf{v}}_\alpha = \frac{\phi S_\alpha}{B_\alpha} (\mathbf{v}_{f\alpha} - \mathbf{v}_s). \quad (2.16)$$

Substituting (2.16) into the mass balance equations for  $\mathbf{v}_{f\alpha}$  and using the relations defined in (2.13), (2.14) and (2.15) leads to

$$\frac{\partial(\phi N_w)}{\partial t} = -\nabla \cdot \bar{\mathbf{v}}_w - \nabla \cdot (\phi N_w \mathbf{v}_s) + q_w, \quad (2.17)$$

$$\frac{\partial(\phi N_o)}{\partial t} = -\nabla \cdot \bar{\mathbf{v}}_o - \nabla \cdot (\phi N_o \mathbf{v}_s) + q_o, \quad (2.18)$$

$$\frac{\partial(\phi N_g)}{\partial t} = -\nabla \cdot (\bar{\mathbf{v}}_g + R_o \bar{\mathbf{v}}_o) - \nabla \cdot (\phi N_g \mathbf{v}_s) + q_g. \quad (2.19)$$

We can further write the above continuity equations in a compact form as

$$\frac{\partial(\phi N_\alpha)}{\partial t} + \nabla \cdot \mathbf{v}_\alpha + \nabla \cdot (\phi N_\alpha \mathbf{v}_s) = q_\alpha, \quad (2.20)$$

with the Darcy velocity  $\mathbf{v}_\alpha$  for each component at the stock tank condition defined as follows,

$$\begin{aligned} \mathbf{v}_w &= \bar{\mathbf{v}}_w, \\ \mathbf{v}_o &= \bar{\mathbf{v}}_o, \\ \mathbf{v}_g &= \bar{\mathbf{v}}_g + R_o \bar{\mathbf{v}}_o. \end{aligned} \quad (2.21)$$

The coupling terms in (2.20) are  $\phi$  and  $\mathbf{v}_s$ . They account for the effect of solid deformation on flows. In conventional reservoir simulation, the term  $\nabla \cdot (\phi N_\alpha \mathbf{v}_s)$  is always neglected in light of the fact that rock deforms very slowly compared to multiphase flow, i.e.,  $\mathbf{v}_s \ll \mathbf{v}_{f\alpha}$ .

In the rest of our discussion we shall replace the solid velocity in (2.20) with displacements. Thus, the displacement variables in the poroelastic model will be shown explicitly in the porous flow equations. We rewrite the continuity equations for both the fluid and the solid phases as

$$\frac{\partial(\phi N_\alpha)}{\partial t} + \nabla \cdot \mathbf{v}_\alpha + \nabla \cdot (\phi N_\alpha \mathbf{v}_s) = q_\alpha, \quad (2.22)$$

$$\frac{\partial(1-\phi)\rho_s}{\partial t} + \nabla \cdot [(1-\phi)\rho_s \mathbf{v}_s] = 0. \quad (2.23)$$

Applying the chain rule to (2.22) and (2.23) yields

$$\frac{\partial(\phi N_\alpha)}{\partial t} + \nabla \cdot \mathbf{v}_\alpha + \mathbf{v}_s \cdot \nabla(\phi N_\alpha) + \phi N_\alpha \nabla \cdot \mathbf{v}_s = q_\alpha, \quad (2.24)$$

$$\frac{\partial(1-\phi)\rho_s}{\partial t} + \mathbf{v}_s \cdot \nabla [(1-\phi)\rho_s] + (1-\phi)\rho_s \nabla \cdot \mathbf{v}_s = 0. \quad (2.25)$$

Applying the definition of a material derivative with respect to a moving solid,

$$\frac{D(\cdot)}{Dt} \equiv \frac{\partial(\cdot)}{\partial t} + \mathbf{v}_s \cdot \nabla(\cdot), \quad (2.26)$$

(2.24) and (2.25) can be further written as

$$\frac{D(\phi N_\alpha)}{Dt} + \nabla \cdot \mathbf{v}_\alpha + \phi N_\alpha \nabla \cdot \mathbf{v}_s = q_\alpha, \quad (2.27)$$

$$\frac{D(1-\phi)\rho_s}{Dt} + (1-\phi)\rho_s \nabla \cdot \mathbf{v}_s = 0. \quad (2.28)$$

Using (2.28) we obtain

$$\nabla \cdot \mathbf{v}_s = -\frac{1}{(1-\phi)\rho_s} \frac{D(1-\phi)\rho_s}{Dt}. \quad (2.29)$$

Introducing  $1-\phi = V_s/V_b$  in (2.29) we get that

$$\nabla \cdot \mathbf{v}_s = -\frac{V_b}{\rho_s V_s} \frac{D}{Dt} \left( \frac{\rho_s V_s}{V_b} \right), \quad (2.30)$$

where  $V_b$  is the bulk volume and  $V_s$  is the solid volume. For constant solid mass ( $d(\rho_s V_s) = 0$ ), (2.30) is equivalent to

$$\nabla \cdot \mathbf{v}_s = \frac{1}{V_b} \frac{DV_b}{Dt}. \quad (2.31)$$

Recall in the linear elasticity theory that the volumetric strain  $\varepsilon_v$  is defined as

$$\varepsilon_v = \nabla \cdot \mathbf{u} = \frac{dV_b}{V_b}.$$

Introducing the above volumetric strain in (2.31) yields

$$\nabla \cdot \mathbf{v}_s = \frac{D\varepsilon_v}{Dt} = \frac{D(\nabla \cdot \mathbf{u})}{Dt}. \quad (2.32)$$

Thus, the divergence of solid velocity in (2.31) simply reflects the rate of bulk volume change. Substituting  $\nabla \cdot \mathbf{v}_s$  to (2.27) with  $D\varepsilon_v/Dt$  we have

$$\frac{D(\phi N_\alpha)}{Dt} + \phi N_\alpha \frac{D\varepsilon_v}{Dt} + \nabla \cdot \mathbf{v}_\alpha = q_\alpha. \quad (2.33)$$

After adding and subtracting  $\varepsilon_v D(\phi N_\alpha)/Dt$  we obtain

$$\frac{D}{Dt} [\phi (1 + \varepsilon_v) N_\alpha] + \nabla \cdot \mathbf{v}_\alpha - \varepsilon_v \frac{D(\phi N_\alpha)}{Dt} = q_\alpha. \quad (2.34)$$

Since the last term on the left hand side of (2.34) is small compared to the term  $\frac{D}{Dt}(\phi N_\alpha)$  which is contained in the first term, i.e.,

$$\varepsilon_v \frac{D(\phi N_\alpha)}{Dt} \ll \frac{D(\phi N_\alpha)}{Dt},$$

$\varepsilon_v \frac{D(\phi N_\alpha)}{Dt}$  can be neglected. Thus (2.34) can be written in a form quite similar to the flow equations in an uncoupled simulation,

$$\frac{D(\phi^* N_\alpha)}{Dt} + \nabla \cdot \mathbf{v}_\alpha = q_\alpha. \quad (2.35)$$

where the Darcy velocity  $\mathbf{v}_\alpha$  is defined in (2.21), and  $\phi^*$  is the so called fluid fraction and is defined as

$$\phi^* = \phi (1 + \varepsilon_v). \quad (2.36)$$

For poroelastic material with small deformation the total bulk volume  $V_b$  can be approximated by a linear function of  $\varepsilon_v$ ,

$$V_b = V_b^0 (1 + \varepsilon_v). \quad (2.37)$$

Taking into account (2.37), the fluid fraction defined by (2.36) is nothing but “porosity” relative to the initial undeformed bulk volume  $V_b^0$ , i.e.,

$$\phi^* = \phi \frac{V_b}{V_b^0} = \frac{V_p}{V_b^0}. \quad (2.38)$$

In non-deformable porous media or in the initial state of deformable porous media (2.38) implies that the fluid fraction  $\phi^*$  is equal to the true porosity  $\phi$ . Notice that  $\phi^*$  in (2.36) depends on both the pore pressure and the total stresses. In an uncoupled reservoir model, however, the contribution of total stresses to the fluid fraction is always approximated by a linear function of pore pressure, i.e.,

$$\phi^* = \phi^0 [1 + c_r(p - p_0)]. \quad (2.39)$$

Next, the true porosity term in (2.36) will be expanded to obtain a full expression of  $\phi^*$  in terms of pressure and displacements. According to Geertsma (1957) and Charlez (1991) the relative porosity variation in a de-

formed porous medium is approximated by

$$\frac{d\phi}{\phi} = \left[ \frac{1}{\phi} \left( \frac{1}{K_b} - \frac{1}{K_s} \right) - \frac{1}{K_b} \right] (d\bar{\sigma} + dp), \quad (2.40)$$

$$\bar{\sigma} = \frac{3\lambda + 2\mu}{3} \varepsilon_v - \alpha p. \quad (2.41)$$

where  $\bar{\sigma}$  is the mean stress;  $K_b$  and  $K_s$  are respectively the bulk moduli of solid skeleton and solid constituent. Biot's coefficient  $\alpha$  can be measured independently in a jacketed drained test (Biot, 1956 and Detournay & Cheng, 1993) and is associated with  $K_b$  and  $K_s$  as follows,

$$\alpha = 1 - \frac{K_b}{K_s}. \quad (2.42)$$

If small strain is assumed, the porosity  $\phi$  can be approximated from (2.40) by

$$\phi = \phi^0 + \left[ \left( \frac{1 - \phi^0}{K_b} - \frac{1}{K_s} \right) \right] (\bar{\sigma} + p). \quad (2.43)$$

Substituting (2.43) into (2.36) for  $\phi$ , dropping the second order terms, and using the relations defined in (2.41) and (2.42), we obtain

$$\phi^* = \phi^0 + [c_b - (1 + \phi^0) c_s] p + (c_b - c_s) \bar{\sigma}, \quad (2.44)$$

or in terms of  $\varepsilon$  and  $p$ ,

$$\phi^* = \phi^0 + \alpha \varepsilon_v + \frac{1}{M} p, \quad (2.45)$$

where  $\frac{1}{M}$  is the Biot constant defined as

$$\frac{1}{M} = (1 - \alpha) (\alpha - \phi_0) c_b. \quad (2.46)$$

In (2.44) and (2.46),  $c_b$  and  $c_s$  are respectively the bulk compressibility and solid grain compressibility. They are related to  $K_b$  and  $K_s$  by

$$\begin{aligned} c_b &= \frac{1}{K_b}, \\ c_s &= \frac{1}{K_s}. \end{aligned}$$

From (2.46) and the fact that  $M > 0$  we require that  $\alpha$  lies within the following bounds,

$$\phi_0 \leq \alpha \leq 1. \quad (2.47)$$

If the solid material is incompressible, i.e.,  $c_s = 0$  and  $\alpha = 1$ , or  $\alpha = \phi_0$ , then  $\frac{1}{M} = 0$ .

The final assumption that we are going to impose on the coupled mass balance equations is

$$\frac{\partial(\phi^* N_\alpha)}{\partial t} \gg \mathbf{v}_s \cdot \nabla(\phi^* N_\alpha), \quad (2.48)$$

that is

$$\frac{D(\phi^* N_\alpha)}{Dt} \simeq \frac{\partial(\phi^* N_\alpha)}{\partial t}. \quad (2.49)$$

The physical interpretation of this approximation is that the medium is undergoing deformation but remains stationary. Thus the multiphase flow equations coupled with geomechanics, (2.35), can finally be written as

$$\frac{\partial}{\partial t} \left[ N_\alpha(\phi^0 + \alpha \nabla \cdot \mathbf{u} + \frac{1}{M} p) \right] + \nabla \cdot \mathbf{v}_\alpha = q_\alpha, \quad (2.50)$$

with  $\phi^*$  in (2.49) being replaced by (2.45). However, in rock mechanics analysis, it is standard to write the coupled equations in terms of total stress and

pressure as

$$\phi^* \frac{\partial N_\alpha}{\partial t} + N_\alpha [c_b - (1 + \phi_0)c_s] \frac{\partial p}{\partial t} + N_\alpha (c_b - c_s) \frac{\partial \bar{\sigma}}{\partial t} + \nabla \cdot \mathbf{v}_\alpha = q_\alpha. \quad (2.51)$$

### 2.2.2 Coupling Parameters

The above derivation is based on volume variations, including bulk volume, pore volume and porosity variations. As a result, displacements or stresses explicitly show up in the continuity equations. Another term that implicitly involves geomechanical effect is the transmissibility term ( $\nabla \cdot \mathbf{v}_\alpha$ ) through permeability changes. Although rock permeability is generally considered to remain constant in standard reservoir simulations, published laboratory studies for stress-sensitive rocks indicate the dependency of permeability on pore pressure and total stress, see Fatt (1952), McLatchie (1958), Wyble (1958), Gray and Bergamini (1963), Wilhelmi (1967) and Somerton (1967), Vairogs, *et al.* (1971), Thomas and Ward (1972), Jones and Owens (1980), Warpinski and Teufel (1990), Holt (1990), Rhett and Teufel (1992), and Morita *et al.* (1992). Gutierrez & Lewis (1998) and Osorio (1999) pointed out that the reduction of permeability in stress-sensitive reservoirs may have significant effect on the reservoir productivity.

Permeability-stress coupling, however, is not as straight forward as pore volume coupling. It is usually conducted in a staggered manner. One simple approach is to assume that permeability depends on porosity as, for example, in the Garman-Kozeny relation commonly used in basin simulators. Other stress-dependent permeability models may also be used, e.g. the one in Morita

*et al.* 1992.

We point out here that pore volume coupling is the primary interest of this work. Permeability-stress coupling is proposed as future work.

### **2.3 Summary of Coupled Geomechanics and the Reservoir Flow Model**

The mathematical description for coupling porous flow and geomechanics must account for the following important characteristics of the system (Osorio 1999):

1. The multi-component nature of the reservoir rock requires descriptions of both the pore fluid and the solid component. Mass and force conservation laws, and constitutive relations, which represent the coupling effects, are used to obtain the coupled equations.
2. Pressure depletion, which occurs inside the reservoir, induces an extended stress-disturbed region outside the reservoir boundaries. The disturbed region affects the evolution of the stress state at the reservoir boundaries, which, in turn, affects the evolution of the stress state inside the reservoir. The geomechanical interaction between the reservoir and its surroundings is a fully coupled process. This indicates that the inclusion of a surrounding environment (overburden, underburden and sideburden) leads to a realistic modeling of the actual geomechanical boundary conditions. The surrounding domain needs to be extensive

enough to ensure that its boundaries are not perturbed by reservoir production or injection during the time period of interest.

The coupling process is thus characterized by the interaction of essentially different physical phenomena within overlapping material domains. For simplicity we have assumed the two domains to be the same. Generally, solving for displacements takes much more CPU time than solving for pressure and concentrations. Thus, assuming a larger domain for the flow calculation will not induce substantial computational overhead.

### 2.3.1 Black-oil Model

Primary variables chosen for the black-oil model are water phase pressure ( $p_w$ ), oil concentration ( $N_w$ ), and gas concentration ( $N_g$ ). Here, pressure and stresses are taken to be their real values instead of their relative values with respect to initial conditions. Pore pressure is taken to be the wetting phase pressure, i.e., water phase pressure  $p_w$ .

Let  $\Omega$  denote the domain of interest. Let  $\Gamma_{sD}$  and  $\Gamma_{sN}$  be the complementary parts of the boundary  $\partial\Omega$ .  $\mathbf{n}$  denotes the unit outward normal vector on  $\partial\Omega$ . Then the coupled mass balance and force balance equations are written as follows:

**Mass conservation equation:**

$$\frac{\partial}{\partial t} \left[ N_w \left( \phi_0 + \alpha(\nabla \cdot \mathbf{u} - \nabla \cdot \mathbf{u}^0) + \frac{1}{M}(p - p^0) \right) \right] + \nabla \cdot \mathbf{v}_w = q_w, \quad (2.52)$$

$$\frac{\partial}{\partial t} \left[ N_o \left( \phi_0 + \alpha(\nabla \cdot \mathbf{u} - \nabla \cdot \mathbf{u}^0) + \frac{1}{M}(p - p^0) \right) \right] + \nabla \cdot \mathbf{v}_o = q_o, \quad (2.53)$$

$$\frac{\partial}{\partial t} \left[ N_g \left( \phi_0 + \alpha(\nabla \cdot \mathbf{u} - \nabla \cdot \mathbf{u}^0) + \frac{1}{M}(p - p^0) \right) \right] + \nabla \cdot \mathbf{v}_g = q_g, \quad (2.54)$$

$$\mathbf{v}_w = -\frac{\mathbf{k}k_{rw}}{\mu_{fw}B_w}(\nabla p_w - \rho_w g \nabla D), \quad (2.55)$$

$$\mathbf{v}_o = -\frac{\mathbf{k}k_{ro}}{\mu_{fo}B_o}(\nabla p_o - \rho_o g \nabla D), \quad (2.56)$$

$$\mathbf{v}_g = -\frac{\mathbf{k}k_{rg}}{\mu_{fg}B_g}(\nabla p_g - \rho_g g \nabla D) + R_o \mathbf{v}_o, \quad (2.57)$$

$$S_o + S_w + S_g = 1,$$

$$R_o = R_{so} \text{ (three phase),}$$

$$R_o = \frac{N_g}{N_o} \text{ (two phase),}$$

$$p_{cow}(S_w) = p_o - p_w,$$

$$p_{cgo}(S_g) = p_g - p_o.$$

**Initial conditions:**

$$p_w = p_w^0,$$

$$S_w = S_w^0,$$

$$S_g = S_g^0,$$

$$R_o = R_o^0.$$

**Boundary conditions:**

$$\mathbf{v}_w \cdot \mathbf{n} = 0, \quad \text{on } \partial\Omega,$$

$$\mathbf{v}_o \cdot \mathbf{n} = 0, \quad \text{on } \partial\Omega,$$

$$\mathbf{v}_g \cdot \mathbf{n} = 0, \quad \text{on } \partial\Omega.$$

### 2.3.2 Poroelasticity Model

The general poroelastic problem is defined by the following equations subject to the specified initial and boundary conditions. Displacements are chosen to be the primary variables.

**Equilibrium equations:**

$$-\nabla \cdot \boldsymbol{\sigma} = \mathbf{f}, \tag{2.58}$$

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}^0 + \frac{E}{1+\nu} \left( 1 + \frac{\nu}{1-2\nu} \mathbf{m} \mathbf{m}^T \right) \boldsymbol{\varepsilon} - \alpha \mathbf{m} (p - p^0), \tag{2.59}$$

$$\varepsilon_{ij} = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right). \tag{2.60}$$

**Initial conditions:**

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}^0,$$

$$p = p^0.$$

Boundary conditions:

$$\mathbf{u} \cdot \mathbf{n} = 0, \quad \text{on } \Gamma_{sD},$$

$$\boldsymbol{\sigma} \mathbf{n} = \mathbf{g}_s^N, \quad \text{on } \Gamma_{sN}.$$

## Chapter 3

### Finite Element Formulation

In this chapter we present the finite element discretization of the coupled equations. An expanded mixed finite element (MFE) or cell-centered finite difference (CCFD) scheme is used for (2.52)-(2.54), while a conforming Galerkin finite element formulation is used for (2.58). The relationship between MFE and CCFD is demonstrated and analyzed in Russell & Wheeler (1983), Weiser & Wheeler (1988) and Arbogast, Wheeler & Yotov (1997). The expanded MFE formulation for multiphase flow equations is presented in Yotov (1996) and Peszyńska, Wheeler and Yotov (2002).

To use the expanded mixed method that allows for proper handling of the degenerate phase mobility, we define

$$\tilde{\mathbf{v}}_a = -\nabla p_a \quad (a = w, o, g). \quad (3.1)$$

Then the Darcy velocities in (2.55)-(2.57) can be written as

$$\mathbf{v}_w = \lambda_w(\tilde{\mathbf{v}}_w + \rho_w g \nabla D), \quad (3.2)$$

$$\mathbf{v}_o = \lambda_o(\tilde{\mathbf{v}}_o + \rho_o g \nabla D), \quad (3.3)$$

$$\mathbf{v}_g = \lambda_g(\tilde{\mathbf{v}}_g + \rho_g g \nabla D) + R_o \mathbf{v}_o. \quad (3.4)$$

where  $\lambda_a$  ( $a = w, o, g$ ) is the phase mobility,

$$\lambda_a = \frac{\mathbf{k}k_{rw}}{\mu_{fw}B_w}. \quad (3.5)$$

Let  $\mathbf{H}(\text{div}, \Omega) \equiv \left\{ \mathbf{s} \in (L^2(\Omega))^3 : \nabla \cdot \mathbf{s} \in L^2(\Omega) \right\}$  and  $\mathbf{S} \equiv \mathbf{H}(\text{div}, \Omega) \cap \{ \mathbf{s} : \mathbf{s} \cdot \mathbf{n} = 0 \text{ on } \partial\Omega \}$ , which is the subspace of  $\mathbf{H}(\text{div}, \Omega)$  consisting of functions with normal trace on  $\partial\Omega$  equal to zero. Let  $W \equiv L^2(\Omega)$ ,  $\tilde{\mathbf{S}} \equiv (L^2(\Omega))^3$  and  $\mathbf{V} \equiv \left\{ \mathbf{v} \in \mathbf{H}^1(\Omega) : \mathbf{v} = 0 \text{ on } \Gamma_s^D \right\}$ . Then the weak formulation of (2.58), (2.52)-(2.54), (3.1) and (3.2)-(3.4) is to: find  $\mathbf{u} \in \mathbf{V}$ ,  $p_a \in W$ ,  $N_a \in W$ ,  $\tilde{\mathbf{v}}_a \in \tilde{\mathbf{S}}$  and  $\mathbf{v}_a \in \mathbf{S}$  such that,

$$\tilde{a}(\mathbf{u}, \mathbf{v}) = (\mathbf{f}, \mathbf{v}) + (\mathbf{g}_s^N, \mathbf{v}), \quad \forall \mathbf{v} \in \mathbf{V}, \quad (3.6)$$

$$\begin{aligned} & (\alpha N_a \nabla \cdot \dot{\mathbf{u}}, w) + \left( \frac{1}{M} N_a \dot{p}, w \right) + (\phi^* \dot{N}_a, w) \\ & + (\nabla \cdot \mathbf{v}_a, w) = (q_a, w), \quad \forall w \in W, \end{aligned} \quad (3.7)$$

$$(\tilde{\mathbf{v}}_a, \mathbf{s}) = (p_a, \nabla \cdot \mathbf{s}), \quad \forall \mathbf{s} \in \mathbf{S}, \quad (3.8)$$

$$(\mathbf{v}_a, \tilde{\mathbf{s}}) = (\lambda_a \tilde{\mathbf{v}}_a, \tilde{\mathbf{s}}) + (\lambda_a \rho_a g \nabla D, \tilde{\mathbf{s}}) + (c_o R_o \mathbf{v}_o, \tilde{\mathbf{s}}), \quad \forall \tilde{\mathbf{s}} \in \tilde{\mathbf{S}}, \quad (3.9)$$

where  $c_o = 1$  if  $a = g$ , and  $c_o = 0$  otherwise; pore pressure  $p$  and fluid fraction  $\phi^*$  are respectively given by

$$\begin{aligned} p &= s_w p_w + s_o p_o + s_g p_g, \\ \phi^* &= \phi^0 + \alpha (\nabla \cdot \mathbf{u} - \nabla \cdot \mathbf{u}^0) + \frac{1}{M} (p - p^0). \end{aligned}$$

The bilinear function  $\tilde{a}(\mathbf{u}, \mathbf{v})$  in (3.6) has the form of “virtual work”, i.e.,

$$\tilde{a}(\mathbf{u}, \mathbf{v}) = \int_{\Omega} \boldsymbol{\sigma}(\mathbf{u}) : \boldsymbol{\varepsilon}(\mathbf{v}) d\mathbf{x}, \quad (3.10)$$

where  $\boldsymbol{\sigma} : \boldsymbol{\varepsilon} = \sum_{i,j=1}^3 \sigma_{ij} \varepsilon_{ij}$ . It is common in engineering to interpret (3.6) as a statement about the balance of external and internal “virtual work”. Test function  $\mathbf{v}$  is then the “virtual displacement”. Applying (2.59) and (2.60) to (3.6) yields:

$$a(\mathbf{u}, \mathbf{v}) - (\alpha p, \nabla \cdot \mathbf{v}) = l(\mathbf{v}), \quad (3.11)$$

where

$$a(\mathbf{u}, \mathbf{v}) = (\lambda \nabla \cdot \mathbf{u}, \nabla \cdot \mathbf{v}) + (2\mu \boldsymbol{\varepsilon}(\mathbf{u}), \boldsymbol{\varepsilon}(\mathbf{v})), \quad (3.12)$$

$$\begin{aligned} L(\mathbf{v}) &= \int_{\Omega} \mathbf{f} \cdot \mathbf{v} d\mathbf{x} - \int_{\Omega} \boldsymbol{\sigma}^0 : \boldsymbol{\varepsilon}(\mathbf{v}) d\mathbf{x} \\ &+ \int_{\Omega} \alpha p^0 \nabla \cdot \mathbf{v} d\mathbf{x} + \int_{\Gamma_s^N} \mathbf{g}_s^N \cdot \mathbf{v} d\mathbf{x}, \end{aligned} \quad (3.13)$$

Let  $\{\mathcal{T}_h\}_{h>0}$  be a non-degenerate, quasi-uniform finite element partition of  $\Omega$  consisting of rectangular parallelepipeds in  $R^3$ . Let  $\mathbf{V}_h$  be the finite dimensional subspace of  $\mathbf{V}$  consisting of continuous piecewise tri-linear polynomials defined on  $\mathcal{T}_h$ ,

$$\mathbf{V}_h = \left\{ \mathbf{v} = (v_1, v_2, v_3) : v_i|_E = \Pi_{j=1}^3 (\alpha_{ij} + \beta_{ij} x_j) \quad \forall E \in \mathcal{T}_h \right\}, \quad (3.14)$$

where  $\alpha_{ij}, \beta_{ij} \in R$ . Let  $\tilde{\mathbf{S}}_h$  be the subspace of  $\tilde{\mathbf{S}}$ , and let  $(W_h, \mathbf{S}_h)$  indicate the lowest order Raviart-Thomas spaces (Raviart and Thomas 1977) defined on  $\mathcal{T}_h$ ,

$$\tilde{\mathbf{S}}_h = \{ \mathbf{s} = (s_1, s_2, s_3) : s_i|_E = r_{i1} + r_{i2} x_i \quad \forall E \in \mathcal{T}_h \}, \quad (3.15)$$

$$\mathbf{S}_h = \left\{ \mathbf{v} \in \tilde{\mathbf{S}}_h : \mathbf{v} \cdot \mathbf{n} \text{ on } \partial\Omega \right\}, \quad (3.16)$$

$$W_h = \{ w : w|_E = \theta \quad \forall E \in \mathcal{T}_h \}, \quad (3.17)$$

where  $r, \theta \in R$ . The finite element approximation of (3.6)-(3.13) is defined as follows: determine  $\mathbf{u}_h \in \mathbf{V}_h$ ,  $p_{ha} \in W_h$ ,  $N_{ha} \in W_h$ ,  $\tilde{\mathbf{v}}_{ha} \in \tilde{\mathbf{S}}_h$  and  $\mathbf{v}_{ha} \in \mathbf{S}_h$  satisfying

$$a(\mathbf{u}_h, \mathbf{v}) - (\alpha p_h, \nabla \cdot \mathbf{v}) = l(\mathbf{v}), \quad \forall \mathbf{v} \in \mathbf{V}, \quad (3.18)$$

$$\begin{aligned} & (\alpha N_{ha} \nabla \cdot \dot{\mathbf{u}}_h, w) + \left( \frac{1}{M} N_{ha} \dot{p}_h, w \right) + (\phi^* \dot{N}_{ha}, w) \\ & + (\nabla \cdot \mathbf{v}_{ha}, w) = (q_a, w), \quad \forall w \in W, \end{aligned} \quad (3.19)$$

$$(\tilde{\mathbf{v}}_{ha}, \mathbf{s}) = (p_{ha}, \nabla \cdot \mathbf{s}), \quad \forall \mathbf{s} \in \mathbf{S}, \quad (3.20)$$

$$(\mathbf{v}_{ha}, \tilde{\mathbf{s}}) = (\lambda_a \tilde{\mathbf{v}}_{ha}, \tilde{\mathbf{s}}) + (\lambda_a \rho_a g \nabla D, \tilde{\mathbf{s}}) + (c_o R_o \mathbf{v}_{ho}, \tilde{\mathbf{s}}), \quad \forall \tilde{\mathbf{s}} \in \tilde{\mathbf{S}} \quad (3.21)$$

For simplicity, let  $u$ ,  $p_a$ ,  $N_a$ ,  $v_a$  and  $\tilde{v}_a$  denote the nodal values of  $\mathbf{u}_h$ ,  $p_{ha}$ ,  $N_{ha}$ ,  $\mathbf{v}_{ha}$  and  $\tilde{\mathbf{v}}_{ha}$  respectively. The discrete problem (3.18) and (3.19) can be written in a matrix form as

$$\begin{aligned} & \begin{bmatrix} 0 & 0 & 0 \\ M_{wu} & M_{wp} & M_{wn} \\ M_{ou} & M_{op} & M_{on} \\ M_{gu} & M_{gp} & M_{gn} \end{bmatrix} \frac{d}{dt} \begin{bmatrix} u \\ p \\ N \end{bmatrix} \\ & + \begin{bmatrix} E_{uu} & E_{up} & 0 \\ 0 & 0 & T_{wv} \\ 0 & 0 & T_{ov} \\ 0 & 0 & T_{gv} \end{bmatrix} \begin{bmatrix} u \\ p \\ v_f \end{bmatrix} = \begin{bmatrix} F_u \\ F_w \\ F_o \\ F_g \end{bmatrix}. \end{aligned} \quad (3.22)$$

where  $N = (N_w, N_o, N_g)^T$ ,  $p = (p_w, p_o, p_g)^T$ , and  $v_f = (v_w, v_o, v_g)^T$ . Applying certain quadrature rules to the vector integrals in (3.20) and (3.21) (see

Arbogast, Wheeler and Yotov 1997) leads to the following algebraic forms:

$$\begin{aligned}\tilde{A}_{\tilde{v}}\tilde{v}_f &= \begin{bmatrix} \tilde{A}_{\tilde{v}_w\tilde{v}_w} & 0 & 0 \\ 0 & \tilde{A}_{\tilde{v}_o\tilde{v}_o} & 0 \\ 0 & 0 & \tilde{A}_{\tilde{v}_g\tilde{v}_g} \end{bmatrix} \begin{bmatrix} \tilde{v}_w \\ \tilde{v}_o \\ \tilde{v}_g \end{bmatrix} \\ &= \begin{bmatrix} \tilde{A}_{\tilde{v}_w p_w} & 0 & 0 \\ 0 & \tilde{A}_{\tilde{v}_o p_o} & 0 \\ 0 & 0 & \tilde{A}_{\tilde{v}_g p_g} \end{bmatrix} \begin{bmatrix} p_w \\ p_o \\ p_g \end{bmatrix} = \tilde{A}_p p, \end{aligned} \quad (3.23)$$

and

$$\begin{aligned}A_v v_f &= \begin{bmatrix} A_{v_w v_w} & 0 & 0 \\ 0 & A_{v_o v_o} & 0 \\ 0 & 0 & A_{v_g v_g} \end{bmatrix} \begin{bmatrix} v_w \\ v_o \\ v_g \end{bmatrix} \\ &= \begin{bmatrix} A_{v_w \tilde{v}_w} & 0 & 0 \\ 0 & A_{v_o \tilde{v}_o} & 0 \\ 0 & A_{v_g \tilde{v}_g} & A_{v_g \tilde{v}_g} \end{bmatrix} \begin{bmatrix} \tilde{v}_w \\ \tilde{v}_o \\ \tilde{v}_g \end{bmatrix} + \begin{bmatrix} F_{v_w} \\ F_{v_o} \\ F_{v_g} \end{bmatrix} = A_{v\tilde{v}}\tilde{v}_f + F_v, \end{aligned} \quad (3.24)$$

where  $\tilde{A}_{\tilde{v}\tilde{v}}$  and  $A_{vv}$  are diagonal matrices, and  $F_{v_a}$  is resulted from the gravity term in (3.21). Thus,  $v_f$  can be solved from (3.23) and (3.24) by

$$v_f = A_v^{-1} \left[ A_{v\tilde{v}}\tilde{A}_{\tilde{v}}^{-1}\tilde{A}_p p + F_v \right]. \quad (3.25)$$

Substituting  $v_f$  into (3.22) we obtain a simplified system in terms of displacements, pressures and concentrations, i.e,

$$\begin{bmatrix} 0 & 0 & 0 \\ M_{wu} & M_{wp} & M_{wn} \\ M_{ou} & M_{op} & M_{on} \\ M_{gu} & M_{gp} & M_{gn} \end{bmatrix} \frac{d}{dt} \begin{bmatrix} u \\ p \\ N \end{bmatrix} + \begin{bmatrix} E_{uu} & E_{up} \\ 0 & T_{wp} \\ 0 & T_{op} \\ 0 & T_{gp} \end{bmatrix} \begin{bmatrix} u \\ p \end{bmatrix} = \begin{bmatrix} F_u \\ \bar{F}_w \\ \bar{F}_o \\ \bar{F}_g \end{bmatrix}, \quad (3.26)$$

which is equivalent to the system resulted from the CCFD scheme. Eq. (3.26)

can be rewritten in an expanded form as

$$\begin{aligned}
& \begin{bmatrix} 0 & 0 & 0 & 0 \\ M_{wu} & M_{wp_w} & M_{wp_o} & M_{wp_g} \\ M_{ou} & M_{op_w} & M_{op_o} & M_{op_g} \\ M_{gu} & M_{gp_w} & M_{gp_o} & M_{gp_g} \end{bmatrix} \frac{d}{dt} \begin{bmatrix} u \\ p_w \\ p_o \\ p_g \end{bmatrix} \\
& + \begin{bmatrix} 0 & 0 & 0 \\ M_{wn_w} & 0 & 0 \\ 0 & M_{on_o} & 0 \\ 0 & 0 & M_{gn_g} \end{bmatrix} \frac{d}{dt} \begin{bmatrix} N_w \\ N_o \\ N_g \end{bmatrix} \\
& + \begin{bmatrix} E_{uu} & E_{up_w} & E_{up_o} & E_{up_g} \\ 0 & T_{wp_w} & 0 & 0 \\ 0 & 0 & T_{op_o} & 0 \\ 0 & 0 & T_{gp_o} & T_{gp_g} \end{bmatrix} \begin{bmatrix} u \\ p_w \\ p_o \\ p_g \end{bmatrix} = \begin{bmatrix} F_u \\ \bar{F}_w \\ \bar{F}_o \\ \bar{F}_g \end{bmatrix}.
\end{aligned} \tag{3.27}$$

Here,  $M_{au}$ ,  $M_{ap_m}$ ,  $M_{an_a}$ ,  $E_{uu}$  and  $E_{up_m}$  for  $a = (w, o, g)$  and  $m = (w, o, g)$  correspond to the following bilinear forms,

$$\begin{aligned}
\mathcal{M}_{au} &= \int_{\Omega} \alpha N_a \nabla \cdot \mathbf{u}_h w d\mathbf{x}, \quad \forall w \in W_h, \\
\mathcal{M}_{ap_m} &= \int_{\Omega} \frac{1}{M} N_a s_m p_{hm} w d\mathbf{x}, \quad \forall w \in W_h, \\
\mathcal{M}_{an_a} &= \int_{\Omega} \phi^* N_{ha} w d\mathbf{x}, \quad \forall w \in W_h,
\end{aligned}$$

and

$$\begin{aligned}
\mathcal{E}_{uu} &= \int_{\Omega} \lambda \nabla \cdot \mathbf{u}_h \nabla \cdot \mathbf{v} d\mathbf{x} \\
&+ \int_{\Omega} 2\mu \boldsymbol{\varepsilon}(\mathbf{u}_h) : \boldsymbol{\varepsilon}(\mathbf{v}) d\mathbf{x}, \quad \forall \mathbf{v} \in \mathbf{V}_h, \\
\mathcal{E}_{up_m} &= - \int_{\Omega} \alpha s_m p_{hm} \nabla \cdot \mathbf{v} d\mathbf{x}, \quad \forall \mathbf{v} \in \mathbf{V}_h.
\end{aligned}$$

Notice that in (3.23) and (3.24) we have

$$\begin{aligned}\tilde{A}_{\tilde{v}_w \tilde{v}_w} &= \tilde{A}_{\tilde{v}_o \tilde{v}_o} = \tilde{A}_{\tilde{v}_g \tilde{v}_g} = \tilde{A}_{\tilde{v} \tilde{v}}, \\ \tilde{A}_{\tilde{v}_w p_w} &= \tilde{A}_{\tilde{v}_o p_o} = \tilde{A}_{\tilde{v}_g p_g} = \tilde{A}_{\tilde{v} p}, \\ A_{v_w v_w} &= A_{v_o v_o} = A_{v_g v_g} = A_{vv}.\end{aligned}$$

Thus, the transmissibility matrices  $T_{wp_w}$ ,  $T_{wp_o}$ ,  $T_{gp_o}$  and  $T_{gp_g}$  in (3.27) are given by

$$\begin{aligned}T_{wp_w} &= \tilde{A}_{\tilde{v} p}^T A_{vv}^{-1} A_{v_w \tilde{v}_w} \tilde{A}_{\tilde{v} \tilde{v}}^{-1} \tilde{A}_{\tilde{v} p}, \\ T_{op_o} &= \tilde{A}_{\tilde{v} p}^T A_{vv}^{-1} A_{v_o \tilde{v}_o} \tilde{A}_{\tilde{v} \tilde{v}}^{-1} \tilde{A}_{\tilde{v} p}, \\ T_{gp_o} &= \tilde{A}_{\tilde{v} p}^T A_{vv}^{-1} A_{v_g \tilde{v}_o} \tilde{A}_{\tilde{v} \tilde{v}}^{-1} \tilde{A}_{\tilde{v} p}, \\ T_{gp_g} &= \tilde{A}_{\tilde{v} p}^T A_{vv}^{-1} A_{v_g \tilde{v}_g} \tilde{A}_{\tilde{v} \tilde{v}}^{-1} \tilde{A}_{\tilde{v} p}.\end{aligned}$$

Applying backward Euler's method to the time derivatives in (3.26), we obtain a fully implicit system,

$$\begin{aligned}& \begin{bmatrix} 0 & 0 & 0 \\ M_{wu} & M_{wp} & M_{wn} \\ M_{ou} & M_{op} & M_{on} \\ M_{gu} & M_{gp} & M_{gn} \end{bmatrix} \begin{bmatrix} \frac{u^{n+1} - u^n}{\Delta t} \end{bmatrix} \\ & + \begin{bmatrix} E_{uu} & E_{up} \\ 0 & T_{wp} \\ 0 & T_{op} \\ 0 & T_{gp} \end{bmatrix} \begin{bmatrix} u^{n+1} \\ p^{n+1} \end{bmatrix} = \begin{bmatrix} F_u \\ \bar{F}_w \\ \bar{F}_o \\ \bar{F}_g \end{bmatrix}. \quad (3.28)\end{aligned}$$

Multiplying both sides of (3.28) by  $\Delta t$ , and moving  $u^n$ ,  $p^n$  and  $N^n$  to the right

side, we get

$$\begin{bmatrix} E_{uu} & E_{up} & 0 \\ M_{wu} & M_{wp} + \Delta t T_{wp} & M_{wn} \\ M_{ou} & M_{op} + \Delta t T_{op} & M_{on} \\ M_{gu} & M_{gp} + \Delta t T_{gp} & M_{gn} \end{bmatrix} \begin{bmatrix} u^{n+1} \\ p^{n+1} \\ N^{n+1} \end{bmatrix} = \begin{bmatrix} \hat{F}_u \\ \hat{F}_w \\ \hat{F}_o \\ \hat{F}_g \end{bmatrix}. \quad (3.29)$$

Bear in mind that (3.29) is a highly nonlinear system because some of the coefficients depend on the solutions of the primary variables. In the next chapter we will discuss the linearization and iterative solution schemes for the above nonlinear system.

