An Unstructured Finite Volume Simulator for Multiphase Flow through Fractured-Porous Media

by

Reena Bajaj

Submitted to the School of Engineering
in partial fulfillment of the requirements for the degree of

Master of Science in Computation for Design and Optimization

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2009

Author .....
School of Engineering
July 30th, 2009

Certified by .....
Ruben Juanes
Assistant Professor, Civil and Environmental Engineering
Thesis Supervisor

Accepted by .....
Jaime Peraire
Professor of Aeronautics and Astronautics
Director, Computation for Design and Optimization Program

An Unstructured Finite Volume Simulator for Multiphase Flow through Fractured-Porous Media

by

Reena Bajaj

Submitted to the School of Engineering
on July 30th, 2009, in partial fulfillment of the
requirements for the degree of
Master of Science in Computation for Design and Optimization

Abstract

Modeling of multiphase flow in fractured media plays an integral role in management and performance prediction of oil and gas reserves. Geological characterization and multiphase flow simulations in fractured media are challenging for several reasons, such as uncertainty in fracture location, complexity in fracture geometry, dynamic nature of fractures etc. There is a need for complex simulation models that resolve the flow dynamics along fractures and the interaction with the porous matrix. The unstructured finite volume model provides a tool for the numerical simulation of multiphase flow (immiscible and incompressible two-phase flow) in two-dimensional fractured media. We use a finite volume formulation, which is locally mass conservative and it allows the use of fully unstructured grids to represent the complex geometry of the fracture networks. Fractures are represented as objects of lower dimensionality than that of the domain (in this case, 1D objects in a 2D domain). The model permits fine-scale simulation of multiphase transport through fractured media. The non-Fickian transport resulting due to the presence of heterogeneity (as fractures or inhomogeneous permeability distribution) is captured by the traditional advection-diffusion equation using a highly discretized system. Today, many macroscopic flow models are being developed which account for the non-Fickian, non-local flow more accurately and efficiently with less computation. The finite volume simulator model described in this thesis will be instrumental as a tool to train and validate the macroscopic flow models which account for anomalous transport behavior. We illustrate the performance of this simulator on several synthetic cases with different fracture geometries and conclude the model effectively captures the multiphase fluid flow pattern in fractured media.

Thesis Supervisor: Ruben Juanes
Title: Assistant Professor, Civil and Environmental Engineering

Dedication

To my late Grandfather and Priyank

Acknowledgments

I thank my advisor Prof. Ruben Junes for giving me an opportunity to work with him during my short stint at MIT. I am grateful to him for providing me constant guidance and amiable working environment.

I thank my husband Priyank for his understanding and enormous patience during my stay here. It was impossible to pursue this degree without his encouragement and support. I am indebt to my whole family; my parents, in-laws, grandparents, cousins, brother and sister for their love and co-operation.

I am thankful to all the members of Juanes research group, in particular, I thank Luis Cueto for his useful advice on several occasions. I also thank Peter Kang and Birendra Jha for time to time healthy discussions and help.

Thanks also to all my CDO classmates for being extremely co-operative. Last but not least, I thank all my friends at MIT and outside MIT for their constant motivation and fun times. In no particular order, I thank Pritesh, Kavya, Ayushman, Durga, Iti, Indira, Abhishek, Alex, Sanjeev, Meetu, Utsav and Yang for being a part of this wonderful journey.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Modeling of multiphase flow in fractured media plays an integral role in management and performance prediction of oil and gas reserves. It is also of high importance in radioactive waste management, enhanced oil recovery in naturally fractured reservoir and groundwater contamination by non aqueous phase liquids (NAPL).

Geological characterization and multiphase flow simulations are challenging in fractured media for several reasons such as uncertainty in fracture location, complexity in fracture geometry, dynamic nature of fractures, etc. There is a need for complex simulation models that resolve the flow dynamics along fractures and the interaction with the porous matrix. The difficulties in the numerical modeling of multiphase flow in fractured media stem from the extremely heterogeneous and anisotropic fracture-matrix system. The large contrast in the rock matrix and fracture permeability coupled with small fracture openings makes the numerical simulation challenging. Unlike in single phase, the non-linearity from the relative permeability and capillary pressure in case of multiphase flow also complicates the numerical simulation. In addition to the numerical complexities, the presence of heterogeneity as fracture and inhomogeneous permeability distribution results in anomalous transport, which is captured accurately by the traditional advection diffusion equation on a highly discretized system. Today, many macroscopic flow models are being developed to

account for such non-Fickian, non-local flow accurately with less computational cost. The main motivation to develop the simulator described in this thesis is to permit fine-scale simulation of multiphase transport through fractured media and be instrumental as a tool to train and validate the macroscopic flow models which account for anomalous transport behavior.

1.2 Related Work

The geological and numerical complexity of the fractured reservoir requires the use of simplified models for flow simulation. This is addressed by using the multiphase flow modeling procedures based on dual porosity, dual permeability model (see, *e.g.*, [4, 11, 20]). In this approach, the flow in the fractures is separated from the flow inside the matrix. The reservoir model is represented by two overlapping continua – one continuum to represent the fracture network, where the main flow occurs, and another to represent the matrix, which acts as the source of fracture continuum. The intersection of these two continua is modeled through a transfer function involving the so called shape factor. The shape factor is in general different for each grid block depending on the underlying geology and the type of flow. This model assumes the geostatistical information about the fracture, e.g., porosity, volume and orientation, is quite efficient but faces the difficulty in evaluating the transfer function between the matrix and the fractures accurately, which can result in inaccurate flow predictions.

An alternative to the dual porosity model is the discrete fracture model (see, *e.g.*, [3, 15]). In this model, each fracture is represented explicitly, thus accounting for the effect of individual fractures on fluid flow. It can model high permeability features (open fractures) as well as low permeability features such as sealing faults. The dimensionality of the fracture grid cells is reduced to $(n-1)$ in an n-dimensional domain, e.g., fractures are represented as 1-D objects (line) in a 2-D domain (figure 1-1) and as 2-D objects (polygon) in 3-D domain (see, *e.g.*, [8, 9, 10]). This approach considerably enhances the computational efficiency of the model. There are also other advantages of using the discrete fracture model. Unlike most other methods to model

fracture, there is no need to calculate exchange function between the matrix and the fracture. The discrete fracture model models the matrix-fracture connectivity more realistically by individual treatment of each fracture and taking into consideration the variations in the physical properties such as transmissibility or geometrical properties such as size, elongation and orientation of each fracture. Thus, the discrete fracture model is more flexible and accurate than the dual porosity model to simulate flow through fractured media.

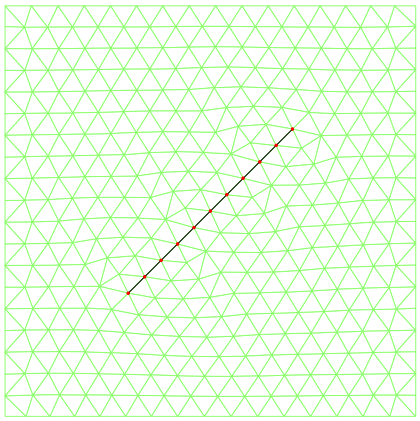


Figure 1-1: 2-D unstructured mesh representing a 1-D fracture as a line element in the discrete fracture model.

The complexity of fractured-porous media can be accurately captured using an unstructured discretization scheme. In the case of unstructured discretization, there are mainly two approaches – finite element and finite volume. Much work has been done using the finite element approach to simulate single-phase flow in fractured media (see, *e.g.*,[3, 8, 12, 15]). In case of multiphase flow in highly heterogeneous and fractured media, finite element methods do not ensure local mass conservation. Finite element formulation based on mixed [6, 13] or discontinuous Galerkin methods [17] ensures mass conservation but is computationally more expensive than the finite volume method. The finite volume formulation ensures both mass conservation and computational efficiency; therefore, it is the preferred approach for fluid flow simulation.

Many previous works have used the finite volume formulation (see, *e.g.*, [7, 10, 12, 14]). Local mass conservation is the main motivation to use finite volume method for discretizing the governing equations to capture subsurface flow through fractured media. The finite volume method is derived from conservation of physical quantities over each cell volume, hence global conservation is also ensured. The existence of sharp fronts and discontinuities, in the multiphase flow through fractured media, due to the high contrast between the fracture and matrix properties also emphasizes on the need for a locally mass conservative method. The methods which are not locally mass conservative do not give the correct location of fronts. For our simulation, we need a numerical scheme that is mass conservative and works well on unstructured grid where the complex fracture geometry can be represented efficiently. Finite volume method fit best in our requirements. Other advantage of finite volume method is that it has the simplicity of finite difference method with the local accuracy of finite element method. Finite volume method is computationally more expensive than the finite difference method and less than finite element method for same accuracy. At the same dimension of discretization, the accuracy of finite volume method is greater than finite difference method and nearly same as finite element method. Finite volume method is implemented by dividing the domain into number of small control volumes, and the grid points where the variables are located is typically defined as being in the center of each control volume. Extra nodes are added at the boundaries (figure 1-2 and figure 1-3).

In our formulation, we use the discrete fracture modeling based on finite volume approach to simulate the multiphase flow in fractured porous media using an unstructured grid. In chapter 2, we derive the mathematical equations which are used for simulating the multiphase flow through the fractured reservoir. In chapter 3, we describe the numerical discretization of the equations developed in the previous chapter. In chapter 4, we discuss the implementation of the numerical formulation laid out in the chapter 3. In chapter 5, we show the results of the simulation on few synthetic cases followed by concluding remarks in chapter 6.

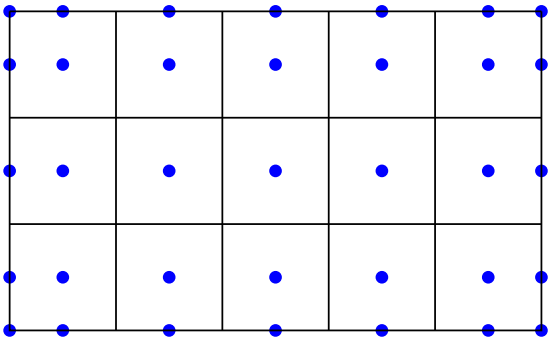


Figure 1-2: In finite volume method, the variables are located at the center of each control volume. The figure shows the location of unknowns in the control volumes at the cell centers and at the boundaries in a structured grid. The variables in the control volumes are located similarly i.e. at the cell center (barycenter of each triangle) in an unstructured grid formed by Delaunay triangles.

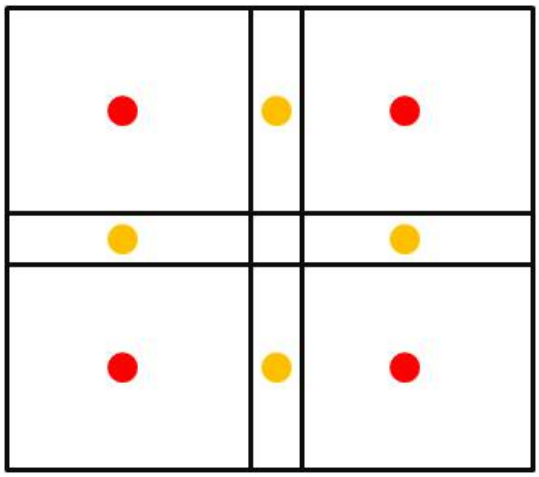


Figure 1-3: The figure shows a square matrix block with two intersecting fractures. The red dots represents the location of variables in the control volume in the matrix cells and the yellow dots represents variables in the control volume in the fracture cells. There is no control volume located at the intersection point between the two fractures, the reason for which is described in the later part of this thesis in detail.

Chapter 2

Mathematical Formulation

Reservoir simulation entails the solution of partial differential equations (PDEs) describing the flow of fluids in the petroleum reservoir. In this chapter, we describe the PDEs which govern fluid flow in the subsurface. The PDEs are based on the fundamental law of mass conservation and Darcy's law.

2.1 Conservation of Mass and Darcy's Law

Consider an arbitrary control volume Ω with the porosity ϕ and outward pointing normal $\mathbf{n}$ at any point on its surface $\partial\Omega$. The total surface area is S. For simplicity we assume that there is a single fluid. The density of the fluid is assumed to be ρ. The mass balance on Ω is as follows

(Rate of inflow - rate of outflow) + source = accumulation

The integral form of mass conservation is given by

$$\int_{\partial\Omega}(-\rho\mathbf{u})\cdot\mathbf{n}dS+\int_{\Omega}\tilde{m}d\Omega=\int_{\Omega}\frac{\partial}{\partial t}(\phi\rho)d\Omega \tag{2.1}$$

where $\mathbf{u}$ is the Darcy velocity and $\tilde{m}$ is the source/sink term ($\tilde{m}>0$ for source term) of units mass per unit volume per unit time. After applying divergence theorem to

the surface integral in equation 2.1 we get the following

$$\int_{\Omega} \left[-\nabla \cdot (\rho \mathbf{u}) + \tilde{m} - \frac{\partial}{\partial t}(\phi \rho) \right] d\Omega = 0 \tag{2.2}$$

Since the above expression is valid for any control volume Ω, we can remove the integral and obtain

$$-\nabla \cdot (\rho \mathbf{u}) + \tilde{m} = \frac{\partial}{\partial t}(\phi \rho) \tag{2.3}$$

Equation 2.3 provides an expression for conservation of mass. From Darcy's law [5], neglecting gravity, we have

$$\mathbf{u} = -\frac{\mathbf{k}}{\mu} \nabla p \tag{2.4}$$

where p is the pressure, $\mathbf{k}$ is the permeability tensor and μ is the fluid viscosity. Inserting equation 2.4 in equation 2.3, we obtain the pressure equation:

$$\nabla \cdot (\frac{\rho}{\mu} \mathbf{k} \nabla p) + \tilde{m} = \frac{\partial}{\partial t}(\phi \rho) \tag{2.5}$$

For an incompressible fluid (ρ = constant) and incompressible rock (ϕ = constant), we get

$$-\nabla \cdot (\frac{\mathbf{k}}{\mu} \nabla p) = \frac{\tilde{m}}{\rho} = \tilde{q} \tag{2.6}$$

where $\tilde{q}$ is the volumetric source term.

2.2 Tracer Transport

We consider a single phase fluid with a tracer dilution. The two components do not react with each other hence there is no change in the fluid property such as density or viscosity by the addition of the tracer. Assuming slightly compressible fluid and neglecting gravity, the equation 2.6 is used to solve for pressure for the tracer transport. To estimate the concentration of the tracer, we use conservation of mass as described below.

