The Pennsylvania State University

The Graduate School

Department or Energy and Mineral Engineering

APPLICATION OF A LOCAL GRID REFINEMENT PROTOCOL TO ANALYZE THE PERFORMANCE OF RESERVOIR SYSTEMS WITH COMPLEX WELL ARCHITECTURES

A Thesis in

Energy and Mineral Engineering

by

Li-Wei Hu

© 2011 Li-Wei Hu

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Master of Science

December 2011

The thesis of Li-Wei Hu was reviewed and approved* by the following:

Turgay Ertekin
Professor of Petroleum and Natural Gas Engineering
Thesis Advisor

Luis F. Ayala H.
Associate Professor of Petroleum and Natural Gas Engineering

John Yilin Wang
Assistant Professor of Petroleum and Natural Gas Engineering

R. Larry Grayson
Professor of Energy and Mineral Engineering
Graduate Program Officer of Energy and Mineral Engineering

*Signatures are on file in the Graduate School

ABSTRACT

Complex architecture multi-lateral wells are relatively new applications in oil and gas field developments because they increase the productivity, reduce the cost of the development, and decrease the footprint and ultimately leading to increased profitability of unconventional oil/gas field projects. In addition to conventional multi-lateral wells, multi-lateral wells with a complex structure can be constructed with not only horizontal branches but also slanted and/or undulating branches, allowing a single well to maximize the productivity by reaching the reservoir layers that conventional multi-lateral wells are unable to reach.

In this thesis, a three dimensional, single-phase compressible fluid simulator is presented to model performance of a vertical well, a horizontal well, and multi-lateral wells with complex structures with the flow regimes of these hydrocarbon reservoirs. As an alternative approach to Peaceman's well equation, wells in the reservoir are approximated by a collection of fine rectangular cells in the locally refined areas, and the residual equation is utilized to calculate the flow from reservoir blocks into well cells. The approach is significantly different from conventional applications using Peaceman's well equations which are limited by the structure or the path of wells. Combining our reservoir approximation using fine rectangular cells with the local grid refinement technique, we obtain precise production forecasts and capture accurate pressure distributions around the wells.

To validate this approach, the developed simulator is used for modeling the productivity of a vertical well and a horizontal well in a single-phase compressible fluid reservoir. These results are compared with predictions from a commercial simulator[a]. A slanted well simulation is also performed and compared with predictions from the commercial simulator. The validation tests show that the approach presented is capable of providing accurate production predictions without the well structure limitation and with shorter simulation times as demanded by the application of a full fine grid refinement method.

[a] The commercial simulator is developed by Computer Measurement Group, Inc.

TABLE OF CONTENTS

LIST OF FIGURES vii

LIST OF TABLES xi

NOMENCLATURE xiii

ACKNOWLEDGEMENTS xvi

CHAPTER 1 INTRODUCTION AND MOTIVATION 1

1.1 Background 1
1.2 Motivation for Simulating Complex Well Structures 3
1.3 Literature Review 4
1.3.1 Numerical Simulation Model 5
1.3.2 Local Grid Refinement Technique 7
1.4 Statement of the Problem 10

CHAPTER 2 NUMERICAL MODEL DEVELOPMENT 12

2.1 Mathmatical Formulation 12
2.1.1 Compressible Flow Equation 14
2.2 Local Grid Refinement (LGR) Technique 16
2.2.1 Static LGR 17
2.2.2 Development of the LGR Scheme 19
2.2.3 Application of the LGR Method 26
2.3 Numerical Model 27
2.3.1 Finite-Difference Approximation for the Single-Phase Flow Continuity Equation 27
2.3.2 Residual Equation for Production Rate 31
2.3.3 Sparse Matrix Schemes 35
2.3.3.1 The Coordinate Schemes 36
2.3.3.1 The Diagonally Structured Schemes 38
2.3.4 The Preconditioned Bi-Conjugate Gradient Stabilized Method 40
2.3.5 Communication Between Coarse Cells and Window Cells 41
2.3.6 Computer Model and Flow Chart 42

CHAPTER 3 SINGLE-PHASE COMPRESSIBLE FLUID MODEL VERIFICATION 45

3.1 Simulation of Productivity of a Single Vertical Well 46
3.2 Simulation of Productivity of a Single Horizontal Well 50
3.3 Simulation of Productivity of a Single Slanted Well 52
3.4 Simulation of Production Rate Specificified Problems 64

CHAPTER 4 PRODUCTION PERFORMANCE OF WELLS WITH COMPLEX ARCHITECTURES 67

4.1 Simulation of Productivity of Complex Well Architectures 67

CHAPTER 5 SUMMARY, CONCLUSION AND RECOMMENDATIONS 73

REFERENCE 76

Appendix A Solver: The Preconditioned Bi-Conjugate Gradient Stabilized Method 79

Appendix B Equations and Correlations Used in the Model 80

Appendix C An Example Run and Output Reports 83

LIST OF FIGURES

Figure 1.1: Traditional horizontal well profiles 3

Figure 2.1: A typical fish-bone (pinnate) structure. 4

Figure 2.1: Refinement without LGR (left) and refinement with LGR (right) 16

Figure 2.2: Grid System and numbering scheme used by Wasserman. 18

Figure 2.3: Allowable vertical grid refinement by Wasserman. 18

Figure 2.4: Areal view of a horizontal well placed in the coarse block system. 20

Figure 2.5: Additional grid lines are introduced to fill up the space between well blocks and the boundaries of the window cells 21

Figure 2.6: A smooth transition provided by gradually increased block sizes. 21

Figure 2.7: Pseudo blocks introduced to the 2-D system in Figure 2.6 22

Figure 2.8: A slanted well crosses through several coarse blocks. 23

Figure 2.9: Conventional LGR technique presented by Goktas. 24

Figure 2.10: Modified LGR scheme for reducing the amount of window blocks. 24

Figure 2.11: Performing LGR to the fish-bone structure 26

Figure 2.12: Determination of fluid flow into well blocks. 32

Figure 2.13: The typical coefficient and the solution vector for the sandface pressure specified system shown in Figure 2.12 33

Figure 2.14: The typical coefficient and the solution vector for the flow rate specified system shown in Figure 2.12. 34

Figure 2.15: An example coefficient matrix constructed for an eight-block system 36

Figure 2.16: An example coordinate scheme for the system described in Figure 2.16. 37

Figure 2.17: Performing matrix multiplication to a coordinate sparse matrix 37

Figure 2.18: An example of the diagonally structured scheme for the system described in Figure 2.16 39

Figure 2.19: Performing matrix multiplication to a diagonally structured sparse matrix 39

Figure 2.20: Flow chart for the computer code developed in this study 44

Figure 3.1: Placement of the 30ft vertical well in the reservoir described in Table 3.1 46

Figure 3.2: Comparison of flow rate of 30ft vertical well 48

Figure 3.3: Rescaled comparison of flow rate of 30ft vertical well 49

Figure 3.4: Comparison of cumulative gas of 30ft vertical well 49

Figure 3.5: Placement of the 100ft horizontal well in the reservoir described in Table 3.3 50

Figure 3.6: Comparison of flow rate of 100ft horizontal well 51

Figure 3.7: Comparison of cumulative gas of 100ft horizontal well 52

Figure 3.8: Placement of the 200.25ft slanted well in the reservoir described in Table 3.4 53

Figure 3.9: Comparison of flow rate of 200.25ft slanted well 54

Figure 3.10: Comparison of cumulative gas of 200.25ft slanted well 55

Figure 3.11: Comparison of the performance of the horizontal well and the slanted well between commercial simulator and residual method 55

Figure 3.12: Placements of wells described in Table 3.5 in the middle of the reservoir 56

Figure 3.13: Comparison of flow rates of wells with different effective lengths and angles - Commercial simulator. 57

Figure 3.14: Comparison of cumulative gas of wells with different effective lengths and angles - Commercial simulator. 57

Figure 3.15: Comparison of flow rate of wells with different effective lengths and angles – Residual Method. 58

Figure 3.16: Comparison of cumulative gas of wells with different effective lengths and angles – Residual Method. 59

Figure 3.17: Rescaled comparison of flow rate of wells shown in Figure 3.15. 59

Figure 3.18: Placements of wells described in Table 3.6 in the middle of the reservoir. 60

Figure 3.19: Comparison of flow rate of wells with different effective lengths and angles –Commercial simulator 61

Figure 3.20: Comparison of cumulative gas of wells with different effective lengths and angles –Commercial simulator 61

Figure 3.21: Comparison of flow rate of wells with different effective lengths and angles – Residual Method 62

Figure 3.22: Comparison of cumulative gas of wells with different effective lengths and angles – Residual Method 62

Figure 3.23: Comparison of production rates after flow rate specification is switched to sandface pressure specification 65

Figure 3.24: Comparison of sandface pressure when flow rate is specified 65

Figure 4.1: Well structures (1), (2) in the reservoir described in Table 4.1 68

Figure 4.2: Well structures (3), (4) in the reservoir described in Table 4.1 68

Figure 4.3: Comparison of well performance in a homogeneous reservoir, $k_x=k_y=k_z=3md$ 70

Figure 4.4: Time needed to produce 50% of OGIP 70

Figure 4.5: Comparison of production performances between wells in a low permeability anisotropic reservoir 71

Figure 4.6: Time needed to produce 30% of OGIP 72

Figure C.1: Placement of the horizontal well in the example reservoir. 84

Figure C.2: Production rate of the example run. 84

Figure C.3: Cumulative gas of the example run 85

Figure C.4: Material balance checks of the example run 85

LIST OF TABLES

Table 3.1 Reservoir and fluid properties used for model verifications. 46

Table 3.2 Refinement information used in verification runs in Figure 3.2 47

Table 3.3 Refinement information used in verification runs in Figure 3.6 50

Table 3.4 Refinement information used in verification runs in Figure 3.9. 53

Table 3.5 Lengths and angels of wells in verification runs in Figure 3.13. 56

Table 3.6 Lengths and angels of wells in verification runs in Figure 3.19. 60

Table 3.7 Reservoir and fluid properties used in the flow rate specified problem. 64

Table 4.1 Reservoir and fluid properties used in the comparison runs between different well structures 67

Table 4.2 Reservoir and fluid properties used in the comparison runs between different well structures 69

Table C.1 Reservoir and fluid properties used in the simulation example 67

Table C.2: Pressure profile of the coarse block system, Time = 0.001 Day 86

Table C.3: Pressure profile of the coarse block system, Time = 0.127 Day 86

Table C.4: Pressure profile of the coarse block system, Time = 1.023 Day 86

Table C.5: Pressure profile of the coarse block system, Time = 10.023 Day 87

Table C.6: Pressure profile of the coarse block system, Time = 50.023 Day 87

Table C.7: Pressure profile of the coarse block system, Time = 100.023 Day 87

Table C.8: Pressure profile of the coarse block system, Time = 200.023 Day 88

Table C.9: Pressure profile of the window system at coarse block (5,6,2), Time = 0.001 Day 88

Table C.10: Pressure profile of the window system at coarse block (6,6,2) , Time = 0.001 Day 88

Table C.11: Pressure profile of the window system at coarse block (7,6,2) , Time = 0.001 Day 89

Table C.12: Pressure profile of the window system at coarse block (5,6,2), Time = 0.127 Day 89

Table C.13: Pressure profile of the window system at coarse block (6,6,2) , Time = 0.127 Day.....89

Table C.14: Pressure profile of the window system at coarse block (7,6,2) , Time = 0.127 Day.....90

Table C.15: Pressure profile of the window system at coarse block (5,6,2), Time = 1.023 Day.....90

Table C.16: Pressure profile of the window system at coarse block (6,6,2) , Time = 1.023 Day.....90

Table C.17: Pressure profile of the window system at coarse block (7,6,2) , Time = 1.023 Day.....91

Table C.18: Pressure profile of the window system at coarse block (5,6,2), Time = 10.023 Day.....91

Table C.19: Pressure profile of the window system at coarse block (6,6,2) , Time = 10.023 Day.....91

Table C.20: Pressure profile of the window system at coarse block (7,6,2) , Time = 10.023 Day.....92

Table C.21: Pressure profile of the window system at coarse block (5,6,2), Time = 50.023 Day.....92

Table C.22: Pressure profile of the window system at coarse block (6,6,2) , Time = 50.023 Day.....92

Table C.23: Pressure profile of the window system at coarse block (7,6,2) , Time = 50.023 Day.....93

Table C.24: Pressure profile of the window system at coarse block (5,6,2), Time = 100.023 Day.....93

Table C.25: Pressure profile of the window system at coarse block (6,6,2) , Time = 100.023 Day.....93

Table C.26: Pressure profile of the window system at coarse block (7,6,2) , Time = 100.023 Day.....94

Table C.27: Pressure profile of the window system at coarse block (5,6,2), Time = 200.023 Day.....94

Table C.28: Pressure profile of the window system at coarse block (6,6,2) , Time = 200.023 Day.....94

Table C.29: Pressure profile of the window system at coarse block (7,6,2) , Time = 200.023 Day.....95

NOMENCLATURE

Roman

A_x, A_y, A_z = Cross section area, ft^2

A = SIP coefficient in negative z direction

B = SIP coefficient in positive z direction

B_g = Gas formation volume factor, $\frac{RB}{STB}$

c = Compressibility, psi^{-1}

E = SIP coefficient in the positive x direction

G = Depth, ft

g_c = Gravitational constant in the equator, $32.2\ \frac{ft}{sec^2}$

g = Gravitational constant

k = Permeability, $perms$

m = Mass

M_w = Molecular Weight, $\frac{g}{mole}$

N = SIP coefficient in the positive y direction

$\bar{p}$ = Average system Pressure, $psia$

p = Pressure, $psia$

p_i = Initial Reservoir Pressure, $psia$

P_{sf} = Sandface pressure, $psia$

q = Flow rate, $\frac{SCF}{Day}$

Q = Total flow rate

$\tilde{q}$ = External source/sink, $\frac{SCF}{Day}$

r_w = Wellbore radius

S = SIP coefficient in negative y direction

t = Time

T = Temperature

W = SIP coefficient in negative x direction

x = Spatial coordinate in x direction

y = Spatial coordinate in y direction

V_b = Bulk volume, ft^3

z = Spatial coordinate in z direction

Greek

Δ = Space between cells, ft

Δt = Time step size, $days$

μ = Viscosity, cp

φ = Porosity, %

ρ = Density, $\frac{lb}{ft^3}$

Γ = SIP coefficient

Subscripts

avg = Average

cr = Critical

g = Gas

i, j, k = Direction indices

res = Reservoir

sc = Srandard Condition

Superscripts

i= Initial Condition

n = Old time level

$n+1$ = New time level

k = Old iteration level

$k+1$ = New iteration level

ACKNOWLEDGEMENTS

First and foremost I would like to express my gratitude to my adviser, Dr. Turgay Ertekin. This thesis would not have been possible without his guidance, suggestions, patience, and generosity throughout my study.

Beside my advisor, I would like to thank the rest of my thesis committee: Dr. Luis F. Ayala H, and Dr. John Yilin Wang for their suggestions and comments on my thesis. I would like to especially thank Dr. Ayala for his valuable advice and encouragement during the last two years.

Also, I gratefully acknowledge helpful discussions with Junjie Yang and Erhan Aslan on several points in the thesis.

Last but most importantly, I wish to express my deepest appreciations to my parents, Dian-Chung Hu and Su-Gee Liu for their devotion, endless support and love.

CHAPTER 1

INTRODUCTION AND MOTIVATION

1.1 Background

Multilateral-well technology is a more recently developed application with the ability to create wells with multiple branches that can target widely spaced reservoir compartments. It provides engineers with a new solution to optimize economic extraction of oil and gas from geologically complex reservoirs by improving production and injection rates. In recent years, the success of multi-lateral horizontal wells has already been proven in producing gas zone, thin heavy oil columns, coalbed methane reservoirs, water-flooding reservoirs and miscible flooding in carbonate reservoirs in North America and Canada[1, 2]. There also are several studies addressing pressure transient behavior in multi-lateral wells. Examples include researches on multiple horizontal wells [3] and field applications of multi-lateral wells in multi-layer formation [4, 5].

Along with the opportunity of optimizing productivity comes the inherent complexity of these well architectures, which mean the higher cost of drilling and completion. Because of this higher field development cost, the well performance should be carefully studied before making decision to ensure that the project turns out profitable.

The methodologies used to analyze well performance are commonly divided in two approaches: numerical and analytical modeling. A numerical method usually requires extensive input data to describe the reservoir and well system, and can be time-

consuming depending on the required accuracy of the solution. Refined grids are usually needed to capture the well trajectory and well location, particularly for complex well structures. In most numerical simulation models, the wellbore pressure drop is neglected. However, for complex well applications, the wellbore pressure drop often becomes a critical factor for well performance. Therefore, these models may provide misleading information when the wellbore pressure drop is not considered.

Analytical well performance models have been developed for vertical wells, horizontal wells, and slanted wells. With some assumptions, these models present explicit relationships between a flow rate and a wellbore flowing pressure. The analytical models are much easier to apply as compared to numerical models. Wellbore pressure drop can be included by coupling inflow models with wellbore flow models. However, the actual reservoir and well structures in reality are usually much more complex than the assumptions made in the closed form models. These closed form models can also, at times, have significant errors due to misrepresentations. For example, most horizontal well models assume that the horizontal wellbore is perfectly horizontal. In reality, horizontal wells are not perfectly horizontal as a result of a lack of sufficient steering control or varying formation structures. Using horizontal well models to evaluate the performance of wells with severe undulations may result in significant errors.[6]

For these reasons, development of a simulation method which is capable of having the flexibility of numerical methods and the accuracy of the analytical methods is extremely desirable for accurate simulation of complex well geometries, like fish-bone well architectures.

1.2 Motivation for Simulating Complex Well Structures

Traditional multi-lateral wells structures are usually constructed with several horizontal wells branches, to allow the single wellhead to extract oil and gas from different geological layers simultaneously or sequentially. Several traditional multi-lateral wells structures are shown in Figure 1.1. Unfortunately, many oil and gas accumulations do not exist in perfectly horizontally oriented structures. Some regions of the reservoir can be difficult to reach by horizontal well branches in traditional multi-lateral well structures.

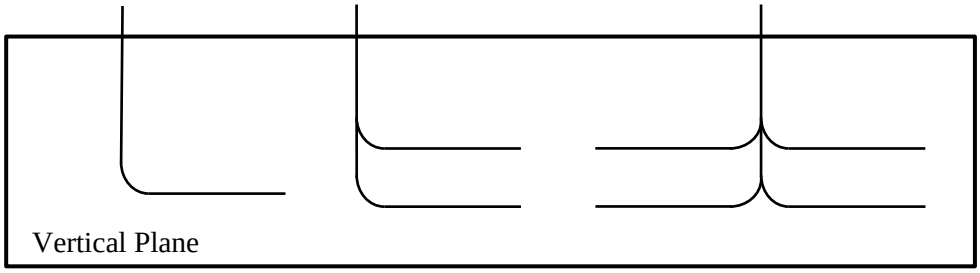


Figure 1.1**: Traditional horizontal well profile.**

Multi-lateral wells with a complex structure, like the fish-bone structure shown in Figure 1.2, can be constructed with several slanted branches. In some case, the branches of the complex well structures can be undulating if needed for special geological condition. A complex well structure also provides a larger reservoir-well contact area than the traditional horizontal well structure, which increases the productivity [2]. However, the conventional well equations cannot accurately calculate the flow rate in well structures like this. Therefore, finding a method which can accurately predict the

production rate of complex well structures is of paramount importance in the field of reservoir engineering.

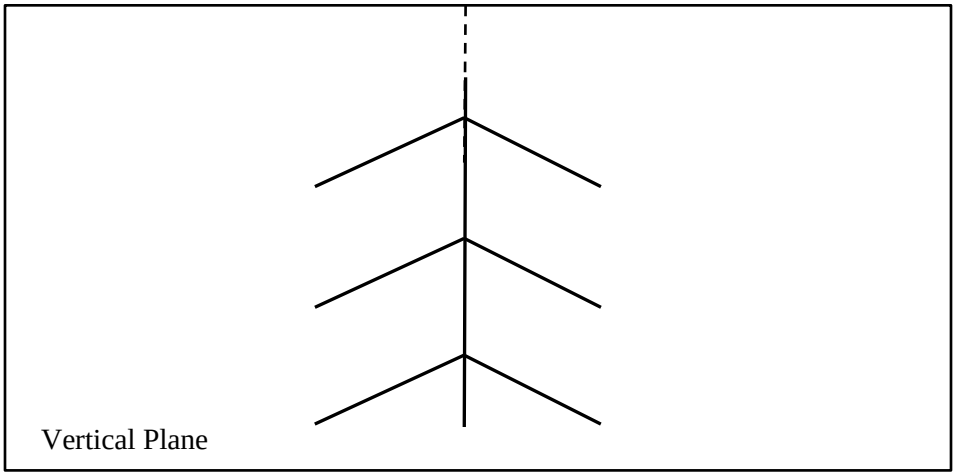


Figure 1.2**: A typical fish-bone (pinnate) structure.**

1.3 Literature Review

Two major concepts directly related to this study will be reviewed in this section: (1) Numerical simulation models and (2) Gridding techniques performed for the numerical simulation. Existing literature about reservoir models, well performance models, and solvers will be reviewed in the first section. Recent research on gridding methods and local grid refinement techniques will be reviewed afterwards.

1.3.1 Numerical Simulation Model

Reservoir studies are commonly performed by numerical models to predict the future performance of a reservoir and to find out the optimized well conditions to maximize the ultimate recovery.

During the early 50's, computer programs were first developed for performing reservoir simulation calculations. These programs used the solution of a set of finite difference equations to describe multiphase flow in 2D and 3D anisotropic porous media. In the 60's, reservoir simulation was aimed at solving for three phase flow in oil reservoirs, termed the black oil model. The trend changed abruptly in the 70's due to the intensive research on enhanced oil recovery processes. Many models related to complex reservoir processes were developed in this period for various processes: compositional model [11], thermal simulation and in-situ combustion [12], chemical and miscible flooding [13,14], and CO_2 or N_2 injection [15] . Research during this period resulted in significant progress in the physical characterization of hydrocarbon movement under the influence of temperature, chemical and multicomponent phase behavior.

In the 80's, technology was developed for modeling naturally fractured reservoirs, including compositional effects. Hydraulic fracturing and horizontal wells models were also developed during this period. During the 90's, Bulter [16] and Furui [17] developed different kinds of models for predicting the performance of horizontal wells in box-shape reservoirs, including the effects of undulating wells.

Goktas and Ertekin[7] showed that the pressure responses of wells with severe

undulation deviate from the responses of perfectly horizontal wells. They also noted that the pressure drop along the wellbore in high production wells should be considered in order to improve the accuracy of the evaluation of well performance. Additionally, the study of Al-Mohannadi[18] confirmed that when the well is extremely undulating, the effective well length is significantly longer than that of the straight wellbore length. This results in the pressure responses of undulating wells to differ from the pressure responses of perfectly horizontal wells.

The computer speed and memory size have increased exponentially after the 80's, which allowed reservoir models to become more sophisticated. The improvement of the matrix solvers was also critical to the progress of numerical reservoir simulation.

The method called preconditioned conjugate gradient method was presented by Watts [19] in 1981 as a replacement of SIP. He claimed that the method was generally faster than SIP method, which was a widely used iterative solver. In 1985, Wallis introduced a new iterative method known as the generalized conjugate residual method. Following shortly after in 1986, Saad and Schultz [21] developed a method generalized minimal residual method (GMRES), which became a part of the Bi-conjugated gradient stabilized method developed by Vorst [22] in 1992. The Bi-conjugated gradient stabilized method (BICGStab) is currently the fastest and the most reliable solver in the group of conjugate gradient methods. BICGStab is utilized with the diagonal preconditioner in this study — providing an outstanding convergence speed comparing to SIP method and GMRES.

As the reservoir and well models become more sophisticated, the speed of computing becomes more important. Since sparse matrix scheme and the speed of solvers

are reaching their limits, using parallel computing and cloud computing techniques might become necessary in the future. However, this is beyond the scope of this study.

1.3.2 Local Grid Refinement Techniques

In most numerical models, time domain and reservoir space are discretized by finite difference or finite element approximations. According to Mattax and Daltion [23], the discretization scheme must able to:

(1) Identify the pressure and saturation at a specific location in the reservoir.

(2) Describe the geometry, geology and physical properties of the reservoir.

(3) Be capable of capturing dynamic pressure and saturation profiles to meet the objective of the studies.

(4) Model the fluid mechanics properly.

The final portion of this literature review will focus on the improvement of local grid refinement (LGR) techniques which improves the accuracy of objectives described in (2) and (3) above. LGR is also a critical approach directly related to this study.

When performing a high resolution 3D reservoir simulation for a large reservoir, a larger number of fine grid-blocks need to be introduced to the system in order to capture highly localized changes in the reservoir. Generally, simulation time and computer memory usage increase with the amount of total grid-blocks. This leads to the simulation scale restricted by the performance of computer hardware. Since the purposes of most

reservoir simulation studies are predicting well performances and monitoring pressure profiles near the wellbores. A high resolution simulation for the entire reservoir domain is not necessary in most studies. Therefore, local grid refinement techniques were developed to capture critical local feature changes only in interested regions, while retaining a low refinement degree in the remaining regions in a reservoir. Gridline for refining the reservoir can be terminated within the focused area. A significant number of grid-blocks can be saved by implementing LGR, as well as the memory usage and computational load.

The LGR techniques are classified into two types: dynamic LGR and static LGR. Dynamic LGR is usually implemented for tracking dynamic local characteristics in reservoir studies. Heinemann [24] first applied dynamic LGR to study multi-phase reservoir problem in 1983. His method is able to track the properties' migration with the dynamic locally refined grid. Static LGR techniques are more commonly used in reservoir studies when the region of interest for high resolution solution does not change with time. Applications in regions around wells, faults, or fractures are good examples. In 1986, Forsyth and Sammon [25] performed a local mesh refinement technique in their study for simulating faults and pinchouts in reservoirs. An implicit algorithm was also developed to solve the matrix structure created by the LGR.

Wasserman [26] presented an application of static LGR to solve a multi-dimensional, multi-phase flow problem in 1987. An efficient solution algorithm for solving the complex data structure created by LGR was demonstrated in detail in his study. Biterge and Ertekin [10] developed an LGR methodology which is able to simulate the profile of properties in the locally refined region while keeping the complete property profile in the

coarse system. Dirichlet type boundary condition was applied to the fine grid system in their study in order to transfer the updated pressure data to the fine grid system.

In 1999, Goktas and Ertekin first implemented the LGR technique to solve a cavity production problem[10]. Locally refined grids were introduced to the system to approximate the physical structure indentations of the cavity well. The fluid flow into the specified well blocks can be calculated directly, and the production rate was then calculated by summing up the fluid flow into all of the well blocks at which pressure is specified. Goktas and Ertekin's method provided a solution for simulating performance of wells without implementing Peaceman's wellbore model. The very same approach was utilized in this study with some modifications for simulating the performances of wells with complex physical structures.

1.4 Statement of the Problem

There are two critical difficulties that need to be overcome to formulate a simulation protocol for a multi-layer well with complex structure.

First, Peaceman's wellbore model cannot be used to calculate the productivity of the wells because it lacks the ability to describe the fluid performance in the well with a multi-branch structure. To solve this problem, the residual equation is used to directly calculate the flow from reservoir cells to well cells. This generalized approach will not be restricted by the angles, shapes, or structures of wells.

The second difficulty is caused by implementing the residual approach. The well grids in the reservoir have to be refined even smaller than the well diameter in order to properly capture the well effectively. The grids surround the well grids should also be small enough to capture the change of pressure and other local characteristics while providing a smooth transition between well blocks and the boundaries of a window system during the simulation. However, the required uniform fine grids are impossible to be implemented without causing an exponential increase in the computational work. The large number of grids generated to describe a well with a complex structure using the conventional refinement approach can deplete the even memory of a modern computer.

Therefore, a mono-level static LGR technique is incorporated to significantly reduce the number of refined grids while retaining the ability to seize local characteristics sufficiently. In this study, the LGR is developed and incorporated into a three dimensional, single-phase compressible fluid model to study performance of multi-lateral

complex well structure. Flow equations are treated implicitly and solved simultaneously in both fine and coarse grid cells. An iterative solver, the preconditioned bi-conjugate gradient stabilized method, is used to solve for the pressure distribution in the reservoir along with the simple iteration method. A sparse matrix algorithm is also implemented to significantly reduce the computational load and the usage of the computer memory.