For single phase flow (3.29) can be further simplified to a symmetric positive definite form, i.e.,

$$\begin{bmatrix} E_{uu} & E_{up} \\ E_{up}^T & -M_{pp} - \Delta t T_{pp} \end{bmatrix} \begin{bmatrix} u^{n+1} \\ p^{n+1} \end{bmatrix} = \begin{bmatrix} \hat{F}_u \\ -\hat{F}_p \end{bmatrix}, \quad (3.30)$$

where  $M_{pp}$  is induced by the compressibility of fluid and solid constituents.

## Chapter 4

### Iteratively Coupled Technique

After discretization in space and time, the linear system for coupled flow and geomechanics, (3.29), can be written as a compact  $2 \times 2$  block system, i.e.,

$$\begin{bmatrix} \mathbf{E}_u & \mathbf{E}_p \\ \mathbf{A}_u & \mathbf{A}_p \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \mathbf{p} \end{bmatrix}^{n+1} = \begin{bmatrix} \mathbf{F}_u \\ \mathbf{F}_p \end{bmatrix}, \quad (4.1)$$

where matrix  $\mathbf{E}_u$  and  $\mathbf{A}_p$  are, respectively, the result of Galerkin FE formulation of the elasticity operator and CCFD (MFE) formulation of multiphase flow equations;  $\mathbf{E}_p$  is a discrete negative gradient operator;  $\mathbf{A}_u$  is a discrete divergence operator;  $\mathbf{u}$  is a vector of nodal displacements, and  $\mathbf{p}$  is a vector of cell pressure and concentrations;  $n + 1$  stands for a new time step level.

The main objective of this chapter is to investigate the operator splitting techniques for solving the above discrete system. In particular, we shall formulate in a more general framework the iterative coupling scheme that is frequently adopted by the reservoir simulation community for a coupled analysis. Here we reexamine this method from mathematical perspective with the intent of obtaining preconditioners which yield improved computational efficiencies. Later, we shall show that this iteratively coupled scheme can be easily adapted to a fully coupled scheme. The resulting algorithm retains

the modularity of the iterative method, but it converges faster in nonlinear iterations.

For simplicity, in the following discussion we assume that linear solvers for each physical model (elasticity and porous flow) have already been developed. For a description of the iterative methods implemented in IPARS for the CCFD discretization of multiphase flow equations we refer the reader to the work of Dawson, Klie, Wheeler and Woodward (1997), Klie (1996), Edwards (1998) and Lacroix, Vassilevski and Wheeler (2000, 2001). Linear solvers and preconditioners  $\mathbf{E}_u$  will be discussed in Chapter 6.

## 4.1 Overview of Different Coupling Techniques

In this section we briefly review different coupling methods that have been used extensively in the petroleum industry. We discuss in detail the single phase flow model since a theoretical convergence analysis for this case has been obtained by Phillips and Wheeler (2003). Moreover, the iterative techniques discussed here can be readily generalized to the multiphase flows.

For single phase flow, we rewrite the algebraic system (3.30) as

$$\begin{bmatrix} \mathbf{E}_u & \mathbf{Q}^T \\ \mathbf{Q} & -(\mathbf{M}_{cf} + \Delta t \mathbf{T}_p) \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \mathbf{p} \end{bmatrix}^{n+1} = \begin{bmatrix} \mathbf{F}_u \\ -\mathbf{F}_p \end{bmatrix}. \quad (4.2)$$

In seeking the solution of (4.2) one can use a fully coupled approach to solve simultaneously for displacements and pressures. However, the assembly and factorization of the coefficient matrix may pose major computational requirements for large scale 3D problems. For some field applications that do not

involve strong fluid/structure interaction, a tight coupling may not be economical. Instead, a loosely coupled method may be preferred.

In past decades, operator or time splitting have been proved to be a useful approach for solving large systems of coupled equations. In Biot's poroelastic modeling, the operator splitting technique is used to design a loosely coupled scheme by separating the elasticity operator from the diffusion operator. Then, each field problem can be solved efficiently by available iterative methods. The degree of coupling is generally based on the time scale and the frequency at which it is necessary for the two physical models to exchange information. More specifically, time evolution is controlled by the flow model, and the degree of coupling depends on how often displacement needs to be updated. Clearly, the choice of coupling scheme affects the stability and accuracy of the solutions as well as the computational efficiency. Trade-offs must be made, sometimes, to optimize the computer running time. Settari and Walter (1999) discuss the different coupling methods and categorized them as decoupled, explicitly coupled, iteratively coupled and fully coupled.

### **Fully Coupled**

A fully coupled approach solves two field equations simultaneously. As shown in Phillips and Wheeler (2003), the method is unconditionally stable in time and optimally accurate in the energy norm for displacements and second order accurate for pressure and velocities. However, its practical usefulness is limited by the fact that special linear solvers are required to handle the fully

coupled system. This makes the coupling of two existing complex individual models even more complicated, especially if one adds in the thermal effects.

### Decoupled

In this approach the flow equations are completely decoupled from the poroelastic model by assuming that  $d\boldsymbol{\sigma} = 0$ . Porosity changes are approximated only by pore pressure changes via a rock compressibility term, i.e.,

$$\phi = \phi^o [1 + c_r (p - p^0)] .$$

The pressure equation is written as

$$(\mathbf{M}_{cr} + \mathbf{M}_{cf} + \Delta t \mathbf{T}_p) \mathbf{p} = \mathbf{F}_p, \quad (4.3)$$

where  $\mathbf{M}_{cr}$  is a pressure mass matrix induced by the rock compressibility  $c_r$ . Once the pressure has been solved, displacement may be obtained whenever necessary by

$$\mathbf{E}_u \mathbf{u} = \mathbf{F}_u - \mathbf{Q}^T \mathbf{p},$$

where the pressure is imposed as an external load.

### Explicitly coupled

Explicit coupling is essentially a staggered partitioning method. It is achieved by lagging the coupling terms in one or more time steps. Generally, quasi-static behavior is assumed for the geomechanics model. Different time steps may be used for displacement and flow calculations respectively. For

instance, several flow time steps may be taken for a given displacement time step. The geomechanical updates are driven by the magnitude of porosity change during the time steps (Dean et al. 2003).

In the case that the same time scale is used for flow and displacement the explicit coupling technique involves the solution of the following decoupled system, as follows:

$$\begin{aligned} (\mathbf{M}_{cr} + \mathbf{M}_{cf} + \Delta t \mathbf{T}) \mathbf{p}^{n+1} &= \mathbf{F}_p, \\ \mathbf{E} \mathbf{u}^{n+1} &= \mathbf{F}_u - \mathbf{Q}^T \mathbf{p}^{n+1}. \end{aligned} \tag{4.4}$$

A major drawback of the explicit method is that it is only conditionally stable, as shown in Booker and Small (1975). Stability considerations may lead to time step restrictions.

### **Iteratively Coupled**

The iterative method is a tightly coupled scheme designed primarily for nonlinear problems. Advantages of this method include:

1. Stability and accuracy. It can produce the same results as a fully coupled technique if a sufficiently tight nonlinear convergence criteria is enforced.
2. Modularity feature. It allows the coupled equations to be processed by separate program modules, taking full advantage of specialized features and disciplinary expertise built into independently developed single-field models.

In an iteratively coupled scheme, solutions for multiphase flow and poroelasticity equations are coupled through the nonlinear iterations in one time step. If  $n$  denotes the time step and  $k$  denotes the nonlinear iteration, an iterative method involves the repeated solution of the following system in each time step  $n + 1$ ,

$$\begin{aligned} (\mathbf{M}_{cf} + \Delta t \mathbf{T}) \mathbf{p}^{n+1,k+1} &= \mathbf{F}_p + \mathbf{Q} \mathbf{u}^{n+1,k}, \\ \mathbf{E} \mathbf{u}^{n+1,k+1} &= \mathbf{F}_u - \mathbf{Q}^T \mathbf{p}^{n+1,k+1}. \end{aligned} \tag{4.5}$$

Details about this method will be discussed in the next section.

## 4.2 Iteratively Coupled Technique

The iterative coupling defined in (4.5) is basically a nonlinear variation of the block Gauss-Seidel method with Picard linearization. In this section, we present an alternative scheme that yields faster convergence. It will be shown later that the method may be viewed as a special case of a preconditioned Richardson method applied to the Schur complement of the pressure equations. Throughout this chapter we denote the time step level by  $n$ , the nonlinear iteration number by  $k$ , and the linear iteration number in each nonlinear iteration by  $l$ .

### 4.2.1 Linearization of Nonlinear Systems

In the case of nonlinear systems (multiphase flow), we modify our iterative strategy, (4.5), by applying a predictor-multicorrector scheme at each

time step. A series of corrected solutions are computed after starting with an initial approximation that is either the solution at the last time step or an extrapolated solution from previous time steps. For this purpose, Newton's method is used here to linearize the system (4.2). Residuals at the  $k$ th nonlinear iteration are computed as:

$$\begin{aligned}\mathbf{R}_u^{n+1,k} &= \mathbf{F}_u - \mathbf{E}_u \mathbf{u}^{n+1,k} - \mathbf{Q}^T \mathbf{p}^{n+1,k}, \\ \mathbf{R}_p^{n+1,k} &= -\mathbf{F}_p - \mathbf{Q} \mathbf{u}^{n+1,k} + (\mathbf{M}_{cf} + \Delta t \mathbf{T}_p) \mathbf{p}^{n+1,k}.\end{aligned}$$

Applying Newton's linearization to the above residual equations yields the following algebraic system:

$$\begin{bmatrix} -\frac{\partial \mathbf{R}_u}{\partial \mathbf{u}} & -\frac{\partial \mathbf{R}_u}{\partial \mathbf{p}} \\ -\frac{\partial \mathbf{R}_p}{\partial \mathbf{u}} & -\frac{\partial \mathbf{R}_p}{\partial \mathbf{p}} \end{bmatrix} \begin{bmatrix} \delta \mathbf{u} \\ \delta \mathbf{p} \end{bmatrix}^{n+1,k+1} = \begin{bmatrix} \mathbf{R}_u \\ \mathbf{R}_p \end{bmatrix}^{n+1,k}. \quad (4.6)$$

The final linearized system is given by:

$$\begin{bmatrix} \mathbf{E}_u & \mathbf{Q}^T \\ \mathbf{Q} & -(\bar{\mathbf{M}}_{cf} + \Delta t \bar{\mathbf{T}}_p) \end{bmatrix} \begin{bmatrix} \delta \mathbf{u} \\ \delta \mathbf{p} \end{bmatrix}^{n+1,k+1} = \begin{bmatrix} \mathbf{R}_u \\ -\mathbf{R}_p \end{bmatrix}^{n+1,k}. \quad (4.7)$$

Newton iterations are performed on (4.7) until a given tolerance for the residuals is satisfied. Solutions at the end of each iteration are corrected by:

$$\begin{aligned}\mathbf{u}^{n+1,k+1} &= \mathbf{u}^{n+1,k} + \delta \mathbf{u}^{n+1,k+1}, \\ \mathbf{p}^{n+1,k+1} &= \mathbf{p}^{n+1,k} + \delta \mathbf{p}^{n+1,k+1}.\end{aligned}$$

#### 4.2.2 Iterative Coupling by Operator Splitting

An iterative coupling scheme is based on the classical operator splitting technique. Namely, the two field equations in (4.7) are solved sequentially at

each Newton iteration by

$$(\bar{\mathbf{M}}_{cf} + \Delta t \bar{\mathbf{T}}_p) \delta \mathbf{p}^{n+1,k+1} = \mathbf{R}_p^{n+1,k} + \mathbf{Q} \delta \mathbf{u}^{n+1,k} \quad (4.8)$$

$$\mathbf{E}_u \delta \mathbf{u}^{n+1,k+1} = \mathbf{R}_u^{n+1,k} - \mathbf{Q}^T \delta \mathbf{p}^{n+1,k+1} \quad (4.9)$$

They are then coupled through the calculation of porosity at the end of the iteration.

The iterative scheme defined in (4.8) and (4.9) is basically one block Gauss-Seidel iteration for (4.7). The method adopted by reservoir engineers (Settari & Mourits 1994 and Settari 1999) is based on an augmented form of (4.8), i.e.,

$$(\mathbf{M}_{cr} + \bar{\mathbf{M}}_{cf} + \Delta t \bar{\mathbf{T}}_p) \delta \mathbf{p}^{n+1,k+1} = \mathbf{R}_p^{n+1,k}. \quad (4.10)$$

Note that the lagged term  $\mathbf{Q} \delta \mathbf{u}^{n+1,k}$  in (4.8) is replaced by  $\mathbf{M}_{cr} \delta \mathbf{p}^{n+1,k+1}$ . Convergence is thus improved.

### 4.2.3 Physics-based Preconditioners to $\mathbf{Q} \delta \mathbf{u}$

The idea comes naturally from the way porosity is approximated in conventional reservoir simulations, i.e.,

$$\phi = \phi^o [1 + c_r (p - p^0)] , \quad (4.11)$$

where the rock compressibility  $c_r$  is generally obtained from a laboratory uniaxial strain test. In a coupled simulation, a term similar to  $\phi$  appearing in the mass balance equation (2.35) is

$$\phi^* = \phi^0 + \alpha (\nabla \cdot \mathbf{u} - \nabla \cdot \mathbf{u}^0) + \frac{1}{M} (p - p^0) . \quad (4.12)$$

The underlying idea is to approximate (4.12) in a form similar to (4.11). Thus, a standard reservoir simulator may be coupled with geomechanics with minimum changes and an improved convergence.

$\nabla \cdot \mathbf{u}$  in (4.12) can be expressed in terms of pore pressure and mean stress as

$$\nabla \cdot \delta \mathbf{u} = \frac{3\alpha}{3\lambda + 2\mu} \delta p + \frac{3}{3\lambda + 2\mu} \delta \bar{\sigma}.$$

Accordingly, the porosity change satisfies

$$\delta \phi^* = \left( \frac{3\alpha^2}{3\lambda + 2\mu} + \frac{1}{M} \right) \delta p + \frac{3\alpha}{3\lambda + 2\mu} \delta \bar{\sigma}.$$

If  $\delta \bar{\sigma}$  during a nonlinear iteration is assumed to be small and thus ignored, we obtain the following approximations:

$$\nabla \cdot \delta \mathbf{u} \approx \frac{3\alpha}{3\lambda + 2\mu} \delta p, \quad (4.13)$$

$$\delta \phi^* \approx \left( \frac{3\alpha^2}{3\lambda + 2\mu} + \frac{1}{M} \right) \delta p. \quad (4.14)$$

The Galerkin FE approximation to (4.13) yields

$$\mathbf{Q} \delta \mathbf{u} = -\mathbf{M}_{cr} \delta \mathbf{p}. \quad (4.15)$$

The pressure equation in (4.7) is then decoupled from the displacement by substituting (4.15) into (4.8) for  $\mathbf{Q} \delta \mathbf{u}$  as shown in (4.10).  $\mathbf{M}_{cr}$  is equivalent to a pressure mass matrix corresponding to

$$\mathcal{M}_{cr} = \int_{\Omega} \frac{3\alpha^2}{3\lambda + 2\mu} p_h w d\Omega \quad \forall w \in W_h. \quad (4.16)$$

Once  $\delta \mathbf{p}$  in (4.10) is solved,  $\delta \mathbf{u}$  is then obtained using (4.9). The current Newton iteration is terminated by the update of  $\phi^*$  using (4.12). The iterative coupling scheme described above is summarized as follows:

1. *Start a new time step  $n+1$ .*
2. *Initialize  $\mathbf{p}^{n+1,0} = \mathbf{p}^n$  and  $\mathbf{u}^{n+1,0} = \mathbf{u}^n$ , or  $\mathbf{p}^{n+1,0} = \mathbf{p}^e$  and  $\mathbf{u}^{n+1,0} = \mathbf{u}^e$  where  $\mathbf{p}^e$  and  $\mathbf{u}^e$  are the extrapolated values from previous time steps.*
3. *Start a new nonlinear iteration  $k+1$ .*
4. *Compute residuals in (4.6) using  $\mathbf{p}^{n+1,k}$ ,  $\mathbf{u}^{n+1,k}$  and  $\phi^{*n+1,k}$  from the last Newton iteration and check for convergence by*

$$\|\mathbf{R}_p\|_{L^\infty} < \text{Tol}$$

*and*

$$\|\phi^{*n+1} - \phi_f^{*n+1}\|_{L^\infty} < \text{Tol},$$

*where  $\phi_f^{*n+1}$  is computed by (4.14). If both tolerances are satisfied, terminate the current time step and go to step 1 for a new time level. Otherwise, continue with the following steps.*

5. *Update Jacobian matrix.*
6. *Solve for  $\delta \mathbf{p}^{k+1}$  using (4.10) and update solutions by  $\mathbf{p}^{n+1} = \mathbf{p}^{n+1,k} + \delta \mathbf{p}^{k+1}$ .*
7. *Compute  $\phi^{*n+1}$  using (4.14) and denote it by  $\phi_f^{*n+1}$ .*

8. *Solve for  $\delta \mathbf{u}^{k+1}$  using (4.9). Update displacement by  $\mathbf{u}^{n+1} = \mathbf{u}^{n+1,k} + \delta \mathbf{u}^{k+1}$ .*
9. *Compute  $\phi^{*n+1}$  with (4.12).*
10. *Go to 3.*

Since (4.13) is derived directly from the constitutive equations, the decoupling scheme above is applicable to general boundary conditions. An ideal case is the one with unconfined lateral boundaries where the total stress change is relatively small. However, in most subsidence and reservoir compaction applications, rock deformation is induced by pressure depletion in the reservoir. As fluid is being pumped out of the reservoir, support for overburden load is gradually transferred from pore fluid to solid matrix. If the reservoir is confined horizontally, the deformation under the act of overburden will be in the vertical direction primarily. This observation suggests that a better approximation than (4.16) may be derived for these types of boundary conditions.

If we assume that lateral displacements are small compared to the reservoir horizontal extensions, and hence lateral strains are close to zero, i.e.,

$$\frac{\partial u_x}{\partial x} + \frac{\partial u_y}{\partial y} \ll \frac{\partial u_z}{\partial z},$$

then the 3D poroelastic model can be approximated by an uniaxial strain

deformation. Thus, we have

$$\begin{aligned}\nabla \cdot \delta \mathbf{u} &\approx \varepsilon_{zz} = \frac{1}{\lambda + 2\mu}(\delta\sigma_{zz} + \alpha\delta p), \\ \delta\phi^* &\approx \left(\frac{\alpha^2}{\lambda + 2\mu}\right)\delta p + \frac{1}{\lambda + 2\mu}\delta\sigma_{zz}.\end{aligned}$$

Assuming  $\delta\sigma_{zz}$  is small enough to be neglected we obtain another approximation to  $\mathbf{Q}\delta\mathbf{u}$ ,

$$\mathcal{M}_{cr} = \int_{\Omega} \frac{\alpha^2}{\lambda + 2\mu} p_h w d\Omega \quad \forall w \in W_h. \quad (4.17)$$

### 4.3 Iterative Coupling as One Iteration of a Preconditioned Richardson Method

In this section we shall reformulate the iterative coupling scheme in a more general framework where the method appears to be one iteration of a preconditioned Richardson method for a fully coupled system. The new formulation casts a new perspective on the potentials of the iterative coupling method. It leads us naturally to the investigation of new preconditioners and provides a fundamental basis for further theoretical convergence analysis. We first consider the fully coupled system (4.7) for time level  $n + 1$  and Newton iteration  $k + 1$  rewritten here as,

$$\mathbf{A}\delta\mathbf{x} = \begin{bmatrix} \mathbf{E}_u & \mathbf{Q}^T \\ \mathbf{Q} & -(\bar{\mathbf{M}}_{cf} + \Delta t \bar{\mathbf{T}}_p) \end{bmatrix} \begin{bmatrix} \delta\mathbf{u} \\ \delta\mathbf{p} \end{bmatrix} = \begin{bmatrix} \mathbf{R}_u \\ -\mathbf{R}_p \end{bmatrix}. \quad (4.18)$$

Applying block Gaussian Elimination yields:

$$\begin{bmatrix} \mathbf{E}_u & \mathbf{Q}^T \\ \mathbf{0} & (\bar{\mathbf{M}}_{cf} + \Delta t \bar{\mathbf{T}}_p) + \mathbf{Q}\mathbf{E}_u^{-1}\mathbf{Q}^T \end{bmatrix} \begin{bmatrix} \delta\mathbf{u} \\ \delta\mathbf{p} \end{bmatrix} = \begin{bmatrix} \mathbf{R}_u \\ \mathbf{R}_p + \mathbf{Q}\mathbf{E}_u^{-1}\mathbf{R}_u \end{bmatrix}, \quad (4.19)$$

where the second diagonal block,

$$\mathbf{S} = (\bar{\mathbf{M}}_{cf} + \Delta t \bar{\mathbf{T}}_p) + \mathbf{Q}\mathbf{E}_u^{-1}\mathbf{Q}^T \quad (4.20)$$

is denoted as a Schur complement to  $\mathbf{E}_u$ . In general, the solution of (4.18) is sought in two ways:

1. **Nested iterative solutions.** Linear solutions involve the repeated solution of

$$\mathbf{S}\delta\mathbf{p}^{n+1,k+1} = \mathbf{R}_p^{n+1,k} + \mathbf{Q}\mathbf{E}_u^{-1}\mathbf{R}_u^{n+1,k} \quad (4.21)$$

and back substitution of

$$\delta\mathbf{u}^{n+1,k+1} = \mathbf{E}_u^{-1}(\mathbf{R}_u^{n+1,k} - \mathbf{Q}^T\delta\mathbf{p}^{n+1,k+1}). \quad (4.22)$$

If an iterative scheme is used to solve the two subsystems, then we have a two-level solver with inner and outer iterations.

2. **Non-nested iterative solutions.** A Krylov subspace approach is used (4.18) with simple preconditioners such as a block diagonal preconditioner,

$$\mathbf{P}_D = \begin{bmatrix} \tilde{\mathbf{E}}_u & \\ & \tilde{\mathbf{S}} \end{bmatrix},$$

where  $\tilde{\mathbf{E}}_u$  and  $\tilde{\mathbf{S}}$  are, respectively, approximations to  $\mathbf{E}_u$  and  $\mathbf{S}$ . Since both  $\tilde{\mathbf{E}}_u$  and  $\tilde{\mathbf{S}}$  are chosen to be readily invertible, no inner iteration is required. Phoon et. al. (2003) suggest an ideal, though academic, block diagonal preconditioner for Biot's system, i.e.,

$$\mathbf{P}_D = \begin{bmatrix} \mathbf{E}_u & \mathbf{0} \\ \mathbf{0} & \beta\mathbf{S} \end{bmatrix}, \quad (4.23)$$

where  $\beta$  is a non-zero scalar. They demonstrated that  $\mathbf{P}_D^{-1}\mathbf{A}$  has three distinct clusters of eigenvalues, each with a diameter of order  $\|\mathbf{S}^{-1}(\bar{\mathbf{M}}_{cf} + \Delta\bar{\mathbf{T}}_p)\|$ .

Clearly, both approaches require a good approximation to  $\mathbf{S}$ . In the following we shall concentrate on the first type of solution method since it represents a two-level iterative scheme.

For simplicity we first assume that an efficient and optimal approximation to  $\mathbf{S}$  has been established. Next, we use  $\tilde{\mathbf{S}}$  to construct iterative schemes for solving the coupled system (4.18). Despite its slow convergence rate, the preconditioned Richardson method is chosen here to demonstrate that an iterative coupling may be formulated in a general framework of a fully coupled scheme.

Given a linear system,

$$\mathbf{Ax} = \mathbf{b},$$

the preconditioned Richardson iteration reads

$$\mathbf{x}^{l+1} = \mathbf{x}^l + \tau\tilde{\mathbf{A}}^{-1}(\mathbf{b} - \mathbf{Ax}^l),$$

where  $\tilde{\mathbf{A}}^{-1}$  is a preconditioner,  $l$  is the linear iteration number, the scalar  $\tau$  is a damping factor, and  $\mathbf{x}^0$  is an initial guess. Applying this approach to (4.21) leads to the recursion of

$$\delta\mathbf{p}^{n+1,k+1,l+1} = \delta\mathbf{p}^{n+1,k+1,l} + \tau\tilde{\mathbf{S}}^{-1}(\mathbf{R}_p + \mathbf{Q}\mathbf{E}_u^{-1}\mathbf{R}_u - \mathbf{S}\delta\mathbf{p}^{n+1,k+1,l}). \quad (4.24)$$

In the following we show that the iterative coupling defined in (4.10) and (4.9) can be generalized to a form equivalent to (4.24). For convenience, we rewrite the iterative scheme for one Newton step as

$$\delta \mathbf{p}^{n+1,k+1} = (\mathbf{M}_{cr} + \bar{\mathbf{M}}_{cf} + \Delta t \bar{\mathbf{T}}_p)^{-1} \mathbf{R}_p, \quad (4.25)$$

$$\delta \mathbf{u}^{n+1,k+1} = \mathbf{E}_u^{-1} (\mathbf{R}_u - \mathbf{Q}^T \delta \mathbf{p}^{n+1,k+1}). \quad (4.26)$$

If  $\mathbf{M}_{cr} + \bar{\mathbf{M}}_{cf} + \Delta t \bar{\mathbf{T}}_p$  in (4.25) is viewed as an approximation to  $\mathbf{S}$ , i.e.,

$$\tilde{\mathbf{S}} = \mathbf{M}_{cr} + \bar{\mathbf{M}}_{cf} + \Delta t \bar{\mathbf{T}}_p, \quad (4.27)$$

then (4.25) can certainly be generalized to multiple iterations for solving  $\delta \mathbf{p}^{n+1,k+1}$ , as shown below.

$$\begin{aligned} \delta \mathbf{p}^{n+1,k+1,l+1} = & \delta \mathbf{p}^{n+1,k+1,l} + \tilde{\mathbf{S}}^{-1} [\mathbf{R}_p + \mathbf{Q} \delta \mathbf{u}^{n+1,k+1,l} \\ & - (\bar{\mathbf{M}}_{cf} + \Delta t \bar{\mathbf{T}}_p) \delta \mathbf{p}^{n+1,k+1,l}]. \end{aligned} \quad (4.28)$$

Substituting (4.26) into (4.28) for  $\delta \mathbf{u}^{n+1,k+1,l}$  immediately yields

$$\delta \mathbf{p}^{n+1,k+1,l+1} = \delta \mathbf{p}^{n+1,k+1,l} + \tilde{\mathbf{S}}^{-1} [\mathbf{R}_p + \mathbf{Q} \mathbf{E}_u^{-1} \mathbf{R}_u - \mathbf{S} \delta \mathbf{p}^{n+1,k+1,l}]. \quad (4.29)$$

Clearly, this iteration scheme represents the preconditioned Richardson method, (4.24), with  $\tau = 1$ . This indicates that the iterative coupling approach can be viewed as the first iteration of a preconditioned Richardson method with zero initial guess ( $\delta \mathbf{p}^{n+1,k+1,0} = \delta \mathbf{u}^{n+1,k+1,0} = 0$ ). If additional iterations are performed, using (4.29) and (4.26), until a given tolerance is satisfied, we have a converged iterative method for solving a fully coupled system. The building blocks needed here are fast solvers for  $\mathbf{E}_u^{-1}$  and  $\tilde{\mathbf{S}}$ . But usually these are

readily available as special software modules. Attractive features of the preconditioned Richardson method described above lie in its accuracy, its stability and software modularity.

Efficiency is another important feature of the iteratively coupled approach. Frequently, the choice of the scheme is based on its ease of implementation and software modularity with the acceptance of slow convergence. However, a close examination of the method shows that this may not always be true. Both the nested and non-nested methods require the inversion, or approximate inversion, of  $\mathbf{E}_u$ . Numerical experiments demonstrate that such an inversion is not trivial. It generally takes more CPU time than solving the pressure block (4.21). The strategy of the iteratively coupled approach is to minimize the steps for solving the elasticity block, say, once per Newton iteration. Poor accuracy in the solution of a linear system (4.18) will invariably cost more Newton iterations to correct. But the computation overhead incurred by the solution of (4.21), as well as the update of the Jacobian matrix (4.6), is well compensated by the reduced CPU time for solving (4.22). Thus, the overall efficiency of an iterative coupling method may be even better than that of a fully coupled scheme, as shown by Dean (2003).

#### 4.3.1 Preconditioners for $\mathbf{S}$

If the iteratively coupled technique is understood as one iteration of a preconditioned Richardson method for the Schur complement of the pressure equation, its convergence hinges decisively on the preconditioning techniques

for  $\mathbf{S}$  and  $\mathbf{E}_u$ . Fast and efficient preconditioners for  $\mathbf{E}_u$  will be discussed in Chapter 6. In the following we shall investigate the preconditioners for  $\mathbf{S}$ . First we present two types of optimal preconditioners by showing their spectral equivalency to  $\mathbf{S}$ . Later, we give several specific forms of these preconditioners. Numerical experiments and comparison results will be shown in Section 4.4.

#### 4.3.1.1 Optimal Preconditioners for $\mathbf{S}$

Here we utilize an inequality established by Wathen & Silvester (1993) to show the optimality of the preconditioners given in (4.16) and (4.17). For MFE approximations to the classical Stokes problems Wathen & Silvester demonstrate that

$$0 \leq \frac{(\mathbf{Q}\mathbf{B}_u^{-1}\mathbf{Q}^T\mathbf{p}, \mathbf{p})}{(\mathbf{M}_p\mathbf{p}, \mathbf{p})} \leq \Gamma_2 \quad \forall \mathbf{p} \in \mathfrak{R}^{n_p}. \quad (4.30)$$

where  $\Gamma_2$  is positive constant independent of  $h$ ;  $\mathbf{M}_p$  is the symmetric positive definite pressure mass matrix with a condition number independent of  $h$ ;  $n_p$  is the total number of discrete pressure variables;  $\mathbf{B}_u$  is the discrete Laplacian operator for  $\mathbf{u}$  corresponding to the following bilinear form,

$$b(\mathbf{u}, \mathbf{v}) = \int_{\Omega} \nabla \mathbf{u} : \nabla \mathbf{v} d\Omega \quad \forall \mathbf{u}, \mathbf{v} \in \mathbf{V}. \quad (4.31)$$

For convenience, in the bilinear form  $a(\mathbf{u}, \mathbf{v})$  in (3.12) we assume  $\mu = 1$ . Then  $\mathbf{E}_u$  is related to  $\mathbf{B}_u$  by

$$\mathbf{E}_u = \mathbf{C}_u + \mathbf{B}_u. \quad (4.32)$$

Next, we demonstrate that (4.30) still holds with  $\mathbf{B}_u$  replaced by  $\mathbf{E}_u$ .

Using Korn's inequality, Ciarlet (1988) proves the  $\mathbf{H}^1(\Omega)$  ellipticity of the bilinear form  $a(\mathbf{u}, \mathbf{v})$ . In other words, the “a” norm in  $\mathbf{V}$  is equivalent to the norm  $\|\cdot\|_{1,\Omega}$ , more precisely,

$$c_1 \|\mathbf{v}\|_{1,\Omega}^2 \leq a(\mathbf{v}, \mathbf{v}) \leq c_2 \|\mathbf{v}\|_{1,\Omega}^2 \quad \forall \mathbf{v} \in \mathbf{V}, \quad (4.33)$$

where  $c_1$  and  $c_2$  are two positive constants;  $\|\mathbf{v}\|_{1,\Omega}$  is defined for 3D elasticity as

$$\|\mathbf{v}\|_{1,\Omega} = \left( \sum_{i=1}^3 \|v_i\|_{1,\Omega}^2 \right)^{1/2}.$$

Since the bilinear form  $b(\mathbf{u}, \mathbf{v})$  in (4.31) also generates a norm in  $\mathbf{V}$  equivalent to the  $\mathbf{H}^1$ -norm, it is clear that the matrices  $\mathbf{E}_u$  and  $\mathbf{B}_u$  are spectrally equivalent, i.e.,

$$\gamma_1 \leq \frac{(\mathbf{B}_u, \mathbf{v}, \mathbf{v})}{(\mathbf{E}_u, \mathbf{v}, \mathbf{v})} \leq \gamma_2 \quad \forall \mathbf{v} \in \mathfrak{R}^{n_u}, \quad (4.34)$$

where  $\gamma_1, \gamma_2 > 0$  and they are independent of  $h$ .

The significance of (4.34) is twofold. First, if  $\mathbf{B}_u$  is used as a preconditioner for  $\mathbf{E}_u$ , the eigenvalue spectrum of  $\mathbf{B}_u^{-1}\mathbf{E}_u$  is highly clustered and bounded independent of  $h$ . In Chapter 6 we describe preconditioners based on  $\mathbf{B}_u$ . Secondly, (4.34) together with (4.30) leads to an optimal preconditioner for the Schur complement  $\mathbf{S}$  as shown below.

Using Lemma 3.2 in Bramble (1993), for SPD matrices  $\mathbf{E}_u$  and  $\mathbf{B}_u$ , (4.34) holds if and only if

$$\gamma_1 \leq \frac{(\mathbf{E}_u^{-1}, \mathbf{v}, \mathbf{v})}{(\mathbf{B}_u^{-1}, \mathbf{v}, \mathbf{v})} \leq \gamma_2 \quad \forall \mathbf{v} \in \mathfrak{R}^{n_u}. \quad (4.35)$$

Note that  $r_1$  and  $r_2$  are the same constants in both inequalities. Since  $\mathbf{v}$  is arbitrary, we choose  $\mathbf{v} = \mathbf{Q}^T \mathbf{p}$ . Then, (4.35) leads to

$$\gamma_1 \leq \frac{(\mathbf{Q}\mathbf{E}_u^{-1}\mathbf{Q}^T \mathbf{p}, \mathbf{p})}{(\mathbf{Q}\mathbf{B}_u^{-1}\mathbf{Q}^T \mathbf{p}, \mathbf{p})} \leq \gamma_2 \quad \forall \mathbf{p} \in \mathfrak{R}^{n_p}. \quad (4.36)$$

From (4.30) and (4.36) we immediately obtain

$$0 \leq \frac{(\mathbf{Q}\mathbf{E}_u^{-1}\mathbf{Q}^T \mathbf{p}, \mathbf{p})}{(\mathbf{M}_p \mathbf{p}, \mathbf{p})} \leq \gamma_2 \Gamma_2 \quad \forall \mathbf{p} \in \mathfrak{R}^{n_p}. \quad (4.37)$$

Now we are ready to prove the following spectral equivalency, i.e., given the inequalities in (4.37) and a positive definite matrix  $\bar{\mathbf{M}}_{cf} + \Delta t \bar{T}_p$  satisfying

$$((\bar{\mathbf{M}}_{cf} + \Delta t \bar{T}_p) \mathbf{p}, \mathbf{p}) \geq \beta (\mathbf{M}_p \mathbf{p}, \mathbf{p}) \quad \forall \mathbf{p} \in \mathfrak{R}^{n_p}, \quad (4.38)$$

then there exist two positive constants  $\theta$  and  $\Theta$ , independent of mesh parameter  $h$ , such that

$$\theta(\tilde{\mathbf{S}} \mathbf{p}, \mathbf{p}) \leq (\mathbf{S} \mathbf{p}, \mathbf{p}) \leq \Theta(\tilde{\mathbf{S}} \mathbf{p}, \mathbf{p}) \quad \forall \mathbf{p} \in \mathfrak{R}^{n_p}, \quad (4.39)$$

where  $\mathbf{S}$  and  $\tilde{\mathbf{S}}$  are defined to be

$$\tilde{\mathbf{S}} = \mathbf{M}_p + \bar{\mathbf{M}}_{cf} + \Delta t \bar{T}_p. \quad (4.40)$$

$$\mathbf{S} = \mathbf{Q}\mathbf{E}^{-1}\mathbf{Q}^T + \bar{\mathbf{M}}_{cf} + \Delta t \bar{T}_p \quad (4.41)$$

**Proof.** From the upper bound in (4.37) we have

$$(\mathbf{S} \mathbf{p}, \mathbf{p}) \leq ((\gamma_2 \Gamma_2 \mathbf{M}_p + \bar{\mathbf{M}}_{cf} + \Delta t \bar{T}_p), \mathbf{p}, \mathbf{p}) \quad \forall \mathbf{p} \in \mathfrak{R}^{n_p}. \quad (4.42)$$

Choosing  $\Theta = \max(\gamma_2 \Gamma_2, 1)$  we obtain

$$(\mathbf{S} \mathbf{p}, \mathbf{p}) \leq \Theta(\tilde{\mathbf{S}} \mathbf{p}, \mathbf{p}) \quad \forall \mathbf{p} \in \mathfrak{R}^{n_p}. \quad (4.43)$$

To prove the lower bound in (4.39), we note that  $\bar{\mathbf{M}}_{cf} + \Delta t \bar{\mathbf{T}} \geq \beta \mathbf{M}_p$  from (4.38). Thus, it is obvious that

$$\bar{\mathbf{M}}_{cf} + \Delta t \bar{\mathbf{T}} + \mathbf{M}_p \leq (1 + \frac{1}{\beta})(\bar{\mathbf{M}}_{cf} + \Delta t \bar{\mathbf{T}}). \quad (4.44)$$

By choosing  $\theta = \frac{1}{1+1/\beta}$  we get

$$\theta(\bar{\mathbf{M}}_{cf} + \Delta t \bar{\mathbf{T}} + \mathbf{M}_p) \leq \bar{\mathbf{M}}_{cf} + \Delta t \bar{\mathbf{T}}. \quad (4.45)$$

Since  $\mathbf{Q}\mathbf{E}_u^{-1}\mathbf{Q}^T \geq 0$  from (4.37) we immediately have

$$\theta(\bar{\mathbf{M}}_{cf} + \Delta t \bar{\mathbf{T}} + \mathbf{M}_p) \leq \bar{\mathbf{M}}_{cf} + \Delta t \bar{\mathbf{T}} + \mathbf{Q}\mathbf{E}_u^{-1}\mathbf{Q}^T, \quad (4.46)$$

or simply,

$$\theta(\tilde{\mathbf{S}}\mathbf{p}, \mathbf{p}) \leq (\mathbf{S}\mathbf{p}, \mathbf{p}) \quad \forall \mathbf{p} \in \mathfrak{R}^{n_p}. \quad (4.47)$$

Therefore, optimal preconditioners for  $\tilde{\mathbf{S}}$  will also give rise to optimal preconditioners for  $\mathbf{S}$  in the sense that the condition number  $\kappa(\tilde{\mathbf{S}}^{-1}\mathbf{S}) = O(1)$  uniformly in  $h$ . For convenience, we rewrite the decoupled matrix in (4.10) as preconditioners to  $\mathbf{S}$ , i.e.,

1. unconfined problem:  $\tilde{\mathbf{S}}_p = \mathbf{M}_{cr}^I + \bar{\mathbf{M}}_{cf} + \Delta t \mathbf{T}_p$ ,
2. confined problem:  $\tilde{\mathbf{S}}_p = \mathbf{M}_{cr}^{II} + \bar{\mathbf{M}}_{cf} + \Delta t \mathbf{T}_p$ .

Here,  $\mathbf{M}_{cr}^I$  and  $\mathbf{M}_{cr}^{II}$  are given, respectively, in (4.16) and (4.17). They are equivalent to the pressure mass matrix  $\mathbf{M}_p$  in (4.40) scaled by a factor of rock compressibility.

Another preconditioner, which was also proven to be optimal by Lipnikov (2002), is obtained by omitting the  $\mathbf{Q}\mathbf{E}_u^{-1}\mathbf{Q}^T$  term in  $\mathbf{S}$ , i.e.,

$$\tilde{\mathbf{S}} = \bar{\mathbf{M}}_{cf} + \Delta t \bar{\mathbf{T}}_p. \quad (4.48)$$

Comparing with (4.40), it is easy to see that preconditioner (4.48) would be more sensitive to fluid compressibility due to the neglecting of rock compressibility. This is especially true if pure Neumann-type flow boundary conditions are imposed.