Consider an arbitrary finite region of flow bounded by closed surface $\partial\Omega$ and fixed

in space with control volume Ω. The conserved quantity is mass. The total amount of mass (M) in the control volume Ω is given by

$$\int_{\Omega} M d\Omega \tag{2.7}$$

The rate of change of total mass in the control volume is given by the flux ($\mathbf{f}$) through the region boundary, plus the internal source ($\tilde{q}$)

$$\frac{\partial}{\partial t}\int_{\Omega} M d\Omega + \int_{\partial\Omega} \mathbf{f}\cdot\mathbf{n} dS = \int_{\Omega} \tilde{q} d\Omega \tag{2.8}$$

The total flux ($\mathbf{f}$) is the sum of the advective and the diffusive flux. Therefore $\mathbf{f}$ can be expressed as

$$\mathbf{f} = \mathbf{f}_{advective} + \mathbf{f}_{diffusive} \tag{2.9a}$$

$$\mathbf{f}_{advective} = \mathbf{u}c \tag{2.9b}$$

$$\mathbf{f}_{diffusive} = -\mathbf{D}\nabla c \tag{2.9c}$$

where $\mathbf{u}$ is the Darcy's velocity, c is the tracer concentration and $\mathbf{D}$ is a diffusion-dispersion tensor [5]. Applying the divergence theorem to surface integral term, equation 2.8 can be written as

$$\frac{\partial M}{\partial t} + \nabla\cdot\mathbf{f} = \tilde{q} \tag{2.10}$$

In above equation, the mass can be expressed as $M = c\phi$ where c is the concentration of the tracer and ϕ is the pore volume per unit volume of porous media. In our formulation, we neglect the diffusive flux, hence using equation 2.9, equation 2.10 can be expressed as

$$\phi\frac{\partial c}{\partial t} + \nabla\cdot\mathbf{u}c = \tilde{q} \tag{2.11}$$

Equation 2.11 is called the transport equation for the tracer flow, whereas, as mentioned earlier, equation 2.6 is known as the pressure equation for the tracer transport. We observe that the transport equation shows one-way coupling with the pressure equation. We need to solve the pressure equation in order to obtain the velocities

in the domain. These velocities are used in the transport equation to estimate the concentrations over time. However, there is no reverse coupling between the pressure and saturation equation, which implies that the pressure equation can be solved independently.

2.3 Multiphase Flow

We consider a system consisting of two phases, water and oil, denoted by subscripts w and o, respectively. In our formulation, the following assumptions are made

- Isothermal system
- No phase transitions
- Quasi incompressible fluids
- Negligible capillary pressure
- Gravity effects neglected (horizontal flow)

Like the tracer flow described in equation 2.2, the governing equations in the porous media for the two incompressible and immiscible phases are also derived from conservation of mass and a generalized Darcy law of each phase [5]. Applying conservation of mass for each phase and keeping above approximations in mind

$$\phi \frac{\partial S_j}{\partial t} + \nabla \cdot \mathbf{u}_j = \tilde{q}_j \qquad j = o, w \tag{2.12}$$

where ϕ is the porosity (fraction of void space), S_j and $\mathbf{u}_j$ denote the saturation and velocity of phase j respectively, and $\tilde{q}_j$ is the volumetric source term. The two phase fill the pore space, therefore

$$S_w + S_o = 1 \tag{2.13}$$

Darcy's law connect the phase velocity $\mathbf{u}_j$ to the phase pressure p_j as shown below. The effect of gravity has been neglected in the following

$$\mathbf{u}_j = -\lambda_j \nabla p_j \tag{2.14a}$$

$$\lambda_j = \frac{\mathbf{k} k_{rj}}{\mu_j} \qquad j = o, w \tag{2.14b}$$

where $\mathbf{k}$, $k_{rj}$and μ_j are the absolute permeability, relative permeability, and the viscosity of the phase j respectively. The capillary pressure p_c is given by following expression

$$p_c = p_o - p_w \tag{2.15}$$

where p_o and p_w is the pressure due to the oil and water phases respectively. Since we neglect the capillary pressure effects, we assume that $p_o = p_w = p$, hence $\nabla p_w = \nabla p_o = \nabla p$. The Darcy equation 2.14 combined with the conservation of mass yields the pressure equation for multiphase flow.

$$\mathbf{u} = -\lambda \nabla p \tag{2.16a}$$

$$\nabla \cdot \mathbf{u} = \tilde{q} \tag{2.16b}$$

where $\mathbf{u} = \mathbf{u}_o + \mathbf{u}_w$, $\lambda = \lambda_o + \lambda_w$ and $\tilde{q} = q_o + q_w$. Dropping the subscript of S_w, so that $S_w = S$ and using equation 2.13, $S_o = (1 - S)$, the conservation equation of water, henceforth called the saturation equation, can be written as

$$\phi \frac{\partial S}{\partial t} + \nabla \cdot (f_w \mathbf{u}) = q_w \tag{2.17}$$

where $f_w = \frac{\lambda_w}{\lambda}$ is the fractional flow function.

2.3.1 Constitutive Relations

The relative permeability models the reduced permeability of a phase due to the presence of the other phase. It is a function of saturation and varies with time and space. This makes the pressure equation non-linear. We assume the following constitutive

relationships to connect the pressure equation 2.16 and saturation equation 2.17 of the two phase flow.

$$\begin{aligned} k_{rw} &= S_w^2 = S^2 \\ k_{ro} &= S_o^2 = (1-S)^2, \qquad\qquad 0 \leq S_w, S_o \leq 1 \end{aligned} \tag{2.18}$$

It is evident that in case of the multiphase flow, the pressure and saturation equation is tightly coupled with each other by relative permeability and Darcy's velocity respectively.

2.4 Boundary and Initial Conditions

In order to complete the formulation of the mathematical model, it is necessary to specify some boundary and initial conditions. In subsurface reservoir simulation, two types of boundary conditions are usually considered, Dirichlet and Neumann.

Dirichlet Boundary Conditions – It is also known as the first kind of boundary conditions. It specifies the value of the unknown at the boundaries. In reservoir simulation, it represents the specification of pressure at the reservoir boundary or at the production and injection wells.

$$p = \bar{p} \quad \text{on} \quad \Gamma_p \subset \partial\Omega \tag{2.19}$$

Neumann Boundary Conditions – It is also called the second kind of boundary conditions. It specifies the value of the first derivative of the unknown at the boundaries. It represents the specification of the flow rate across the simulation domain.

$$\mathbf{u} \cdot \mathbf{n} = \bar{u} \quad \text{on} \quad \Gamma_u \subset \partial\Omega \tag{2.20}$$

where $\mathbf{u}$ is the Darcy velocity at the boundaries and $\mathbf{n}$ is the outward unit normal to the boundary. Also, $\Gamma_u \cap \Gamma_p = \emptyset$, $\Gamma_u \cup \Gamma_p = \partial\Omega$. In our formulation, we consider the

following boundary conditions to solve the governing equations.

$$p = 1 \quad \text{at the injector} \tag{2.21a}$$

$$p = 0 \quad \text{at the producer} \tag{2.21b}$$

$$\mathbf{u} \cdot \mathbf{n} = 0 \quad \text{everywhere in simulation boundary} \tag{2.21c}$$

where p and $\mathbf{u}$ denotes the oil pressure and velocity in the domain respectively.

Initial Condition – These are the conditions at an initial time ($t = 0$) from which a given set of mathematical equations or physical system evolves. We assume that initially the water saturation is zero everywhere in the domain.

$$S(t = 0) = 0 \tag{2.22}$$

where S denotes the water saturation.

The PDEs developed in this chapter are also valid for flow through the fractures. Using the governing equations, we solve for the pressure and saturation of the multiphase flow in fractured media. We describe the discretization and implementation details in the following chapters.

Chapter 3

Numerical Modeling

In this chapter, we discuss the discretization scheme used for the pressure and saturation equations developed in chapter 2.

3.1 Discretization of Governing Equations

We discretized the pressure equation 2.16 and saturation equation 2.17 using finite volume scheme.

3.1.1 Pressure Equation

To derive the set of finite volume mass balance equation for pressure equation, consider a grid cell Ω_i in the domain denoted by Ω. Consider the volume of the domain to be V and the surface area to be S. Taking the following integral over Ω_i

$$\int_{\Omega_i} (\tilde{q} - \nabla \cdot \mathbf{u}) dV = 0 \tag{3.1}$$

Using divergence theorem, the equation 3.1 transforms into the following

$$\int_{\partial\Omega_i} \mathbf{u} \cdot \mathbf{n} dS = \int_{\Omega_i} \tilde{q} dV \tag{3.2}$$

where $\mathbf{n}$ denotes the outward pointing normal on $\partial\Omega_i$. The Darcy velocity $\mathbf{u}$ can be expressed as $\mathbf{u} = -\lambda\nabla p$, where λ is the total mobility and p is the potential. Therefore, equation 3.2 can be written as

$$\int_{\partial\Omega_i} -\lambda\nabla p \cdot \mathbf{n} dS = \int_{\Omega_i} \tilde{q} dV \tag{3.3}$$

Let i and j be two neighboring control volumes as shown in the figure 3-1 below.

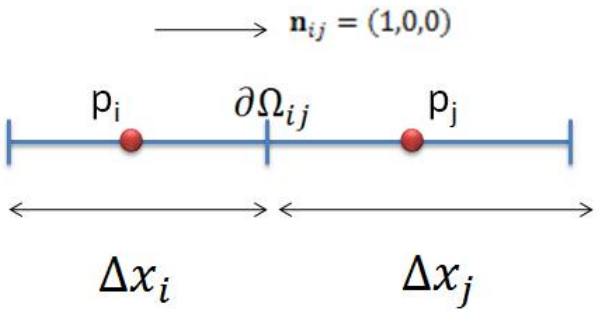


Figure 3-1: Geometrical representation of the two adjacent control volumes in one dimension.

The finite volume method is derived by obtaining the approximation of the potential p with a cell wise constant function and estimating the flux $\mathbf{f}_{ij} = -\int_{\partial\Omega_{ij}} (\lambda\nabla p \cdot \mathbf{n}) dS$ across the interfaces $\partial\Omega_{ij} = \Omega_i \cap \Omega_j$ from a set of neighboring cell pressures. As the name suggests, the two point flux approximation (TPFA) uses two points, the neighboring cell averages p_i and p_j to approximate the flux through the interface between the adjoining cells. We show the derivation of the TPFA for a simple case and then extend it to an unstructured mesh formed by Delaunay triangles.

As shown in figure 3-1, consider two adjacent control volumes in 1-D. Assume that the $\partial\Omega_{ij}$ is the interface between the adjacent cells in the x-coordinate direction such that $\mathbf{n}_{ij} = (1, 0, 0)$. The gradient of potential at the interface, in TPFA method, can be approximated as

$$\delta p_{ij} = \frac{2(p_j - p_i)}{(\Delta x_i + \Delta x_j)} \tag{3.4}$$

where Δx_i and Δx_j denote the respective cell dimensions in the x-coordinate direc-

tion. Thus, the flux $\mathbf{f}_{ij}$ at the interface can be approximated as

$$\mathbf{f}_{ij} = \frac{2(p_j - p_i)}{(\Delta x_i + \Delta x_j)} \int_{\partial\Omega_{ij}} \lambda dS \tag{3.5}$$

To approximate λ on the interface $\partial\Omega_{ij}$, in TPFA we take the distance-weighted harmonic average of the respective cell permeabilities λ_i and λ_j. The permeability field is assumed to be isotropic.

$$\lambda_{ij} = (\Delta x_i + \Delta x_j)(\frac{\Delta x_i}{\lambda_i} + \frac{\Delta x_j}{\lambda_j})^{-1} \tag{3.6}$$

Therefore, the flux through the interface $\partial\Omega_{ij}$ is

$$\mathbf{f}_{ij} = -\lambda_{ij}\delta p_{ij} S = 2S(\frac{\Delta x_i}{\lambda_i} + \frac{\Delta x_j}{\lambda_j})^{-1}(p_i - p_j) \tag{3.7}$$

where S is the area of the interface. Thus, by summing over all the interfaces of a cell, an approximation to the total flux through the cell, $\int_{\partial\Omega_i} \mathbf{u} \cdot \mathbf{n} dS$, and the associated TPFA method is obtained. The mass balance equation 3.2 needs to be fulfilled by each grid cell in the domain. The flux $\mathbf{f}_{ij}$ in equation 3.7 can be expressed as

$$T_{ij} = 2S(\frac{\Delta x_i}{\lambda_i} + \frac{\Delta x_j}{\lambda_j})^{-1} \tag{3.8a}$$

$$\mathbf{f}_{ij} = T_{ij}(p_i - p_j) \tag{3.8b}$$

where T_{ij} is defined as the interface transmissibility. By inserting equation 3.8 in equation 3.3, we get the TPFA scheme for the pressure equation.

$$\Sigma_j T_{ij}(p_i - p_j) = \int_{\Omega_i} \tilde{q} dV \qquad \forall\ \Omega_i \subset \Omega \tag{3.9a}$$

$$\Sigma_j T_{ij}(p_i - p_j) = Q \tag{3.9b}$$

where Q is the volumetric flux. The j denotes the number of interfaces of the cell.

For an unstructured mesh, the TPFA is also derived by material balance in each control volume[9]. In figure 3-2, two adjacent control volumes of an unstructured

mesh is shown. Consider a grid cell i and a grid cell j with the centers of the cell marked as C_i and C_j respectively. The interface $\partial\Omega_{ij}$ lies in between the control volumes i and j with the center C_o. The TPFA for an unstructured mesh at the

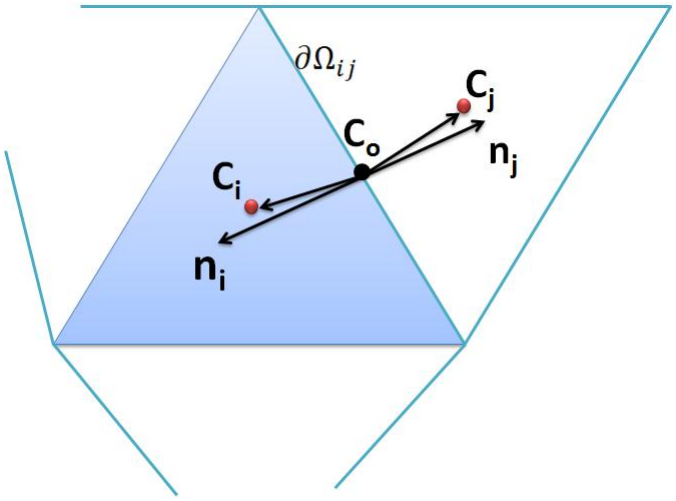


Figure 3-2: Geometrical representation of two adjacent control volumes in an unstructured mesh along with the normal at the interface and distance of the cell centers from the interface. These parameters are used in calculation of transmissivity at the interface between the adjacent cells in TPFA.

interface $\partial\Omega_{ij}$ is

$$Q_{ij} = T_{ij}\lambda\ (p_i - p_j) \tag{3.10}$$

where p_i and p_j are the pressures in cell i and cell j respectively. T_{ij} represents the geometric transmissibility and λ represents the fluid mobility at the interface (same as λ_{ij}). In multiphase flow, each phase has different pressure, flow rate and motilities. However, the geometric part of the transmissibility remains same for each phase. For this reason, separating the geometrical transmissibility from the mobility is computationally efficient and preferable. The geometric part of the transmissibility is given by

$$T_{ij} = \frac{\alpha_i \alpha_j}{\alpha_i + \alpha_j} \quad \text{with} \quad \alpha_i = \frac{AK_i}{D_i}\mathbf{n}_i \cdot \mathbf{d}_i \tag{3.11}$$

where A_i is the area of the interface between the adjacent cells, K_i is the intrinsic permeability of the cell i, D_i is the distance between the cell center and the center of the interface (C_oC_i), $\mathbf{n}_i$ is the unit normal to the interface inside the control volume

i, and $\mathbf{d}_i$ is the unit normal along the line joining the center of the triangle i to the center of the interface (C_oC_i). α_i is evaluated for each cell and the transmissibility is calculated for all the interfaces in the domain using the above equation.