The developed model provides a solution for predicting productivities of complex well structures. It not only gives a precise representation to the physical structure of the wells, but also provides a high-resolution the pressure distribution around the well area with the help of the LGR technique.

CHAPTER 2

NUMERICAL MODEL DEVELOPMENT

2.1 Mathematical Formulation

Continuity equation is used to describe the physical phenomena of fluid flow in porous media. A differential approach was used to derive the continuity equation in this study, focusing on fluid flow through a fixed elemental volume by applying the conservation mass principle.

When a Cartesian coordinate system is used for the system (see Figure 2.1), mass flux entering a control volume can be written as:

$$m_x=\rho u_x \tag{2.1a}$$

$$m_y=\rho u_y \tag{2.1b}$$

$$m_z=\rho u_z \tag{2.1c}$$

Mass fluxes leaving a control volume are:

$$m_{x+\Delta x}=\rho u_{x+\Delta x} \tag{2.2a}$$

$$m_{y+\Delta y}=\rho u_{y+\Delta y} \tag{2.2b}$$

$$m_{z+\Delta z}=\rho u_{z+\Delta z} \tag{2.2c}$$

The mass balance of fluid flow through a control volume over a time period Δt can be described as:

$$\text{Mass in} - \text{Mass out} + \text{Source/Sink} = \text{Change in mass} \tag{2.3}$$

Substituting Equation (2.1) and Equation (2.2) into Equation (2.3), one obtains:

$$[(\rho u_x A_x)_x - (\rho u_x A_x)_{x+\Delta x}] + \left[\left(\rho u_y A_y\right)_y - \left(\rho u_y A_y\right)_{y+\Delta y}\right] + [(\rho u_z A_z)_z - (\rho u_z A_z)_{z+\Delta z}] + \tilde{q} = \frac{(\varphi \Delta x \Delta y \Delta z \rho)_{t+\Delta t} - (\varphi \Delta x \Delta y \Delta z \rho)_t}{\Delta t} \quad (2.4)$$

where A_x, A_y, A_z are areas perpendicular to flow direction, and $\tilde{q}$ is the source/sink term.

By defining $V_b = \Delta x \Delta y \Delta z$, Equation (2.4) can be rewritten as:

$$-[(\rho u_x)_{x+\Delta x} - (\rho u_x)_x] - \left[\left(\rho u_y\right)_{y+\Delta y} - \left(\rho u_y\right)_y\right] - [(\rho u_z)_{z+\Delta z} - (\rho u_z)_z] + \frac{\tilde{q}}{V_b} = \frac{(\varphi \rho)_{t+\Delta t} - (\varphi \rho)_t}{\Delta t} \quad (2.5)$$

The derivative of a function $f(s)$ is defined as:

$$\lim_{\Delta s \to 0} \frac{f(s+\Delta s) - f(s)}{\Delta s} = \frac{\partial f}{\partial s} \quad (2.6)$$

Applying Equation (2.6) to Equation (2.5), and take the limits of Equation (2.5) as

$\Delta x \to 0, \Delta y \to 0, \ \Delta z \to 0$

and $\Delta t \to 0$, the continuity equation becomes:

$$-\frac{\partial}{\partial x}(\rho u_x) - \frac{\partial}{\partial y}\left(\rho u_y\right) - \frac{\partial}{\partial z}(\rho u_z) + \frac{\tilde{q}}{V_b} = \frac{\partial}{\partial t}(\varphi \rho) \quad (2.7)$$

or,

$$\nabla \rho \mu - \frac{\tilde{q}}{V_b} = -\frac{\partial}{\partial t}(\varphi \rho) \quad (2.8)$$

Introducing the Darcy's law as the constitutive equation in the formulation,:

$$u = -\frac{k}{\mu}\nabla\Phi \tag{2.9}$$

in which flow potential Φ is defined as $\Phi = \int_{psc}^{p}(\frac{\partial p}{\rho(p)} + \frac{g}{gc}G)$

by substituting Equation (2.9) into Equation (2.8) yields:

$$\nabla\left\{\rho\frac{k}{\mu}(\nabla \mathrm{p} - \rho\frac{g}{gc}\nabla G)\right\} + \frac{\tilde{q}}{V_b} = \frac{\partial}{\partial t}(\varphi\rho) \tag{2.10}$$

Equation (2.10) is known as the basic flow equation in porous medium, there are several assumptions associated with this equation[7]:

(1) No chemical reaction between the fluid and the porous medium.

(2) Flow is single phase, laminar, viscous, and isothermal.

(3) Electrokinetics and slippage effect are negligible.

2.1.1 Compressible Flow Equation

In this study, the compressible flow equation was applied to validate the residual approach incorporated into the formulation:

Density of compressible fluids can be expressed by the equation of state:

$$\rho = \frac{pM_w}{zRT} \tag{2.11}$$

Formation volume factor is defined as:

$$B = \frac{\rho_{sc}}{\rho} \tag{2.12}$$

Substituting Equation (2.11) into Equation (2.12) yields:

$$B = \frac{p_{sc} T z}{T_{sc} p} \tag{2.13}$$

Assuming that porosity is constant, substituting Equation (2.11) into the left-hand side of Equation (2.10) and Equation (2.13) into the right-hand side of the Equation (2.10) gives:

$$\nabla \left\{ \frac{k}{\mu B} (\nabla \mathrm{p} - \rho \frac{g}{gc} \nabla G) \right\} + \frac{\tilde{q}}{V_b} = \varphi \frac{T_{sc}}{p_{sc}} \frac{\partial}{\partial t} \left(\frac{p}{z} \right) \tag{2.14}$$

Equation (2.14) is called the diffusivity equation of compressible fluid flow in porous media[8], which provide the formalism describing the fluid behavior in this study.

2.2 Local Grid Refinement (LGR) Technique

The development of the LGR scheme is one of the more important components of this study: it provides not only small enough grids to approximate well structures, but also the ability to capture pressure change around the region we focus on. Also, implementation of the LGR technique significantly reduces the computer memory usage and simulation time as compared to conventional refinement methods.

The LGR method is a domain decomposition where fine grids, which are also known as a window system, are created within specified coarse grid blocks in three dimensions. Figure 2.1 illustrates the LGR technique in a two-dimensional plan view.

There are basically two types of LGR technique: dynamic LGR and static LGR. Dynamic LGR enable us to track dynamic local characteristics by introducing fine grids into coarse grids as a function of time. It is commonly used in numerical simulations studies designed for front tracking problems. In well performance simulations, we focus on the characteristic profiles around the wells. Since the well location is fixed, using the static LGR technique suits our purpose effectively.

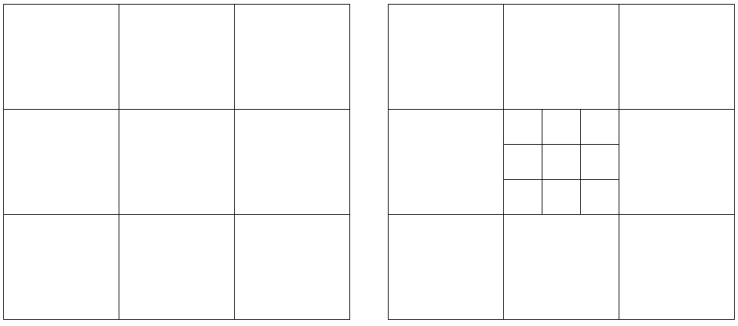


Figure 2.1**: Refinement without LGR (left) and Refinement with LGR (right).**

2.2.1 Static LGR

The grid refinement process of static LGR is done only once and remains unchanged during the whole simulation process. A typical static LGR technique is developed by Wasserman[9], the grid system numbering scheme and the restriction with his technique is shown in Figure 2.2 and Figure 2.3.

There are two rules that have to be followed with his method:

(1) Refinement is not allowed at a coarse grid boundary. All window blocks have to be surrounded by coarse blocks.

(2) Vertical grid lines cannot be terminated within an aerial window, the idea is shown in Figure 2.3.

Wasserman solved the equations writing for coarse cells and window cells of the whole system simultaneously. His method was capable of providing stable result, however, the complex data structure created by his LGR scheme significantly slowed down the simulation speed. Also, a multi-level LGR cannot be performed with Wasserman's scheme.

Biterge and Ertekin[10]developed a modified grid refinement technique which solved the coarse and the fine systems in a sequential manner. They solved the coarse system for the variable distribution in the reservoir first, and then they solve the fine system by assigning Dirichlet type boundary conditions at the fine-coarse grid boundaries. This method retains a hepta-diagonal data structure in the coarse grid system and allows multi-level grid refinements to be performed within the coarse grid. In this study, the LGR scheme developed by Biterge and Ertekin[10] is used to refine the reservoir. Some

parts of the procedures have been modified to meet the purposes of this study and the details will be described in the next section.

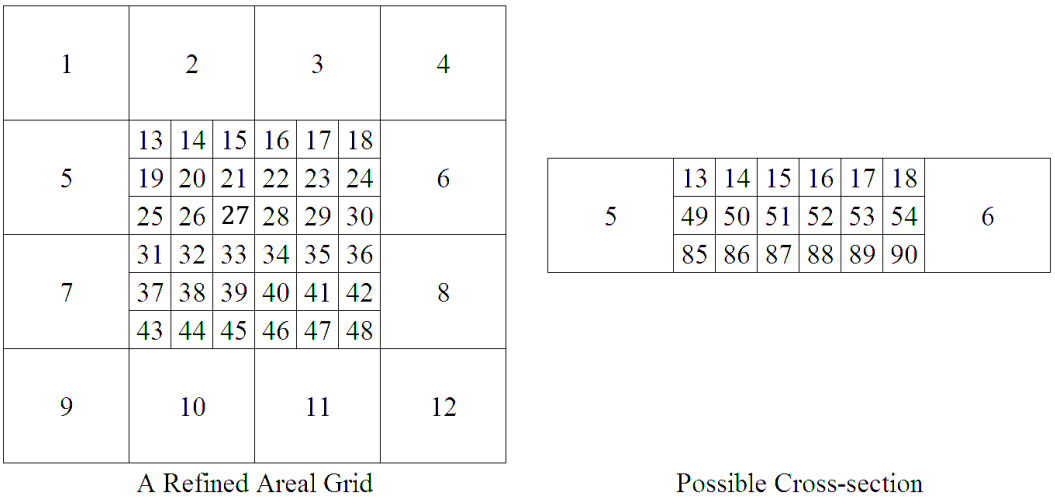


Figure 2.2**: Grid System and numbering scheme used by Wasserman[9].**

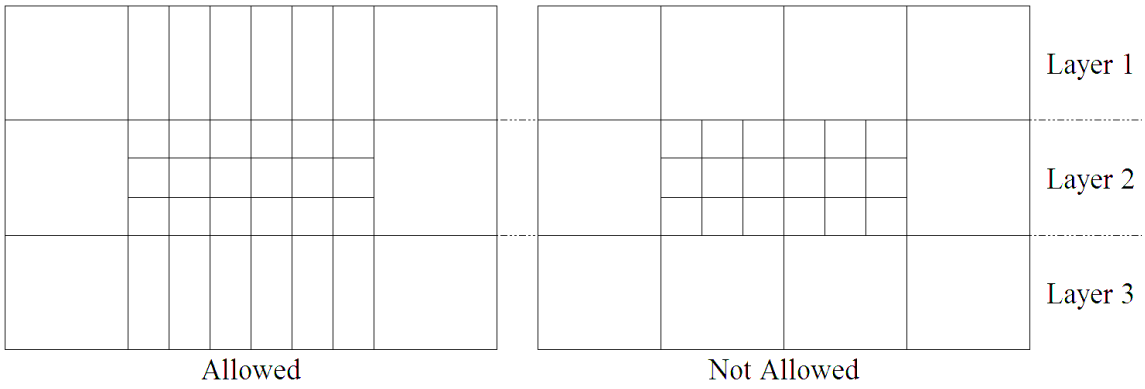


Figure 2.3**: Allowable vertical grid refinement by Wasserman[9].**

2.2.2 Development of the LGR Scheme

The development of the LGR scheme in this study is similar to the one developed by Goktas and Ertekin[7] with some modifications, and will be explained in detail later in this section. First, let us see the two dimensional example shown in Figure 2.4.

The location of the well in the coarse system has to be determined first, then the coarse blocks to be decomposed can be decided. There are two coarse blocks will be locally refined in this example. Next, additional blocks representing the well structure and volume are introduced into the coarse blocks, the remaining unrefined space in the coarse blocks is then refined, see Figure 2.5. To minimize the potential errors that will incur by dramatic changes of block sizes, a smooth transition is needed not only between the boundary of window block and their surrounding coarse blocks, but also between the neighboring blocks inside the windows. Therefore, as shown in Figure 2.7, following rules is applied to decide the block sizes between well blocks and the boundary of the windows:

$$\Delta x_{i+1,j} = \Delta x_{i,j} \times \alpha, \qquad while\ \Delta x < 2 \tag{2.15}$$

$$\Delta y_{i,j+1} = \Delta y_{i,j} \times \beta, \qquad while\ \Delta y < 2 \tag{2.16}$$

otherwise,

$$\Delta x_{i+1,j} = \left(\Delta x_{i,j}\right)^{\alpha} \tag{2.17}$$

$$\Delta y_{i,j+1} = \left(\Delta y_{i,j}\right)^{\beta} \tag{2.18}$$

the value of α and β have to be larger than 1 to smoothly expand the grid sizes and the effect of changing α and β value will be shown in Chapter 3.

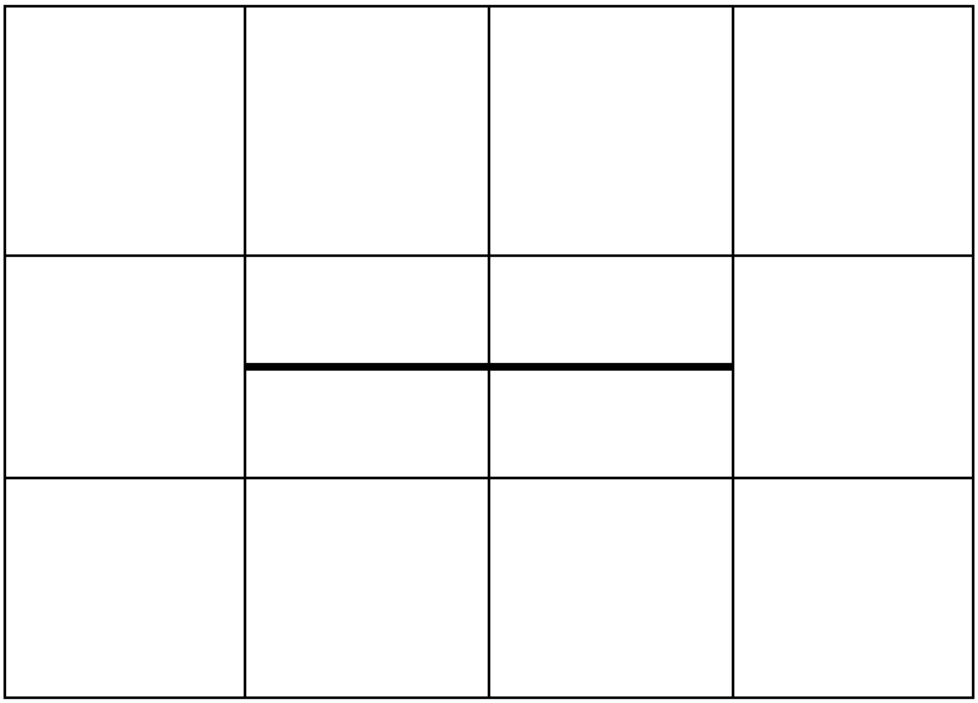


Figure 2.4**: Areal view of a horizontal well placed in the coarse block system.**

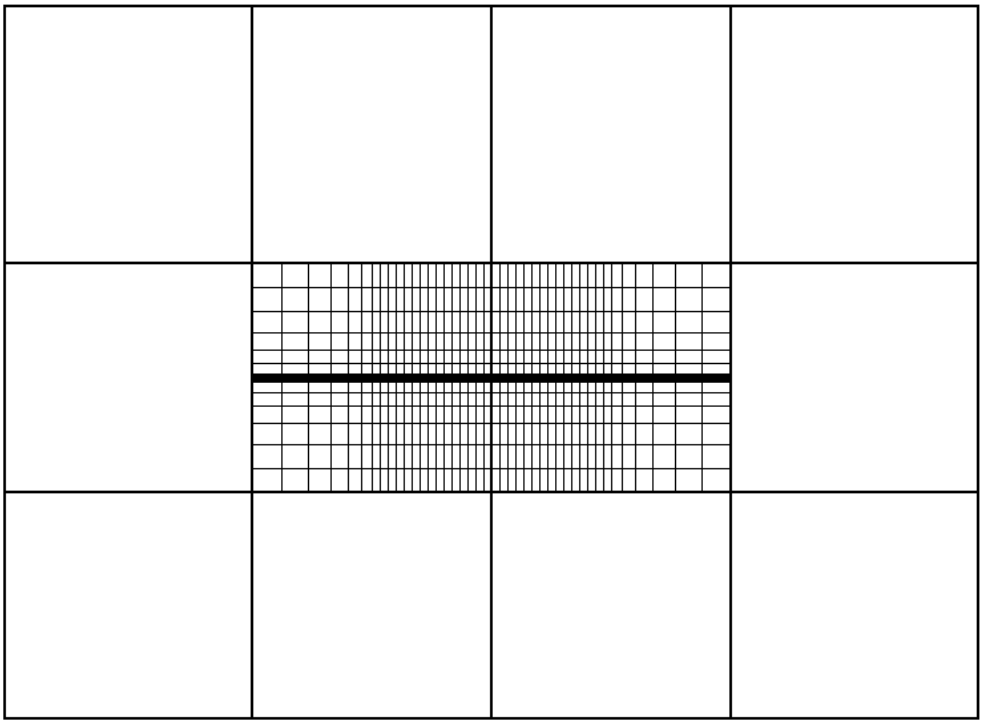


Figure 2.5**: Additional grid lines are introduced to fill up the space between well blocks and the boundaries of the window cells.**

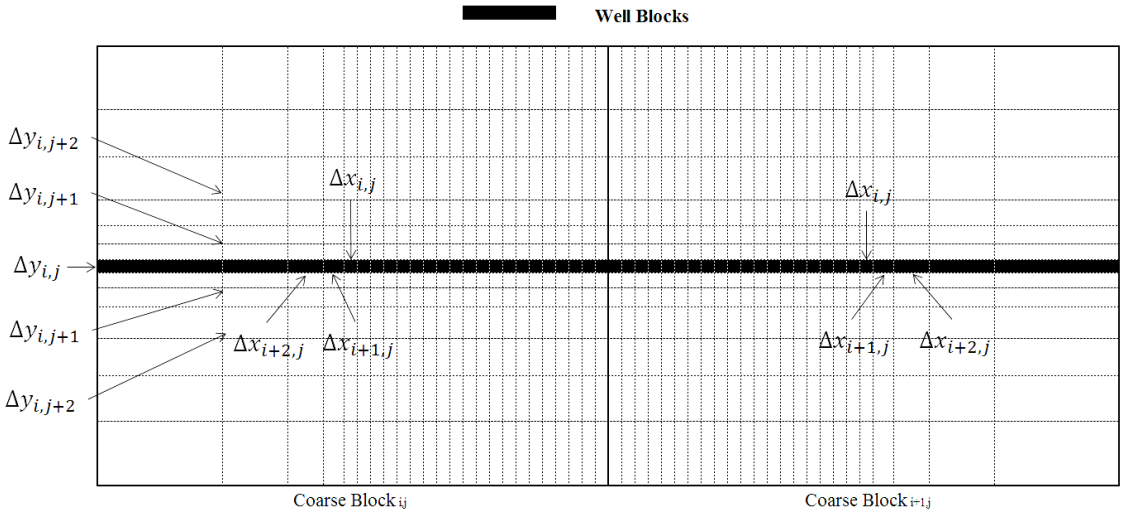


Figure 2.6**: A smooth transition provided by gradually increased block sizes.**

The pressure profiles of coarse blocks surrounding the window blocks are needed in order to calculate the pressure at the coarse block-window block interfaces, which implies that no LGR should be applied to the course blocks at the boundary of the system. Pseudo pressure profiles in the coarse blocks surrounding the windows are calculated by the linear interpolations of pressures in a three dimensional system. The example in Figure 2.8 shows the pseudo blocks which should be created surrounding the windows in the two dimensional system we have in Figure 2.6. For a two dimensional system, the pseudo pressures P_a and P_b in Figure 2.7 can be calculated by the following equations:

$$P_a = \frac{\Delta x_1 P_2 + \Delta x_2 P_1}{\Delta x_1 + \Delta x_1} \tag{2.19}$$

$$P_b = \frac{\Delta y_1 P_5 + \Delta y_2 P_1}{\Delta y_1 + \Delta y_2} \tag{2.20}$$

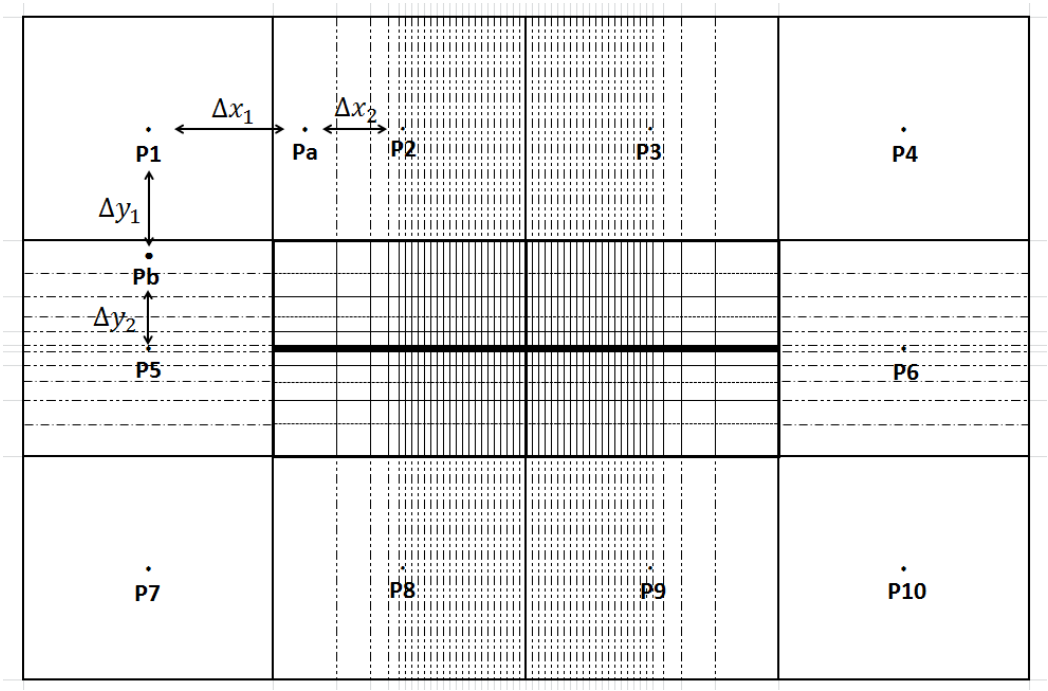


Figure 2.7**: Pseudo blocks introduced to the 2-D system in Figure 2.6.**

The pseudo pressure we are calculating above will be used for calculating the transmissibility terms at the boundaries of window blocks and coarse blocks. The linear interpolations will also have to be done in the vertical direction when a three dimensional system is considered.

A modified LGR scheme is implemented when a slanted well crosses through multiple coarse blocks diagonally, see Figure 2.8. The LGR scheme presented by Goktas is then performed as shown in Figure 2.9 [7]. Due to the restriction of Goktas' scheme, grid lines will have to extend to the coarse block which is not holding well blocks, and grid lines placed in two neighboring coarse block should coincide. This scheme will introduce a huge number of blocks into the window system, which will cause some memory and performance problems during the conduct of simulation runs.

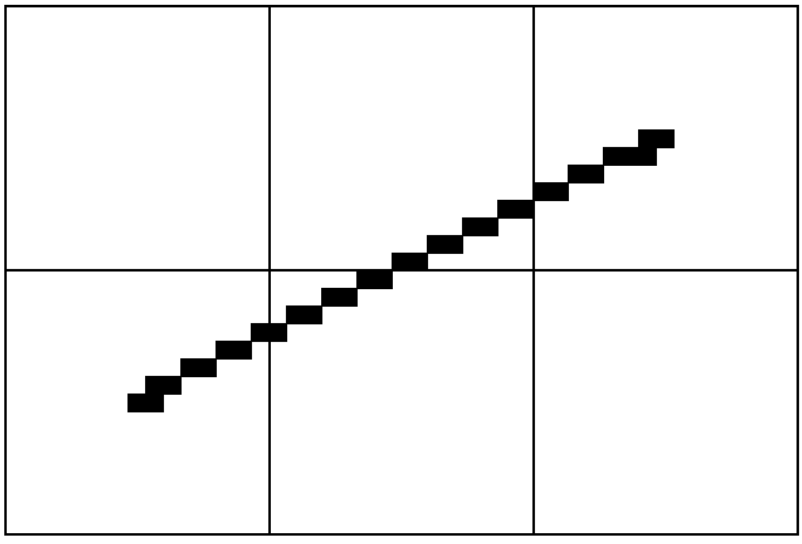


Figure 2.8**: A slanted well crosses through several coarse blocks.**

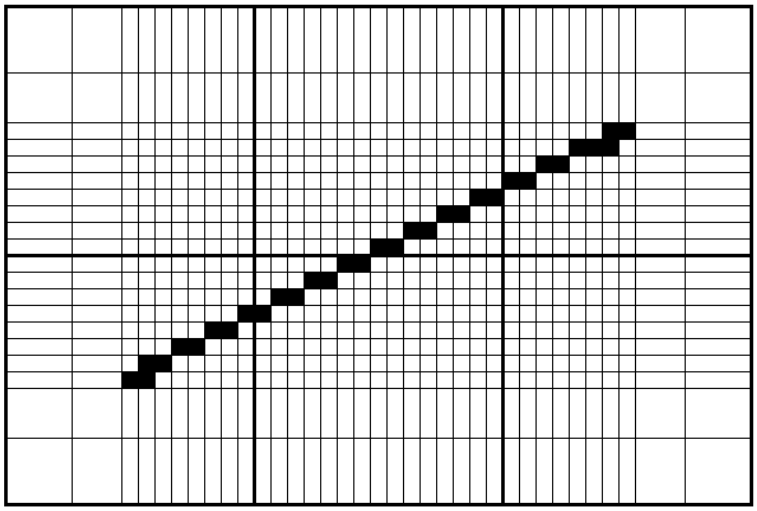


Figure 2.9**: Conventional LGR technique presented by Goktas[7].**

The modified LGR scheme shown in Figure 2.10 is performed when multiple coarse blocks are intersected by the well. All the grid lines introduced by LGR can be terminated within the coarse blocks, and no refinement is needed in the coarse blocks which are not hosting the well. At this time, all the transmissibility terms at the boundary between window blocks and coarse blocks are calculated with the pressure value interpolated using the pressures in the coarse blocks system.