#### 4.3.1.2 Preconditioners Based on Other Approximations of $\mathbf{Q}\mathbf{E}_u^{-1}\mathbf{Q}^T$

Suppose that fast efficient linear solvers have already been developed for the operator  $\bar{\mathbf{M}}_{cf} + \Delta t \bar{\mathbf{T}}_p$  in a standard reservoir simulator, then preconditioning the Schur complement  $\mathbf{S}$  depends on the construction of  $\mathbf{Q}\mathbf{E}_u^{-1}\mathbf{Q}^T$ . However, the inversion of  $\mathbf{E}_u$  is generally too expensive, and a direct calculation of  $\mathbf{Q}\mathbf{E}_u^{-1}\mathbf{Q}^T$  is obviously not an option. Therefore, constructing a good approximation of  $\mathbf{Q}\mathbf{E}_u^{-1}\mathbf{Q}^T$  turns out to be the fundamental issue in preconditioning  $\mathbf{S}$ . It is essential that the stencil structure of an approximation matrix be compatible with the structure of  $\bar{\mathbf{M}}_p + \Delta t \bar{\mathbf{T}}_p$ . Otherwise, we may not be able to take advantage of the existing linear solver packages for multiphase flow equations.

Several approximations of  $\mathbf{Q}\mathbf{E}_u^{-1}\mathbf{Q}^T$  can be obtained intuitively, such as

1.  $\{\mathbf{Q} [\text{bdiag}(\mathbf{E}_u)]^{-1} \mathbf{Q}^T\}_{27\text{-point}},$

Preconditioner	$\mathbf{S} = \mathbf{Q}\mathbf{E}_u^{-1}\mathbf{Q}^T + \bar{\mathbf{M}}_{cf} + \Delta t\bar{\mathbf{T}}_p$
$\tilde{\mathbf{S}}_{p1}$	$\bar{\mathbf{M}}_{cf} + \Delta t\bar{\mathbf{T}}_p$
$\tilde{\mathbf{S}}_{p2}$	$\mathbf{M}_{cr}^I + \bar{\mathbf{M}}_{cf} + \Delta t\bar{\mathbf{T}}_p$
$\tilde{\mathbf{S}}_{p3}$	$\mathbf{M}_{cr}^{II} + \bar{\mathbf{M}}_{cf} + \Delta t\bar{\mathbf{T}}_p$
$\tilde{\mathbf{S}}_{p4}$	$\beta \text{diag} \{ \mathbf{Q} [\text{bdiag}(\mathbf{E}_u)]^{-1} \mathbf{Q}^T \} + \bar{\mathbf{M}}_{cf} + \Delta t\bar{\mathbf{T}}_p$
$\tilde{\mathbf{S}}_{p5}$	$\beta \text{diag} \{ \mathbf{Q} [\text{diag}(\mathbf{E}_u)]^{-1} \mathbf{Q}^T \} + \bar{\mathbf{M}}_{cf} + \Delta t\bar{\mathbf{T}}_p$
$\tilde{\mathbf{S}}_{p6}$	if $\left( \frac{\delta\bar{\sigma}}{\bar{\sigma}} < \frac{\delta\sigma_{zz}}{\sigma_{zz}} \right)$ $\tilde{\mathbf{S}}_p = \tilde{\mathbf{S}}_{p2}$ else $\tilde{\mathbf{S}}_p = \tilde{\mathbf{S}}_{p3}$

Table 4.1: Preconditioners for the Schur complement of the pressure equation.

2.  $\text{diag} \{ \mathbf{Q} [\text{bdiag}(\mathbf{E}_u)]^{-1} \mathbf{Q}^T \},$
3.  $\text{diag} \{ \mathbf{Q} [\text{diag}(\mathbf{E}_u)]^{-1} \mathbf{Q}^T \}.$

In the first approximation,  $\text{bdiag}(\mathbf{E}_u)$  represents the block diagonal of  $\mathbf{E}_u$ . Each block is a  $3 \times 3$  matrix corresponding to the coupling of three displacement components on a nodal point. The resulting  $\tilde{\mathbf{S}}$  has a 27-point stencil for 3D rectangular parallelepipeds, which is substantially larger than the 7-point stencil for the pressure equation. The other two approximations are promising due to the diagonal structure. Adding these diagonal matrices to the flow equations is equivalent to taking into account the rock compressibility effect.

In summary we construct five preconditioners that are easy to implement in the IPARS framework, see Table 4.1. Numerical experiments and comparison results are given in Section 4.4.

### 4.3.2 $\tilde{\mathbf{S}}_p$ as a Preconditioner for Krylov Subspace Methods

As we have demonstrated in Section 4.3, an iterative coupling approach can easily be generalized to a fully coupled scheme with preconditioned Richardson iterations. Note that in (4.24)  $\mathbf{p}^{l+1}$  is obtained by using only the information of  $\mathbf{p}^l$  and none of the previous  $\mathbf{p}^m$  ( $m < l$ ) is used explicitly. If a Krylov subspace accelerator is applied instead, convergence can be substantially improved.

Krylov subspace methods generally applied for Biot's system include GMRES (Generalized Minimum Residual), CG (Conjugate Gradient), MINRES (Minimal Residual), GCR (Generalized Conjugate Residuals) and others. They are constructed either as a two-level scheme or as a single level scheme. Both schemes require a good preconditioner for  $\mathbf{S}$  to achieve reasonable convergence. However, that better linear solver performance can be obtained by adding an approximate  $\mathbf{Q}\mathbf{E}_u^{-1}\mathbf{Q}^T$  to the pressure block has not been well recognized. In some cases, it is merely neglected as shown in Prevost (1997) and Lipnikov (2002). Advantages of accounting for  $\mathbf{Q}\mathbf{E}_u^{-1}\mathbf{Q}^T$  in the approximation of  $\mathbf{S}$  are twofold, namely:

1. Acceleration of convergence rate by reducing the condition number of a preconditioning matrix.
2. More stable flow model system. In the case of incompressible fluid and  $\partial\Omega^D = 0$ , decoupling of pressure and displacement equations using  $\tilde{\mathbf{S}} =$

$\bar{\mathbf{M}}_p + \Delta t \bar{\mathbf{T}}_p$  leads to an elliptic problem with  $\partial\Omega = \partial\Omega^N$ , i.e.,

$$\Delta t \bar{\mathbf{T}}_p \delta \mathbf{p} = \mathbf{R}_f,$$

which does not have a unique solution. Adding the  $\mathbf{Q}\mathbf{E}_u^{-1}\mathbf{Q}^T$  term to  $\mathbf{S}$  transforms an elliptic problem back into a parabolic one.

## 4.4 Numerical Results

In this section we present several numerical examples to evaluate the performance of different preconditioners suggested in Table 4.1. Convergence rate is reported based on the Newton iteration counts for single or multiple time steps. Numerical examples are tested on a 2.0 GHz AMD Athlon Dual processor machine with 1 GB memory for each processor. In all cases, though, only one processor is used.

Newton convergence is achieved when (1) the relative residual for each component (oil, water and gas) with respect to the total in-situ fluid volume is less than  $10^{-6}$  and (2) the maximum relative pore volume error is less than  $10^{-4}$ . The algebraic linear system for the single phase or multiphase flow equations (4.25) is solved iteratively by GMRES with LSOR preconditioning. The backsolve for displacement is either by a direct method or by an iterative scheme. In the case that an iterative method is used, tolerance for linear solver convergence is chosen to be tight enough to ensure that the outer Newton iterations are not affected.

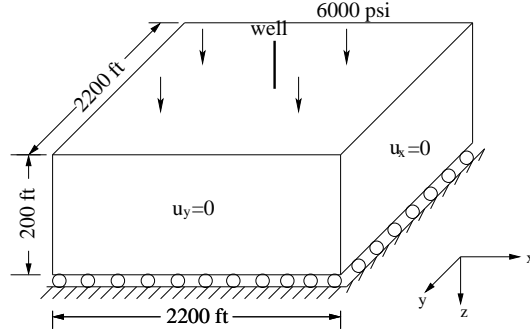


Figure 4.1: Example 1: Confined reservoir with no flow boundaries.

#### 4.4.1 Examples

##### 4.4.1.1 Example 1: Depletion in a Confined Reservoir

Reservoir geometry and boundary conditions are shown in Figure 4.1. A zero normal displacement condition is enforced on all lateral and bottom boundary faces (confined problem). A constant vertical stress of 6000 psi is applied on the top. The finite element mesh consists of 1210 ( $11 \times 11 \times 10$ ) uniform 8-noded brick elements. Initial reservoir pressure is 3000 psi at the depth of 6000 ft with a vertical gradient of 0.4333 psi/ft. Initial horizontal stresses are 4000 psi over the entire reservoir. Initial vertical stress is 6000 psi at the depth of 6000 ft with a vertical gradient of 1.0231 psi/ft. A vertical well with a wellbore radius of 0.25 ft is completed in the center of the reservoir. It produces at a constant rate of 15,000 stb/day for 500 days. A no-flow boundary condition is assumed. Other parameters specified in this case are  $k_x = k_y = 50$  md,  $k_z = 5$  md,  $\phi = 0.2$ ,  $c_f = 0.0$  1/psi,  $\mu_f = 1.0$  cp,  $E = 10000$  psi,  $\nu = 0.3$ ,  $\alpha = 1$ , and  $\frac{1}{M} = 0.0$ .

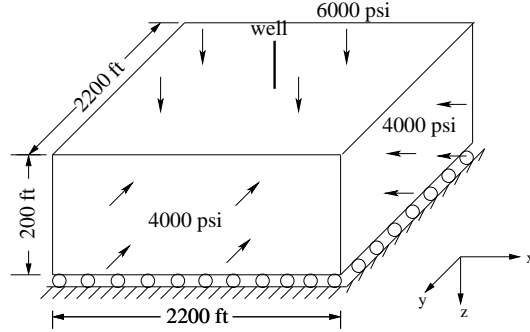


Figure 4.2: Example 2: Unconfined reservoir with no flow boundaries.

Numerical studies are conducted to investigate the convergence behavior of the preconditioners in Table 4.1 with respect to fluid compressibility, rock permeability, Young's modulus and Poisson's ratio. A constant time step of 10 days is used for this problem.

#### 4.4.1.2 Example 2: Depletion in an Unconfined Reservoir

The second example is slightly modified from the first by enforcing traction boundary conditions on the lateral boundary faces, see Figure 4.2. This case is used to verify the assumptions under which  $\tilde{S}_{p2}$  is derived.

#### 4.4.1.3 Example 3: A Confined Reservoir with Surrounding Non-pay Rocks

The reservoir is at a depth of 10,000 ft underground. It has an area of  $22,000 \times 11,000$  ft. The total thickness is 250 ft. To characterize the changing boundary conditions around the reservoir our numerical model includes the surrounding non-pay rocks in the calculation, as shown in Figure 4.3. The

dimensions of the computational domain are  $62,000 \times 31,000 \times 10,450$  ft, including an overburden of 10,000 ft, an underburden of 200 ft, and a sideburden of 10,000 ft at each side of the reservoir flanks. The domain is non-uniformly discretized into  $21 \times 21 \times 12$  8-noded brick elements with much coarser grids in the extended regions (Figure 4.4). The model is confined at the bottom and lateral boundary faces, i.e., no normal displacement is allowed at those boundaries. The top of the grid is traction-free. A no-flow boundary condition is applied for the surrounding non-pay rocks.

The initial rock porosity is 0.25. The horizontal and vertical permeabilities are respectively 100 md and 10 md in the reservoir while they are set to zero in the non-pay rocks. Both regions contain the same single phase fluid with the following properties:  $c_f = 3.0 \times 10^{-6}$  psi<sup>-1</sup> and  $\mu_f = 1$ cp. The initial reservoir pressure satisfies the local equilibrium condition and is set to be 14.7 psi at the surface. The initial vertical stress is 0 psi at the top and increases with depth at a gradient of 0.9869 psi/ft. The horizontal stresses are assumed to be half of the vertical stress. The elastic moduli are  $1 \times 10^4$  psi in the reservoir and  $1 \times 10^6$  psi in its surroundings. Poisson's ratio is 0.25 everywhere.

A vertical well is completed in the center of the reservoir. It is only perforated in the reservoir layers. The model is simulated for a 4000-day production scenario. The well produces at a constant rate of 50,000 stb/day. Small time steps of 20 days each are used for the first 500 days followed by time steps of 200 days until the end of the simulation.

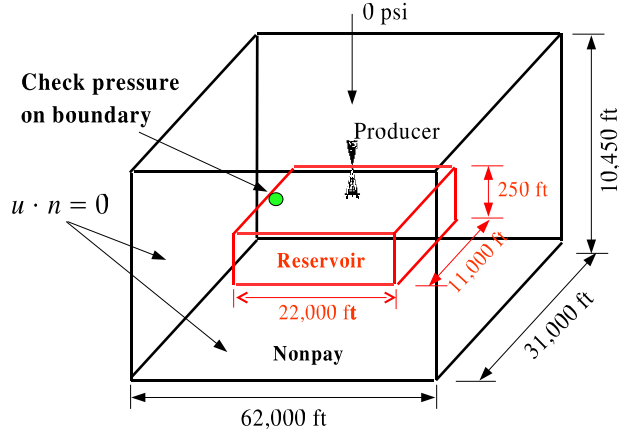


Figure 4.3: Example 3: Confined reservoir with surrounding non-pay rocks.

If  $E_1$  and  $E_2$  denote, respectively, the Young's modulus inside and outside the reservoir, the sensitivity of our preconditioners with respect to the jump ratio ( $E_2/E_1$ ) will be investigated.

#### 4.4.1.4 Example 4: an Unconfined Reservoir with Stiff Non-pay Rocks

This is an unconfined case modified from Example 3. It is designed to show that boundary conditions are not the sole factor in determining an efficient preconditioner  $\tilde{\mathbf{S}}_p$ . The convergence speed of an iteratively coupled scheme also depends on reservoir heterogeneity and the interaction between the reservoir and its surroundings. Here we have chosen  $E_1 = 10^4$  psi (inside the reservoir) and  $E_2 = 10^6$  psi (outside the reservoir). Notice that the non-pay rocks are stiffer than the reservoir rocks. Small time steps of 20 days each

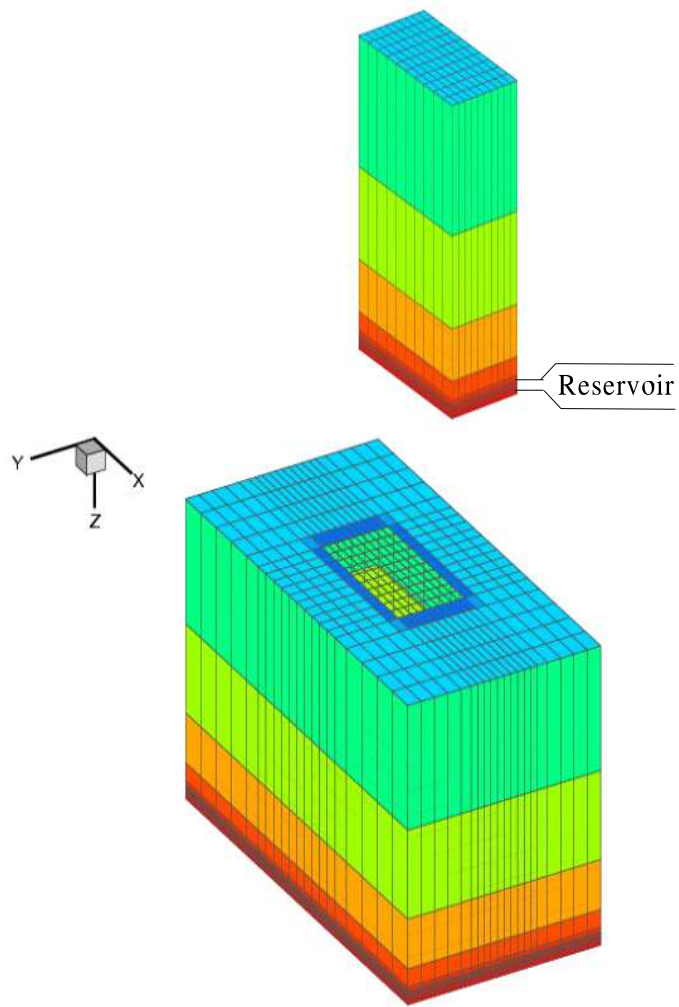


Figure 4.4: Example 3: Finite element discretization

are used for the first 500 days followed by time steps of 200 days until the end of the simulation.

#### **4.4.1.5 Example 5: an Unconfined Reservoir with Soft Non-pay Rocks**

The only difference between Example 5 and Example 4 lies in the Young's moduli specified in and out of the reservoir. In this case, a stiffer reservoir is surrounded by softer non-pay rocks, i.e.,  $E_1 = 10^6$  psi and  $E_2 = 10^4$  psi. The time steps used here are the same as those used in Example 3. This case will be tested to determine the preconditioner with the fastest convergence rate, which may not necessarily be the  $\tilde{S}_{p2}$  in Table 4.1.

#### **4.4.1.6 Example 6: Three-phase Flow in a Confined Reservoir**

This is a three-dimensional, three-phase water-flooding problem in a quarter of a five spot pattern. The size of the reservoir is  $1056 \times 1056 \times 160$  ft. The domain is discretized uniformly into a grid of  $16 \times 16 \times 8$  which will be further refined to evaluate the performance of  $(\tilde{S}_p)$  with respect to the mesh parameter  $h$ . The initial in-situ porosity is 0.3. The initial oil saturation is 0.8. No free gas is present. The permeability field varies by layers. The Young's modulus and Poisson's ratio are, respectively,  $3.095 \times 10^4$  psi and 0.3. Two wells are completed at the reservoir corners. The injection well injects water at a rate of 500 stb per day. The production well produces 750 stb of oil per day. A load of 6000 psi is applied on the top of the reservoir. All the other reservoir boundary faces are confined in the normal directions. A time step

that varies from 0.1 to 10 days is specified in this example.

#### 4.4.2 Convergence with Respect to Fluid and Rock Properties

The Schur complement matrix  $\mathbf{S}$  in (4.20) is composed of two parts, namely,  $\bar{\mathbf{M}}_{cf} + \Delta t \mathbf{T}_p$  and  $\mathbf{Q}\mathbf{E}_u^{-1}\mathbf{Q}^T$ , which correspond to two physical processes: fluid flow and rock deformation. We are interested in the effect of these physical interactions on the performance of our preconditioners. Thus, the first example is tested with varying fluid or rock properties. We report the total number of Newton iterations for the first two time steps.

Comparison results are plotted in Figure 4.5. The missing points for  $\tilde{\mathbf{S}}_{p1}$  indicate that the iterative scheme does not converge. Our observations based on these numerical results are as follows:

1. While  $\tilde{\mathbf{S}}_{p1}$  is an optimal preconditioner for the Schur complement matrix  $\mathbf{S}$ , its performance depends on the fluid compressibility ( $c_f$ ). For pure Neumann flow boundaries, the fluid has to be sufficiently compressible to achieve convergence. For example,  $\tilde{\mathbf{S}}_{p1}$  fails to converge until  $c_f$  is larger than  $10^{-4}$  psi<sup>-1</sup>, as shown in the upper left picture in Figure 4.5.
2. In the case of small  $c_f$ , the Newton convergence is greatly affected by the approximation of  $\mathbf{Q}\mathbf{E}_u^{-1}\mathbf{Q}^T$ . Different choices of  $\tilde{\mathbf{S}}_p$  result in substantially different numbers of nonlinear iterations. However, as  $c_f$  is getting larger, the performance of these preconditioners tends to be the same.
3. In reservoirs with high permeability, the strong flow dominant conditions

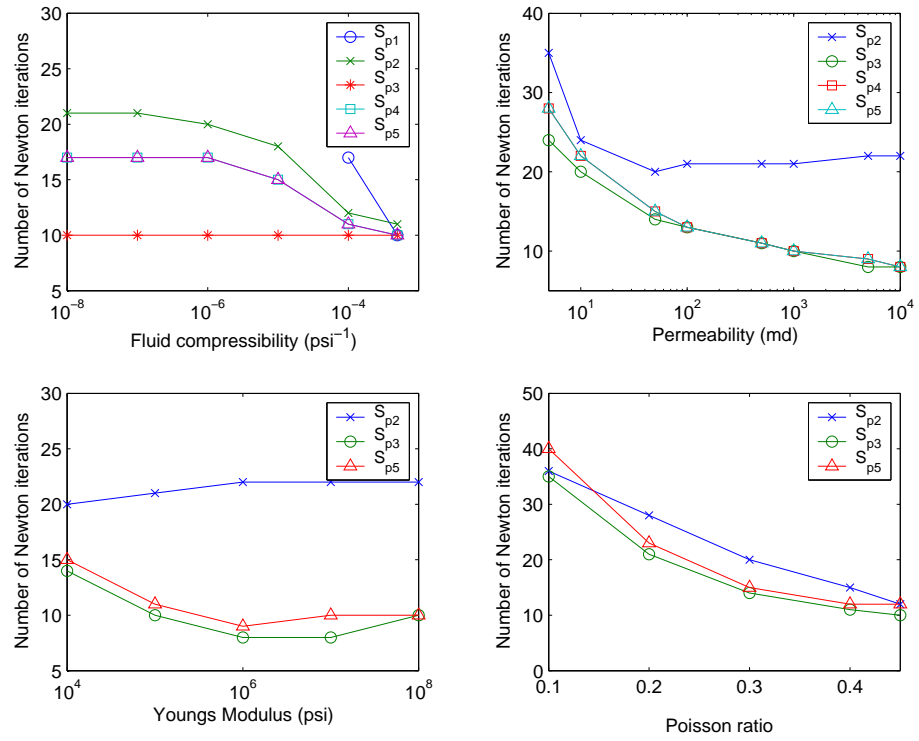


Figure 4.5: Example 1 with varying fluid compressibility and rock properties: Number of Newton iterations for the first two time steps.

dictate a consolidation scenario that is closer to a drained deformation. The upper right picture in Figure 4.5 demonstrate a faster convergence at high permeability than thoses at low permeabilities. However, differences in the performance of these preconditioners still exist, whether  $k$  is low or high.

4. The effect of Young's modulus on the Newton convergence  $\tilde{S}_p$  is not as drastic as that of fluid compressibility and rock permeability. Faster convergence is observed with a moderate value of  $E$ .
5. Poisson's ratio also plays an important role in controlling the nonlinear iterations. As  $\nu$  increases, all preconditioners achieve significant reductions in the Newton iteration counts.
6. Example 1 is a confined homogeneous reservoir. The underlying assumption for  $\tilde{S}_{p3}$  is well satisfied. Thus, it performs consistently better over the full range of our parameter tests. In contrast,  $\tilde{S}_{p2}$  performs poorly as compared to other preconditioners.
7.  $\tilde{S}_{p4}$  and  $\tilde{S}_{p5}$  hardly show any difference in terms of Newton iteration counts. Thus, the decoupling of displacement components at each nodal point is clearly a method of choice.

#### 4.4.3 Convergence with Respect to Coefficient Jump

In Example 3, we consider a non-homogeneous domain that consists of two subdomains. At the interface there is a discontinuous jump in the Young's

$E_2/E_1$	$\tilde{\mathbf{S}}_{p1}$	$\tilde{\mathbf{S}}_{p2}$	$\tilde{\mathbf{S}}_{p3}$	$\tilde{\mathbf{S}}_{p4}$	$\tilde{\mathbf{S}}_{p5}$
$10^0$	*	83	47	39	39
$10^1$	*	41	24	17	17
$10^2$	*	80	46	24	24
$10^3$	*	149	85	43	43

Table 4.2: Example 3 with varying ratios of jump in Young’s modulus: Total number of Newton iterations for the first two time steps. The symbol \* indicates failure to converge.

modulus ( $E$ ). Numerical experiments are conducted to investigate the robustness of our preconditioners with respect to the jump ratio ( $E_2/E_1$ ). Table 4.2 reports the number of Newton iterations for the first two time steps, and Table 4.3 shows the total Newton iteration counts and CPU time in seconds for the entire run. Comparison of these results clearly shows that,

1.  $\tilde{\mathbf{S}}_{p1}$  does not converge for any choice of  $E_2/E_1$  even though the fluid is slightly compressible ( $c_f = 3.0 \times 10^{-6}$  psi<sup>-1</sup>). Numerical experiments suggest that fluid compressibility has to be roughly in the same order of magnitude as  $\frac{1}{E}$  in order for  $\tilde{\mathbf{S}}_{p1}$  to converge.
2. As expected,  $\tilde{\mathbf{S}}_{p2}$  converges almost two times slower than the other preconditioners.
3.  $\tilde{\mathbf{S}}_{p4}$  and  $\tilde{\mathbf{S}}_{p5}$  are superior to  $\tilde{\mathbf{S}}_{p3}$  with faster convergence (95.7 seconds as compared to 146.5 seconds in Table 4.3) and less degradation with large jump ratios (Table 4.2) .

Preconditioner	Newton iterations	CPU time (sec)
$\tilde{\mathbf{S}}_{p2}$	333	228.1
$\tilde{\mathbf{S}}_{p3}$	189	146.5
$\tilde{\mathbf{S}}_{p4}$	106	95.7
$\tilde{\mathbf{S}}_{p5}$	106	96.1

Table 4.3: Example 3 with  $E_1 = 10^4$  psi and  $E_2 = 10^6$  psi: Total number of Newton iterations for the entire simulation with a total of 38 time steps.

Figure 4.6 shows the time evolution of Newton iteration counts per time step over the entire simulation. Notice that the Newton convergence is much slower at the early stage of depletion (less than 100 days in Figure 4.6). An increased time step may also cause slow convergence (400 - 1000 days in Figure 4.6). In these two cases, preconditioning the Schur complement matrix  $\mathbf{S}$  is very important. It determines the efficiency of an iteratively coupled approach.  $\tilde{\mathbf{S}}_{p5}$  appears to be more robust in handling coefficient jumps and performs consistently better than the other preconditioners.

#### 4.4.4 Convergence with Respect to Grid Refinement

Robustness with respect to the discretization parameter  $h$  is another important criteria in evaluating the performance of a preconditioner. Example 6 is used here to test the deterioration of convergence with grid refinement. The number of Newton iterations for the first two time steps are given in Table 4.4.

$\tilde{\mathbf{S}}_{p1}$  still does not converge. The other preconditioners demonstrate a fairly stable convergence rate with refined meshes. This observation is justified

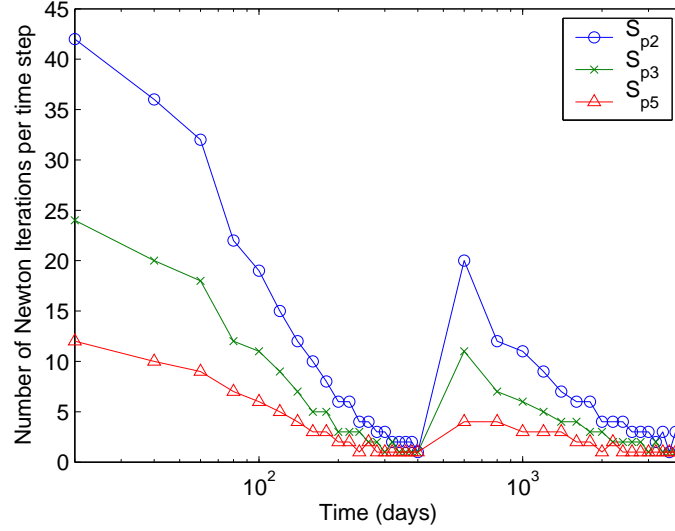


Figure 4.6: Example 3 with  $E_1 = 10^4$  psi and  $E_2 = 10^6$  psi: Time evolution of Newton iteration counts per time step over the entire reservoir depletion.

by the spectral equivalency analysis in Section 4.3.1.1.

#### 4.4.5 Convergence with Respect to Reservoir Heterogeneity

In Section 4.2.3 we derive two approximations to  $\mathbf{Q}\delta\mathbf{u}$  based on reservoir boundary conditions. However, in some applications the reservoir is strongly heterogeneous and the interaction between the reservoir and its surroundings is

Grid	$\tilde{\mathbf{S}}_{p1}$	$\tilde{\mathbf{S}}_{p2}$	$\tilde{\mathbf{S}}_{p3}$	$\tilde{\mathbf{S}}_{p4}$	$\tilde{\mathbf{S}}_{p5}$
$16 \times 16 \times 8$	*	12	9	18	18
$32 \times 32 \times 16$	*	12	11	19	19
$64 \times 64 \times 32$	*	13	13	19	19

Table 4.4: Example 6 with grid refinements: Total number of Newton iterations for the first two time steps. The symbol \* indicates failure to converge.

complicated. Thus, the two simplified models may not represent real physics.

The poor performance of a preconditioner, chosen on the sole base of boundary types, can be clearly seen from the numerical results shown in Table 4.5 and Figure 4.7. Since Example 4 is an unconfined reservoir,  $\tilde{\mathbf{S}}_{p2}$  is expected to be the optimal preconditioner. However, the Newton convergence history in Figure 4.7 demonstrates the opposite.  $\tilde{\mathbf{S}}_{p2}$  converges at least 1.5 times slower than  $\tilde{\mathbf{S}}_{p3}$ . This indicates that the reservoir deforms more like a uniaxial strain type even though the domain is not confined laterally. Numerical experiments with Example 5 shows similar results (Table 4.5 and the lower picture in Figure 4.7).

Here we propose a practical strategy for choosing an effective preconditioner for the iterative coupling. It is based on the relative changes in total stress and vertical stress. We denote the resulting preconditioner by  $\mathbf{S}_{p6}$  in Table 4.1. The procedure is summarized as follows:

1. *Start with time step  $n$ .*

- (a) *If  $(n = 1)$ , compute  $\mathbf{M}_{cr}$  according to the boundary conditions.*
- (b) *If  $(n > 1)$ , compute the relative changes in mean stress and vertical stress in each cell by*

$$\begin{aligned}\delta\bar{\sigma}^{rel} &= \frac{\bar{\sigma}^e - \bar{\sigma}^{n-1}}{\bar{\sigma}^{n-1}}, \\ \delta\sigma_{zz}^{rel} &= \frac{\sigma_{zz}^e - \sigma_{zz}^{n-1}}{\sigma_{zz}^{n-1}},\end{aligned}$$

	Example 4		Example 5	
$\tilde{\mathbf{S}}_p$	Newt. iterations	Time (sec)	Newt. iterations	Time (sec)
$\tilde{\mathbf{S}}_{p2}^*$	331	244.0	300	296.1
$\tilde{\mathbf{S}}_{p3}$	189	157.6	234	247.8
$\tilde{\mathbf{S}}_{p6}$	210	162.9	197	212.5

Table 4.5: Examples 4 & 5: Total number of Newton iterations and CPU time for the entire simulation with 38 time steps. Example 4:  $E_1 = 10^4$  psi and  $E_2 = 10^6$  psi; Example 5:  $E_1 = 10^6$  psi and  $E_2 = 10^4$  psi.

or simply,

$$\begin{aligned}\delta\bar{\sigma}^{rel} &= \frac{\bar{\sigma}^{n-1} - \bar{\sigma}^{n-2}}{\bar{\sigma}^{n-2}}, \\ \delta\sigma_{zz}^{rel} &= \frac{\sigma_{zz}^{n-1} - \sigma_{zz}^{n-2}}{\sigma_{zz}^{n-2}},\end{aligned}$$

where  $\bar{\sigma}^e$  and  $\sigma_{zz}^e$  are the extrapolated values of  $\bar{\sigma}$  and  $\sigma_{zz}$  respectively.

- i. if  $(\delta\bar{\sigma}^{rel} \leq \delta\sigma_{zz}^{rel})$  choose  $\tilde{\mathbf{S}}_p = \tilde{\mathbf{S}}_{p2}$ .
- ii. if  $(\delta\bar{\sigma}^{rel} > \delta\sigma_{zz}^{rel})$  choose  $\tilde{\mathbf{S}}_p = \tilde{\mathbf{S}}_p^3$ .

2. Solve the coupled problem iteratively for pressure, concentration and displacement.
3. Compute total stresses and extrapolate the values to the next time step.
4. Go back to step 1 for a new time step.

The residual reduction history in Figure 4.7 shows that  $\tilde{\mathbf{S}}_{p6}$  is more effective than the default  $\tilde{\mathbf{S}}_{p2}$  preconditioner. It almost reproduces the history

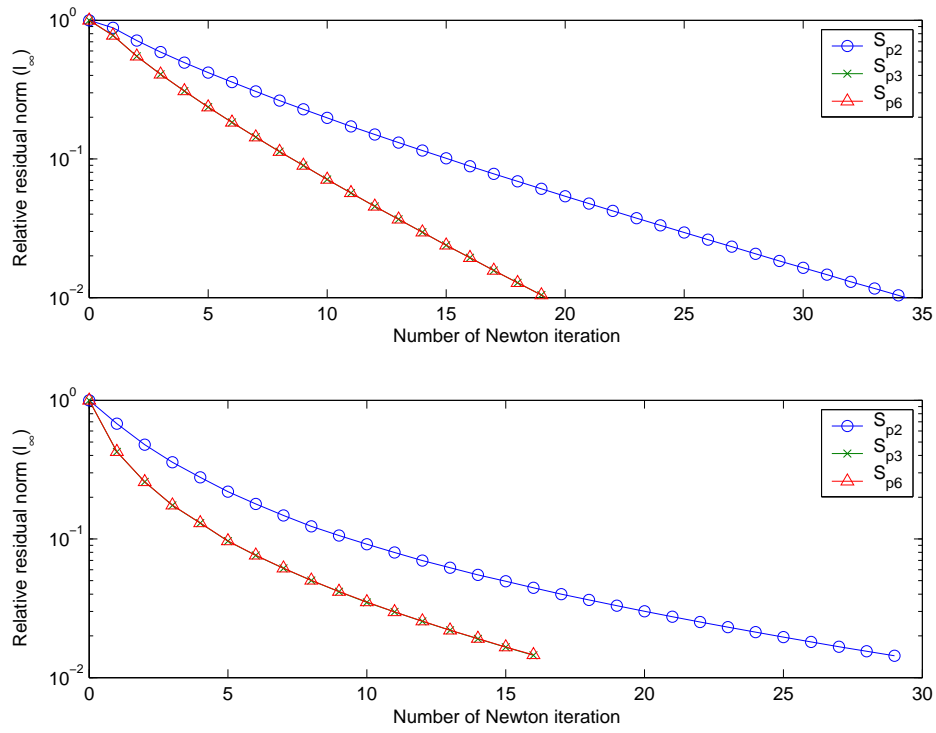


Figure 4.7: Examples 4 & 5: Residual reduction histories with respect to the number of Newton iterations in the second time step. Upper: Example 4; Lower: Example 5.

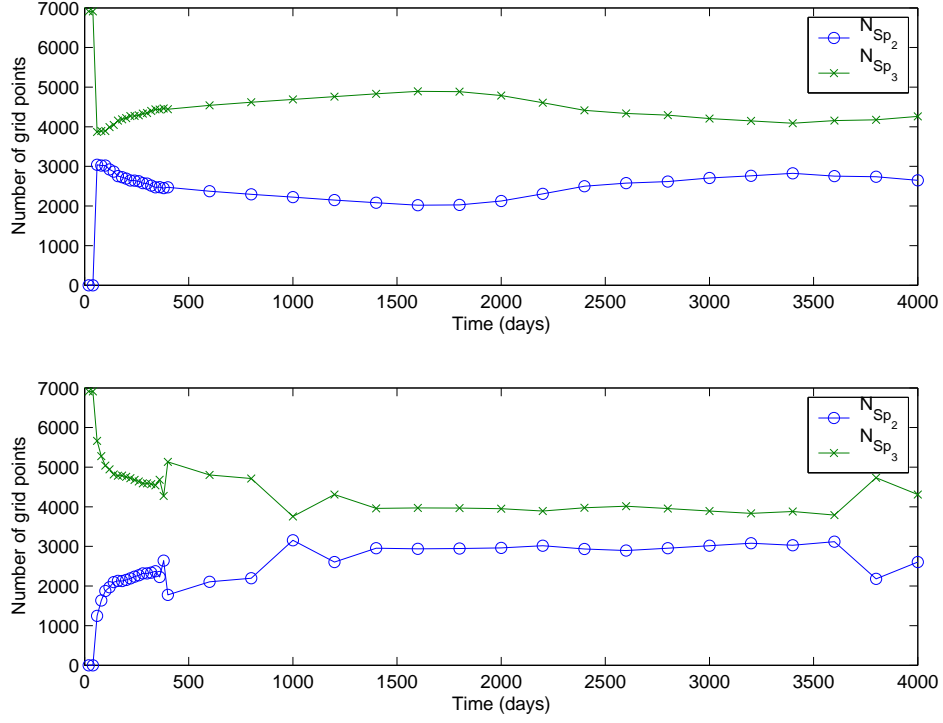


Figure 4.8: Total numbers of grid points using  $\mathbf{S}_{p2}(\mathbf{M}_{cr}^I)$  and  $\mathbf{S}_{p3}(\mathbf{M}_{cr}^{II})$  respectively at different reservoir depletion time. Upper: Example 4; Lower: Example 5.

by  $\tilde{\mathbf{S}}_{p3}$ . In the case of Example 5 where a stiff reservoir is surrounded by soft non-pay rocks,  $\tilde{\mathbf{S}}_{p6}$  performs better than  $\tilde{\mathbf{S}}_{p2}$  and  $\tilde{\mathbf{S}}_{p3}$  (Table 4.5).

Figure 4.8 shows the total number of grid cells that use  $\tilde{\mathbf{S}}_{p2}$  and  $\tilde{\mathbf{S}}_{p3}$ , respectively, as reservoir depletion proceeds.. The transition between  $\tilde{\mathbf{S}}_{p2}$  and  $\tilde{\mathbf{S}}_{p3}$  mostly occurs at the early stage of consolidation or at large time step changes.

	$\beta$ in $\beta \text{diag} \{ \mathbf{Q} [\text{diag} (\mathbf{E}_u)]^{-1} \mathbf{Q}^T \}$						
Time step $n$	1.0	1.1	1.2	1.28 <sup>a</sup>	1.4 <sup>o</sup>	1.6	1.8
Step 1	21	15	11	9	8	9	11
Step 2	23	16	12	11	8	9	11
Step 3	22	16	12	10	8	9	11
Total ( $n = 3$ )	66	47	35	30	24	27	33
Total ( $n = 50$ )	401	292	231	201 <sup>a</sup>	176 <sup>o</sup>	188	213

Table 4.6: Example 1: Newton iteration counts using  $\tilde{\mathbf{S}}_{p5}$  with varying scaling factors ( $\beta^o$ : optimal factor,  $\beta^a$ : approximate factor).

	$\beta$ in $\beta \text{diag} \{ \mathbf{Q} [\text{diag} (\mathbf{E}_u)]^{-1} \mathbf{Q}^T \}$						
Time step $n$	1.0	1.2	1.4	1.6	1.8 <sup>o</sup>	1.9	2.07 <sup>a</sup>
Step 1	*	61	17	9	7	7	8
Step 2		65	19	11	7	7	7
Step 3		65	19	11	7	6	7
Total ( $n = 3$ )		191	55	31	21	20	22
Total ( $n = 50$ )		1021	307	191	182 <sup>o</sup>	191	207 <sup>a</sup>

Table 4.7: Example 2: Newton iteration counts using  $\tilde{\mathbf{S}}_{p5}$  with varying scaling factors ( $\beta^o$ : optimal factor,  $\beta^a$ : approximate factor). The symbol \* indicates failure to converge.

#### 4.4.6 Scaling Factor $\beta$ in $\tilde{\mathbf{S}}_{p5}$

As shown in Section 4.4.3,  $\tilde{\mathbf{S}}_{p5}$  is more robust in handling reservoir heterogeneity. However, its convergence rate depends on the scaling factor  $\beta$  (Table 4.1). For some unconfined problems  $\mathbf{S}_{p5}$  fails to converge with  $\beta = 1.0$ . In this section, we present a simple, but not the optimal, approach for computing  $\beta$ .

In Tables 4.6—4.10 we report the number of Newton iterations using  $\tilde{\mathbf{S}}_{p5}$

	$\beta$ in $\beta \text{diag} \{ \mathbf{Q} [\text{diag} (\mathbf{E}_u)]^{-1} \mathbf{Q}^T \}$					
Time step $n$	0.55	0.6	$0.8^o$	$1.07^a$	1.2	1.4
Step 1	10	7	10	13	14	16
Step 2	11	7	9	11	12	14
Step 3	10	7	7	10	11	13
Total ( $n = 3$ )	31	21	26	21	37	43
Total ( $n = 50$ )	104	94	$91^o$	$110^a$	121	138

Table 4.8: Example 3: Newton iteration counts using  $\tilde{\mathbf{S}}_{p5}$  with varying scaling factors ( $\beta^o$ : optimal factor,  $\beta^a$ : approximate factor).