The TPFA method to discretize the pressure equation is valid for both single phase and multiphase flow. The main difference is that for the two phase flow the pressure is a dynamic function of saturation, and hence must be solved repeatedly throughout the simulation. The TPFA calculations can be applied to any geometry of the cells, though the accuracy of the solution highly depends on the geometry of the grid cells. The well shaped and symmetric cells give more accurate results compared to skew and non symmetric cells in the mesh. The system of equations formed by equation 3.10 is clearly symmetric for structured grid. However, in case of unstructured grid, the symmetry is not preserved. The matrix structure is described in later sections.

3.1.2 Saturation Equation

As shown in figure 3-3, consider a control volume Ω_i with the interface $\partial\Omega_{ij}$ and asso-

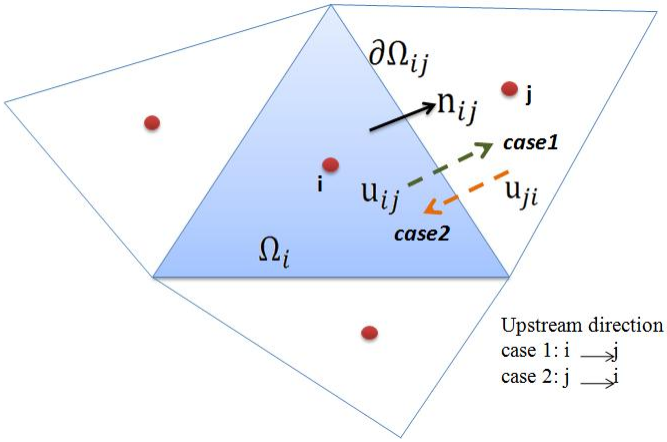


Figure 3-3: The geometrical representation of adjacent control volumes in an unstructured mesh showing the upstream direction for evaluating flux through the interface using single point upwind method.

ciated normal vector $\mathbf{n}_{ij}$. Using the finite volume method in space and a generalized trapezoidal rule (θ-rule) in time, the saturation equation 2.17 for any cell i can be

discretized as

$$\frac{\phi_i}{\Delta t}(S_i^{n+1} - S_i^n) + \frac{1}{|\Omega_i|}\sum_{j \neq i}\left[\theta F_{ij}(S^{n+1}) + (1-\theta)F_{ij}(S^n)\right] = q_i \tag{3.12}$$

where ϕ_i is the porosity, q_i denotes the source term, Δt denotes the time step, S_i^n denotes the cell average of the water saturation at time $t = t^n$ and F_{ij} is a numerical approximation of the flux over the edge Ω_{ij}. We only consider the advective flux, therefore F_{ij} is expressed as

$$F_{ij}(S) = \int_{\partial\Omega_{ij}} f_w(S)_{ij}\mathbf{u}_{ij} \cdot \mathbf{n}_{ij} dA \tag{3.13}$$

where $f_w(S)_{ij}$ denotes the fractional flow of water defined as ratio of mobility of water to the total mobility and $\mathbf{u}$ is the Darcy velocity at the interface. We use a first order upwind scheme to evaluate the value of value of fractional flow at the interface. The flow through the interface can be in either direction as indicated by green and orange arrows in the figure 3-3. In single point upwind scheme, the fractional flow at the interface takes the value in the cell in the upstream direction. Upstream directions for two different flow direction (case 1 and case 2) at the interface is shown in figure 3-3. Mathematically it can be written as

$$f_w(S)_{ij} = \begin{cases} f_w(S)_i & \text{for } \mathbf{u} \cdot \mathbf{n}_{ij} \geq 0 \qquad \text{case 1} \\ f_w(S)_j & \text{for } \mathbf{u} \cdot \mathbf{n}_{ij} \leq 0 \qquad \text{case 2} \end{cases} \tag{3.14}$$

In equation 3.12, if $\theta = 0$, in temporal discretization, we end up using the first order explicit scheme. This scheme has limitations on time stepping due to CFL condition. $\theta = 1$ gives a fully implicit scheme and is more stable. Details of time stepping and implementations are described in later sections.

3.2 Treatment of Fractures

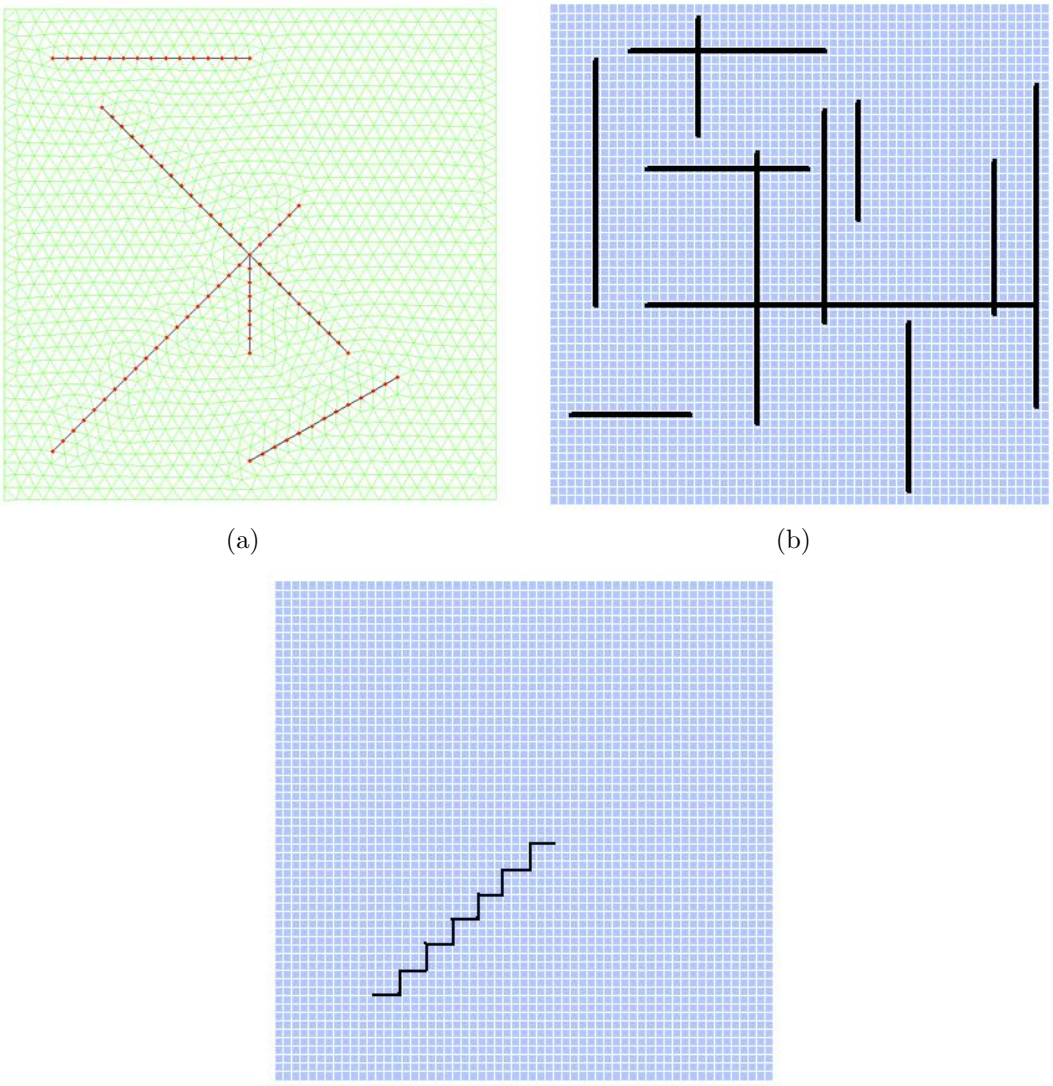
Based on discrete fracture model, the fractures are treated as $(n-1)$ dimensional entities in n dimensional domain. In the finite volume simulator described in the thesis, we consider 2-D domain with fractures represented as 1-D line segments (figure 1-1). Some other assumptions/simplifications applicable to the fracture model in this thesis are listed below.

- The Darcy's law and multiphase equation of fluid flow (equation 2.16 and equation 2.17) valid for the rock matrix is also applicable to the fracture system. We apply these equations in 2-D in matrix system and in 1-D in fracture system.

- The fractures are stochastically generated. They are represented as either horizontal or vertical interconnected fractures on regular grids or inclined discrete or interconnected fractures on an unstructured grid (figure 3-4(a) and 3-4(b)). In order to represent inclined fractures on a structured grid, we need to follow the grid boundaries as shown in the figure 3-4(c). This approach is accurate only on a very fine grid, as a result of which it is computationally very expensive. Also, in this approach, we approximate the location of the fracture by following the grid boundaries, which may lead to high error in a domain with multiple fractures. By using unstructured grid, we can overcome these issues and resolve the fracture geometry more accurately and conveniently on the mesh.

- We assume that the width of the fracture is of very small magnitude ($\leq 10^{-3}$) compared to the its length.

- Fractures are assumed to transmit flow faster than the surrounding matrix. Flow will move quickly through the connected parts of the fracture web. For this reason, we assume that fractures have higher transmissibility ($\geq 10^{5}$) than the surrounding matrix cells (transmissibility of matrix varies based on permeability distribution). Sometimes, the fracture may also act as a barrier due to filling or deposition inside it. In such cases, the transmissibility of the fracture significantly reduces ($\leq 10^{-5}$) and the flow detour around the fracture.



(a) (b)

(c)

Figure 3-4: (a) Fracture representation on an unstructured Mesh (b) Orthogonal fracture representation on a structured mesh (c) Inclined fracture representation on a structured mesh.

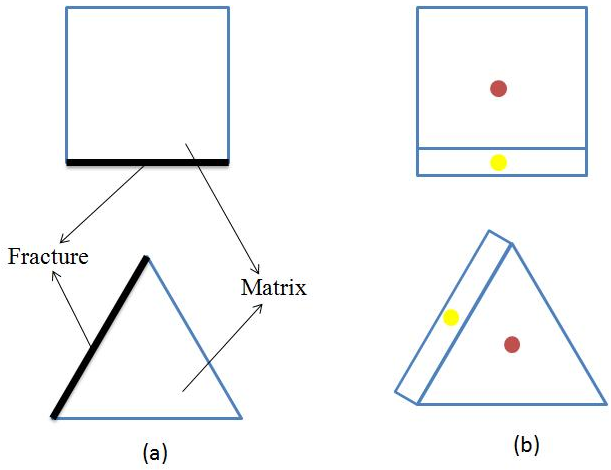


Figure 3-5: (a) Geometrical representation of matrix and fracture domain (b) Computational representation of the matrix and fracture domain. The red dot indicates the matrix control volume and yellow dot indicates the fracture control volume in the computational domain.The conservation of mass is applied in all the fracture and matrix control volume to solve for pressure and saturation at different times.

- The fractures are assumed to have thin rectangular control volume unlike the matrix control volume which is either square in case of a structured grid or triangular in the case of an unstructured grid (figure 3-5).

- In presence of a fracture, the flow is transmitted via fractures between the matrix blocks, i.e. the matrix blocks are connected via fracture (if present) with each other.

- For a single fracture, the transmissibility remains constant throughout the fracture, though it varies among different fractures in the domain.

- The transmissibility for the matrix fracture intersection is assumed to be the harmonic mean of the fracture-fracture transmissibility and matrix-matrix transmissibility.

- We avoid the control volume at the fracture intersection by using the star-delta transformation (figure 3-6) as described by *Karimi-Fard et al.* [9]. The intersection control volumes are very small compared to the adjacent control volumes. Hence, it causes numerical complexities including the limitation on

time stepping in the transport equation. The equivalent transmissibility for n connecting fractures, can be generalized as

$$T_{ij} = \frac{\alpha_i \alpha_j}{\Sigma_{k=1}^{n} \alpha_k} \tag{3.15}$$

where α_i is defined in equation 3.11.

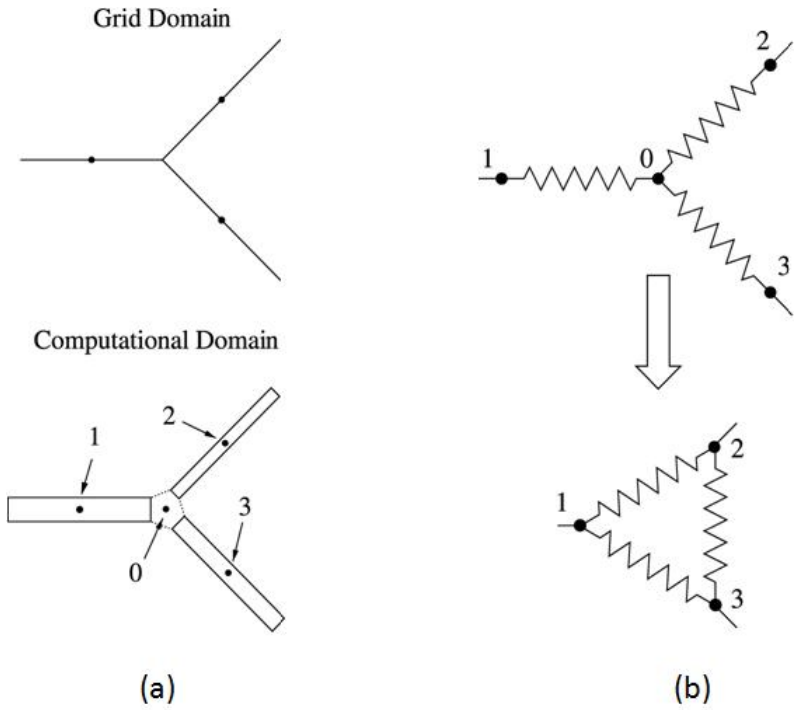


Figure 3-6: (a) Fracture intersection in the grid and fracture domain (b) Star-Delta transformation to approximate the equivalent transmissibility for intersecting fracture network[9].The analogy between flow through the porous media and conductance through a network resistor is applied to calculate the equivalent transmissibility of a heterogeneous porous media.