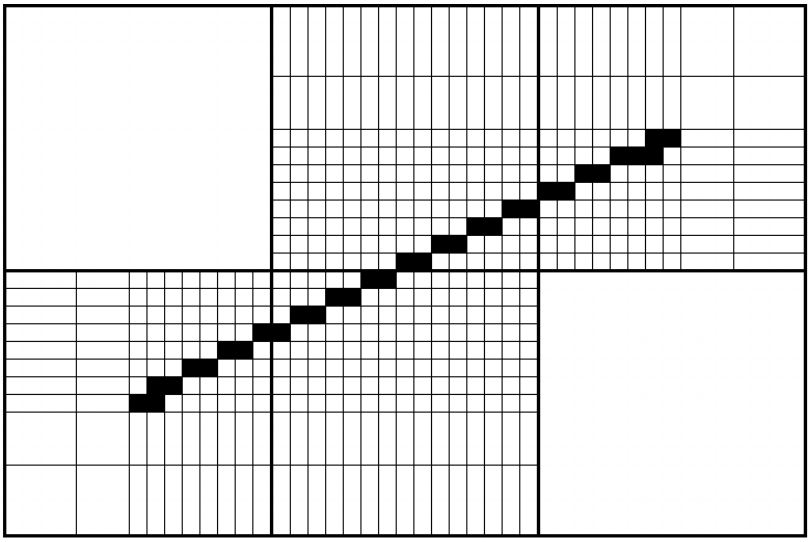


Figure 2.10**: Modified LGR scheme for reducing the amount of window blocks.**

This modified scheme significantly reduces the number of blocks in the system as compared to Goktas' approach and accordingly, it speeds up the simulation and saves more computer memory. Some errors will be introduced by this method theoretically, because transmissibility terms calculated for window blocks at the boundaries between coarse blocks and window blocks are all calculated with the pseudo-pressure value as obtained from the coarse block system. However, the center pressure of a coarse block cannot accurately represent the center pressure at the middle of a window, and pressure profiles in a window usually does not distribute linearly because of the effect of well blocks. Errors are then introduced to the transmissibility terms when interpolating pseudo pressure profiles. The amount of fluid flow into a well block at the boundary of a window will be affected by this error, which becomes slightly higher.

However, the errors are minimized when the degree of refinement in a window increases.

2.2.3 Application of the LGR Method

By introducing locally refined blocks into a coarse block system, the internal volume of wells can be matched by the volume of well blocks[7], and the well structure can be approximated by defining a collection of well blocks in the window. Thus, a well with a complex structure, like a fish-bone structure, can also be approximated introducing by enough number of LGR windows (see the example in Figure 2.11).

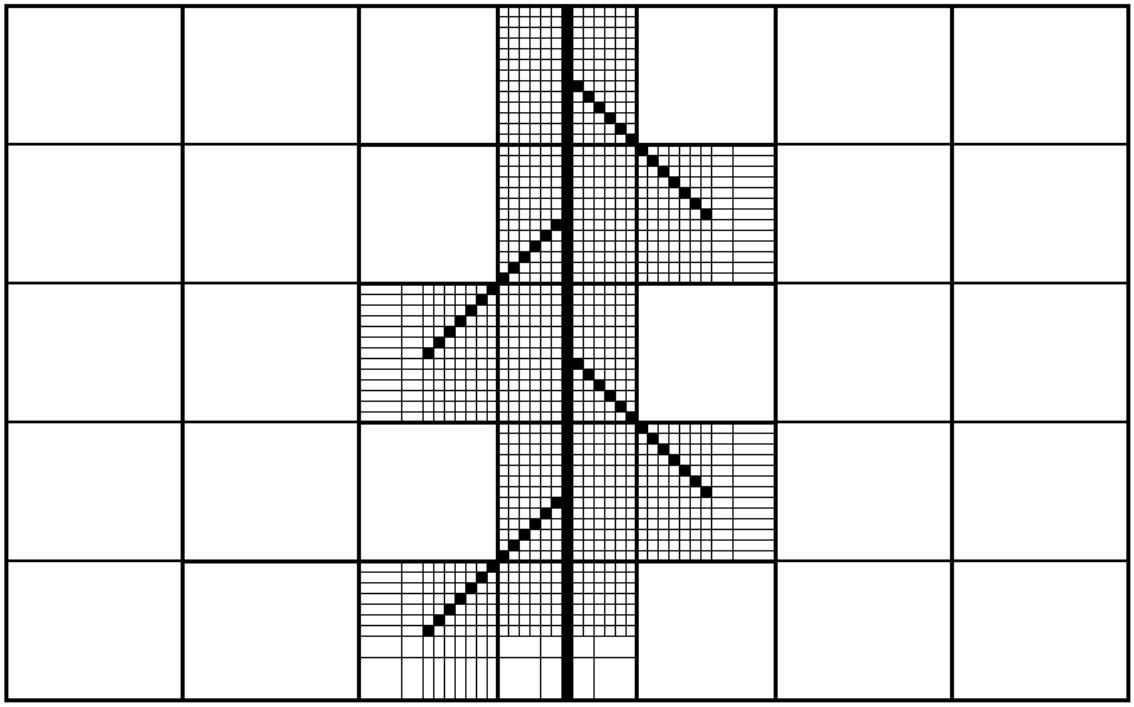


Figure 2.11**: Performing LGR to the fish-bone structure.**

2.3 Numerical Model

There are several critical treatments that have to be implemented to the computer code to solve the continuity flow equations in the cells in the coarse blocks and locally refined window blocks numerically. The following discussion will show (1) how the continuity equation is solved numerically; (2) how the production rate is calculated without using the Peaceman's model; (3) how sparse matrix schemes enhance the performance of the program; (4) which solver is used for solving equations written for the system simultaneously; (5) how windows system and coarse blocks system communicate; and (6) the basic task functions of the developed simulation program.

2.3.1 Finite-Difference Approximation for the Single-Phase Compressible Fluid Flow Continuity Equation

Equation (2.14) is a mathematical expression describing the physics of single-phase compressible fluid flow in porous media. It needs to be solved numerically in consideration of the existing nonlinear terms in the equation. The body centered grid system is utilized to describe the reservoir system, and a second order finite-difference approximation is applied to Equation (2.14) to obtain the solution at discrete points in the gridding system.

The gravity terms can be ignored in the compressible fluid flow equation, and Equation (2.14) can be rewritten as:

$$\frac{\partial}{\partial x}\left\{\frac{A_x k_x}{\mu_g B_g}\frac{\partial x}{\partial p}\right\}\Delta x + \frac{\partial}{\partial y}\left\{\frac{A_y k_y}{\mu_g B_g}\frac{\partial y}{\partial p}\right\}\Delta y + \frac{\partial}{\partial z}\left\{\frac{A_z k_z}{\mu_g B_g}\frac{\partial z}{\partial p}\right\}\Delta z + q_g$$

$$= \frac{T_{sc} V_b \varphi}{p_{sc} T}\frac{\partial}{\partial t}\left(\frac{p}{z}\right) \tag{2.21}$$

In Equation (2.21) B_g is defined as:

$$B_g = \frac{p_{sc} T Z}{5.615 T_{sc} p}$$

Writing Equation (2.15) in its finite-difference form, it becomes:

$$\left(\frac{A_x k_x}{\mu_g B_g \Delta x}\right)^{n+1}_{i+\frac{1}{2},j,k}\left(p^{n+1}_{i+1,j,k} - p^{n+1}_{i,j,k}\right) - \left(\frac{A_x k_x}{\mu_g B_g \Delta x}\right)^{n+1}_{i-\frac{1}{2},j,k}\left(p^{n+1}_{i,j,k} - p^{n+1}_{i-1,j,k}\right)$$

$$+\left(\frac{A_x k_x}{\mu_g B_g \Delta y}\right)^{n+1}_{i,j+\frac{1}{2},k}\left(p^{n+1}_{i,j,k} - p^{n+1}_{i,j+1,k}\right) - \left(\frac{A_x k_x}{\mu_g B_g \Delta y}\right)^{n+1}_{i,j-\frac{1}{2},k}\left(p^{n+1}_{i,j,k} - p^{n+1}_{i,j-1,k}\right)$$

$$+\left(\frac{A_x k_x}{\mu_g B_g \Delta z}\right)^{n+1}_{i,j,k+\frac{1}{2}}\left(p^{n+1}_{i,j,k+1} - p^{n+1}_{i,j,k}\right) - \left(\frac{A_x k_x}{\mu_g B_g \Delta z}\right)^{n+1}_{i+\frac{1}{2},j,k-\frac{1}{2}}\left(p^{n+1}_{i,j,k} - p^{n+1}_{i,j,k-1}\right)$$

$$+q_g = \frac{T_{sc} V_b \varphi}{p_{sc} T z \Delta t}\left(p^{n+1}_{i,j,k} - p^{n}_{i,j,k}\right) \tag{2.22}$$

After writing all the transmissibility terms in matrix notation, Equation (2.22) becomes:

$$E_{i,j,k}^{n+1}\left(p_{i+1,j,k}^{n+1}-p_{i,j,k}^{n+1}\right)\ -\ W_{i,j,k}^{n+1}\left(p_{i,j,k}^{n+1}-p_{i-1,j,k}^{n+1}\right)$$

$$+S_{i,j,k}^{n+1}\left(p_{i,j,k}^{n+1}-p_{i,j+1,k}^{n+1}\right)\ -\ N_{i,j,k}^{n+1}\left(p_{i,j,k}^{n+1}-p_{i,j-1,k}^{n+1}\right)$$

$$+B_{i,j,k}^{n+1}\left(p_{i,j,k+1}^{n+1}-p_{i,j,k}^{n+1}\right)\ -\ A_{i,j,k}^{n+1}\left(p_{i,j,k}^{n+1}-p_{i,j,k-1}^{n+1}\right)\ +\ q_g$$

$$=\frac{\Gamma_{i,j,k}}{\Delta t}\left(p_{i,j,k}^{n+1}-p_{i,j,k}^{n}\right) \tag{2.23}$$

where,

$$E_{i,j,k}^{n+1}=\left(\frac{A_x k_x}{\mu_g B_g \Delta x}\right)_{i+\frac{1}{2},j,k}^{n+1}$$

$$W_{i,j,k}^{n+1}=\left(\frac{A_x k_x}{\mu_g B_g \Delta x}\right)_{i-\frac{1}{2},j,k}^{n+1}$$

$$S_{i,j,k}^{n+1}=\left(\frac{A_x k_x}{\mu_g B_g \Delta y}\right)_{i,j+\frac{1}{2},k}^{n+1}$$

$$N_{i,j,k}^{n+1}=\left(\frac{A_x k_x}{\mu_g B_g \Delta y}\right)_{i,j-\frac{1}{2},k}^{n+1}$$

$$B_{i,j,k}^{n+1}=\left(\frac{A_x k_x}{\mu_g B_g \Delta z}\right)_{i,j,k+\frac{1}{2}}^{n+1}$$

$$A_{i,j,k}^{n+1}=\left(\frac{A_x k_x}{\mu_g B_g \Delta z}\right)_{i+\frac{1}{2},j,k-\frac{1}{2}}^{n+1}$$

and,

$$\Gamma_{i,j,k}=\frac{T_{sc}V_b\varphi}{p_{sc}Tz}$$

The generated flow equations are solved implicitly in the study. In addition, we have to consider the weakly non-linear nature of the transmissibility terms. Take $E_{i,j,k}^{n+1}$ as an example, the transmissibility term can be separated into two parts:

$$E_{i,j,k}^{n+1} = \left(\frac{A_x k_x}{\Delta x}\right)_{i+\frac{1}{2},j,k} \left(\frac{1}{\mu_g B_g}\right)_{i+\frac{1}{2},j,k} \tag{2.24}$$

The first part of the transmissibility terms are decided after the size of the refined blocks are decided, the value of the first part remains constant during the entire simulation process. The harmonic averaging is used to calculate this part of the transmissibility terms:

$$\left(\frac{A_x k_x}{\Delta x}\right)_{i+\frac{1}{2},j,k} = \frac{2A_{x_{i,j,k}} A_{x_{i+1,j,k}} k_{x_{i,j,k}} k_{x_{i+1,j,k}}}{A_{x_{i,j,k}} k_{x_{i,j,k}} \Delta_{x_{i+1,j,k}} + A_{x_{i+1,j,k}} k_{x_{i+1,j,k}} \Delta_{x_{i,j,k}}} \tag{2.25}$$

The second part of the transmissibility is a non-linear term which is a function of pressure and temperature, and its value needs to be updated every time when pressure or temperature value change. Linear interpolation is performed to find the value of this term at grid boundaries interfaces:

$$\left(\frac{1}{\mu_g B_g}\right)_{i+\frac{1}{2},j,k} = \frac{\Delta_{x_{i,j,k}} \left(\frac{1}{\mu_g B_g}\right)_{i+1,j,k} + \Delta_{x_{i+1,j,k}} \left(\frac{1}{\mu_g B_g}\right)_{i,j,k}}{\Delta_{x_{i,j,k}} + \Delta_{x_{i+1,j,k}}} \tag{2.26}$$

2.3.2 Residual Equation for Production Rate

Using Peaceman's equation to calculate production rate of a well is limited by the shape and structure of the well, and correlations have to be made depending on well types. In this study, the residual form of the equation is used to determine the amount of fluid flows into the well. This method can be utilized for any well structure and provides an accurate solution.

Consider the example shown in Figure 2.12 when a horizontal well is placed in two LGR windows. We assume that hypothetical blocks are located at the interface between well blocks and reservoir blocks, and their volume is small enough to ignore the accumulation term. Fluid flow from a reservoir block to a well block next to it will first be assigned to the hypothetical block between them, then flow into the well block. The fluid flow into a well block can then be rewritten as:

$$
\begin{aligned}
q_{i,j,k} = &-\left(\frac{A_x k_x}{\mu_g B_g \Delta x}\right)^{n+1}_{i+\frac{1}{2},j,k} \left(p^{n+1}_{i+1,j,k} - p^{n+1}_{i,j,k}\right) + \left(\frac{A_x k_x}{\mu_g B_g \Delta x}\right)^{n+1}_{i-\frac{1}{2},j,k} \left(p^{n+1}_{i,j,k} - p^{n+1}_{i-1,j,k}\right) \\
&-\left(\frac{A_x k_x}{\mu_g B_g \Delta y}\right)^{n+1}_{i,j+\frac{1}{2},k} \left(p^{n+1}_{i,j,k} - p^{n+1}_{i,j+1,k}\right) + \left(\frac{A_x k_x}{\mu_g B_g \Delta y}\right)^{n+1}_{i,j-\frac{1}{2},k} \left(p^{n+1}_{i,j,k} - p^{n+1}_{i,j-1,k}\right) \\
&-\left(\frac{A_x k_x}{\mu_g B_g \Delta z}\right)^{n+1}_{i,j,k+\frac{1}{2}} \left(p^{n+1}_{i,j,k+1} - p^{n+1}_{i,j,k}\right) + \left(\frac{A_x k_x}{\mu_g B_g \Delta z}\right)^{n+1}_{i+\frac{1}{2},j,k-\frac{1}{2}} \left(p^{n+1}_{i,j,k} - p^{n+1}_{i,j,k-1}\right)
\end{aligned}
\tag{2.27}
$$

or using the transmissibility terms:

$$q_{i,j,k} = -(E_{i,j,k}^{n+1}\left(p_{i+1,j,k}^{n+1}\right) + W_{i,j,k}^{n+1}\left(p_{i-1,j,k}^{n+1}\right) + S_{i,j,k}^{n+1}\left(p_{i,j+1,k}^{n+1}\right) + N_{i,j,k}^{n+1}\left(p_{i,j-1,k}^{n+1}\right) + B_{i,j,k}^{n+1}\left(p_{i,j,k+1}^{n+1}\right) + A_{i,j,k}^{n+1}\left(p_{i,j,k-1}^{n+1}\right) + C_{i,j,k}^{n+1}\left(p_{i,j,k}^{n+1}\right)) \quad (2.28)$$

where,

$$C_{i,j,k}^{n+1}\left(p_{i,j,k}^{n+1}\right) = -(E_{i,j,k}^{n+1} + W_{i,j,k}^{n+1} + S_{i,j,k}^{n+1} + N_{i,j,k}^{n+1} + B_{i,j,k}^{n+1} + A_{i,j,k}^{n+1})$$

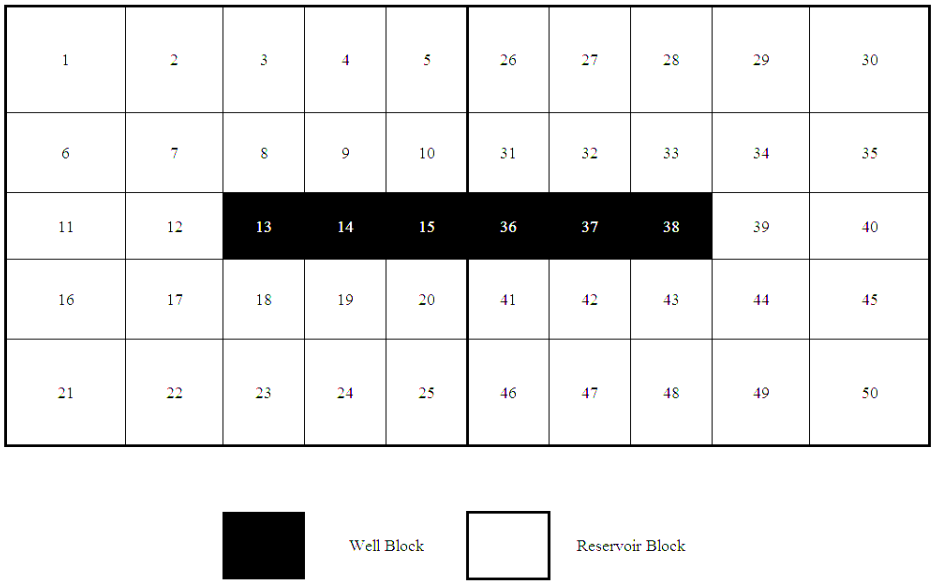


Figure 2.12**: Determination of fluid flow into well blocks.**

In this study, multiple well blocks are used to approximate the structure of a well, Equation (2.28) is calculated for all of the well blocks and have to be summed up to find of the total flow rate:

$$q_{total} = \sum_{1}^{n} q_{i,j,k} \quad (2.29)$$

The upper limit of the summation sign n represents the number of the well blocks. To update the pressure in the coarse block system, we also need to know the flow coming

out from each coarse block which is hosting the well. The flow rate for each coarse block can be calculated from:

$$q_{I,J,K} = \sum_{1}^{n_{I,J,K}} q_{i,j,k}^{I,J,K} \tag{2.30}$$

In Equation (2.30) I, J, K are the indices indicating the locations of the coarse blocks, and $n_{I,J,K}$ represents the number of the refined well blocks within the coarse block (I, J, K).

When calculating $q_{i,j,k}$ by specifying sandface pressure, all the well blocks pressure are specified as the sandface pressure. The wellbore frictional pressure drop and gravity terms are ignored. All the transmissibilities at the interfaces between well blocks and reservoir blocks are determine by using the reservoir properties only, there is no need to set a large permeability for the well blocks during the calculation. Let us consider the example in Figure 2.12, since the pressure of the well blocks is specified, the typical coefficient and the solution vector for the system becomes as shown in Figure 2.13:

	1	2	…	13	14	15	…	36	37	38	…	49	50				
1	C_1	E_1	…	0	0	0	…	0	0	0	…	0	0		P_1		Q_1
2	W_2	C_2	…	0	0	0	…	0	0	0	…	0	0		P_2		Q_2
⋮	⋮	⋮		⋮	⋮	⋮		⋮	⋮	⋮		⋮	⋮		⋮		⋮
13	0	0	…	1	0	0	…	0	0	0	…	0	0		P_{13}		P_{sf}
14	0	0	…	0	1	0	…	0	0	0	…	0	0		P_{14}		P_{sf}
15	0	0	…	0	0	1	…	0	0	0	…	0	0		P_{15}		P_{sf}
⋮	⋮	⋮		⋮	⋮	⋮		⋮	⋮	⋮		⋮	⋮	•	⋮	=	⋮
36	0	0	…	0	0	0	…	1	0	0	…	0	0		P_{36}		P_{sf}
37	0	0	…	0	0	0	…	0	1	0	…	0	0		P_{37}		P_{sf}
38	0	0	…	0	0	0	…	0	0	1	…	0	0		P_{38}		P_{sf}
⋮	⋮	⋮		⋮	⋮	⋮		⋮	⋮	⋮		⋮	⋮		⋮		⋮
49	0	0	…	0	0	0	…	0	0	0	…	C_{49}	E_{49}		P_{49}		Q_{49}
50	0	0	…	0	0	0	…	0	0	0	…	W_{50}	C_{50}		P_{50}		Q_{50}

Figure 2.13: The typical coefficient matrix and the solution vector for the sandface pressure specified system shown in Figure 2.12.

When flow rate of the well is specified, Equation (2.29) have to be solved implicitly to determine the pressure of well blocks, which is also sandface pressure. Consider the example in Figure 2.12, the q_{total} can be calculate as:

$$\begin{aligned} q_{total} &= q_{13} + q_{14} + q_{15} + q_{36} + q_{37} + q_{38} \\ &= -(E_{13}P_{14} + W_{13}P_{12} + N_{13}P_{8} + S_{13}P_{18} + C_{13}P_{13}) \\ &-(E_{14}P_{15} + W_{14}P_{13} + N_{14}P_{9} + S_{14}P_{19} + C_{14}P_{14}) \\ &-(E_{15}P_{36} + W_{15}P_{14} + N_{15}P_{10} + S_{15}P_{20} + C_{15}P_{15}) \\ &-(E_{36}P_{37} + W_{36}P_{15} + N_{36}P_{31} + S_{36}P_{41} + C_{36}P_{36}) \\ &-(E_{37}P_{38} + W_{37}P_{36} + N_{37}P_{32} + S_{37}P_{42} + C_{37}P_{37}) \\ &-(E_{38}P_{39} + W_{38}P_{37} + N_{38}P_{33} + S_{38}P_{43} + C_{38}P_{38}) \end{aligned} \tag{2.31}$$

the typical coefficient matrix and the solution vector for the total flow rate specified system become as shown in Figure 2.14:

	1	2	…	13	14	15	…	36	37	38	…	49	50				
1	C_1	E_1	…	0	0	0	…	0	0	0	…	0	0		P_1		Q_1
2	W_2	C_2	…	0	0	0	…	0	0	0	…	0	0		P_2		Q_2
⋮	⋮	⋮		⋮	⋮	⋮		⋮	⋮	⋮		⋮	⋮		⋮		⋮
13	0	0	…	$C_{13}+W_{14}$	$E_{13}+C_{14}+W_{15}$	$E_{14}+C_{15}$	…	$C_{36}+W_{37}$	$E_{36}+C_{37}+W_{38}$	$E_{37}+C_{38}$	…	0	0		P_{13}		q_{total}
14	0	0	…	$C_{13}+W_{14}$	$E_{13}+C_{14}+W_{15}$	$E_{14}+C_{15}$	…	$C_{36}+W_{37}$	$E_{36}+C_{37}+W_{38}$	$E_{37}+C_{38}$	…	0	0		P_{14}		q_{total}
15	0	0	…	$C_{13}+W_{14}$	$E_{13}+C_{14}+W_{15}$	$E_{14}+C_{15}$	…	$C_{36}+W_{37}$	$E_{36}+C_{37}+W_{38}$	$E_{37}+C_{38}$	…	0	0		P_{15}		q_{total}
⋮	⋮	⋮		⋮	⋮	⋮		⋮	⋮	⋮		⋮	⋮	•	⋮	=	⋮
36	0	0	…	$C_{13}+W_{14}$	$E_{13}+C_{14}+W_{15}$	$E_{14}+C_{15}$	…	$C_{36}+W_{37}$	$E_{36}+C_{37}+W_{38}$	$E_{37}+C_{38}$	…	0	0		P_{36}		q_{total}
37	0	0	…	$C_{13}+W_{14}$	$E_{13}+C_{14}+W_{15}$	$E_{14}+C_{15}$	…	$C_{36}+W_{37}$	$E_{36}+C_{37}+W_{38}$	$E_{37}+C_{38}$	…	0	0		P_{37}		q_{total}
38	0	0	…	$C_{13}+W_{14}$	$E_{13}+C_{14}+W_{15}$	$E_{14}+C_{15}$	…	$C_{36}+W_{37}$	$E_{36}+C_{37}+W_{38}$	$E_{37}+C_{38}$	…	0	0		P_{38}		q_{total}
⋮	⋮	⋮		⋮	⋮	⋮		⋮	⋮	⋮		⋮	⋮		⋮		⋮
49	0	0	…	0	0	0	…	0	0	0	…	C_{49}	E_{49}		P_{49}		Q_{49}
50	0	0	…	0	0	0	…	0	0	0	…	W_{50}	C_{50}		P_{50}		Q_{50}

Figure 2.14: The typical coefficient matrix and the solution vector for the flow rate specified system shown in Figure 2.12.

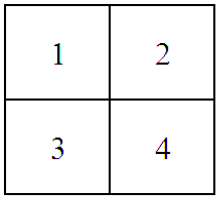
since $P_{13} = P_{14} = P_{15} = P_{36} = P_{37} = P_{38} = P_{sf}$, forty-five unknown pressure can be solved by forty-four continuity equations and a residual equation.

2.3.3 Sparse Matrix Schemes

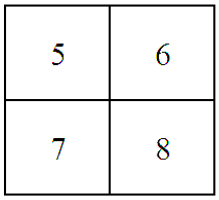
In order to solve the continuity equation writing for all coarse blocks and window blocks simultaneously, two coefficient matrices have to be developed with the transmissibility terms. The size of a coefficient matrix depends on the number of blocks to be solved in the system. For example, a coarse block system holding N reservoir blocks will creates an N by N coefficient matrix containing N^2 elements (see Figure 2.15). In other words, the number of elements in a coefficient matrix will grow quadratically while the number of blocks in the system grows linearly. Computational storage may become a serious problem when the LGR technique is performed in a three dimensional reservoir simulation because of the huge number of locally refined blocks created by the LGR. The simulation speed will also be significantly slowed down due to the computational overhead increases with the increase in the number of the elements in the coefficient matrix. Most of these problems are caused by the zero elements in the coefficient matrix that have no mathematical effect to the calculation but unnecessarily costing memory and computational time.

In order to take advantage of the large number of zero elements in coefficient matrices, special schemes are required to store sparse matrices. The main goal is to represent only the nonzero elements and to able to perform the common matrix operations while saving storage space and increasing the simulation speed.

Two types of sparse matrix scheme were used in the study: (1) the coordinate format, and (2) the diagonally structured format. Each sparse matrix scheme has its advantages and limitations, choosing the right scheme for development of coefficient matrices will optimize the speed and the memory usage of the simulation.



Layer 1



Layer 2

C1	E1	S1	0	B1	0	0	0
W2	C2	0	S2	0	B2	0	0
N3	0	C3	E3	0	0	B3	0
0	N4	W4	C4	0	0	0	B4
A5	0	0	0	C5	E5	S5	0
0	A6	0	0	W6	C6	0	S6
0	0	A7	0	N7	0	C7	E7
0	0	0	A8	0	N8	W8	C8

Jacobian Matrix

Figure 2.15: An example coefficient matrix constructed for an eight-block system.

2.3.3.1 The Coordinate Scheme

The coordinate scheme is the simplest storage scheme for sparse matrix, and it is advantageous for its flexibility. It is used for constructing the sparse coefficient matrix for the window blocks in this study, because of the number of diagonals in the coefficient matrix is not a constant number in the window system.

The data structure of this scheme consists of three arrays: (1) an integer array containing all the column indices of the nonzero elements, (2) an integer array containing their row indices, and (3) a real array containing all their nonzero value. The real value

saved in the array can be in any order, and all three arrays have the length of the number of nonzero element. An example of this scheme is shown in Figure 2.16.

Column	1	2	3	4	5	6	7	8	2	4	…	3	4
Row	1	2	3	4	5	6	7	8	1	3	…	7	8
Value	C1	C2	C3	C4	C5	C6	C7	C8	E1	E3		A7	A8

Figure 2.16**: An example coordinate scheme for the system described in Figure 2.15.**

In addition, the Ax = b type matrix multiplication of the coordinate scheme can be performed as shown in Figure 2.17.

```
for(int i=0; i<number of the element; i++)
{
        int row = A.row(i);
        int col = A.colmn(i);
        b[row] += A.value(i) * x[col];
}
```

Figure 2.17: Performing matrix multiplication to a coordinate sparse matrix.

In the computer program developed for this study, the data type of the value in column and row arrays are declared as integer, which cost 4 bytes of memory for each value. The data type of the nonzero value of elements is declared as double, which cost 8 bytes of memory for each value. That is to say, 16 bytes of the computer memory will be occupied for saving a nonzero element in this type sparse matrix. In the example shown in Figure 2.15, no memory savings will be achieved by the coordinate scheme, however, a significant amount of memory will be saved when more blocks are introduced into the system.

Furthermore, only half of the calculations are needed for the coordinate scheme to perform the same matrix multiplication for the example coefficient matrix of Figure 2.15 as compared to that of the conventional matrix structure. The computational time saving will also increase with the number of the blocks in the system.