	$\beta$ in $\beta \text{diag} \{ \mathbf{Q} [\text{diag} (\mathbf{E}_u)]^{-1} \mathbf{Q}^T \}$						
Time step $n$	0.55	0.6	0.8	$1.0^o$	1.2	1.4	$1.92^a$
Step 1	11	8	10	12	14	16	22
Step 2	11	7	9	10	12	14	19
Step 3	11	6	7	9	11	13	17
Total ( $n = 3$ )	33	21	26	21	37	43	58
Total ( $n = 50$ )	293	194	118	$110^o$	123	138	$183^a$

Table 4.9: Example 4: Newton iteration counts using  $\tilde{\mathbf{S}}_{p5}$  with varying scaling factors ( $\beta^o$ : optimal factor,  $\beta^a$ : approximate factor).

	$\beta$ in $\beta \text{diag} \{ \mathbf{Q} [\text{diag} (\mathbf{E}_u)]^{-1} \mathbf{Q}^T \}$						
Time step $n$	1.0	1.35	1.4	$1.5^o$	$1.56^a$	1.6	1.7
Step 1	*	37	21	20	21	22	23
Step 2		40	21	18	19	20	21
Step 3		39	21	16	17	17	18
Total ( $n = 3$ )		116	63	54	57	59	62
Total ( $n = 50$ )		245	185	$183^o$	$192^a$	199	212

Table 4.10: Example 5: Newton iteration counts using  $\tilde{\mathbf{S}}_{p5}$  with varying scaling factors ( $\beta^o$ : optimal factor,  $\beta^a$ : approximate factor). The symbol \* indicates failure to converge.

Example	Boundary type	$\tilde{\mathbf{S}}_{p5}(\beta = 1.0)$	$\tilde{\mathbf{S}}_{p2}$ or $\tilde{\mathbf{S}}_{p3}$ (based on B.C.s)	
		$c_r^{\mathbf{S}_{p5}}(\text{psi}^{-1})$	$c_r(\text{psi}^{-1})$	$c_r/c_r^{\mathbf{S}_{p5}}$
1	confined	$5.81 \times 10^{-5}$	$7.43 \times 10^{-5}$	1.28
2	unconfined	$5.81 \times 10^{-5}$	$1.2 \times 10^{-4}$	2.07
3	confined	$1.09 \times 10^{-6}$	$1.17 \times 10^{-6}$	1.07
4	unconfined	$1.09 \times 10^{-6}$	$2.10 \times 10^{-6}$	1.92
5	unconfined	$1.53 \times 10^{-4}$	$2.38 \times 10^{-4}$	1.56

Table 4.11: Rock compressibility predicted by different approximations to  $\mathbf{Q}\mathbf{E}_u^{-1}\mathbf{Q}^T$ .

with varying scaling factors ( $\beta$ ). In these tables,  $\beta^o$  denotes the optimal scaling factor obtained by numerical experiments, and  $\beta^a$  denotes our approximate value. We summarize the results as follows:

1. All these results show that the scaling factor affects the performance of  $\tilde{\mathbf{S}}_{p5}$ . The optimal value of  $\beta$  is problem dependent.
2. The average rock compressibility ( $c_r^{\tilde{\mathbf{S}}_{p5}}$ ) predicted by  $\tilde{\mathbf{S}}_{p5}$  ( $\beta = 1.0$ ) is close to the value estimated by  $\tilde{\mathbf{S}}_{p3}$  for confined reservoirs (Example 1 & 3 in Table 4.11). However, for unconfined problems,  $c_r^{\tilde{\mathbf{S}}_{p5}}$  is much smaller than the compressibility computed by  $\tilde{\mathbf{S}}_{p2}$  (Example 2, 4 & 5 in Table 4.11).

Thus, the scaling factor  $\beta$  can be simply approximated by

$$\beta = \frac{\bar{c}_r}{\bar{c}_r^{\tilde{\mathbf{S}}_{p5}}}, \quad (4.49)$$

where  $\bar{c}_r$  is the average rock compressibility estimated by  $\tilde{\mathbf{S}}_{p2}$  or  $\tilde{\mathbf{S}}_{p3}$  according to the boundary conditions, as shown below:

$$\begin{aligned}\bar{c}_r^{\tilde{\mathbf{S}}_{p2}} &= \frac{\sum_{i=1}^{n_p} (M_{cr}^I)_{ii}}{\int_{\Omega} d\Omega} \quad (\text{unconfined}), \\ \bar{c}_r^{\tilde{\mathbf{S}}_{p3}} &= \frac{\sum_{i=1}^{n_p} (M_{cr}^{II})_{ii}}{\int_{\Omega} d\Omega} \quad (\text{confined}).\end{aligned}$$

Similarly,  $\bar{c}_r^{\tilde{\mathbf{S}}_{p5}}$  is computed by

$$\bar{c}_r^{\tilde{\mathbf{S}}_{p5}} = \frac{\sum_{i=1}^{n_p} \{ \mathbf{Q} [\text{diag}(\mathbf{E}_u)]^{-1} \mathbf{Q}^T \}_{ii}}{\int_{\Omega} d\Omega}$$

In Tables 4.6—4.10, the columns of  $\beta^a$  show the total number of Newton iterations using  $\beta$  calculated by (4.49). We observe that the approximation of (4.49) leads to reasonable convergence rates that are close to the optimal ones. It should be mentioned that Example 4 behaves more like a uniaxial strain deformation even though it is an unconfined problem. Thus,  $\text{bar}c_r$  in (4.49) is over estimated by  $\tilde{\mathbf{S}}_{p2}$ , and the resulting  $\beta$  leads to poor convergence.

#### 4.4.7 A Fully Coupled Approach with Preconditioned Richardson Iterations

If an iterative coupling is understood as the first iteration of a preconditioned Richardson method defined in (4.28), then it can be adapted easily to a converged iterative scheme or a fully coupled scheme. This is achieved by applying the operator splitting repeatedly in each Newton step until a given tolerance is satisfied.

Here we use the first three examples in Section 4.4.1 to compare the efficiency of the two coupling schemes. For the iterative coupling we choose

		Iteratively coupled		Fully coupled	
Example	$\tilde{\mathbf{S}}_p$	Newt. Iter.	Time (sec)	Newt/linear Iter.	Time (sec)
1	$\tilde{\mathbf{S}}_{p3}$	272	24.7	106/362	30.1
2	$\tilde{\mathbf{S}}_{p2}$	240	22.7	105/360	29.7
3	$\tilde{\mathbf{S}}_{p3}$	316	273.7	41/250	234.7
3	$\tilde{\mathbf{S}}_{p5}$	194	210.7	41/166	193.1

Table 4.12: Comparison of Newton iteration count and CPU time in seconds for the iteratively coupled and the fully coupled approach. The discrete elasticity system is solved by a direct solver.

		Iteratively coupled		Fully coupled	
Example	$\tilde{\mathbf{S}}_p$	Newt. Iter.	Time (sec)	Newt/linear Iter.	Time (sec)
1	$\tilde{\mathbf{S}}_{p3}$	277	76.2	106/406	94.7
2	$\tilde{\mathbf{S}}_{p2}$	249	228.9	105/364	371.4
3	$\tilde{\mathbf{S}}_{p3}$	316	389.4	41/250	315.1
3	$\tilde{\mathbf{S}}_{p5}$	194	226.8	41/166	215.6

Table 4.13: Comparison of Newton iteration count and CPU time in seconds for the iteratively coupled and the fully coupled approach. The discrete elasticity system is solved by ICCG.

a tolerance that is sufficient tight to reproduce the pressure and displacement solutions that are computed by the fully coupled technique. Preconditioned Richardson iterations are terminated once the initial residual ( $l_\infty$  norm) of the pressure equation (4.28) is reduced by a factor of  $10^2$ .

Results in Table 4.13 demonstrate that a significant reduction in the total number of Newton iterations is obtained by the converged iterative scheme, (4.28) and (4.26). However, the faster Newton convergence is at the expense of more inner iterations for pressure and displacement. Notice that the nonlin-

earity of this problem is due to the rate specified wells. Thus, more than one Newton iterations are required to satisfy the tolerance for nonlinear iterations. The efficiency of the iterative coupling lies in that it solves the elasticity block once per Newton iteration. Thus, its overall merits, in terms of efficiency and accuracy, are comparable to a fully coupled approach. Similar comparison results can be found in Dean et al. (2003).

# Chapter 5

## Numerical Verification

In this chapter, our goal is to provide numerical validation for the coupled flow and geomechanics model described in Chapter 2. First it will be demonstrated that the governing equations can be solved to a satisfactory accuracy using the FE approximations proposed in Chapter 3. Next, we will verify that the numerical solutions computed by the iterative coupling technique are as accurate as those computed by a fully coupled scheme. Two examples will be presented. The first one is the classical Mandel problem with the well-known Mandel-Cryer effect (Mandel 1953, Cheng 1988 and Abousleiman et al. 1996). Numerical and analytical solutions for this plane strain problem will be compared. The second example is a 3D single-phase and single-well reservoir production problem with surrounding non-pays (Dean et al. 2003). Its numerical results will be validated using ACRES (ARCO's Comprehensive Reservoir Simulator) (see ARCO Reservoir Simulator Development 1999).

### 5.1 Mandel's Problem

The geometry of Mandel's problem is depicted in Figure 5.1. An infinitely long specimen (perpendicular to the paper) with a rectangular cross-

section is sandwiched between two rigid, frictionless and impermeable plates. The specimen consists of incompressible solid constituents, and it is saturated with a single-phase incompressible fluid. The initial pore pressure,  $p^0$ , is also the ambient pressure. At  $t = 0^+$  a force of  $2F$  per unit thickness of the specimen is applied at the top and bottom. The lateral boundary surfaces perpendicular to the  $x$  direction are traction free and exposed to the ambient pressure  $p^0$ . As predicted by the Skempton effect (Skempton 1954), a uniform pressure rise will be observed inside the specimen upon the exertion of a force  $2F$  on the rigid plates. Thus, drainage will occur at the side boundaries. As time passes on, pore pressure near these boundaries must dissipate due to drainage access. Later, the pressure depletion region will propagate into the center of the specimen. Fluid drainage will finally stop once the initial pressure rise totally vanishes over the entire domain.

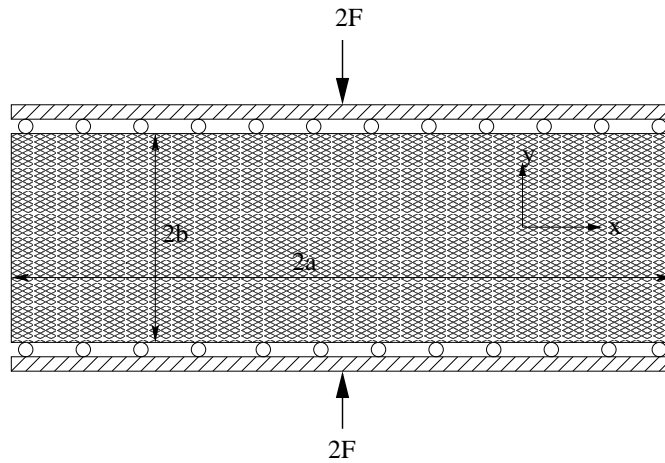


Figure 5.1: Mandel's problem.

Parameter	Value
$a$	100 m
$b$	10 m
$F$	$1.0 \times 10^8 \text{ N}$
$p^0$	0.0 Pa
$E$	$1.0 \times 10^8 \text{ Pa}$
$\nu$	0.2
$\alpha$	1.0
$c_f$	$0.0 \text{ Pa}^{-1}$
$k$	100 md
$\phi$	0.2

Table 5.1: Parameters in Mandel's problem.

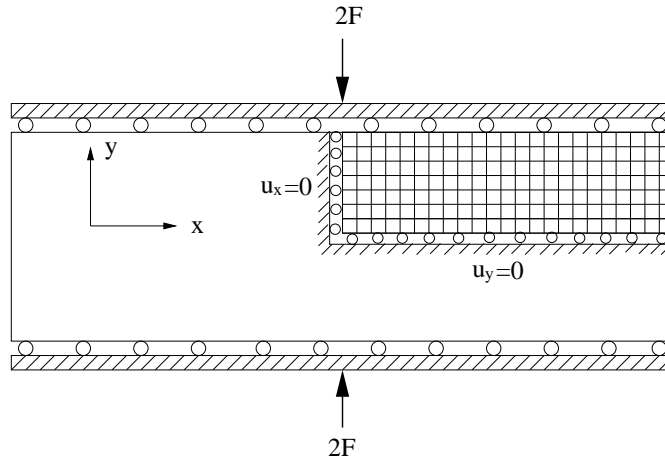


Figure 5.2: Mandel's problem computation domain.

As shown in Figure 5.1, the axis of material rotational symmetry is the  $y$ -axis. A plane strain condition is assumed in the  $z$  direction. Table 5.1 gives the dimensions of the specimen and its material properties as used in this calculation. For a finite element approximation, the 2D problem is solved numerically using our 3D poroelastic model. However, proper boundary conditions must be specified in the  $z$  direction to ensure the plane strain condition. A quarter symmetry feature about the  $x$ - and  $y$ -axes allows us to choose only a quarter of the physical domain as our computation domain, as shown in Figure 5.2. The rigid plate condition is enforced by adding constrained equations such that vertical displacements on the top plate are all equal to an unknown constant value.

The original Mandel's solution (1953) provides only the analytical form for the pore pressure. Later, Abousleiman et al. (1996) extend the solution to all field quantities for materials with transverse isotropy, as well as compressible pore fluids and solid constituents. Their analytical solutions are given as

follows,

$$\begin{aligned}
u_x &= \left[ \frac{F\nu}{2Ga} - \frac{F\nu_u}{Ga} \sum_{i=1}^{\infty} \frac{\sin \beta_i \cos \beta_i}{\beta_i - \sin \beta_i \cos \beta_i} \exp(-\beta_i^2 ct/a^2) \right] x \\
&\quad + \frac{F}{G} \sum_{i=1}^{\infty} \frac{\cos \beta_i}{\beta_i - \sin \beta_i \cos \beta_i} \sin \frac{\beta_i x}{a} \exp(-\beta_i^2 ct/a^2), \\
u_y &= \left[ -\frac{F(1-\nu)}{2Ga} + \frac{F(1-\nu_u)}{Ga} \sum_{i=1}^{\infty} \frac{\sin \beta_i \cos \beta_i}{\beta_i - \sin \beta_i \cos \beta_i} \exp(-\beta_i^2 ct/a^2) \right] y, \\
p &= \frac{2FB(1+\nu_u)}{3a} \sum_{i=1}^{\infty} \frac{\sin \beta_i}{\beta_i - \sin \beta_i \cos \beta_i} \left( \cos \frac{\beta_i x}{a} - \cos \beta_i \right) \exp(-\beta_i^2 ct/a^2), \\
\sigma_{yy} &= -\frac{F}{a} - \frac{2F(\nu_u - \nu)}{a(1-\nu)} \sum_{i=1}^{\infty} \frac{\sin \beta_i}{\beta_i - \sin \beta_i \cos \beta_i} \cos \frac{\beta_i x}{a} \exp(-\beta_i^2 ct/a^2) \\
&\quad + \frac{2F}{a} \sum_{i=1}^{\infty} \frac{\sin \beta_i \cos \beta_i}{\beta_i - \sin \beta_i \cos \beta_i} \exp(-\beta_i^2 ct/a^2), \\
\sigma_{xx} &= \sigma_{xy} = 0.
\end{aligned}$$

In the above equations,  $\beta_i$  satisfies

$$\tan \beta_i = \frac{1-\nu}{\nu_u - \nu} \beta_i,$$

and the other parameters, which are not listed in Table 5.1, are the shear modulus  $G$ , the Skempton pore pressure coefficient  $B$ , the undrained Poisson ratio  $\nu_u$  and the diffusivity coefficient  $c$ . They are defined respectively as (Rice

1976)

$$\begin{aligned}
G &= \frac{E}{2(1 + \nu)}, \\
B &= 1 - \frac{\phi K (K_s - K_f)}{K_f (K_s - K) + \phi K (K_s - K_f)}, \\
\nu_u &= \frac{3\nu + B(1 - 2\nu)(1 - K/K_s)}{3 - B(1 - 2\nu)(1 - K/K_s)}, \\
c &= \frac{2kB^2G(1 - \nu)(1 + \nu_u)^2}{9\mu_f(1 - \nu_u)(\nu_u - \nu)} = \frac{k}{\mu_f S},
\end{aligned}$$

where  $K$ ,  $K_s$  and  $K_f$  are, respectively, the bulk moduli of the solid skeleton, the solid constituent and the pore fluid. In the special case of incompressible fluids and solid grains, the above coefficients are reduced to

$$\begin{aligned}
B &= 1, \\
\nu_u &= 0.5, \\
c &= \frac{2kG(1 - \nu)}{\mu_f(1 - 2\nu)}.
\end{aligned}$$

Numerical results at ten different time steps (see Table 5.2) are computed to compare with the analytical solutions. Figures 5.3—5.6 demonstrate the comparison results for  $p$ ,  $\sigma_{yy}$ ,  $u_x$  and  $u_y$  respectively. In these pictures the distances in the  $x$  direction are normalized by  $a$ . According to the exact solution, at the instant of loading an initial uniform pressure rise,  $\Delta p(x, y, 0^+) = FB(1 + \nu_u)/3a$ , should be observed. The upper plate will instantaneously deform by the amount of  $u_y(x, b, 0^+) = -Fb(1 - \nu_u)/2Ga$ , and, eventually, it will stabilize at  $u_y(x, b, \infty) = -Fb(1 - \nu)/2Ga$ . After the initial outward movement of  $u_x(a, y, 0^+) = F\nu_u/2G$ , the side boundaries will contract

toward the center, and its final state should be  $u_x(a, y, \infty) = F\nu/2G$ . During the entire consolidation scenario both the horizontal stress,  $\sigma_{xx}$ , and the shear stress,  $\sigma_{xy}$ , are identically zero. As shown in Figures 5.3—5.6, all quantities mentioned above have been verified by our numerical results.

$t_i$	Time (seconds)
$t_1$	$1 \times 10^3$
$t_2$	$5 \times 10^3$
$t_3$	$1 \times 10^4$
$t_4$	$1 \times 10^5$
$t_5$	$5 \times 10^5$
$t_6$	$8 \times 10^5$
$t_7$	$1 \times 10^6$
$t_8$	$2 \times 10^6$
$t_9$	$3 \times 10^6$
$t_{10}$	$5 \times 10^6$

Table 5.2: Mandel’s problem output time steps.

The contribution of Mandel’s solution lies not only in the fact that it can be used as a benchmark problem for testing and validating numerical poroelasticity codes but also as a demonstration of non-monotonic pore pressure response under constant boundary conditions. This non-monotonic behavior of pore pressure is known as the Mandel-Cryer effect as Cryer (1963) presents similar results at the center of a sphere under a hydrostatic pressure. Later, this interesting physical phenomenon is confirmed by the results of both laboratory (Gibson et al. 1963 and Verruijt 1965) and field tests (Verruijt 1969). The Mandel-Cryer effect refers to the observation that, after an initial instantaneous rise, pore pressure near the center region continues to increase

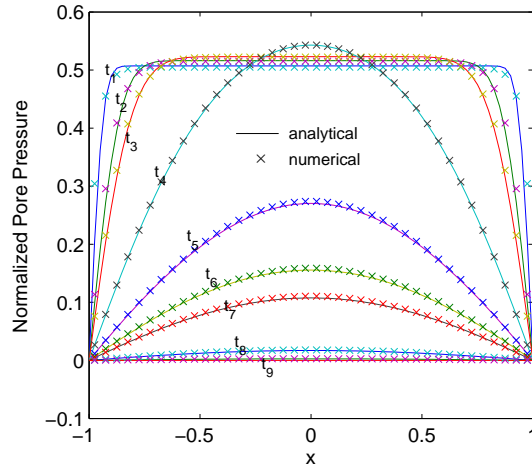


Figure 5.3: Mandel's problem pore pressure distribution along the x-axis.

$$\text{Normalized pore pressure} = ap/F.$$

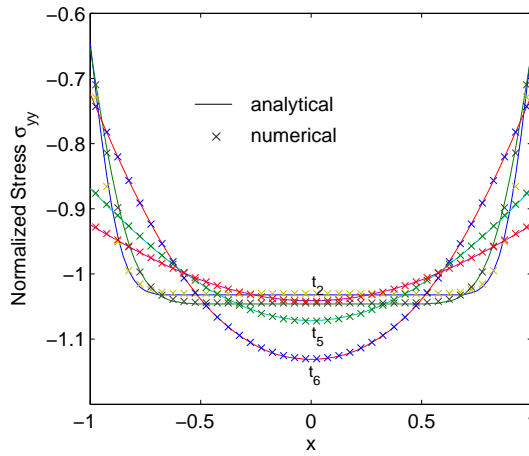


Figure 5.4: Mandel's problem stress ( $\sigma_{yy}$ ) distribution along the x-axis.

$$\text{Normalized stress} = a\sigma_{yy}/F.$$

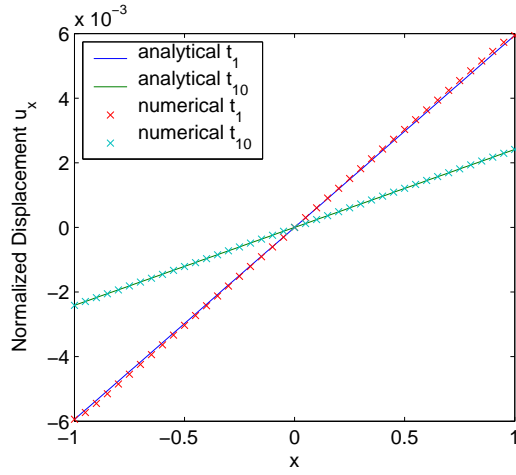


Figure 5.5: Mandel's problem displacement  $u_x$  along the x-axis.

$$\text{Normalized displacement} = u_x/a.$$

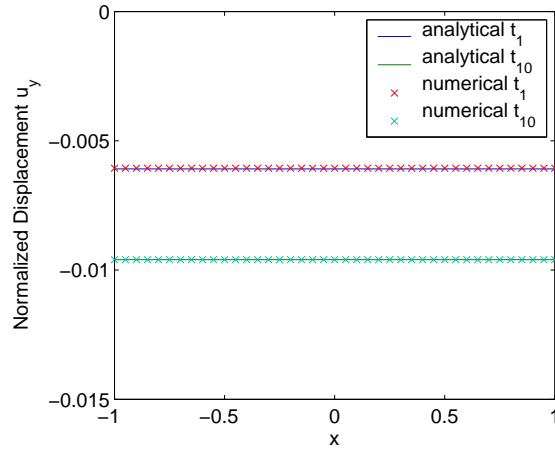


Figure 5.6: Mandel's problem displacement  $u_y$  along the x-axis.

$$\text{Normalized displacement} = u_y/b.$$

for some time before it starts to dissipate. This phenomenon is contrary to what is observed in general free drainage problems. The non-monotonic pore pressure response shown in Figure 5.7 is due to the loading condition through a rigid plate. As pressure depletion continues near the drainage boundaries the material is effectively softened around those areas. Since the loading is applied through a rigid plate there is a gradual load transfer of compressive total stress to the stiffer center region. The transferred load serves as a source term in the porous flow diffusivity equation. As a result, the pressure build-up at the center region will last for a while before the entire domain starts to deplete.

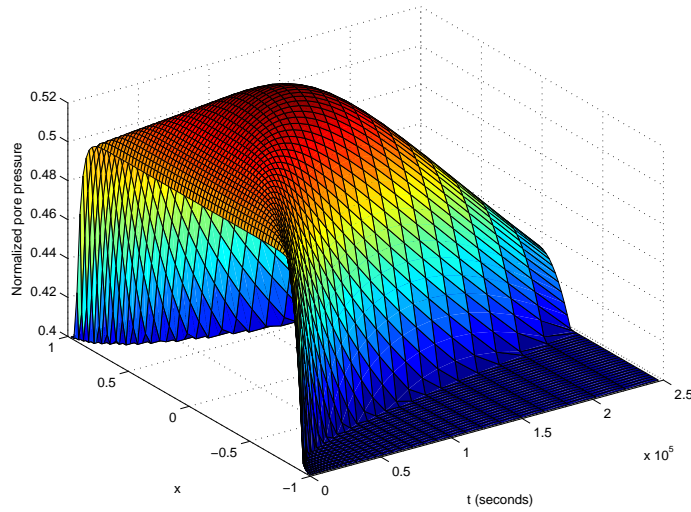


Figure 5.7: Mandel's problem pore pressure history along the x-axis.

## 5.2 A 3D Confined Reservoir with Non-pay Rocks

The numerical example described in Section 4.4.1.3 is used here to test our 3D poroelasticity code. Numerical results are verified using the ACRES simulator (ARCO Reservoir Simulator Development 1999). A fully coupled technique is implemented in the ACRES's program.

In Section 2.3 we mention briefly the effect of reservoir boundary conditions on the pressure and stress evolutions inside the reservoir. However, it is very difficult, if not impossible, to accurately define these changing boundary conditions before solving the coupled equations. One solution is to incorporate an outer domain surrounding the reservoir. Gutierrez & Lewis (1998) and Dean et al. (2003) construct numerical examples that take into account the exterior non-pays. Their numerical results also demonstrate a non-monotonic pressure response at the reservoir flanks. In this section we consider Dean's problem first to validate our 3D poroelastic model, then to demonstrate the effect of material properties in the outer domain on surface subsidence and reservoir compaction.

The reservoir geometry is depicted schematically in Figure 4.3. The FE discretization is shown in Figure 4.4. A description of the reservoir properties, the initial and boundary conditions, as well as the well configurations, can be found in Section 4.4.1.3.

Table 5.3 presents the numerical values for pressure, displacement, stress and strain at the end of 4000 days. Pore pressure histories at cell

(6,11,6) and cell (11,11,6) are plotted in Figure 5.8 and 5.9 respectively. It appears that our iterative coupling technique nearly reproduces the results of the fully coupled scheme in ACRES.

	Acres	IPARS
Avg. reservoir pressure	3435.3 psi	3435.6 psi
Pressure (6,11,6)	3506.1 psi	3506.4 psi
Pressure (11,11,8)	2690.7 psi	2690.8 psi
$u_x(6,11,1)$ , surface	1.02 ft	1.02 ft
$u_y(11,6,1)$ , surface	1.15 ft	1.15 ft
$u_z(11,11,1)$ , surface	4.04 ft	4.03 ft
$u_z(11,11,6)$ , top of reservoir	1.15 ft	1.15 ft
$\varepsilon_{xx}(11,11,8)$	0.08%	0.08%
$\varepsilon_{yy}(11,11,8)$	0.13%	0.13%
$\varepsilon_{zz}(11,11,8)$	3.13%	3.13%
$\sigma_{xx}(11,11,8)$	3395.4 psi	3395.5 psi
$\sigma_{yy}(11,11,8)$	3399.3 psi	3399.4 psi
$\sigma_{zz}(11,11,8)$	8635.0 psi	8635.9 psi

Table 5.3: Results of the 3D problem at the end of 4000 days.

A non-monotonic pressure change similar to the Mandel-Cryer effect is observed at the reservoir boundary cell (6,11,6), as shown in Figure 5.8. As the reservoir depletion proceeds the pore pressure at cell (6,11,6) is not monotonically decreasing, but continues to increase for about 200 days. This is a distinctive feature of a coupled analysis. The time evolution of the Mandel-Cryer zone can be seen more clearly in Figures 5.10 and 5.11.

Gutierrez & Lewis (1998) discuss the effect of reservoir material properties on surface subsidence and reservoir compaction. In the following we

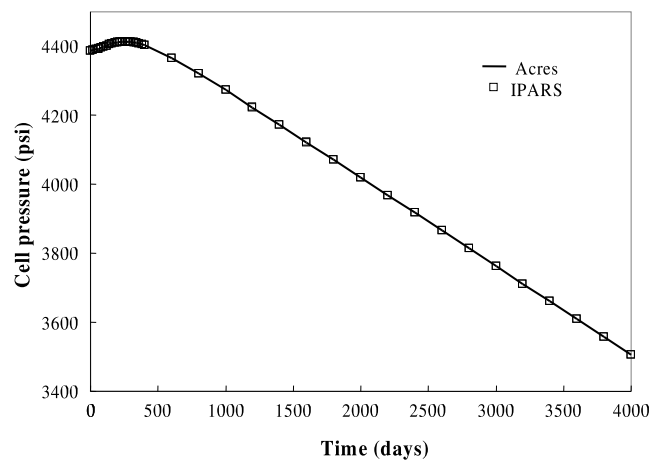


Figure 5.8: Pore pressure history at cell (6, 11, 6) for the 3D problem.

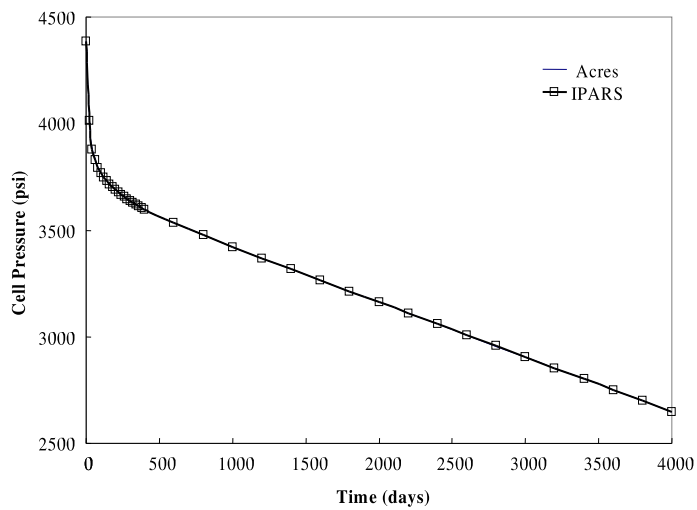


Figure 5.9: Pore pressure history at cell (11, 11, 6) for the 3D problem.

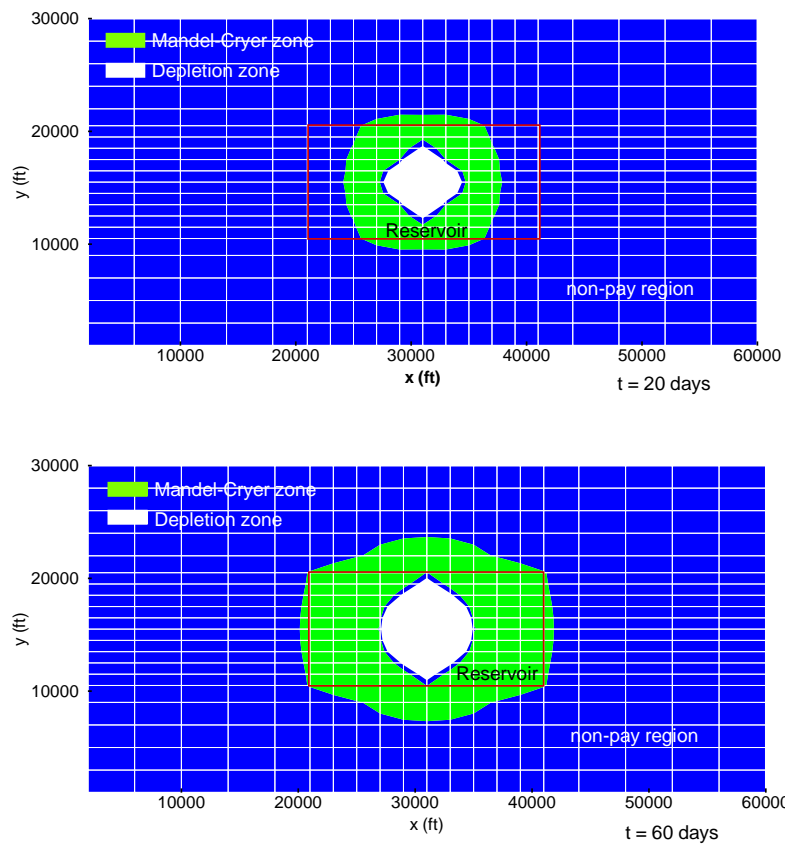


Figure 5.10: Pressure Mandel-Cryer zones at the end of 20 and 60 days for the 3D problem.

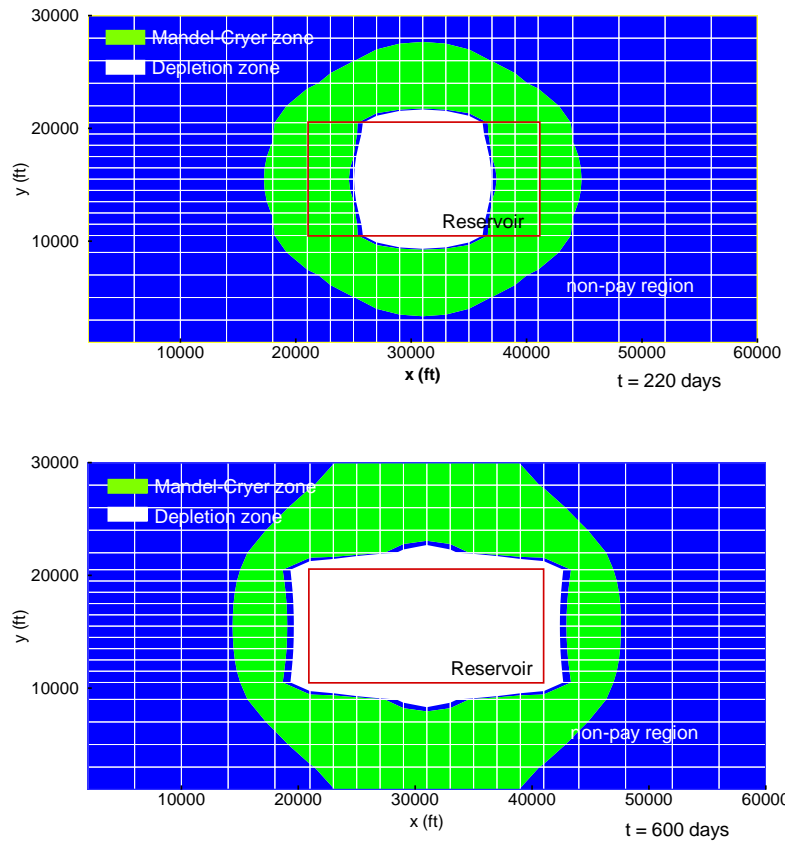


Figure 5.11: Pressure Mandel-Cryer zones at the end of 220 and 600 days for the 3D problem.

investigate the effect of the material properties of the surrounding non-pays on surface subsidence and reservoir compaction. Three runs are conducted for the purpose of comparison. Parameters used in each run are shown in Table 5.4.

Comparison runs		Parameters	
		Reservoir	Non-pay
Uncoupled case		$c_r = 3.3 \times 10^{-4} \text{psi}^{-1}$	$c_r = 3.3 \times 10^{-6} \text{psi}^{-1}$
Coupled case	Stiff non-pay	$E = 1.0 \times 10^4 \text{psi}$	$E = 1.0 \times 10^6 \text{psi}$
	Soft non-pay	$E = 1.0 \times 10^4 \text{psi}$	$E = 1.0 \times 10^4 \text{psi}$

Table 5.4: Parameters used in comparison runs for the 3D problem with surrounding non-pay rocks.

Significant differences in the average reservoir pressure are observed in Figure 5.12. Soft non-pays result in a larger reservoir compaction, but a minor surface subsidence as compared with stiff non-pays (see Figure 5.13). The non-monotonic pressure evolution is observed only in the reservoir with stiffer non-pays, as shown in Figure 5.14.

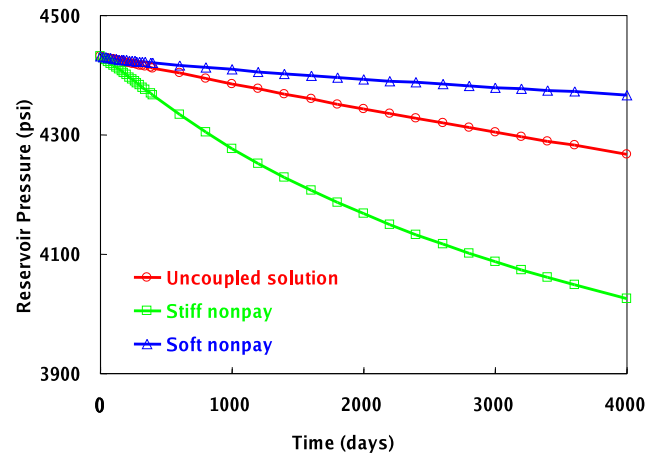


Figure 5.12: Comparison of average reservoir pressure history.

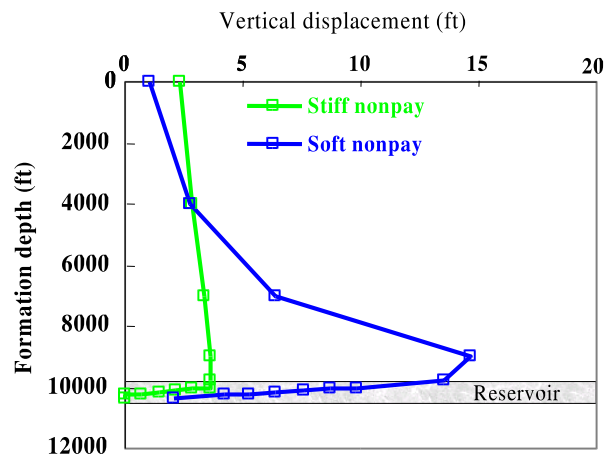


Figure 5.13: Comparison of vertical displacements from cell (11,11,1) to cell (11,11,12).

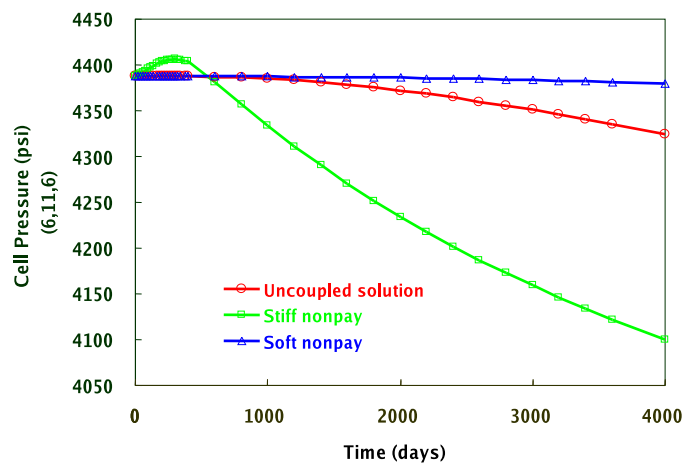


Figure 5.14: Comparison of pore pressure history at boundary cell (6,11,6).

## Chapter 6

### Linear Solvers for the Poroelastic Model

The convergence of the iterative coupling technique discussed in Chapter 4 depends crucially on the accuracy of the displacement solutions. However, solving the elasticity system frequently poses a computational bottleneck in coupled simulations. First, the Galerkin finite element approximation for the elasticity equations generates an algebraic linear system that is substantially larger than the 7-point stencil in the CCFD formulation of multiphase flow equations. Secondly, to characterize the interactions between a reservoir and its surroundings, the computational domain needs to be extended to the overburden, underburden and sideburden. Numerical experiments indicate that more than 70 percent of the total CPU time for a coupled simulation is spent on the solution of displacement. In summary, we are solving a coupled system in a large physical domain as generally preferred on a full-field scale with great vertical depth. Such a coupled analysis involves a large linear system with millions of unknowns, whose solution needs to be computed on parallel machines.

In this chapter we discuss linear solution techniques for a discrete elasticity system. In particular, domain decomposition preconditioners with a

super-coarsening multigrid (SCMG) and displacement decomposition (DiD) preconditioners are presented. Their performance is evaluated based on numerical experiments on a single processor machine. Parallel implementation issues will be addressed in the next chapter.