3.3 IMPES Method

IMPES stands for Implicit Pressure Explicit Saturation and is a standard method for solving the coupled two phase flow equation by separating the calculation of pressure and saturation. The pressure and saturation equations are solved using implicit and explicit time approximation approaches respectively. The main advantage of this method is that it is simple to set, efficient to implement and computationally less expensive compared to other schemes like fully implicit method or sequential method. The disadvantage of implicit method is that it imposes stability restriction in the time steps use to advance the transport. The limitation on the time step due to the CFL condition can be approximated as

$$\Delta t = C\frac{\Delta x}{v_{max}} \tag{3.16}$$

where Δx is the smallest grid cell size, v_{max} is the maximum velocity in the domain, and C is a constant whose value depends on the properties of the medium like porosity etc.

For solving the pressure implicitly, we use the TPFA method described in section 3.1. Equation 3.10 results in system of equation of the form

$$AP = Q \tag{3.17}$$

where A is the matrix of transmissibilities of dimension $N \times N$; N is the sum of the number of degree of freedom of the matrix and the fracture control volumes. P is the vector of matrix and fracture pressure, which we need to solve for. It has a dimension $N \times 1$. Q is the vector of source terms, with dimension $N \times 1$, for each control volume in the mesh. The schematic representation of the system is shown in the figure 3-7 below.

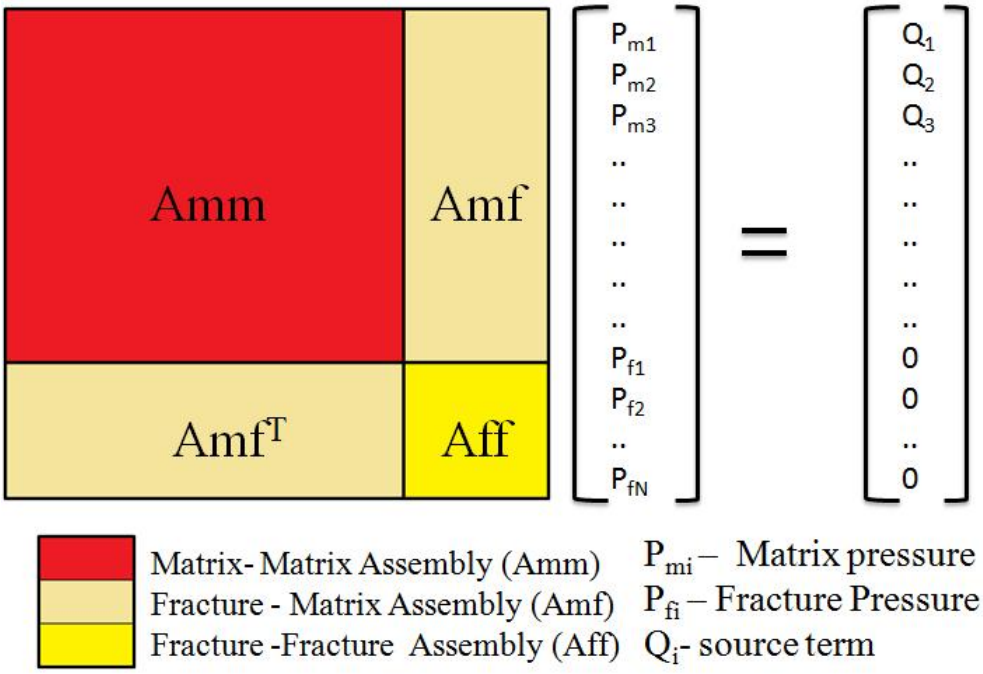
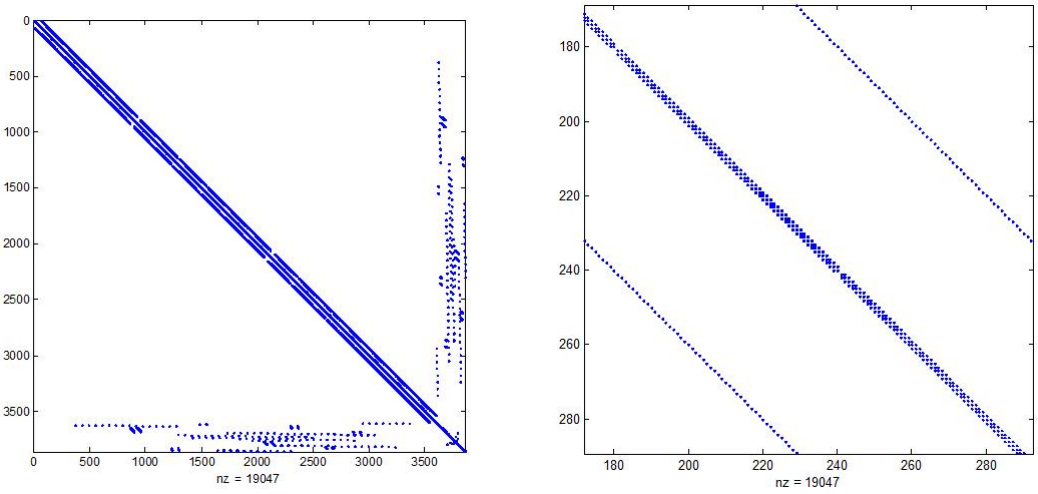


Figure 3-7: Matrix structure formed for solving the matrix and fracture pressures implicitly using equation 3.10.

The matrix A in the equation 3.17 is a sparse matrix formed by the assembly of Amm, Amf and Aff matrix. In case of a rectangular mesh, the Amm matrix is pentadiagonal shaped. The shape of matrix A for structured mesh with square cells is shown in figure 3-8 below.

In an unstructured mesh, the Amm matrix does not have any defined structure. The number of entries in each row of Amm is four except in the rows for the boundary cells which has three entries. This is because except at the boundaries, each triangle is connected to three other triangles thereby sharing a common edge. The shape and number of entries in Amf and Aff depends on the location of the fractures and their connectivity. The figure 3-9 below shows the matrix A for an unstructured grid.



(a) (b)

Figure 3-8: (a) The matrix of transmissibility A for a structured mesh (60×60, with 15 fractures) with square cells. It consists of Amm, Amf and Aff matrix assembled together. The Amm matrix is pentagonal shaped while the shape of Amf and Aff depends on the location of fracture and fracture connectivity (b) The close up view of Amm matrix in a structured mesh. It is a sparse pentagonal shaped matrix, with five entries in every row (except boundary rows).

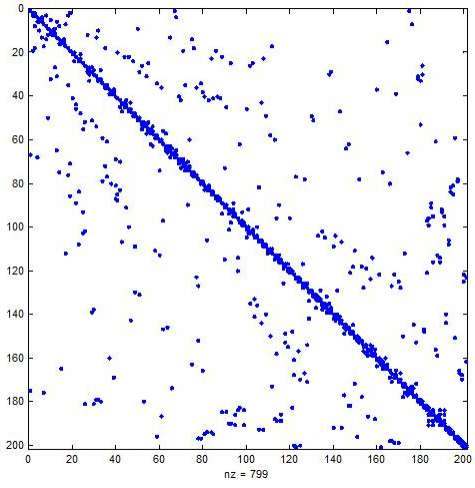


Figure 3-9: Matrix of transmissibility A for an unstructured grid. It is a sparse matrix with no well defined structure. It consists of Amm, Amf and Aff matrix assembled together. The dimension of this matrix is $N \times N$ where N is the sum of the number of Delaunay triangles in an unstructured mesh and number of fractures in the domain.

We solve pressure equation implicitly and update the transport explicitly. The general sequence for the IMPES method for two phase flow is shown in figure 3-10.

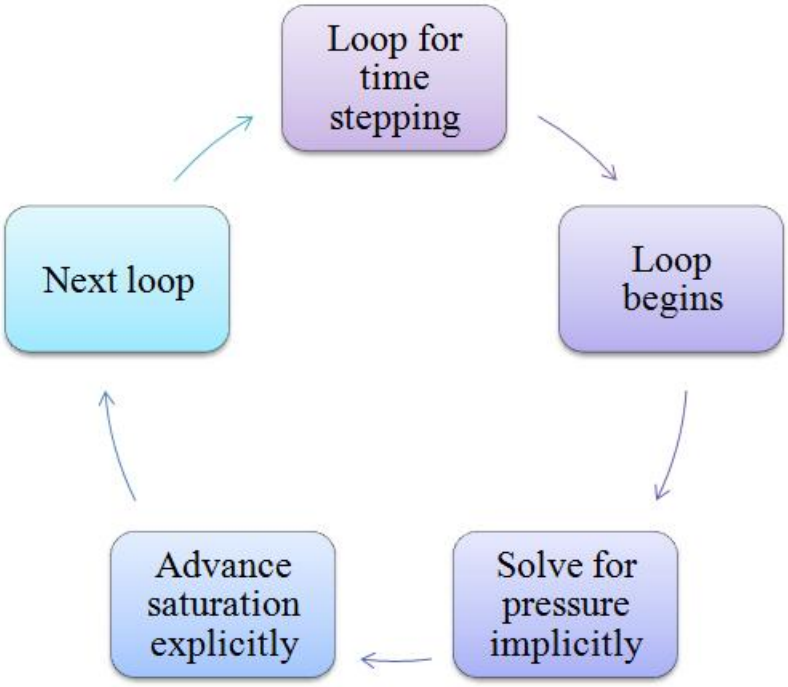


Figure 3-10: Schematic diagram of IMPES method for solving two phase flow.

We discuss the details of the implementation process in the next chapter.

Chapter 4

Implementation

In this chapter, we discuss the generation of unstructured grids and necessary implementation details for simulating two phase flow on unstructured mesh composed of Delaunay triangles.

4.1 Mesh Generation and Data Structure

For generating the unstructured mesh, we use **Distmesh** [16]. Distmesh is a simple MATLAB code for generation of unstructured triangular and tetrahedral meshes. Distmesh uses the Delaunay triangulation routine in MATLAB. The code is very efficient, simple to use and can be modified according to the needs. We made modification to the Distmesh code for fracture generation and local grid refinement. The mesh size, fracture position, domain shape are some of the basic inputs to the Distmesh function. The output of the Distmesh function is the matrix of node positions p of the Delaunay triangles and the matrix of triangles vertex t i.e. for each triangle in the mesh, the three nodes forming with the triangle is specified by the matrix t. The dimension of matrix p is $N_n \times 2$ where N_n is the number of nodes in the mesh. The dimension of the matrix t is $N_t \times 3$ where N_t is the number of Delaunay triangle in the mesh.

To solve the pressure and saturation equation, we need the connectivity information between the triangle elements. In a 2-D Delaunay mesh, each triangle is linked

with 3 adjacent triangles with the same flux across the common interface. Using the output (p, t) generated by Distmesh, we generate the data structure needed to implement the governing equations for simulating the multiphase flow on fractured media. The following is computed to generate information about the unstructured mesh from the outputted values of the Distmesh function.

- Set of neighboring triangles for each element
- Set of triangles sharing a common edge
- Boundary triangles
- Boundary edges
- Fracture edges
- Fracture triangles

We show the data structure used in the implementation of the Delaunay mesh using the following example. For simplicity, here we use a very coarse grid formed by 18 nodes, 39 edges and 22 triangles as shown in the figure 4-1. Referring figure 4-1, the edges are numbered in magenta, the node numbers are represented in blue, barycenter of the triangles and the triangle numbering is shown in green and black respectively. The black line represents the fracture in the domain, with red dots as the fracture nodes. The node numbers and triangle numbers are decided based on the values of p and t outputted by the Distmesh. The row number of the matrix p and t corresponds to the node number of the vertices and triangles respectively. The sample value of p and t for the mesh in figure 4-1 is shown in table 4.1.

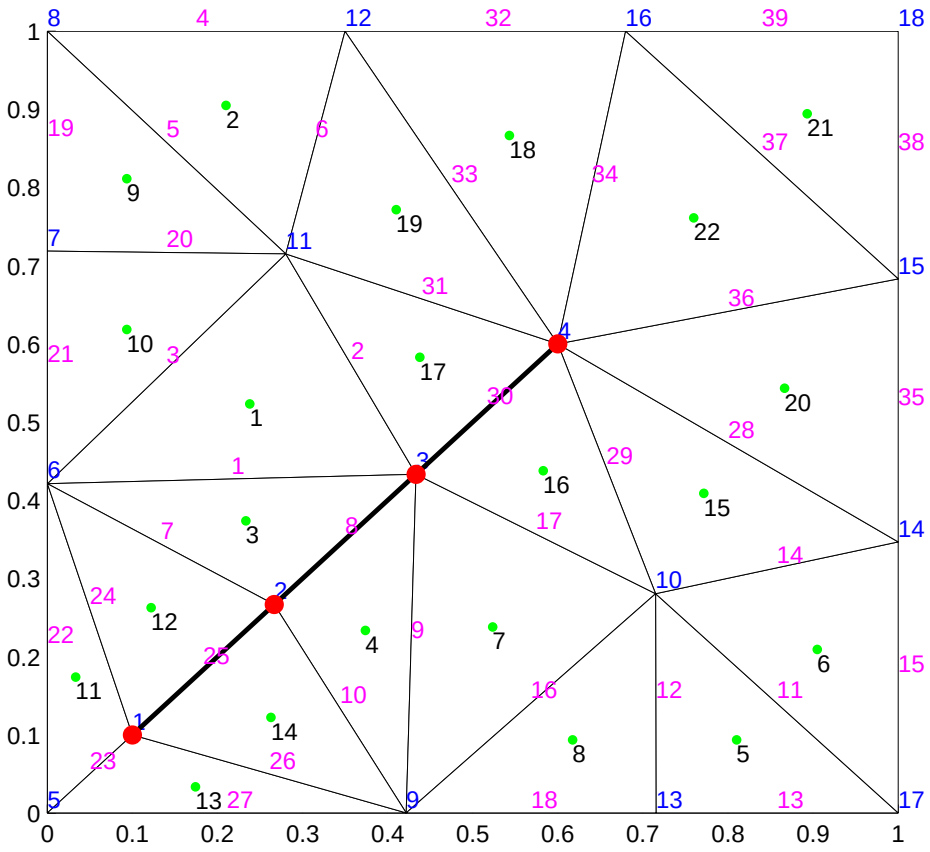


Figure 4-1: Very coarse unstructured mesh formed by Delaunay triangles with the numbering of nodes, edges and triangles required to generate the data structure for implementing multiphase flow through porous-fractured media. In this mesh, number of triangles N_t=22, number of edges $N_e = 39$ and number of nodes N_n=18. The black line represents the fracture, and the red dots indicate the fracture nodes.