2.3.3.2 The Diagonally Structured Scheme

Diagonally structured matrices are common to see when the reservoir is refined by rectangular blocks. A typical two dimensional rectangular gridding system using the standard numbering method will create a penta-diagonal coefficient matrix, and a typical three dimensional rectangular gridding system will create a hepta-diagonal coefficient matrix. In this study, a diagonally structured sparse matrix scheme is used for constructing the coefficient matrix for the coarse-block reservoir system.

The advantage of the scheme is that no memory needs to be used to save row and column indices of an element in the coefficient matrix. However, the number of the diagonals must also be known and the offset of each of the diagonals with respect to the main diagonal must be known, that is why it is not used for the windows system in this study.

For the system in Figure 2.15, the diagonally structured sparse coefficient matrix will be developed as shown in Figure 2.18. In the example, seven arrays are created with their own index (the offset of each of the diagonals with respect to the main diagonal) for the hepta-diagonal structure and only the value of the elements on these diagonals will be saved.

									Index
C	C1	C2	C3	C4	C5	C6	C7	C8	0
E	0	E1	0	E3	0	E5	0	E7	1
S	0	0	S1	S2	0	0	S5	S6	2
W	W2	0	W4	0	W6	0	W8	0	-1
N	N3	N4	0	0	N7	N8	0	0	-2
A	A5	A6	A7	A8	0	0	0	0	-4
B	0	0	0	0	B1	B2	B3	B4	4

Figure 2.18: An example of the diagonally structured scheme for the system described in Figure 2.15.

The $Ax = b$ type matrix multiplication of the diagonally structured scheme can be performed as shown in Figure 2.19.

```
for (int i=0; i<number of diagonals; i++)
{
    if (index>0)
    {
        for (int j=index; j<length of array; j++)
            b[j-index] += Ai[j]*x[j];
    }
    else if (index>0)
    {
        for (int j=0; j< length of array+index; j++)
            b[j-index] += Ai[j]*x[j];
    }
    else
    {
        for (int i=0; j< length of array; j++)
            b[j] += Ai[j]*x{j];
    }
}
```

Figure 2.19: Performing matrix multiplication to a diagonally structured sparse matrix.

The diagonally structured scheme uses less memory as compared to coordinate scheme only if most elements in a matrix are located on a small number of diagonals. The speed of performing a matrix multiplication for a diagonally structured sparse matrix can be slower than doing the same operation for a sparse matrix using the coordinate scheme. This is because some zero elements are also saved by the diagonally structure scheme, and an extra time of calculation will need to be allocated for these zero elements.

2.3.4 Solver: The Preconditioned Bi-Conjugate Gradient Method

The simulation speed of the developed computer program is mainly associated with the convergence speed of the solver. Commonly used solvers in reservoir simulation, like SIP, spends additional time to select a proper number of iteration parameters. This procedure will affect the performance of the program[7]. In addition to the problem of speed, SIP procedure has convergence problems in this study when the window system becomes very large.

The preconditioned Bi-Conjugate Gradient Method not only provides a stable solution to the system, the convergence speed is also significantly increased by using a preconditioner[7]. In this study, a diagonal preconditioner is being used, which is constructed by the main diagonal of the coefficient matrix. A diagonal preconditioner performs effectively when the amount of blocks introduced by LGR is not too large. When the size of coefficient matrix becomes larger and more complex, a diagonal preconditioner is not effective anymore, the speed of convergence then slowed down.

This problem can be solved by implementing a more complex preconditioner which represents the coefficient matrix better, like D-ILU preconditioner[7].

2.3.5 Communication between Coarse Cells and Window Cells

In implementing the LGR method , two seperated systems are developed for a single reservoir, which are the window system and the coarse block system. Since the two systems can not be solved simultaneously, they need to be solved one after another to update their pressure profile until the pressure profiles of the window and the coarse systems have a good agreement.

After initializing the coarse and window systems, the window system is then to be solved first. Residual equation is used to calculate the production rate of the well in every coarse blocks holding well blocks. These values are transfered to the coarse system as known production rate from every coarse blocks, then the coarse system is solved with flow rate specification condition.

Because Dirichlet type boundary condition is specified between the window system and the coarse block system, the neighboring pressure of the boundary blocks in window system need to be updated by the coarse system. This implies that after the pressure of the coarse system is updated, the pressure of the surrounding windows should be transferred back to the window system to provide the boundary condition of the windows.

The aforementioned feedback between coarse system and high resolution cells will be continuously performed until the difference of pressure profiles between iterations in the coarse system and the window system becomes negligible.

Once the two systems converge within the desired degree of accuracy, the simulation can then continue for the next time-step computations.

2.3.6 Computer Model and Flow Chart

The computer program for this study is developed using C++ under Microsoft Visual Studio 2010. Single-phase compressible fluid model is implemented for three dimensional simulations in this program. Following are the capability of this computer program:

(1) Automatic LGR implementation:

After specifying the starting point and the ending point of the perforated well section in the coarse block system, the computer code automatically generates LGR windows based on the well structure and the rules we have discussed in this chapter. The window block expanding factors α and β values can be specified based on the required accuracy.

(2) Development of a complex well structure:

Several separated well branches can be connected together to construct a multi-later well or a complex well structure. Undulating well can also be approximated by specifying multiple points in the coarse blocks.

(3) Data storage and solver:

Two kinds of sparse matrix data scheme are provided to users, coordinate scheme and diagonally constructed scheme, and one can be switched from one to the other. The preconditioned bi-conjugate gradient stabilized method is utilized to solve the system, and a diagonal preconditioner is used for increasing the convergence speed. This solver is proven to be fast and stable for a single phase system studied by Goktas and Ertekin[7].

Figure 2.20 is the flow chart of the computer program. Automatic time-step procedure is implemented. The simulation starts from a very small time-step (default is 0.001 days) and increases gradually until it meets the maximum time-step set at the beginning. The value of maximum time-step will affect the pressure convergence during communication between coarse system and window system. The more the LGR blocks introduced to the system, the smaller the maximum time-step should be set. Generally, a 10-day time step is proved to be practical for single vertical well, horizontal well, or slanted well. However, the time-step should not larger than five days when a complex structure is placed into the system.

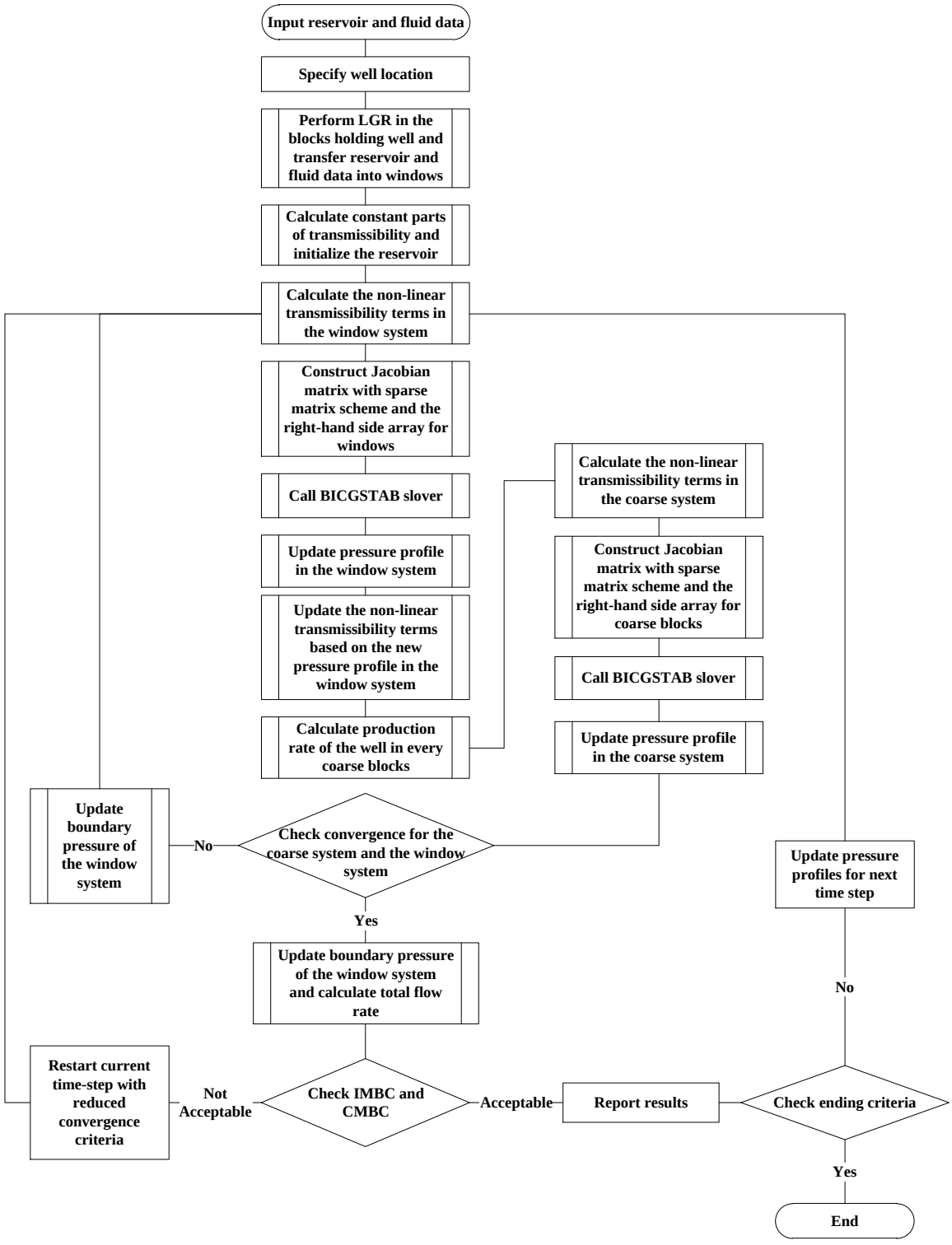


Figure 2.20: Flow chart for the computer code developed in this study.

CHAPTER 3

SINGLE-PHASE COMPRESSIBLE FLUID MODEL VERIFICATION

In this chapter, the production performance of a vertical well, a horizontal well, and a slanted well in a single-phase compressible fluid reservoir will be simulated in three dimensions using the model developed in this work. Results, then, will be compared against results generated by commercial model. After the model verification, production performances of wells with complex structures will be simulated under different scenarios.

Goktas and Ertekin showed that a radial-cylindrical hole can be approximated by a series of rectangular blocks when the rectangular blocks occupy the same volume on the cylindrical hole [7]. Accordingly, the size of well blocks in the model is set by using the following equation:

$$\Delta x = \Delta y = \Delta z = \sqrt{\pi r_w^2} \tag{3.1}$$

In addition, all the verification runs are performed with the reservoir and fluid properties listed in Table 3.1, only the number of coarse block, α ,and β value are changed in order to compare the accuracy of simulation results between the developed model and the commercial model.

3.1 Simulation of Productivity of a Single Vertical Well

A constant temperature, homogeneous and isotropic reservoir with no-flow outer boundaries is considered here, the gravity terms are ignored, and the fluid and rock properties are defined as in Table 3.1. A 30ft vertical well is placed in the middle of the reservoir (see Figure 3.1). Several refinement degrees to the reservoir are simulated by the commercial model in order to observe the improvement of accuracy related to the number of refined blocks in the reservoir. Table 3.2 shows the refinement information of the runs in the comparison.

Table 3.1 Reservoir and fluid properties used for model verifications

Reservoir Size; x,y,z	1500 ft * 1500 ft * 60 ft
Porosity, ϕ	0.1
Permeability; $k_x=k_y=k_z$	3 md
Initial Reservoir Pressure, P_i	3000 psia
$T_{reservoir}$	660˚R
$P_{standard\ condition}$	14.7 psia
$T_{standard\ condition}$	520˚R
Fluid Molecular weight, M_w	16.043
Fluid Critical Temperature, T_{cr}	343.1˚R
Fluid Critical Pressure, P_{cr}	666.4 psi
Sandface Pressure, P_{sf}	1000 psia
Wellbore radius, r_w	0.25 ft

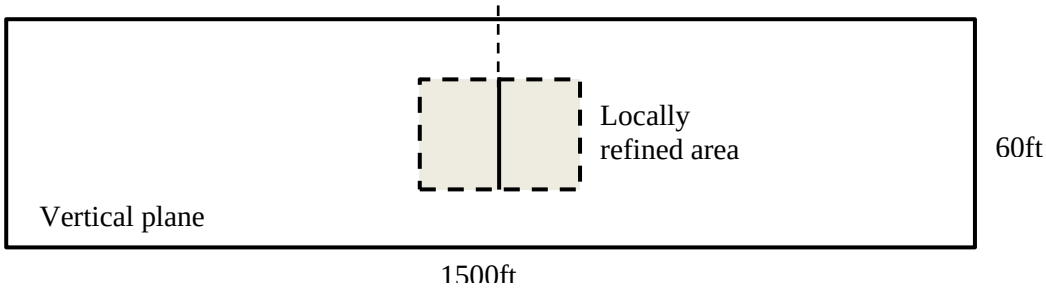


Figure 3.1: Placement of the 30ft vertical well in the reservoir described in Table 3.1.

Table 3.2 Refinement information used in verification runs in Figure 3.2				
Case	α=β	x dimension	y dimension	z dimension
Commercial model-1		15*100 ft	15*100 ft	2*10ft, 4*5ft, 2*10ft
Commercial model-2		15*100 ft	15*100 ft	40*1.5 ft
Residual method-1	1.3	15*100 ft	15*100 ft	6*10 ft
Residual method-2	1.5	15*100 ft	15*100 ft	6*10 ft
Residual method-3	2.0	15*100 ft	15*100 ft	6*10 ft

Theoretically, a higher degree of refinement of the reservoir introduced to the commercial model should turn out a more accurate result. Similarly in the residual method, when α and β factors become smaller, a smoother transition in the windows system can be obtained and more window blocks will be introduced to the system theoretically yields more accurate results.

Now, let us look into the results shown in Figure 3.2. Production rates in the early time received from the commercial model decreased when the refinement degree of reservoir increased. On the other side, production rate in the early time received from residual method increased when α and β values became smaller. This phenomenon shows a trend that the difference between two simulation programs became smaller when the accuracy of both method increased. In addition, the late time production rate from two simulation programs became close according to the rescaled plot shown in Figure 3.3.

The comparison of cumulative gas produced is shown in Figure 3.4. The amounts of cumulative gas produced after 2500 days between two models are slightly different. This difference may be caused by the methods we used to calculate the Z factors are

different. Peng-Robinson EOS is utilized in the commercial model, but Dranchuk-Abou-Kassem EOS is used in the program developed for this study. Different values of the z factor will affect the OGIP values calculated for the reservoir.

However, the results in Figure 3.2 and Figure 3.3 still showed a good match between commercial model and the residual method when the performance of a vertical well is being simulated.

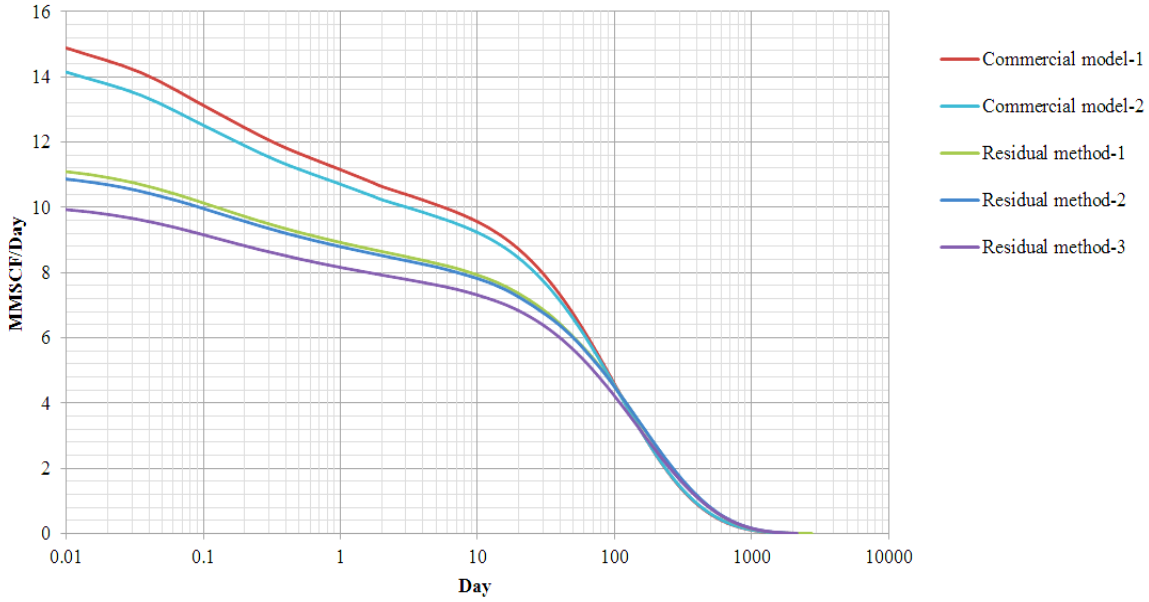


Figure 3.2: Comparison of flow rate of 30ft vertical well.

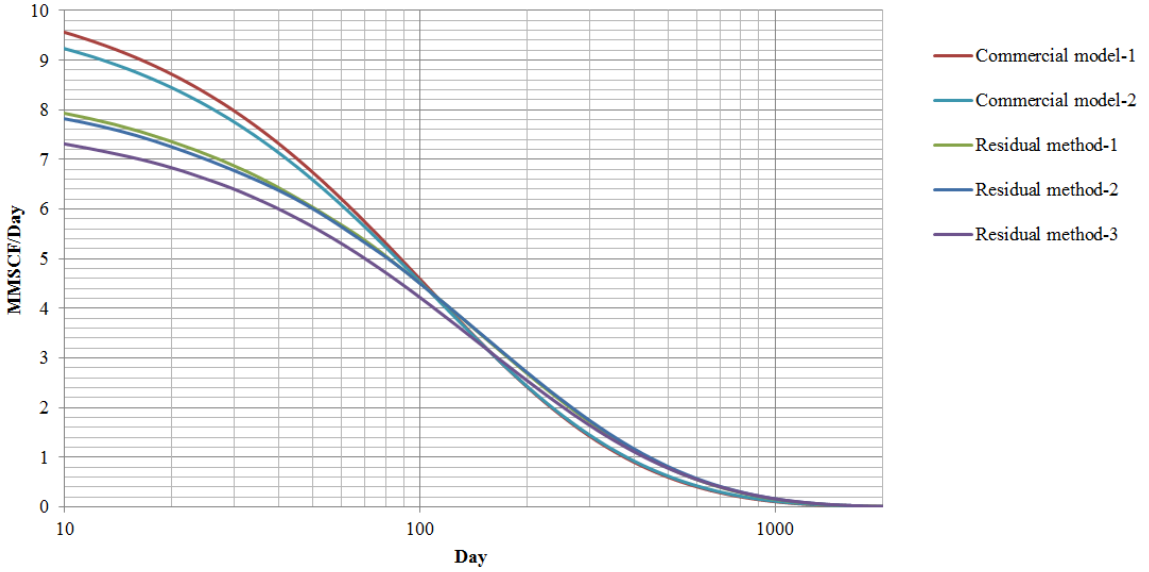


Figure 3.3: Rescaled comparison of flow rate of 30ft vertical well.

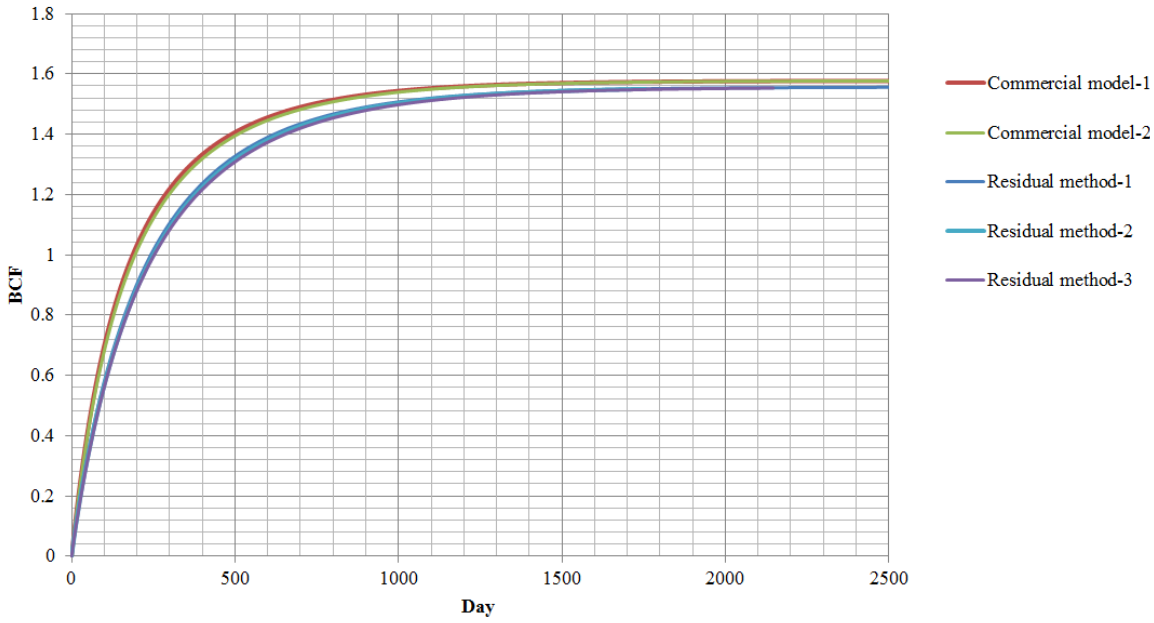


Figure 3.4: Comparison of cumulative gas of 30ft vertical well.

3.2 Simulation of Productivity of a Single Horizontal Well

Now, let us review the validation runs for a horizontal well. A 100ft x-direction horizontal well is placed at the middle of the reservoir (see Figure 3.5) with properties in Table 3.1. Table 3.3 displays the refinement degrees used for the comparison runs.

Table 3.3 Refinement information of for verification runs in Figure 3.6

Case	α=β	x dimension	y dimension	z dimension
Commercial model-1		6*100ft, 6*50ft, 6*100ft	6*100ft, 5*60ft, 6*100ft	10ft, 3*6ft, 4ft, 3*6ft, 10ft
Commercial model-2		5*100ft, 2*50ft, 12*25ft, 2*50ft, 5*100ft	6*100ft, 11*27.2727ft, 6*100ft	10ft, 3*6ft, 4ft, 3*6ft, 10ft
Commercial model-3		5*10ft0, 2*75ft, 40*5ft, 2*75ft, 5*100ft	6*100ft, 7*42.8571ft, 6*100ft	10ft, 3*6ft, 4ft, 3*6ft, 10ft
Residual method-1	1.5	15*100 ft	15*100 ft	5*12 ft
Residual method-2	2.0	15*100 ft	15*100 ft	5*12 ft
Residual method-3	3.0	15*100 ft	15*100 ft	5*12 ft

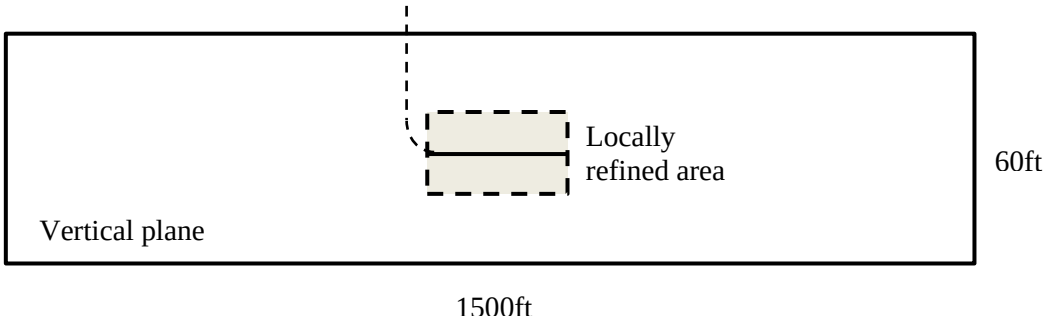


Figure 3.5: Placement of the 100ft horizontal well in the reservoir described in Table 3.3.

Figure 3.6 shows the comparison of the flow rates from the commercial model and the residual method. The flow rate predictions from two models seem significantly different during the early time (< 100 days), see the curves commercial model-3 and residual method-1 in Figure 3.6. The flow rate differences between two models are reduced by increasing the refinement degree of the reservoir, and two curves merge by 30 days.

A similar phenomenon can be observed in Figure 3.7, on the comparison of cumulative gas. Amounts of production through the horizontal well are very different when the refinement degree is not high enough. After modifying the refinement degree of the reservoir in the commercial model, the cumulative gas curves turn out to have good match among each other.

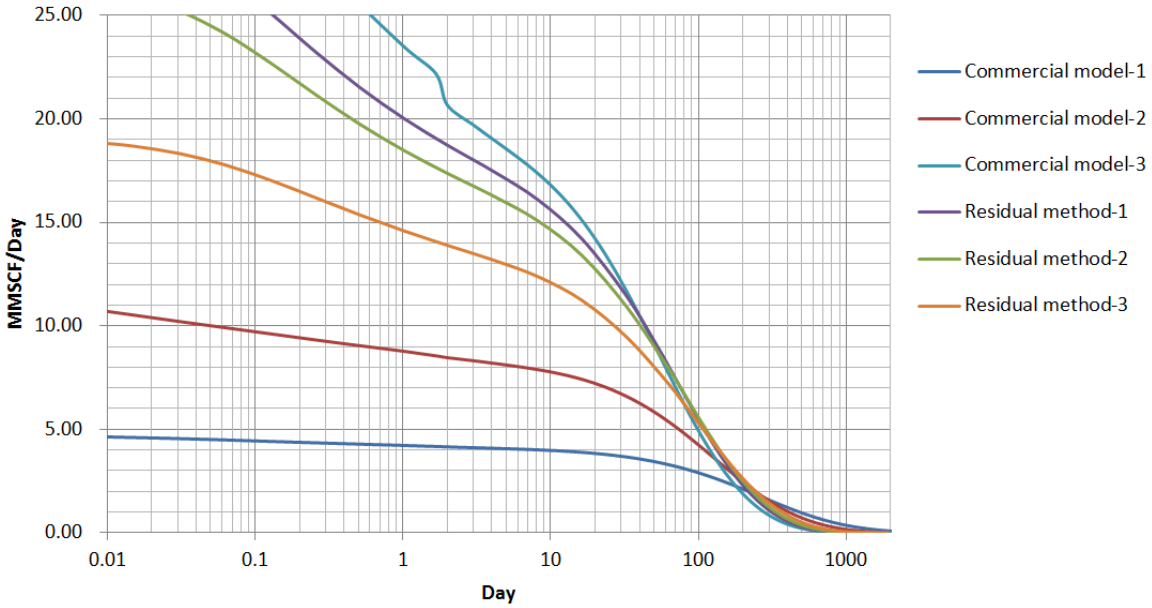


Figure 3.6: Comparison of flow rate of 100ft horizontal well.

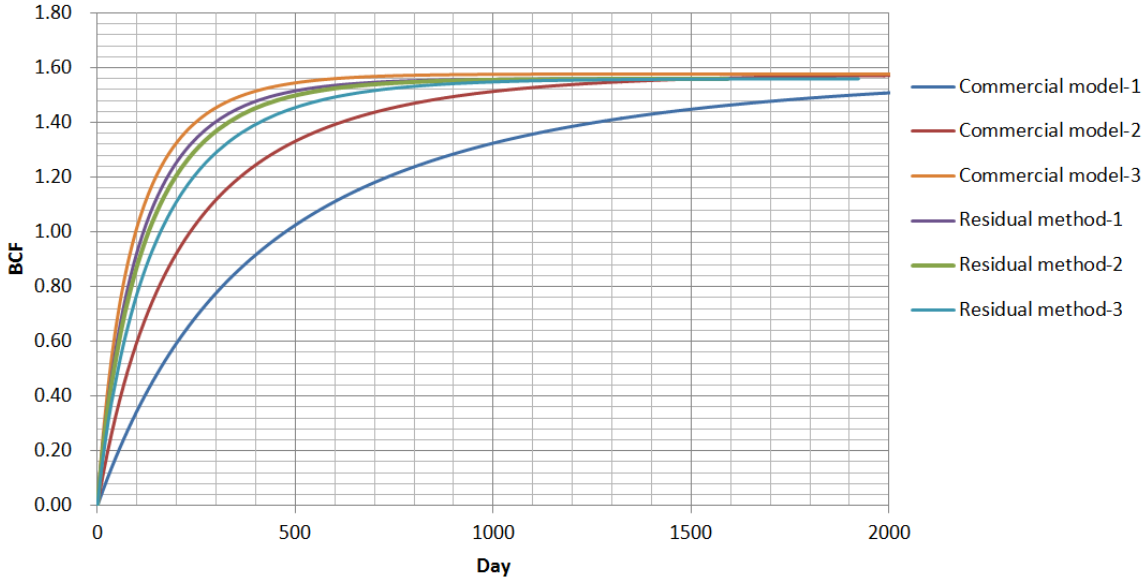


Figure 3.7: Comparison of cumulative gas of 100ft horizontal well.