## **6.1 Krylov Linear Iterative Solvers: GMRES and BiCG-STAB**

The Krylov iterative solvers chosen for the solution of displacement include the preconditioned conjugate gradient (PCG), the right preconditioned generalized minimal residual (GMRES) (Saad and Schultz, 1986) and the bi-conjugate gradient stabilized (BiCG-STAB) method (Van Der Vorst, 1992). PCG is a popular technique for solving large scale symmetric positive definite linear systems. But it may fail to converge with non-symmetric preconditionings. GMRES is known to be the most robust Krylov subspace method for solving non-symmetric systems, or symmetric systems with non-symmetric preconditioning matrices. A major drawback of the GMRES method is the large memory requirement for storing all the Krylov spaces. A restart version of GMRES can alleviate these intensive memory demands. BiCG-STAB requires less memory storage, but it involves two preconditioning steps and two matrix-vector products in one iteration. Thus, it expends more CPU time per iteration than the GMRES method.

Table 6.1 shows the comparison results for the GMRES and the BiCG-STAB methods. Linear solver iterations are terminated when the initial resid-

	GMRES			BiCG-STAB		
Preconditioner	IC(0)	IC(1)	RB-GS	IC(0)	IC(1)	RB-GS
Iteration	6025	4535	3577	4272	3332	5279
Time (seconds)	516.8	554.9	811.1	563.5	656.6	942.4

Table 6.1: Total numbers of linear solver iterations and CPU times in seconds for GMRES and BiCG-STAB methods.

ual ( $l_2$ norm) is decreased by a factor of  $10^6$ . CPU times are measured in seconds on a 1.2 GHz Athlon Linux machine with 512 MB RAM memory. We use an incomplete Choleksy factorization (IC) and a 4-color line Gauss-Seidel as preconditioners. In all cases, GMRES converges faster than BiCG-STAB. Thus, in the following discussion, all numerical results are obtained using the GMRES algorithm unless stated otherwise.

## 6.2 Domain Decomposition Preconditioners

The widespread availability of parallel computers and their potential for the numerical solution of partial differential equations has led to a large amount of research in domain decomposition (DD) methods. DD methods are generally flexible methods for the solution of linear or nonlinear partial differential equations (PDEs). For linear problems, DD methods can often be used as preconditioners for Krylov subspace acceleration techniques, such as PCG and GMRES methods. For details about DD methods we refer the reader to the book by Smith, Bjorstad and Gropp (1996)

The advantage of applying DD-type preconditioners lies in their ease of

parallelization and their satisfactory parallel performance. In the next three sections we employ the general framework of space decomposition and subspace correction to present several one-level and multilevel DD preconditioners.

### 6.2.1 One-level Overlapping DD Preconditioners

We first consider a finite element discretization of a boundary value problem in a domain  $\Omega$ . Let  $\mathcal{T}_h(\Omega)$  be a non-degenerate, quasi-uniform finite element partition of  $\Omega$  that defines the finite dimensional subspace  $V = V_h \subset \mathcal{V}$ . The resulting linear system takes the form of

$$Au = b. \quad (6.1)$$

Here,  $A$  is the discrete elasticity operator, and  $u$  is the displacement vector. We also assume that  $\Omega$  is decomposed into a set of overlapping subdomains  $\{\Omega_i\}_{i=1}^p$ . One way of defining the subdomains and the associate partition is by starting with disjoint open sets  $\{\Omega_i^0\}_{i=1}^p$  of size  $h_0$  with  $\bar{\Omega} = \cup_{i=1}^p \bar{\Omega}_i^0$ , where  $\bar{\Omega} = \Omega \cup \partial\Omega$  and  $\bar{\Omega}_i^0 = \Omega_i^0 \cup (\partial\Omega_i^0 \cap \partial\Omega)$ . Then the subdomain  $\Omega_i$  is defined to be a subdomain containing  $\Omega_i^0$  with the distance from  $\partial\Omega_i \cap \Omega$  to  $\Omega_i^0$  greater than or equal to  $ch_0$  for some prescribed constant  $c$  called the number of overlapping. For simplicity, we further assume that each subdomain  $\Omega_i$  can be represented as a union of a group of finite elements from  $\mathcal{T}_h(\Omega)$ . Then the division of  $\Omega$  results in a decomposition of the space  $V$  into a number of finite element subspaces  $V_i$  ( $i = 1, \dots, p$ ), i.e.,

$$V = V_1 + \dots + V_p, \quad (6.2)$$

where

$$V_i = \{v \in V : v(x) = 0, \forall x \in \Omega \setminus \Omega_i\}.$$

The above domain decomposition (6.2) allows us to construct different iterative schemes (additive, multiplicative and hybrid Schwarz methods) by correcting residuals in these subspaces. The goal is to divide a large global problem into a number of smaller, local ones. In solving the linear system (6.1) a DD method will start from an old approximation  $u^n$  to compute a new approximation  $u^{n+1}$  by

$$u^{n+1} = u^n + e \quad (6.3)$$

where  $e$  is the subdomain correction.

In this implementation, a domain is always divided in the horizontal directions ( $x$  and  $y$ ) with one overlap ( $c = 1$ ). A subdomain problem can be solved by a direct method or by one sweep of an incomplete Cholesky factorization (IC). Three one-level DD preconditioners have been implemented, namely, an additive Schwarz with a direct subdomain solver (DDa-Direct), an additive Schwarz with an incomplete Cholesky factorization (DDa-IC), and a 4-color line Gauss-Seidel (DDm-4color-LineGS).

---

**Algorithm 1** DDa-Direct

---

- 1: Compute residual:  $r^n = b - Au^n$ .
  - 2: Restrict and solve the subdomain corrections directly:  $e_i^n = A_i^{-1} R_i r^n$ , for  $i = 1, \dots, p$ .
  - 3: Extend and update solution:  $u^{n+1} = u^n + R_i^T e_i^n$ .
- 

In Algorithm 1  $R_i$  is the restriction operator ( $R_i : V \rightarrow V_i$ ) while  $R_i^T$  is the prolongation operator ( $R_i^T : V_i \rightarrow V$ );  $A_i$  is the restriction of  $A$  to the

subspace  $V_i$ , which is computed by

$$A_i = R_i^T A R_i.$$

Algorithm 1 can be written in a one-step form as,

$$u^{n+1} = u^n + M_{DDa-Direct} (b - Au^n),$$

where the preconditioner  $M_{DDa-Direct}$  is given by

$$M_{DDa-Direct} = \sum_{i=1}^p R_i^T A_i^{-1} R_i. \quad (6.4)$$

---

**Algorithm 2** DDa-IC

---

- 1: Compute residual:  $r^n = b - Au^n$ .
  - 2: Restrict and solve the subdomain correction approximately by incomplete Cholesky factorization:  $e_i^n = \tilde{A}_{i,IC(k)}^{-1} R_i r^n$ , for  $i = 1, \dots, p$
  - 3: Extend and update solution:  $u^{n+1} = u^n + R_i^T e_i^n$ .
- 

Here  $\tilde{A}_{i,IC(k)}^{-1}$  represents the application of an incomplete Cholesky factorization in subdomain  $\Omega_i$  with a fill-in level of  $k$ . The one step representation of Algorithm 2 is

$$u^{n+1} = u^n + M_{DDa-IC(k)} (b - Au^n),$$

where

$$M_{DDa-IC(k)} = \sum_{i=1}^p R_i^T \tilde{A}_{i,IC(k)}^{-1} R_i. \quad (6.5)$$

Similar to the classical convergence results for Jacobi and Gauss-Seidel methods the multiplicative Schwarz preconditioners have a faster numerical

convergence rate than the additive Schwarz methods. However, they have little potential for parallelism. The problem can be overcome by the general idea of multicoloring. In multicoloring, each subdomain is associated with a color such that two subdomains having the same color do not share common grid points. Thus, solutions can be updated simultaneously (in parallel) for all the subdomains of the same color. In reservoir simulation a subdomain is generally taken to be a vertical line. A 4-color line Gauss-Seidel algorithm (DDm-4Color-LineGS) is defined as follows.

---

**Algorithm 3** DDm-4Color-LineGS

---

**for**  $i = color_1, \dots, color_p$  **do**

    Compute residual:  $r = b - Au^{n+(i-1)/p}$ .

    Restrict and directly solve the lines associated with color  $i$ :

$$e_i = \sum_{j \in color_i} R_j^T A_j^{-1} R_j r.$$

    Extend and update solution:  $u^{n+i/p} = u^{n+(i-1)/p} + e_i$ .

**end for**

---

The above preconditioner can be written as

$$M_{DDm-LineGS} = \left[ I - \prod_{i=color_1}^{color_p} (I - B_i A) \right] A^{-1}, \quad (6.6)$$

where

$$B_i = \sum_{j \in color_i} R_j^T A_j^{-1} R_j.$$

Multicolor Gauss-Seidel is frequently used as a preconditioner for the Krylov subspace methods. But it is also common to use the method together

with a one-level additive Schwarz preconditioner in a parallel implementation. The resulting preconditioner is essentially a hybrid scheme that is given by:

$$M_{DDh} = M_{DDa-Direct} + M_{DDm-LineGS}(I - AM_{DDa-Direct}), \quad (6.7)$$

or

$$M_{DDh} = M_{DDa-IC(k)} + M_{DDm-LineGS}(I - AM_{DDa-IC(k)}), \quad (6.8)$$

where  $M_{DDa-Direct}$ ,  $M_{DDa-IC(k)}$  and  $M_{DDm-LineGS}$  are respectively defined in (6.4), (6.5) and (6.6).

### 6.2.2 Multilevel Domain Decomposition Preconditioners

The efficiency of single level methods deteriorates with increasing numbers of subdomains. Table 6.2 shows the numerical results of a test case with a  $16 \times 16 \times 8$  grid discretization. The entire domain is divided into a varying numbers of subdomains. An additive Schwarz preconditioner,  $M_{DDa-Direct}$ , is applied. Linear solver iterations are terminated when the initial residual ( $l_2$  norm) is reduced by a factor of  $10^5$ . Clearly, GMRES takes more iterations to converge as the number of subdomains increases from 4 to 256. To overcome this problem we need a mechanism for the global communication of information in each iteration. The best known of these techniques are the multilevel or multigrid methods. Among the extensive multigrid literature, we refer the reader to the book by Hackbusch (1985) and Trottenberg et al. (2001). The following discussion is organized as follows: Section 6.2.2.1 gives a brief introduction to the multigrid methods; a description of the super-coarsening

multigrid (SCMG) algorithm for the elasticity system follows; several multilevel schemes using the SCMG and the one-level Schwarz preconditioners, (6.4), (6.5), and (6.6), are presented in Section 6.2.2.3.

Total Number of Subdomains	Subdomain Size $n_x \times n_y \times n_z$	DDa-Direct Iterations
4	$8 \times 8 \times 8$	1520
16	$4 \times 4 \times 8$	2035
32	$4 \times 2 \times 8$	3277
64	$2 \times 2 \times 8$	3448
256	$1 \times 1 \times 8$	4336

Table 6.2: Total number of (GMRES) linear iterations for different domain decompositions.

#### 6.2.2.1 A Multigrid Introduction

Multigrid is motivated by the observation that standard iterative methods, like Jacobi and Gauss-Seidel, are effective at eliminating the high-frequency or oscillatory components of the error but are ineffective in reducing the low frequency or smooth components. Another important observation is that smooth modes on a fine grid appear to be less smooth on coarser grids. This suggests that solutions on the fine grid may be projected to the coarse grid such that low frequency or smooth contents of the error can be reduced by the same iterative scheme as used on the fine grid. Multigrid methods are well-known for their high efficiency and the low arithmetic complexity of  $O(n)$  or, at most,  $O(n \log(n))$ .

In the context of subspace corrections, a multigrid algorithm can be

defined using the coarse grid function spaces. Consider a nested sequence of gradually coarsened grids,  $\Omega_L \subset \dots \subset \Omega_2 \subset \Omega_1 = \Omega$ . Corresponding to each grid  $\Omega_m$ , a finite element space  $\mathcal{M}_m$  is defined such that

$$\mathcal{M}_L \subset \dots \subset \mathcal{M}_2 \subset \mathcal{M}_1 = V.$$

Each space  $\mathcal{M}_m$  is equipped with an inner product  $(\cdot, \cdot)_m$  and a coarse operator  $A_m$  that defines a second inner product  $(\cdot, \cdot)_{A_m} = (A_m \cdot, \cdot)_m$ . A conventional multigrid method requires three types of operators:

1. Grid transfer operators, namely, restriction operators  $(I_m^{m+1})$  and prolongation operators  $(I_{m+1}^m)$ ;
2. Coarse grid operator  $A_m$ ;
3. Cheap iterative solvers that are effective at reducing the high frequency errors on each grid.

Generally, operator  $A_m$  may be formed in one of two ways. We can either algebraically construct a Galerkin coarse grid matrix by  $I_{m-1}^m A_{m-1} I_m^{m-1}$ , or create a new finite element problem on each coarse grid, thereby allowing the finite element implementation to construct the matrix. Using the above components we construct the following multigrid V-cycle scheme.

Several authors have investigated varying combinations of multigrid and the conjugate gradient methods. Keller (1982) and Kettler and Meijerink (1981) use multigrids as preconditioners for the PCG solver while Bank &

---

**Algorithm 4**  $\text{MGV}(A_m, b_m)$ 

---

```
if there is a coarser grid  $m + 1$ , i.e.,  $m < L$  then
    Relax  $k_1$  times on  $A_m u_m = b_m$ .
    Perform coarse grid correction:
    {
        Set  $u_{m+1} = 0$ .
        Compute  $b_{m+1} = I_m^{m+1}(b_m - A_m u_m)$ .
        Recursively apply MGV on level  $m + 1$  :  $u_{m+1} =$ 
         $\text{MGV}(I_m^{m+1} A_m I_{m+1}^m, b_{m+1})$ .
        Correct the solution by  $u_m := u_m + I_{m+1}^m u_{m+1}$ .
        Relax  $k_2$  times on  $A_m u_m = b_m$ .
    }
else
     $u_L = A_L^{-1} b_L$ .
    return  $u_m$ .
end if
```

---

Douglas (1985) treat PCG as a relaxation method for the multigrid solver. Braess (1986) considers these two combinations and reports that the conjugate gradient method, with multigrid preconditioning, is effective for elasticity problems. Here we propose a multilevel preconditioner for the GMRES and PCG type solvers. The method obtains coarse level corrections using a super-coarsening multigrid algorithm. Due to the low arithmetical complexity of SCMG, the resulting preconditioners are very efficient.

### 6.2.2.2 Super-coarsening Multigrid Preconditioner

In the reservoir simulation community, the first attempt to apply the super-coarsening type of preconditioners was by Watts (1971) for solving 2D elliptic problems with Neumann boundary conditions. In his work, a one-

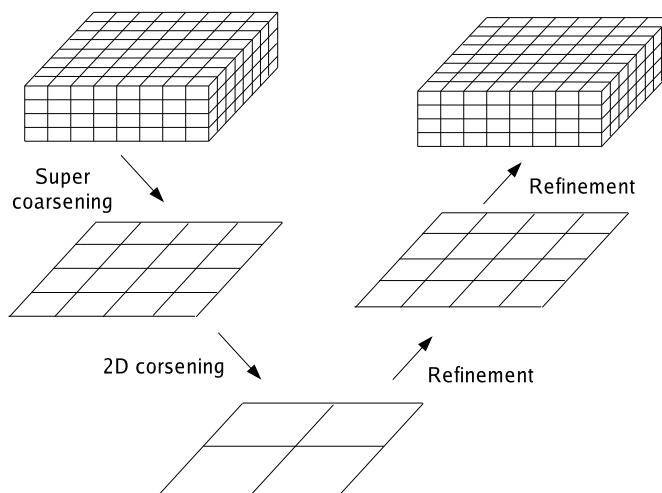


Figure 6.1: Super-coarsening multigrid preconditioner.

dimensional residual correction is applied after each line-SOR (LSOR) sweep to dampen the low frequency components. Later, the one dimensional correction is generalized to two dimensions using the constrained residual technique (Wallis, 1985). This method generates subspaces by aggregating all system equations along a certain direction, solves the reduced system in a lower dimension, and then projects the solution back to the original grid. It is recommended that the 2D problem is solved by either a direct solver or an iterative scheme. However, such a reduced problem may still be too large to solve efficiently. Killough and Wheeler (1985) implement a 2D correction scheme to improve the performance of a 3D domain decomposition preconditioner. To reduce the effect of aggregation direction on linear solver convergence, residual

corrections in the three directions are computed alternatively. The aggregate problem is solved very roughly though. A parallel version of the 2D correction for a 3D LSOR method is implemented by Wheeler and Smith in 1989 (Wheeler and Smith, 1989). They augment the original method by adding a second level 2D coarse grid that is chosen to be sufficiently coarse so that direct methods may be applied. The two-level scheme is later generalized to a multigrid V-cycle by Lacroix et al (2000) for multiphase flow problems. Based on the theoretical analysis of Watts (1973), and justified by the observation that in most reservoir applications the vertical depth of the domain is generally much smaller than the horizontal extent, collapsing is, thus, conducted only in the vertical direction.

In this work the SCMG method is extended to the algebraic linear system arising from the finite element discretization of elasticity equations. The newly developed modules are called immediately after a one-level DD preconditioner ( $M_{DDa-Direct}$  in (6.4),  $M_{DDa-IC(k)}$  in (6.5) and  $M_{DDm-LineGS}$  in (6.6)) for the coarse level corrections. Numerical experiments show that the 2D correction is not accurate in the case of strongly heterogeneous layers. However, it is capable of dampening the low frequency aggregated components which is the main purpose of applying SCMG.

In the SCMG method, a 3D grid  $\Omega_1$  is first super-coarsened to a 2D grid  $\Omega_2$ . The coarsening from  $\Omega_1$  to  $\Omega_2$  is conducted in two steps: 1) perform aggregation along each vertical line to form a collapsed 2D grid; 2) conduct standard  $2 \times 2$  coarsening to generate  $\Omega_2$ . Letting  $v^2 = I_1^2 v^1$ , the restriction

operator may be written as

$$\begin{aligned}
v_{i,j}^2 &= \frac{1}{16} \sum_{k=1}^{n_z} \left[ v_{2i-1,2j-1}^1 + v_{2i-1,2j+1}^1 + v_{2i+1,2j-1}^1 + v_{2i+1,2j+1}^1 \right. \\
&\quad \left. + 2 \left( v_{2i,2j-1}^1 + v_{2i,2j+1}^1 + v_{2i-1,2j}^1 + v_{2i+1,2j}^1 \right) + 4v_{2i,2j}^1 \right], \\
1 \leq i \leq \frac{n_x}{2} - 1, \quad 1 \leq j \leq \frac{n_y}{2} - 1,
\end{aligned}$$

where  $n_x$ ,  $n_y$  and  $n_z$  are respectively the number of grid points in the  $x$ ,  $y$  and  $z$  directions. On other coarser level grids ( $\Omega_m$ ,  $m > 2$ ) the components of the SCMG algorithm are defined as follows:

1. Full weighting restriction operator. In a stencil notation, it reads:

$$I_m^{m+1} = \frac{1}{16} \left[ \begin{array}{ccc} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{array} \right]_h^{2h}.$$

2. Bilinear interpolation operator mapping corrections from a  $2h$ -grid to an  $h$ -grid. It is related to the restriction operator by

$$I_{m+1}^m = (I_m^{m+1})^T.$$

3. Galerkin coarse grid operator  $A_m$ . It is computed by

$$A_m = I_{m-1}^m A_{m-1} I_m^{m-1}.$$

4. 4-color Gauss-Seidel (point GS for 2D and line GS for 3D) is used for pre- and post-smoothing on each level.

---

**Algorithm 5** SCMG( $A_m, b_m$ )

---

**if** there is a coarser grid  $m + 1$ , i.e.,  $m < L$  **then**  
  Perform  $k_1$  line GS relaxations:  $u_m := u_m + M_{DDm-LineGS}(b_m - A_m u_m)$   
  Apply coarse grid correction:  
  {  
    Set  $u_{m+1} = 0$ .  
    Compute  $b_{m+1} = I_m^{m+1}(b_m - A_m u_m)$ .  
    Apply SCMG on level  $m + 1$ :  $u_{m+1} = SCMG(I_m^{m+1} A_m I_m^m, b_{m+1})$ .  
  }  
  Perform  $k_2$  line GS smoothing:  $u_m := u_m + M_{DDm-LineGS}(b_m - A_m u_m)$ .  
**else**  
   $u_L = A_L^{-1} b_L$  or  $u_L := u_L + M_{DDm-LineGS}(b_L - A_L u_L)$ .  
  return  $u_m$ .  
**end if**

---

Using these components we construct an SCMG scheme as described in Algorithm 5.

According to our experience with the SCMG method as applied to multiphase flow equations (Lacroix, Vassilevski and Wheeler 2001), this algorithm features:

1. Low arithmetic complexity due to the reduction of unknowns by super-coarsening;
2. Faster convergence rate for reservoirs with moderate thickness and heterogeneity in rock layers;
3. Dependence of convergence on mesh sizes in horizontal directions;
4. Sensitivity (not very strong) to coefficient jumps.

### 6.2.2.3 Multilevel Preconditioners

As shown by the numerical results in Section 6.4, the performance of the SCMG method deteriorates with stronger reservoir heterogeneity. This ineffectiveness is caused by the fact that a 3D operator is corrected by 2D solutions. However, the loss of efficiency may be ameliorated by applying strong pre-smoothings on the first level grid  $\Omega_1$ . Using the one-level DD preconditioners presented in Section 6.2.1, we construct two multilevel preconditioners, namely, MLDD-Direct/SCMG in Algorithm 6 and MLDD-IC/SCMG in Algorithm 7, where MLDD stands for a multilevel method while Direct and IC denote the type of subdomain solvers on  $\Omega_1$ . We write the iteration schemes in Algorithms 6 and 7 in a compact form as follows,

$$u^{n+1} = u^n + M_{MLDD-Direct/SCMG}(b - Au^n), \quad (6.9)$$

$$u^{n+1} = u^n + M_{MLDD-IC(k)/SCMG}(b - Au^n), \quad (6.10)$$

where the multilevel preconditioners are respectively given by

$$M_{MLDD-Direct/SCMG} = M_{DDa-Direct} + M_{SCMG}(I - AM_{DDa-Direct}), \quad (6.11)$$

$$M_{MLDD-IC(k)/SCMG} = M_{DDa-IC(k)} + M_{SCMG}(I - AM_{DDa-IC(k)}). \quad (6.12)$$

Using Algorithm 6 as a preconditioner for the BiCG-STAB solver we reran the problem presented in Section 6.2.2. Numerical results are shown in Table 6.3 for comparison. Clearly, we observe substantial reductions in the linear iteration counts by applying SCMG. The deterioration of the one-level schemes with increasing numbers of subdomains is also alleviated by the

---

**Algorithm 6** MLDD-Direct/SCMG

---

1: Solve  $Au = b$  approximately by DDa-Direct in Algorithm 1, i.e.,

$$u^{n+1/2} = u^n + M_{DDa-Direct}(b - Au^n).$$

2: Correct  $u^1$  by SCMG

$$u^{n+1} = u^{n+1/2} + M_{SCMG}(b - Au^{n+1/2}).$$

---

---

**Algorithm 7** MLDD-IC/SCMG

---

1: Solve  $Au = b$  approximately by DDa-IC(k) in Algorithm 2, i.e.,

$$u^{n+1/2} = u^n + M_{DDa-IC(k)}(b - Au^n).$$

2: Correct  $u^1$  by SCMG

$$u^{n+1} = u^{n+1/2} + M_{SCMG}(b - Au^{n+1/2}).$$

---

Total number of subdomains	DDa-Direct		MLDD-Direct/SCMG	
	Iterations	Time (sec.)	Iterations	Time (sec.)
4	1520	172.9	454	112.4
16	2035	121.8	531	83.3
32	3277	152.1	580	87.1
64	3448	121.1	602	68.4
256	4336	118.4	746	68.6
CG	Takes 12891 iterations and 249.1 seconds.			

Table 6.3: Total number of linear solver iterations and CPU time for one-level and multilevel DD preconditioners.

global coarse level corrections. More numerical experiments with the multilevel preconditioners may be found in Section 6.4.

### 6.3 Displacement Decomposition Preconditioner (DiD)

Another important type of space decomposition method for solving solid mechanics problems is the displacement decomposition (DiD) method. We may simply describe it as a block diagonal preconditioner for the stiffness matrix  $A$ . Each block or subsystem corresponds to a separate displacement component. Axelsson and Gustafsson (1978) and Blaheta (1994) present theoretical analysis of the DiD preconditioners. In this section we shall investigate the application of the DiD methods to the algebraic linear system arising from the FE discretization of elasticity equations.

#### 6.3.1 Displacement Decomposition

To describe the DiD algorithms, we rewrite the weak formulation of the poroelasticity equation (3.11) here as

$$a(u, v) = f(v), \quad \forall v \in V, \quad (6.13)$$

where  $f(v)$  includes the initial conditions, boundary conditions and pore pressures. The bilinear form  $a(u, v)$  is given by

$$\begin{aligned}
a(u, v) &= \int_{\Omega} \left[ (\lambda + 2\mu) \sum_i \frac{\partial u_i}{\partial x_i} \frac{\partial v_i}{\partial x_i} \right] d\Omega \\
&+ \int_{\Omega} \left[ \sum_i \sum_{j \neq i} \mu \frac{\partial u_i}{\partial x_j} \frac{\partial v_i}{\partial x_j} \right] d\Omega \\
&+ \int_{\Omega} \left[ (\lambda + \mu) \left( \frac{\partial u_j}{\partial x_j} \frac{\partial v_i}{\partial x_i} + \frac{\partial u_i}{\partial x_j} \frac{\partial v_j}{\partial x_i} \right) \right] d\Omega. \tag{6.14}
\end{aligned}$$

Discretizing (6.13) in the finite element space  $V_h$  leads to the symmetric positive definite linear system (6.1) whose solution represents the nodal displacements on a discretized grid. If displacement components are ordered separately the assembled matrix  $\mathbf{A}$  takes the form of a block structure, and the linear system (6.1) is rewritten as

$$\begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{12}^T & A_{22} & A_{23} \\ A_{13}^T & A_{23}^T & A_{33} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}.$$

Thus, a straightforward block diagonal preconditioner may be constructed as

$$B = \begin{bmatrix} A_{11} & & \\ & A_{22} & \\ & & A_{33} \end{bmatrix},$$

where each block  $A_{ii}$  is obtained by the FE discretization of a reduced form of  $a(u, v)$  in (6.14), i.e.,

$$\hat{a}(u, v) = \int_{\Omega} \left\{ (\lambda + 2\mu) \sum_i \frac{\partial u_i}{\partial x_i} \frac{\partial v_i}{\partial x_i} + \sum_i \sum_{j \neq i} \mu \frac{\partial u_i}{\partial x_j} \frac{\partial v_i}{\partial x_j} \right\} d\Omega. \tag{6.15}$$

The auxiliary Laplacian equations are

$$\begin{aligned} -(\lambda + 2\mu)\frac{\partial^2 u_1}{\partial x_1^2} - \mu\frac{\partial^2 u_1}{\partial x_2^2} - \mu\frac{\partial^2 u_1}{\partial x_3^2} &= f_1, \\ -\mu\frac{\partial^2 u_2}{\partial x_1^2} - (\lambda + \mu)\frac{\partial^2 u_2}{\partial x_2^2} - \mu\frac{\partial^2 u_2}{\partial x_3^2} &= f_2, \\ -\mu\frac{\partial^2 u_3}{\partial x_1^2} - \mu\frac{\partial^2 u_3}{\partial x_2^2} - (\lambda + \mu)\frac{\partial^2 u_3}{\partial x_3^2} &= f_3, \end{aligned}$$

with boundary conditions corresponding to the considered elasticity problem. In Section 4.3.1.1 we demonstrate the spectral equivalence of matrix  $B$  to the stiffness matrix  $A$ , i.e.,

$$c_1 (Bv, v) \leq (Av, v) \leq c_2 (Bv, v), \quad \forall v \in V,$$

where  $c_1$  and  $c_2$  are constants that are independent of  $h$ . It follows that the eigenvalue spectrum of  $B^{-1}A$  is clustered and bounded independently of  $h$ , i.e.,  $\kappa(B^{-1}A) = O(1)$  uniformly in  $h$ . More specifically, it is shown by Axelsson and Gustafsson (1978) and Blaheta (1994) that

$$\kappa(B^{-1}A) \leq (n-1)c \frac{1-\nu}{1-2\nu} \quad (6.16)$$

where  $n$  is the spatial dimension and  $c$  is a positive constant depending only on the computational domain and the boundary conditions. Thus,  $B$  is an optimal preconditioner of  $A$  with respect to  $h$ , but it is not optimal with respect to Poisson's ratio  $\nu$ . However, the inversion of  $B$  may still be too costly, so further preconditioning to the subblocks of  $B$  is required. Methods that have been traditionally used include incomplete block factorizations (Axelsson and

Gustafsson 1978 and Blaheta 1994) and inner PCG iterations preconditioned by incomplete factorizations (Blaheta et al. 2003).

Lirkov (2003) considers an even more simplified matrix corresponding to the following Laplacians,

$$\nabla^2 \mathbf{u} = \mathbf{f}. \quad (6.17)$$

The bilinear form in the weak formulation is written as

$$\tilde{a}(u, v) = \int_{\Omega} \sum_i \sum_j \frac{\partial u_i}{\partial x_j} \frac{\partial v_i}{\partial v_j} d\Omega. \quad (6.18)$$

In the case of rectangular parallelepipeds, the FE approximation of (6.17) with special quadrature rules results in

$$\tilde{B} = \begin{bmatrix} \tilde{A}_1 & & \\ & \tilde{A}_2 & \\ & & \tilde{A}_3 \end{bmatrix}, \quad (6.19)$$

where subblock  $\tilde{A}_i$  has a 7-point finite difference stencil. Each displacement component may either be solved by a fast direct solver based on separation of variables or be approximated by any preconditioner chosen from the numerous libraries designed for the Laplace operator. The attraction of this type of decomposition lies in its efficiency in both CPU time and memory storage. A major drawback, however, is that  $\tilde{B}$  is not very effective at handling reservoir heterogeneities. To overcome this problem we propose to apply the same discretization (7-point finite difference) to the weak form of (6.15) so that varying material properties may be better characterized. In doing so, we

obtain the following displacement decomposed (DiD) system,

$$\begin{bmatrix} A_{DiD,1} & & \\ & A_{DiD,2} & \\ & & A_{DiD,3} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}. \quad (6.20)$$

Similar to the DD preconditioners, the above DiD scheme may also be defined in the context of space decomposition and subspace corrections (Blaheta et al., 2003). Consider the solution of a 3D elasticity problem in a domain  $\Omega \subset R^d$  ( $d = 3$ ). Let  $V$  denote the finite dimensional subspace of a Hilbert space  $\mathcal{V} = \mathcal{V}(\Omega)$ . Then, the displacement decomposition (6.20) may be defined as follows:

$$\begin{aligned} V &= V_1 \oplus \cdots \oplus V_d, \\ V_i &= \{v = (v_1, \dots, v_d) \in V, v_l = 0 \text{ for } l \neq i\}. \end{aligned} \quad (6.21)$$

Since the displacement components are completely decoupled, we may write  $M_{DiD-Direct}$  in (6.20) as a space decomposition preconditioner in an additive form,

$$M_{DiD-Direct} = \sum_{i=1}^d R_i^T A_{DiD,i}^{-1} R_i, \quad (6.22)$$

where  $R_i$  and its transpose are, respectively, the restriction and prolongation operators between  $V$  and  $V_i$ . DiD preconditioning is described in Algorithm 8.

It appears that the implementation of  $M_{DiD-Direct}$  is as simple as an additive Schwarz method. However, the direct subspace solutions incurred by  $A_{DiD,i}^{-1}$  may impose major preconditioning overheads. Thus we need further preconditioning to the subproblem  $A_{DiD,i} u_i = b_i$ .

---

**Algorithm 8** DiD-Direct

---

- 1: Compute residual:  $r^n = b - Au^n$ .
  - 2: Restrict and solve the subspace corrections directly:  $e_i^n = A_{DiD,i}^{-1} R_i r^n$  for  $i = 1, \dots, d$ .
  - 3: Extend and update solution:  $u^{n+1} = u^n + R_i^T e_i^n$
- 

### 6.3.2 Preconditioners for DiD Subsystems

Our preconditioners for the subsystems in (6.20) are still based on the domain decomposition theory. If a domain  $\Omega$  is decomposed into  $p$  subdomains, the subspace  $V_i$  ( $i = 1, \dots, d$ ) for  $u_i$  in (6.21) will be further decomposed into  $p$  sub-subspaces accordingly,

$$V_i = V_{i,1} + V_{i,2} + \dots + V_{i,p}, \quad (6.23)$$

where

$$V_{i,j} = \{v \in V_i : v(x) = 0, \forall x \in \Omega \setminus \Omega_j\}.$$

Clearly, preconditioners based on the decomposition of (6.23) are combinations of the DiD and DD methods.

#### 6.3.2.1 Red-black Line Gauss-Seidel

One simple preconditioner that may be constructed from (6.23) is the red-black line Gauss-Seidel (RB-LineGS). In a red-black coloring, we associate one color (either red or black) to each vertical column so that lines marked with the same color may be updated simultaneously.

Applying  $m$  iterations of RB-lineGS for the subproblem  $A_{DiD,i} u_i = b_i$

leads to the following DiD preconditioner,

$$M_{DiD-LineGS} = \sum_{i=1}^d R_i^T M_{DiD-LineGS,i}^m R_i. \quad (6.24)$$

For one sweep of RB-lineGS we can write  $M_{DiD-LineGS,i}$  as

$$M_{DiD-LineGS,i} = \left\{ I - \prod_{j=1}^{color_p} \left[ I - \left( \sum_{k \in color_i} R_{i,k}^T A_{i,k}^{-1} R_{i,k} \right) A_{DiD,i} \right] \right\} A_{DiD,i}^{-1}.$$

While  $A_{DiD,i}^{-1}$  is included here, it is never formed in the actual implementation.

The efficiency of the DiD methods is well demonstrated by the residual reduction histories shown in Figure 6.2. Applying a 4-color line GS smoother, i.e,  $M_{DDm-LineGS}$  in (6.6), to the elasticity system ( $Au = b$ ) certainly leads to faster numerical convergence (the upper picture in Figure 6.2). However, it takes more CPU time per iteration (1.81 versus 0.85 seconds) as compare to the RB-LineGS scheme as applied to a decoupled system ( $A_{DiD}u = b$ ). Thus,  $M_{DDm-LineGS}$  converges slower than  $M_{DiD-LineGS}$  in terms of CPU time, as shown by the lower picture in Figure 6.2.

### 6.3.2.2 Algebraic Multigrid (AMG)

It is well known that the AMG methods (Ruge & Stüben 1985 and 1987, Trottenberg et al. 2001 and Vaněk et al. 1994) are very effective preconditioners for many linear systems stemming from a finite element or finite difference discretization of partial differential equations. They are widely adopted in practice for two main reasons. First, the methods are based solely on the information of the linear system to be solved. Hence, they obviate the need for

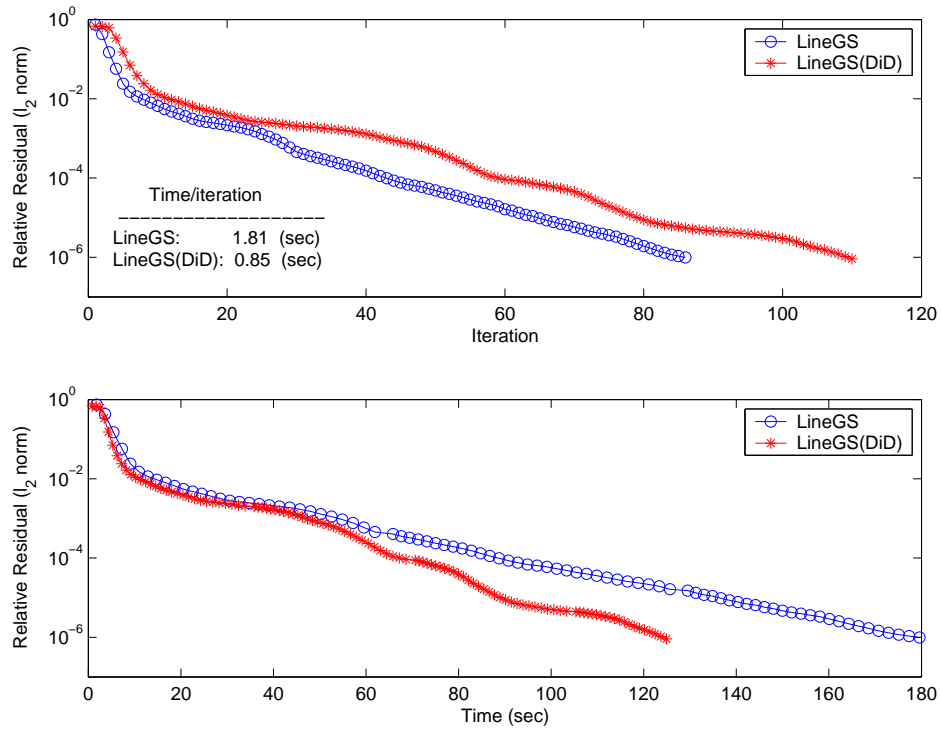


Figure 6.2: Example 1 with  $64 \times 64 \times 32$  grid cells: Convergence histories of multi-color line Gauss-Seidel preconditioners for the elasticity system and the displacement decomposed system.

constructing a sequence of nested grids, which is difficult for many applications with unstructured meshes. Second, AMG is known to be robust with respect to discontinuous coefficients and singular perturbations. In light of these advantages we propose to use an AMG algorithm for solving the subproblems in (6.20).

The building components for AMG are the same as those in a standard multigrid (Section 6.2.2.1), namely, grid transfer operators, coarse grid matrices and a relaxation scheme on each level of the grid. Moreover, AMG follows the same procedures as defined in a standard MG method. For convenience we write a recursively called AMG V-cycle scheme as follows:

---

**Algorithm 9**  $\text{AMGV}(A_m, b_m)$

---

```

if there is a coarser grid  $m + 1$ , i.e.,  $m < L$  then
    Relax on  $A_m u_m = b_m$ .
    Perform coarse grid correction:
    {
    Set  $u_{m+1} = 0$ .
    Compute  $b_{m+1} = I_m^{m+1}(b_m - A_m u_m)$ .
    Solve  $u_{m+1}$  with  $\text{AMGV}$ :  $u_{m+1} = \text{AMGV}(I_m^{m+1} A_m I_{m+1}^m, b_{m+1})$ .
    Correct the solution by  $u_m := u_m + I_{m+1}^m u_{m+1}$ .
    }
    Smooth on  $A_m u_m = b_m$ .
else
     $u_L = A_L^{-1} b_L$ .
    return  $u_m$ 
end if

```

---

Recall that  $m$  ( $1 \leq m \leq L$ ) stands for the level of the coarse grids. Applying Algorithm 9 to the DiD system (6.20) is equivalent to replacing  $A_{DiD,i}^{-1}$  in (6.22) with an approximate inversion by the AMG method, i.e.,

$\tilde{A}_{DiD,i}^{-1} = M_{AMG,i}$ . We obtain the following iterative scheme,

$$u^{n+1} = u^n + \sum_i^d R_i^T M_{AMG,i} R_i (b - Au^n). \quad (6.25)$$

In the case that the DiD subspaces are further decomposed by the DD method, as shown in (6.23), Algorithm 9 is used as a subdomain solver for each DiD component. Several sweeps of RB-lineGS are applied immediately after the DiD system is solved. The iteration scheme takes the same form as (6.25) with  $M_{AMG,i}$  replaced by

$$\begin{aligned} M_{DiD-DD-AMG,i} &= \sum_{j=1}^p R_{i,j}^T M_{AMG,i,j} R_{i,j} \\ &+ M_{DiD-LineGS} \left( I - \sum_{j=1}^p R_{i,j}^T M_{AMG,i,j} R_{i,j} A_{DiD,i} \right), \end{aligned}$$

where  $R_{i,j}$  defines a restriction from subspace  $V_i$  to  $V_{i,j}$  ( $i = 1, \dots, d$  and  $j = 1, \dots, p$ ), and  $M_{DiD-LineGS}$  is given by (6.24).