(a) Matrix p

X -coordinate	Y-coordinate
0.1000	0.1000
0.2667	0.2667
0.4333	0.4333
0.6000	0.6000
0.0000	0.0000
0.0000	0.4212
0.0000	0.7188
0.0000	1.0000
0.4219	0.0000
0.7149	0.2804
0.2804	0.7153
0.3498	1.0000
0.7155	0.0000
1.0000	0.3467
1.0000	0.6833
0.6798	1.0000
1.0000	0.0000
1.0000	1.0000

(b) Matrix t

t1	t2	t3
11	6	3
11	12	8
3	6	2
2	9	3
13	17	10
17	14	10
3	9	10
10	9	13
7	11	8
6	11	7
1	6	5
1	2	6
5	9	1
9	2	1
4	10	14
3	10	4
4	11	3
4	16	12
12	11	4
15	4	14
18	16	15
16	4	15

Table 4.1: Output matrix p and t generated by Distmesh for the coarse grid shown in figure 4-1.

It can be seen from the mesh and the table, that the row number of output p corresponds to the node number with the corresponding columns specifying the x and y coordinates of that node. Similarly, any row of output t specifies the triangle number. The columns of t in any row, lists the vertices which form that triangle. The nodes are listed anticlockwise for each triangle in the matrix t. For example, the row 3 in the output t in table 4.1 gives us the information that the triangle 3 is formed by the nodes 3, 6, 2. The node numbers 3, 6, and 2 are in anticlockwise order for the triangle 3.

Using the above output, we generated the connectivity list. We coded to extract the information about the triangle to triangle connectivity, triangle to edge connectivity and fracture-triangle connectivity. Also, we distinguished the boundary edges, boundary triangles and fracture edges using the connectivity list. The connectivity list for triangle to triangle connectivity, triangle to edge connectivity and fracture-triangle connectivity for the coarse mesh in figure 4-1 is shown in the table 4.2, table 4.3 and table 4.4 respectively.

Triangle to Triangle connectivity list: As observed in the table 4.2, the size of triangle to triangle matrix is $N_t \times 3$. Each row of the table denotes the triangle number, and each column in a row indicates the connecting triangle number. For example, the row 4 in the list gives us the information that the triangle 4 is connected to triangles 7, 3 and 14 on it's three side. In other words, triangle 4 shares a common edge with each of the triangle 7, 3 and 14. The connecting triangles are listed anticlockwise for each triangle. The presence of a negative number in a row indicates that the triangle has a boundary edge. For instance, the triangle 2 (row 2) has columns with numbers -1, 9, 19. It implies that triangle 2 is surrounded by triangles 9 and 19 and has one boundary edge.

Triangle 1	Traingle 2	Triangle 3
3	17	10
-1	9	19
12	4	1
7	3	14
6	8	-1
15	5	-1
8	16	4
-1	5	7
2	-1	10
9	-1	1
-1	13	12
3	11	14
14	11	-1
12	13	4
6	20	16
15	17	7
1	16	19
-1	19	22
17	18	2
15	-1	22
22	-1	-1
20	21	18

Table 4.2: Triangle to Triangle connectivity list.

Triangle to Edge connectivity list: Table 4.3 shows the triangle to edge connectivity list. The size of this matrix is $N_e \times 4$, where N_e is the total number of triangle edges in the domain. The row number is related to the edge number. The first two columns in the list are the node numbers forming an edge in the domain and the last two columns in the list are the triangles sharing that edge. For example, the 6^{th} edge (denoted by 6^{th} row) is formed by node 11 and 12, and is common between triangle 2 and triangle 19. The negative number in the last column in any row implies that the particular edge is a boundary edge and belongs to the triangle specified in column 3. The values -1, -2, -3, -4 corresponds to the bottom, right, top, and left boundary edge of the domain respectively. For instance, the 39^{th} edge (last row) formed by nodes 18 and 16 belongs to triangle 21 and lies on the top boundary (-3) of the domain (refer figure 4-1).

Node 1	Node 2	Triangle 1	Triangle 2
6	3	1	3
3	11	1	17
11	6	1	10
12	8	2	-3
8	11	2	9
11	12	2	19
6	2	3	12
2	3	3	4
9	3	4	7
2	9	4	14
17	10	5	6
10	13	5	8
13	17	5	-1
14	10	6	15
17	14	6	-2
9	10	7	8
10	3	7	16
9	13	8	-1
8	7	9	-4
7	11	9	1
7	6	10	-4
6	5	11	-4
5	1	11	13
1	6	11	12
1	2	12	14
9	1	13	14
5	9	13	-1
14	4	15	20
4	10	15	16
4	3	16	17
4	11	17	19
16	12	18	-3
12	4	18	19
4	16	18	22
14	15	20	-2
15	4	20	22
16	15	21	22
15	18	21	-2
18	16	21	-3

Table 4.3: Triangle to Edge connectivity list.

Fracture-Triangle connectivity list: It is important to identify the edges which form the fractures and the triangles which are connected to the fractures. The fracture connectivity list shown in the table 4.4 provides the required details.

Fracture node 1	Fracture node 2	Traingle 1	Triangle 2
1	2	12	14
2	3	3	4
4	3	16	17

Table 4.4: Fracture - Triangle connectivity list.

The first two column represent the nodes which constitutes a fracture edge. The last two columns are the triangle numbers which are connected to the fracture edge. For example, in the table 4.4, the nodes 1 and 2 form a fracture edge, and it is connected to triangles 12 and 14 (refer figure 4-1).

4.2 Solution Steps

Once the data structure is formed, IMPES method is used to solve for pressure and saturation in multiphase flow on an unstructured mesh. The main steps are outlined below.

Required input to the simulator

Mesh geometry, Permeability field, Porosity, Mobility ratio

Desired output from the simulator

Saturation at any desired time

Basic Solution Steps

- Loop starts for the time integration
- Solve for matrix and fracture pressure implicitly as described in section 3.3
- Loop starts for number edges in the domain

- Determine nodes i, j forming the an edge l in the loop
- Extract the outward normal n_{ij} to edge l
- Use single point upwind method to calculate the flux through the edge in the matrix (or fracture if edge l is a fracture edge)
- Add and subtract the flux contribution in the adjoining control volumes based on upwind direction
- Loop ends for number of edges in the domain
- Update the saturation explicitly for the matrix and the fracture control volumes
- Loop ends for the time integration

In next chapter, we show the simulation results of multiphase flow through fractured-porous media on an unstructured and structured mesh for various cases.

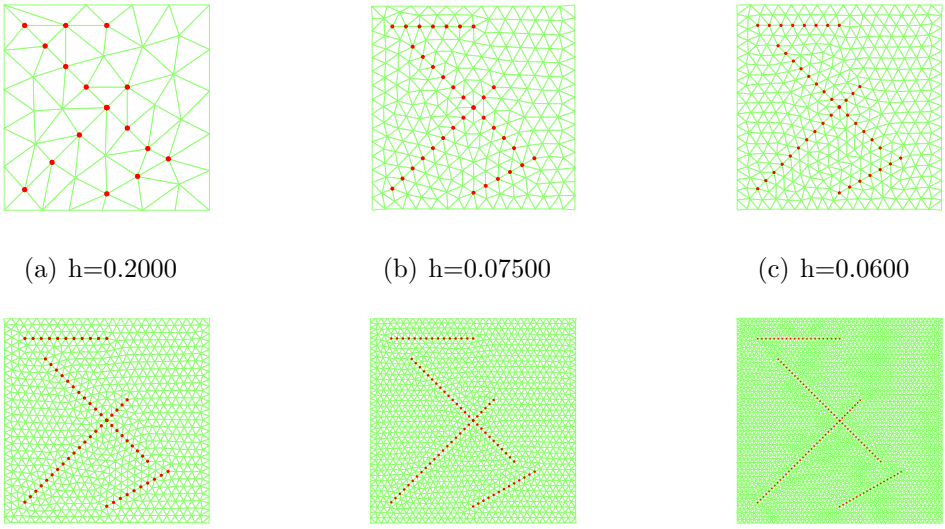
Chapter 5

Numerical Simulations

In this chapter, we illustrate the performance of the unstructured multiphase finite volume simulator on various synthetic cases of increasing complexity. We restrict ourselves to two dimensional problems on structured and unstructured grids.

5.1 Convergence Analysis

To examine the accuracy of our solution, we conduct a convergence analysis using a reference solution as there is no standard analytical solution for multiphase flow in fractured media. The reference solution is the solution obtained using a very fine mesh and is assumed to be the true solution. As the degree of grid refinement increases, the solution obtained at each level of grid refinement should approach the reference solution. Therefore, error should decrease with increasing mesh refinement. We consider various grids with increasing level of refinement as show in figure 5-1. The reference solution is the solution obtained on the mesh 5-1(f).



(a) h=0.2000 (b) h=0.07500 (c) h=0.0600

(d) h=0.0400 (e) h=0.0300 (f) h=0.0200

Figure 5-1: Unstructured mesh with varying degree of grid refinement. h denotes the edge size of a triangular element in the mesh.

We use a quarter-five spot simulation for convergence study. The quarter five-spot is a pattern in which a injector and a producer is located at diagonally opposite vertices of a grid and the boundaries of the domain is a no flow boundary (figure 5-2). It is assumed that the domain is initially filled with oil and water is injected at the left corner to produce oil at the opposite corner. We did quarter five spot simulations on the meshes shown in figure 5-1 to plot the breakthrough curve at the producer. The breakthrough curves are shown in figure 5-3.

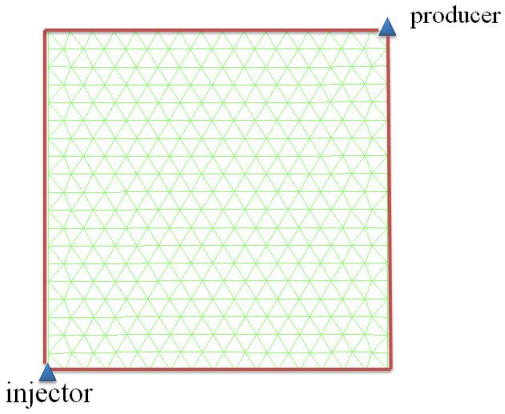


Figure 5-2: Quarter five-spot model with diagonally located injector and producer. The red lines indicate the no flow boundaries in the domain.

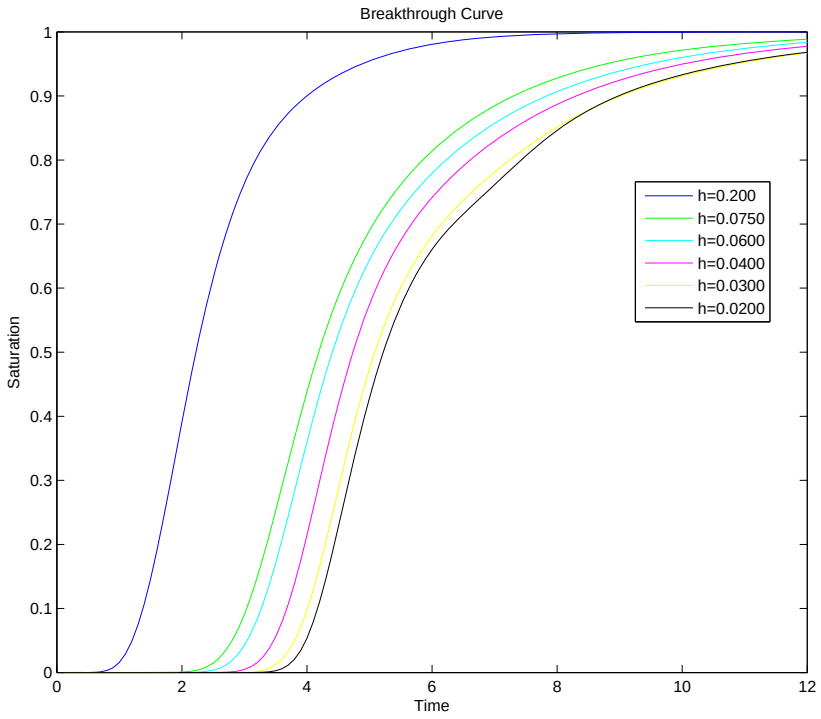


Figure 5-3: The breakthrough curve at the producer using quarter five spot simulation for various unstructured meshes shown figure 5-1.

It can be observed from the breakthrough curve that as the mesh size h is decreased, the solution obtained approaches the reference solution. It implies that the solution converges to reference solution as the mesh refinement increases. If the exact analytical solution was known, then the solution obtained using a fine mesh will converge to the exact solution.