3.3 Simulation of Productivity of Slanted Wells

The verification runs in previous sections indicate that the residual method is able to simulate the production rate of vertical or horizontal wells accurately without implementing Peaceman's wellbore model. However, before running a simulation for a wish-bone well, a verification of the performance of a slanted well should also be tested.

A 200.25 ft slanted well is placed in the middle of the reservoir (see Figure 3.8) with properties in Table 3.1, and the refinement degrees of the reservoir for the verification runs are shown in Table 3.4.

Table 3.4 Refinement information used in verification runs in Figure 3.9				
Case	α=β	x dimension	y dimension	z dimension
Commercial model-1		6*100ft, 6*50ft, 6*100ft	15*100ft	2*10ft, 5ft, 4*2.5ft, 5ft, 2*10ft
Commercial model-2		6*100ft, 2*25ft, 10*20ft, 2*25ft, 6*100ft	6*100, 5*60ft, 6*100	2*10ft, 5ft, 10*1ft, 5ft, 2*10ft
Commercial model-3		6*100ft, 30*10ft, 6*100ft	15*100ft	2*12.5ft, 10*1ft, 2*12.5ft
Residual method-1	1.5	15*100 ft	15*100 ft	6*10 ft
Residual method-2	2.0	15*100 ft	15*100 ft	6*10 ft
Residual method-3	3.0	15*100 ft	15*100 ft	6*10 ft

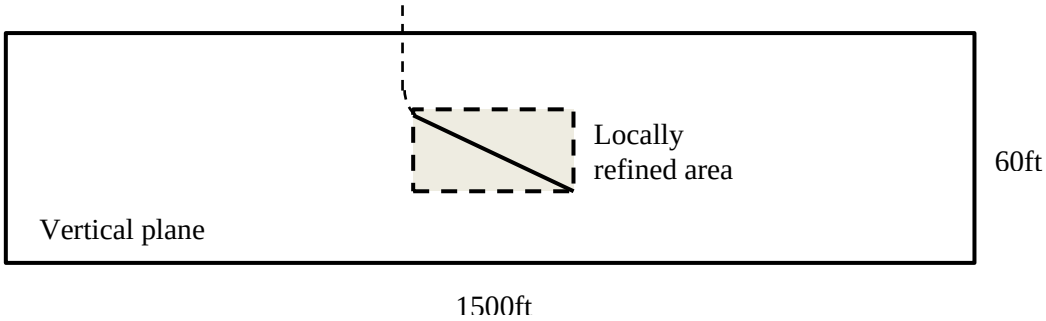


Figure 3.8: Placement of the 200.25 ft slanted well in the reservoir described in Table 3.4.

The results are shown in Figure 3.9 and Figure 3.10. The difference of production rates between the commercial model and the residual method is obvious. That is to say, one of the models is generating a significant error when simulating this kind of well structure.

Now, we compare the production rate predictions shown in Figure 3.6. Theoretically, the production rate of the slanted well placed in the reservoir should be slightly higher than the horizontal well we had in the last section because that the effective length of the slanted well is longer than the horizontal well achieving a better reservoir contact. However, when a similar degree of reservoir refinement is performed in the commercial model, the production rate shown in Figure 3.9 for the slanted well is way lower than the production rate for the horizontal well shown in Figure 3.6. This result is not expected and is not reasonable.

The same comparison between the horizontal well and the slanted well is also performed to the residual method (see Figure 3.11). The production rate from the slanted well is higher than the horizontal well when the residual method is used for the simulation.

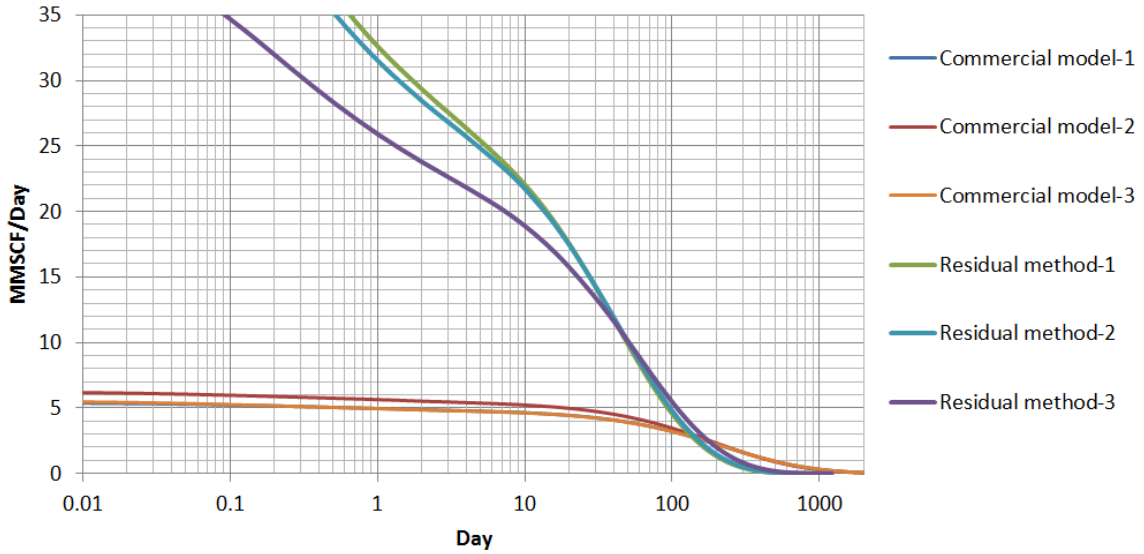


Figure 3.9: Comparison of flow rate of 200.25ft slanted well.

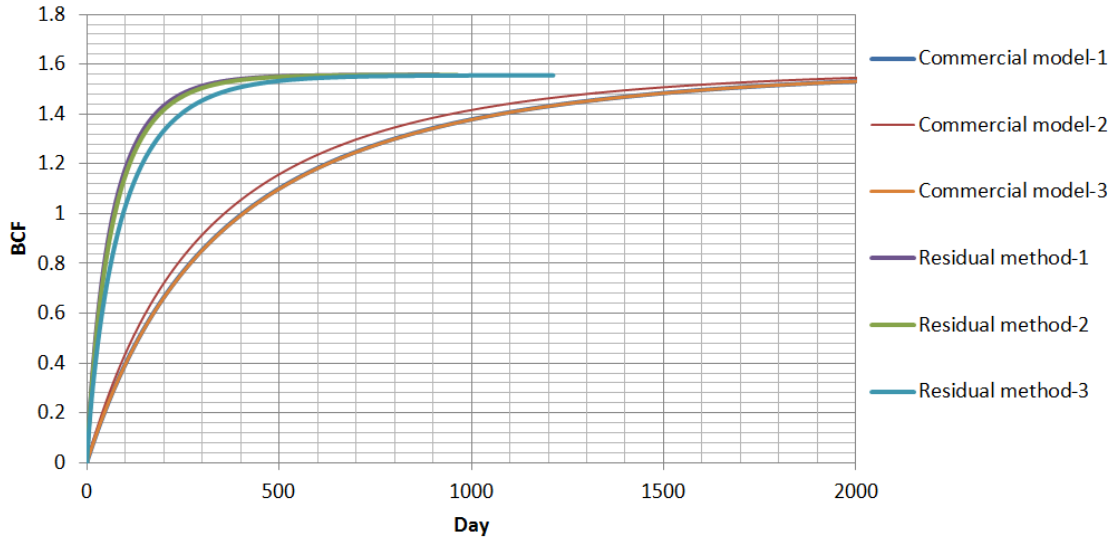


Figure 3.10: Comparison of cumulative gas of 200.25ft slanted well.

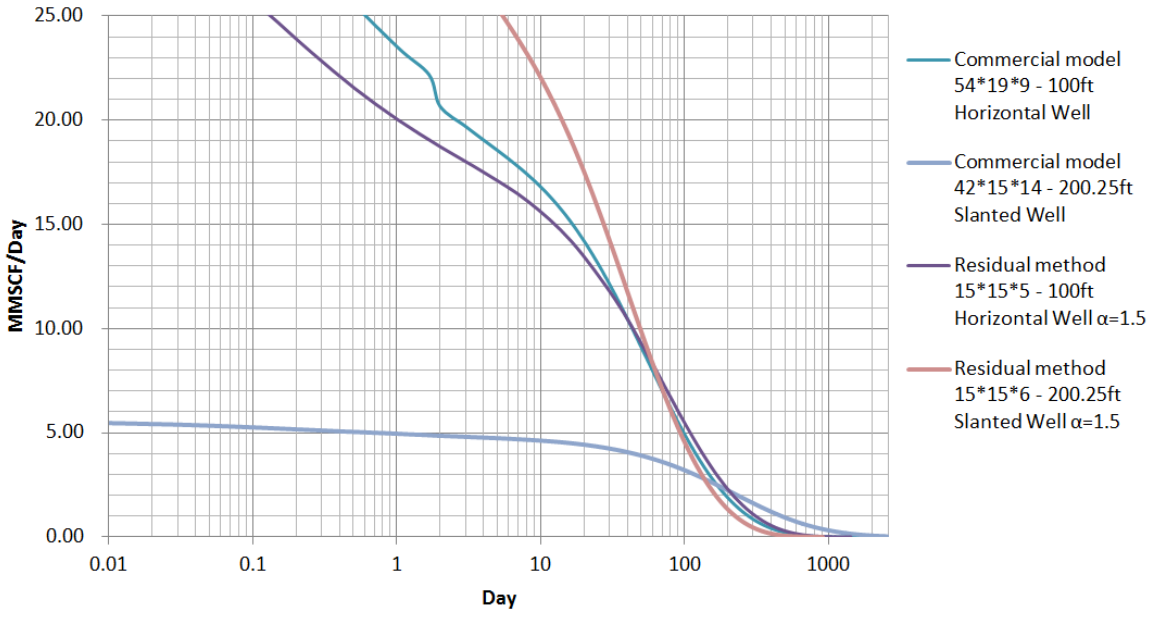


Figure 3.11: Comparison of the performance of the horizontal well and the slanted well between commercial simulator and residual method.

To verify the capability of the commercial model to simulate the performance of slanted wells, another set of verification is presented. The production performances of five wells with different angle and length are simulated with the commercial model, see Table 3.5 and Figure 3.12 for information of well. All the wells are placed in the middle of the homogeneous reservoir with permeability equal to 3md in all direction. The result is shown in Figure 3.13 and Figure 3.14.

Table 3.5 Lengths and angels of wells in verification runs in Figure 3.13

Case	Well Length	Angle to Horizontal
(1) Vertical Well	10ft	90 ˚
(2) Slanted Well-1	67.41ft	8.53 ˚
(3) Slanted Well-2	200.25ft	2.86 ˚
(4) Slanted Well-3	400.12ft	1.43 ˚
(5) Horizontal Well	400ft	0˚

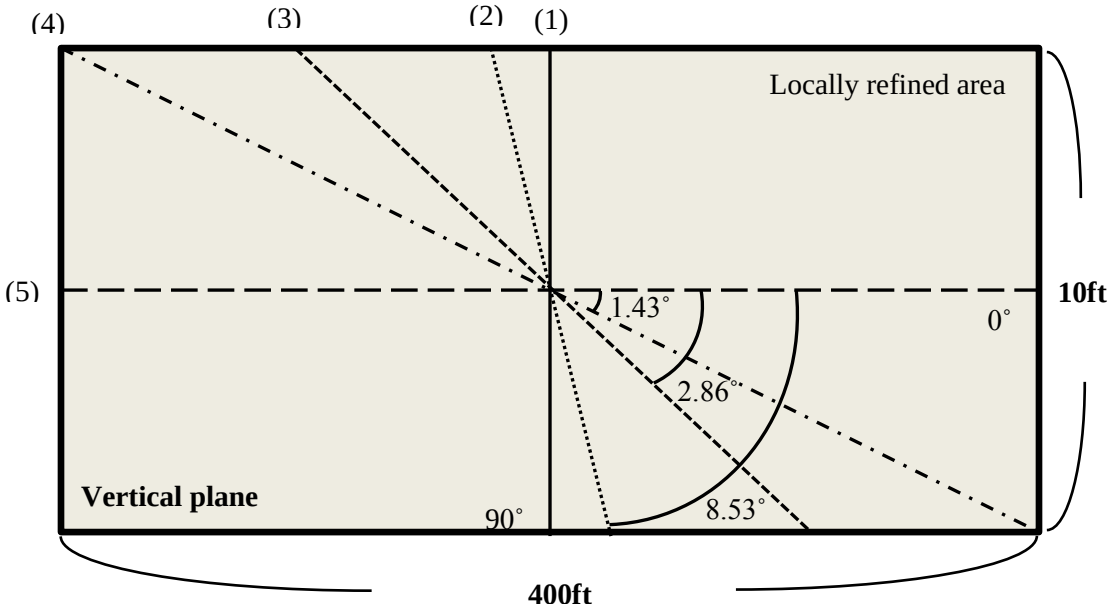


Figure 3.12: Placements of wells described in Table 3.5 in the middle of the reservoir.

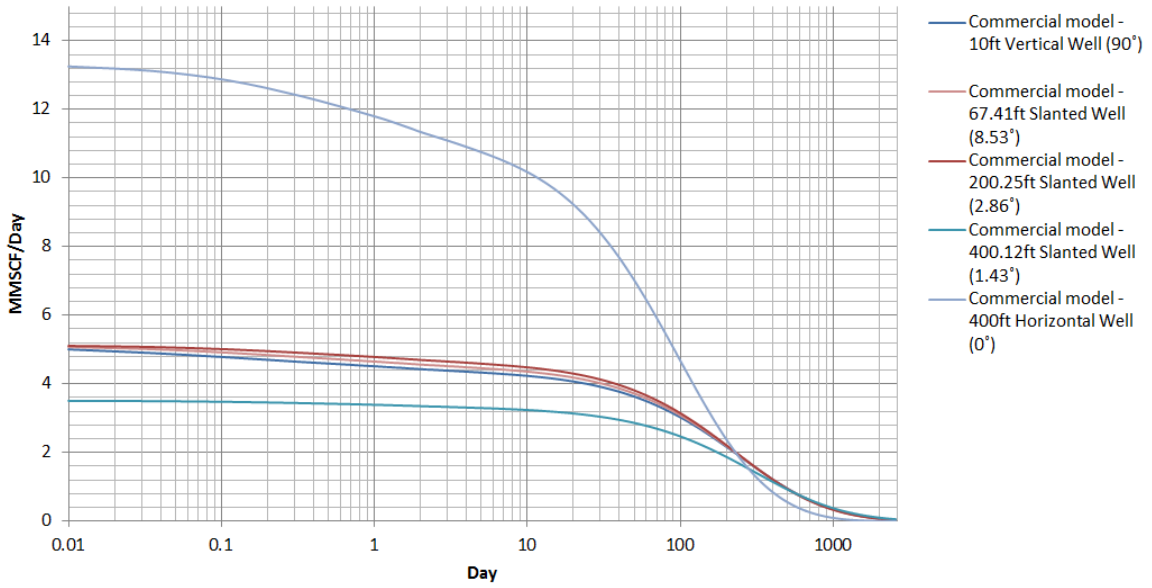


Figure 3.13: Comparison of flow rates of wells with different effective lengths and angles – Commercial simulator.

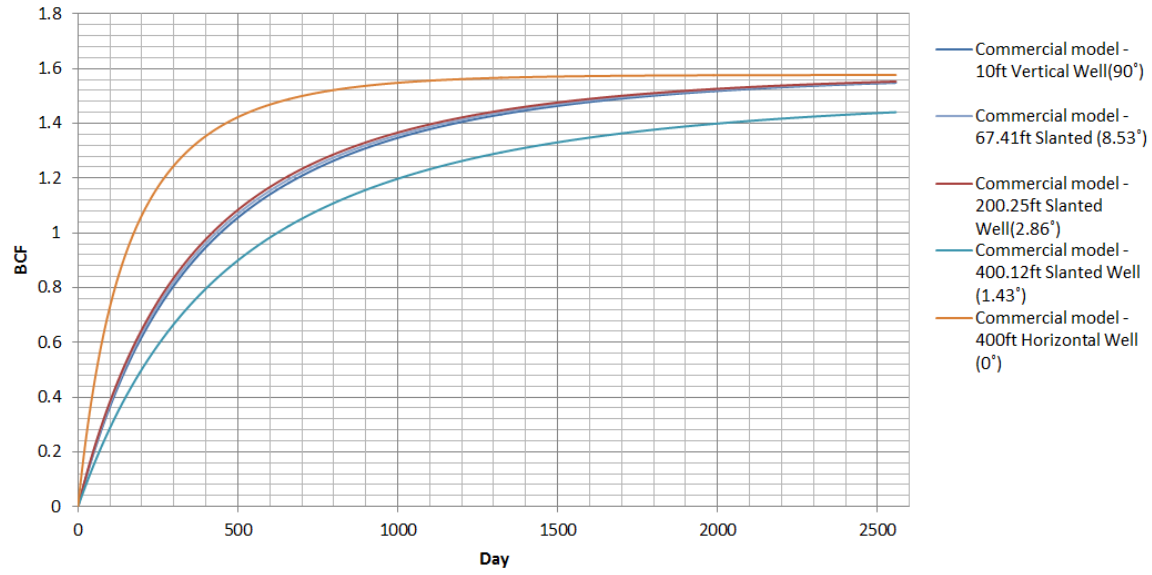


Figure 3.14: Comparison of cumulative gas of wells with different effective lengths and angles - Commercial simulator.

Theoretically, in a homogeneous reservoir, the production rates of wells during the early time should be $q_4 > q_5 > q_3 > q_2 > q_1$ due to the effective lengths of the wells. However, no significant difference between three slanted wells production rates can be found in Figure 3.13 and Figure 3.14, even the length of the wells are very different.

The same wells data set in Table 3.5 is used for verifying the accuracy of program implementing residual method. Results are shown in Figure 3.15 and Figure 3.16, which confirm with our expectation that the sequence of production rate should be $q_4 > q_5 > q_3 > q_2 > q_1$ due to the difference between their effective lengths. The differences between production rates of wells with different length are also reasonable, for example, the production rate of case (4) should be slightly higher than case (5) but should also be very close due to the small angle of case (4) and practically the same effective well lengths.

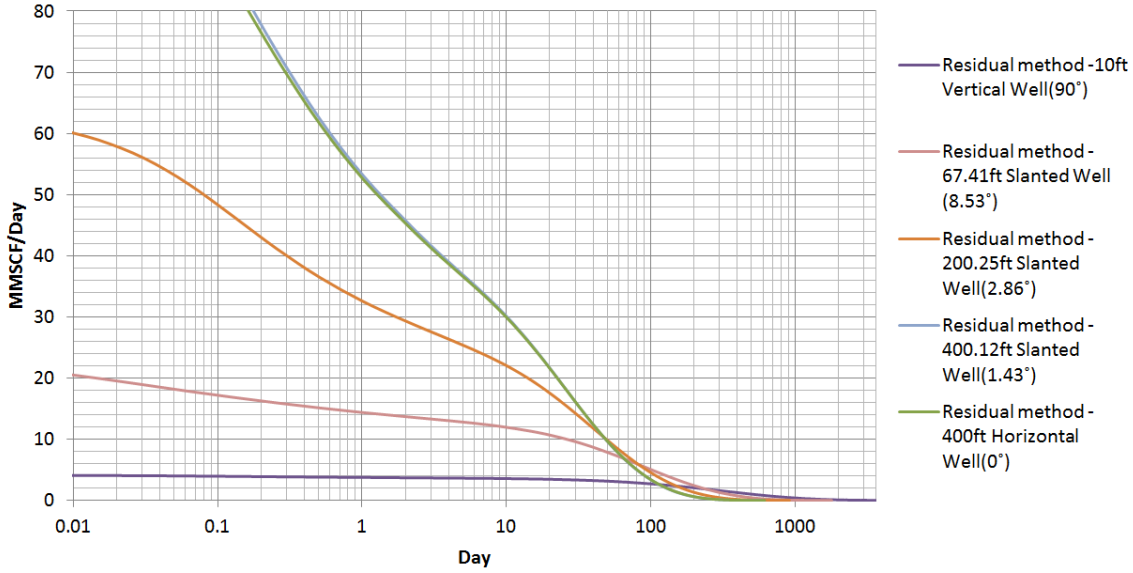


Figure 3.15: Comparison of flow rate of wells with different effective lengths and angles – Residual Method.

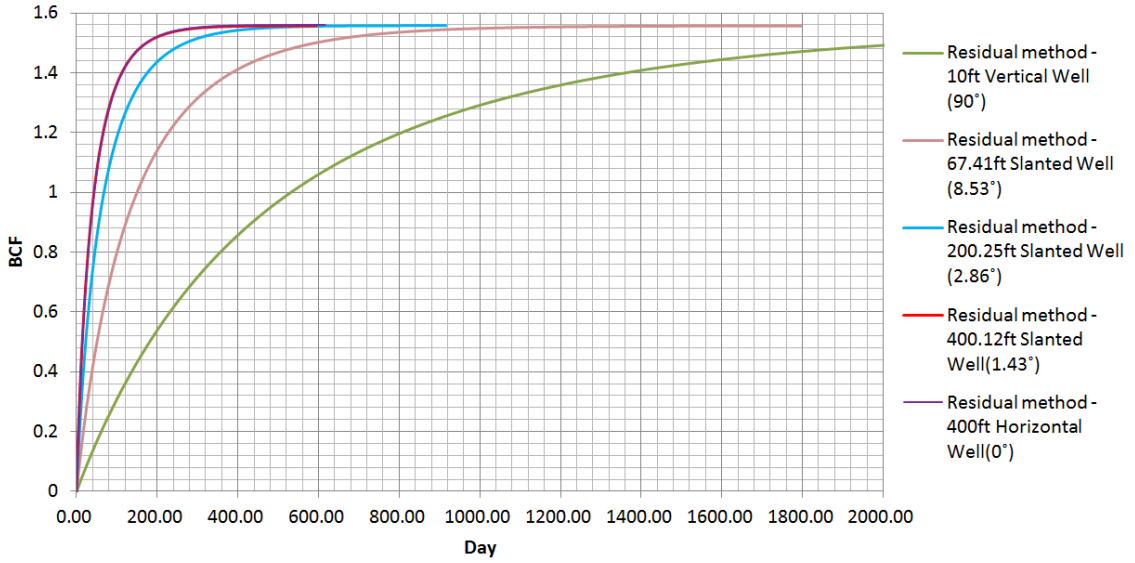


Figure 3.16: Comparison of cumulative gas of wells with different effective lengths and angles – Residual Method.

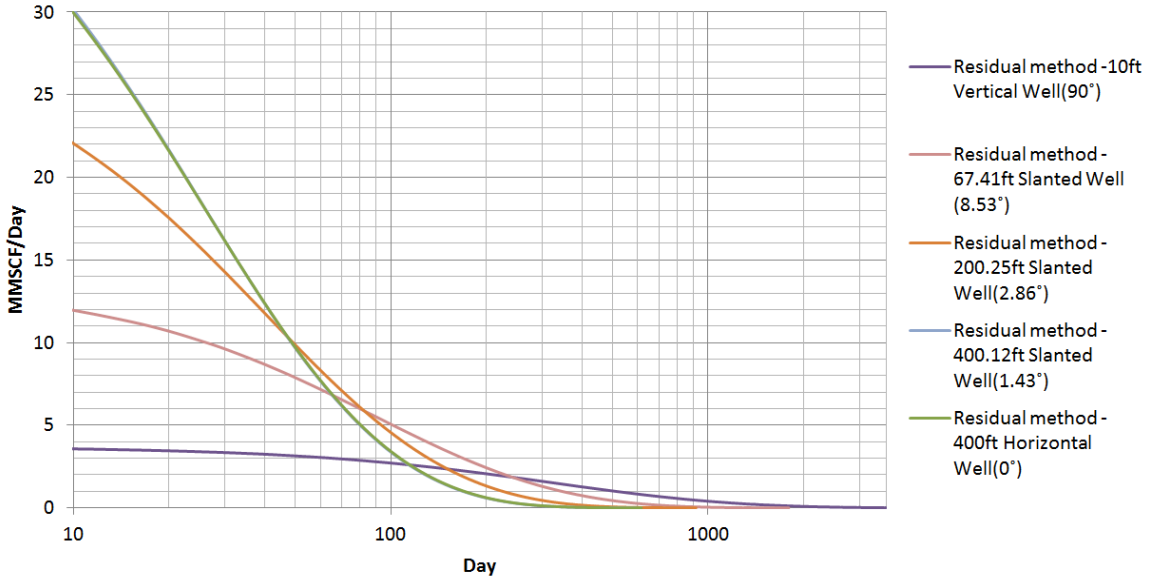


Figure 3.17: Rescaled comparison of flow rate of wells shown in Figure 3.15.

A similar set of verification test is performed to double check the previous results. The performances of four wells with different lengths and angles are simulated. Different from the previous set of tests, the thickness of the reservoir was increased from 60ft to 550ft and formation thickness of the blocks in which the well is placed is increased from 10ft to 200ft. The settings of wells are shown in Table 3.6 and Figure 3.18, and the results are shown in Figure 3.19, Figure 3.20, Figure 3.21, and Figure 3.22.

.

Table 3.6 Lengths and angels of wells in verification runs in Figure 3.19

Case	Well Length	Angle to Horizontal
(1) Vertical Well	200ft	90˚
(2) Slanted Well-1	282.84ft	45˚
(3) Slanted Well-2	412.31ft	14.04˚
(4) Horizontal Well	400ft	0˚

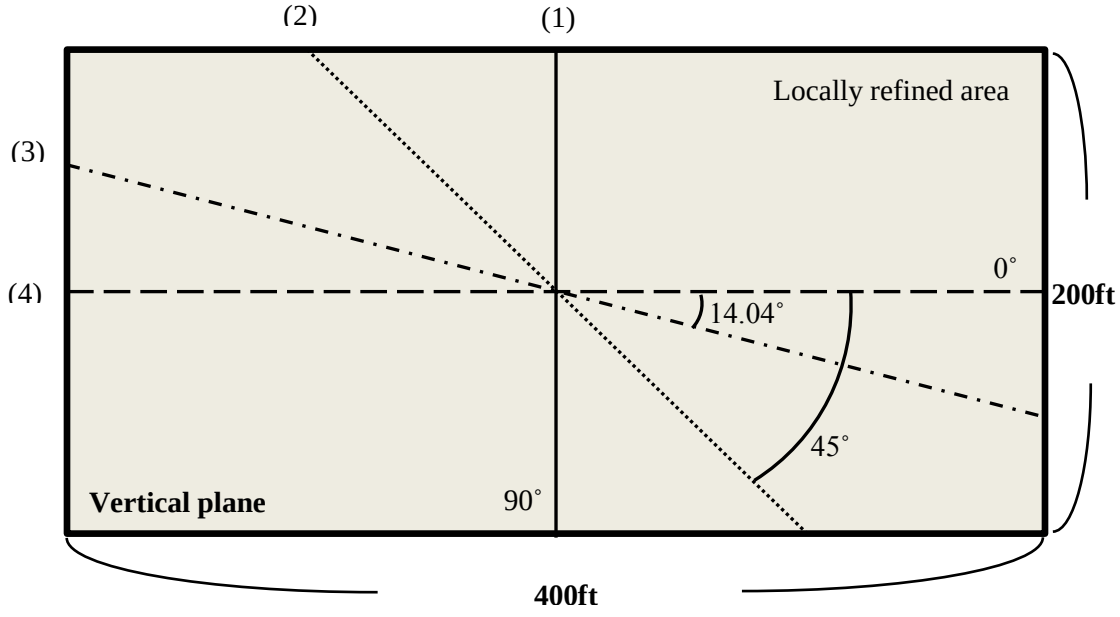


Figure 3.18: Placements of wells described in Table 3.6 in the middle of the reservoir.