A major difference of AMG from a geometrical multigrid method lies in the coarse grid construction. Geometrical multigrids rely on an explicit coarse grid mesh to form standard finite element function spaces while the AMG method uses the information of the stiffness matrix on one grid to construct subspaces on the next grid. The components of AMG are built in a separate setup phase, as shown in Algorithm 10.

The goal of the setup step is to construct a set  $C_m$  for the coarse grid points. For each fine grid point  $i \in F_m \equiv \Omega_m \setminus C_m$ , a small set  $C_{m,i} \subset C_m$  is chosen to be its interpolating points, while, for each coarse grid point  $j \in$

---

**Algorithm 10** AMG setup phase

---

1. Set  $m=1$ .
  2. Partition  $\Omega_m$  into disjoint sets  $C_m$  (coarse grid) and  $F_m$  (fine grid ).
  3. Set  $\Omega_{m+1} = C_m$ .
  4. Define interpolation operator  $I_{m+1}^m$  and restriction operator  $I_m^{m+1}$ .
  5. Set  $A_{m+1} = I_m^{m+1} A_m I_{m+1}^m$ .
  - if**  $\Omega_{m+1}$  is small enough **then**
    - Set  $L = m + 1$ .
    - Stop.
  - else**
    - Set  $m = m + 1$ .
    - Go to step 2.
  - end if**
- 

$C_m \equiv \Omega_{m+1}$ , a small set  $F_{m,j} \subset F_m$  is selected to be its restriction points. The interpolation and restriction operators are then decided, respectively, by

$$\begin{aligned} (I_{m+1}^m u_{m+1})_i &= \begin{cases} u_{m+1,i} & \text{if } i \in C_m, \\ \sum_{j \in C_{m,i}} \omega_{ij} u_{m+1,j} & \text{if } i \in F_m \end{cases}, \\ (I_m^{m+1} u_m)_j &= u_{m,j} + \sum_{i \in F_{m,j}} \omega_{ji} u_{m,i}, \end{aligned}$$

where  $\omega_{ij}$  is the weight.

The coarse grid set  $C_m$  on each level is chosen based on the degree of coupling between unknowns. For fixed  $0 \leq \alpha \leq 1$  and the index set  $N := \{1, \dots, n\}$  of all unknowns, an unknown  $i$  is said to be strongly coupled with another unknown  $j$  if  $(|a_{ij}| \geq \alpha \cdot \max_{k \in N} |a_{ik}|)$ . Thus we define

$$S_i := \{j \mid |a_{ij}| \geq \alpha \max_{k \in N} |a_{ik}|\}$$

as the set of all unknowns  $j$  to that  $i$  is strongly coupled and define  $S_i^T$  as the set of all points which are strongly coupled to  $i$ , i.e.,  $j \in S_i^T$  if  $i \in S_j$ . The

coarse grid set  $C_m$  and the fine grid set  $F_m$  in Algorithm 10 are constructed from the sets of  $S_i$  and  $S_i^T$ .

The AMG method is implemented in this work using the AMG1R5 code by Stüben. More detailed discussion of the method may be found in Stüben (1983)

### 6.3.2.3 Algebraic Multigrid for an Aggregated System

The major drawback of the AMG algorithm described above is the significant CPU time spent on the setup phase as compared to its iteration time. For a linear elasticity problem this is acceptable as the stiffness matrix is constant over time, and the coarsening process is, thus, required only in the first solve. However, for nonlinear plasticity problems, the stiffness matrix is to be reinitialized at each Newton step. Due to the fast convergence rate of this method, the large amount of initialization time will not be compensated by the iteration time. To overcome this problem we suggest applying the AMG method to an aggregated system  $A_{DiD,i}^C$  where subscript  $i$  denotes a displacement component. Numerical experiments demonstrate that this strategy is successful for the linear system generated by the CCFD approximation of multiphase flow equations (Vassilevski 2000a and 2000b). Aggregation is achieved by the summation of coefficients and residuals over each small grid block of size  $n_x \times n_y \times n_z$ . The coarse grid matrix  $A_{DiD,i}^C$  is computed by

$$A_{DiD,i}^C = I_1^C A_{DiD,i} (I_1^C)^T,$$

where  $I_1^C$  is the restriction operator from the original grid  $\Omega_1$  to the aggregated grid  $\Omega_C$ . Clearly,  $A_{DiD,i}^C$  is a Galerkin projection of  $A_{DiD,i}$  to the subspace defined on  $\Omega_C$ . In the case that we group unknowns within a small block  $(3 \times 1 \times 1)$  that represents an aggregation along the  $x$  direction, the restriction operator reads,

$$I_1^C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & \cdots & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & \cdots & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 1 \end{bmatrix}.$$

It corresponds to the summation over the following subsets along each grid line in  $x$ :  $\{1\}$ ,  $\{2, 3, 4\}$ ,  $\{5, 6, 6\}$ , ...,  $\{N_x\}$ .

---

**Algorithm 11** DiD-AMGC

---

**for**  $i = 1, \dots, d$  ( $d = 3$ ) **do**

1. Perform  $k_2$  sweeps of RB line Gauss-Seidel on  $A_{DiD,i}u_i = b_i$ .
2. Set correction on the aggregated coarse grid to zero:  $e_{C,i} = 0$ .
3. Compute residual and restrict to  $\Omega_C$ :  $b_{C,i} = I_1^C(b_i - A_{DiD,i}u_i)$ .
4. Solve  $A_{DiD,i}^C e_{C,i} = b_{C,i}$  with AMG:  $e_{C,i} = \text{AMGV}(A_{DiD,i}^C, b_{C,i})$ .
5. Correct the solution by  $u_i := u_i + (I_1^C)^T e_{C,i}$ .
6. Perform  $k_2$  sweeps of RB line Gauss-Seidel on  $A_{DiD,i}u_i = b_i$ .

**end for**

---

Algorithm 11 may be written in one iteration as

$$u^{n+1} = u^n + \sum_{i=1}^d [R_i M_{DiD-AMGC,i} R_i^T] (b - Au^n), \quad (6.26)$$

where  $M_{DiD-AMGC,i}$  is given by

$$\begin{aligned} M_{DiD-AMGC,i} &= (I_1^C)^T M_{DiD-AMG,i} I_1^C \\ &+ M_{DiD-LineGS,i} [I - A_{DiD,i} (I_1^C)^T M_{DiD-AMG,i} I_1^C]. \end{aligned}$$

If the displacement decomposition is combined with a domain decomposition, the preconditioner  $M_{DiD-AMGC,i}$  in (6.26) will be replaced by

$$\begin{aligned} M_{DiD-DD-AMGC,i} &= \sum_{j=1}^p R_{i,j}^T M_{DiD-AMGC,i,j} R_{i,j} \\ &+ M_{DiD-LineGS,i} \left[ I - A_{DiD,i} \sum_{j=1}^p R_{i,j}^T M_{DiD-AMGC,i,j} R_{i,j} \right], \end{aligned}$$

where  $R_{i,j}$  is a restriction operator from  $V_i$  to  $V_{i,j}$ .  $M_{DiD-AMGC,i,j}$  stands for the solution of displacement  $u_i$  in subdomain  $\Omega_j$ . In this preconditioning each subsystem in (6.20),  $A_{DiD,i} u_i = b_i$ , is solved by Algorithm 11 followed by several sweeps of global smoothings by line Gauss-Seidel.

## 6.4 Numerical Experiments

In this section we evaluate the performance of our DD preconditioners ( $M_{DDa}$  and  $M_{MLDD}$ ) and DiD preconditioners ( $M_{DiD}$ ) on four numerical examples. The test cases are designed to include several features of challenging finite element simulations: scalability with mesh refinement, large Poisson's ratios and discontinuities in material properties. The goal of these experiments is to evaluate the effectiveness and robustness of our preconditioners over a wide range of conditions as defined by these features.

All examples are tested on a 2.0 GHz AMD Athlon Dual processor machine with 1 GB of memory each. Only one processor is used, which means that we are considering the DD preconditioners with one subdomain ( $p = 1$ ). In all cases GMRES is used as the linear solver. Iterations are terminated once

the relative residual norm is less than  $10^{-6}$ .

### **6.4.1 Numerical examples**

#### **6.4.1.1 Example 1**

The first example is a three-dimensional, three-phase water-flooding problem in a quarter of a five spot pattern. The size of the reservoir is 1056 ft×1056 ft×160 ft. The domain is discretized uniformly into a mesh consisting of 16×16×8 cells. Later, the grid will be further refined to evaluate the scalability of our linear solvers with respect to the mesh size  $h$ . The initial in-situ porosity is 0.3. The initial oil saturation is 0.8. No free gas is present. The permeability field varies by layers. The Young's modulus and Poisson's ratio are, respectively,  $3.095 \times 10^4$  psi and 0.3. Two wells are completed at the reservoir corners. The injection well injects water at a rate of 500 stb per day. The production well produces at a rate of 750 stb oil per day. A load of 6000 psi is applied on the top of the reservoir. All the other boundary faces are confined (zero displacement in the normal direction and zero traction in the tangential directions). No flow occurs at any of the reservoir boundaries.

#### **6.4.1.2 Example 2**

Example 2 has the same reservoir configuration and parameter distribution as example 1. The difference lies in the boundary conditions for the equilibrium equations. Instead of confining the reservoir at its lateral boundaries, a normal stress of 2600 psi is applied. The rectangular domain is dis-

cretized by a mesh of  $32 \times 64 \times 64$  grid cells. To study the robustness of our preconditioners with respect to large Poisson's ratios, a scenario of simulations is conducted with varying Poisson's ratios.

#### **6.4.1.3 Example 3**

The third example, presented in Section 4.4.1.3, is used here to test the performance of our preconditioners with respect to jumping coefficients. The surrounding non-pay rocks are also included in the numerical calculation for a better characterization of the reservoir boundary conditions as in Figure 4.3. The entire domain is non-uniformly discretized into  $24 \times 24 \times 12$  grid cells with much coarser grids in the outer domain.

In our numerical testing, the Young's modulus in the non-pay rocks ( $E_2$ ) varies in each run so that a jump discontinuity is created at the interface of the reservoir and its surroundings. Performance of the DD and DiD preconditioners with increasing jump ratios is compared.

#### **6.4.1.4 Example 4**

In the solution of initial displacement, only body forces and boundary conditions affect force equilibrium since the initial pore pressure has not been disturbed. Hence, the convergence behavior of a linear solver in the initial solve may not be representative for the entire pressure evolution history. Performance of different preconditioners needs to be compared over multiple time steps to account for the pressure effects. Example 4 is slightly modified from

Grid	DDa-IC		MLDD		
$N_x \times N_y \times N_z$	IC(0)	IC(1)	SCMG	IC(0)/SCMG	IC(1)/SCMG
Displacement solution in initialization					
$8 \times 8 \times 4$	17	11	4	6	5
$16 \times 16 \times 8$	36	22	5	8	7
$32 \times 32 \times 16$	73	46	5	15	12
$64 \times 64 \times 32$	156	97	6	28	23
Displacement solution in the first Newton step					
$8 \times 8 \times 4$	18	11	5	6	5
$16 \times 16 \times 8$	39	24	6	9	7
$32 \times 32 \times 16$	71	45	10	14	12
$64 \times 64 \times 32$	146	93	19	27	22

Table 6.4: Example 1 with varying grid refinements: Total number of linear solver iterations for one-level DD (DDa-IC) and multilevel DD (MLDD-IC/SCMG) preconditioners.

Example 3 by imposing traction conditions on the four lateral boundary faces. The unconfined problem is simulated for 10 time steps of 20 days each. The entire simulation involves 228 solves of the elasticity system.

#### 6.4.2 Convergence with Respect to Grid Refinement

Example 1 is initially discretized with a  $8 \times 8 \times 4$  grid. We refine the mesh three times further. The last refinement generates a grid of  $64 \times 64 \times 32$ . Poisson's ratio is fixed at 0.3. The total number of iterations in the initial solve on each grid is reported in Table 6.4.

The IC factorization preconditioners with both zero and one fill-in level converge slowly and appear to degrade severely with the mesh refinements (e.g. IC(0) takes 17 iterations to converge on a  $8 \times 8 \times 4$  grid while it takes 156

Preconditioner MLDD-IC/SCMG	Memory (MB)	Initialization time (sec)	Sol. time/iteration (sec)
IC(0)	845	2.06	2.37
IC(1)	1039	7.20	3.25

Table 6.5: Example 1 with  $64 \times 64 \times 32$  grid cells: Comparison of memory storage, linear solver initialization time and solution time per iteration for the incomplete Cholesky factorization with zero and one fill-in levels.

iterations on a  $64 \times 64 \times 32$  grid). The multilevel DD preconditioners are superior to the one-level DD methods. They converge faster and have more stable performance with smaller mesh size  $h$ . Despite the poor performance of DDa-IC(0), its multilevel variant, MLDD-IC(0)/SCMG, is barely inferior to MLDD-IC(1)/SCMG. This observation may obviate the use of IC(1) as a first level smoother in multilevel schemes. The advantage of MLDD-IC(0)/SCMG over MLDD-IC(1)/SCMG is demonstrated in Table 6.5 by comparing their memory storage and initialization time.

Figure 6.3 shows decreasing relative residual norms with linear iteration counts on a mesh of  $64 \times 64 \times 32$  grid cells. The better performance of SCMG is due to the initial and boundary conditions imposed for this problem. Under these conditions, body force is the only external force. If the reservoir is confined horizontally, it deforms primarily in the vertical direction, and strong couplings are also in the vertical direction. Thus, pre-smoothings using line Gauss-Seidel together with the super-coarsening are very effective at reducing both the high frequency and low frequency error components. The convergence shows minimal dependence on the mesh parameter  $h$ . However, once the time

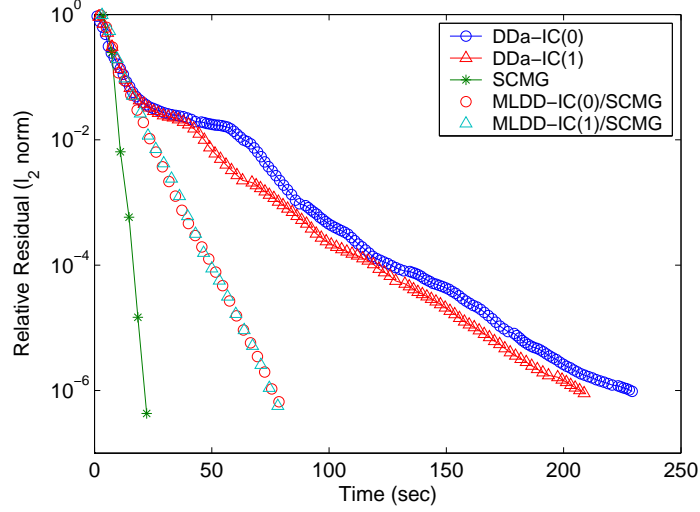


Figure 6.3: Example 1 with  $64 \times 64 \times 32$  grid cells: Convergence histories of one-level DD and multilevel DD (MLDD) preconditioners.

evolution starts, and pore pressure plays a role in the deformation, SCMG shows deterioration with smaller  $h$ . The degradation is well demonstrated by comparing the iteration counts in the initial solve and the first Newton step (Table 6.4).

The DiD preconditioners behave similarly to the multilevel DD methods (Table 6.6), but convergence deterioration with mesh parameter  $h$  is improved. For simplicity, we denote the application of AMG on an aggregated coarse grid by  $\text{AMG}(n_x, n_y, n_z)$ , where  $((n_x, n_y, n_z))$  represents the dimension of each subgrid block for aggregation. Applying AMG on an aggregated system causes only slight deterioration in numerical convergence. For example, on a mesh of  $64 \times 64 \times 32$ ,  $\text{AMG}(1,1,1)$  takes 21 iterations to converge while  $\text{AMG}(1,1,3)$

Grid	DiD-AMG				
	(1,1,1)	(1,1,3)		(3,3,2)	
$N_x \times N_y \times N_z$	0/0	0/1	0/2	0/1	0/2
$8 \times 8 \times 4$	10	14	9	13	7
$16 \times 16 \times 8$	17	18	12	18	12
$32 \times 32 \times 16$	17	21	15	20	16
$64 \times 64 \times 32$	21	23	18	21	17

Table 6.6: Example 1 with varying grid refinements: Total number of linear solver iterations using DiD-AMG preconditioners.  $(n_x, n_y, n_z)$  indicates the dimension of a small subgrid block for aggregation.  $k_1/k_2$  denotes the number of pre/post smoothings using multi-color line Gauss-Seidel.

Preconditioner DiD-AMG	Memory (MB)	Initialization time (sec)	Sol. time/iteration (sec)
(1,1,1)	956	17.90	2.58
(1,1,3)	746	3.24	1.18
(3,3,2)	647	0.65	0.88

Table 6.7: Example 1 with  $64 \times 64 \times 32$  grid cells: Comparison of memory storage, linear solver initialization time and solution time per iteration for DiD preconditioners.

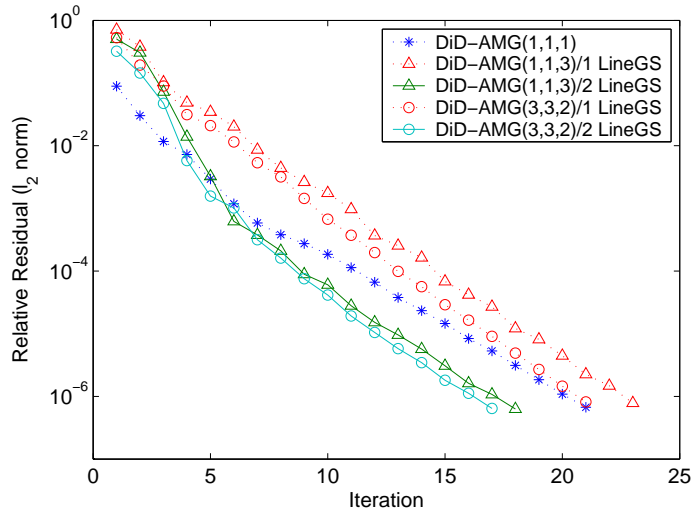


Figure 6.4: Example 1 with  $64 \times 64 \times 32$  grid cells: Convergence histories of DiD preconditioners. DiD-AMG( $n_x, n_y, n_z$ )/ $k$ -LineGS indicates that AMG is applied to a coarse grid (generated by aggregations over each subgrid block of  $n_x \times n_y \times n_z$ ) with  $k$  sweeps of post-smoothing.

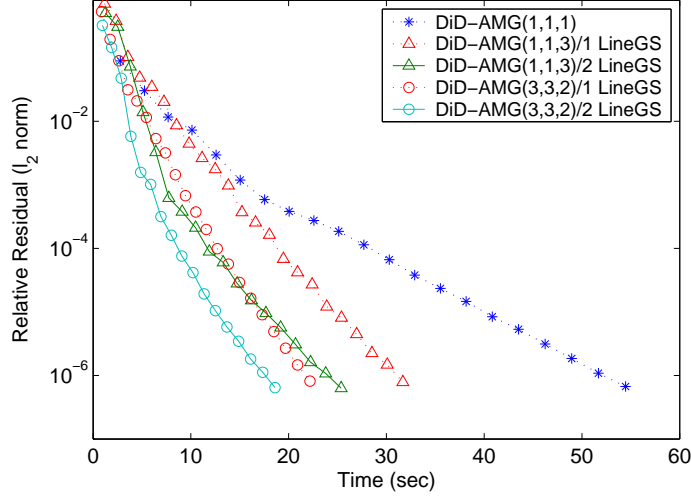


Figure 6.5: Example 1 with  $64 \times 64 \times 32$  grid cells: Convergence histories of DiD preconditioners. DiD-AMG( $n_x, n_y, n_z$ )/ $k$ -LineGS indicates that AMG is applied to a coarse grid (generated by aggregations over each subgrid block of  $n_x \times n_y \times n_z$ ) with  $k$  sweeps of post-smoothing.

and AMG(3,3,2) take 23 and 21 iterations, respectively. Adding one more sweep of line Gauss-Seidel for post-smoothing further reduces the numbers of iterations to 18 and 17 in each case. The advantage of using AMG on a coarser grid is demonstrated by comparing DiD-AMG(1,1,1), DiD-AMG(1,1,3) and DiD-AMG(3,3,2) in terms of memory storage, initialization time and linear solver time per iteration (Table 6.7). Observations are:

1. As much as 32 percent of memory space is saved by using AMG on an aggregated coarse grid.
2. The execution time per iteration is reduced at least by a factor of 2.

3. The preconditioner AMG(1,1,1) takes a substantial amount of time to initialize as compared to its solution time per iteration, which may greatly limit its practical usefulness for nonlinear plasticity problems. However, applying AMG to the Galerkin projection of the original displacement blocks we obtain significant reductions in the initialization time.
4. The residual reduction histories shown in Figure 6.4 indicate that preconditioning an aggregated matrix may cause a slow numerical convergence, but the slight degradation is well compensated by the low computation cost per iteration. Thus, the total execution time is reduced considerably (Figure 6.5).

### 6.4.3 Convergence with Respect to Poisson's Ratio

It is well known that the condition number of a discrete elasticity system depends on Poisson's ratio ( $\nu$ ). As  $\nu$  approaches 0.5 the stiffness matrix approaches singularity. In the following, we use the second example above to investigate the robustness of our DD and DiD preconditioners with respect to Poisson's ratio.

Table 6.8 shows the numerical results of using the SCMG preconditioner. We apply point Gauss-Seidel ( $\omega = 1.0$ ) and SOR ( $\omega = 1.3$ ) with varying numbers of pre- and post-smoothings on the 2D coarse grids.  $(k_1/k_2)$  in Table 6.8 denotes a SCMG V-cycle with  $k_1$  pre- and  $k_2$  post-smoothing steps. As expected, the speed of convergence becomes slower with increasing

Poisson ratio ( $\nu$ )	$\omega = 1.0$	$\omega = 1.3$	
	(3/3)	(3/3)	(7/7)
0.05	40	36	33
0.1	41	36	34
0.2	47	36	34
0.3	50	39	36
0.34	51	43	40
0.36	53	49	47
0.4	60	54	47
0.44	82	63	57
Time/iteration (sec)	2.86	2.86	2.90

Table 6.8: Example 2 with  $64 \times 64 \times 32$  grid cells with varying Poisson's ratios: Total number of linear solver iterations using SCMG with different SOR relaxation parameters  $\omega$ . ( $k_1/k_2$ ) indicates the number of pre- and post-smoothings in SCMG.

Poisson's ratios. Both the type of smoother and the number of smoothing steps affect the linear solver convergence for this unconfined problem, though they have virtually no effect for the first example (confined problem). The SOR smoother with  $\omega = 1.3$  is even more effective at larger Poisson's ratios. It is interesting to note, however, that additional sweeps of SOR smoothing on each multigrid level do not incur significant computation overhead. Results in Table 6.8 show that adding 8 more smoothing steps (pre- and post-smoothing together) cause but an increment of 0.04 seconds in the total execution time per iteration. Figure 6.6 demonstrates the total solution times for different smoothers with varying numbers of smoothing steps. It shows that the effect of smoothing quality on the performance of SCMG becomes more essential with larger values of Poisson's ratio.

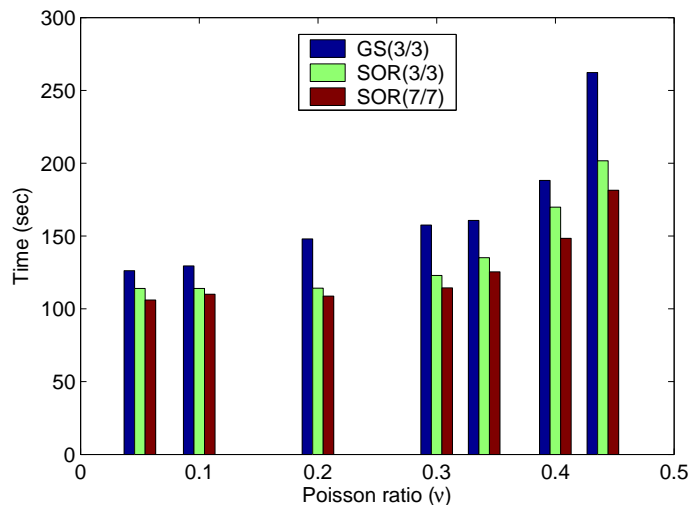


Figure 6.6: Example 2 with  $64 \times 64 \times 32$  grid cells and  $\nu = 0.44$ : Total linear solver solution time using SCMG with point Gauss-Seidel and SOR smoothers of varying smoothing steps.

The other MLDD methods, DDa-IC(0)/SCMG and DDa-IC(1)/SCMG, deliver results similar to those of the SCMG method (Table 6.9 and Figure 6.7). They all seem to be robust for Poisson's ratios less than 0.3, though moderate deterioration of convergence is observed for larger Poisson's ratios.

The DiD preconditioners converge more independently of Poisson's ratio than the multilevel methods. Iteration counts listed in Table 6.10 suggest that applying AMG to an aggregated system leads to a more stable convergence rate if the aggregated grid is not too coarse, e.g., AMG(1,1,3). Applying additional pre- or post-smoothings by line Gauss-Seidel is only effective for coarser aggregated grids, e.g., DiD-AMG(3,3,2) in Table 6.10.

Total linear solver solution time with varying Poisson's ratios are plot-

Poisson ratio ( $\nu$ )	SCMG $\omega = 1.3$ (7/7)	Hybrid MLDD DDa-IC(0)/SCMG	Hybrid MLDD DDa-IC(1)/SCMG
0.05	33	45	38
0.1	34	45	38
0.2	34	46	38
0.3	36	48	39
0.34	40	53	41
0.36	43	56	43
0.4	47	63	46
0.44	57	*	51

Table 6.9: Example 2 with  $64 \times 64 \times 32$  grid cells and varying Poisson's ratios: Total number of linear solver iterations using multilevel DD (MLDD) preconditioners.

Poisson ratio ( $\nu$ )	DiD-AMG						
	(1,1,1)	(1,1,3)			(3,3,2)		
	0/0	0/1	1/1	0/2	0/1	1/1	0/2
0.05	35	46	46	44	46	48	45
0.1	45	43	44	43	51	51	48
0.2	35	41	41	42	35	39	28
0.3	46	41	42	43	49	50	47
0.34	47	46	46	47	53	48	46
0.36	48	32	33	30	55	48	47
0.4	43	45	43	44	62	53	54
0.44	47	51	49	47	95	62	62

Table 6.10: Example 2 with  $64 \times 64 \times 32$  grid cells and varying Poisson's ratios: Total number of linear solver iterations for DiD-AMG preconditioners.  $k_1/k_2$  indicates the number of pre- and post-smoothings.

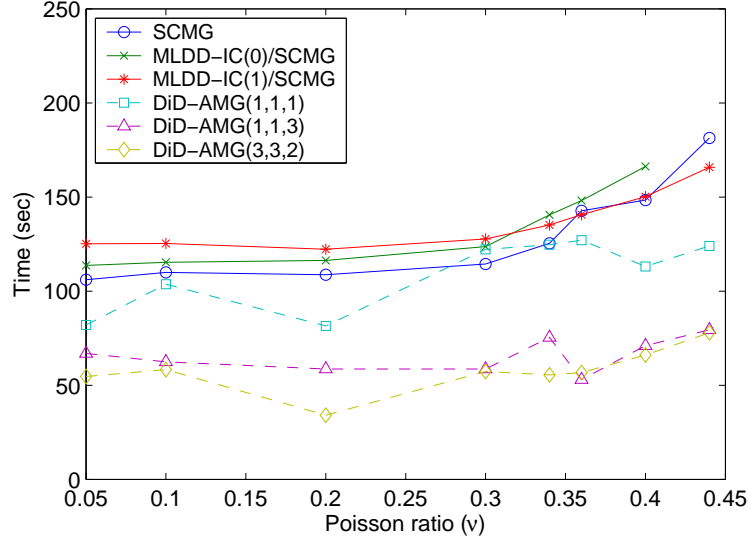


Figure 6.7: Example 2 with  $64 \times 64 \times 32$  grid cells and varying Poisson's ratios: Total linear solver solution time for multilevel DD and DiD preconditioners.

ted in Figure 6.7 for different preconditioners. The results of multilevel DD preconditioners are comparable with the AMG(1,1,1) method. The preconditioning of AMG for the Galerkin projection of a fine grid problem greatly reduces the overall grid/operator complexity. Hence, both initialization time and CPU time per iteration are significantly decreased. The resulting preconditioner is more robust with respect to Poisson's ratio and converges two times faster than the other methods.

#### 6.4.4 Convergence with Respect to Coefficient Jump

In the former numerical experiments we used constant material properties for the entire domain of the reservoir. However, such ideally homogeneous

reservoirs rarely exist in practice. Reservoirs generally consist of different rock types with different properties and demonstrate some degree of heterogeneity in spatial distributions. The nonuniform distribution of material properties leads to coefficient jumps in the elasticity operator. As a result, the corresponding linear system is even more difficult to solve. Robustness with respect to rock heterogeneity is one of the most important criteria for evaluating linear solver performance in a coupled simulation. In this section we use Example 3, as described in Section 6.4, to examine the convergence behavior of our linear solvers and preconditioners with varying degrees of jumps in the elastic modulus.

The computation domain has two subdomains, as shown in Figure 4.3. We choose the Young's modulus inside the reservoir to be  $E_2 = 10^4 \text{ psi}^{-1}$  and solve the problem with different Young's moduli in the surrounding rocks ( $E_1$ ). Table 6.11 gives the number of linear iterations for a discretization of  $24 \times 24 \times 15$  elements. The magnitude of the jump ( $E_1/E_2$ ) is on an order of 1 to 6. Figure 6.8 shows the decreasing relative residual norms with CPU times for minimum and maximum jump ratios that are, respectively,  $E_1/E_2 = 1$  and  $E_1/E_2 = 10^6$ .

As shown in the previous examples, DiD preconditioners provide better convergence results than multilevel DD methods. Their convergence rate is fairly stable with increasing jump discontinuity. The multilevel methods display a slight sensitivity to coefficient jumps. The robustness of DiD preconditioners becomes even more obvious in the residual reduction histories

$E_1/E_2$	DiD-AMG			MLDD		
	(1,1,1)	(1,1,3)	(3,3,3)	SCMG	IC(0)/SCMG	IC(1)/SCMG
$10^0$	24	26	23	18	13	10
$10^1$	25	27	24	19	13	11
$10^2$	26	28	25	22	17	13
$10^3$	26	28	26	24	18	13
$10^4$	26	28	27	24	22	13
$10^5$	26	28	27	24	23	13
$10^6$	26	28	27	24	23	13

Table 6.11: Example 3 with varying magnitudes of jumps in Young’s modulus: Total number of linear solver iterations for DiD-AMG and MLDD preconditioners.

shown in Figure 6.8. Figure 6.9 plots total linear solver solution time with respect to varying magnitudes of jumps in  $E$ . Our main observations are:

1. DiD-AMG methods are robust in handling jump coefficients. Their convergence rate is almost independent of the discontinuity in  $E$ .
2. The performance of MLDD preconditioners deteriorates with increasing values of  $E_1/E_2$ . But the degradation may be overcome or significantly improved by adopting a stronger smoother on the first level grid, e.g. replacing IC(0) with IC(1). The resulting preconditioner (DDa-IC(1)/SCMG) is comparable to DiD-AMG methods on a coarser grid.

Numerical tests on a refined grid of  $24 \times 48 \times 48$  demonstrate similar convergence results (Figure 6.10). DiD preconditioners show their robustness in handling strong heterogeneity. Their converge rate is almost independent of

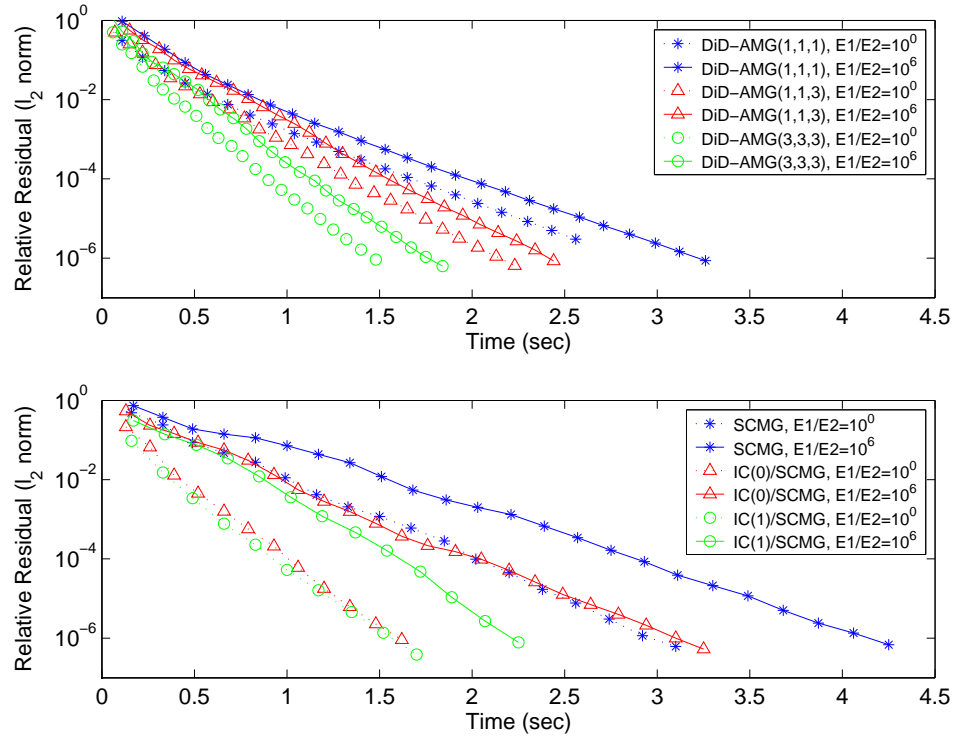


Figure 6.8: Example 3 with varying magnitudes of jumps in Young's modulus: Convergence histories of different space decomposition methods. Upper: DiD preconditioners; Lower: multilevel DD preconditioners.

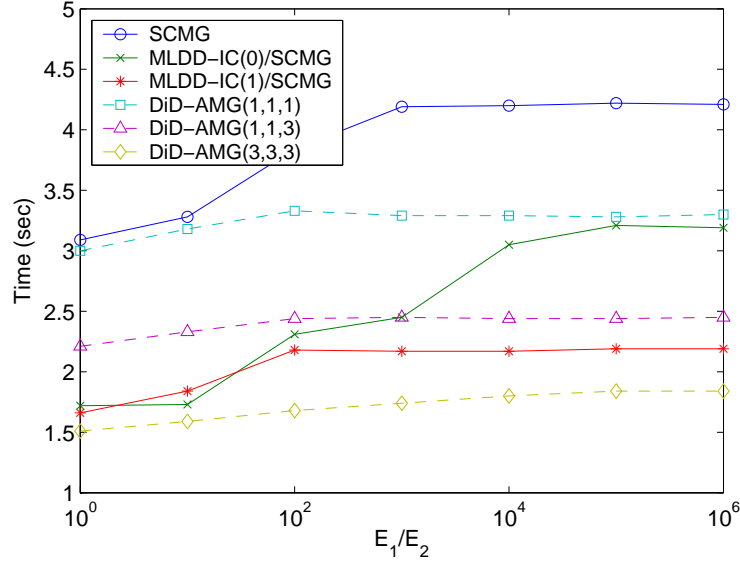


Figure 6.9: Example 3 with varying magnitudes of jumps in Young's modulus: Linear solver solution times for DiD-AMG and MLDD preconditioners.

coefficient jumps. Multilevel methods, however, are only comparable to DiD-AMGs for small or moderate discontinuities ( $E_1/E_2 < 100$ ). For larger jumps their convergence degrades quickly even with DDa-IC(1)/SCMG, which shows fairly good results on the original mesh. The ineffectiveness of MLDD methods for larger values of  $E_1/E_2$  may be improved by enhancing the smoothing quality on the first level grid. For example, the DDa-IC(2)/SCMG preconditioner reduces the convergence deterioration significantly, as shown in Figure 6.10. However, applying stronger smoothers on 3D grids incurs higher computation costs per iteration. The resulting preconditioners may be robust, but they are not efficient. Thus, the development of a good preconditioner centers on the issue of balancing efficiency and robustness.

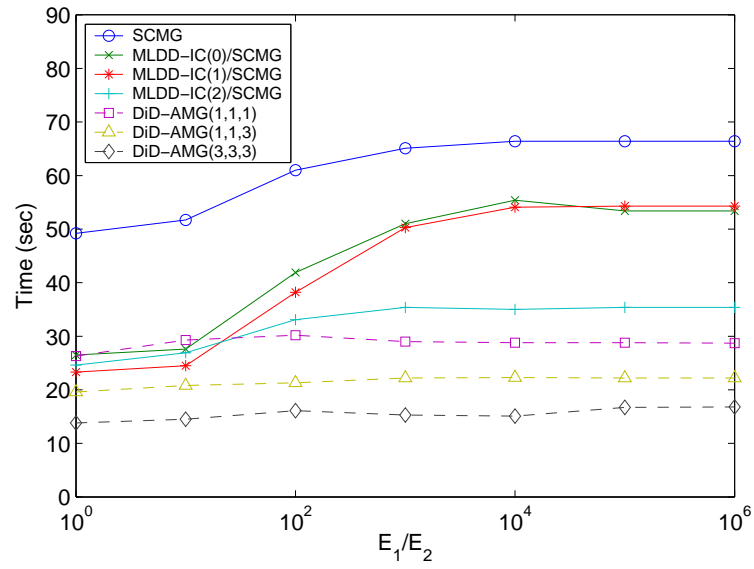


Figure 6.10: Example 3 with a refined grid ( $24 \times 48 \times 48$ ) and varying magnitudes of jumps in Young's modulus: Linear solver solution time for DiD-AMG and MLDD preconditioners.

#### 6.4.5 Comparison over Multiple Time Steps

In the former numerical experiments, linear iteration counts and CPU times are recorded in either the initialization or the first Newton step. Example 4 is used here to evaluate the overall performance of our preconditioners over multiple time steps. Young's moduli are chosen to be  $E_1 = 10^6$  psi and  $E_1 = 10^4$  psi. The entire simulation takes 10 time steps for a 200-day production history. The time step is fixed at 20 days.

Figure 6.11 shows the residual histories in the first Newton step using different preconditioners. Multilevel schemes converge much faster than one-level DD methods, which indicates that SCMG is very effective in removing the low frequency components of the error.

Comparison of total CPU times for different preconditioners are presented in Figure 6.12. They demonstrate results similar to those in Figure 6.11, namely, multilevel preconditions are faster than one-level method. They are even slightly faster than DiD-AMG with no grid aggregation. However, the fastest method is the DiD-AMG algorithm as applied to a coarse grid.

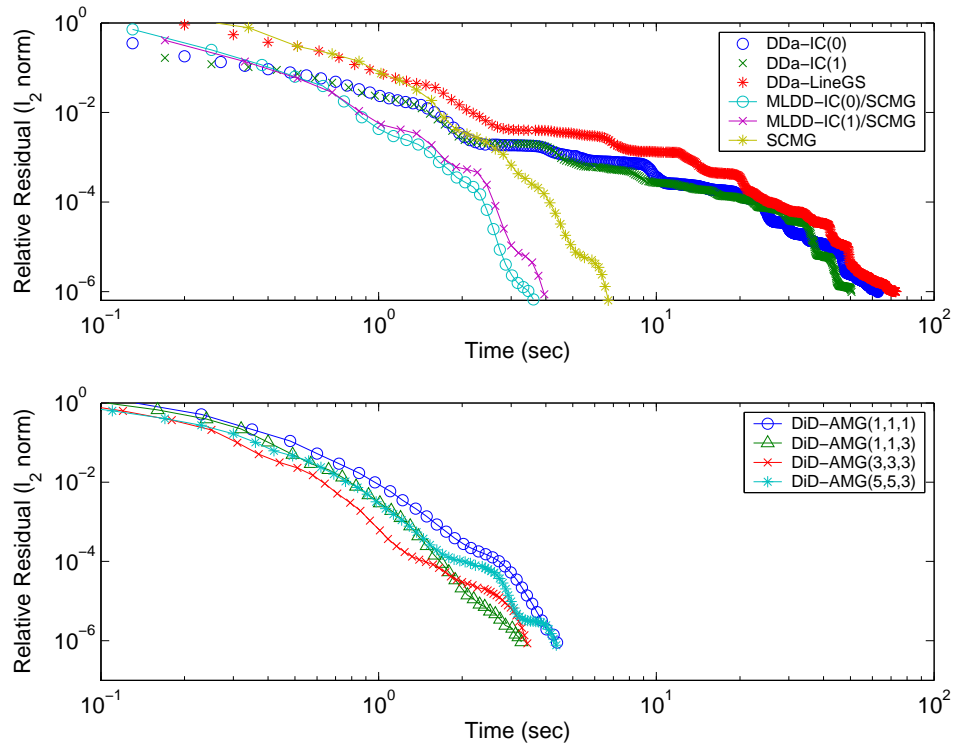


Figure 6.11: Example 4: Residual reduction history vs. computer running time in the first Newton step.