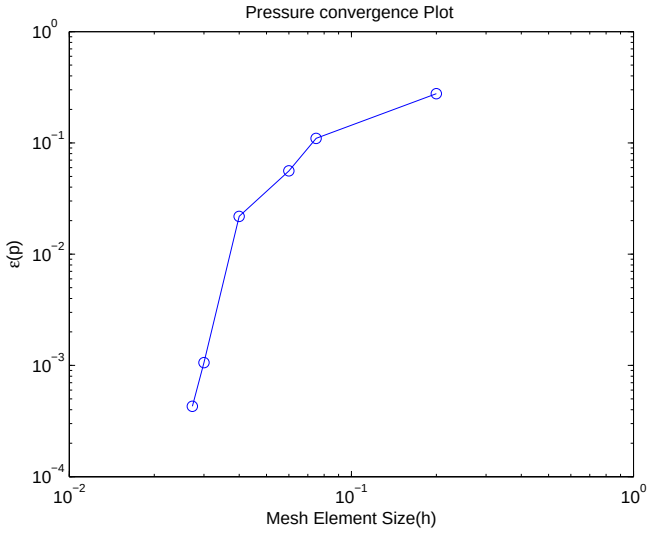
We compute the mean saturation error as follows :

$$\epsilon(s) = \frac{\|s_{ref} - s\|^2}{\|s_{ref}\|^2} \tag{5.1}$$

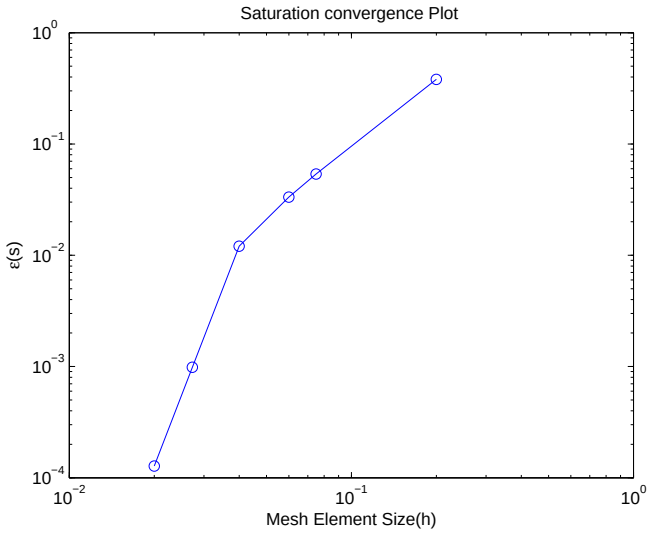
where s_{ref} and s are the vectors containing the saturation values at the producer at any time using reference and non-reference mesh respectively. The $\|\cdot\|$ is the usual l^2 norm. The mean pressure error is computed as

$$\epsilon(p) = \frac{\|p_{ref} - p\|^2}{\|p_{ref}\|^2} \tag{5.2}$$

where p_{ref} and p are the vectors containing the pressure values at the producer on reference and non-reference mesh respectively. The figure 5-4 shows the log-log plot of mean pressure and saturation error in relation with the mesh element size. As



(a)



(b)

Figure 5-4: Log-Log plot for pressure and saturation solution with respect to the reference solution.

observed from the figure, the error decreases as the mesh is refined. The source of error is the numerical diffusion, the positional error in the centroid of control volume at the producer in different meshes and time integration error (in case of transport

advancement). Nevertheless, we can conclude that if the exact solution was known, then the solution obtained on fine mesh would converge to the exact solution using finite volume scheme. In the next section, we show some examples of numerical simulations on various synthetic cases on an unstructured and structured mesh.

5.2 Quarter Five-Spot Simulations on Unstructured Mesh

We present various simulation examples on an unstructured mesh in this section. Before illustrating the examples, we discuss about local grid refinement for simulating the multiphase flow in fractured media.

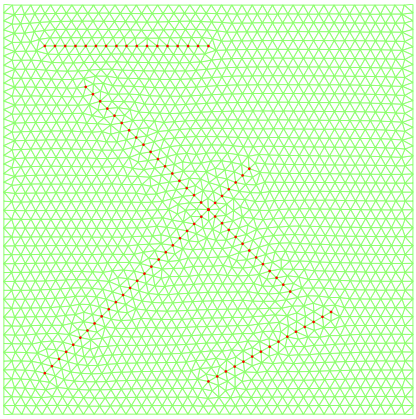
5.2.1 Local Grid Refinement

The specific requirement of simulating the fluid flow in the fractured domain is to have more detail (fine grid) in the vicinity of the fractures to minimize error caused by the high contrast in rock matrix and fracture properties. This can be done by using the conventional refined grid in the whole domain. In practice, choosing a sufficiently fine uniform grid with result in an accurate solution. Unfortunately, this approach is rather time and memory intensive. Local grid refinement around the fracture can resolve this issue by locally refining the the grids around the fracture without extending it the simulation boundaries. The computational grid become coarser farther away from the fractures. This approach also gives accurate solution and is very effective with respect to computer memory and computing speed.

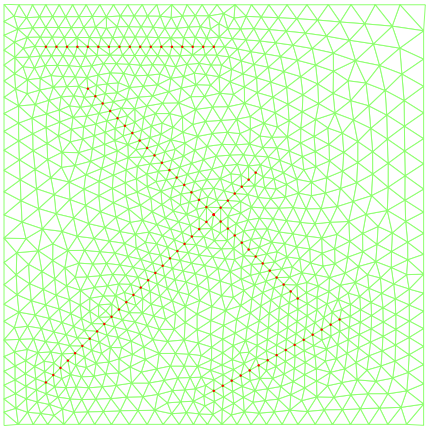
The figure 5-5 shows the two type of grids.

1. Conventional fine grid mesh : The mesh consists of 3186 triangles with all mesh element have edge size of $h = 0.0250$.

2. Locally refined grid mesh : The mesh consists of 2164 triangles with the smallest fine mesh element having edge size of $h = 0.0250$.



(a) h=0.0250



(b) h=0.0250

Figure 5-5: (a) Conventional fine grid mesh (b) Locally refined mesh with fine grids in the vicinity of the fracture and coarsening away from the fracture.

We observe that number of mesh elements used in case of locally refined mesh is much smaller than the conventional fine grid mesh. The matrix formed to solve the system in locally refined grid is smaller than that resulting from the conventional refined grid. Therefore, summarizing the advantages of using locally refined grid over the conventional grid system

1. Reduction in the number of elements necessary to properly model the reservoir domain containing fractures, implying more effectiveness in terms of memory and computational cost.

2. Improvement of accuracy of solution in fractured regions or regions of interest.

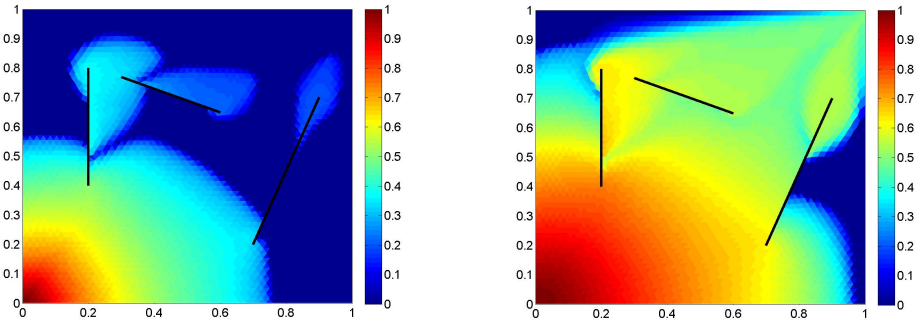
5.2.2 Homogeneous Domain

In this section, we show the results of the multiphase finite volume simulator in a domain with homogeneous isotropic permeability distribution consisting of discrete and intersecting fractures.The fractures are considered to have permeability of order 10^{-5}.

Figure 5-6(a) and figure 5-6(b) shows the flow profile of the two phase flow in a homogeneous domain with discrete fractures. We have used a conventional fine grid for this simulation. The mesh element size(h) is 0.0200 and number of triangular elements is 5777. It is clearly seen in the figure that fractures, if present in the fluid path are main flow conduits.

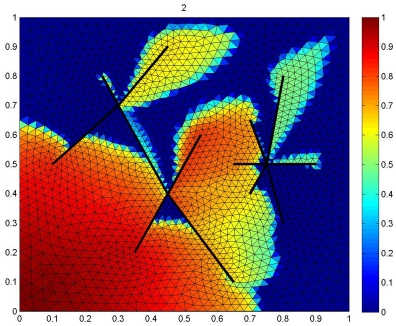
Figure 5-6(c) and figure 5-6(d) shows the domain with intersecting fractures. We use unstructured mesh with local grid refinement for computational efficiency. The smallest mesh element size(h) is 0.0200 and number of triangular elements is 3647. The presence of intersecting fracture network results in early breakthrough at the producer.

Figure 5-6(c) and figure 5-6(d) shows the homogeneous domain consisting of both the discrete and intersecting fractures on unstructured mesh with local refinement. The number of triangular element is 3647 with smallest mesh size(h) of 0.0200.

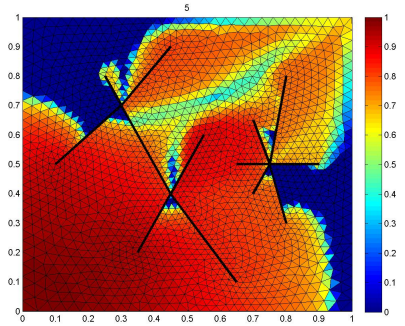


(a)

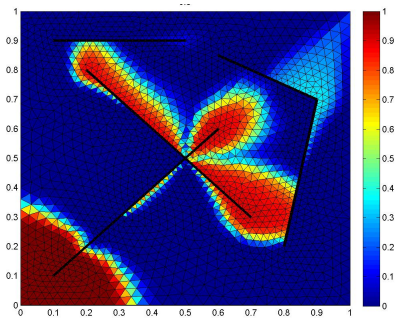
(b)



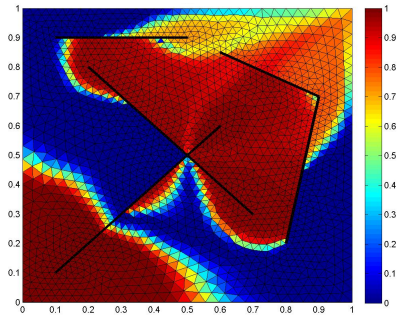
(c)



(d)



(e)



(f)

Figure 5-6: Snapshot of multiphase flow in a fractured domain with homogeneous permeability distribution using an unstructured mesh.

5.2.3 Heterogeneous Domain

We illustrate the simulation of the multiphase flow in a heterogeneous domain with isotropic random permeability distribution as shown the figure 5-7. The fracture permeability is 10^{-5}.

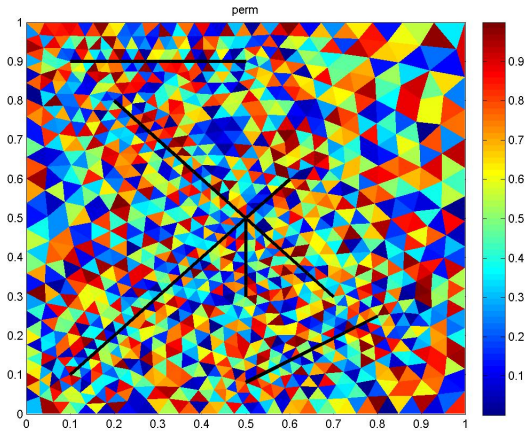
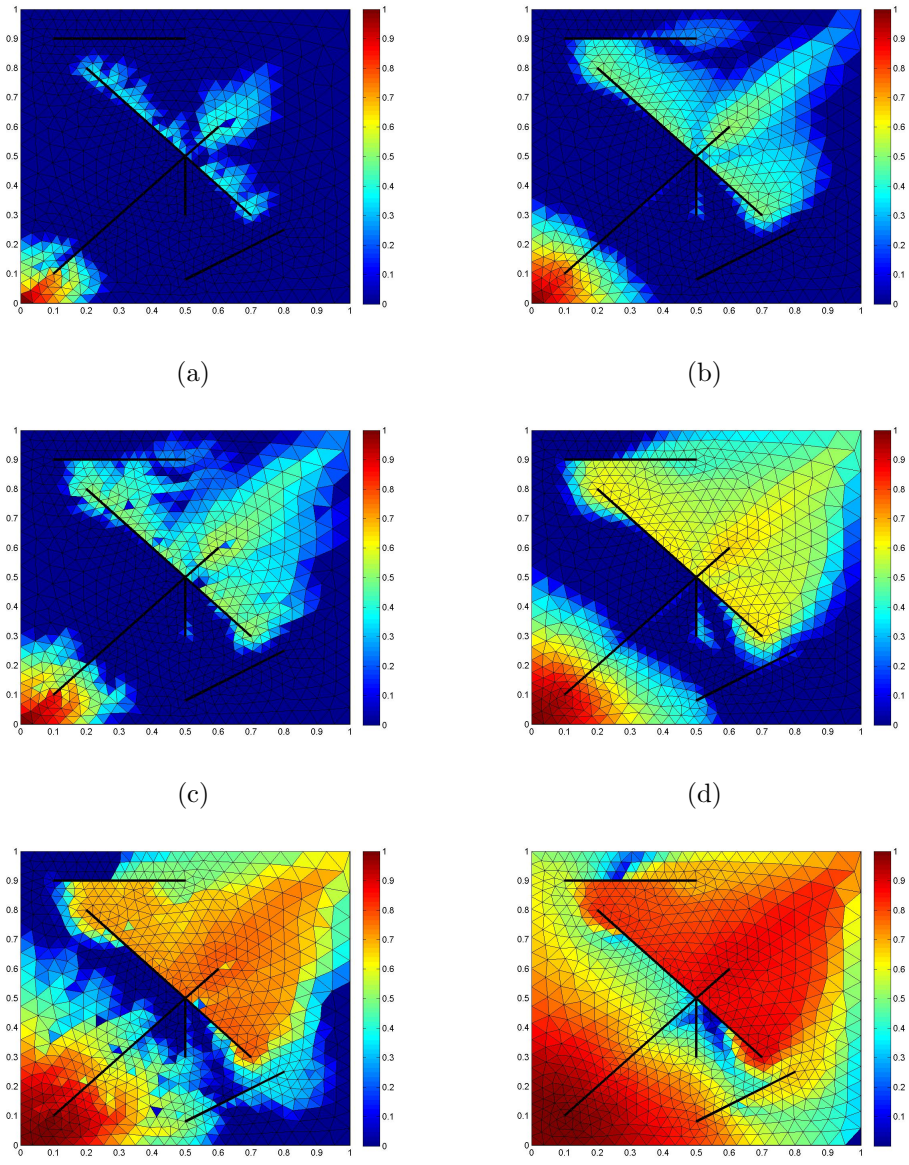


Figure 5-7: Random permeability distribution.

The figure 5-8 shows the comparison between the flow in homogeneous and heterogeneous domain. We have used a locally refined unstructured mesh with element size(h) equal to 0.0300. The number of triangles in the mesh is 1562. The left hand side of the figure 5-8 shows the simulation result of the multiphase flow in a heterogeneous domain while the right hand side shows the simulation results in a homogeneous domain at the same dimensionless times. The homogeneous domain has a uniform permeability of one while heterogeneous medium has range of permeabilities ranging from 0.1 to 1. We observe that the fluid flow in the homogeneous domain is more uniform while in the heterogeneous case, the flow is guided by the permeability distribution in the domain.