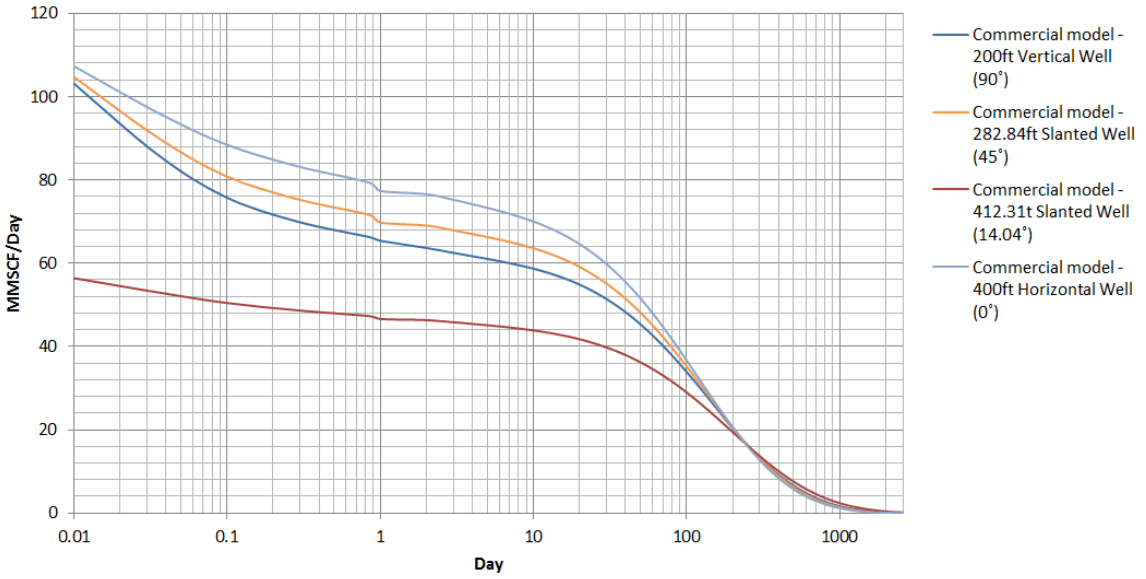


Figure 3.19: Comparison of flow rate of wells with different effective lengths and angles – Commercial simulator.

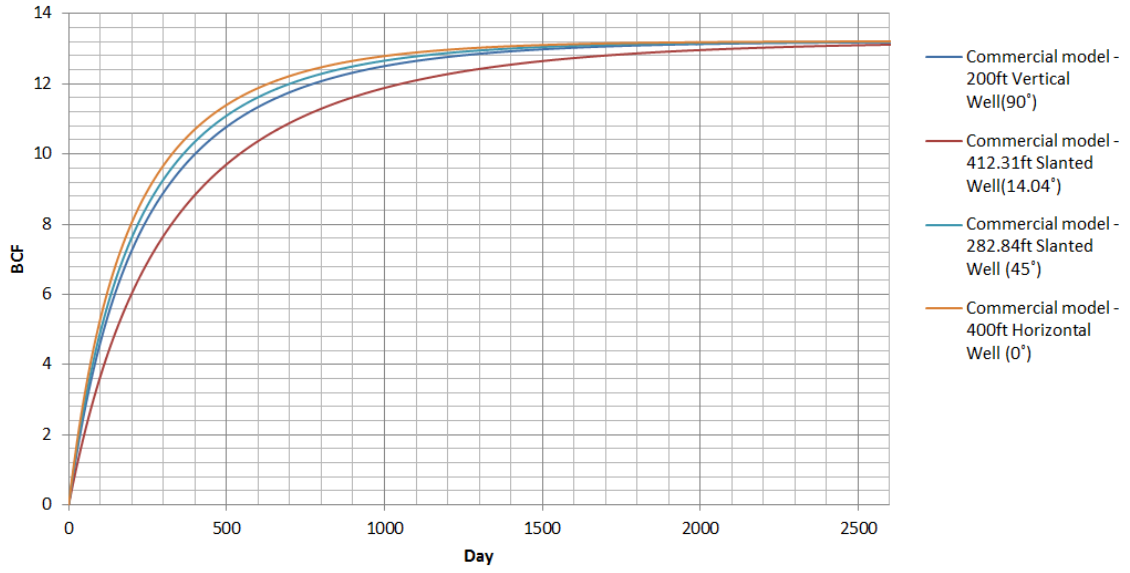


Figure 3.20: Comparison of cumulative gas of wells with different effective lengths and angles – Commercial simulator.

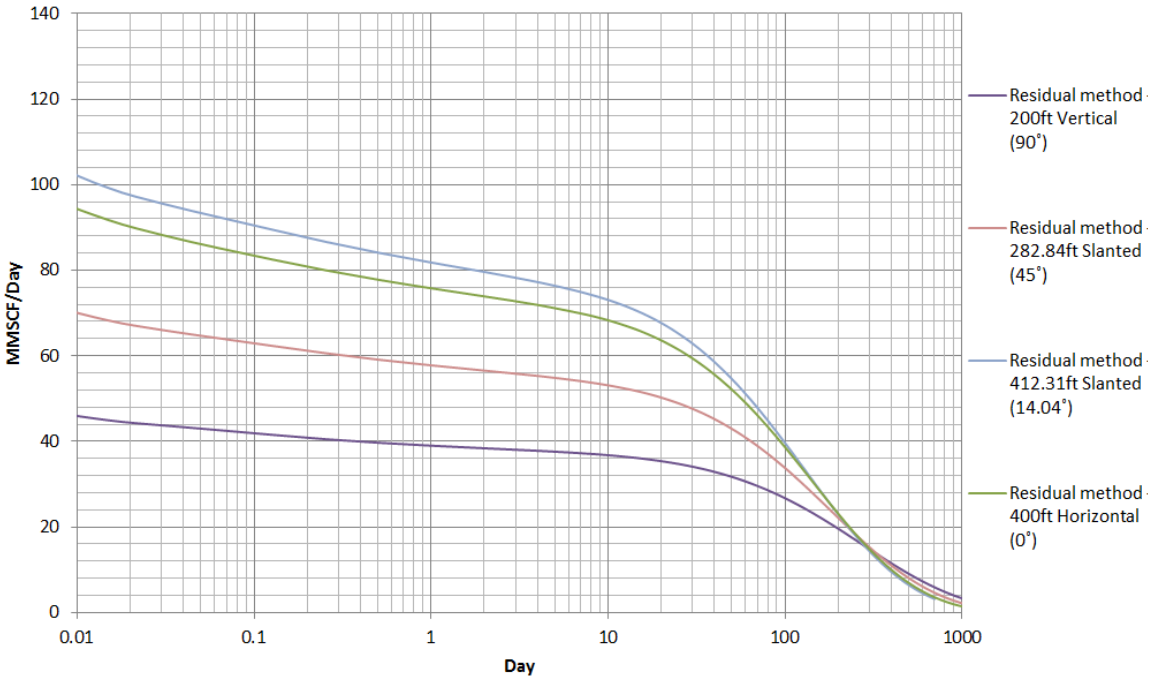


Figure 3.21: Comparison of flow rate of wells with different effective lengths and angles – Residual Method.

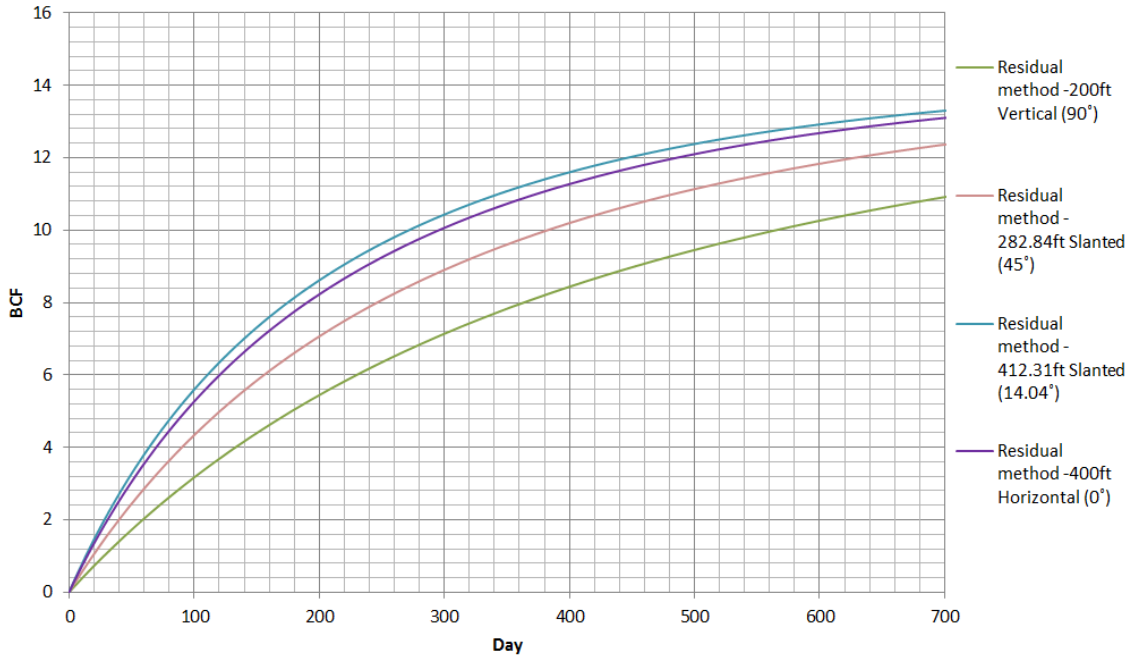


Figure 3.22: Comparison of cumulative gas of wells with different effective lengths and angles – Residual Method.

Again, when simulating slanted wells, commercial simulator returned an unreasonable result that the production rate of slanted well-2 (412.31ft, 14.04 ˚) was lower than any of the others. Several refinement degrees had been applied to this well, but all of them produced similar results. On the other hand, residual method kept providing reasonable and reliable results which matched our expectations.

From the comparisons presented above, we can conclude that commercial simulator could perform an accurate simulation when a slanted well is introduced to the reservoir, the production rates of slanted wells will be under estimated. Besides, from the verification runs presented above, the residual method utilized in this study is proven to be capable of providing a more reliable result comparing to the commercial simulator when a well is placed into the reservoir diagonally. The observation is important to this study, because a slanted well forms the basic element of a fish-bone structure or any other complex well structures. After assuming that the developed model is capable representing the complex structures accurately, the simulation runs for complex well system such as fish-bone structures can then become more meaningful.

3.4 Simulation of Production Rate Specified Problems

The sandface pressure, P_{sf} can be calculated by solving Equation 2.31when the production rate is specified in the simulation. To prove that the model is capable of simulating this type of problems, verification runs are performed as described below.

A 30ft vertical well is place in the middle of the reservoir with the properties shown in Table 3.7. The production rate of gas is specified as 4 MMSCF/Day, the production rate and sandface pressure response profiles are shown in Figure 3.23 and Figure 3.24, respectively.

Table 3.7 Reservoir and fluid properties used in the flow rate specified problem.

Reservoir Size; x,y,z	1500 ft * 1500 ft * 60 ft
Porosity, ϕ	0.1
Permeability; $k_x=k_y=k_z$	3 md
Initial Reservoir Pressure, P_i	3000 psia
$T_{reservoir}$	660˚R
$P_{standard\ condition}$	14.7 psia
$T_{standard\ condition}$	520˚R
Fluid Molecular weight, M_w	16.043
Fluid Critical Temperature, T_{cr}	343.1˚R
Fluid Critical Pressure, P_{cr}	666.4 psi
Specified Gas Rate, q_g	4 MMSCF/Day
Wellbore radius, r_w	0.25 ft

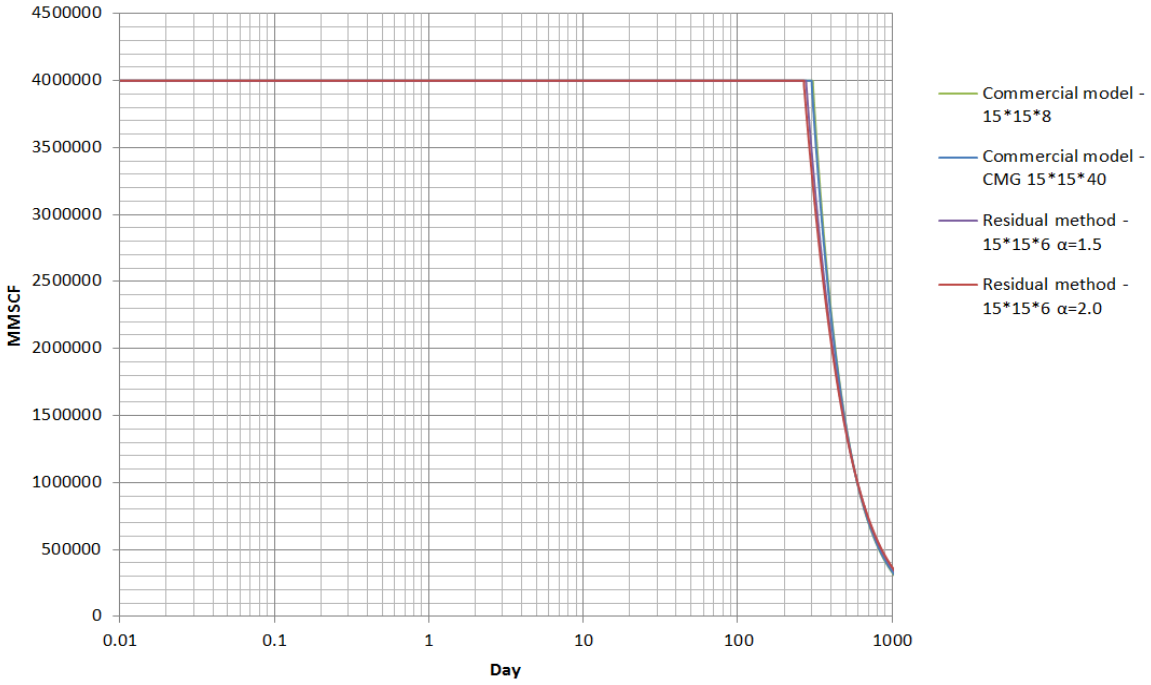


Figure 3.23: Comparison of production rates after flow rate specification is switched to sandface pressure specification.

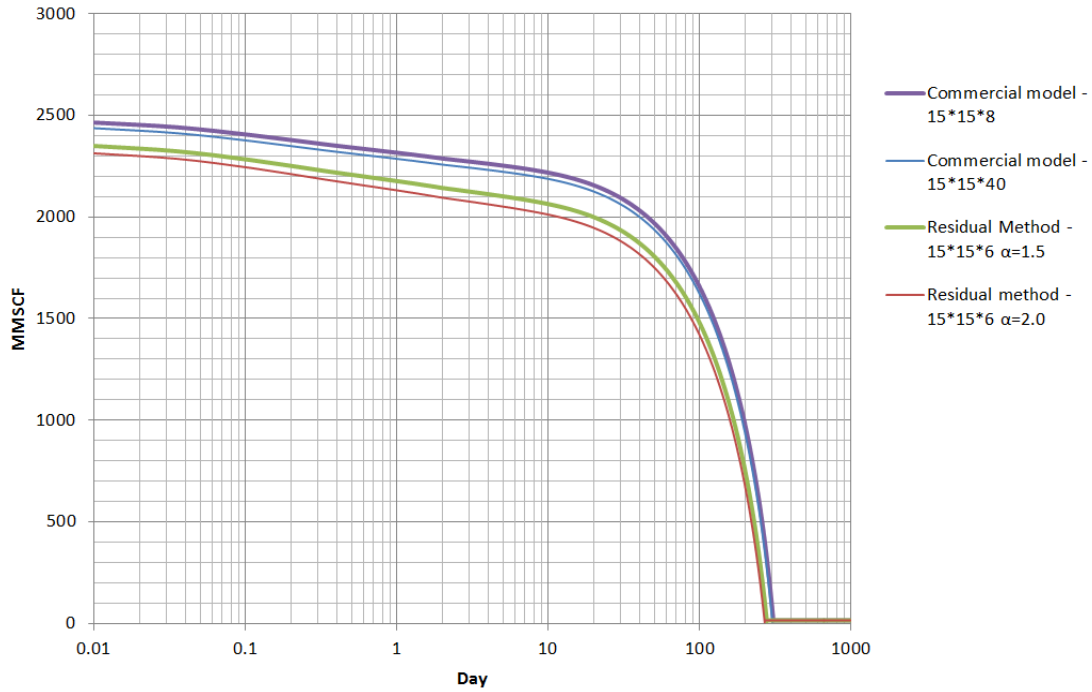


Figure 3.24: Comparison of sandface pressure when flow rate is specified.

It can be discovered in Figure 3.24 that the trends of the sandface pressure responds match the profiles generated by the commercial simulator closely. It can be also found that the higher the degree of refinement, the smaller the differences between the commercial simulator and the method proposed in this thesis.

Chapter 4

PRODUCTION PERFORMANCE OF WELLS WITH COMPLEX ARCHITECTURES

In this chapter, the performances of a well with complex well structure and a well with fish-bone structure are simulated to compare against the performances of a vertical well and a horizontal well in a homogeneous reservoir and a anisotropic reservoir.

4.1 Simulation of Productivity of Complex Well Architectures

First, a homogeneous reservoir with properties shown in Table 4.1 is considered, and four well structures are shown in Figure 4.1 and Figure 4.2.

Table 4.1 Homogeneous reservoir and compressible fluid data used for the comparison runs between different well structures

Reservoir Size; x,y,z	21*50ft, 21*50ft, 11*50ft
Porosity, ϕ	0.1
Permeability; $k_x=k_y=k_z$	3 md
Initial Reservoir Pressure, P_i	3000 psia
$T_{reservoir}$	660˚R
$P_{standard\ condition}$	14.7 psia
$T_{standard\ condition}$	520˚R
Fluid Molecular weight, M_w	16.043
Fluid Critical Temperature, T_{cr}	343.1˚R
Fluid Critical Pressure, P_{cr}	666.4 psi
Sandface Pressure, P_{sf}	1000 psia
Wellbore radius, r_w	0.25 ft

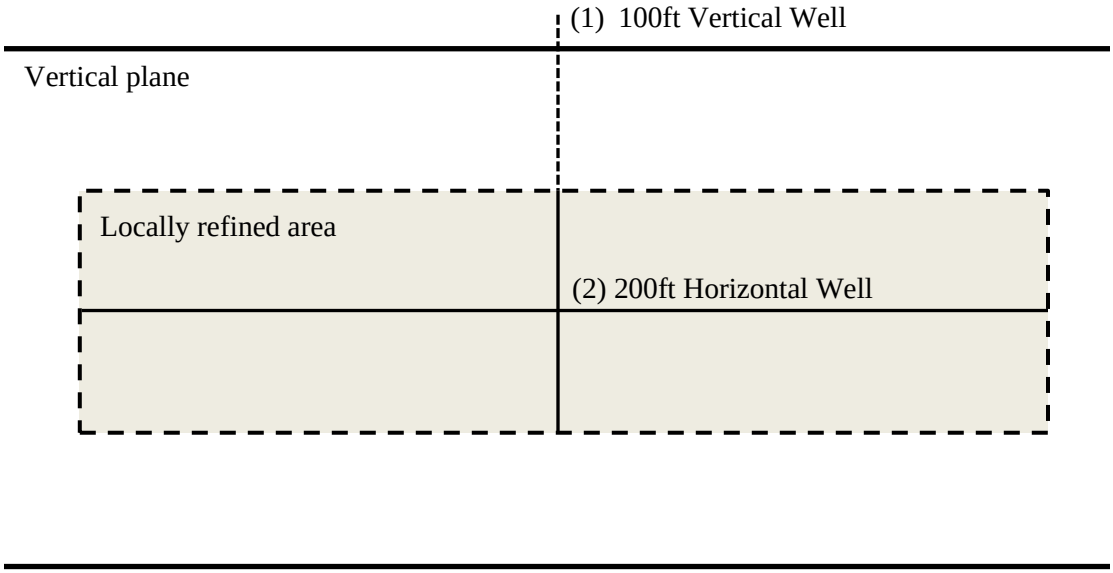


Figure 4.1: Well structures (1), (2) in the reservoir described in Table 4.1.

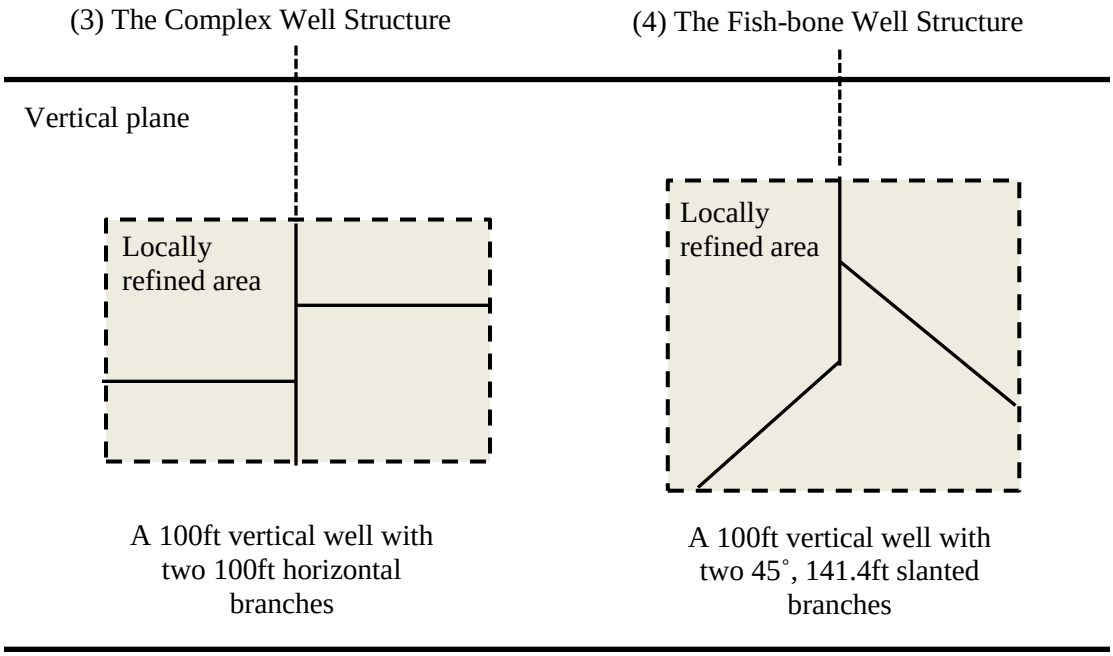


Figure 4.2: Well structures (3), (4) in the reservoir described in Table 4.1.

The simulation results are shown in Figure 4.3 and Figure 4.4. The result shows that the well with a fishbone structure has the highest production rate comparing to other three wells. At day 1, the difference between production rates of the complex well structure and the fish-bone structure is about 19% of the production rate of the fish-bone well. In addition, only about 70% of time is need for a well with the complex structure or about 56% of time for a fish-bone well to produce 50% of OGIP as compared to the horizontal well.

Then, a anisotropic reservoir with properties shown in Table 4.2 is considered to see the change between performances of wells when the permeability in the vertical direction becomes 50% of the permeability in horizontal directions . The results are shown in Figure 4.5 and Figure 4.6.

Table 4.2 Reservoir and fluid properties used in the comparison runs between different well structures.

Reservoir Size; x,y,z	21*50ft, 21*50ft, 11*50ft
Porosity, ϕ	0.1
Permeability; $k_x=k_y$	0.1 md
Permeability; k_z	0.05md
Initial Reservoir Pressure, P_i	3000 psia
$T_{reservoir}$	660˚R
$P_{standard\ condition}$	14.7 psia
$T_{standard\ condition}$	520˚R
Fluid Molecular weight, M_w	16.043
Fluid Critical Temperature, T_{cr}	343.1˚R
Fluid Critical Pressure, P_{cr}	666.4 psi
Sandface Pressure, P_{sf}	1000 psia
Wellbore radius, r_w	0.25 ft

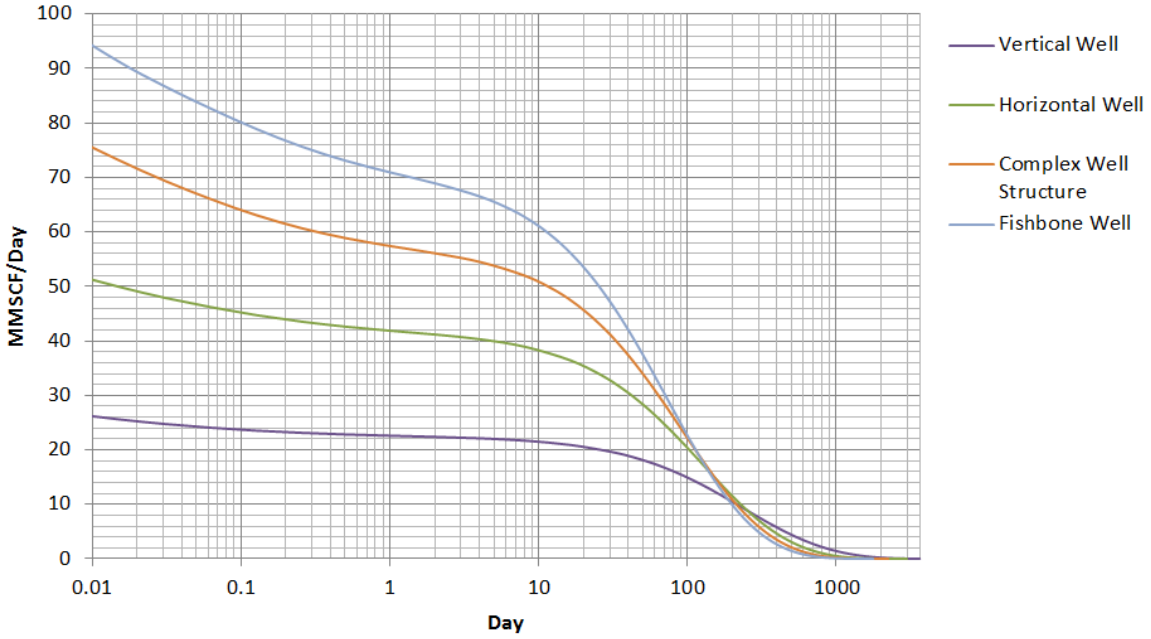


Figure 4.3: Comparison of well performance in a homogeneous reservoir, $k_x=k_y=k_z=3md$.

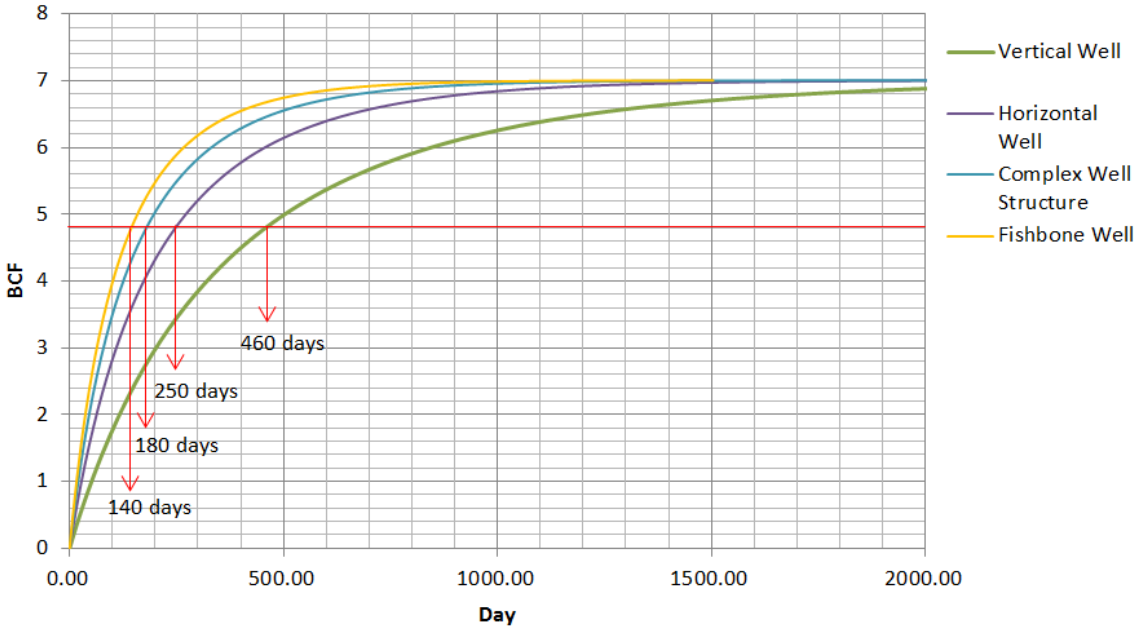


Figure 4.4: Time needed to produce 50% of OGIP.