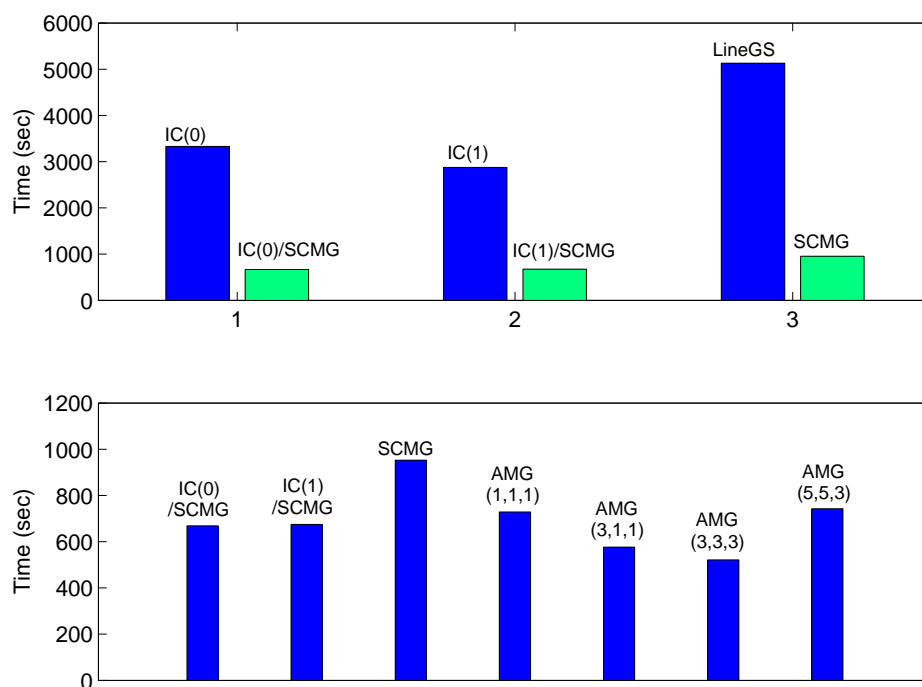


Figure 6.12: Example 4: Total CPU time comparison. Upper: comparison of one-level DD and multilevel DD preconditioners; Lower: comparison of multilevel DD and DiD preconditioners.

## Chapter 7

### Parallel Implementation of the Poroelastic Model

A parallel version of the coupled reservoir flow and geomechanics model is developed under the IPARSv2 framework using the message-passing interface (MPI). The main objective of this chapter is to consider the parallelization issues of the domain decomposition (DD) and displacement decomposition (DiD) preconditioners discussed in Chapter 6. This work is motivated by the petroleum industry's increasing demand for coupled analysis for large scale full-field 3D applications, together with the wide spread availability of ever more powerful parallel computers. Note that a parallel three-phase reservoir flow model (black-oil model) and its iterative linear solvers have already been implemented by other contributors to IPARS (Lu 2000 and Vassilevski 2000). A parallel performance analysis shows a nearly linear speedup and more than 90% parallel efficiency (Figure 7.1). A detailed description of the IPARS parallel data structure is given by Edwards (1998).

This chapter proceeds as follows: Grid partitioning and data allocation among a given number of processors in IPARS are introduced in Section 7.1; In Section 7.2 we discuss in general the inter-processor communications re-

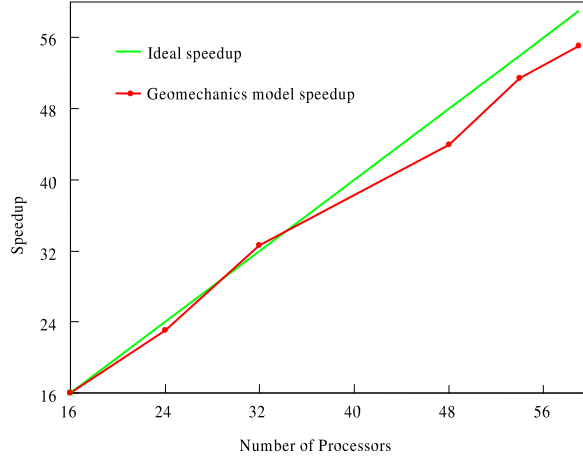


Figure 7.1: Parallel scaling of the black-oil model on a 64-node Beowulf Pentium II 400 MHz PC cluster with a 1.28 gigabit/sec Myrinet network.

quired by our linear solvers and preconditioners; In particular, parallel aspects of a super-coarsening multigrid (SCMG) algorithm are addressed in Section 7.3 which is followed by another Section describing the communication reduction techniques employed in this work to improve parallel efficiency; Numerical results are presented in Section 7.5 and 7.6 to demonstrate the parallel performance of our coupled simulations.

In the following discussion we mainly use the terminology of parallel machines/computers for distributed memory machines.

## 7.1 Grid Partitioning

The IPARS framework uses the domain decomposition method to divide the original grid into as many subgrids as the number of processors used.

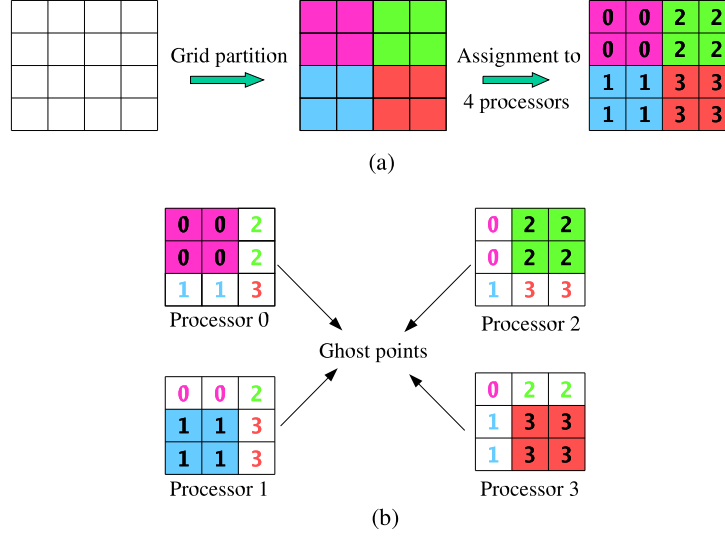


Figure 7.2: Grid partitioning based on domain decomposition.

We choose the partition in such a way that every grid point belongs to only one subgrid or subdomain. Each processor is then assigned to one subgrid, as shown in Figure 7.2 (a). To handle the dependency of a grid point on its adjacent interface points, one overlapping boundary (ghost point) is added around each subgrid (Figure 7.2 (b)). To obviate the necessity of data exchange between the flow and geomechanics model among processors, both models use the same grid partitioning and share the same memory storage. The solution proceeds by solving the original problem in each subdomain followed by data exchanges of the overlapping boundary conditions on the padded interfaces.

In IPARS grid cells and degrees of freedom are grouped in clusters of vertical columns. Subdomains are obtained by decomposing the original

grid in horizontal directions ( $x$  and  $y$ ). This implies that degrees of freedom along a vertical grid line are associated with only one processor. The goal of grid allocation is to distribute the total arithmetical workload evenly among processors. Since the arithmetical workload is assumed to be proportional to the number of grid points, the decomposition algorithm distributes grid columns as evenly as possible over processors. Another major concern in grid partitioning is the so called surface-to-volume ratio. In order to reduce inter-processor communications, the ratio needs to be minimized.

## 7.2 General Inter-processor Communications

After grid partition and data distribution, the actual solution process can be separated into two phases, namely, the setup phase and the linear system solution phase. The setup phase involves the assembly of the stiffness matrix and the residual calculation for discretized poroelasticity equations. Since we are generating a nodal based linear system of equations, only “ghost point” updates are required at this stage. Parameters that need to be updated include:

1. Material properties:  $E$  and  $\nu$ ;
2. Biot’s constants:  $\alpha$  and  $\frac{1}{M}$ ;
3. Fluid variables: pressure, saturation and density for each phase;
4. Initial stresses and displacements.

In the linear solution phase using Krylov subspace methods, generally three types of operations require data communications among processors:

1. Matrix-vector product:  $f = Au$ ;
2. Preconditioning matrix-vector product, e.g.,  $f_i = A_{DiD,i}u_i$  where  $A_{DiD,i}$  is the subspace operator arising from the displacement decomposition;
3. Global scalar vector-vector product:  $a = u \cdot v$ .

For a matrix-vector product, only the data at ghost points needs to be transferred between neighboring processors. A vector-vector product, however, requires a global summation over all processors, involving the passage of data from slave processors to a master processor and broadcasting the summation result from the master processor to all other processors. Frequently, the MPI libraries for collective communications are employed here to optimize communications.

Parallel implementation of a preconditioner depends on its type. For all the preconditioners developed in this work, the “ghost point” communication technique is sufficient, except for the SCMG scheme that will be discussed in detail in the next section.

### 7.3 Parallel Super-coarsening Multigrid (SCMG)

If a multigrid algorithm is to be implemented on a parallel system, additional aspects (mathematical and technical) must be taken into account.

Here, we are interested in practical issues such as the objective of minimizing the corresponding parallelization overhead. In grid partitioning of a multigrid algorithm, the mapping of subgrids or subdomains on each grid level to individual processors is based on the decomposition at the fine grid level. In general, there is no reason to change the partitioning of subdomains or the mapping of coarse grids to individual processors. In other words, the same geometric points on different grid levels always belong to the same processor. Otherwise, additional communications would be required in intergrid transfer operations.

In terms of data communications, two types of overlappings are required by the multigrid components (Lang et al. 2000), i.e.,

1. Horizontal overlap (HGhost): after a coarse grid is decomposed, at least one overlapping boundary is placed around each subdomain. The corresponding ghost point communications are required by pre- and post-smoothings, as well as restriction operations;
2. Vertical overlap (VGhost): in intergrid transfer operations, whenever a coarse grid point requires the data information of a fine grid point that belongs to other processors, the coarse grid point must have a local copy of the data at that fine grid point.

Here we consider the five components of a SCMG scheme. Data communications in each component are detailed as follows.

## **Relaxation**

Assume that an initial guess is given and that the HGhost nodes have been updated. Then, using a standard smoother, each processor may independently perform one sweep of relaxation over its own subgrid. After all processors have completed this sweep, the information at the HGhost points must be updated for either the next relaxation sweep or the residual calculation. In the case that a multi-color Gauss-Seidel is used for smoothing, the HGhost points must be updated after the relaxation for each color.

## **Residual calculation**

The pre-smoothing step in a coarse grid correction scheme is followed by the residual calculation. Since we have already updated the HGhost points after each relaxation, there is no need for new updates prior to the defect calculation. Each processor can complete this task independently.

## **Restriction**

If full weighting is used as a restriction operator, each coarse grid point must know the residual values of all its neighboring fine grid points. Local copies are required in the restriction operation if some of these fine grid points belong to other processors. Since the data communications here involve two grid levels, local copies of the data should be stored in the VGhost points. Later, we will introduce a simple scheme to transform VGhost (fine-to-coarse) communications to HGhost (coarse-to-coarse) communications for the purpose

of reducing communication overhead.

### **Interpolation**

Using a bilinear interpolation, as in our case, if the HGhost nodes on a coarse grid have already been updated after the last smoothing step, we do not need additional communications to compute the fine grid corrections. If the HGhost nodes on the fine grid have not been changed since the last smoothing step, we may even perform the correction step on those HGhost points without data transfers between processors.

### **Generation of coarse grid operator**

In the case of using Galerkin coarse grid operators, the calculation of matrix coefficients at a coarse grid node requires the coefficient data of all its neighboring fine grid nodes. The communications involved here are quite similar to those in a restriction operation. They are basically the inter-grid data transfers among processors. The amount of data to be transferred, however, is significantly larger than that in the residual restrictions.

## **7.4 Communication Reduction Techniques**

The parallel efficiency of an algorithm depends on the ratio of communication to computation time. It is directly proportional to the ratio of volume to surface area for each subdomain. In a multigrid cycle, the volume-to-surface area ratio decreases as the grid becomes increasingly coarser. In addition, as

fewer and fewer grid points are mapped onto multiple processors, more and more processors are left without any grid point to compute on very coarse grids. Whereas the idling processors on very coarse grids appear to be the main problem at first sight, experience and theoretical considerations demonstrate that the large communication overhead on the coarse grids is usually of greater concern. Special techniques have been developed to reduce coarse level communications, e.g., coarse grid agglomeration and employing different cycle schemes. However, as long as one demands that the results of a parallel algorithm be identical with those of a sequential algorithm, a substantial reduction in the total communication costs will not be achieved.

In a simple but practically useful communication model, the time required for sending a message of length  $L$  in one packet is modeled by (Trottenberg 2001),

$$t_{comm} = \alpha + \beta L, \quad (7.1)$$

where  $\alpha$  is the start-up time for communication that must be expended whenever a message is sent;  $\beta$  represents the time necessary to transfer one word, and it depends on the bandwidth of the respective communication channel. For a realistic evaluation of the performance of a solution method on a particular parallel system,  $t_{comm}$  must be compared with the computing time  $t_{comp}$  needed, e.g., for an arithmetic operation. Employing this model, the overall time spent in communication is determined by the architectural parameters  $\alpha$  and  $\beta$ , the total number of words to be transferred, and the number of messages to be sent. Given a parallel computer, it is always useful, even necessary, to

take the size of  $\alpha$  and  $\beta$  into account when an algorithm is to be parallelized. A rule of thumb frequently used is as follows: If  $\alpha$  is large, the number of messages should be minimized, while, if  $\beta$  is large, the communication volume (the total number of words) is the primary concern. Here we present several simple and easily implemented techniques to reduce linear solver parallelization overhead by reducing both the number of messages and the number of words to be transferred.

1. Inter-grid (VGhost) communications from fine to coarse grid may be replaced by intra-grid (HGhost) communications on the coarse grid. For example, in a restriction operation, to handle the dependency of a coarse grid point on its fine grid neighbors that have been assigned to other processors, one may simply send those fine grid residuals to the processor that owns the coarse grid point and performs the final weighted summation. However, a more efficient approach is to let each individual processor compute its own share of the restricted defect and then send its share to the processor where the coarse grid node belongs. In doing so, we not only distribute the arithmetic operation (multiplication and summation in a restriction operator) over more processors but also reduce the amount of data to be sent and received. The total number of messages in communication stays the same, though. The same strategy may be applied to the calculation of Galerkin coarse grid operators.
2. On coarse grid levels the start-up time  $\alpha$  becomes increasingly dominant

in the total communication time due to the decreasing number of grid points. To overcome this problem, one may pack several variables in one packet instead of sending one variable at a time.

3. If a multi-color Gauss-Seidel is used for smoothing the data volume  $L$  in (7.1) may be reduced by the corresponding multi-color update after each fractional sweep.
4. Communications overhead is incurred by message handling and network latency, as shown in (7.1). The latter may be partially or completely hidden by overlapping communication with computation. For example, in a displacement decomposition algorithm the displacement components are completely decoupled from each other. Thus, RB line Gauss-Seidel iterations may be performed independently for each component, and one component may be relaxed while a processor is waiting for incoming data for other components.

## 7.5 Numerical Example 1

The first example is a 3D waterflooding problem. The reservoir has a dimension of  $76,800 \text{ ft} \times 76,800 \text{ ft} \times 1059 \text{ ft}$ . It is discretized into  $256 \times 256 \times 22$  grid cells with  $\Delta x = \Delta y = 300 \text{ ft}$  in the horizontal directions and a varying thickness in the vertical direction. In total, there are around 1.52 million grid points and 8.88 million unknowns. Porosity, permeability, Young's modulus and Poisson's ratio are all heterogeneous by layers. The initial reservoir pres-

sure is 3500 psi, and the initial oil saturation is 0.8. No free gas is present. A water injection well is located in one corner of the reservoir, and twenty-five production wells spread out over the entire reservoir. The injection well injects 4000 stb of water per day. Each production well produces 2000 stb of oil per day. Linear solver tolerance for the elasticity system is  $10^{-4}$ . Nonlinear iteration tolerance for the coupled system is  $10^{-5}$ . The purpose of testing this case is to evaluate the parallel performance of the SCMG method as defined in Algorithm 5.

Numerical experiments are performed on a Cray-Dell Linux cluster at the Texas Advanced Computing Center (<http://www.tacc.utexas.edu>). The cluster employs of 600 3.06 GHz Xeon processors, including 282 Dell dual-processor PowerEdge 1750 compute nodes, 16 Dell dual-processor PowerEdge 2650 compute-I/O server-nodes and 2 Dell dual-processor PowerEdge 2650 login/development nodes. Each compute node has 2GB memory. A Myrinet-2000 switch fabric interconnects the nodes using PCI-X interfaces. The network has a point-point bandwidth of 250MB/sec. Here we report the number of linear iterations, the CPU time  $T_p$  on  $p$  processors, the parallel speed-up  $S_p = T_1/T_p$  and the parallel efficiency  $E_p = S_p/p$ . Since the memory of one processor is not enough for this particular application, we compute  $T_1$  ideally by  $T_1 = T_8 \times 8$  where eight is the minimum number of processors required to run the example. Total CPU time and total number of linear solver iterations for the first three time steps are presented.

As we mentioned earlier, communication cost is the primary concern

	SCMG					
	3 levels			6 levels		
$p$	Iterations	$T_p$ (sec.)	$E_p$	Iterations	$T_p$ (sec.)	$E_p$
8	221	1063.2	1.000	197	954.0	1.000
16	221	517.0	1.030	197	468.2	1.020
24	221	349.0	1.020	197	316.3	1.000
32	221	261.0	1.020	197	240.9	0.990
48	221	176.7	1.000	197	160.0	0.994
64	221	133.0	1.000	197	122.5	0.974
80	221	108.9	0.977	197	100.5	0.949
128	221	77.7	0.855	197	72.9	0.819

Table 7.1: Example 1: Comparison of SCMG with different number of coarse grid levels.

in the parallel implementation of a multigrid algorithm. This is especially true for a SCMG scheme that is featured by the low arithmetical complexity. Besides the idling processors, network latency may finally dominate the arithmetic operations on very coarse grids. This may result in a significant loss of efficiency for the overall performance of the SCMG method if coarse grid communications are not handled properly.

Table 7.1 shows the comparison results for SCMG with three and six coarse grid levels, respectively. In the later case, the coarsest grid is  $2 \times 2 \times 1$  with at most 4 processors while the original grid is  $256 \times 256 \times 22$  with as many as 128 processors. Obviously, large amount of data communications is involved in the coarsening process. However, numerical results in Table 7.1 show that the parallel efficiency is nearly 95%, even with six multigrid levels, if no more than 80 processors are employed. In the case of 128 processors,

the loss of efficiency is due to the fact that we do not scale up the size of the problem as we increase the number of processors, and, as a result, the computational intensity on each processor is getting lower as compared to the increasing amount of data communications.

Note further that the number of coarse grid levels does affect the convergence rate of the SCMG method. For instance, with three multigrid levels, our linear solver takes 221 iterations to converge while it takes only 197 iterations with six multigrid levels. However, as the number of processors increases, the CPU time gained by faster numerical convergence rate is offset by the larger communication overhead on those additional coarse grids. Thus, the overall performance of SCMG with different number of grid levels demonstrates only marginal difference if more than 32 processors participate in the computation.

The displacement decomposition (DiD) preconditioners are also tested using this numerical example. On a parallel system, these preconditioners are implemented as a combination of DiD and DD methods, as defined in (6.26) and (6.27). Numerical results (total number of linear iterations, total CPU time, parallel efficiency and parallel speed-up) for DiD-DD-AMG(1,1,1) are plotted in Figure 7.3 with varying number of processors. As expected, its numerical convergence rate deteriorates rapidly as more and more processors participate in the computation. This is shown clearly by the increasing linear iteration counts in Figure 7.3. Degradation is due to the lack of global coarse grid corrections after the one-level DD smoothing. This observation motivates our investigation of new multilevel schemes using DiD-AMG as the

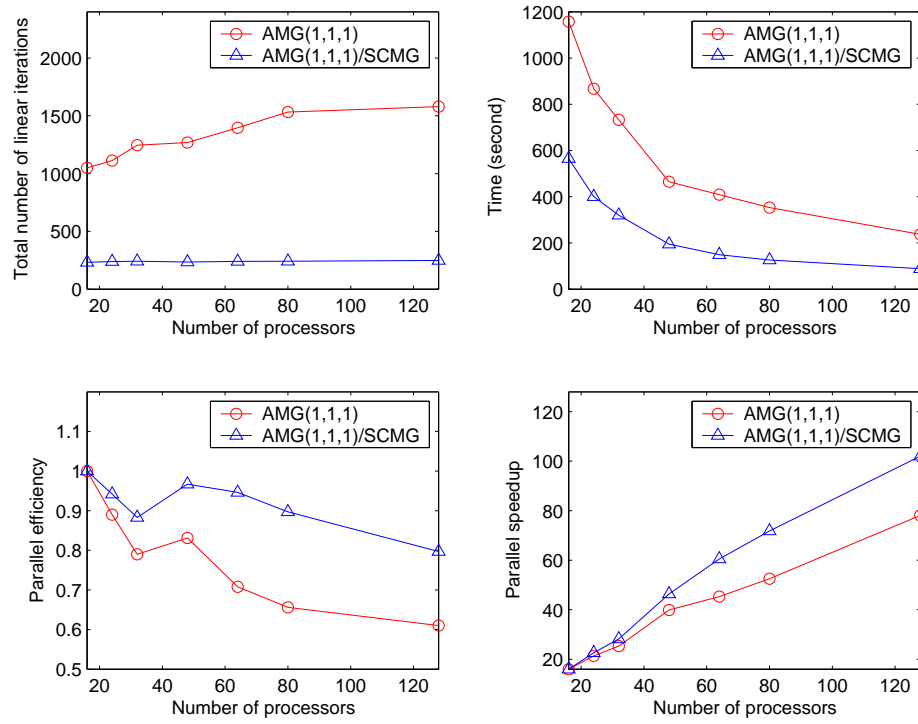


Figure 7.3: Example 1: Comparison of DiD-DD-AMG(1,1,1) with and without SCMG corrections.

first level smoother followed by a global coarse grid correction by SCMG. The corresponding results below are also plotted in Figure 7.3 for comparison.

1. The SCMG method is effective at accelerating the numerical convergence rate of the DiD-DD-AMG preconditioner. The resulting multilevel scheme (DiD-DD-AMG/SCMG) is at least three times faster than DiD-DD-AMG in terms of iteration counts.
2. The convergence deterioration experienced by the DiD-DD-AMG method is substantially improved. The number of total linear iterations is fairly constant with varying numbers of processors.
3. The new multilevel preconditioner is two times faster than the DiD-DD-AMG method.
4. Parallel efficiency is increased from 0.61 to nearly 0.8.

Experiments with DiD-DD-AMG applied to an aggregated system (DiD-DD-AMG(2,2,2)) demonstrate similar results, i.e., faster convergence rate and less convergence degradation, as shown in Figure 7.4. In the case of 128 processors, the efficiency is improved by as much as 25%. The convergence rate is almost 1.5 times faster than the DiD-DD-AMG(1,1,1)/SCMG preconditioner.

## 7.6 Numerical Example 2

In the former example we have a relatively thin reservoir with large horizontal extensions. The pancake reservoir topology results in a larger number

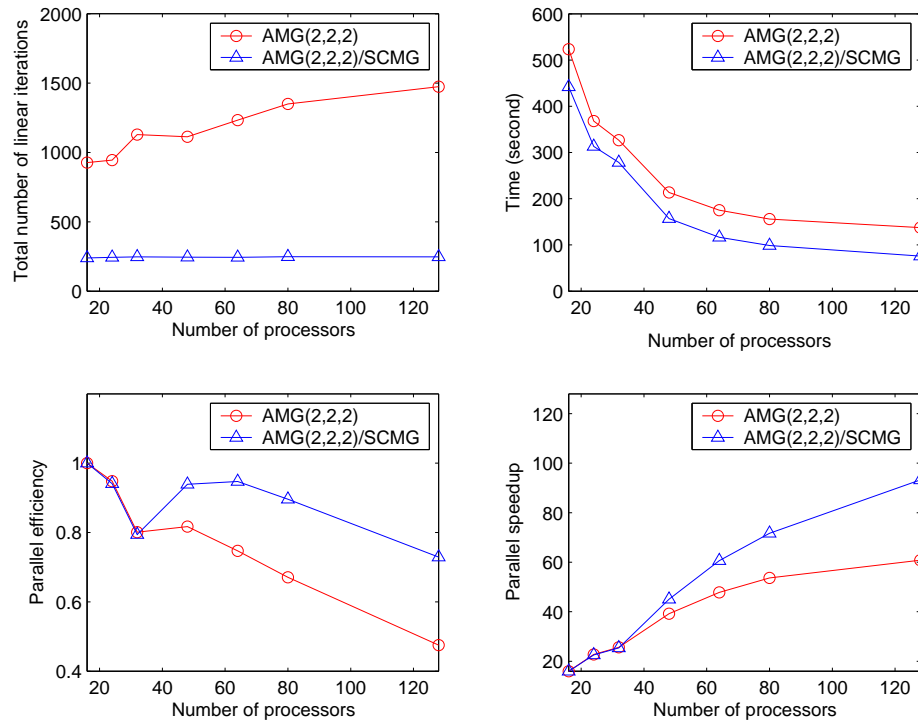


Figure 7.4: Example 1: Comparison of DiD-DD-AMG(2,2,2) with and without SCMG corrections.

of grid cells in the horizontal directions as compared to the vertical direction. Our numerical studies indicate that super-coarsening in the vertical direction is very effective at damping the low frequency aggregated components of the error. Thus, SCMG is clearly a method of choice in constructing efficient multilevel schemes for this type of reservoir application. However, the effectiveness of the SCMG method depends crucially on the relative thickness of a reservoir with respect to its areal extensions and the heterogeneity of different rock layers. In the following we present the numerical results of SCMG based on a real life application. It will be shown later that the SCMG method is less effective due to the reservoir's large thickness and strong heterogeneity.

Figure 7.5 shows a reservoir below an overburden with varying thickness. For a better characterization of the changing boundary conditions around the reservoir, we include the surrounding non-pay rocks in our calculation. The entire domain has an area of  $5,245.4 \text{ ft} \times 5,246.1 \text{ ft}$  and a thickness of  $3,275.6 \text{ ft}$ . It is uniformly discretized into  $128 \times 64 \times 111$  grid cells. In total, there are about 5.5 million degrees of freedom. The reservoir surface elevation shown in Figure 7.5 indicates a dipping reservoir. Realistic data of porosity, permeability and elastic properties are used in the simulation. Figures 7.8 and 7.9 show the spatial distribution of the Poisson's ratio and Young's modulus, respectively.

From the saturation profile shown in Figure 7.7, it can be clearly seen that there are two pay-zones inside the reservoir. We denote the upper zone by layer I and the lower zone by layer II. Well placement in each layer is based on

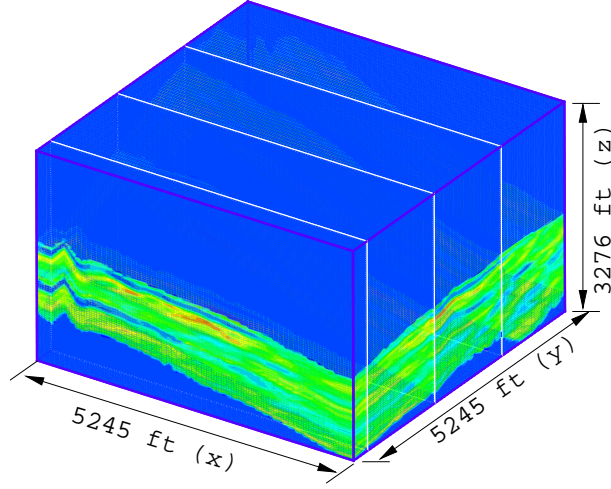


Figure 7.5: Example 2: Reservoir porosity distribution.

the in-situ oil distribution, as shown in Figures 7.10 and 7.11. There are four injection and three production wells in layer I while there are two injection and two production wells in layer II.

Numerical studies are conducted on the same cluster as used for the first example. Convergence tolerance for Newton iterations is chosen to be  $10^{-5}$ . A relative tolerance of  $10^{-4}$  is used for the linear solution of elasticity equations.

We observe that the SCMG method is ineffective for this example due to its large thickness and strong heterogeneity in the vertical direction. Thus, numerical results are only reported for MLDD-IC(0)/SCMG and DiD-DD-AMG preconditioners which are defined in (6.12) and (6.27), respectively. Tables

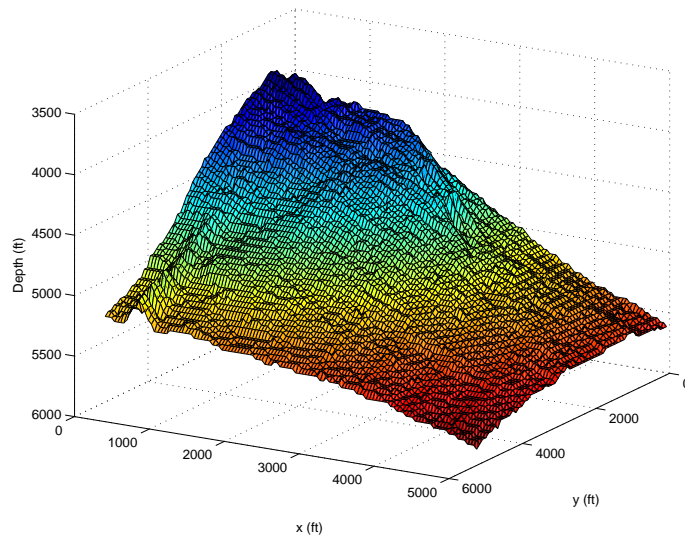


Figure 7.6: Example 2: Reservoir surface depth.

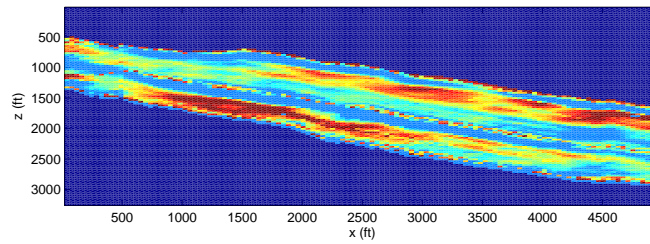


Figure 7.7: Example 2: Oil saturation profile at  $y = 2582.1$  ft.

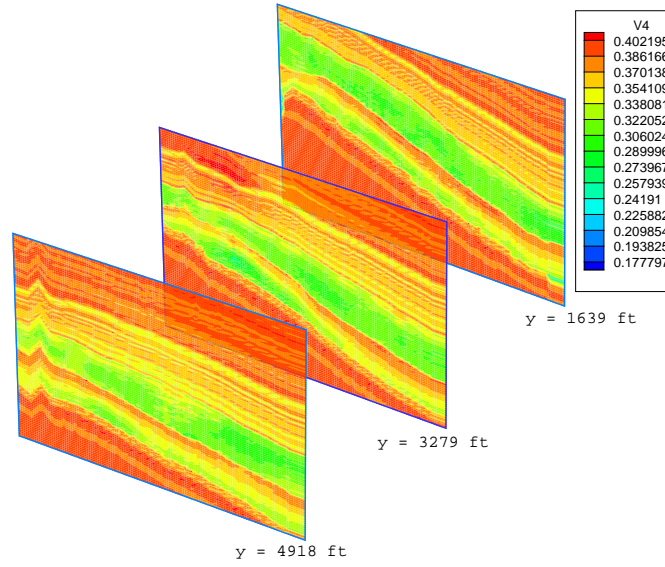


Figure 7.8: Example 2: Poisson's ratio.

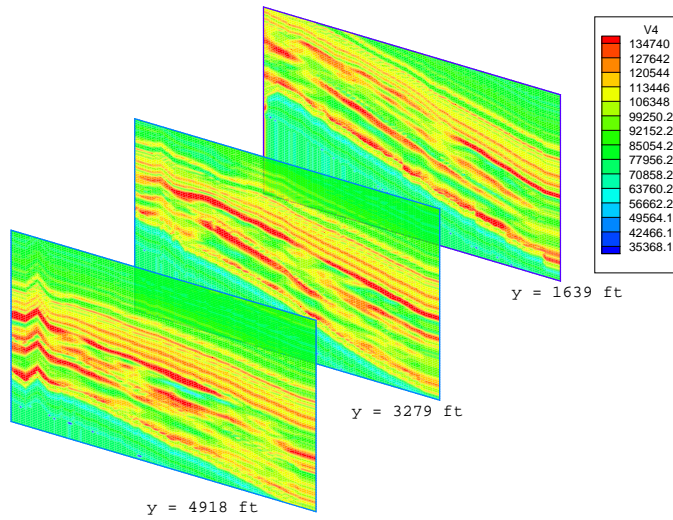


Figure 7.9: Example 2: Young's modulus.

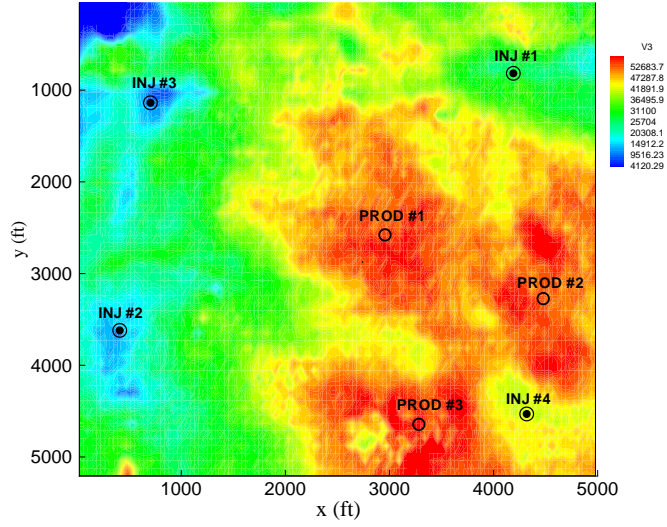


Figure 7.10: Example 2: Well distribution in layer I.

$p$	MLDD-IC(0)/SCMG	DiD-DD-AMG(1,1,1)	DiD-DD-AMG(2,2,5)
8	2228	2200	2192
16	2255	2479	2457
24	2252	2611	2611
32	2262	2780	2781
48	2286	2996	2977
64	2316	3226	3161

Table 7.2: Example 2: Comparison of total linear iteration counts for different preconditioners.

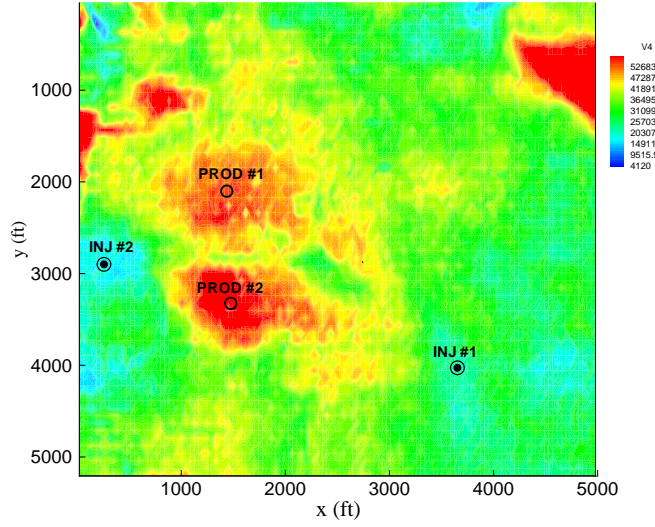


Figure 7.11: Example 2: Well distribution in layer II.

$p$	MLDD-IC(0)/SCMG	DiD-DD-AMG(1,1,1)	DiD-DD-AMG(2,2,5)
8	6075.7	3699.2	1808.9
16	3259.7	2137.1	1165.1
24	2360.0	1594.6	918.8
32	1658.0	1243.4	694.3
48	1188.7	896.4	521.1
64	946.9	705.3	417.1

Table 7.3: Example 2: Comparison of total CPU time in seconds for different preconditioners.

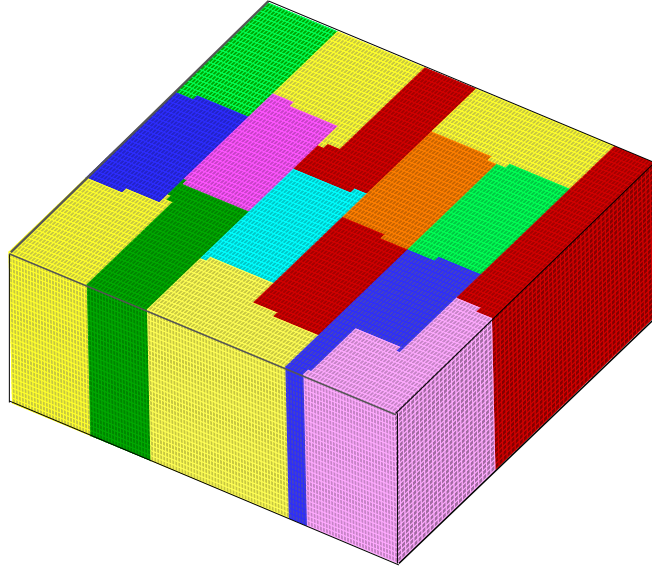


Figure 7.12: Example 2: Domain decomposition with 16 processors.

$p$	MLDD-IC(0)/SCMG	DiD-DD-AMG(1,1,1)	DiD-DD-AMG(2,2,5)
8	1.0	1.0	1.0
16	0.932	0.866	0.776
24	0.858	0.773	0.656
32	0.916	0.743	0.651
48	0.852	0.688	0.579
64	0.802	0.656	0.542

Table 7.4: Example 2: Comparison of parallel efficiency for different preconditioners.

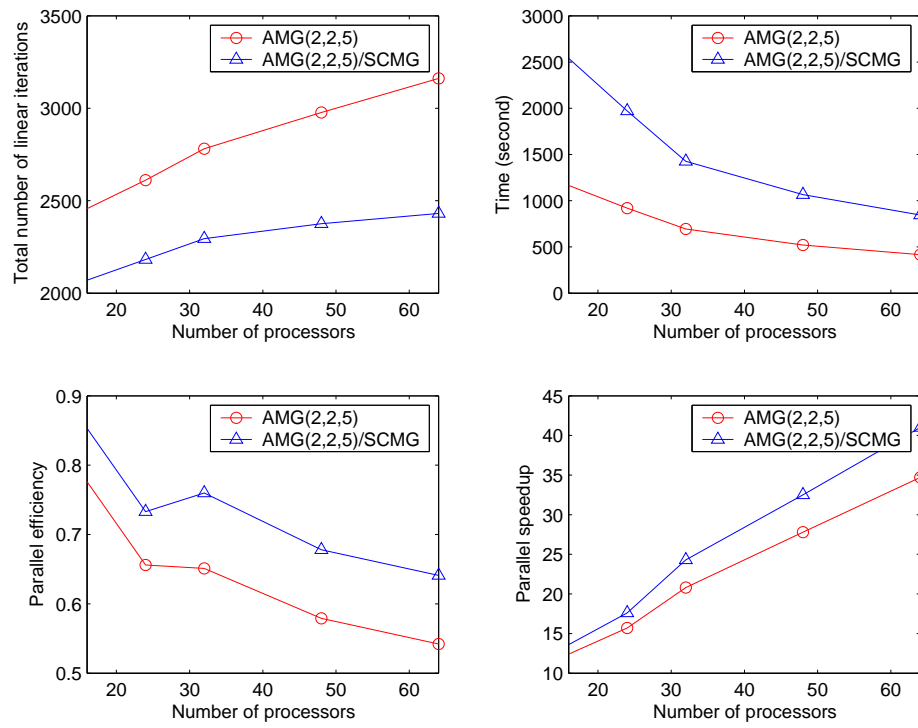


Figure 7.13: Example 2: Comparison of DiD-DD-AMG(2,2,5) with and without SCMG corrections.