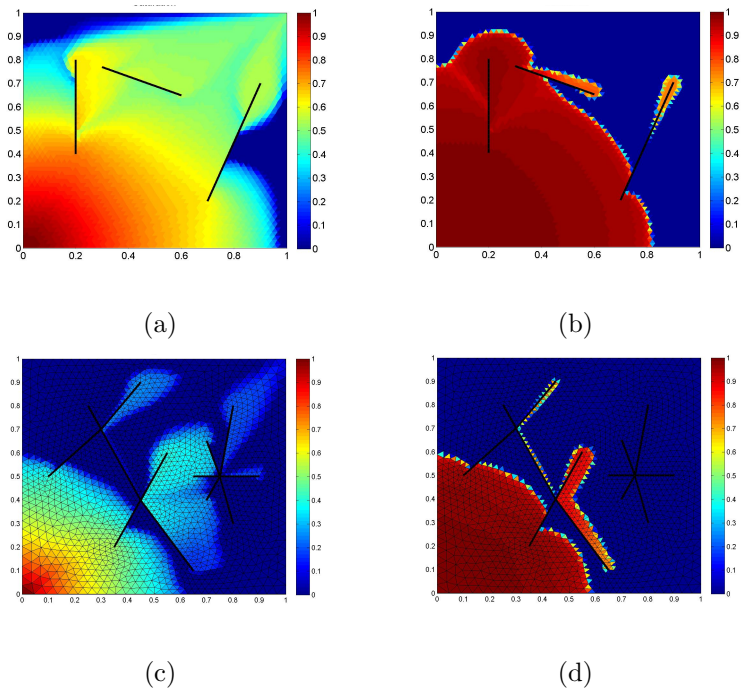


(a) (b) (c) (d) (e) (f)

Figure 5-8: Simulation result of multiphase flow on an unstructured mesh in a fractured domain, at same dimensionless times, with heterogeneous (left) and homogeneous (right) permeability distribution.

5.2.4 Effect of Viscosity Ratio

The fluid flow pattern in the two phase flow is affected by the viscosity ratio between the two phases. If the displacing fluid (water) is more viscous than the displaced fluid (oil) then the fluid interface between the fluid is stable. In the other case, if the displacing fluid is less viscous than the displaced fluid, the fluid interface is unstable causing fingering. Due to numerical diffusion, we do not capture the finger formation. However, in the figure 5-9, we see the effect of viscosity ratio on two phase flow with different fracture configurations at the same dimensionless time(for each fracture configuration). The left hand side of the figure has the viscosity ratio of oil to water as 10 whereas the right hand side has viscosity ratio of water to oil as 10. The right hand side of the figure has more stable interface and moves slowly. The sweep is more efficient in the case where the viscosity ratio of the displacing fluid is greater than the displaced fluid.



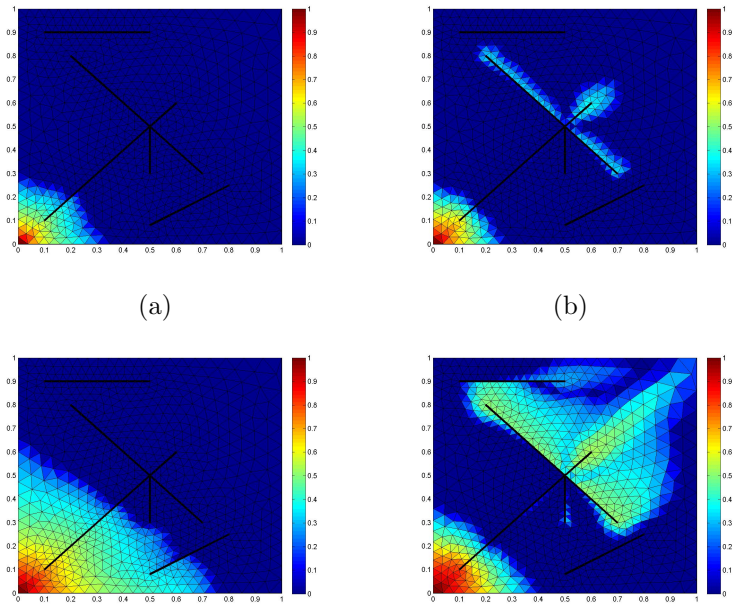
(a) (b)

(c) (d)

Figure 5-9: Simulation results showing the effect of viscosity ratio on two phase flow. In the left hand side figures, the viscosity ratio of oil to water is ten while in the right hand side figures, the viscosity ratio of water to oil is ten. More stable fluid interface and better sweep efficiency is observed in right hand side figures where the viscosity of displacing fluid is greater than that of the displaced fluid.

5.2.5 Impermeable Fractures

The left hand side of the figure 5-10 demonstrates the simulation result with impermeable fractures in the domain. The right hand side of the figure 5-10 shows simulation results in the domain with same fracture network containing permeable fractures. At same dimensionless times, we observe that in case of impermeable fracture, the flow detours around the fracture network. The impermeable fracture act as a barrier to the flow path (figure 5-10(c) and figure 5-10(a)). On the other hand, the permeable fracture acts as flow conduit and allows the passage of the fluid through it resulting in early breakthrough at the producer (figure 5-10(c)).



(a) (b)

(c) (d)

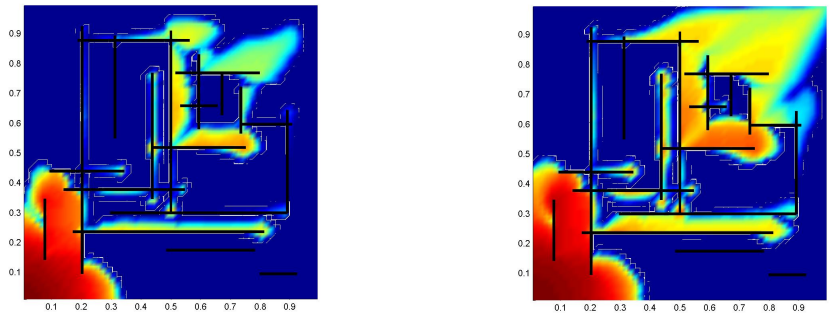
Figure 5-10: Simulation results with impermeable and permeable fractures in the domain. The left hand side of the figure shows a fracture network with impermeable fractures while the right hand side shows the same fracture network with permeable fractures. In case of impermeable fractures,the flow moves around the fracture while in case of permeable fractures, the flow moves through the fractures resulting in early breakthrough at the producer.

5.3 Quarter Five-Spot Simulations on Structured Mesh

In this section, we illustrate few examples of multiphase flow simulation using finite volume scheme on structured mesh. In all the examples shown below, the mesh is formed by square cells and has dimension 64×64. The fractures are generated stochastically in all the cases. We vary the permeability of the domain in the examples, while considering the fractures to be highly permeable fluid flow conduit with isotropic permeability of order 10^{-5}.

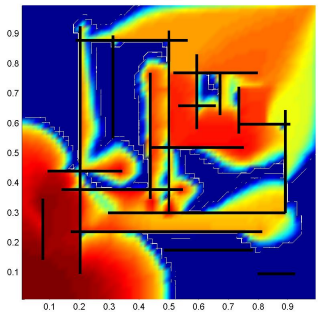
5.3.1 Homogeneous Domain

We consider the unit square domain with uniform isotropic permeability distribution of one. The saturation profiles at different dimensionless time is shown in the figure 5-11. As soon as the fluid encounters a fracture in flow path , it flows through the interconnected permeable fracture network. Eventually, the saturation reaches unity in the entire domain. Hence, an early breakthrough results due to the presence of permeable fracture in the flow path.

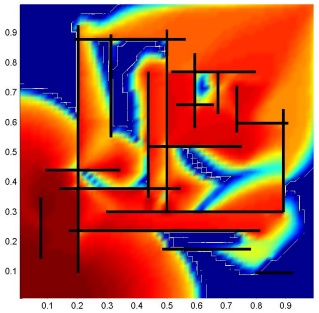


(a)

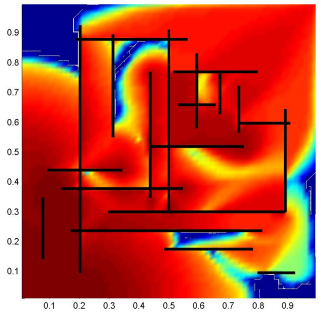
(b)



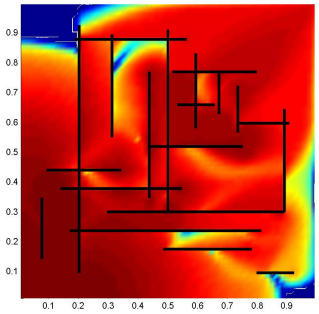
(c)



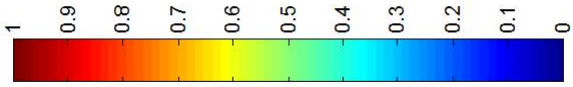
(d)



(e)



(f)



(g) colorbar

Figure 5-11: Saturation profile of multiphase flow in a fractured domain with isotropic and homogeneous permeability distribution at different times.

5.3.2 Heterogeneous Domain with Horizontal Channels

We illustrate the performance of the finite volume multiphase simulator in the domain with the permeability distribution shown in figure 5-12.

The simulation result is shown in figure 5-13. The fluid flows through the connected fracture network and emerges out in the parts of domain with high permeability. Eventually, the fluid migrates in the other region of the domain. After a long period of time, the entire domain is saturated with the water.

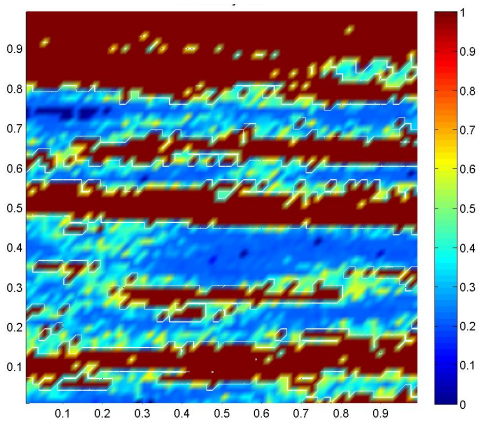


Figure 5-12: Permeability distribution in horizontal channels.

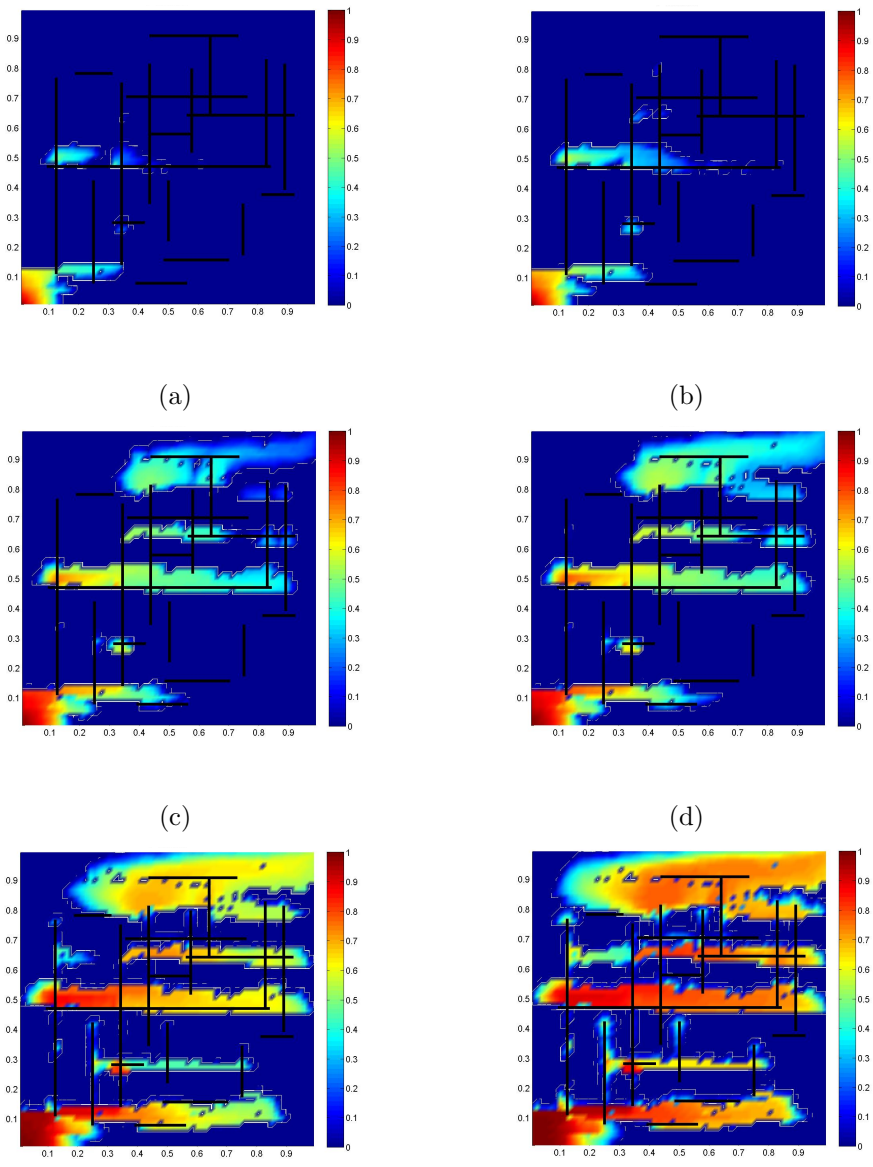


Figure 5-13: Snapshot of multiphase flow in a fractured domain with permeability distribution in horizontal channels (shown in figure 5-12).

5.3.3 Heterogeneous Domain with Diagonal Channels

This example shows the results obtained by finite volume multiphase simulator in the domain with isotropic and heterogeneous permeability distribution in diagonal channels as shown in figure 5-14.

Figure 5-15 shows the fluid saturation in the domain at different times. The fluid flow is diagonal indicating that the fluid flows preferentially in the direction of high permeability zone (figure 5-15(a)). In case of the presence of a fracture in the flow path, the fluid flows through the fracture network, erupts out in the matrix and starts flowing preferentially in the high permeability direction(figure 5-15(e) and figure 5-15(f))

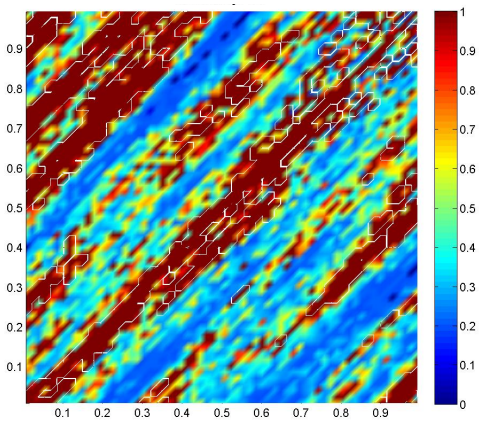
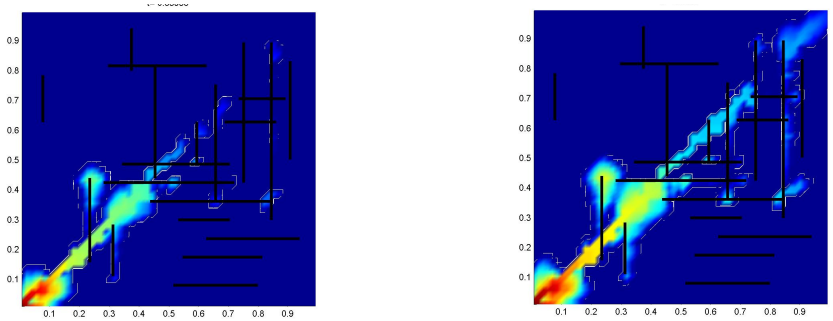
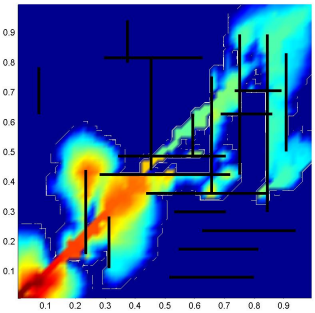


Figure 5-14: Permeability distribution in horizontal channels.