It is observed that the performance of a fishbone structure is still better than the complex well structure with horizontal branches. At day 1, the production rate of the fish-bone well is about 27% higher than the production rate of the well with the complex structure. The observation shows that the fish-bone well is more effective than other three kinds of well when producing gas from a low permeability reservoir, only the fish-bone well can produce 30% of OGIP within 2000 days.

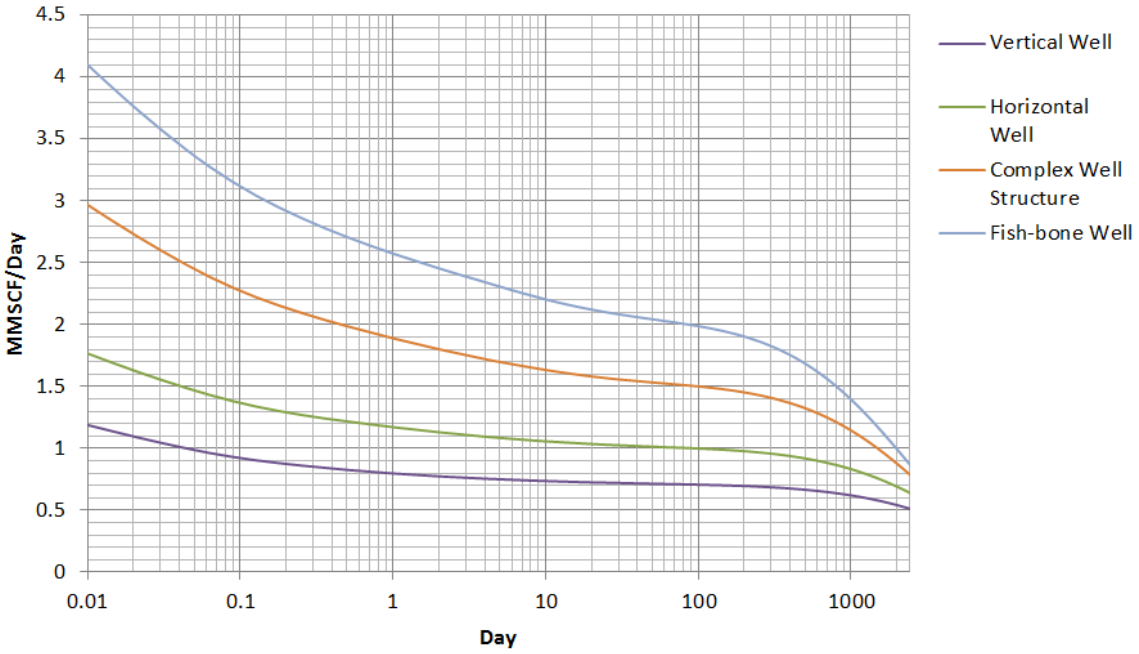


Figure 4.5: Comparison of production performances between wells in a low permeability anisotropic reservoir.

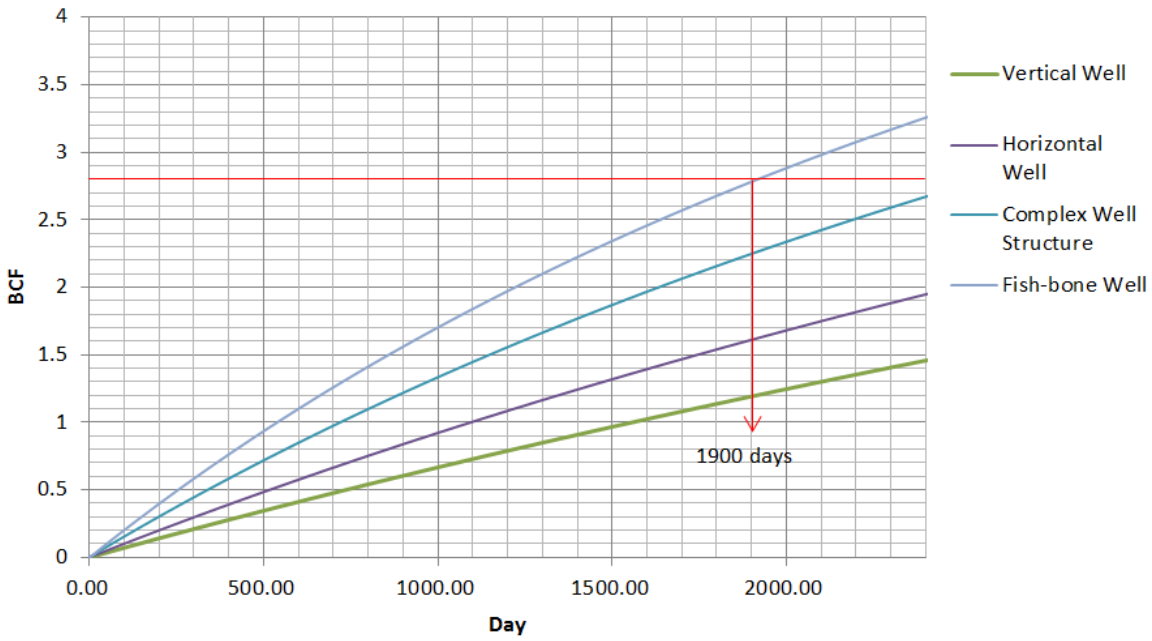


Figure 4.6: Time needed to produce 30% of OGIP.

Chapter 5

SUMMARY, CONCLUSION AND RECOMMENDATIONS

A three dimensional, single-phase compressible fluid flow numerical model which is capable of simulating performances of wells complex structures has been developed in this study. An LGR technique is implemented to approximate well structures with a collection of high-resolution blocks. The LGR technique avoids unnecessary blocks to be introduced to the discretized system, which significantly reduces computational overhead of simulations. A coordinate sparse matrix schemes is used to store the non-symmetric data structure created by implementing the LGR, and a diagonally structured sparse matrix scheme is implemented in coarse block systems in order to save more computer memory. The coefficient matrices created with transmissibility terms of the window and coarse block systems are solved using the robust solver: the pre-conditioned bi-conjugate gradient stabilized method with a main diagonal preconditioner.

The developed computer program simulates the well performance by using the residual method, which is a radically different approach as compared to calculating flow rate using conventional formulations such as Peaceman's wellbore equation. Residual equation is utilized to calculate the fluid flow into the well blocks in locally refined regions directly, which implies that the proposed algorithmic protocol is not restricted by the limitations of Peaceman's equation.

The verifications show that the developed model is capable of providing accurate predictions of production rate when simulating vertical wells and horizontal wells. Besides, when simulating slanted wells performances, the developed model can provide a more accurate and more reasonable prediction as compared to commercial simulator. The accuracy can be further increased by decreasing the expansion factors α and β in window systems. In addition, the model is able to capture the dynamics of pressure profiles surrounding the wells during early time of simulations, while retaining the complete pressure profiles in coarse block systems.

A well with a complex structure and a well with a fish-bone structure are also simulated to compare the performances of different well structures in reservoirs with different properties. The results show that, wells perform better if their structures were constructed with slanted wells in anisotropic reservoirs with relative smaller vertical permeability, k_z.

To sum up, this study verifies that the residual equation can be used to calculate the fluid flow into the specified well structure accurately when the well geometry is approximated by the LGR technique. Also, the modified LGR method is proven to be accurate, which updates the pressure profiles in coarse block systems by the flow rates calculated in window systems by using the residual equation.

In future works, the implementation performance of the developed model can be extended improved in several ways:

(1) Instead of dealing with single-phase model, the approach can be applied to multi-phase flow problems.

(2) The model can be used to simulate performances of hydraulic fractures by approximate the physical structure of the fractures more accurately.

(3) The speed of the developed program can be improved with several methods: incorporating parallel/multi-threads computing algorithms, implementing advanced sparse matrix schemes, using a faster solver or providing an advanced preconditioner to the solver to improve convergence speed.

REFERENCES

1. Joshi, S. (2000). Horizontal and Multi-Lateral Wells : Performance Analysis — An Art or a Science? *October*, *39*(10).
2. Maricic, N., Mohaghegh, S., & Artun, E. (2008). A Parametric Study on the Benefits of Drilling Horizontal and Multilateral Wells in Coalbed Methane Reservoirs. *SPE Reservoir Evaluation & Engineering*, *11*(6), 9-12.
3. Economides, M. J., Brand, C. W. and Frick T. P. Anisotropic (1996), "Well configurations in Reservoirs", SPEFE, December 1996, 257-262.
4. Vo, D. T. And Madden, M. V. (1995), "Performance Evaluation of Examples", Trilateral Wells: Field SPERE, February 1995, 22-28
5. Technology D. (2008), and Tiab,Youcef, B., Sonatrach"Increased Oil Production by Unconventional Wells – Short-Radius, Horizontal, and Multilateral in the Messaoud: Hassi Thick Multilayer Cambrian Formation, Algeria", SPE 114645 presented at the CIPC/SPE Gas Technology Symposium 2008 Joint Conference (June 16-19) Calgary, Alberta, Canada.
6. Kamkom, R. (2007). Modeling Performance Of Horizontal, Undulating, And Multilateral Wells.
7. Goktas, B., & Ertekin, T. (1999). Development of a Local Grid-Refinement Technique for Accurate Representation of Cavity-Completed Wells in Reservoir Simulators.
8. Ertekin, T., Abou-Kassen, J.H. and King, G.R.: Basic Applied Reservoir Simulation, preliminary from of SPE undergraduate textbook in Reservoir Simulation, 1994.
9. Wasserman, M.L.: "Local Grid Refinement for Three Dimensional Simulators', paper SPE 16013 presented at the SPE 9th Reservoir Simulation Symposium held in San Antonio, TX, Feb. 1-4, 1987
10. Biterge, M.B. and Ertekin, T.: "The development and Testing of a Static/Dynamic Local Grid Refinement Technique," paper SPE 19803 presented at SPE 64th Annual Technical Conference and Exhibition held in San Antonio, TX, Oct. 8-11,1989.

11. Kazemi, H., Vestal, C.R. and Shank, G.D.: "An Efficient Multicomponent Numerical Simulator," *Society of Petroleum Engineers journal* (Oct. 1978)355-368.
12. Coats, K.H.: "A Highly Implicit Steamflood Model," *Society of Petroleum Engineers journal* (Oct. 1978)369-383.
13. Bonder, P.L., Hirasaki, G.J. and Tham, M.J.: "Mathematical Simulation of Polymer Flooding in Complex Reservoirs." *Society of Petroleum Engineers journal* (Oct. 1972)369-382.
14. Chaudhari, N.M.: "A Numerical Solution with Second-Order Accuracy for Multi component Compressible Stable Miscible Flow," *Society of Petroleum Engineers journal* (April 1979)84-92.
15. Todd, M.R., Cobb, M. and McCarter, E.D.: "CO_2 Flood Performance Evaluation for the Cornell Unit, Wasson San Andres Field", Paper SPE 10292 presented at the SPE 56th Annual Technical Conference and Exhibition, San Antonio, TX, Oct. 5-7, 1981.
16. Butler, R.M.: Horizontal Wells for the Recovery of oil, Gas and Bitumen, Petroleum Monograph, Petroleum Society of CIM (1994) 2.
17. Furui, K., Zhu, D. and Hill, A.D.: "A Rigorous Formation Damage Skin Factor and Reservoir Inflow Model for Horizontal Well, "SPEPF (August 2003)151.
18. Al-Mohannadi, N., Ozkan, E. and Kazemi, H.:" Pressure-Transient Respinses of Horizontal and Curved Wells in Anticlines and Domes," paper SPE 84378 presented at 2003 SPE Annual Technical Conference and Exhibition, Denver, 5-8 October.
19. Watts, J.W.:"A Conjugate Gradient-Truncated Direct Method for the Iterative Solution of the Reservoir Simulation Pressure Equation," JPT (June 1981)345-353.
20. Wallis, J.R., Kendall, R.P., Little, T.E. and Nolen, J.S.:"Constrained Residual Acceleration of Conjugate Residual Method," paper SPE 13536 presented at the SPE Reservoir Simulation Symposium held in Dallas, TX, Feb. 10-13, 1985.
21. Yousef Saad and Martin H. Schultz. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. SIAM Journal on Scientific and Statistical Computing, 7:856–869, 1986.

22. H.A. van der Vorst (1992), "Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems", *SIAM J. Sci. Stat. Comput.* **13** (2): 631–644
23. Mattax, C.C. and Dalton, R.L.: Reservoir Simulation, SPE Monograph Series, Richardson, TX, 1990.
24. Heinemann, Z.E., Gerken, G. and von Hantelmann, G.: "Using Local Grid Refinement in a Multiple-Application Reservoir Simulation," paper SPE 12255 presented ar the SPE 7th Reservoir Simulation Symposium held in San Francisco, CA, Nov. 15-18, 1983.
25. Forsyth, P.A. and Sammon, P.H.: "Local Mesh Refinement and Modeling of Faults and Pinchouts," SPERE (June 1986) 275-285.
26. Wasserman, M.L.: "Local Grid Refinement for Three-Dimensional Simulators," paper SPE 16013 presented at the SPE 9th Reservoir Simulation Symposium held in San Antonio, TX, Feb. 1-4, 1987.

Appendix A

Solver: The Preconditioned Bi-Conjugate Gradient Stabilized Method

. To solve a linear system $\boldsymbol{Ax} = \boldsymbol{b}$ with a preconditioner $\boldsymbol{K} = \boldsymbol{K}_1\boldsymbol{K}_2 \approx \boldsymbol{A}$, preconditioned bi-conjugate gradient method starts with an initial guess $\boldsymbol{x}_0$ and proceeds as the processes shown below. Preconditioners are usually used to accelerate convergence of iterative methods.

1. $r_0 = b - Ax_0$
2. Create arbitrary vector $\hat{r}_0$ such that $(\hat{r}_0, r_0) \neq 0$. For example, $\hat{r}_0 = r_0$
3. $\rho_0 = \alpha = \omega_0 = 1$
4. $v_0 = p_0 = 0$
5. For i = 1, 2, 3, …
   (1) $\rho_i = (\hat{r}_0, r_{i-1})$
   (2) $\beta = (\rho_i/\rho_{i-1})(\alpha/\omega_{i-1})$
   (3) $p_i = r_{i-1} + \beta(p_{i-1} - \omega_{i-1}v_{i-1})$
   (4) $y = K^{-1}p_i$
   (5) $v_i = Ay$
   (6) $\alpha = \rho_i/(\hat{r}_0, v_i)$
   (7) $s = r_{i-1} - \alpha v_i$
   (8) $z = K^{-1}s$
   (9) $t = Az$
   (10) $\omega_i = (K^{-1}t, K^{-1}s) / (K^{-1}t, K^{-1}t)$
   (11) $x_i = x_{i-1} + \alpha y + \omega_i z$
   (12) If $(x_i - x_{i-1})^2 < 10^{-16}$, it is accurate enough, quit.
   (13) $r_i = s - \omega_i t$

Appendix B

Equations and Correlations Used in the Model

The following equations and approximations are used in the model for calculating unknown properties:

C++ Code for Calculating Z-Factor by Using Dranchuk-Abou-Kassem Correlation

```
void dranchuk (matrix3D* Pr, double Tr, matrix3D* Z)
{
        matrix3D* ro  = new_matrix(Pr->dimen1,Pr->dimen2,Pr->dimen3);
        matrix3D* ron = new_matrix(Pr->dimen1,Pr->dimen2,Pr->dimen3);
        matrix1D* A   = new_matrix(11);

        A->data[0]      = 0.3265;
        A->data[1]      = -1.07;
        A->data[2]      = -0.5339;
        A->data[3]      = 0.01569;
        A->data[4]      = -0.05165;
        A->data[5]      = 0.5475;
        A->data[6]      = -0.7361;
        A->data[7]      = 0.1844;
        A->data[8]      = 0.1056;
        A->data[9]      = 0.6134;
        A->data[10]     = 0.721;

        double c1 =  A->data[0]
                        + A->data[1] / pow(Tr,1)
                        + A->data[2] * pow(Tr,-3)
                        + A->data[3] * pow(Tr,-4)
                        + A->data[4] * pow(Tr,-5);

        double c2 = A->data[5]
                         +A->data[6] / pow(Tr,1)
                         +A->data[7] / pow(Tr,2);

        double c3 = A->data[6] / pow(Tr,1)
                        +A->data[7] / pow(Tr,2);
```

```
        for(int i=0;i<Pr->dimen1;i++)
        {
                for(int j=0;j<Pr->dimen2;j++)
                {
                        for(int k=0;k<Pr->dimen3;k++)
                        {
                                ro->data[i][j][k] = 0.27 * Pr->data[i][j][k] / Tr;
                                double diff = 1.0;
                                while(diff>0.0001)
                                {
                                  double f = 1 - 0.27*Pr->data[i][j][k]/(ro->data[i][j][k]*Tr)
                                                + c1*ro->data[i][j][k]
                                                + c2*pow(ro->data[i][j][k],2)
                                                -  A->data[8]*c3*pow(ro->data[i][j][k],5)
                                                + A->data[9]* (  1+A->data[10]*pow(ro->data[i][j][k],2)  )
                                                * pow(ro->data[i][j][k],2)
                                                * exp(-1*A->data[10]*pow(ro->data[i][j][k],2))/pow(Tr,3);

                                  double fd = 0.27*Pr->data[i][j][k]/ (  pow(ro->data[i][j][k],2)*Tr  )
                                                + c1 + 2*c2*ro->data[i][j][k]
                                                 - 5*A->data[8]*c3*pow(ro->data[i][j][k],4)
                                                +2*A->data[9]*ro->data[i][j][k]
                                                * (  1+A->data[10]*pow(ro->data[i][j][k],2)
                                                - pow(A->data[10],2) * pow(ro->data[i][j][k],4))
                                               * exp(-1*A->data[10]*pow(ro->data[i][j][k],2)) / pow(Tr,3);

                                ron->data[i][j][k] = ro->data[i][j][k] - f/fd;
                                diff = abs(ron->data[i][j][k]-ro->data[i][j][k]);
                                ro->data[i][j][k] = ron->data[i][j][k];
                                }
                                Z->data[i][j][k] = 0.27*Pr->data[i][j][k]/ ( ro->data[i][j][k]*Tr );
                        }
                }
        }
        delete_matrix(&ro);
        delete_matrix(&ron);
        delete_matrix(&A);
}
```

Lee,Gonzalez and Eakin Correlation for Calculating Viscosity

$$\mu = 10^{-4} K \exp\left[X\left(\frac{\rho}{62.4}\right)^{Y}\right]$$

$$X = 3.448 + \frac{986.4}{T} + 0.01009MW$$

$$Y = 2.447 - 0.2224X$$

$$K = \frac{(9.379 + 0.01607MW)T^{1.5}}{(209.2 + 19.26MW + T)}$$

,where

MW = Molecular Weight (lb/lbmol)

T = Reservoir Temperature (R)

μ = Gas Viscosity (cp)

ρ = Gas Density (lb/ft^3)

Density of the Gas, ρ

$$\rho = \frac{pMW}{ZRT} \quad (\text{lb/ft}^3)$$

where

P = Pressure of the Reservoir(psia)

R = Gas Constant =10.731 (psia-ft^3/lbmol-R)

Formation Volume Factor of Gas, B_g

$$B_g = \frac{P_{sc}\, Z\, T}{5.615\, P\, T_{sc}} \quad (\text{RB/SCF})$$

Appendix C

An Example Run and Output Reports

An example simulation for a 30ft horizontal well and its results are presented below to show the form of outputs of the developed program. The properties used in the simulation are shown in Table C.1 and the system is described as shown in Figure C.1. The simulation reports are provided below:

Table C.1 Reservoir and fluid properties used in the simulation example

Reservoir Size and Degree of Refinement; x,y,z	11*10ft, 11*10ft, 3*20ft
Porosity, ϕ	0.1
Permeability; $k_x=k_y=k_z$	3 md
Initial Reservoir Pressure, P_i	3000 psia
$T_{reservoir}$	660˚R
$P_{standard\ condition}$	14.7 psia
$T_{standard\ condition}$	520˚R
Fluid Molecular weight, M_w	16.043
Fluid Critical Temperature, T_{cr}	343.1˚R
Fluid Critical Pressure, P_{cr}	666.4 psi
Sandface Pressure, P_{sf}	1000 psia
Wellbore radius, r_w	0.25 ft

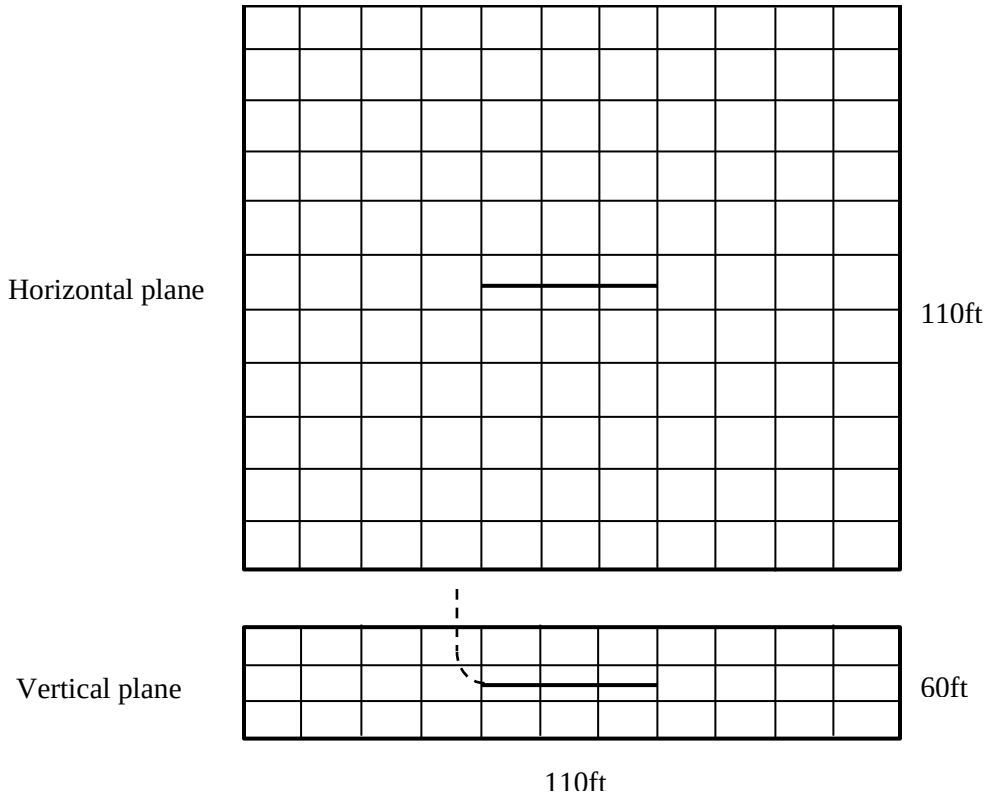


Figure C.1**: Placement of the horizontal well in the reservoir.**

Flow Rate

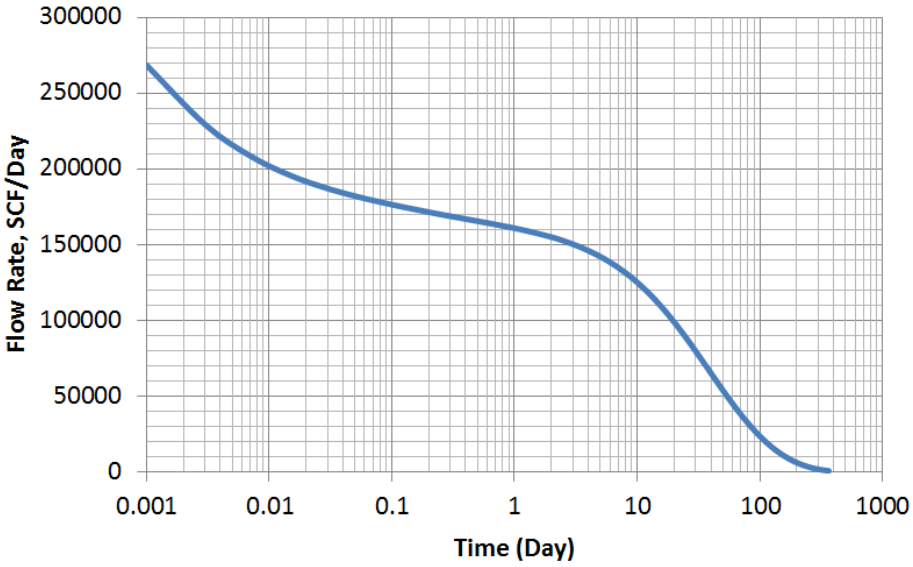


Figure C.2**: Production rate of the example run.**

Cumulative Gas

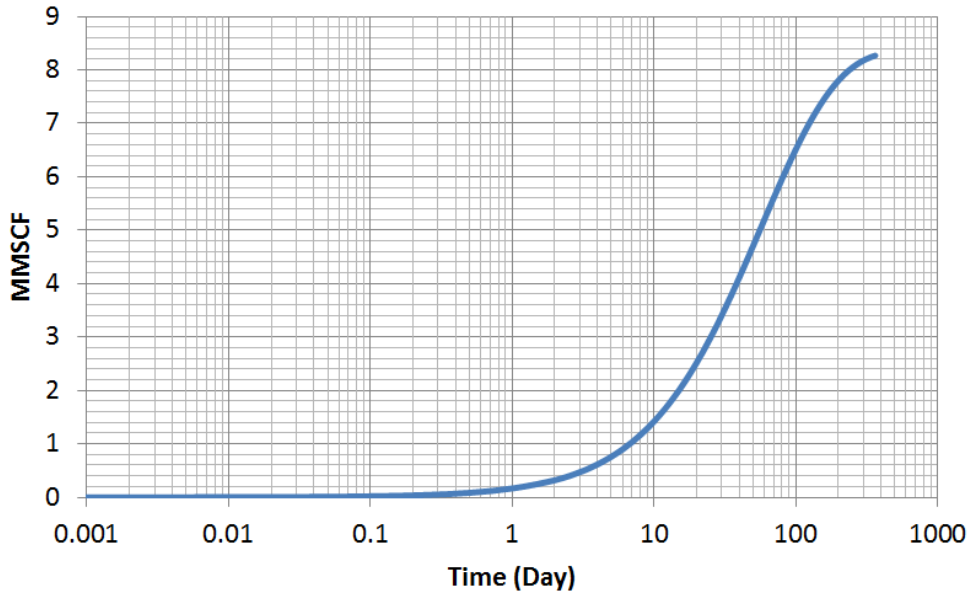


Figure C.3**: Cumulative gas of the example run.**

Material Balance Checks

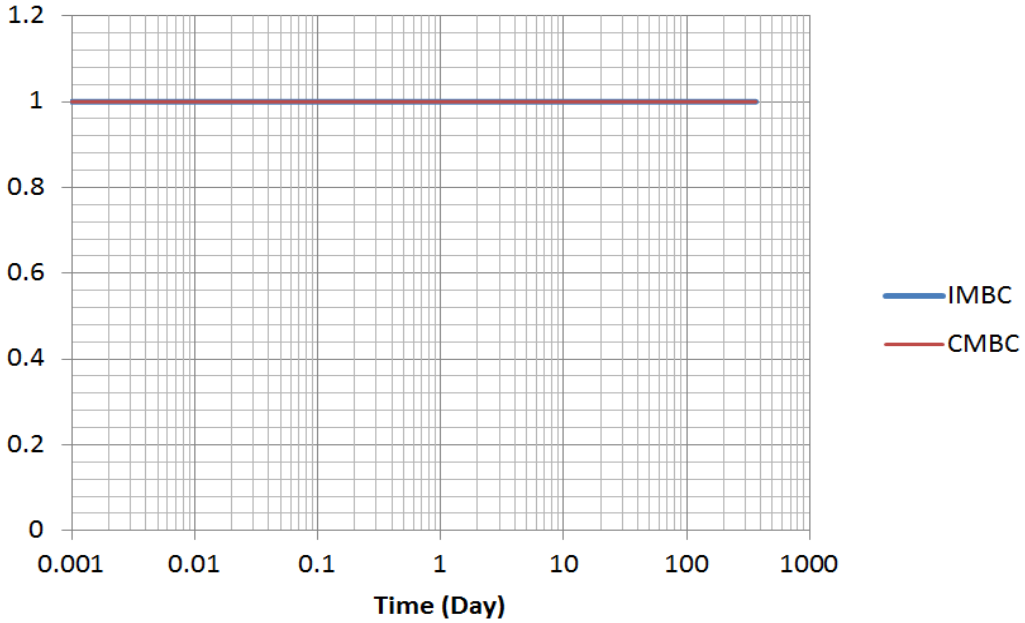


Figure C.4**: Material balance checks of the example run.**

Pressure Profiles of Coarse Block System (Layer 2)

LGR is performed in the blocks with the darker color, unit of pressure is **psia**.