7.2, 7.3 and 7.4 compare the multilevel DD and the DiD-DD preconditioners in terms of linear iteration count, total CPU time and parallel efficiency for the first ten time steps. Our observations are as follows:

1. MLDD-IC(0)/SCMG has a faster and more stable numerical convergence rate than the DiD-DD type of preconditioners. It takes less linear iterations to converge. As the number of processors increases, the method shows slight convergence degradation. This is reflected by the fairly constant linear iteration counts in Table 7.2 and the high parallel efficiency (more than 80%) in Table 7.4. However, MLDD-IC(0)/SCMG preconditioner appears to be more costly in terms of CPU time per iteration. Thus, its overall performance is still inferior to the DiD-DD type of preconditioners, as shown in Table 7.3, even though it converges numerically faster.
2. Both DiD-DD-AMG(1,1,1) and DiD-DD-AMG(2,2,5) perform consistently better than the multilevel scheme (Table 7.3). Recall that DiD-DD-AMG(2,2,5) represents applying the DiD-DD-AMG preconditioner to an aggregated coarse grid by grouping each subgrid system of  $2 \times 2 \times 5$ . It converges more than two times faster than the MLDD-IC(0)/SCMG method. A major problem of DiD-DD preconditioners is that they do not scale with an increasing number of processors. For example, in the case of 64 processors, the parallel efficiency of DiD-DD-AMG(2,2,5) is as low as 54.2%. The severe loss of scalability is mainly caused by:

- (a) Large amount of data communications incurred by using more processors.
- (b) Convergence deterioration with an increasing number of subdomains due to the lack of global coarse grid corrections (Table 7.2).
- (c) Low computation intensity as compared to large communication overhead. Applying the AMG method for solving each displacement components is shown to be fast and efficient. The communication time, thus, appears to be more significant.

In general, the poor parallel scalability due to a convergence degradation can be alleviated by applying coarse space corrections. In Figure 7.13, we do observe a reduction in the total number of linear iterations and an improved parallel efficiency and scalability. However, a comparison of the total CPU time indicates that the improved numerical convergence rate is still not fast enough to compensate the additional CPU time incurred by SCMG. It immediately follows that for some reservoir applications with large thickness and strong heterogeneity in the vertical direction, applying SCMG may be counter productive. Therefore, improving the effectiveness and efficiency of the SCMG preconditioner for such applications is the top priority of our future work.

## Chapter 8

### Concluding Remarks and Future Work

In this dissertation, a mathematical model of multiphase flow in deformable porous media is derived based on the Biot's consolidation theory. Finite element discretization and numerical schemes for solving the coupled system of equations are presented. Preconditioners for both outer and inner iterations are developed. Numerical studies are conducted to evaluate the performance of different preconditioning techniques. Parallel implementation issues of the poroelastic model are also addressed.

#### 8.1 Conclusions

The main accomplishments of this dissertation are summarized as follows:

##### **Mathematical model and numerical discretization**

- Based on the Biot's consolidation theory, a mathematical model of multiphase flow coupled with geomechanic features is derived.
- A Galerkin finite element scheme is used to discretize the poroelastic equations, and an expanded mixed finite element algorithm is used to

discretize the multiphase flow equations.

## Coupling Scheme

- An iterative coupling technique is employed to solve the two coupled field equations. The iterative method, frequently adopted by reservoir engineers in coupled geomechanics and reservoir simulation, is reformulated in a general framework. By doing so, the method can be viewed as one iteration of a preconditioned Richardson scheme applied to a fully coupled system. In addition, adding a rock compressibility term to the pressure equation is equivalent to approximating the Schur complement matrix  $\mathbf{S}$ . The resulting preconditioner is shown to be spectrally equivalent to  $\mathbf{S}$ . Thus, it is optimal in the sense that convergence rate is independent of discretization parameter  $h$ .
- Several preconditioners for the iteratively coupled technique are proposed. Their effectiveness and robustness are investigated over a wide range of fluid and rock properties such as fluid compressibility, rock permeability and jumps in material properties.
- Numerical studies indicate that preconditioners chosen on the sole base of boundary types (confined or unconfined) may not provide optimal convergence results due to reservoir heterogeneity and complex interactions between reservoir and its surroundings. A practical strategy is introduced for dynamically choosing the right preconditioner element by element. The numerical efficiency of this method is demonstrated.

- A new introduced preconditioner  $\tilde{\mathbf{S}}_{p5}$  appears to be more capable of handling jump coefficients. But its convergence behavior depends crucially on the scaling factor  $\beta$ . A practical guidance on the choice of  $\beta$  is recommended. Numerical experiments show that these approximate values of  $\beta$  lead to a convergence rate which is quite close to the one generated by an optimal value.
- A converged iteratively coupled scheme for single phase flow is developed by casting an outer loop for iterations between two field equations, namely, flow and poroelasticity. Its convergence behavior is compared to that of an iteratively coupled approach with different preconditioners. Numerical results indicate that the converged iterative scheme can accelerate the nonlinear convergence rate. But it may be less efficient than the iterative coupling for some problems.

### **Linear solvers and preconditioners for discrete elasticity system**

- Krylov subspace methods such as PCG, GMRES and BiCG-STAB are implemented for solving the algebraic linear system generated by the FE discretization of poroelastic equations. Comparison results for non-symmetric preconditioners suggest that GMRES with restart capability is the method of choice.
- One level and multilevel domain decomposition preconditioners are developed. The multilevel schemes use a super-coarsening multigrid (SCMG) for residual correction in the coarse spaces. It is characterized by low

arithmetic complexity due to a substantial reduction of unknowns. For reservoir applications with a relative small dimension in the vertical direction as compared to the areal extensions, the method is shown to be very effective in reducing the low frequency aggregated components of the error. Thus, performance of a one level scheme can be substantially improved by applying SCMG for additional corrections.

- A displacement decomposition technique is used to decouple the elasticity system into subsystems respectively for each displacement component. Applying certain quadrature rule in the numerical integration leads to a linear system which is equivalent to the one generated by a 7-point stencil finite difference scheme. An algebraic multigrid (AMG) method is then used to solve each subproblem. The so called displacement decomposition preconditioner (DiD-AMG) is shown to be very efficient and robust with respect to grid refinement, high Poisson ratio and large jumps in coefficients.
- Major drawbacks of the DiD-AMG method lie in its large memory requirement and long CPU time for initialization, which may greatly limit its practical usefulness for nonlinear poroplasticity problems. The problem is overcome in this work by applying AMG for solving the Galerkin projection of the original subsystem on an aggregated coarse grid. If the aggregated grid is not too coarse, the resulting preconditioner exhibits only slight deterioration in numerical convergence. But the initialization

time and actual solution time are well balanced. It converges at least two times faster than the original method.

### **Parallel Performance of Coupled Reservoir Flow and Geomechanics**

- A parallel version of coupled multiphase flow and geomechanics simulator is implemented in the IPARS framework. Special techniques are applied to minimize the communication overhead induced by our preconditioners, especially the SCMG method.
- For the SCMG preconditioner, numerical experiments on a synthetic case with 8.88 millions of unknowns show above 85% parallel efficiency. The performance of DiD-AMG type of methods tend to degrade as more processors are used, although they converge very fast. The convergence deterioration can be overcome or alleviated by constructing a multilevel scheme with the methods of DiD-AMG and SCMG. Parallel performance of one-level DiD preconditioners can be significantly improved.
- Ineffectiveness of the SCMG, in the case that a reservoir is thick and strongly heterogeneous, is shown by a real life application. Numerical results indicate that applying the SCMG method in such case can improve the parallel efficiency of a linear solver, but may slow down its overall convergence rate.

## 8.2 Future Work

Directions of future research suggested by this work are:

- Further investigation of preconditioners for the iterative coupling technique, which includes both theoretical analysis and numerical tests.
- Improving the performance of the SCMG method.
- Implementation of a fully coupled scheme using a Krylov subspace method and the preconditioners developed in this work.
- Adaptive time step selection with different time scales for flow and displacements.
- Mesh adaptivity.
- Adding the dependence of permeability field on pressure and stresses.
- Taking into account thermal effects in a coupled analysis for multiphase flow and developing efficient iterative scheme for solving pressure, saturations, displacements and temperature.
- Coupling of multiphase flow with poroplasticity on parallel computers.
- Coupling an existing compositional model with poroelasticity model.
- Modifying the elasticity scheme by adding discontinuous Galerkin approximation in the neighborhood of low permeability and low fluid compressibility.

## Bibliography

- [1] Y. Abousleiman, A. H.-D. Cheng, L. Cui, E. Detournay, and J.-C. Roegiers. Mandel's problem revisited. *Géotechnique*, 46(2):187–195, 1996.
- [2] M. Adams. Parallel multigrid solvers for 3D unstructured finite element problems in large deformation elasticity and plasticity. *Int. J. Num. Meth. Eng.*, 48(8):1241–1262, 2000.
- [3] M. Adams. Evaluation of three unstructured multigrid methods on 3D finite element problems in solid mechanics. *Int. J. Num. Meth. Eng.*, 55(5):519–534, 2002.
- [4] D. R. Allen. Physical changes of reservoir properties caused by subsidence and repressurizing operations, Wilmington field, California. *JPT*, pages 23–29, 1968.
- [5] D. R. Allen. Environmental aspects of oil producing operations-Long Beach, California. February 1972. SPE 30752.
- [6] T. Arbogast, M. F. Wheeler, and I. Yotov. Mixed finite elements for elliptic problems with tensor coefficients as cell-centered finite differences. *SIAM J. Numer. Anal.*, 34:828–852, 1997.

- [7] O. Axelsson. A survey of vectorizable preconditioning methods for large scale finite element matrices. In J. G. Verwer, editor, *Colloquium Topics in Applied Numerical Analysis, Syllabus 4*, CWI, pages 21–47, Amsterdam, 1983. Springer-Verlag.
- [8] O. Axelsson, S. Brinkkemper, and V. P. Il'in. On some versions of incomplete block-matrix factorization methods. *Linear Algebra Appl.*, 38:3–15, 1984.
- [9] O. Axelsson and V. L. Eijkhout. Robust vectorizable preconditioners for three-dimensional elliptic difference equations with anisotropy. In H. J. J. te Riele, Th. J. Dekker, and H. A. van der Vorst, editors, *Algorithms and Applications on Vector and Parallel Computers*, North-Holland, Amsterdam, 1987.
- [10] O. Axelsson and I. Gustafsson. Iterative methods for the solution of the Navier's equations of elasticity. *Comput. Methods Appl. Mech. Engrg.*, 15:241–258, 1978.
- [11] O. Axelsson and A. Padiy. On the additive version of the algebraic multilevel iteration method for anisotropic elliptic problems. *SIAM Journal on Scientific Computing*, 20(5):1807–1830, 1999.
- [12] O. Axelsson and B. Polman. On approximate factorization methods for block-matrices suitable for vector and parallel processors. *Linear Algebra Appl.*, 77:3–26, 1986.

- [13] O. Axelsson and P. S. Vassilevski. Algebraic multilevel preconditioning methods I. *Numerische Mathematik*, 56:157–177, 1989.
- [14] O. Axelsson and P. S. Vassilevski. Algebraic multilevel preconditioning methods II. *SIAM Journal on Numerical Analysis*, 27(6):1569–1590, 1990.
- [15] R. E. Bank and C. Douglas. Sharp estimates for multigrid rates of convergence with general smoothing and acceleration. *SIAM J. Numer. Anal.*, 22(4):617–633, 1985.
- [16] J. Bear and Y. Bachmat. *Introduction to Modeling Phenomena of Transport in Porous Media*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 1990.
- [17] M. A. Biot. Consolidation settlement under a rectangular load. *J. Appl. Phys.*, 12:426–430, 1941.
- [18] M. A. Biot. General theory of three-dimensional consolidation. *J. Appl. Phys.*, 12:155–164, 1941.
- [19] M. A. Biot. Theory of elasticity and consolidation for a porous anisotropic solid. *J. Appl. Phys.*, 26:182–185, 1955.
- [20] M. A. Biot. General solutions of the equations of elasticity and consolidation for a porous material. *J. Appl. Mech.*, 27:91–96, 1956.

- [21] M. A. Biot. Theory of deformation of a porous viscoelastic anisotropic solid. *J. Appl. Phys.*, 27:452–469, 1956.
- [22] M. A. Biot. Thermoelasticity and irreversible thermodynamics. *J. Appl. Phys.*, 27:240–253, 1956.
- [23] M. A. Biot. Mechanics of deformation and acoustic propagation in porous media. *J. Appl. Phys.*, 33:1482–1498, 1962.
- [24] M. A. Biot. Nonlinear and semilinear rheology of porous solids. *J. Geophys. Res.*, 78:4924–4937, 1973.
- [25] M. A. Biot and D. G. Willis. The elastic coefficients of the theory of consolidation. *J. Appl. Mech.*, 24:594–601, 1957.
- [26] R. Blaheta. Displacement decomposition-incomplete factorization preconditioning techniques for linear elasticity problems. *Num. Lin. Alg. Appl.*, 1:107–128, 1994.
- [27] R. Blaheta, P. Byczanski, O. Jakl, and J. Starý. Space decomposition preconditioners and their application in geomechanics. *Mathematics and Computers in Simulation*, 61(3-6):409–420, 2003.
- [28] J. R. Booker and J. C. Small. An investigation of the stability of numerical solutions of Biot’s equations of consolidation. *Int J. Solids Struct.*, 11:907–917, 1975.

- [29] D. Braess. On the combination of multigrid methods and conjugate gradients. In W. Hackbusch and U. Trottenberg, editors, *Multigrid Methods*, volume 1228 of Lecture Notes in Mathematics, pages 52–64, Berlin, 1986. Springer-Verlag.
- [30] D. Braess. *Finite Elements : Theory, Fast Solvers, and Applications in Solid Mechanics*. Cambridge University Press, Cambridge, 2nd edition, April 2001.
- [31] J. H. Bramble. *Multigrid methods*, volume 294 of Pitman Research Notes in Mathematics Series. Longman Scientific & Technical, New York, 1993.
- [32] A. Brandt. Multilevel adaptive solutions to boundary-value problem. *Math. Comp.*, 31:333–309, 1997.
- [33] F. Brezzi and M. Fortin. *Mixed and Hybrid Finite Element Methods*. Springer-Verlag, New York, 1991.
- [34] V. E. Bulgakov and M. Y. Belyi. On the multi-grid technique for solving three dimensional boundary value engineering problems. *Int. J. Num. Meth. Eng.*, 33:753–764, 1992.
- [35] V. E. Bulgakov and G. Kuhn. High-performance multilevel iterative aggregation solver for large finite element structural analysis problems. *Int. J. Num. Meth. Eng.*, 38, 1995.

- [36] T. F. Chan, S. D. Margenov, and P. Vassilevski. Performance of block-ILU factorization preconditioners based on block-size reduction for 2D elasticity systems. *SIAM J. Sci. Comput.*, 18(5):1355–1366, 1997.
- [37] T. F. Chan and B. F. Smith. Multigrid and domain decomposition on unstructured grids. In D. F. Keyes and J. Xu, editors, *Seventh Annual International Conference on Domain Decomposition on unstructured grids*, AMS, 1995.
- [38] Ph. A. Charlez. *ROCK MECHANICS volume 1: THEORETICAL FUNDAMENTALS*. Editions Technip, Paris, 1991.
- [39] H. Y. Chen. Coupled fluid flow and geomechanics in reservoir study - 1. theory and governing equations. In *the SPE Annual Technical Conference and Exhibition*, Dallas, Texas, October 22-25 1995. SPE 30752.
- [40] A. H.-D. Cheng. A direct boundary element method for plane strain poroelasticity. *Int. J. Numer. Anal. Methods. Geomech.*, 12:551–572, 1988.
- [41] L. Y. Chin and R. R. Boade. Full-field, 3-D finite element subsidence model for Ekofisk. In *3rd North Sea Chalk Symposium*, Copenhagen, June 11-12 1990.
- [42] L. Y. Chin, R. Raghavan, and L. K. Thomas. Fully-coupled geomechanics and fluid-flow analysis of wells with stress-dependent permeability. In

*the 1998 SPE International Conference and Exhibition*, Beijing, China, November 2-6 1998.

- [43] P. G. Ciarlet. *Mathematical Elasticity, volume 1: Three Dimensional Elasticity*. North-Holland, New York, 1988.
- [44] National Research Council. *Mitigating Losses from Land Subsidence in the United States*. National Academy Press, Washington D.C., 1991.
- [45] O. Coussy. A general theory of thermoporoelastoplasticity for saturated porous materials. *Transport in Porous Media*, 4:281–293, 1989.
- [46] C. Cryer. A comparison of the three-dimensional consolidation theories of Biot and Terzaghi. *Q. J. Mech. Appl. Math.*, 16:401–412, 1963.
- [47] C. N. Dawson, H. Klie, M. F. Wheeler, and C. S. Woodward. A parallel, implicit, cell-centered method for two-phase flow with preconditioned Newton-Krylov solver. *Journal on Computational Geosciences*, 1:215–249, 1997.
- [48] R. H. Dean. A poroelastic multicomponent reservoir simulator. *unpublished*, 2000.
- [49] R. H. Dean, X. Gai, C. M. Stone, and S. E. Minkoff. A comparison of techniques for coupling porous flow and geomechanics. In *the SPE Reservoir Simulation Symposium*, Houston, Texas, February 3-5 2003. SPE 79709.

- [50] E. Detournay and A. Cheng. *Fundamentals of Poroelasticity*, volume II. Pergaman Press, Dordrecht, The Netherlands, 1993.
- [51] ARCO Reservoir Simulator Development. Appendix 30: Poroelastic calculations. In *ACRES Reference Manual*, ARCO, Plano, Texas, 1999.
- [52] H. C. Edwards. A parallel multilevel-preconditioned GMRES solver for multi-phase flow models in the Implicit Parallel Accurate Reservoir Simulator (IPARS). Technical Report TICAM 98-04, The University of Texas at Austin, 1998.
- [53] H. C. Elman and G. H. Golub. Inexact and preconditioned Uzawa algorithms for saddle point problems. *SIAM J. Numer. Anal.*, 31:1645–1661, 1994.
- [54] C. Farhat. A method of finite element tearing and interconnecting and its parallel solution algorithm. *Int. J. Num. Meth. Eng.*, 32:1205–1227, 1991.
- [55] C. Farhat, M. Lesoinne, and K. Pierson. A scalable dual-primal domain decomposition method. *Numer. Linear Algebra Appl.*, 7:687–714, 2000.
- [56] C. Farhat, M. Lesoinne, P. Le Tallec, K. Pierson, and D. Rixen. FETI-DP: a dual-primal unified FETI method - part I: a faster alternative to the two-level FETI method. *Int. J. Numer. Methods Engrg.*, 50:1523–1544, 2001.

- [57] C. Farhat and J. Mandel. The two-level FETI method for static and dynamic plate problems - part I: an optimal iterative solver for biharmonic systems. *Comput. Methods Appl. Mech. Engrg.*, 155:129–152, 1998.
- [58] I. Fatt and D. H. Davis. Reduction in permeability with overburden pressure. *Trans. AIME*, 195:329–341, 1952.
- [59] Y. T. Feng, D. Peric, and D. R. J. Owen. A non-nested Glerkin multi-grid method for solving linear and nonlinear solid mechanics problems. *Comput. Methods Appl. Mech. Engrg.*, 144:307–325, 1997.
- [60] Y. T. Feng, D. Peric, and D. R. J. Owen. A multi-grid enhanced GMRES algorithm for elasto-plastic problems. *Int. J. Num. Meth. Eng.*, 42:1441–1462, 1998.
- [61] E. J. Fielding, R. G. Blom, and R. M. Goldstein. Rapid subsidence over oil fields measured by SAR interferometry. *Geophysical Research Letters*, 25(17):3215–3218, 1998.
- [62] J. Fish, M. Pandheeradi, and V. Belsky. An efficient multilevel solution scheme for large scale non-linear systems. *Int. J. Num. Meth. Eng.*, 38:1597–1610, 1995.
- [63] J. T. Fredrich, J. G. Arguello, G. L. Deitrick, and E. P. de Rouffignac. Geomechanical modeling of reservoir compaction, surface subsidence,

and casing damage at the Belridge diatomite field. *SPE Reservoir Eval. and Eng.*, 3(4):348–359, August 2000.

- [64] J. T. Fredrich, J. G. Arguello, B. J. Thorne, W. R. Wawersik, G. L. Deitrick, E. P. de Rouffignac, L. R. Myer, and M. S. Bruno. Three-dimensional geomechanical simulation of reservoir compaction and implications for well failures in the Belridge diatomite. In *the 1996 SPE Annual Technical Conference and Exhibition*, pages 195–210, Denver, Colorado, October 6-9 1996. SPE 36698.
- [65] L. S. K. Fung, L. Buchanan, and R. G. Wan. Coupled geomechanical-thermal simulation for deforming heavy-oil reservoirs. *J. Cdn. Pet. Tech.*, 22, April 1994.
- [66] X. GAI, R. H. Dean, M. F. Wheeler, and R. Liu. Coupled geomechanical and reservoir modeling on parallel computers. In *the SPE Reservoir Simulation Symposium*, Houston, Texas, February 3-5 2003. SPE 79700.
- [67] J. Geertsma. The effect of fluid pressure decline on volumetric changes of porous rocks. *Trans. AIME*, 210:331–340, 1957.
- [68] J. Ghaboussi and E. L. Wilson. Flow of compressible fluid in porous elastic media. *Int. J. Num. Meth. Engry.*, 5:419–442, 1973.
- [69] R. E. Gibson, K. Knight, and P. W. Taylor. A critical experiment to examine theories of three-dimensional consolidation. In *Proc. Eur. Conf. Soil Mech.*, volume 1, pages 69–76, Wiesbaden, 1963.

- [70] D. H. Gray, I. Fatt, and G. Bergamini. The effect of stress on permeability of sandstones cores. *SPEJ*, pages 95–100, June 1963.
- [71] M. Griebel, D. Oeltz, and M. A. Schweitzer. An algebraic multigrid method for linear elasticity. *SIAM Journal on Scientific Computing*, 25(2):385–407, 2003.
- [72] I. Gustafsson and G. Lindskog. On parallel solution of linear elasticity problems. part I: theory. *Numer. Lin. Algebra. Appl.*, 5(2):123–139, 1998.
- [73] I. Gustafsson and G. Lindskog. On parallel solution of linear elasticity problems. part II: Methods and some computer experiments. *Numer. Lin. Algebra. Appl.*, 9(3):205–221, 2002.
- [74] M. Gutierrez and R. W. Lewis. The role of geomechanics in reservoir simulation. In *the 1998 SPE/ISRM Eurock Conference*, Trondheim, July 8-10 1998. SPE/ISRM 47392.
- [75] W. Hackbusch. *Multi-grid methods and applications*. Springer, Berlin, Germany, 1985.
- [76] K. S. Hansen, M. Prats, and C. K. Chan. Modeling of reservoir compaction and surface subsidence at south Belridge. In *The 1993 SPE Western Regional Meeting*, Anchorage AK, May 26-28 1993. SPE 26074.
- [77] H. Hermansen, L. K. Thomas, and B. T. Aasboe. Twenty five years of Ekofisk reservoir management. In *The 1997 SPE Annual Technical*

- Conference and Exhibition*, San Antonio, Texas, October 5-8 1997. SPE 38927.
- [78] R. M. Holt. Permeability reduction induced by a nonhydrostatic stress field. *SPEFE*, pages 444–448, December 1990.
  - [79] J. C. Jaeger and N. Cook. *Fundamentals for Rock Mechanics*. Chapman and Hall, London, 1979.
  - [80] S. Kacau and I. D. Parsons. A parallel multigrid method for history-dependent elastoplasticity computations. *Comput. Methods Appl. Mech. Engrg.*, 108:1–21, 1993.
  - [81] R. Kettler. Analysis and comparison of relaxation schemes in robust multigrid and preconditioned conjugate gradient methods. In W. Hackbusch and U. Trottenberg, editors, *Multigrid Methods*, volume 960 of *Lecture Notes in Mathematics*, pages 502–534, Berlin, 1982. Springer-Verlag.
  - [82] R. Kettler and J. A. Meijerink. A multigrid method and a combined multigrid-conjugated gradient method for elliptic problems with strongly discontinuous coefficients in general domains. Technical Report 604, Shell Oil Company, 1981.
  - [83] J. Killough and M. Wheeler. Parallel iterative linear equation solvers: An investigation of domain decomposition algorithms for reservoir simu-

- lation. In *the SPE 1987 Reservoir Simulation Symposium*, San Antonio, Texas, February 1-4 1987. SPE 16021.
- [84] H. Klie. *Krylov-Secant Methods for Solving Large Scale Systems of Coupled Nonlinear Parabolic Equations*. PhD thesis, Rice University, Houston, Texas, 1996.
- [85] N. C. Koutsabeloulis and S. A. Hope. Coupled stress/fluid/thermal multi-phase reservoir simulation studies incorporating rock mechanics. In *SPE/ISRM Eurock'98*, Trondheim, Norway, July 5-10 1998. SPE/ISRM 47393.
- [86] S. Lacroix, Yu. Vassilevski, and M.F. Wheeler. Iterative solvers of the implicit parallel accurate reservoir simulator, I: Single processor case. Technical Report TICAM 00-28, The University of Texas at Austin, 2000.
- [87] S. Lacroix, Yu. Vassilevski, and M.F. Wheeler. Decoupling preconditioners in the implicit parallel accurate reservoir simulator (IPARS). *Numer. Linear Algebra Appl.*, 8(8):537–549, 2001.
- [88] S. Lang, Christian Wieners, and G. Wittum. Rock mechanics volume 1: Theoretical fundamentals. In Erwin Stein, editor, *Error-Controlled Adaptive Finite Element Methods in Solid Mechanics*. John Wiley & Sons Ltd, 2000.

- [89] F. H. Lee, K. K. Phoon, K. C. Lim, and S. H. Chan. Performance of Jacobi preconditioner in Krylov sub-space solution of finite element equations. *Intl. J. for Num. and Anal. Methods in Geomech.*, 26(4):341–372, April 2002.
- [90] R. W. Lewis and B. A. Schrefler. *The Finite Element Method in The Static and Dynamic Deformation and Consolidation of Porous Media*. John Wiley & Sons, New York, second edition, 1998.
- [91] R. W. Lewis and Y. Sukirman. Finite element modeling of three-phase flow in deforming saturated oil reservoirs. *Intl. J. for Num. and Anal. Methods in Geomech.*, 17:577, 1993.
- [92] X. Li and O. C. Zienkiewicz. Multiphase flow in deforming porous media and finite element solution. *Computers and Structures*, 45:211, 1992.
- [93] K. Lipnikov. *Numerical Methods for the Biot Model in Poroelasticity*. PhD thesis, University of Houston, Houston Texas, 2002.
- [94] I. Lirkov. MPI solver for 3D elasticity problems. *Mathematics and Computers in Simulation*, 61(3-6):509–516, 2003.
- [95] R. Liu. *Discontinuous Galerkin for Mechanics*. PhD thesis, The University of Texas at Austin, Austin, Texas, 2004.
- [96] Q. Lu, M. Peszyńska, and X. Gai. Implicit black-oil model in IPARS framework. Technical Report TICAM 01-33, The University of Texas at Austin, 2001.

- [97] Qin Lu. *A Parallel Multi-Block/Multi-Physics Approach for Multi-Phase Flow in Porous Media*. PhD thesis, University of Texas at Austin, Austin, Texas, 2000.
- [98] J. Mandel. Consolidation des sols (étude mathématique). *Géotechnique*, 3:287–299, 1953.
- [99] A. S. McLatchie, R. A. Hemstock, and J. W. Young. The effective compressibility of reservoir rock and its effects on permeability. *Trans. AIME*, 213:386–388, 1958.
- [100] M. D. Mihajlović and S. Mijalković. A component decomposition preconditioning for 3D stress analysis problems. *Numer. Linear Algebra Appl.*, 00:1–16, 2002.
- [101] S. Minkoff, C. M. Stone, J. G. Arguello, S. Bryant, J. Eaton, M. Peszyńska, and M. F. Wheeler. Staggered in time coupling of reservoir flow simulation and geomechanical deformation: Step 1 - one-way coupling. In *the 1999 SPE Reservoir Simulation Symposium*, Houston, Texas, 1999. SPE 51920.
- [102] N. Morita. Rock-property changes during reservoir compaction. *SPEFE*, pages 197–205, September 1992.
- [103] J. G. Osorio, H. Y. Chen, and L. W. Teufel. Fully coupled fluid-flow and geomechanics simulation of stress-sensitive reservoirs. In *the Reservoir Simulation Symposium*, Dallas, Texas, June 8-11 1997. SPE 38023.

- [104] J. G. Osorio, H. Y. Chen, and L. W. Teufel. Numerical simulation of the impact of flow-induced geomechanics response on the productivity of stress-sensitive reservoirs. In *the SPE Reservoir Simulation Symposium*, Houston, Texas, February 14-17 1999. SPE 51929.
- [105] A. Padiy. On a robust multilevel method applied for solving large-scale linear elasticity problems. *Commun. Numer. Meth. in Engng.*, 15:153–165, 1999.
- [106] M. Parashar, J. A. Wheeler, J. C. Browne, G. Pope, K. Wang, and P. Wang. A new generation EOS compositional reservoir simulator: part II - framework and multiprocessing. In *the 1997 SPE Reservoir Simulation Symposium*, 1997. SPE 37977.
- [107] M. Peszyńska, M. F. Wheeler, and I. Yotov. Mortor upscaling for multiphase flow in porous media. *Computational Geosciences*, 6:73–100, 2002.
- [108] P. J. Phillips and M. F. Wheeler. A coupling of mixed and Galerkin finite element methods for poro-elasticity. 2003.
- [109] K. K. Phoon, K. C. Toh, S. H. Chan, and F. H. Lee. An efficient diagonal preconditioner for finite element solution of Biot’s consolidation equations. *Intl. J. for Num. and Anal. Methods in Geomech.*, 55(4):377–400, Oct 2002.

- [110] J. H. Prevost. Partitioned solution procedure for simultaneous integration of coupled-field problems. *Communications in Numerical Methods in Engineering*, 13(4):239–247, 1997.
- [111] R. A. Raviart and J. M. Thomas. A mixed finite element. In *Springer Lecture Notes of Mathematics*, volume 606, pages 292–315, New York, 1977. Springer Verlag.
- [112] D. W. Rhet and L. W. Teufel. Effect of reservoir stress path on compressibility and permeability of sandstones. In *the SPE Annual Technical Conference and Exhibition*, Washington, DC., October 4-7 1992. SPE 24756.
- [113] J. R. Rice and M. P. Cleary. Some basic stress diffusion solutions for fluid-saturated elastic porous media with compressible constituents. *Rev. Geophys. Space Phys.*, 14:227–241, 1976.
- [114] J. Ruge. AMG for problems of elasticity. *Applied Mathematics and Computation*, 23:293–309, 1986.
- [115] J. W. Ruge and K. Stüben. Efficient solution of finite difference and finite element equations by algebraic multigrid. In D.J. Paddon and H. Holstein, editors, *Multigrid Methods for Integral and Differential Equations*. The Institute of Mathematics and its Applications Conference Series, Clarendon Press, 1985.

- [116] J. W. Ruge and K. Stüben. Algebraic multigrid (AMG). In S. F. McCormick, editor, *Multigrid Methods*, volume 3 of *Frontiers in Applied Mathematics*, pages 73–130, Philadelphia, PA, 1987.
- [117] T. F. Russell and M. F. Wheeler. Finite element and finite difference methods for continuous flows in porous media. In R. E. Ewing, editor, *The Mathematics of Reservoir Simulation*, pages 35–106, Philadelphia, 1983. SIAM.
- [118] Y. Saad and M. Schultz. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM J. Sci. Stat. Comput.*, 7:856–869, 1986.
- [119] R. S. Sanhu and E. L. Wilson. Finite element analysis of seepage in elastic media. *J. Engrg. Mech. Div. Amer. Soc. Civil Engrg.*, 95:641–652, 1969.
- [120] A. Settari and F. M. Mourits. Coupling of geomechanics and reservoir simulation models. In Siriwardane and Zeman, editors, *Comp. Methods and Advances in Geomech.*, pages 2151–2158, Balkema, Rotterdam, 1994.
- [121] A. Settari and D. A. Walters. Advances in coupled geomechanics and reservoir modeling with applications to reservoir compaction. In *the SPE Reservoir Simulation Symposium*, Houston, Texas, February 14-17 1999. SPE 51927.

- [122] A. W. Skempton. The pore pressure coefficients A and B. *Géotechnique*, 4:143–147, 1954.
- [123] J. C. Small, J. R. Booker, and E. H. Davis. Elastoplastic consolidation of soil. *Int. J. Solids Struct.*, 12:431–448, 1976.
- [124] B. F. Smith, P. E. Bjorstad, and W. D. Gropp. *Domain Decomposition: Parallel Multilevel Methods for Elliptic Partial Differential Equations*. Cambridge University Press, New York, 1996.
- [125] T. W. Stone, C. Xian, Z. Fang, E. Manalac, R. Marsden, and J. Fuller. Coupled geomechanical simulation of stress dependent reservoirs. In *the SPE Reservoir Simulation Symposium*, Houston, Texas, February 3-5 2003. SPE 79697.
- [126] K. Stüben. Algebraic multigrid (AMG): experiences and comparisons. *Appl. Math. Comput.*, 13:419–452, 1983.
- [127] K. Stüben. Some studies on algebraic multigrid (AMG). In *International Workshop on Algebraic Multigrid Methods*, St. Wolfgang, June 26-28 2000.
- [128] R. M. Sulak and J. Danielsen. Reservoir aspects of Ekofisk subsidence. *JPT*, pages 709–716, July 1989.
- [129] R. M. Sulak, L. K. Thomas, and R. R. Boade. 3D reservoir simulation of Ekofisk compaction drive. In *JPT*, pages 1272–1278, Oct 1991.

- [130] K. Terzaghi. *Theoretical Soil Mechanics*. Wiley, New York, 1943.
- [131] S. Timoshenko and J. N. Goodier. *Theory of Elasticity*. McGraw-Hill, New York, 3rd edition, 1969.
- [132] W. S. Tortike and S. M. Farouq Ali. A framework for multiphase non-isothermal fluid flow in a deforming heavy oil reservoir. In *the SPE Reservoir Simulation Symposium*, San Antonio, February 1-4 1987. SPE 16030.
- [133] W. S. Tortike and S. M. Farouq Ali. Reservoir simulation integrated with geomechanics. In *Proc. CIM Annual Tech. Conf.*, pages 39–1–39–20, Calgary, Canada, June 7-10 1992.
- [134] U. Trottenberg, C. W. Oosterlee, and A. Schü ller. *Multigrid*. Academic Press, San Deigo, 2001.
- [135] P. Vanek, J. Mandel, and M. Brezina. Algebraic multigrid by smoothed aggregation for second and fourth order elliptic problems. In *7th Copper Mountain Conference on Multigrid Methods*, 1995.
- [136] P. Vaněk, J. Mandel, and M. Brezina. Algebraic multigrid on unstructured meshes. Technical Report UCD/CCM 34, University of Colorado at Denver, 1994.
- [137] Yu. Vassilevski. Iterative solvers for the implicit parallel accurate reservoir simulator (IPARS), II: Parallelization issues. Technical Report TICAM 00-33, The University of Texas at Austin, 2000.

- [138] A. Verruijt. Discussion. In *Proc. 6th Int. Conf. Soil Mech.*, volume 3, pages 401–402, Montreal, 1965.
- [139] A. Verruijt. Elastic storage of aquifers. In R. De Wiest, editor, *Flow Through Porous Media*, New York, 1969. Academic Press.
- [140] H. Van Der Vorst. BiCG-STAB: a fast and smoothly convergent variant of Bi-CG for the solution of nonsymmetric linear systems. *SIAM J. Sci. Stat. Comput.*, 13:631–644, 1992.
- [141] J. R. Wallis, R. P. Kendall, and T. E. Little. Constrained residual acceleration of conjugate residual methods. In *the SPE 1985 Reservoir Simulation Symposium*, Dallas, Texas, February 10-13 1985. SPE 13536.
- [142] N. R. Warpinski and L. W. Teufel. Determination of the effective-stress law for permeability and deformation in low-permeability rocks. In *the SPE Annual Technical Conference and Exhibition*, New Orleans, September 23-26 1990. SPE 20572.
- [143] A. Wathen and D. Silvester. Fast iterative solution of stabilized Stokes systems. part I: Using simple diagonal preconditioners. *SIAM J. Numer. Anal.*, 30:630–649, 1993.
- [144] J. W. Watts. An iterative matrix solution method suitable for anisotropic problems. *Soc. Pet. Eng. J.*, pages 47–51, 1971.

- [145] J. W. Watts. A method of improving line successive overrelaxation in anisotropic problems - a theoretical analysis. *Soc. Pet. Eng. J.*, pages 105–118, 1973.
- [146] A. Weiser and M. F. Wheeler. On convergence of block-centered finite-difference for elliptic problems. *SIAM J. Numer. Anal.*, 25:351–375, 1988.
- [147] J. Wheeler. Integrated parallel accurate reservoir simulator (IPARS). In *the 8th Annunial Industrial Affiliate Meeting*, The University of Texas at Austin, October 27-28 1998. Center of Subsurface Modeling.
- [148] J. Wheeler and R. Smith. Reservoir simulation on a hypercube. In *the SPE 1989 Ann. Tech. Conf.*, San Antonio, Texas, October 8-11, 1989. SPE 19804.
- [149] M. Wheeler, M. Peszyńska, X. Gai, and O. El-Domeiri. Modeling subsurface flow on PC clusters. In *High Performance Computing 2000*, April 2000.
- [150] W. A. White and R. A. Morton. Wetland losses related to fault movement and hydrocarbon production, southeastern Texas coast. *Journal Of Coasted Research*, 13(4):1305–1320, 1997.
- [151] D. O. Wyble. Effect of applied pressure on the conductivity, porosity and permeability. *Trans. AIME*, 213:430–432, 1958.

- [152] J. Xu. *Theory of Multilevel Methods*. PhD thesis, Cornell University, Ithaca, NY, 1998.
- [153] I. Yotov. *Mixed finite element methods for flow in porous media*. PhD thesis, Rice University, Houston Texas, 1996.
- [154] O. C. Zienkiewicz, A. H. C. Chan, M. Pastor, B. A. Schrefler, and T. Shiomi. *Computational Geomechanics with special reference to earthquake engineering*. John Wiley & Sons, New York, 1999.

## Vita

Xiuli Gai, the daughter of Wenxiang Gai and Guiqin Wang, was born in Jiaohe, China on September 20, 1970. She graduated with a Bachelor of Science degree in Geology from Daqing Petroleum Institute, Anda, China, in July 1992. She received a Master of Science degree in Petroleum Geology from the same institute in May 1995. In June 1995, she subsequently joined China National Offshore Oil Company, Beijing, China, and worked there as a reservoir engineer until June 1997. In August 1998, she enrolled in the graduate school at the University of Texas at Austin to pursue a PhD. in the Department of Petroleum of Geosystems Engineering.

Permanent address: 2501 Lake Austin Blvd. Apt J105  
Austin, Texas 78703

This dissertation was typeset with L<sup>A</sup>T<sub>E</sub>X<sup>†</sup> by the author.

---

<sup>†</sup>L<sup>A</sup>T<sub>E</sub>X is a document preparation system developed by Leslie Lamport as a special version of Donald Knuth's T<sub>E</sub>X Program.