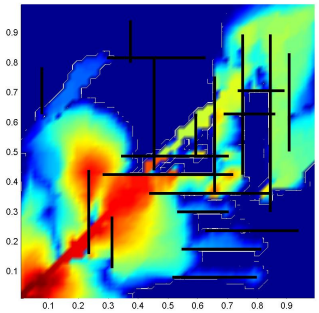


(a)

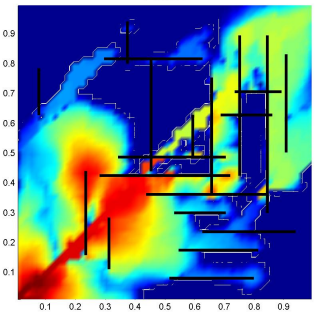
(b)



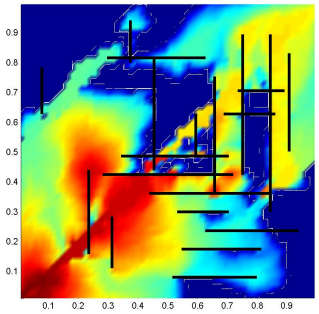
(c)



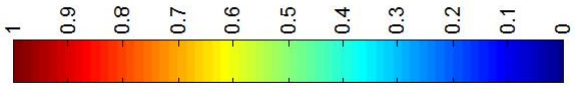
(d)



(e)



(f)



(g) colorbar

Figure 5-15: Snapshot of multiphase flow in a fractured domain with diagonal permeability distribution (figure 5-14).

5.3.4 Heterogeneous Domain with Log-Normal Permeability Distribution

The figure 5-16 shows the isotropic and log-normal permeability distribution. The figure 5-17 shows the simulation snapshot in the domain with log normal permeability distribution at various dimensionless times.

Clearly it is seen that the flow is preferentially either through the fracture network (if present) or in the high permeability zones in the domain. The fluid saturation rises in the regions of high permeability zones initially and spreads out eventually in other low permeability regions.

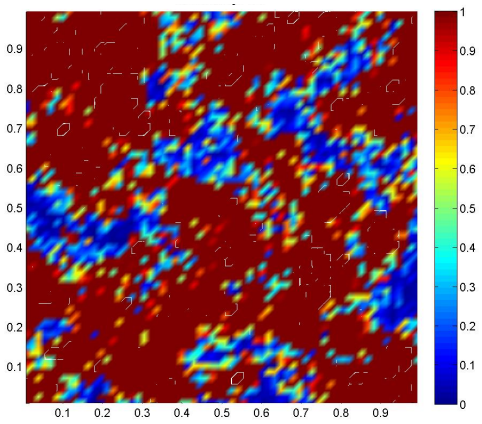
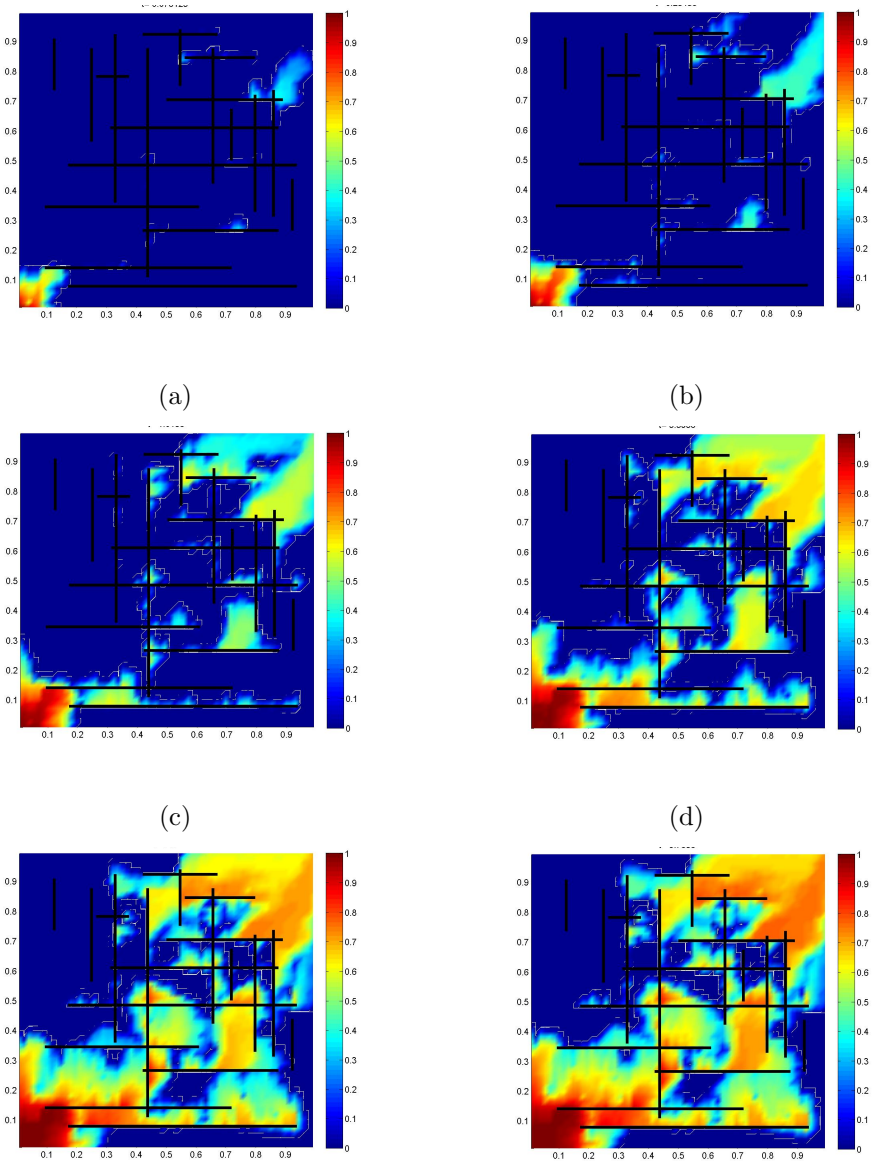


Figure 5-16: Log-normal permeability distribution.

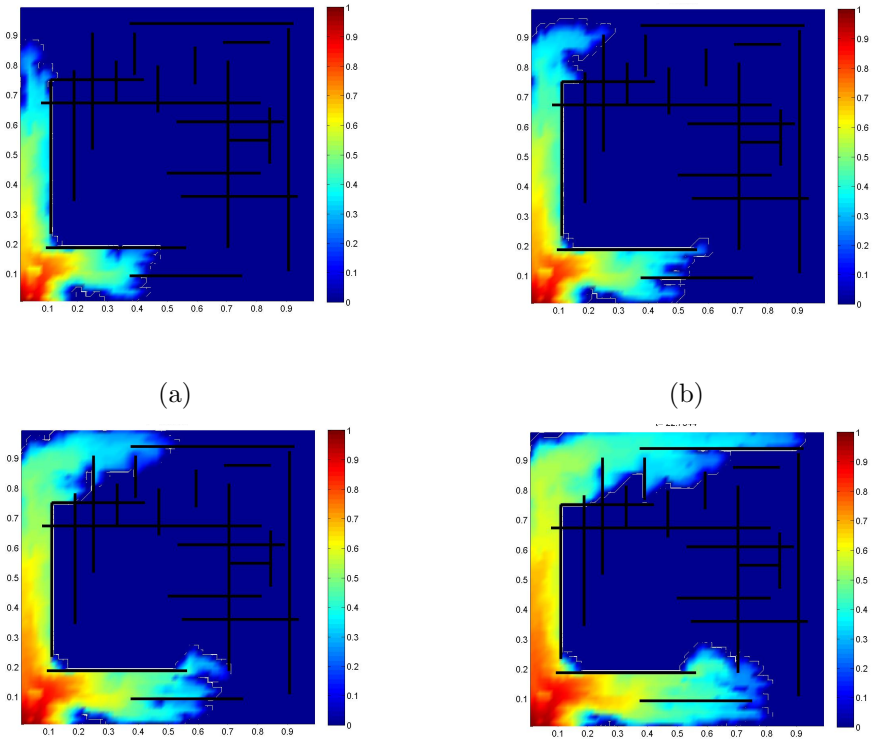


(a) (b) (c) (d) (e) (f)

Figure 5-17: Snapshot of multiphase flow in a fractured domain with isotropic and log-normal permeability distribution(figure 5-16).

5.3.5 Impermeable Fractures

The figure 5-18 consists of impermeable fractures in a domain with log normal permeability distribution. It is observed that the impermeable fractures act as flow barriers causing fluids to move around the fractures to flow through the matrix.



(a) (b)

(c) (d)

Figure 5-18: Simulation results for multiphase flow in a domain consisting of impermeable fracture network. The impermeable fractures act as barrier to the flow.

Chapter 6

Conclusion

The key ingredients of the model used for simulating multiphase flow through porous-fractured domain as described in this thesis are as follows :

- The mass conservative finite volume method is used to simulate the multiphase flow in fractured media using discrete fracture model on an unstructured and structured mesh. The model is applicable for 2-D systems and accounts for matrix-matrix, matrix-fracture, and fracture-fracture flow.

- The use of unstructured mesh allows us to resolve the complex fracture geometry accurately compared to the structured mesh. Additionally, an unstructured mesh with local grid refinement provides greater computational efficiency.

- The specialized star-delta transformation described by [9] was used to eliminate the control volumes at the fracture intersections, for improved computational efficiency and numerical stability.

- The convergence analysis and numerical simulation examples in the previous section demonstrate that the finite volume method provides accurate solution for multiphase flow in fractured media with different set of properties on a fine grid. Therefore, we can conclude that this finite volume simulator for multiphase flow is an effective and reliable tool for validating the macroscopic flow models.

The possible extensions of interest to this finite volume multiphase model can be

- Extending this 2-D finite volume multiphase flow simulator to handle 3-D mesh.
- Accounting for the effect of capillarity and gravity in the multiphase flow.
- Using multi point flux approximation instead of two point flux approximation for solving the pressure equation with improved accuracy.
- Using more advance numerical methods instead of IMPES to solve the governing equations. This will avoid the restriction due to CFL condition on time integration, and will be computationally more stable.

Bibliography

[1] J.E. Aarnes and V.L. Hauge. Modeling of two-phase flow in fractured porous media on unstructured non-uniformly coarsened grids. *Transport In Porous Media*, 77:373–398, 2009.

[2] K. Aziz and A. Settari. *Petroleum Reservoir Simulation*. Elsevier Applied Science Publisher, March 2002.

[3] R. G. Baca, R. C. Arnett, and D. W. Langford. Modelling fluid flow in fractured-porous rock masses by finite-element techniques. *International Journal for Numerical Methods in Fluids*, 4:337–348, 1984.

[4] G.I. Barrenblatt, I.P. Zheltov, and I.N. Kochina. Basic concepts in the theory of seepage of homogeneous liquids in fissured rocks. *Journal of Applied Mathematical Mechanics*, 24:12861303, 1960.

[5] J. Bear. *Dynamics of Fluids in Porous Media*. American Elsevier, New York, 1972.

[6] L.J. Durlofsky. Accuracy of mixed and control volume finite element approximations to darcy velocity and related quantities. *Water Resources Research*, 30:965–973, April 1994.

[7] S. Granet, P. Fabrie, P. Lemonnier, and M. Quitard. A two-phase flow simulation of a fractures reservoir using a new fissure element method. *Journal of Petroleum Science and Engineering*, 15:35–52, 2001.

[8] R. Juanes, J. Samper, and J. Molinero. A general and efficient formulation of fractures and boundary conditions in the finite element method. *International Journal for Numerical Methods in Engineering*, 54:1751 – 1774, 2002.

[9] M. Karimi-Fard, L.J. Durlofsky, and K. Aziz. An efficient discrete-fracture model applicable for general-purpose reservoir simulators. *SPE Journal*, 9:227–236, 2004.

[10] M. Karimi-Fard and A. Firoozabadi. Numerical simulation of water injection in 2d fractured media using discrete-fracture model. In *SPE Annual Technical Conference and Exhibition*, number 71615-MS, 2001.

[11] H. Kazemi. Pressure transient analysis of naturally fractured reservoirs, with uniform fracture distribution. *SPE Journal*, 9:451–462, 1969.

[12] J. Kim and M.D. Deo. Comparison of the performance of a discrete fracture multiphase model with those using conventional methods. *AIChE Journal*, 46:1120–1130, 2000.

[13] S.F. Matringe, R. Juanes, and H.A. Tchelepi. Robust streamline tracing for the simulation of porous media flow on general triangular and quadrilateral grids. *Journal of Computational Physics*, 219:992–1012, Dec 2006.

[14] J.E.P. Monteagudo and A. Firoozabadi. Control-volume method for numerical simulation of two-phase immiscible flow in two- and three-dimensional discrete-fractured media. *Water Resources Research*, 40:W07405, 2004.

[15] J. Noorishad and M. Mehran. An upstream finite element method for solution of transient transport equation in fractured porous media. *Water Resources Research*, 18:588–596, 1982.

[16] P.-O. Persson and G. Strang. A simple mesh generator in Matlab. *SIAM Review*, 46:329–345, June 2004.

[17] B. Riviere, M.F. Wheeler, and K. Banas. Discontinuous Galerkin method applied to a single phase flow in porous media. *Computational Geosciences*, 4:337–349, 2004.

[18] K. Slough. Numerical simulation of multiphase flow and phase partitioning in discretely fractured geologic media. *Journal of Contaminant Hydrology*, 40:107–136, 1999.

[19] K.J. Slough, E.A. Sudicky, and P.A Forsyth. Grid refinement for modeling multiphase flow in discretely fractured porous media. *Advances In Water Resources*, 23:261–269, 1999.

[20] J.E. Warren and P.J. Root. The behavior of naturally fractured reservoirs. *SPE Journal*, 3:245–255, 1963.