3000	3000	3000	3000	3000	3000	3000	3000	3000	3000	3000
3000	3000	3000	3000	3000	3000	3000	3000	3000	3000	3000
3000	3000	3000	3000	3000	3000	3000	3000	3000	3000	3000
3000	3000	3000	3000	3000	3000	3000	3000	3000	3000	3000
3000	3000	3000	3000	2999.93	2999.88	2999.93	3000	3000	3000	3000
3000	3000	3000	2999.93	2993.55	2988.28	2993.44	2999.93	3000	3000	3000
3000	3000	3000	3000	2999.93	2999.88	2999.93	3000	3000	3000	3000
3000	3000	3000	3000	3000	3000	3000	3000	3000	3000	3000
3000	3000	3000	3000	3000	3000	3000	3000	3000	3000	3000
3000	3000	3000	3000	3000	3000	3000	3000	3000	3000	3000
3000	3000	3000	3000	3000	3000	3000	3000	3000	3000	3000

Table C.2**: Pressure profile of the coarse block system, Time = 0.001 Day.**

2999.9	2999.82	2999.66	2999.41	2999.13	2999.01	2999.13	2999.41	2999.66	2999.82	2999.89
2999.81	2999.65	2999.25	2998.57	2997.74	2997.34	2997.73	2998.56	2999.25	2999.64	2999.81
2999.6	2999.17	2998.06	2995.86	2992.82	2991.28	2992.8	2995.83	2998.04	2999.17	2999.59
2999.22	2998.21	2995.25	2988.5	2977.39	2971.37	2977.28	2988.42	2995.22	2998.19	2999.21
2998.71	2996.7	2990.04	2971.6	2931.76	2908.57	2931.32	2971.38	2989.96	2996.68	2998.7
2998.38	2995.51	2984.61	2945.98	2807.13	2718.39	2805.36	2945.5	2984.47	2995.47	2998.37
2998.71	2996.7	2990.04	2971.6	2931.76	2908.57	2931.32	2971.38	2989.96	2996.68	2998.7
2999.22	2998.21	2995.25	2988.5	2977.39	2971.37	2977.28	2988.42	2995.22	2998.19	2999.21
2999.6	2999.17	2998.06	2995.86	2992.82	2991.28	2992.8	2995.83	2998.04	2999.17	2999.59
2999.81	2999.65	2999.25	2998.57	2997.74	2997.34	2997.73	2998.56	2999.25	2999.64	2999.81
2999.9	2999.82	2999.66	2999.41	2999.13	2999.01	2999.13	2999.41	2999.66	2999.82	2999.89

Table C.3**: Pressure profile of the coarse block system, Time = 0.127 Day.**

2975.76	2973.9	2970.56	2966.55	2963.17	2961.83	2963.14	2966.5	2970.5	2973.84	2975.7
2973.63	2971.08	2966.34	2960.35	2954.99	2952.76	2954.94	2960.28	2966.26	2971.01	2973.56
2969.69	2965.66	2957.74	2946.83	2935.96	2931.17	2935.87	2946.71	2957.63	2965.56	2969.6
2964.76	2958.38	2944.91	2923.98	2899.42	2887.66	2899.21	2923.76	2944.74	2958.24	2964.64
2960.31	2951.12	2929.9	2890.87	2830.74	2798.72	2830.16	2890.45	2929.64	2950.94	2960.17
2958.36	2947.44	2919.98	2857.61	2697.32	2599.79	2695.45	2856.9	2919.63	2947.24	2958.21
2960.31	2951.12	2929.9	2890.87	2830.74	2798.72	2830.16	2890.45	2929.64	2950.94	2960.17
2964.76	2958.38	2944.91	2923.98	2899.42	2887.66	2899.21	2923.76	2944.74	2958.24	2964.64
2969.69	2965.66	2957.74	2946.83	2935.96	2931.17	2935.87	2946.71	2957.63	2965.56	2969.6
2973.63	2971.08	2966.34	2960.35	2954.99	2952.76	2954.94	2960.28	2966.26	2971.01	2973.56
2975.76	2973.9	2970.56	2966.55	2963.17	2961.83	2963.14	2966.5	2970.5	2973.84	2975.7

Table C.4**: Pressure profile of the coarse block system, Time = 1.023 Day.**

2650.48	2648.47	2644.92	2640.73	2637.25	2635.87	2637.2	2640.64	2644.81	2648.35	2650.35
2648.2	2645.58	2640.79	2634.86	2629.64	2627.5	2629.58	2634.75	2640.66	2645.45	2648.06
2644.07	2640.14	2632.55	2622.34	2612.38	2608.03	2612.28	2622.19	2632.39	2639.98	2643.91
2639	2633.02	2620.63	2601.81	2580.16	2569.88	2579.96	2601.57	2620.42	2632.83	2638.82
2634.55	2626.14	2607.1	2572.86	2521.03	2493.59	2520.51	2572.46	2606.81	2625.91	2634.35
2632.65	2622.75	2598.39	2544.37	2407.63	2324.77	2406.03	2543.73	2598.03	2622.5	2632.43
2634.55	2626.14	2607.1	2572.86	2521.03	2493.59	2520.51	2572.46	2606.81	2625.91	2634.35
2639	2633.02	2620.63	2601.81	2580.16	2569.88	2579.96	2601.57	2620.42	2632.83	2638.82
2644.07	2640.14	2632.55	2622.34	2612.38	2608.03	2612.28	2622.19	2632.39	2639.98	2643.91
2648.2	2645.58	2640.79	2634.86	2629.64	2627.5	2629.58	2634.75	2640.66	2645.45	2648.06
2650.48	2648.47	2644.92	2640.73	2637.25	2635.87	2637.2	2640.64	2644.81	2648.35	2650.35

Table C.5**: Pressure profile of the coarse block system, Time = 10.023 Day.**

1847.28	1846.17	1844.21	1841.89	1839.97	1839.21	1839.95	1841.85	1844.15	1846.1	1847.21
1846.02	1844.57	1841.93	1838.65	1835.77	1834.59	1835.74	1838.6	1841.86	1844.5	1845.94
1843.73	1841.56	1837.38	1831.74	1826.25	1823.85	1826.19	1831.66	1837.29	1841.48	1843.65
1840.94	1837.64	1830.8	1820.41	1808.47	1802.82	1808.36	1820.28	1830.68	1837.53	1840.84
1838.48	1833.84	1823.33	1804.44	1775.86	1760.8	1775.58	1804.22	1823.17	1833.71	1838.37
1837.43	1831.96	1818.52	1788.72	1713.35	1668.09	1712.48	1788.37	1818.33	1831.83	1837.31
1838.48	1833.84	1823.33	1804.44	1775.86	1760.8	1775.58	1804.22	1823.17	1833.71	1838.37
1840.94	1837.64	1830.8	1820.41	1808.47	1802.82	1808.36	1820.28	1830.68	1837.53	1840.84
1843.73	1841.56	1837.38	1831.74	1826.25	1823.85	1826.19	1831.66	1837.29	1841.48	1843.65
1846.02	1844.57	1841.93	1838.65	1835.77	1834.59	1835.74	1838.6	1841.86	1844.5	1845.94
1847.28	1846.17	1844.21	1841.89	1839.97	1839.21	1839.95	1841.85	1844.15	1846.1	1847.21

Table C.6**: Pressure profile of the coarse block system , Time = 50.023 Day.**

1427.08	1426.48	1425.41	1424.16	1423.12	1422.71	1423.11	1424.13	1425.38	1426.44	1427.04
1426.39	1425.61	1424.18	1422.4	1420.85	1420.21	1420.83	1422.37	1424.14	1425.57	1426.35
1425.16	1423.98	1421.71	1418.67	1415.7	1414.41	1415.67	1418.62	1421.67	1423.93	1425.11
1423.64	1421.85	1418.15	1412.54	1406.11	1403.06	1406.05	1412.48	1418.09	1421.8	1423.59
1422.31	1419.8	1414.12	1403.93	1388.56	1380.49	1388.41	1403.81	1414.03	1419.73	1422.25
1421.74	1418.78	1411.52	1395.46	1355.1	1331.17	1354.64	1395.27	1411.42	1418.71	1421.68
1422.31	1419.8	1414.12	1403.93	1388.56	1380.49	1388.41	1403.81	1414.03	1419.73	1422.25
1423.64	1421.85	1418.15	1412.54	1406.11	1403.06	1406.05	1412.48	1418.09	1421.8	1423.59
1425.16	1423.98	1421.71	1418.67	1415.7	1414.41	1415.67	1418.62	1421.67	1423.93	1425.11
1426.39	1425.61	1424.18	1422.4	1420.85	1420.21	1420.83	1422.37	1424.14	1425.57	1426.35
1427.08	1426.48	1425.41	1424.16	1423.12	1422.71	1423.11	1424.13	1425.38	1426.44	1427.04

Table C.7**: Pressure profile of the coarse block system , Time = 100.023 Day.**

1131.38	1131.19	1130.84	1130.43	1130.1	1129.96	1130.09	1130.43	1130.83	1131.17	1131.37
1131.16	1130.9	1130.44	1129.87	1129.36	1129.16	1129.36	1129.86	1130.43	1130.89	1131.15
1130.76	1130.38	1129.64	1128.66	1127.7	1127.28	1127.69	1128.64	1129.63	1130.36	1130.74
1130.27	1129.69	1128.49	1126.68	1124.61	1123.63	1124.59	1126.66	1128.47	1129.67	1130.25
1129.83	1129.02	1127.19	1123.9	1118.98	1116.43	1118.93	1123.87	1127.16	1129	1129.82
1129.65	1128.69	1126.35	1121.18	1108.35	1100.92	1108.21	1121.13	1126.32	1128.67	1129.63
1129.83	1129.02	1127.19	1123.9	1118.98	1116.43	1118.93	1123.87	1127.16	1129	1129.82
1130.27	1129.69	1128.49	1126.68	1124.61	1123.63	1124.59	1126.66	1128.47	1129.67	1130.25
1130.76	1130.38	1129.64	1128.66	1127.7	1127.28	1127.69	1128.64	1129.63	1130.36	1130.74
1131.16	1130.9	1130.44	1129.87	1129.36	1129.16	1129.36	1129.86	1130.43	1130.89	1131.15
1131.38	1131.19	1130.84	1130.43	1130.1	1129.96	1130.09	1130.43	1130.83	1131.17	1131.37

Table C.8**: Pressure profile of the coarse block system , Time = 200.023 Day.**

Pressure Profiles of Window System (Layer 6)

The unit of pressure is **psia**.

2999	2995	2992	2990	2988	2987	2986	2985	2985	2985	2985
2996	2966	2924	2881	2858	2845	2837	2833	2830	2830	2830
2995	2938	2799	2554	2483	2455	2443	2437	2435	2434	2434
2995	2924	2649	1000	1000	1000	1000	1000	1000	1000	1000
2995	2938	2799	2554	2483	2455	2443	2437	2435	2434	2434
2996	2966	2924	2881	2858	2845	2837	2833	2830	2830	2830
2999	2995	2992	2990	2988	2987	2986	2985	2985	2985	2985

Table C.9**: Pressure profile of the window system at coarse block (5,6,2) Time = 0.001 Day.**

2984	2984	2984	2984	2984	2984	2984	2984	2984	2984	2984	2984	2984	2984	2984	2984
2830	2828	2827	2827	2827	2826	2826	2826	2826	2826	2826	2826	2827	2827	2828	2829
2435	2433	2432	2431	2431	2431	2431	2431	2431	2431	2431	2431	2431	2432	2432	2433
1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000
2435	2433	2432	2431	2431	2431	2431	2431	2431	2431	2431	2431	2431	2432	2432	2433
2830	2828	2827	2827	2827	2826	2826	2826	2826	2826	2826	2826	2827	2827	2828	2829
2984	2984	2984	2984	2984	2984	2984	2984	2984	2984	2984	2984	2984	2984	2984	2984

Table C.10**: Pressure profile of the window system at coarse block (6,6,2) Time = 0.001 Day.**

2985	2985	2985	2985	2985	2986	2987	2988	2990	2992	2995	2999
2830	2829	2829	2830	2833	2837	2845	2858	2881	2924	2966	2996
2435	2434	2434	2435	2437	2443	2455	2483	2554	2799	2938	2995
1000	1000	1000	1000	1000	1000	1000	1000	1000	2649	2924	2994
2435	2434	2434	2435	2437	2443	2455	2483	2554	2799	2938	2995
2830	2829	2829	2830	2833	2837	2845	2858	2881	2924	2966	2996
2985	2985	2985	2985	2985	2986	2987	2988	2990	2992	2995	2999

Table C.11**: Pressure profile of the window system at coarse block (7,6,2)**

Time = 0.001 Day.

2858	2796	2768	2750	2737	2725	2716	2708	2702	2698	2696
2828	2690	2595	2519	2471	2436	2411	2394	2382	2376	2375
2820	2641	2446	2186	2091	2043	2014	1997	1986	1981	1980
2819	2625	2313	1000	1000	1000	1000	1000	1000	1000	1000
2820	2641	2446	2186	2091	2043	2014	1997	1986	1981	1980
2828	2690	2595	2519	2471	2436	2411	2394	2382	2376	2375
2858	2796	2768	2750	2737	2725	2716	2708	2702	2698	2696

Table C.12**: Pressure profile of the window system at coarse block (5,6,2)**

Time = 0.127 Day.

2667	2657	2648	2641	2635	2630	2627	2624	2623	2622	2623	2625	2628	2632	2637	2643	2651	2660
2356	2338	2325	2314	2307	2301	2296	2293	2292	2292	2292	2294	2298	2303	2310	2318	2330	2345
1969	1952	1940	1932	1925	1920	1917	1914	1913	1913	1914	1915	1918	1922	1928	1935	1945	1959
1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000
1969	1952	1940	1932	1925	1920	1917	1914	1913	1913	1914	1915	1918	1922	1928	1935	1945	1959
2356	2338	2325	2314	2307	2301	2296	2293	2292	2292	2292	2294	2298	2303	2310	2318	2330	2345
2667	2657	2648	2641	2635	2630	2627	2624	2623	2622	2623	2625	2628	2632	2637	2643	2651	2660

Table C.13**: Pressure profile of the window system at coarse block (6,6,2)**

Time = 0.127 Day.

2693.35	2693.55	2695.71	2699.86	2705.98	2714.08	2724.10	2735.92	2749.29	2767.39	2795.23	2856.12
2374.39	2371.73	2373.68	2380.24	2391.87	2409.52	2434.79	2470.00	2517.84	2593.57	2688.93	2824.53
1980.59	1977.51	1978.92	1984.68	1995.47	2013.06	2041.51	2090.50	2185.29	2444.68	2640.69	2816.69
1000.00	1000.00	1000.00	1000.00	1000.00	1000.00	1000.00	1000.00	1000.00	2312.53	2624.16	2815.21
1980.59	1977.51	1978.92	1984.68	1995.47	2013.06	2041.51	2090.50	2185.29	2444.68	2640.69	2816.69
2374.39	2371.73	2373.68	2380.24	2391.87	2409.52	2434.79	2470.00	2517.84	2593.57	2688.93	2824.53
2693.35	2693.55	2695.71	2699.86	2705.98	2714.08	2724.10	2735.92	2749.29	2767.39	2795.23	2856.12

Table C.14**: Pressure profile of the window system at coarse block (7,6,2)**

Time = 0.127 Day.

2769	2706	2678	2660	2646	2634	2624	2616	2609	2605	2602
2739	2604	2512	2439	2392	2358	2334	2316	2305	2298	2296
2732	2558	2369	2121	2030	1983	1955	1938	1927	1922	1920
2730	2542	2243	1000	1000	1000	1000	1000	1000	1000	1000
2732	2558	2369	2121	2030	1983	1955	1938	1927	1922	1920
2739	2604	2512	2439	2392	2358	2334	2316	2305	2298	2296
2769	2706	2678	2660	2646	2634	2624	2616	2609	2605	2602

Table C.15**: Pressure profile of the window system at coarse block (5,6,2)**

Time = 1.023 Day.

2571	2561	2553	2546	2540	2536	2532	2530	2528	2528	2529	2531	2533	2537	2542	2548	2556	2564
2275	2258	2245	2236	2228	2223	2218	2216	2214	2214	2215	2217	2220	2225	2231	2239	2250	2264
1907	1892	1881	1872	1866	1862	1858	1856	1855	1855	1855	1857	1859	1863	1868	1875	1885	1898
1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000
1907	1892	1881	1872	1866	1862	1858	1856	1855	1855	1855	1857	1859	1863	1868	1875	1885	1898
2275	2258	2245	2236	2228	2223	2218	2216	2214	2214	2215	2217	2220	2225	2231	2239	2250	2264
2571	2561	2553	2546	2540	2536	2532	2530	2528	2528	2529	2531	2533	2537	2542	2548	2556	2564

Table C.16**: Pressure profile of the window system at coarse block (6,6,2)**

Time = 1.023 Day.

2599	2598	2599	2602	2607	2614	2622	2633	2645	2658	2677	2705	2766
2301	2294	2293	2295	2302	2314	2332	2357	2391	2437	2510	2603	2735
1927	1920	1917	1919	1926	1936	1954	1981	2028	2119	2368	2557	2728
1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	2242	2541	2726
1927	1920	1917	1919	1926	1936	1954	1981	2028	2119	2368	2557	2728
2301	2294	2293	2295	2302	2314	2332	2357	2391	2437	2510	2603	2735
2599	2598	2599	2602	2607	2614	2622	2633	2645	2658	2677	2705	2766

Table C.17**: Pressure profile of the window system at coarse block (7,6,2) Time = 1.023 Day.**

2469	2415	2392	2376	2365	2355	2346	2339	2333	2329	2327
2443	2329	2250	2188	2149	2120	2099	2084	2075	2069	2067
2437	2289	2129	1917	1840	1800	1777	1763	1754	1750	1748
2436	2276	2020	1000	1000	1000	1000	1000	1000	1000	1000
2437	2289	2129	1917	1840	1800	1777	1763	1754	1750	1748
2443	2329	2250	2188	2149	2120	2099	2084	2075	2069	2067
2469	2415	2392	2376	2365	2355	2346	2339	2333	2329	2327

Table C.18**: Pressure profile of the window system at coarse block (5,6,2) Time = 10.023 Day.**

2301	2292	2286	2280	2275	2271	2268	2266	2265	2265	2265	2267	2269	2272	2277	2282	2288	2295
2049	2036	2025	2017	2010	2006	2002	2000	1998	1998	1999	2001	2003	2007	2013	2020	2029	2041
1738	1725	1716	1709	1704	1700	1697	1695	1694	1694	1694	1696	1698	1701	1705	1711	1719	1730
1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000
1738	1725	1716	1709	1704	1700	1697	1695	1694	1694	1694	1696	1698	1701	1705	1711	1719	1730
2049	2036	2025	2017	2010	2006	2002	2000	1998	1998	1999	2001	2003	2007	2013	2020	2029	2041
2301	2292	2286	2280	2275	2271	2268	2266	2265	2265	2265	2267	2269	2272	2277	2282	2288	2295

Table C.19**: Pressure profile of the window system at coarse block (6,6,2) Time = 10.023 Day.**

2324	2324	2325	2327	2331	2337	2344	2353	2363	2375	2390	2414	2466
2071	2066	2064	2067	2073	2083	2097	2118	2148	2187	2249	2328	2440
1754	1748	1746	1748	1753	1762	1776	1799	1839	1916	2128	2288	2434
1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	2019	2275	2433
1754	1748	1746	1748	1753	1762	1776	1799	1839	1916	2128	2288	2434
2071	2066	2064	2067	2073	2083	2097	2118	2148	2187	2249	2328	2440
2324	2324	2325	2327	2331	2337	2344	2353	2363	2375	2390	2414	2466

Table C.20**: Pressure profile of the window system at coarse block (7,6,2) Time = 10.023 Day.**

1747	1717	1704	1696	1689	1684	1679	1675	1672	1670	1669	1668
1733	1669	1626	1591	1570	1554	1543	1535	1530	1527	1526	1527
1729	1647	1558	1442	1402	1381	1369	1362	1357	1355	1354	1356
1729	1639	1497	1000	1000	1000	1000	1000	1000	1000	1000	1000
1729	1647	1558	1442	1402	1381	1369	1362	1357	1355	1354	1356
1733	1669	1626	1591	1570	1554	1543	1535	1530	1527	1526	1527
1747	1717	1704	1696	1689	1684	1679	1675	1672	1670	1669	1668

Table C.21**: Pressure profile of the window system at coarse block (5,6,2) Time = 50.023 Day.**

1654	1650	1646	1643	1640	1638	1636	1635	1635	1635	1635	1636	1637	1639	1641	1644	1647	1651
1517	1509	1503	1499	1496	1493	1491	1490	1489	1489	1489	1490	1492	1494	1497	1501	1505	1512
1349	1343	1338	1334	1332	1330	1328	1327	1327	1327	1327	1328	1329	1330	1333	1336	1340	1345
1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000
1349	1343	1338	1334	1332	1330	1328	1327	1327	1327	1327	1328	1329	1330	1333	1336	1340	1345
1517	1509	1503	1499	1496	1493	1491	1490	1489	1489	1489	1490	1492	1494	1497	1501	1505	1512
1654	1650	1646	1643	1640	1638	1636	1635	1635	1635	1635	1636	1637	1639	1641	1644	1647	1651

Table C.22**: Pressure profile of the window system at coarse block (6,6,2) Time = 50.023 Day.**

1667	1667	1667	1669	1671	1674	1678	1683	1689	1695	1703	1717	1746
1528	1525	1524	1525	1529	1534	1542	1553	1569	1591	1625	1668	1731
1357	1354	1353	1354	1356	1361	1368	1380	1401	1442	1558	1647	1727
1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1497	1639	1727
1357	1354	1353	1354	1356	1361	1368	1380	1401	1442	1558	1647	1727
1528	1525	1524	1525	1529	1534	1542	1553	1569	1591	1625	1668	1731
1667	1667	1667	1669	1671	1674	1678	1683	1689	1695	1703	1717	1746

Table C.23**: Pressure profile of the window system at coarse block (7,6,2) Time = 50.023 Day.**

1373	1357	1350	1345	1342	1339	1336	1334	1333	1332	1331	1331
1365	1331	1308	1290	1278	1270	1264	1260	1257	1256	1255	1256
1363	1319	1272	1212	1191	1180	1175	1171	1169	1167	1167	1168
1363	1315	1240	1000	1000	1000	1000	1000	1000	1000	1000	1000
1363	1319	1272	1212	1191	1180	1175	1171	1169	1167	1167	1168
1365	1331	1308	1290	1278	1270	1264	1260	1257	1256	1255	1256
1373	1357	1350	1345	1342	1339	1336	1334	1333	1332	1331	1331

Table C.24**: Pressure profile of the window system at coarse block (5,6,2) Time = 100.023 Day.**

1323	1321	1319	1317	1316	1315	1314	1313	1313	1313	1313	1313	1314	1315	1316	1318	1320	1322
1250	1246	1243	1241	1239	1238	1237	1237	1236	1236	1236	1237	1238	1239	1240	1242	1245	1248
1165	1161	1159	1157	1156	1155	1154	1154	1154	1154	1154	1154	1155	1155	1156	1158	1160	1163
1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000
1165	1161	1159	1157	1156	1155	1154	1154	1154	1154	1154	1154	1155	1155	1156	1158	1160	1163
1250	1246	1243	1241	1239	1238	1237	1237	1236	1236	1236	1237	1238	1239	1240	1242	1245	1248
1323	1321	1319	1317	1316	1315	1314	1313	1313	1313	1313	1313	1314	1315	1316	1318	1320	1322

Table C.25**: Pressure profile of the window system at coarse block (6,6,2) Time = 100.023 Day.**

1330	1330	1330	1331	1332	1334	1336	1338	1341	1345	1349	1356	1372
1256	1255	1254	1255	1257	1259	1264	1270	1278	1289	1307	1331	1364
1169	1167	1167	1167	1168	1171	1174	1180	1191	1211	1272	1319	1362
1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1240	1315	1362
1169	1167	1167	1167	1168	1171	1174	1180	1191	1211	1272	1319	1362
1256	1255	1254	1255	1257	1259	1264	1270	1278	1289	1307	1331	1364
1330	1330	1330	1331	1332	1334	1336	1338	1341	1345	1349	1356	1372

Table C.26**: Pressure profile of the window system at coarse block (7,6,2)**
Time = 100.023 Day.

1114	1108	1106	1105	1104	1103	1102	1101	1101	1100	1100	1100
1111	1100	1092	1087	1083	1080	1079	1077	1076	1076	1076	1076
1111	1096	1081	1062	1056	1052	1051	1050	1049	1049	1048	1049
1110	1095	1071	1000	1000	1000	1000	1000	1000	1000	1000	1000
1111	1096	1081	1062	1056	1052	1051	1050	1049	1049	1048	1049
1111	1100	1092	1087	1083	1080	1079	1077	1076	1076	1076	1076
1114	1108	1106	1105	1104	1103	1102	1101	1101	1100	1100	1100

Table C.27**: Pressure profile of the window system at coarse block (5,6,2)**
Time = 200.023 Day.

1098	1097	1096	1096	1095	1095	1094	1094	1094	1094	1094	1094	1095	1095	1095	1096	1096	1097
1074	1073	1072	1071	1071	1070	1070	1070	1070	1070	1070	1070	1070	1070	1071	1072	1072	1073
1048	1047	1046	1045	1045	1045	1045	1044	1044	1044	1044	1044	1045	1045	1045	1046	1046	1047
1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000
1048	1047	1046	1045	1045	1045	1045	1044	1044	1044	1044	1044	1045	1045	1045	1046	1046	1047
1074	1073	1072	1071	1071	1070	1070	1070	1070	1070	1070	1070	1070	1070	1071	1072	1072	1073
1098	1097	1096	1096	1095	1095	1094	1094	1094	1094	1094	1094	1095	1095	1095	1096	1096	1097

Table C.28**: Pressure profile of the window system at coarse block (6,6,2)**
Time = 200.023 Day.

1100	1100	1100	1100	1100	1101	1102	1102	1103	1105	1106	1108	1113
1076	1076	1075	1076	1076	1077	1078	1080	1083	1086	1092	1100	1111
1049	1048	1048	1048	1049	1049	1051	1052	1056	1062	1081	1096	1110
1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1071	1095	1110
1049	1048	1048	1048	1049	1049	1051	1052	1056	1062	1081	1096	1110
1076	1076	1075	1076	1076	1077	1078	1080	1083	1086	1092	1100	1111
1100	1100	1100	1100	1100	1101	1102	1102	1103	1105	1106	1108	1113

Table C.29**: Pressure profile of the window system at coarse block (7,6,2) Time = 200.023 Day.**